



# Universidad **Mariana**

Accesibilidad aplicada a la creación de contenidos digitales para personas con discapacidad  
visual

Carlos Alberto Botina Carpio

Universidad Mariana  
Facultad de Ingeniería  
Programa de Ingeniería de Sistemas  
San Juan de Pasto  
2024

Accesibilidad aplicada a la creación de contenidos digitales para personas con discapacidad  
visual

Carlos Alberto Botina Carpio

Trabajo de grado como requisito para obtener el título de Ingeniero de  
Sistemas

Sandro Fabián Parra Pay

Asesor

Universidad Mariana  
Facultad de Ingeniería  
Programa de Ingeniería de Sistemas  
San Juan de Pasto  
2024

Artículo 71: los conceptos, afirmaciones y opiniones emitidos en el Trabajo de Grado son responsabilidad única y exclusiva del (los) Educando (s)

Reglamento de Investigaciones y Publicaciones, 2007  
Universidad Mariana

## **Agradecimientos**

Se extienden sinceros agradecimientos a todas las instituciones y personas que desempeñaron un papel fundamental en la realización de este proyecto. En primer lugar, se agradece al programa de Mercadeo de la Universidad Mariana, y en particular a los docentes Edgar Mauricio Salas Leiva y Lorena Tatiana Luna Tobar, por su constante apoyo y orientación a lo largo de la investigación. Asimismo, se reconoce la colaboración y participación del equipo de la fundación FUNDONAR en la ciudad de Pasto, con especial mención a la Dra. Yuli Carolina Delgado Solarte. Además, se valora enormemente la contribución de los expertos en accesibilidad y tecnologías móviles, cuya experiencia y conocimientos enriquecieron significativamente el proyecto. Por último, se agradece a todos los usuarios y participantes que dedicaron su tiempo y proporcionaron sus comentarios para probar y mejorar la aplicación desarrollada. Sin la generosa contribución de todos los mencionados, este proyecto no habría sido posible.

## **Dedicatoria**

Con amor y gratitud, dedico este proyecto a mis padres, quienes han sido mi mayor apoyo a lo largo de este viaje académico. Su constante aliento, sabiduría y amor han sido el pilar fundamental que me ha impulsado a alcanzar mis metas. Gracias por haberme inculcado los valores que me han convertido en la persona que soy hoy en día. Su ejemplo de integridad, generosidad y sacrificio ha sido una guía constante en mi vida, inspirándome a esforzarme por alcanzar mis metas y superar mis desafíos. Cada paso en este proyecto es un reflejo de los principios que me han enseñado y el amor incondicional que siempre me han brindado. A ustedes les debo mis logros.

Carlos Alberto Botina Carpio

## Contenido

Introducción .....	16
1. Elementos del proceso.....	20
1.1. Estado de la cuestión .....	20
1.1.1. Blind Explorer .....	20
1.1.2. Voice Guided Navigation for visually impaired.....	21
1.1.3. BARD Mobile.....	22
1.1.4. Ballyland Code .....	23
1.1.5. Accessible 3D Audio Maze Game.....	24
1.1.6. Sensors-plus example .....	26
1.2. Título .....	27
1.3. Problema de Investigación .....	27
1.3.1. Descripción del problema.....	27
1.3.2. Formulación del problema .....	28
1.4. Objetivos .....	28
1.4.1. Objetivo general .....	28
1.4.2. Objetivos específicos.....	29
1.5. Justificación.....	29
1.6. Método de desarrollo.....	31
1.6.1. Método Seleccionado – Hybrid Solo eXtreme Programming.....	31
1.6.2. Etapas .....	32
1.6.3. Artefactos .....	35
1.6.4. Roles.....	36
1.6.5. Métricas .....	37
1.7. Línea y áreas temáticas .....	37
1.8. Presupuesto.....	37
1.9. Cronograma.....	39

1.9.1. Objetivo Específico 1.....	39
1.9.2. Objetivo específico 2 .....	39
1.9.3. Objetivo específico 3 .....	40
1.10. Productos esperados .....	42
1.11. Condiciones de entrega .....	42
2. Desarrollo de objetivos .....	43
2.1. Objetivo específico 1.....	43
2.1.1. Desarrollo .....	43
2.1.2. Discusión.....	51
2.2. Objetivo específico 2.....	52
2.2.1. Fase de exploración.....	52
2.2.2. Fase de planeación .....	64
2.2.3. Fase de iteraciones .....	65
2.2.4. Fase de producción.....	162
2.2.5. Fase de muerte.....	166
2.2.6. Discusión.....	167
2.3. Objetivo específico 3.....	169
2.3.1. Desarrollo .....	169
2.3.2. Discusión.....	181
3. Conclusiones .....	183
4. Recomendaciones.....	184
Referencias bibliográficas .....	185

## Índice de Tablas

Tabla 1. Blind Explorer .....	20
Tabla 2. Voice Guided Navigation for visually impaired. ....	21
Tabla 3. BARD Mobile .....	22
Tabla 4. Ballyland Code.....	24
Tabla 5. Accessible 3D Audio Maze Game .....	25
Tabla 6. Sensors-plus example.....	26
Tabla 7. Plantilla de Historia de Usuario .....	35
Tabla 8. Especificación de caso de prueba.....	36
Tabla 9. Product Backlog .....	36
Tabla 10. Presupuesto global del proyecto.....	38
Tabla 11. Descripción de la inversión en personal.....	38
Tabla 12. Otros rubros.....	38
Tabla 13. Diagrama de Gantt del Cronograma .....	41
Tabla 14. Criterios de aceptación - Instrumento de recolección .....	45
Tabla 15. Tabla de contingencia con edad .....	49
Tabla 16. Asistencia utilizada vs Interés por cursos .....	50
Tabla 17. Interés por cursos vs Interés por juegos interactivos .....	51
Tabla 18. Product Backlog .....	52
Tabla 19. Criterios de aceptación - Normas WCAG 2.1.....	56
Tabla 20. Selección de normas WCAG 2.1 según criterios de aceptación .....	57
Tabla 21. Atributo de calidad - Accesibilidad.....	58
Tabla 22. Distribución de requerimientos en iteraciones .....	64
Tabla 23. Especificación historia de usuario 1 .....	65
Tabla 24. Especificación historia de usuario 2.....	66
Tabla 25. Especificación historia de usuario 3.....	67
Tabla 26. Especificación historia de usuario 4.....	68
Tabla 27. Pruebas unitarias para la clase AuthController .....	76
Tabla 28. Pruebas unitarias para la clase AuthService.....	79
Tabla 29. Pruebas de integración para funcionalidad de autenticación .....	82

Tabla 30. Especificación historia de usuario 10.....	85
Tabla 31. Especificación historia de usuario 11.....	86
Tabla 32. Especificación historia de usuario 12.....	87
Tabla 33. Especificación requerimiento no funcional 3.....	88
Tabla 34. Pruebas unitarias para la clase ThemeController .....	96
Tabla 35. Pruebas unitarias para la clase ThemeService .....	97
Tabla 36. Pruebas de integración para funcionalidad de cambio de tema .....	99
Tabla 37. Pruebas unitarias para la clase FontSizeController.....	101
Tabla 38. Pruebas unitarias para la clase FontSizeService .....	103
Tabla 39. Pruebas de integración para funcionalidad de tamaño de fuente .....	106
Tabla 40. Especificación historia de usuario 5.....	107
Tabla 41. Especificación historia de usuario 6.....	108
Tabla 42. Especificación requerimiento no funcional 1.....	109
Tabla 43. Especificación requerimiento no funcional 2.....	109
Tabla 44. Pruebas unitarias para la clase courseColorsController .....	118
Tabla 45. Pruebas de integración para funcionalidad de cursos .....	119
Tabla 46. Especificación historia de usuario 7.....	121
Tabla 47. Especificación historia de usuario 8.....	122
Tabla 48. Especificación historia de usuario 9.....	123
Tabla 49. Especificación requerimiento no funcional 4.....	124
Tabla 50. Especificación requerimiento no funcional 5.....	124
Tabla 51. Especificación requerimiento no funcional 6.....	124
Tabla 52. Especificación requerimiento no funcional 7.....	125
Tabla 53. Pruebas de integración para funcionalidad de juegos .....	133
Tabla 54. Especificación historia de usuario 13.....	136
Tabla 55. Especificación historia de usuario 14.....	137
Tabla 56. Especificación historia de usuario 15.....	138
Tabla 57. Pruebas de caja negra - Iteración 5 .....	144
Tabla 58. Especificación historia de usuario 16.....	150
Tabla 59. Especificación historia de usuario 17.....	151
Tabla 60. Especificación historia de usuario 18.....	152

Tabla 61. Pruebas de caja negra – Iteración 6.....	157
Tabla 62. Implementación de normas de accesibilidad en el proceso de desarrollo.....	169
Tabla 63. Clasificación de preguntas en instrumento de recolección .....	172
Tabla 64. Asignación de valores de respuesta según tipo de pregunta .....	174
Tabla 65. Preguntas de cuestionario para evaluación de expertos .....	176

## Índice de Figuras

Figura 1. Blind Explorer.....	21
Figura 2. Voice Guided Navigation for visually impaired.....	22
Figura 3. BARD Mobile.....	23
Figura 4. Ballyland Code .....	24
Figura 5. Accessible 3D Audio Maze Game.....	25
Figura 6. GitFlow Workflow.....	34
Figura 7. Evidencia de caracterización de la población .....	44
Figura 8. Distribución de Ocupaciones .....	46
Figura 9. Aplicaciones utilizadas por la población .....	47
Figura 10. Nube de palabras - Preferencia en aplicaciones.....	48
Figura 11. Preferencia en imágenes .....	49
Figura 12. Riverpod Architecture.....	60
Figura 13. Capa de presentación .....	61
Figura 14. Capa de datos .....	62
Figura 15. Capa de aplicación.....	63
Figura 16. Comparación de estructura feature-first vs layer-first .....	63
Figura 17. Selección de colores y diseño de logotipo .....	68
Figura 18. Diseño de interfaces de usuario para autenticación .....	69
Figura 19 Estructura de documentos para colección de usuarios.....	70
Figura 20 Diagrama UML para funcionalidad de autenticación.....	71
Figura 21. Comunicación de componentes - Autenticación .....	72
Figura 22. Renderizado condicional en página de autenticación .....	74
Figura 23. Proceso de registro en la aplicación.....	74
Figura 24. Ejemplo de mocking para pruebas unitarias. ....	75
Figura 25. Resultados de pruebas unitarias para la clase AuthController.....	80
Figura 26. Resultado de pruebas unitarias para la clase AuthService.....	81
Figura 27. Resultados de pruebas de integración para funcionalidad de autenticación .....	83
Figura 28. Selección de colores y diseño de logotipo en alto contraste.....	89
Figura 29. Diseño de interfaces de usuario para configuraciones de accesibilidad .....	90

Figura 30. Diseño de base de datos para funcionalidades de tema y fuente .....	91
Figura 31. Diagrama UML para funcionalidad de temas .....	92
Figura 32. Diagrama UML para funcionalidad de tamaño de fuente.....	92
Figura 33. Comunicación de componentes – Temas.....	93
Figura 34. Comunicación de componentes – Tamaño de fuente .....	94
Figura 35. Límites en factor de escala de texto.....	95
Figura 36. Cambio de temas en aplicación .....	95
Figura 37. Cambio de tamaño de fuente en aplicación .....	96
Figura 38. Resultados de pruebas unitarias para la clase ThemeController.....	99
Figura 39. Resultados de pruebas unitarias para la clase ThemeService .....	99
Figura 40. Resultados de pruebas de integración para funcionalidad de cambio de tema .....	101
Figura 41. Resultados de pruebas unitarias para la clase FontSizeController .....	104
Figura 42. Resultados de pruebas unitarias para la clase FontSizeService .....	105
Figura 43. Resultados de pruebas de integración - cambio de tamaño de fuente .....	107
Figura 44. Diseño de interfaces pantalla de cursos. ....	110
Figura 45. Diseño de interfaces para pantalla de curso.....	111
Figura 46. Colección de cursos y contenidos de curso .....	111
Figura 47. Diagrama UML para funcionalidad de cursos.....	112
Figura 48. Comunicación de componentes – cursos y contenidos.....	113
Figura 49. Selección de componentes en interfaz gráfica de cursos .....	114
Figura 50. Página de cursos en aplicación .....	115
Figura 51. Página de curso en aplicación.....	116
Figura 52. Implementación de norma 1.1.1. – contenido no textual.....	117
Figura 53. Aplicación con modo de depuración semántica .....	117
Figura 54. Resultados de pruebas unitarias para la clase courseColorsController.....	120
Figura 55. Resultados de pruebas de integración para funcionalidad de cursos .....	121
Figura 56. Diseño de interfaces de usuario para juego de trivia .....	125
Figura 57. Diseño de interfaces de usuario para juego de ahorcado .....	126
Figura 58. Estructura de colección de juegos.....	127
Figura 59. Diagrama UML para funcionalidad de juegos.....	127
Figura 60. Implementación de norma 2.4.6 y 2.4.10 en sección de accesibilidad .....	129

Figura 61. Prueba de accesibilidad para norma 2.5.5.....	130
Figura 62. Implementación de norma 4.1.2. mediante depuración semántica .....	130
Figura 63. Comunicación de componentes - juegos .....	131
Figura 64. Resultados de pruebas de integración para juego de ahorcado.....	134
Figura 65. Resultados de pruebas de integración para juego de trivia .....	135
Figura 66. Diseño de interfaz de usuario – creación de cursos y juegos.....	139
Figura 67. Diagrama UML para funcionalidad de cursos (administrador).....	140
Figura 68. Diagrama UML para funcionalidad de cursos – operaciones CRUD.....	141
Figura 69. Diagrama UML para funcionalidad de juegos (administrador).....	142
Figura 70. Comunicación de componentes – Creación de cursos.....	143
Figura 71. Comunicación de componentes – Creación de juegos.....	143
Figura 72. Prueba de creación de cursos exitosa (CP-079).....	147
Figura 73. Pruebas de creación de cursos fallida (CP-080, CP-081) .....	148
Figura 74. Pruebas de creación de juego de ahorcado fallida (CP-083, CP-084, CP-085).....	148
Figura 75. Prueba de creación de juego de ahorcado exitosa (CP-086).....	149
Figura 76. Prueba de creación de juego de trivia fallida (CP-087, CP-088, CP-089).....	149
Figura 77. Prueba de creación de juego de trivia exitosa (CP-090).....	149
Figura 78. Diseño de interfaz de usuario para creación de cursos .....	152
Figura 79. Diagrama UML para funcionalidad de contenidos de cursos.....	153
Figura 80. Diagrama UML para funcionalidad de cursos (Administrador).....	154
Figura 81. Comunicación de componentes – Agregar contenido a curso.....	155
Figura 82. Panel de contenidos de cursos en aplicación de administrador .....	156
Figura 83. Comunicación de componentes – Publicación de cursos .....	157
Figura 84. Prueba de agregar video a curso exitosamente (CP-091) .....	160
Figura 85. Pruebas de agregar video a curso con fallos (CP-092, CP-093, CP-094).....	160
Figura 86. Prueba de agregar juego a curso exitosamente (CP-095) .....	160
Figura 87. Prueba de agregar juego a curso con fallos (CP-096, CP-097, CP-099) .....	161
Figura 88. Prueba de publicar curso sin contenidos (CP-099).....	161
Figura 89. Prueba de publicar curso con contenidos (CP-100).....	161
Figura 90. Clasificación de la aplicación en PlayStore.....	162
Figura 91. Primera versión de la aplicación en revisión .....	163

Figura 92. Resultados de pruebas de rendimiento en la aplicación .....	164
Figura 93. Resultados de pruebas de accesibilidad en aplicación.....	165
Figura 94. Aplicación en acceso anticipado de PlayStore .....	165
Figura 95. Contenidos de prueba creados para probar la aplicación.....	170
Figura 96. Evidencia de pruebas de la aplicación en la población.....	171
Figura 97. Interés de los participantes por seguir utilizando la aplicación .....	172
Figura 98. Apoyo técnico para usuarios de la aplicación.....	173
Figura 99. Aprender a utilizar la aplicación rápidamente. ....	174
Figura 100. Resultados de instrumento de recolección después del mapeo de datos. ....	175
Figura 101. Representación de los resultados de cuestionario SUS .....	175
Figura 102. Seguimiento de metodología en proyecto.....	177
Figura 103. Herramientas utilizadas en el proyecto.....	177
Figura 104. Costos y complejidad al agregar nuevas funcionalidades .....	178
Figura 105. Repetición de código en el proyecto.....	179
Figura 106. Estructura de directorios del proyecto .....	179
Figura 107. Tiempo de inicio en la aplicación .....	180
Figura 108. Utilidad del desarrollo de aplicaciones para personas con discapacidad visual .....	180

## **Índice de Anexos**

Anexo A. Carta que justifica el desarrollo tecnológico .....	189
Anexo B. Instrumento de recolección para prueba piloto .....	190
Anexo C. Instrumento de recolección final para caracterización.....	195
Anexo D. Análisis de datos – Caracterización FUNDONAR.....	201
Anexo E. Instrumento SUS para evaluación de usabilidad.....	212
Anexo F. Análisis de datos - Cuestionario SUS.....	217
Anexo G. Instrumento de evaluación - Proceso de desarrollo de software.....	222
Anexo H. Análisis de datos – Evaluación de expertos.....	230
Anexo I. Carta de entrega del desarrollo tecnológico .....	236
Anexo J. Otros enlaces .....	237

## **Introducción**

En la era actual, marcada por rápidos cambios en las Tecnologías de la Información y la Comunicación (TIC), el acceso a contenidos digitales se ha vuelto crucial para la integración de las personas en la sociedad moderna. Sin embargo, mientras estas tecnologías avanzan, existen poblaciones que enfrentan barreras significativas al acceder a este tipo de contenidos, como es el caso de las personas con discapacidad visual. El presente proyecto se centra en la implementación de normativas de accesibilidad dentro del proceso de desarrollo de software, para la creación de contenidos digitales dirigidos a personas con discapacidad visual. Surge como complemento a una investigación profesoral previa realizada por el programa de Mercadeo de la Universidad Mariana, cuyo objetivo fue el diseño de un material didáctico para las personas con discapacidad visual en la ciudad de Pasto, enfocándose en la población de la fundación FUNDONAR.

En ese contexto, se desarrolló una estrategia computacional basada en tecnologías móviles para las personas con discapacidad visual de la fundación FUNDONAR, que consiste en una aplicación móvil que ofrece contenidos digitales variados en formato de cursos, incluyendo adaptaciones de minijuegos populares y contenido multimedia; además de una aplicación web de administrador que sirve para gestionar los contenidos presentados en aplicación móvil. Esta estrategia se construyó utilizando Flutter como Framework de desarrollo multiplataforma y Firebase como servicio de Backend. En el desarrollo de la aplicación móvil, se aplicó la normativa internacional de accesibilidad WCAG<sup>1</sup>, para lo cual, se realizó una revisión sistemática de la normativa mencionada, aplicando un filtro para seleccionar las normas a implementar según criterios de inclusión específicos.

Como parte del primer objetivo específico del proyecto, se caracterizó a la población de FUNDONAR, utilizando una muestra de 24 pacientes quienes fueron encuestados por medio de llamadas telefónicas. A partir de esta caracterización se identificó las principales barreras que enfrenta la población al consumir contenidos digitales, enfatizando en la experiencia a la hora de acceder a educación en línea. Se identificó que las principales dificultades que enfrenta la población

---

<sup>1</sup> Web Content Accessibility Guidelines, es un estándar internacional de accesibilidad orientado a personas con distintos tipos de discapacidad.

al acceder a contenidos digitales están ligadas a su presentación visual en pantalla, teniendo en cuenta factores como el tamaño de letra y el uso de colores. Por otro lado, se observó que los encuestados que utilizan lectores de pantalla no presentan problemas de compatibilidad con las aplicaciones que usan. Sin embargo, existen situaciones donde los componentes no contienen etiquetas descriptivas, lo cual dificulta el uso de estos lectores en determinados contextos.

Además, se encontró que Android es la plataforma más utilizada por la población, y que los medios preferidos de interacción con los dispositivos son el audio y la vibración (Ver Anexo D. Análisis de datos – Caracterización FUNDONAR.) También se preguntó por el interés sobre acceder a contenidos digitales accesibles por medio de una estrategia computacional, a lo que la mayoría de los encuestados respondieron de manera positiva. Adicionalmente, se identificaron preferencias diversas a la hora de consumir contenidos digitales, teniendo en cuenta factores como el color, contraste, tamaño de fuente, entre otros. Esto hizo necesario implementar todos estos parámetros en la aplicación.

Para el proceso de desarrollo de la estrategia computacional se siguió una metodología híbrida entre Solo eXtreme Programming y SCRUM, tomando las prácticas, artefactos y conceptos que mejor se ajustan a un proceso con un solo desarrollador. Esto incluyó la documentación de todas las fases de la metodología propuesta, desde el levantamiento de requerimientos hasta el despliegue a producción y lanzamiento de la aplicación a PlayStore poniéndola a disposición del público. El proceso completo fue acompañado por la doctora especialista en baja visión de FUNDONAR, quién brindó retroalimentación constante sobre la aplicación desarrollada, además de apoyar el proceso de caracterización y pruebas en la población.

Una vez completado el proceso de desarrollo, se evaluó la calidad de la estrategia computacional desarrollada, teniendo en cuenta las métricas accesibilidad, usabilidad, adecuación funcional y mantenibilidad. La accesibilidad fue evaluada a lo largo del proceso de desarrollo mediante pruebas de integración y pruebas de Google al realizar el despliegue a producción. Para la evaluación de usabilidad se utilizó el instrumento SUS en una prueba realizada a 10 pacientes de la población, quienes hicieron uso de la aplicación durante aproximadamente cinco minutos. Se obtuvo un puntaje de 80.5, el cual está por encima de la media y entra en la categoría de aceptable según el

cuestionario. Los pacientes expresaron que la aplicación les pareció agradable, así como su interés por utilizarla con frecuencia. Al utilizar un diseño sencillo, los pacientes no tuvieron que aprender muchas cosas antes de utilizar la aplicación, y la mayoría comentó que no necesitaría de apoyo técnico para seguirla utilizando. Por lo tanto, la evaluación de usabilidad de la aplicación fue exitosa.

Luego, para evaluar la adecuación funcional y mantenibilidad de la aplicación se elaboró un instrumento de recolección tomando como base la estructura del instrumento SUS, y se sometió la estrategia computacional desarrollada a una evaluación por parte de expertos en el área de construcción de software, obteniendo una valoración positiva en general, junto a comentarios de retroalimentación sobre aspectos por mejorar a nivel técnico.

Actualmente, la aplicación es utilizada como material didáctico para los pacientes de FUNDONAR, quienes completan los cursos ofrecidos en la aplicación según las indicaciones de la especialista en baja visión de la fundación. De esta manera, no solamente se impulsa a la población a acceder a contenidos digitales, sino que, además, es posible enfocar estos contenidos hacia un ámbito educativo para lograr un mayor beneficio y aporte en la población. Estos contenidos pueden servir también como medio para difundir información sobre diversas opciones de accesibilidad, herramientas y plataformas disponibles para los usuarios. Por otro lado, la aplicación web de administrador sirve como puente entre los contenidos digitales y el usuario con discapacidad visual, presentando estos contenidos de manera accesible y aplicando normativas específicas de manera automática. Al realizar la entrega de esta aplicación de administrador, se garantiza su uso para generar nuevos contenidos de manera constante y así aportar a las personas con discapacidad visual que utilicen la aplicación.

Finalmente, el desarrollo de este proyecto también contribuye como aporte a la literatura en cuanto al desarrollo e implementación de normativas de accesibilidad en proyectos de software reales. Teniendo en cuenta de que tanto el código fuente como la aplicación están disponibles al público, estos pueden ser tomados como referencia para futuras investigaciones enfocadas al desarrollo de software en general, y el desarrollo móvil en particular. De esta manera, incentiva el uso de normativas de accesibilidad en futuros proyectos, impulsando así la creación de entornos

digitales más inclusivos y accesibles para para personas en situación de discapacidad, generando un proyecto de alto impacto social.

## 1. Elementos del proceso

### 1.1. Estado de la cuestión

Con el fin de explorar el estado del arte y las tendencias actuales en accesibilidad y desarrollo móvil, se realizó una búsqueda sistemática de antecedentes en diversas fuentes de información, tales como repositorios de código abierto, publicaciones científicas y plataformas de distribución de aplicaciones móviles. Como resultado de este proceso, se seleccionaron algunos desarrollos tecnológicos relevantes que pueden contribuir al diseño y desarrollo de la estrategia computacional de este trabajo.

#### 1.1.1. *Blind Explorer*

Es una aplicación móvil que brinda guiado sensorial a personas con discapacidad visual. Esta aplicación implementa la navegación a través de sonidos envolventes y puede servir como guía para la implementación de este tipo de sonidos en la estrategia computacional a desarrollar.

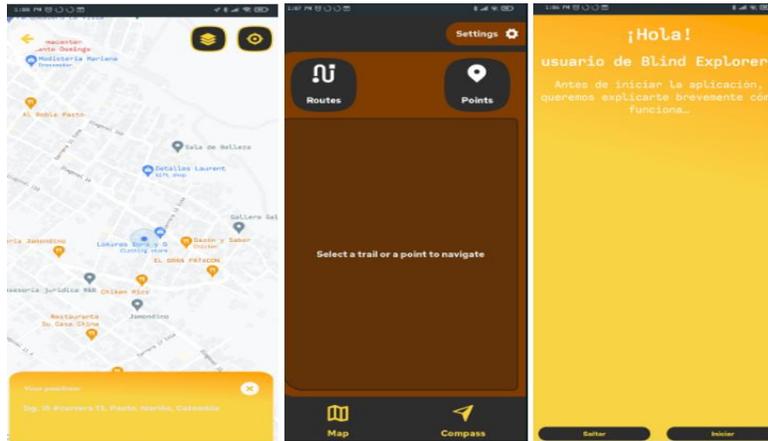
**Tabla 1**

*Blind Explorer*

<b>Objetivo</b>	App que busca brindar un guiado sensorial a personas con discapacidad visual total
<b>Tiempo de existencia</b>	4 años
<b>Frecuencia de actualización</b>	Fecha de actualización: 11/3/2019
<b>Funcionalidad</b>	Ofrece navegación en espacios abiertos mediante la integración de sonidos envolventes en combinación con GPS
<b>Atributos de calidad</b>	Usabilidad, Portabilidad, Adecuación Funcional
<b>Licenciamiento</b>	Gratuito

**Figura 1**

*Blind Explorer*



Fuente: (*Blind Explorer*, 2022)

### 1.1.2. *Voice Guided Navigation for visually impaired.*

Navegación por medio de voz para personas con discapacidad visual. El sistema de guía por voz puede servir de ejemplo para la realización de este proyecto. Tiene un sistema de ingreso sencillo, basado en gestos táctiles. Además, su código fuente está disponible en GitHub, por lo cual sirve de guía para el desarrollo del proyecto.

**Tabla 2**

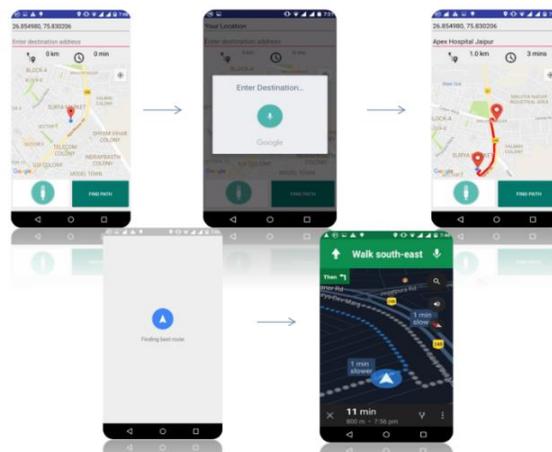
*Voice Guided Navigation for visually impaired.*

<b>Objetivo</b>	Guiar a personas con discapacidad visual eligiendo el camino más rápido para llegar destinos
<b>Tiempo de existencia</b>	4 años
<b>Frecuencia de actualización</b>	Aproximadamente 7 actualizaciones por año

<b>Funcionalidad</b>	Navegación por medio de gestos táctiles disponibles desde la pantalla de inicio. Navegación guiada por voz.
<b>Atributos de calidad</b>	Usabilidad, Portabilidad, Adecuación Funcional
<b>Licencia</b>	Código Abierto

**Figura 2**

*Voice Guided Navigation for visually impaired.*



Fuente: (Gaur, 2019)

### 1.1.3. *BARD Mobile*

Es una biblioteca accesible para personas con discapacidad visual. Puede servir de guía en cuanto a la interacción con contenidos digitales textuales a través de audios text-to-speech, así como la navegación dentro de la plataforma por parte de las personas con discapacidad visual.

**Tabla 3**

*BARD Mobile*

<b>Objetivo</b>	Ofrecer una biblioteca accesible personas con discapacidad visual a manera de braille o audio
<b>Tiempo de existencia</b>	8 años
<b>Frecuencia de actualización</b>	Ultima actualización 14 de febrero 2023
<b>Funcionalidad</b>	Navegación por medio de gestos y voz.
<b>Atributos de calidad</b>	Usabilidad, Portabilidad, Adecuación Funcional, Fiabilidad
<b>Licencia</b>	Copyright © United States Government Work

### Figura 3

#### *BARD Mobile*



Fuente: (*BARD Mobile*, 2023)

#### **1.1.4. Ballyland Code**

Usa la gamificación de manera accesible para enseñar lógica de programación, enfocándose en estudiantes con discapacidad visual. Sirve como referente para la realización de este proyecto en áreas como la usabilidad y la accesibilidad. Al estar enfocado en un área como la programación, hace uso de distintas estrategias pedagógicas aplicadas de manera accesible, las cuales pueden servir de ejemplo para incorporar funcionalidades en el presente proyecto.

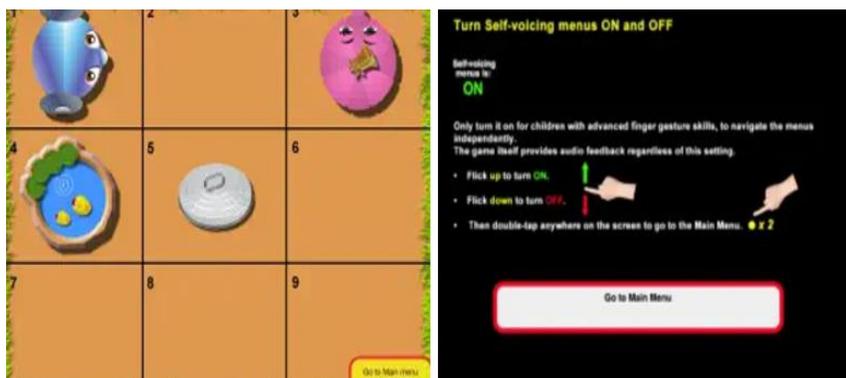
**Tabla 4**

*Ballyland Code*

<b>Objetivo</b>	Enseñar lógica de programación a estudiantes con discapacidad visual total o parcial
<b>Tiempo de existencia</b>	-
<b>Frecuencia de actualización</b>	-
<b>Funcionalidad</b>	Gamificación del aprendizaje en programación
<b>Atributos de calidad</b>	Usabilidad, Portabilidad, Adecuación Funcional
<b>Licenciamiento</b>	© 2018 Sonokids Australia

**Figura 4**

*Ballyland Code*



Fuente: (*Ballyland Code*, s/f)

### 1.1.5. Accessible 3D Audio Maze Game

Es un juego diseñado para personas con discapacidad visual total, el cual hace uso del giroscopio como medio de interacción entre el usuario y el juego. El juego consiste en resolver un laberinto,

donde se debe llevar una pelota desde un punto hasta otro, usando la gravedad para moverla. El usuario mueve el dispositivo, que debe estar horizontal y perpendicular al suelo, de manera que la pelota se mueva a lo largo del laberinto. Para saber la posición de la pelota, se emiten sonidos cuando esta choca con las paredes del laberinto, o cuando esta está rodando. Esta aplicación sirve de guía en cuanto a cómo implementar el giroscopio para la navegación en el aplicativo, además de utilizar sonido 3D para localizar la posición de la pelota.

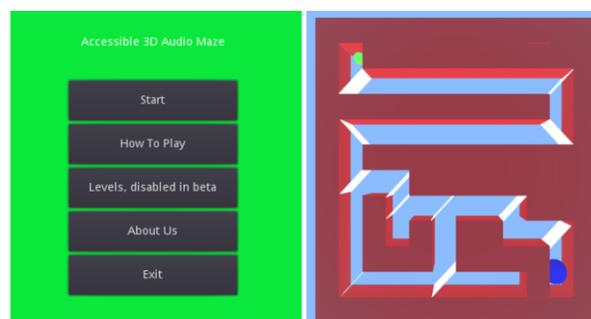
**Tabla 5**

Accessible 3D Audio Maze Game

<b>Objetivo</b>	Ofrecer entretenimiento a personas con discapacidad visual parcial o total.
<b>Tiempo de existencia</b>	5 meses
<b>Frecuencia de actualización</b>	Última actualización 11 oct 2022
<b>Funcionalidad</b>	Navegación por medio de gestos, voz y uso del giroscopio
<b>Atributos de calidad</b>	Usabilidad, Portabilidad
<b>Licenciamiento</b>	Gratuito

**Figura 5**

Accessible 3D Audio Maze Game



Fuente: (Accessible 3D Audio Maze Game, 2022)

### 1.1.6. *Sensors-plus example*

Es un repositorio de ejemplo de uso de “sensors\_plus”, un paquete de Flutter que da acceso a dos sensores, el acelerómetro y el giroscopio para controlar la posición del dispositivo. Este repositorio puede ser de gran utilidad porque muestra una alternativa a Android Studio para manejar estos sensores, ofreciendo ventajas como servir tanto para Android como para IOS.

**Tabla 6**

*Sensors-plus example*

<b>Objetivo</b>	Mostrar el funcionamiento del giroscopio en dispositivos móviles usando Flutter
<b>Tiempo de existencia</b>	1 año
<b>Frecuencia de actualización</b>	Julio 21, 2021
<b>Funcionalidad</b>	Uso de giroscopio para controlar posición de objetos en la pantalla
<b>Atributos de calidad</b>	Adecuación Funcional
<b>Licenciamiento</b>	Código Abierto

Las aplicaciones previamente mencionadas han sido diseñadas para ser compatibles con TalkBack, una alternativa de accesibilidad presente en los dispositivos móviles. El control y la navegación en estas aplicaciones se llevan a cabo a través de gestos en la pantalla, específicamente deslizando hacia la izquierda y derecha para moverse entre las diferentes opciones, y pulsando la pantalla dos veces para seleccionar una opción en particular. Cada uno de estos movimientos se comunica mediante un sistema de audio que utiliza una voz sintética para describir el contenido en pantalla. Por lo tanto, personas con discapacidad visual total pueden hacer uso de estas aplicaciones.

Además, algunas aplicaciones como emplean el giroscopio como un medio adicional de navegación y proporcionan su código fuente en GitHub, lo cual es una contribución significativa al aspecto técnico de este proyecto. Al hacer que su código esté disponible públicamente, estas

aplicaciones permiten que otros desarrolladores realicen aportes a su trabajo o lo tomen como referencia para desarrollar más aplicaciones accesibles con funcionalidades similares. Dentro de los referentes seleccionados, se incluyen aplicaciones desarrolladas en Flutter y Android Studio, lo cual sirve como guía a la hora de elegir la tecnología más apropiada para usar en este proyecto. Cabe destacar que, de las aplicaciones descritas, solamente una está disponible en idioma español, el resto funciona solamente en inglés.

## **1.2. Título**

Accesibilidad aplicada a la creación de contenidos digitales para personas con discapacidad visual

## **1.3. Problema de Investigación**

### ***1.3.1. Descripción del problema***

En los últimos años se han presentado cambios disruptivos y avances en lo que refiere a Tecnologías de la Información y la comunicación (TIC), haciendo que las herramientas digitales sean cada vez más necesarias para vivir en la sociedad actual. Sin embargo, existen poblaciones que no tienen acceso a estas tecnologías y que, por ende, no tienen la posibilidad de adaptarse a estos cambios (Escandell Bermúdez et al., 2016). Una de esas poblaciones son las personas con discapacidad visual. Como lo menciona (Velázquez et al., 2020), la discapacidad visual es un problema global que afecta la percepción total o parcial de las imágenes en las personas. Esta población enfrenta barreras significativas en su capacidad para realizar tareas cotidianas como navegar por la web, comunicarse a través de medios digitales y utilizar dispositivos tecnológicos; lo cual puede afectar su libre desarrollo (Salas et al., 2022).

Como expresa (Paniagua L. et al., 2020), la accesibilidad permite que los contenidos digitales sean utilizados por cualquier persona, independientemente de su condición. A pesar de los esfuerzos por mejorar la accesibilidad a través de normativas a nivel nacional, como la NTC 5854, estas no han sido implementadas de manera eficiente en los medios digitales (Enrique Cortés Fandiño, 2022). Desde la posición de (Zambrano et al., 2020), una posible causa de esto es que los

expertos desarrolladores aún tienen cierto desconocimiento en cuanto a accesibilidad web y estándares relacionados. Esto significa que se omite considerar a las personas con discapacidad visual en la fase inicial del diseño de contenido digital, a pesar de que la incorporación de medidas de accesibilidad en esta etapa resulta ser más eficiente que realizar ajustes posteriores (Henry, 2022). Además, Paniagua señala que aun cuando los desarrolladores conocen las falencias de accesibilidad en las aplicaciones, muy pocas veces son consideradas al construir software.

Las anteriores situaciones hacen que las personas con discapacidad visual presenten un problema para acceder a contenidos digitales. De continuar esta situación, se seguirá extendiendo la denominada brecha digital que describe Escandell Bermúdez como la desigualdad entre las personas con discapacidad y las personas sin discapacidad visual, en cuanto al acceso a contenidos digitales.

Por lo tanto, surge la necesidad de investigar cómo realizar contenido digital para personas con discapacidad visual y aportar al cierre de la brecha digital para esta población, pues es fundamental considerar la accesibilidad desde el inicio del proceso de diseño de contenido digital. Esto implica tener en cuenta las necesidades de las personas con discapacidad visual y pensar en soluciones inclusivas desde el principio del proceso de creación. Así se podrá garantizar no solo que el contenido digital sea accesible para la población con discapacidad visual, sino que, además este contenido sea fácil de utilizar y ofrezca una experiencia agradable a las personas.

### ***1.3.2. Formulación del problema***

¿Cómo disminuir la brecha digital existente en cuanto al acceso a contenidos digitales en personas con discapacidad visual pertenecientes a la fundación FUNDONAR en el municipio de Pasto?

## **1.4. Objetivos**

### ***1.4.1. Objetivo general***

Proponer una estrategia computacional basada en tecnologías móviles para disminuir la brecha digital existente en cuanto al acceso a contenidos digitales en personas con discapacidad visual pertenecientes a la fundación FUNDONAR en el municipio de Pasto.

#### **1.4.2. Objetivos específicos**

- Caracterizar las principales barreras tecnológicas que enfrenta la población con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto.
- Desarrollar una estrategia computacional basada en tecnologías móviles para las personas con discapacidad visual de la fundación FUNDONAR.
- Evaluar la calidad de la estrategia computacional basada en tecnologías móviles, desarrollada para la población con discapacidad visual de la fundación FUNDONAR.

#### **1.5. Justificación**

La discapacidad visual es una de las discapacidades más prevalentes en todo el mundo. Según la Organización Mundial de la Salud (*World Health Organization: WHO, 2022*), se estima que más de 220 millones de personas en todo el mundo tienen algún tipo de discapacidad visual, y se espera que esta cifra aumente a medida que la población envejece y aumenta la prevalencia de enfermedades oculares. Esta discapacidad hace que existan dificultades al realizar tareas cotidianas, entre ellas, acceder a contenidos digitales. Un contenido digital es accesible cuando este puede ser utilizado por todas las personas, independientemente de sus habilidades o discapacidades, incluyendo la discapacidad visual. Existen normativas a nivel internacional que definen estándares de accesibilidad para contenido digital, entre ellos, está la WCAG (*Web Content Accessibility Guidelines (WCAG) 2.1., 2018*), expedida por el World Web Consortium (*World Web Consortium (W3C), s/f*). Esta normativa define tres niveles de conformidad en cuanto a accesibilidad para los sitios web, los cuales son A, AA, y AAA, siendo A el más bajo y AAA el más alto.

De acuerdo con el Instituto Nacional Para Ciegos (Comunicaciones INCI, 2020), se registraron a 1.948.332 personas con discapacidad visual en Colombia, país donde la accesibilidad web se rige a partir de la Norma Técnica Colombiana 5854 (*Ministerio de Tecnologías de la Información y las*

*Comunicaciones* (MINTIC), s/f), la cual está basada en la WCAG 2.1. A nivel legislativo, la ley 1712 de 2020 establece que toda entidad pública, órganos estatales, partidos políticos, entre otros, deben cumplir con un nivel de conformidad AA como mínimo. No obstante, esta normativa tampoco se cumple efectivamente a nivel nacional. El estudio realizado por (Águila et al., 2022) comparó diversos sitios web gubernamentales de las principales ciudades de Perú y Colombia y determinó que solo el sitio web de Villavicencio-Colombia califica como accesible, registrando menos de 30 errores de accesibilidad. Asimismo, (Niño-Vega et al., 2022) evaluó la accesibilidad en sitios web de las mejores universidades de Colombia según el ranking U-Sapiens, y concluyó que ninguna era accesible desde que no cumplían la totalidad de ningún nivel de conformidad descritos por la WCAG 2.1.

Según el INCI, en la región de Nariño la cifra de personas con discapacidad visual fue de 24.616 en el año 2021, representando un 5% del total de la población con discapacidad visual en Colombia. Teniendo en cuenta esto, se busca proponer una estrategia computacional basada en tecnologías móviles que aporte al cierre de la brecha digital para esta población y que le permita acceder a contenidos digitales, siguiendo los estándares de accesibilidad de la WCAG 2.1.

Si bien se han realizado numerosos desarrollos orientados a personas con discapacidad visual, esta estrategia computacional pretende brindar una estimulación multisensorial al usuario para así ofrecer una experiencia agradable e innovadora, pues como menciona Jiménez y Ochoa, citado por Salas, cuando el sentido de la vista se ve afectado, otros sentidos cobran mayor importancia.

Para ello es posible hacer uso de tecnologías propias de dispositivos móviles, tales como la navegación basada en gestos táctiles, vibración del dispositivo, sonidos envolventes por medio de auriculares, uso del giroscopio, acelerómetro, entre otros. De esta forma, es posible que las personas con discapacidad visual usen la estrategia computacional propuesta de una manera distinta, empleando más sentidos y de una forma más interactiva.

La realización de este proyecto beneficia a la población con discapacidad visual en el municipio de Pasto, particularmente a las personas afiliadas a la fundación FUNDONAR, la cual ofrece servicios oftalmológicos accesibles para personas con discapacidad visual de bajos recursos y

dificultades de desplazamiento geográfico. Esta población fue caracterizada en un proyecto de Investigación del programa de Mercadeo de la Universidad Mariana, que tomó una muestra de 29 personas afiliadas a FUNDONAR. Se pretende que el presente proyecto utilice parte de la población trabajada en la investigación mencionada, delimitando el rango de edad entre 10 y 25 años.

Es útil realizar este tipo de proyectos porque se brinda opciones a las personas con discapacidad visual para acceder a contenidos digitales, ofreciendo contenido de calidad. Por otro lado, al construir software accesible supone un aporte valioso al campo del desarrollo, pues se requiere hacer uso de técnicas y herramientas que fomentan la innovación tecnológica. Además, este proyecto puede utilizarse como punto de referencia para futuros desarrollos orientados a personas con discapacidad visual, de manera similar a cómo los referentes de este proyecto sirvieron como guía para su realización.

## **1.6. Método de desarrollo**

### ***1.6.1. Método Seleccionado – Hybrid Solo eXtreme Programming***

Para el desarrollo de la estrategia computacional basada en tecnologías móviles propuesta, se seguirá una metodología híbrida basada esencialmente en eXtreme Programming, con detalles añadidos de Scrum. Estas dos metodologías son las más utilizadas en la actualidad, pues ofrecen maneras efectivas de enfrentarse en cambios en el ambiente de negocio (Akhtar et al., 2022). Además, ambas son consideradas metodologías ágiles, iterativas e incrementales, lo cual ofrece un proceso de desarrollo flexible y adaptable al cambio permitiendo entregar software de calidad en poco tiempo (Bakhtawar et al., 2021).

Como expresa Akhtar, la metodología eXtreme Programming (XP) se recomienda para proyectos donde el equipo es conformado desde 2 hasta 10 integrantes; mientras que Scrum se recomienda para proyectos que cuenten con varios equipos de al rededor de 10 integrantes. Sin embargo, como lo menciona (Cronin, 2001), XP es ideal también para equipos conformados por un solo integrante. Por esta razón, se elige a XP como metodología de desarrollo principal en la

metodología híbrida propuesta. Además, XP es altamente moldeable para cubrir necesidades concretas dependiendo del contexto de cada equipo de trabajo, por ello se han realizado varias versiones personalizadas de XP, siendo una de ellas eXtreme Solo Programming, descrita por Cronin. La metodología XP consiste en seis fases principales, que son Exploración, Planeación, Iteraciones, Producción, Mantenimiento y fase muerta. Adicionalmente, XP se basa en varios principios que buscan que el desarrollo de software sea de calidad, entre los más destacados está Test Driven Development (TDD), que consiste en construir pruebas unitarias antes de implementar cualquier código, y Simple Design, que busca encontrar la solución más sencilla a un problema.

Por otro lado, de acuerdo con Akhtar, Scrum consiste en tres fases: “pre-game”, “game” y “postgame”. Pregame tiene como objetivo crear el Product Backlog, que es una lista de requerimientos funcionales y no funcionales, la cual puede ser modificada continuamente dependiendo de las necesidades del cliente (Benefield et al., 2010). Game, por otro lado, consiste en realizar iteraciones llamadas “sprints” que duran desde una semana hasta cuatro. Finalmente, postgame es la fase donde el producto final es entregado. En el presente proyecto, se pretende adaptar los conceptos de Product Backlog en la fase de planeación del proyecto, pues ofrece una descripción detallada y granular de los requerimientos del producto.

### ***1.6.2. Etapas***

Existen 6 fases principales en la metodología Solo eXtreme Programming, las cuales son Exploración, Planeación, Iteraciones, Producción, Mantenimiento y Fase de muerte. No obstante, (Beck, 2000), inventor de la metodología XP, menciona que estas fases corresponden a un ciclo de vida idealizado, es decir que su estructura puede variar, de manera que dos proyectos XP no deberían ser nunca exactamente iguales. Por lo tanto, a continuación, se presenta una versión simplificada de las fases descritas por Beck.

**1.6.2.1. Fase de Exploración.** El objetivo de esta fase es definir los requerimientos funcionales y no funcionales del proyecto por medio de historias de usuario. Esto supone realizar reuniones entre el cliente y el desarrollador, donde se dialoga sobre funcionalidades, estimación de prioridades y fechas de entrega. En dichas reuniones es común hacer uso de pizarras, esbozos y notas para lograr una comunicación más visual con el cliente. En la metodología híbrida propuesta, se adoptará el concepto de Scrum Product Backlog, donde se recopilan todas las historias de usuario pertinentes para realizar el proyecto. Una vez recopilados los requerimientos del cliente, se realizan propuestas de las tecnologías a utilizar en el sistema, así como posibles arquitecturas a seguir. Beck sugiere que se realicen tres o cuatro propuestas distintas. Esta fase puede durar varias semanas, dependiendo de la complejidad del proyecto y de las necesidades del cliente.

**1.6.2.2. Fase de Planeación.** En esta fase se distribuyen las historias de usuario en varias iteraciones, asignando una fecha de entrega en cada iteración, así como las historias de usuario que se espera terminar. Esta fase puede tener lugar antes de cada iteración. El resultado de cada planeación es un documento donde se constata la duración de la iteración y sus historias de usuario.

**1.6.2.3. Fase de Iteraciones.** Corresponde a cada ciclo donde se desarrolla un grupo concreto de historias de usuario. Generalmente cada iteración dura entre 1 a 4 semanas. Para terminar el proyecto pueden ser necesarias varias iteraciones. En cada iteración se realiza una fase de planeación donde se constata las historias de usuario a trabajar.

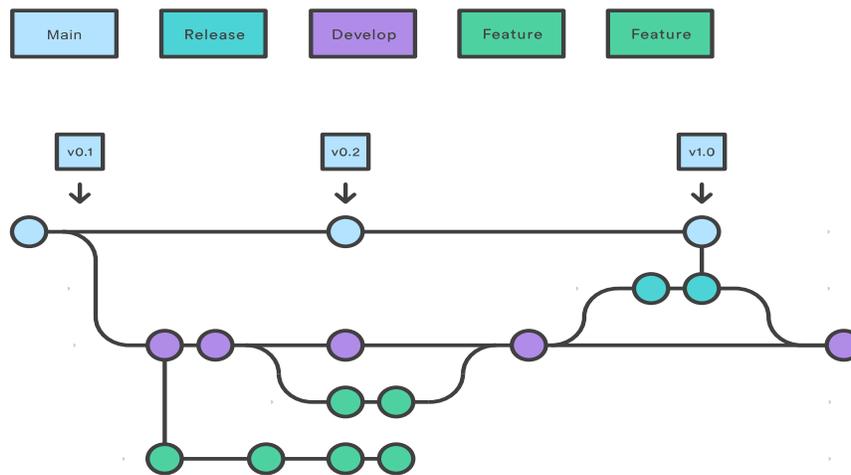
**1.6.2.3.1. Test Driven Development (TDD).** El proceso de desarrollo se sustenta en la práctica Test Driven Development, descrita por Cronin, planteando las pruebas unitarias antes de implementar el código que las apruebe. Además, al desarrollar se utilizará el modelo de branching GitFlow, que permite llevar el control de versiones del proyecto de una manera organizada, escalable y consistente, como lo recomienda (Zuber, 2015).

**1.6.2.3.2. Control de versiones.** Git es un sistema de control de versiones gratuito y de código abierto. Existen múltiples modelos de branching para Git, es decir, convenciones al nombrar y utilizar ramas en un proyecto.

El modelo de branching GitFlow propone el uso de los siguientes tipos de ramas: 'main', que almacena cada lanzamiento estable del programa; 'develop' que sirve de integración para las distintas ramas 'feature', las cuales almacenan cada nueva funcionalidad por separado sin interactuar directamente con 'main'. Cuando la rama 'develop' ha adquirido suficientes funcionalidades para lanzar una versión oficial del programa, se utiliza una rama 'release' para hacer la integración con main. En esta rama se agregan detalles como documentación referente a la versión específica. El modelo de branching GitFlow se describe en la siguiente figura.

**Figura 6**

*GitFlow Workflow*



Fuente: (ATLASSIAN, s/f)

**1.6.2.4. Fase de Producción.** Akhtar menciona que eXtreme Programming busca liberar nuevas funcionalidades de manera constante. El código producido al final de cada iteración debería cumplir con las pruebas unitarias abordadas en el TDD. Una vez terminadas las iteraciones necesarias para cumplir con los requerimientos del cliente, se realiza el proceso de despliegue, haciendo que el sistema esté disponible para que el cliente pueda probarlo.

**1.6.2.5. Fase de muerte.** El final del proceso ocurre cuando el cliente está satisfecho con el sistema, y no se desea agregar más funcionalidades al mismo. Este es el momento de redactar la documentación final del software. Beck recomienda que la documentación se extienda desde cinco a diez páginas, donde se dé un tour por el sistema, lo suficientemente explícito como para que, si alguien desea cambiar algo dentro de cinco años, pueda guiarse de esta documentación.

### 1.6.3. Artefactos

**1.6.3.1. Historias de Usuario.** Desde la perspectiva de Akhtar, las historias de usuario deberían elaborarse utilizando tarjetas durante la etapa de planeación. Para ello, se utilizará el siguiente formato:

**Tabla 7**

*Plantilla de Historia de Usuario*

Historia de Usuario			
<b>Código:</b>	HU-X		
<b>Nombre:</b>	[Nombre de Historia de Usuario]		
<b>Actor:</b>	[Actor de Historia de Usuario]		
<b>Descripción:</b>	Como [Actor] quiero [Funcionalidad] que [beneficio]		
<b>Criterios de aceptación:</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando [Condición]	Entonces, [Resultado]
	2	Cuando [Condición]	Entonces, [Resultado]

Nota: Formato elaborado con base en la propuesta presentada por (Villamizar Suaza, 2013)

**1.6.3.2. Casos de Prueba.** Corresponden a las pruebas unitarias realizadas al desarrollar el software, las cuales aportan al cumplimiento de los criterios de aceptación de cada historia de usuario. Estos casos de prueba serán definidos mediante el siguiente artefacto:

**Tabla 8**

*Especificación de caso de prueba*

<b>Id</b>	<b>Nombr</b>	<b>Clas</b>	<b>Métod</b>	<b>H</b>	<b>Valores de</b>	<b>Resultado</b>
<b>CP</b>	<b>e</b>	<b>e</b>	<b>o</b>	<b>U</b>	<b>entrada</b>	<b>esperado</b>

Nota: Artefacto elaborado con base en la propuesta presentada por (Villalobos Salcedo, 2008)

**1.6.3.3. Product Backlog.** Artefacto resultante de la fase de Exploración, contiene todas las historias de usuario

**Tabla 9**

*Product Backlog*

<b>Código</b>	<b>Como</b>	<b>Quiero</b>	<b>Para</b>	<b>Prioridad</b>	<b>Iteración</b>	<b>Estado</b>
---------------	-------------	---------------	-------------	------------------	------------------	---------------

**1.6.3.4. Diagramas de Arquitectura.** Artefacto resultante de la fase de exploración, consiste en una representación gráfica del sistema a construir, indicando sus componentes y las relaciones que existen entre ellos.

**1.6.3.5. Documentación del proyecto.** Una vez finalizado el proyecto se presenta un documento de guía donde se explique el funcionamiento del programa, además de aspectos técnicos relevantes.

#### **1.6.4. Roles**

**1.6.4.1. Cliente.** Docente Mauricio Salas. Programa de Mercadeo de la Universidad Mariana.

**1.6.4.2. Desarrollador.** Carlos Alberto Botina. Estudiante de Ingeniería de Sistemas de la Universidad Mariana.

**1.6.4.3. Supervisor.** Sandro Fabián Parra Pay. Docente del Programa de Ingeniería de Sistemas de la Universidad Mariana.

### ***1.6.5. Métricas***

**1.6.5.1. Accesibilidad.** Es el grado en el cual el software puede ser utilizado satisfactoriamente por todos los usuarios, independientemente de si presentan o no algún tipo de discapacidad

**1.6.5.2. Usabilidad.** Un software es usable cuando este es fácil de utilizar por el usuario, haciendo que la navegación y realización de tareas sea rápida e intuitiva.

**1.6.5.3. Adecuación funcional.** Es la capacidad del software para satisfacer las necesidades del usuario. Para estimar esta métrica se tienen en cuenta los requerimientos funcionales y no funcionales incluidos en el Project Backlog, los cuales son evaluados por pruebas unitarias y de integración.

**1.6.5.4. Mantenibilidad.** Capacidad del software de ser modificado de manera efectiva y eficiente. El software debe ser modular, reusable, analizable, modificable y testeable. Para cumplir con esta métrica, se siguen buenas prácticas de programación, como principios SOLID y uso de patrones de diseño, propuestos por (Martin, 2000).

## **1.7. Línea y áreas temáticas**

- Línea: Ingeniería, Informática y computación.
- Áreas Temáticas: Innovación, modelamiento y desarrollo de software.

## **1.8. Presupuesto**

**Tabla 10**

*Presupuesto global del proyecto*

<b>Rubros</b>	<b>Valor (\$)</b>
Inversión En Personal	\$ 3.093.333
Otros Rubros	\$ 2.400.000
<b>TOTAL</b>	<b>\$ 5.493.333</b>

**Tabla 11**

*Descripción de la inversión en personal*

<b>Nombre investigador</b>	<b>Valor hora investigador</b>	<b>Dedicación Número total de horas</b>	<b>Valor</b>
Sandro Fabián Parra Pay	\$ 19.333	64	\$ 1.237.333
Carlos Alberto Botina Carpio	\$ 9.666	192	\$ 1.856.000
<b>TOTAL</b>			<b>\$ 3.093.333</b>

**Tabla 12**

*Otros rubros*

<b>Rubro</b>	<b>Justificación</b>	<b>Valor total</b>
Equipos	Renta de computador portátil y dispositivo móvil Android	\$ 1.300.000
Software	Costos de alojamiento en la nube	\$ 100.000
Eventos académicos	Inscripción para ponencias	\$ 500.000

Publicaciones	Publicaciones en revistas	\$ 500.000
<b>TOTAL</b>		<b>\$ 2.400.000</b>

## 1.9. Cronograma

El desarrollo del proyecto se llevará a cabo en dos semestres académicos, es decir 32 semanas en total. En ese tiempo se desarrollarán los objetivos específicos del proyecto, con las siguientes actividades:

### 1.9.1. *Objetivo Específico 1*

Caracterizar las principales barreras tecnológicas que enfrenta la población con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto.

- O1A1. Construir un instrumento de recolección de información (encuesta) con preguntas de selección múltiple dirigido a la población con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto.
- O1A2. Realizar una prueba piloto a 5 personas antes de ejecutar la prueba en la población real.
- O1A3: Hacer ajustes al instrumento de recolección
- O1A4: Aplicar el instrumento de recolección en 25 personas con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto
- O1A5: Análisis e interpretación de la información recolectada con el instrumento de recolección.
- O1A6: Discusión de los datos obtenidos.

### 1.9.2. *Objetivo específico 2*

Desarrollar una estrategia computacional basada en tecnologías móviles para las personas con discapacidad visual de la fundación FUNDONAR.

- O2A1: Fase de Exploración

- O2A2: Fase de Planeación
- O2A3: Fase de Iteraciones
- O2A4: Fase de Producción
- O2A5: Fase de Muerte

### ***1.9.3. Objetivo específico 3***

Evaluar la calidad de la estrategia computacional basada en tecnologías móviles desarrollada

- O3A1: Elaborar un instrumento de recolección de información (encuesta) para medir la usabilidad de la estrategia computacional desarrollada.
- O3A2: Probar la estrategia computacional desarrollada en 10 personas con discapacidad visual de la fundación FUNDONAR de la ciudad de Pasto.
- O3A3: Aplicar la encuesta de usabilidad a las personas que probaron la estrategia computacional desarrollada.
- O3A4: Someter la estrategia computacional desarrollada a una evaluación por parte de expertos en el área de desarrollo de software.

**Tabla 13.** Diagrama de Gantt del Cronograma

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32			
<b>Objetivo específico 1:</b> Caracterizar las principales barreras tecnológicas que enfrenta la población con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto.																																			
O1A1	X	X																																	
O1A2			X	X																															
O1A3					X	X	X																												
O1A4								X	X																										
O1A5										X																									
O1A6										X																									
<b>Objetivo específico 2:</b> Desarrollar una estrategia computacional basada en tecnologías móviles para las personas con discapacidad visual de la fundación FUNDONAR.																																			
O2A1											X	X	X																						
O2A2												X	X																						
O2A3														X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		
O2A4																																		X	
O2A5																																		X	
<b>Objetivo específico 3:</b> Evaluar la calidad de la estrategia computacional basada en tecnologías móviles, desarrollada para la población con discapacidad visual de la fundación FUNDONAR.																																			
O3A1																																X	X		
O3A2																																	X	X	X
O3A3																																	X	X	X
O3A4																																	X	X	X

### **1.10. Productos esperados**

Compromisos adquiridos, en esta parte se describe los productos (tangibles) a obtener en la investigación tales como:

- Monografía (documento de informe final)
- Informe Técnico
- Carta de Innovación
- Código fuente completo
- Tutorial o video tutorial de la instalación de la solución
- Participación en eventos (certificado)

### **1.11. Condiciones de entrega**

Como se planea realizar la entrega de los productos anteriores (físico, digital, archivos comprimidos, CD rotulados, correo electrónico, entre otros)

## **2. Desarrollo de objetivos**

### **2.1. Objetivo específico 1**

#### ***2.1.1. Desarrollo***

El primer objetivo específico busca caracterizar las principales barreras tecnológicas que enfrenta la población con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto. Para ello, se construyó un instrumento de recolección piloto durante las primeras dos semanas del cronograma. En este proceso, se realizó una revisión sistemática de investigaciones anteriores enfocadas a personas con discapacidad visual. Se tomó como principal referencia la investigación de Salas, realizada sobre la misma población en la fundación FUNDONAR, enfocada en el diseño de un material didáctico según los intereses y afinidades de esta población. Dicha investigación incluye un proceso de caracterización basado en preguntas de selección múltiple de única respuesta. Por otro lado, se revisó la investigación de Shera et al., (2021), quien propone un modelo de cuestionario basado en escalas de 1 a 5 o también denominada Likert para recolectar información numérica referente a niveles de dificultad o satisfacción de los usuarios con discapacidad visual. De esta manera, dentro del instrumento de recolección para la presente investigación se incluyeron distintos escenarios donde el usuario deberá evaluar el nivel de dificultad que presenta al realizar ciertas actividades cuando accede a contenidos digitales, con el fin de identificar las principales barreras tecnológicas que enfrenta esta población.

Por otro lado, se incluyeron preguntas de caracterización enfocadas al modo de interacción del usuario con sus dispositivos, con el fin de identificar incluir estos medios en el desarrollo de la estrategia computacional a desarrollar. Se tuvo como referencia el formulario realizado por Kamaghe et al. (2020) que describe los tipos de tecnologías asistivas más utilizadas para personas con discapacidad visual.

Con esta primera versión del instrumento, mostrada en el Anexo B. Instrumento de recolección para prueba piloto, se realizó la prueba piloto, seleccionando cinco personas de la población de manera aleatoria. Esta prueba se llevó a cabo el 9 de septiembre de 2023 desde las instalaciones de

la Fundación FUNDONAR. El instrumento de recolección se aplicó por medio de llamadas telefónicas a cada uno de los pacientes seleccionados. Estas llamadas fueron gestionadas con el respaldo y apoyo de la Dra. Yuli Delgado Solarte, a quien se referenciará en adelante como “especialista en baja visión de FUNDONAR”, quien cuenta con acceso a los datos e historia clínica de los pacientes. Cabe destacar que se utilizó hasta tres bases de datos de la fundación para extraer información como nombre, edad, estrato, diagnóstico y demás datos que se incluyeron en el instrumento de recolección; de manera que no se incluyeron dichas preguntas en las llamadas telefónicas, con el fin de hacer este proceso más ameno para los pacientes. La prueba piloto arrojó un desempeño satisfactorio en cuanto a la recopilación de datos. No obstante, se identificaron áreas de mejora que requieren atención antes de la implementación completa de la investigación. Las principales observaciones y recomendaciones provinieron de la especialista en baja visión de FUNDONAR, el cliente y el asesor del proyecto.

### **Figura 7**

*Evidencia de caracterización de la población*



Se recomendó la incorporación de preguntas adicionales relacionadas con el conocimiento y uso de herramientas digitales específicas por parte de los encuestados, como el uso de correo electrónico, herramientas de Inteligencia artificial, o cursos de educación por internet. Esta ampliación permitirá obtener una visión más completa de las habilidades digitales de los participantes.

Por otro lado, se propuso realizar ajustes en la clasificación de edades de los encuestados. En lugar de incluir todas las edades como opciones, se sugiere agruparlas en rangos de 10 años. Este cambio facilitará el análisis posterior de los datos demográficos y mejorará la precisión en la interpretación de los resultados. También se sugirió revisar y simplificar el lenguaje utilizado en el instrumento de recolección de datos. Esto implicaba evitar jerga técnica y emplear palabras y frases más accesibles y de uso común. Al hacerlo, se buscaba facilitar la participación de los encuestados y asegurar que las preguntas fueran claras y comprensibles para un público diverso.

Finalmente, el cliente sugirió agregar preguntas sobre el interés de la población en utilizar materiales didácticos por medio de contenidos digitales a través de la estrategia computacional a desarrollar, esto con el fin de que el presente proyecto materialice el diseño obtenido como resultado de la investigación de Salas.

Una vez realizados los cambios sugeridos (Ver Anexo C. Instrumento de recolección final para caracterización), se procedió a aplicar el instrumento de recolección final. Este proceso se realizó el día 30 de septiembre de 2023, esta vez desde una clínica externa donde la especialista en baja visión de FUNDONAR realiza consultas particulares en la ciudad de Pasto. Debido a cuestiones de tiempo y dificultad para realizar las encuestas, se recolectó la información de 7 pacientes solamente. Por lo tanto, se tuvo que ampliar las fechas del cronograma destinadas a la recolección de datos. Finalmente, se aplicó el instrumento de recolección a un total de 24 pacientes de la fundación, seleccionados de acuerdo con los criterios de la Tabla 14.

**Tabla 14**

*Criterios de aceptación - Instrumento de recolección*

<b>Criterios de inclusión</b>	<b>Criterios de exclusión</b>
Que tenga discapacidad visual	Que no tenga discapacidad visual
Participación voluntaria	Negarse a participar
Aceptar el consentimiento informado	No aceptar
Personas vinculadas a FUNDONAR	Personas no vinculadas a FUNDONAR
Nacionalidad colombiana	Extranjero(a)

---

Residente en Nariño

Residente fuera de Nariño

---

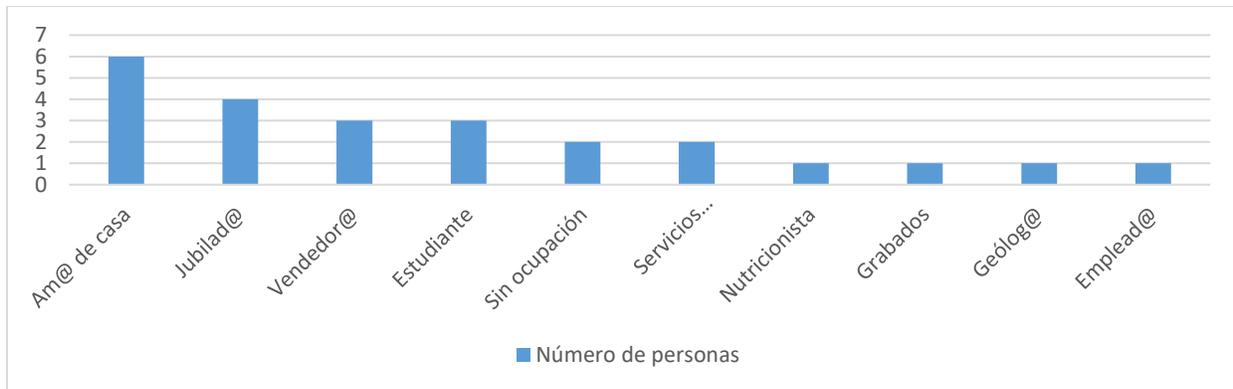
Este proceso tuvo una duración de 3 semanas debido a la dificultad de acceso a los pacientes de la fundación, dado que se necesita la disponibilidad de la especialista en baja visión de FUNDONAR para establecer este contacto. Una vez aplicado el instrumento de recolección final, se realizó un proceso de análisis de datos con base en las respuestas obtenidas. Para esto, se utilizó gráficos generados automáticamente por Google Forms. No obstante, para las preguntas con respuesta abierta se realizó un proceso de limpieza de datos en Python (Ver Anexo D. Análisis de datos – Caracterización FUNDONAR), haciendo uso de bibliotecas orientadas al análisis de datos como “pandas”, “matplotlib” y “nltk”, además de realizar cálculos estadísticos con dichas herramientas. A continuación, se presentan algunos de los resultados más destacados de la encuesta.

En cuanto a los datos sociodemográficos de la población, un 95.8% pertenecen al estrato socioeconómico 1; únicamente una persona pertenece al estrato 3. Por otro lado, un 79.2% de los encuestados son mujeres, mientras que el 20.8% restante son hombres. No hubo registros de personas no-binarias. En la población encuestada, hay 3 personas con ceguera legal y una persona con ceguera total; el resto tienen diagnóstico de baja visión.

Después de realizar un proceso de limpieza y análisis de datos, se tiene que la ocupación más frecuente es “Am@ de casa”, con 6 ocurrencias; seguido de “jubilad@” con 4 y “vendedor@” con 3.

## **Figura 8**

*Distribución de Ocupaciones*



Para analizar el uso de herramientas tecnológicas, se preguntó qué dispositivos utilizan los encuestados para acceder a aplicaciones y contenidos digitales. El 91.3% de los encuestados utiliza smartphone, siendo este el dispositivo más utilizado, seguido por el computador de escritorio (47.8%) y computador portátil (26.1%). Esta información sugiere que los dispositivos móviles son una buena opción como plataforma para la estrategia computacional a desarrollar. Se preguntó también sobre las tecnologías de asistencia utilizadas. De los resultados se destaca que ningún encuestado utiliza dispositivos de braille para acceder a contenidos digitales. Por el contrario, la tecnología de asistencia más utilizada es el magnificador, un dispositivo mecánico que actúa como lupa sobre las pantallas y que permite ampliar el contenido para facilitar su lectura. Únicamente 5 personas utilizan lectores de pantalla (TalkBack o VoiceOver).

Como se observa en la Figura 9, el tipo de aplicaciones más utilizado por los encuestados son de mensajería (65.2%), además de redes sociales (47.8%) y navegación por internet (43.5%). Por lo tanto, se las tomará como referencia al desarrollar la estrategia computacional, así como la manera en la que se aplica la accesibilidad en dichas aplicaciones.

### Figura 9

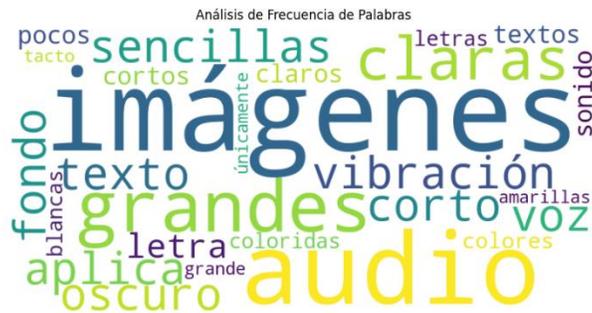
*Aplicaciones utilizadas por la población*



Por otra parte, los métodos de interacción con los dispositivos preferidos por los encuestados son “respuestas por vibración” y “respuestas mediante audio”, dato que se ve reflejado también al analizar la pregunta “Para usted ¿Qué es importante que tenga una aplicación cuando la va a utilizar?”, con la cual se generó la siguiente nube de palabras:

**Figura 10**

*Nube de palabras - Preferencia en aplicaciones*



De la misma pregunta, se obtiene que las características más repetidas son imágenes grandes, audio y vibración; dato a tener en cuenta a la hora de diseñar la estrategia computacional. En ese orden de ideas, es importante también conocer cómo prefiere la población que se presenten los contenidos digitales. La mayoría de encuestados utiliza un tamaño de letra ampliado en su dispositivo. También, un 54.2% prefiere que los contenidos no tengan variedad de colores, sino

que presenten una paleta de colores simple y minimalista, al igual que las imágenes, pues ese mismo porcentaje prefiere que las imágenes no contengan muchos detalles.

## Figura 11

### Preferencias en textos



Sin embargo, los resultados a la pregunta relacionada a la presentación de textos y fondos varían bastante en cada paciente. Según la especialista en baja visión de FUNDONAR, esto ocurre debido a que algunos pacientes presentan condiciones oculares específicas que les hace preferir un tipo de contraste en específico para leer. En consecuencia, se debe tener en cuenta todo este tipo de contrastes a la hora de desarrollar la estrategia computacional.

Las preguntas relacionadas a experiencia en formación indican lo siguiente. Ninguno de los encuestados ha realizado algún curso educativo en internet. Sin embargo, a un 66.7% le gustaría realizar uno. Ante esto, llama la atención el 33.3% restante que no desea realizar cursos educativos en internet. Se realizó una tabla de contingencia en Python, donde se cruzó estas respuestas con algunas variables para obtener información al respecto.

## Tabla 15

### Interés por cursos vs Edad

Edad (años)
-------------

<b>Le gustaría realizar cursos educativos en internet</b>	<b>&lt;10</b>	<b>10-20</b>	<b>20-30</b>	<b>30-40</b>	<b>40-50</b>	<b>&gt;50</b>
<b>No</b>	1	0	0	0	0	7
<b>Si</b>	0	3	1	4	4	4

Se observa que la mayoría de las personas que no le gustaría realizar cursos en internet tienen más de 50 años, mientras que a las personas entre 20 y 50 años si les gustaría realizar este tipo de cursos. Por lo tanto, la estrategia computacional a desarrollar estará enfocada en este rango de edad.

En ese orden de ideas, se realizan más tablas de contingencia para obtener información relevante sobre las personas a las que les gustaría realizar cursos educativos en internet (16 personas). Por ejemplo, 14 personas tienen diagnóstico de baja visión, mientras que las otras 2 personas tienen diagnóstico de ceguera legal. Otro dato interesante, es que el dispositivo más utilizado entre este grupo de personas es el smartphone, y que cinco de estas 16 personas usa lectores de pantalla.

**Tabla 16**

*Asistencia utilizada vs Interés por cursos*

<b>Asistencia Utilizada</b>	<b>Le gustaría realizar cursos educativos en internet</b>	
	<b>No</b>	<b>Si</b>
<b>Lectores de Pantalla</b>	0	5
<b>Magnificador</b>	4	12
<b>No utiliza</b>	4	2

Así mismo, como se muestra en la Tabla 17, a todas las personas encuestadas a las que les gustaría realizar cursos educativos en internet, también les gustaría aprender por medio de juegos interactivos utilizando su dispositivo; lo cual indica que aplicar estrategias de gamificación en la estrategia computacional a desarrollar puede ser una buena opción.

**Tabla 17**

*Interés por cursos vs Interés por juegos interactivos*

<b>Le gustaría aprender por medio de juegos interactivos</b>	<b>Le gustaría realizar cursos educativos en internet</b>	
	<b>No</b>	<b>Si</b>
<b>Si</b>	7	16
<b>No</b>	1	0

### **2.1.2. Discusión**

En síntesis, los resultados de la encuesta proporcionaron información valiosa sobre la población encuestada. Se identificó que las dificultades al acceder a contenidos digitales giran en torno a la forma en que estos contenidos son presentados en pantalla, incluyendo variables como el tamaño de fuente y disposición de colores. Esto hace que cada paciente tenga preferencias específicas a la hora de acceder a contenidos digitales. Por lo tanto, la estrategia computacional a desarrollar debe implementar funcionalidades que permitan modificar el contraste y tamaño de letra del dispositivo, ajustándose al diagnóstico de cada usuario. De esta manera, se garantiza que los contenidos sean accesibles para personas con diversas condiciones visuales.

Se observó un interés creciente por la educación en línea; especialmente en pacientes entre los 10 y 50 años, por lo tanto, es viable materializar el resultado de la investigación de Salas por medio de la estrategia computacional a desarrollar en el presente proyecto. Al desarrollar la estrategia computacional con un enfoque educativo y didáctico, esta puede servir como reemplazo a ejercicios impresos que la especialista en baja visión de FUNDONAR utiliza con sus pacientes, incluyendo el uso de la estrategia computacional a desarrollar en la historia clínica de los pacientes. Además, los pacientes encuestados mostraron interés por el aprendizaje basado en gamificación, de manera que dicho enfoque puede ser efectivo para hacer que los contenidos digitales sean agradables para el usuario. Por otro lado, la estrategia computacional a desarrollar debe tener principalmente funcionalidades de audio y vibración, pues los pacientes encuestados mostraron preferencia por estos dos métodos de interacción con el dispositivo.

Finalmente, teniendo en cuenta los hallazgos a raíz de esta caracterización, se concluye que la estrategia computacional basada a desarrollar tendrá un enfoque educativo, aplicando estrategias de gamificación con estímulos basados en audio y vibración.

## 2.2. Objetivo específico 2

### 2.2.1. Fase de exploración

**2.2.1.1. Historias de usuario.** En esta fase se realizaron reuniones con el docente del programa de mercadeo Mauricio Salas, quien tiene el rol de Cliente dentro del proceso de desarrollo. En dichas reuniones se especificaron los requerimientos de la estrategia computacional a desarrollar, representados por las historias de usuario que se incluyen en el Product Backlog, donde además se indica la prioridad y complejidad de cada una.

**Tabla 18**

*Product Backlog*

No	Historia de Usuario	Aporte [1-10]	Urgencia [1-10]	Prioridad	Complejidad
1	Como usuario con discapacidad visual quiero autenticarme con Google para poder ingresar a la aplicación.	8	10	80	Media
2	Como usuario con discapacidad visual quiero autenticarme con Facebook para poder ingresar a la aplicación.	8	10	80	Media
3	Como usuario con discapacidad visual	7	10	70	Alta

	quiero crear mi cuenta para poder guardar mi progreso en los distintos cursos de la aplicación				
<b>4</b>	Como usuario con discapacidad visual quiero cerrar sesión en la aplicación para proteger mis datos y progreso en los cursos.	5	8	40	Muy baja
<b>5</b>	Como usuario con discapacidad visual quiero consultar todos los cursos disponibles, para poder cursar los que sean de mi interés.	10	8	80	Alta
<b>6</b>	Como usuario con discapacidad visual quiero consultar los contenidos de un curso para aprender por medio del contenido	8	8	64	Media
<b>7</b>	Como usuario con discapacidad visual quiero reproducir videos de un curso para aprender sobre temas de mi interés	8	9	72	Alta
<b>8</b>	Como usuario con discapacidad visual quiero jugar al ahorcado	10	6	60	Media

	con niveles relacionados a la temática de cada curso para afianzar mis conocimientos				
<b>9</b>	Como usuario con discapacidad visual quiero jugar a la trivia con niveles relacionados a la temática de cada curso para afianzar mis conocimientos.	10	6	60	Media
<b>10</b>	Como usuario con discapacidad visual quiero modificar el tamaño de la fuente de la aplicación para visualizar los contenidos de manera clara	10	7	70	Alta
<b>11</b>	Como usuario con discapacidad visual quiero alternar el tema oscuro de la aplicación para visualizar los contenidos de manera cómoda.	10	8	80	Muy alta
<b>12</b>	Como usuario con discapacidad visual quiero alternar el tema de alto contraste de la aplicación para	10	8	80	Muy alta

	visualizar los contenidos de manera cómoda.				
<b>13</b>	Como administrador quiero crear un nuevo curso para poder agregar contenidos a dicho curso	10	5	50	Baja
<b>14</b>	Como administrador quiero crear un nuevo juego de ahorcado para asociarlo a un curso existente	10	5	50	Media
<b>15</b>	Como administrador quiero crear un nuevo juego de trivia para asociarlo a un curso existente	10	5	50	Media
<b>16</b>	Como administrador quiero agregar un nuevo video a un curso existente para que los usuarios con discapacidad visual puedan acceder a dicho video mediante el curso existente.	10	5	50	Alta
<b>17</b>	Como administrador quiero agregar un juego a un curso existente para que los usuarios con	10	5	50	Alta

	discapacidad visual puedan acceder a dicho juego mediante el curso existente.					
18	Como usuario administrador quiero publicar un curso ya creado para que esté disponible en la aplicación.	10	5	50	Media	

**2.2.1.2. Atributos de calidad.** Por otro lado, los requerimientos relacionados con accesibilidad para personas con discapacidad visual fueron elaborados con base en la normativa WCAG 2.1. Dicha normativa se organiza en cuatro principios fundamentales: Perceptible, Operable, Comprensible y Robusto, cada uno de ellos con múltiples normas asociadas. Cabe mencionar que, además, se incluyen pautas para distintos tipos de discapacidades, además de la visual. Por lo tanto, se realizó una selección de las normas a aplicar, teniendo en cuenta los siguientes criterios de aceptación.

**Tabla 19**

*Criterios de aceptación - Normas WCAG 2.1*

<b>Criterios de inclusión</b>	<b>Criterios de exclusión</b>
Que esté enfocada a la discapacidad visual	Que esté enfocada a otro tipo de discapacidades (auditiva, motora, cognitiva)
Que esté enfocada a dispositivos móviles	Que esté enfocada a otro tipo de plataformas, sin incluir a los dispositivos móviles.
Que no incluya tecnologías de asistencia externas al dispositivo móvil (ejemplo: teclas de braille, teclado externo, etc.)	Que incluya tecnologías de asistencia externas al dispositivo móvil (ejemplo: teclas de braille, teclado externo, etc.)

Una vez aplicados los criterios de aceptación, las normas seleccionadas fueron las siguientes:

**Tabla 20**

*Selección de normas WCAG 2.1 según criterios de aceptación*

<b>Categoría</b>	<b>Subcategoría</b>	<b>Norma</b>
<b>Perceptible</b>	1.1. Alternativas de texto	1.1.1. Contenido no textual (AA)
	1.3. Adaptable	1.3.2. Secuencia Significativa (A)
		1.3.3. Características sensoriales (A)
	1.4. Distinguible	1.4.1. Uso del color (A)
		1.4.3. Contraste (AA)
		1.4.4. Redimensionar texto (AA)
		1.4.6. Contraste (AAA)
		1.4.8. Presentación visual (AAA)
		1.4.9. Imágenes de texto (AAA)
		1.4.10. Reflujo (AA)
		1.4.11. Contraste no textual (AA)
<b>Operable</b>		2.4. Navegable
	2.4.10. Encabezados de sección (AAA)	
	2.5. Modalidades de entrada	2.5.3. Etiqueta en nombre (A)
<b>Comprensible</b>	3.2. Predecible	2.5.5. Tamaño de objetivo (AAA)
		3.2.3. Navegación consistente (AA)
<b>Robusto</b>	4.1. Compatible	3.2.4. Identificación consistente (AA)
		4.1.2. Nombre, Rol, Valor (A)
		4.1.3. Mensajes de Estado (AA)

De esta manera, se especifica el atributo de calidad “Accesibilidad”, descrito en la tabla a continuación:

**Tabla 21**

*Atributo de calidad - Accesibilidad*

Atributo de calidad	Propiedades
Accesibilidad	Alternativas de texto [IRNF-001]
	Adaptable [IRNF-002]
	Distinguible [IRNF-003]
	Navegable [IRNF-004]
	Modalidad de entrada [IRNF-005]
	Predecible [IRNF-006]
	Compatible [IRNF-007]

**2.2.1.4. Selección de tecnologías.** Teniendo en cuenta los requerimientos del proyecto, se realiza la selección de tecnologías de manera estratégica, buscando una experiencia de desarrollo fluida en todas las plataformas, aprovechando las fortalezas de cada herramienta utilizada.

**2.2.1.4.1. Flutter.** Framework de desarrollo multiplataforma seleccionado para realizar el desarrollo móvil. Se eligió porque además de tener una documentación bastante completa; tiene múltiples librerías y componentes que agilizan el proceso de desarrollo. Además, es un framework que soporta Hot Reload, es decir que, los cambios realizados en el código se ven reflejados de manera instantánea en el dispositivo móvil o emulador que se esté utilizando. Se elige también para realizar la aplicación web del usuario administrador, que le permite crear y editar cursos para la aplicación móvil, pues ofrece una estructura robusta y bien organizada para el desarrollo frontend.

**2.2.1.4.2. Firebase.** Conjunto de servicios web en la nube que incluyen:

- **Cloud Firestore:** Base de datos no relacional basada en documentos JSON. Al ser una base de datos NoSQL basada en documentos, Cloud Firestore es una elección apropiada para almacenar datos de cursos debido a su flexibilidad y escalabilidad. Además, su integración sencilla con Flutter facilita la manipulación de datos en la aplicación móvil.

- **Firestore Storage:** Bucket para almacenamiento de archivos en la nube. Utilizar Firestore Storage para almacenar contenidos digitales es beneficioso debido a su fácil acceso y gestión de archivos en la nube. Esto asegura una entrega eficiente de recursos multimedia relacionados con los cursos. Además, es de fácil integración con otros servicios de Firestore, como Cloud Firestore.
- **Firestore Authentication:** Servicio que permite autenticar usuarios por medio de distintos métodos, como correo electrónico, número de teléfono, Google, Facebook, Twitter, entre otros. Se escoge como estrategia de autenticación porque es de fácil integración con Flutter, pues ambas tecnologías son desarrolladas por Google. Además, ofrece seguridad y seguridad, las cuales son administradas por Firestore de manera automática.

**2.2.1.5. Propuestas de arquitectura.** Posteriormente, se realiza una selección de arquitecturas para aplicar en el proceso de desarrollo. Como lo recomienda Beck, se consideraron tres opciones distintas.

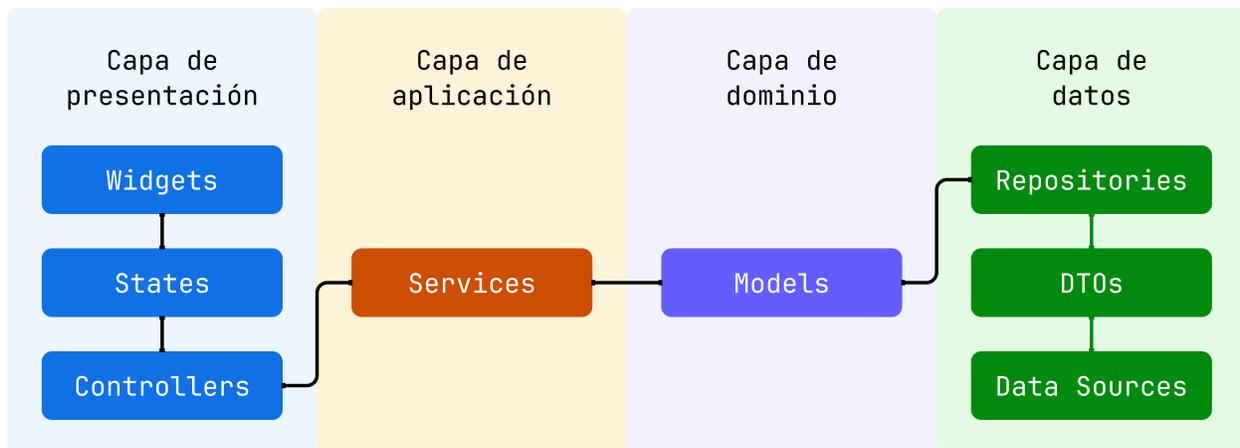
- **Model-View-Controller (MVC).** Patrón de arquitectura introducida en los años 70 con el fin de separar la lógica y la presentación. De acuerdo con (Deacon, 2009), si no se realiza esta separación, cualquier cambio afectaría a la aplicación y a la interfaz de usuario; haciendo que sea menos flexible y mantenible.
- **Clean Architecture.** Arquitectura propuesta por (Martin et al., s/f), que consiste en separar los componentes de la aplicación en cuatro capas, cuyo flujo de información va desde las capas exteriores hasta las capas interiores.
- **Riverpod Architecture.** Arquitectura propuesta por (Bizzotto, 2023), diseñada especialmente para el desarrollo de aplicaciones en Flutter, utilizando la biblioteca Riverpod para el manejo del estado de la aplicación. Toma como base los principios de Clean Architecture.

**2.2.1.6. Arquitectura seleccionada – Riverpod Architecture.** Se elige a Riverpod Architecture debido a su enfoque en aplicaciones de Flutter, siendo diseñada para enfrentar problemáticas específicas que surgen al desarrollar software en este Framework. Además, su documentación posee ejemplos prácticos realizados en Flutter con Dart como lenguaje, lo cual facilita su comprensión e implementación en este proyecto. No obstante, también se tiene en cuenta principios y prácticas mencionadas en las otras dos arquitecturas.

Esta arquitectura comprende cuatro capas: datos, dominio, aplicación y presentación. Esta separación de responsabilidades facilita la incorporación de nuevas funcionalidades al software, ya que solo se requiere modificar la capa correspondiente sin afectar al resto de la aplicación. De igual manera, cada capa es independiente, lo cual permite realizar pruebas unitarias y de integración de manera sencilla. Además, se propone una nomenclatura preestablecida para nombrar a los componentes de cada capa, lo que facilita tareas como la búsqueda de directorios y ficheros en el proceso de desarrollo de software.

**Figura 12**

*Riverpod Architecture*



Nota: Diagrama elaborado con base en la propuesta presentada por (Bizzotto, 2023). Los nombres de los componentes se conservan en inglés por convención.

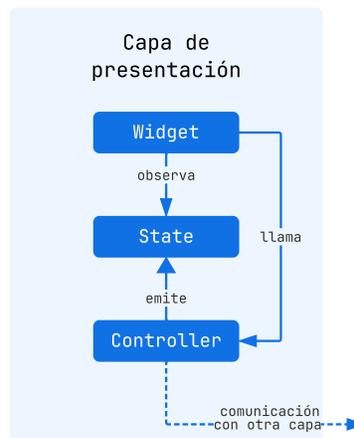
**2.2.1.6.1. Capa de presentación.** También llamada capa de Interfaz de Usuario. Su función principal es mostrar información en la pantalla y servir como punto principal de interacción del usuario. Esta capa consta de tres componentes principales:

- **Widgets.** Son clases de Flutter que representan elementos visuales que aparecen en pantalla. Poseen atributos que permiten personalizar su apariencia y comportamiento. Además, pueden reconstruirse a lo largo del tiempo y observar cambios del estado. Son los encargados de llamar a las funciones de los controladores.
- **Estado (state).** Puede ser cualquier tipo de información, representado por cualquier tipo de dato soportado por el lenguaje de programación Dart.
- **Controladores (controllers).** Expone al estado y posee funciones que pueden actualizarlo. Cuando esto sucede, el widget se vuelve a construir, mostrando la información actualizada.

La Figura 13 ilustra cómo se comunican estos tres componentes

**Figura 13**

*Capa de presentación*



**2.2.1.6.2. Capa de dominio.** Contiene a los modelos, los cuales son clases que representan las entidades que participan en el proceso. Los modelos deben:

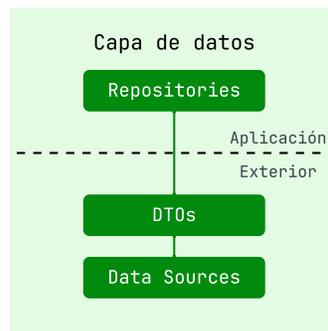
- Ser inmutables, es decir, que sus atributos no se puedan cambiar y que sólo puedan asignarse una vez al instanciar el objeto.
- Contener lógica de serialización, para poder representarse como JSON, o instanciarse a partir de uno.
- Sobrescribir el operador “==”, para que los objetos sean comparados por valor, y no por referencia. Es decir que dos objetos alojados en distintos espacios de memoria son considerados idénticos si sus atributos (o ciertos atributos) son exactamente iguales.

**2.2.1.6.3. Capa de datos.** Se encarga de aislar las operaciones de interacción con bases de datos o APIs externas. Convierten objetos de transferencia (DTOs) en entidades validadas (modelos de la capa de dominio). Opcionalmente pueden realizar operaciones de caching. Esta capa está conformada por:

- **Fuente de Datos (Data Sources).** Es el lugar de donde provienen los datos, como bases de datos, API o servicios externos.
- **Objetos de Transferencia de Datos (DTO).** Son utilizados para transferir información de las fuentes de datos. Generalmente se representan en formato JSON
- **Repositorios.** Utilizados para obtener DTO de las fuentes de datos. Es el medio de comunicación entre la aplicación y las fuentes de datos externas. Contienen operaciones de creación, modificación, lectura y eliminación de datos.

**Figura 14**

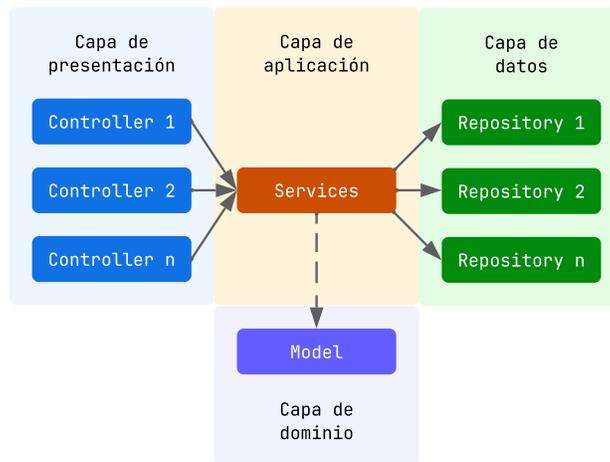
*Capa de datos*



**2.2.1.6.4. Capa de aplicación.** Capa opcional para casos en los que la lógica depende de múltiples repositorios o necesita ser compartida por más de un widget. Esta capa contiene servicios que integran lógica más compleja. En estos casos, distintos controladores pueden llamar a un mismo servicio, que hace uso de uno o más repositorios.

**Figura 15**

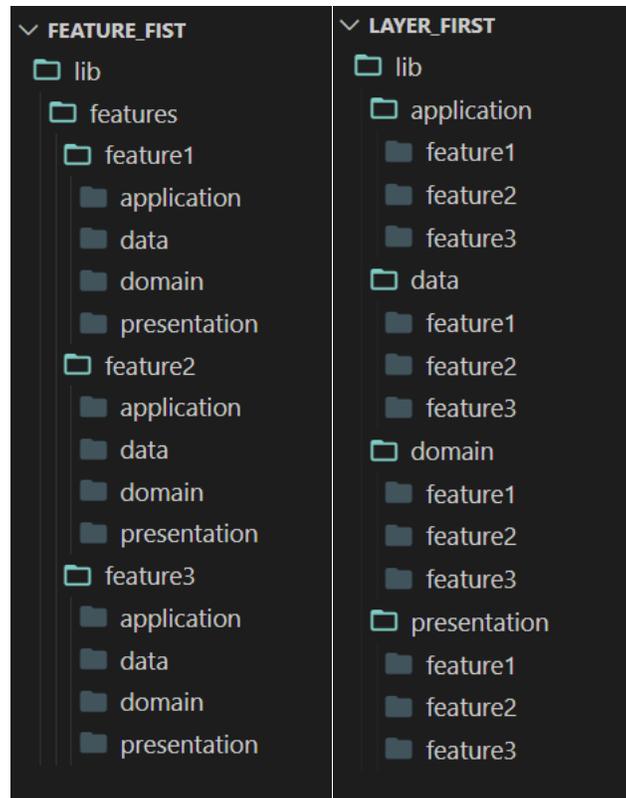
*Capa de aplicación*



**2.2.1.7. Estructura del proyecto.** Una vez seleccionada la arquitectura, se decide cómo estructurar el proyecto, esto garantiza que se siga una convención clara y que las funcionalidades sean agregadas de manera consistente. Por lo tanto, se exploró dos enfoques comunes para realizar dicha estructuración. Por un lado, está el enfoque “layer-first”, que consiste en colocar las funcionalidades dentro de cada capa de la arquitectura; mientras que el enfoque “feature-first” consiste en darle a cada funcionalidad un directorio por separado, donde se aplican las distintas capas de manera independiente.

**Figura 16**

*Comparación de estructura feature-first vs layer-first*



Se optó por aplicar el enfoque “feature-first”, pues se acopla mejor a proyectos con varias funcionalidades, proporcionando una estructura modular donde cada funcionalidad tiene su propio directorio. Además, este enfoque se alinea con el modelo de branching utilizado en el proyecto (GitFlow). Cada rama feature correspondería a una funcionalidad separada del proyecto, de manera que, en su mayoría, los cambios se aplicarían dentro de un solo directorio. Otro beneficio es que facilita la diagramación e interpretación de los diagramas UML para cada funcionalidad por separado, haciendo que cada diagrama corresponda a una funcionalidad.

### 2.2.2. Fase de planeación

En esta fase se realizó la siguiente tabla indicando las historias de usuario a desarrollar, así como las propiedades del atributo de calidad “Accesibilidad”.

**Tabla 22**

*Distribución de requerimientos en iteraciones*

<b>Iteración</b>	<b>Historias de Usuario</b>	<b>Requerimientos no funcionales</b>	<b>Duración</b>
1	HU-001, HU-002, HU-003, HU-004	-	3 semanas
2	HU-010, HU-011, HU-012	IRNF-003	1 semana
3	HU-005, HU-006	IRNF-001, IRNF-002	1 semana
4	HU-007, HU-008, HU-009	IRNF-004, IRNF-005, IRNF-006, IRNF-007	2 semanas
5	HU-013, HU-014, HU-015	-	1 semana
6	HU-016, HU-017, HU-018	-	1 semana

### **2.2.3. Fase de iteraciones**

Cada iteración tuvo cuatro etapas: análisis, diseño, codificación y pruebas.

**2.2.3.1. Primera iteración.** La primera iteración del proceso de desarrollo se enfocó en implementar las historias de usuario relacionadas a la autenticación. En este contexto, los cambios referentes a esta iteración fueron implementados en la rama “feature/auth”.

**2.2.3.1.1. Etapa de análisis.** Se llevó a cabo la especificación de las historias de usuario seleccionadas para esta iteración. En este proceso se definieron los criterios de aceptación asociados a cada historia de usuarios, los cuales serán utilizados posteriormente para elaborar las pruebas unitarias y de integración.

### **Tabla 23**

*Especificación historia de usuario 1*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-001</b>		
<b>Nombre:</b>	Autenticación con Google		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero autenticarme con Google para poder ingresar a la aplicación.		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el usuario tenga una cuenta asociada con las credenciales de Google ingresadas	Entonces se debe permitir el ingreso a la aplicación
	2	Cuando el usuario no tenga una cuenta asociada con las credenciales de Google ingresadas	Entonces, se debe redirigir a la pantalla de Registro.
	3	Cuando la autenticación con Google falle	Entonces, se debe redirigir a la pantalla de Login.
	4	Cuando el usuario cancele la autenticación con Google	Entonces, se debe redirigir a la pantalla de Login.

**Tabla 24**

*Especificación historia de usuario 2*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-002</b>		
<b>Nombre:</b>	Autenticación con Facebook		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero autenticarme con Facebook para poder ingresar a la aplicación.		
	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>

<b>Criterios de Aceptación</b>	1	Cuando el usuario tenga una cuenta asociada con las credenciales de Facebook ingresadas	Entonces se debe permitir el ingreso a la aplicación
	2	Cuando el usuario no tenga una cuenta asociada con las credenciales de Facebook ingresadas	Entonces, se debe redirigir a la pantalla de registro.
	3	Cuando la autenticación con Facebook falle	Entonces, se debe redirigir a la pantalla de Login.
	4	Cuando el usuario cancele la autenticación con Facebook	Entonces, se debe redirigir a la pantalla de Login.

**Tabla 25**

*Especificación historia de usuario 3*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-003</b>		
<b>Nombre:</b>	Crear Cuenta		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero crear mi cuenta para poder guardar mi progreso en los distintos cursos de la aplicación		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el usuario ingrese los campos “nombre”, “edad” y “género” correctamente.	Entonces se debe crear una cuenta nueva en la base de datos.
	2	Cuando el usuario deje alguno de los campos vacío	Entonces se debe mostrar un mensaje de alerta indicando que todos los campos son requeridos

**Tabla 26**

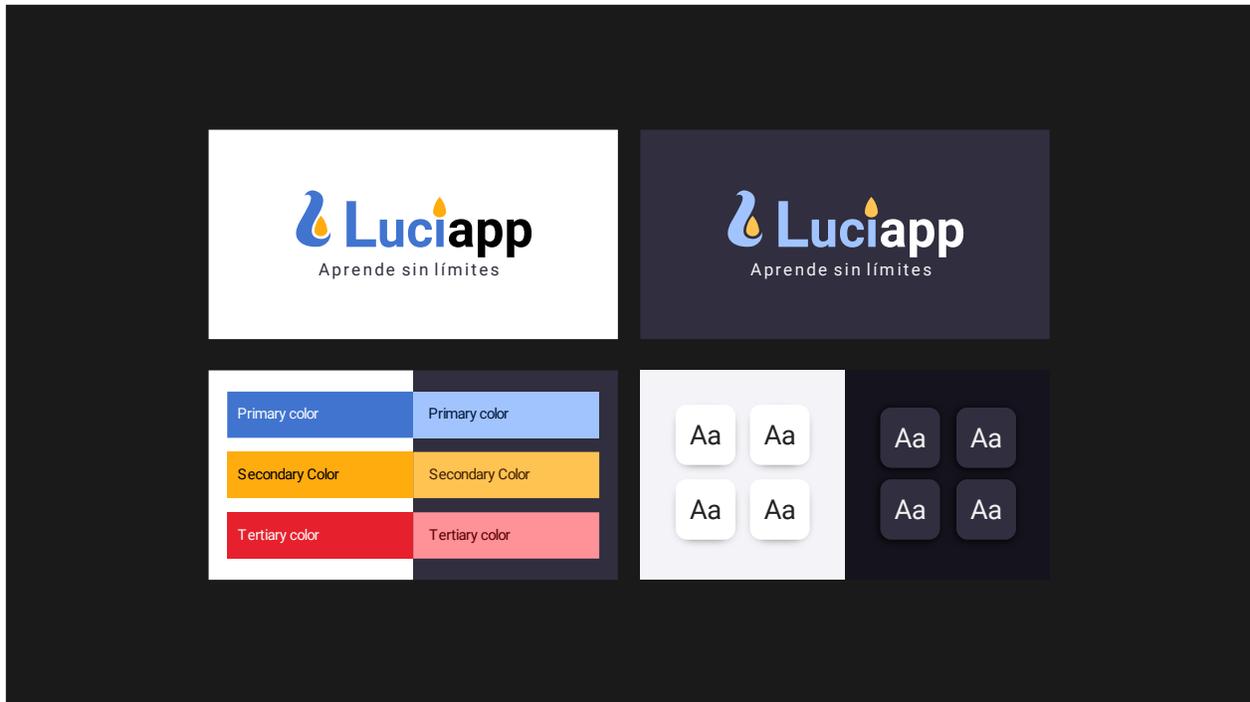
*Especificación historia de usuario 4*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-004</b>		
<b>Nombre:</b>	Cerrar Sesión		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero cerrar sesión en la aplicación para proteger mis datos y progreso en los cursos.		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el usuario seleccione la opción de cerrar sesión	Se debe redirigir a la pantalla de autenticación

**2.2.3.1.2. Etapa de diseño.** Antes de elaborar las interfaces de usuario para la funcionalidad de autenticación, se realizó la selección de paleta de colores de acuerdo con la normativa WCAG 2.1. así como el diseño del logotipo de la aplicación, buscando transmitir la esencia y el propósito del proyecto de manera gráfica. Estos recursos se elaboran para ser utilizados en la elaboración de interfaces de usuario en futuras iteraciones.

**Figura 17**

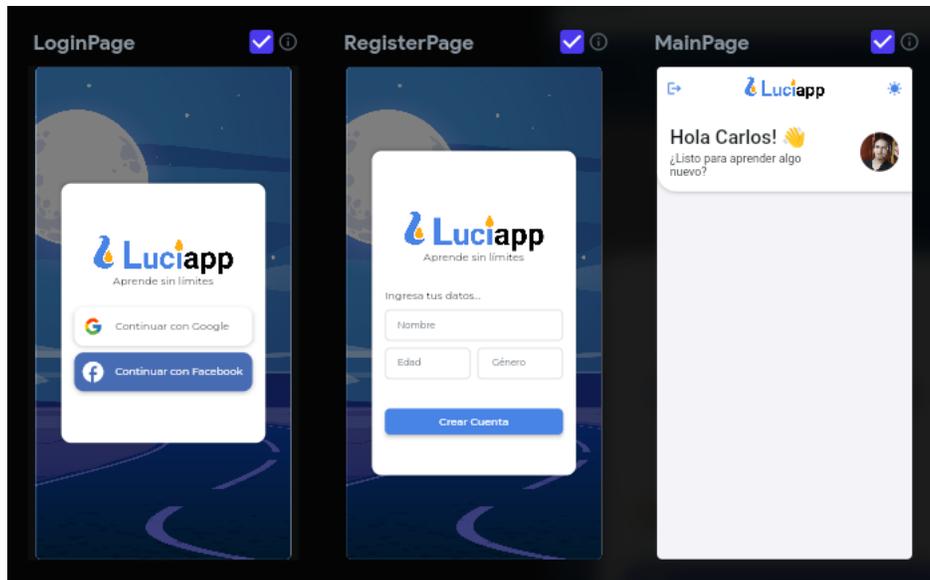
*Selección de colores y diseño de logotipo*



Luego, se realizó el diseño de las interfaces de usuario. Para ello se utilizó la herramienta FlutterFlow debido a que permite utilizar widgets propios de Flutter y agregarlos al diseño por medio de una interfaz gráfica que permite situar los componentes en pantalla de manera visual, lo cual resulta en un diseño más ágil y acorde a las herramientas que ofrece el Framework. Para esta iteración se realiza únicamente el diseño en modo claro. Posteriormente, al desarrollar funcionalidades relacionadas a temas de color, se diseñará la respectiva versión en tema oscuro y de alto contraste de estas interfaces gráficas.

### **Figura 18**

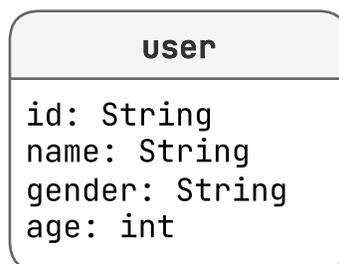
*Diseño de interfaces de usuario para autenticación*



Una vez elaboradas las interfaces de usuario, se inició con el diseño la base de datos. Para esta funcionalidad se eligió utilizar la base de datos no relacional Firebase Firestore, definiendo la estructura de los documentos que se almacenarían en la colección de usuarios. Se consideró pertinente incluir pocos atributos para que el proceso de registro sea sencillo, recopilando únicamente la información necesaria para brindar una experiencia de usuario personalizada.

### Figura 19

*Estructura de documentos para colección de usuarios*

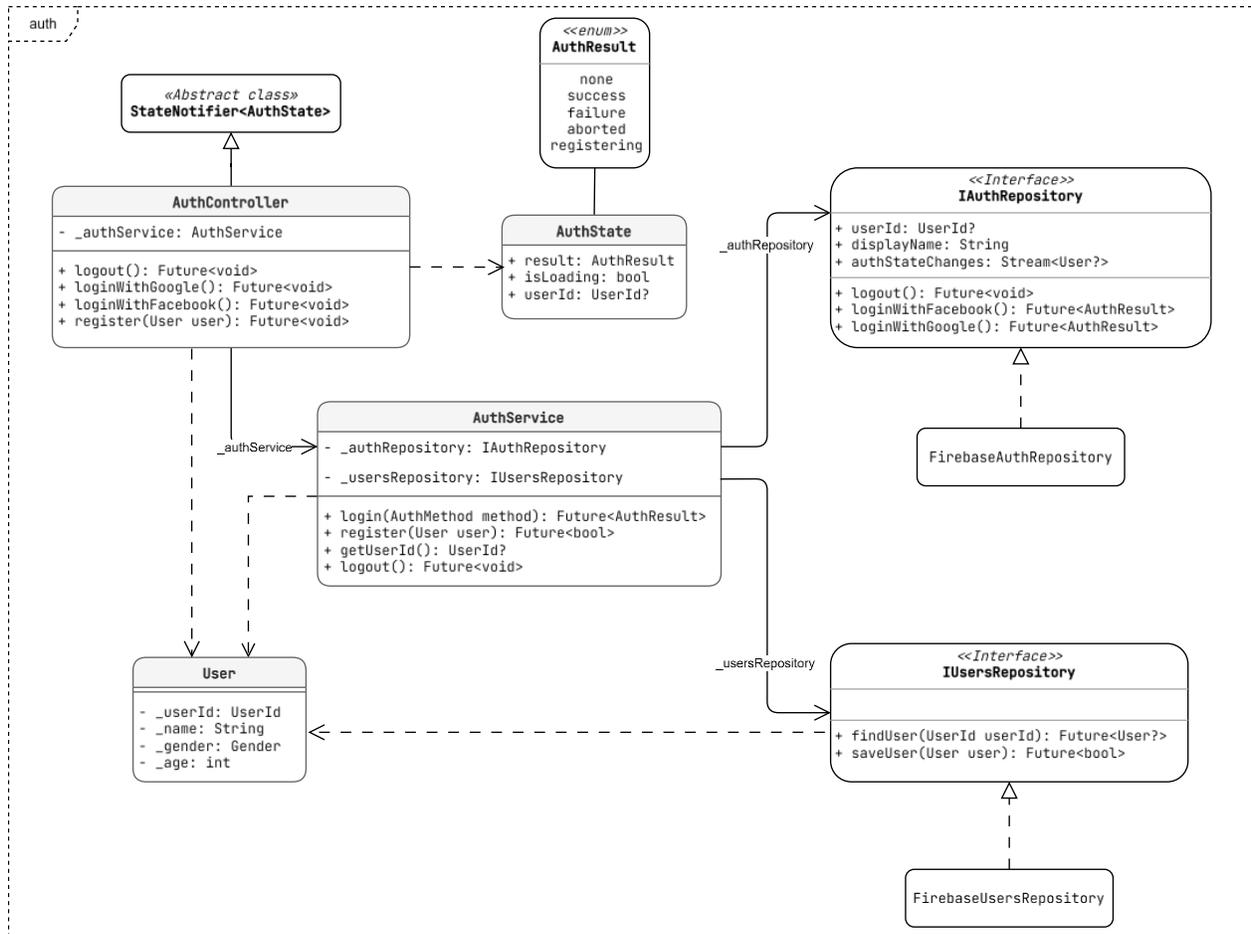


Posteriormente, se realizó el diagrama UML, donde se ilustran las clases a implementar en el sistema, siguiendo la arquitectura Riverpod. Es importante mencionar que se omiten las clases widgets debido a que son elementos efímeros que representan la interfaz de usuario y están destinados a cambiar en respuesta a eventos y cambios en la aplicación. Además, la jerarquía de

clases en Flutter, especialmente en los widgets, es altamente compleja, por lo que incluirlos en el diagrama UML dificultaría su interpretación.

**Figura 20**

*Diagrama UML para funcionalidad de autenticación*



Cómo se observa en el diagrama, se utiliza una clase de servicio dentro de la capa de aplicación, debido a que este caso de uso específico requiere dos repositorios para funcionar. Por un lado, está el repositorio de autenticación, el cual tiene como objetivo conectarse directamente con el datasource de autenticación (FirebaseAuth). Por otro lado, está el repositorio de usuarios, que se comunica con la base de datos para usuarios (Firebase Firestore)

Ambos repositorios implementan interfaces (clases abstractas en Dart), las cuales definen los contratos que los repositorios concretos deberán implementar. De esta manera se logra desacoplar las fuentes de datos del resto de la aplicación. La clase AuthService depende de abstracciones de los repositorios, y no de sus implementaciones. Luego, es cuestión de definir dichas implementaciones por medio de inyección de dependencias para elegir qué tipo de repositorio concreto utilizar en cada caso. Por ejemplo, de ser necesario cambiar la base de datos de Firebase por una base de datos MongoDB, solo habría que crear una clase que implemente la interfaz IUsersRepository e inyectarla como dependencia a la clase AuthService, sin modificar en lo absoluto esta última clase.

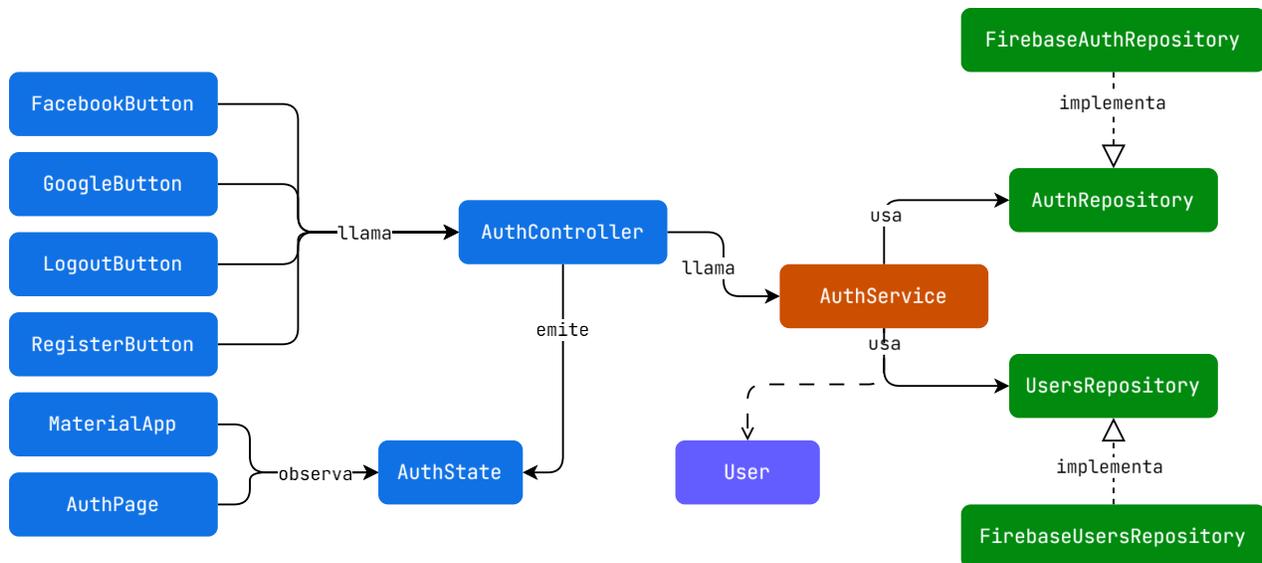
De esta manera concluye la etapa de diseño. La selección de colores, con el diseño de las interfaces de usuario, el diseño de base de datos y el diseño de diagrama UML se establecen bases sólidas para iniciar la fase de codificación.

**2.2.3.1.3. Etapa de codificación.** Siguiendo el enfoque feature-first, se almacenó todos los elementos relacionados a la funcionalidad de autenticación en la carpeta “auth” del proyecto. El primer paso de esta fase fue la implementación de las interfaces de usuario. Se abordó este proceso con un diseño modular, buscando abstraer funcionalidades para evitar la repetición de código y siguiendo el principio de responsabilidad única. Así, los componentes se organizaron en widgets separados, lo cual permite su reutilización y facilita la legibilidad del código.

Luego se procede a implementar las distintas capas de la arquitectura seleccionada, comenzando desde los repositorios hasta el controlador. El siguiente diagrama ilustra la comunicación de los componentes de esta funcionalidad. A diferencia del diagrama UML, en este caso si se incluyen los widgets de Flutter, indicando únicamente su nombre. Para esta funcionalidad, se crearon cuatro widgets de tipo botón y una página que alberga los botones y otros componentes. Material App es la aplicación entera de Flutter.

## **Figura 21**

Comunicación de componentes - Autenticación



Como se puede observar, el flujo comienza con la interacción del usuario con los componentes de botones. Al ser presionados, estos llaman a las funciones del controlador AuthController, cuyo propósito es llamar a la función correspondiente en la clase de servicio AuthService, donde se aplica la lógica de negocio y se utilizan los repositorios necesarios. Es importante recalcar que las clases de controladores no almacenan lógica compleja; simplemente redirigen a la función del servicio correspondiente. Por otro lado, la otra función del controlador es emitir el estado (AuthState) que será observado por otros widgets.

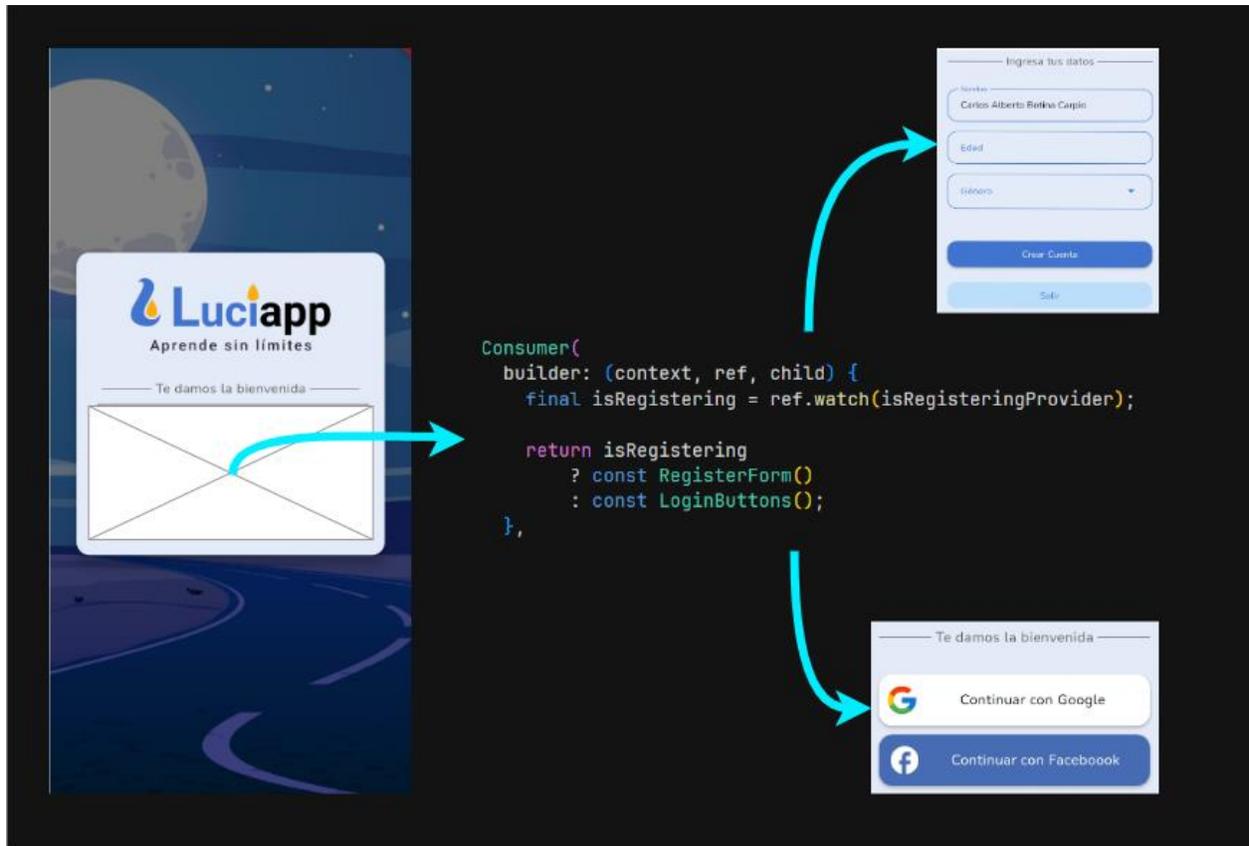
Como se mostró en la Figura 20, la clase AuthState tiene tres atributos, el resultado de autenticación, un booleano que indica si hay un proceso en ejecución o no, y el id del usuario en sesión. Cuando el estado cambia, los widgets que estén suscritos a dicho estado (o a sus atributos) se vuelven a renderizar instantáneamente. Un ejemplo de esto es el widget MaterialApp, que corresponde a la aplicación entera de Flutter. Este widget observa el resultado de autenticación del estado AuthState. Si este resultado es satisfactorio, se mostrará la página de inicio MainPage; de otro modo, se mostrará la página de autenticación AuthPage.

Lo mismo ocurre con el widget AuthPage, que renderiza los botones de login o el formulario del registro, dependiendo del estado de autenticación del usuario. Esta forma de construir la interfaz de usuario hace que únicamente se vuelva a renderizar el componente suscrito al estado, haciendo

que componentes como la imagen de fondo o el logotipo no se vuelvan a renderizar innecesariamente. De esta manera se mejora el rendimiento en general de la aplicación.

**Figura 22**

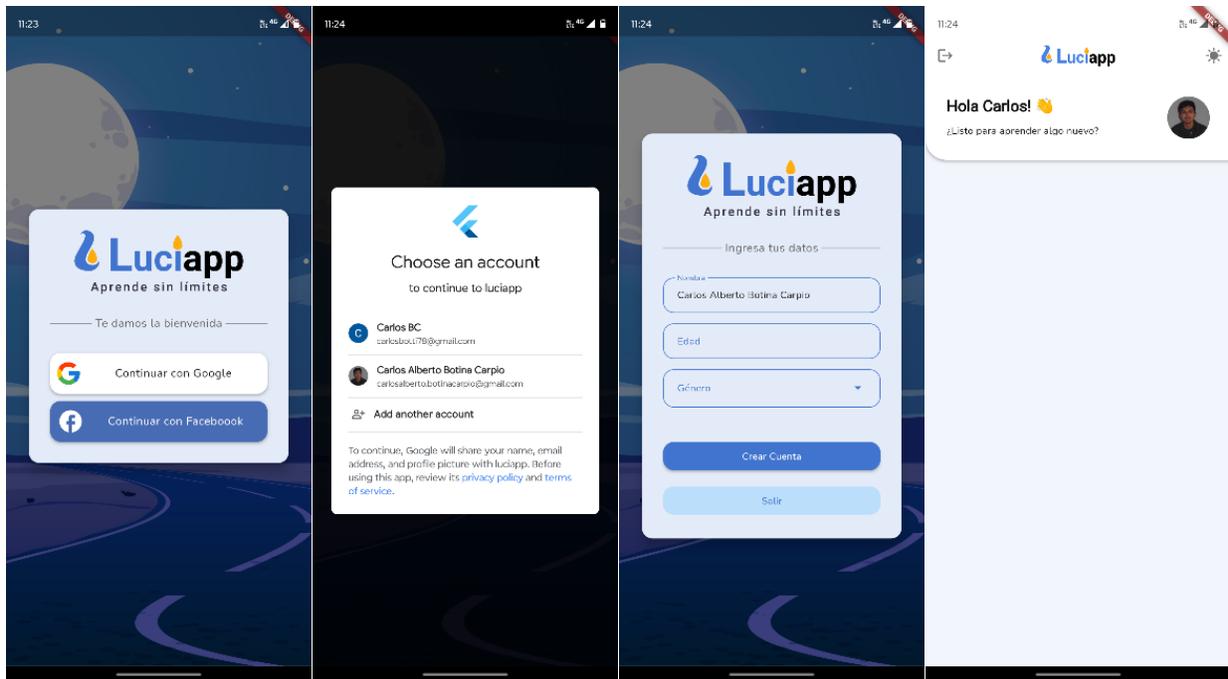
*Renderizado condicional en página de autenticación*



Finalmente, se termina de implementar el sistema de autenticación con los proveedores Google y Facebook. Al utilizar estos proveedores, se prescinde de procesos como verificación de correos o creación de contraseñas. De esta manera, se obtuvo un flujo ágil y sencillo, como se observa en la siguiente figura.

**Figura 23**

*Proceso de registro en la aplicación*



**2.2.3.1.4. Etapa de pruebas.** Una vez terminada la etapa de codificación, donde la aplicación ya tiene un funcionamiento correcto, se procede a elaborar las pruebas unitarias y de integración para verificar que cada uno de los criterios de aceptación para las historias de usuario se cumplan de manera satisfactoria. Es importante realizar estas pruebas para garantizar la calidad y la fiabilidad del software desarrollado.

Las pruebas unitarias permiten evaluar el funcionamiento individual de cada componente o unidad de código, asegurando que estas unidades operen correctamente de manera aislada. Para la funcionalidad de autenticación, se realiza pruebas unitarias a cada uno de los métodos de las clases AuthController y AuthService. Como estas clases poseen dependencias, es necesario “mockear” estas dependencias para simular su comportamiento. Un “mock” es una implementación que replica el comportamiento de un componente en concreto de manera controlada. Esto permite mantener la independencia de las pruebas unitarias.

## Figura 24

*Ejemplo de mocking para pruebas unitarias*

```

void main() {
    late MockAuthService mockAuthService;

    setUp(() {
        mockAuthService = MockAuthService();
    });
    group(TestNames.unitTest, () {
        test(TestNames.cp001, () async {
            when(() => mockAuthService.login(AuthMethod.google))
                .thenAnswer((_) => Future.value(AuthResult.success));

            when(mockAuthService.getUserId).thenAnswer((_) => '1234');

            final controller = AuthController(authService: mockAuthService);

            expect(controller.state, const AuthState.unknown());
        });
    });
}

```

La Figura 24. muestra una prueba unitaria elaborada para la clase AuthController. Como se observa en el diagrama UML (Figura 20), AuthController tiene una sola dependencia, que es AuthService. En este ejemplo se hace uso de la librería “mocktail” para realizar el mock de AuthService. Posteriormente, se sobrescribe su comportamiento cuando se llama a la función login y al getter getUserId. Luego, se inyecta a este mock como dependencia para AuthController, en lugar de usar la implementación real de AuthService. Finalmente, se procede con la elaboración de la prueba unitaria.

Así mismo se realiza el resto de las pruebas, asociando cada una a una historia de usuario. Las siguientes tablas describen las pruebas unitarias realizadas para las clases AuthController y AuthService, haciendo uso del artefacto mostrado en la Tabla 8.

, donde se especifica las características de la prueba y los resultados esperados para que sea considerada como aprobada. Las pruebas se elaboraron teniendo en cuenta los posibles escenarios que pueden surgir al ejecutar todos los métodos de las clases mencionadas.

### Tabla 27

*Pruebas unitarias para la clase AuthController*

<b>Id</b>	<b>Nombre</b>	<b>Clase / Método</b>	<b>HU</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-001	Autenticación exitosa con Google	AuthController / loginWithGoogle	HU-001	-	AuthState con resultado exitoso, cargando falso y el id del usuario
CP-002	Fallo en el Autenticación con Google	AuthController / loginWithGoogle	HU-001	-	AuthState con resultado de fallo, cargando falso y userId nulo
CP-003	Aborto en Autenticación con Google	AuthController / loginWithGoogle	HU-001	-	AuthState con resultado de aborto, cargando falso y userId nulo
CP-004	Redirección a Registro desde Google	AuthController / loginWithGoogle	HU-001	-	AuthState con resultado de registro, cargando falso y el id del usuario
CP-005	Autenticación exitosa con Facebook	AuthController / loginWithFacebook	HU-002	-	AuthState con resultado exitoso, cargando falso y el id del usuario

CP-006	Fallo en Autenticación con Facebook	AuthController / loginWithFacebook	HU-002	-	AuthState con resultado de fallo, cargando falso y userId nulo
CP-007	Aborto en Autenticación con Facebook	AuthController / loginWithFacebook	HU-002	-	AuthState con resultado de aborto, cargando falso y userId nulo
CP-008	Redirección a Registro desde Facebook	AuthController / loginWithFacebook	HU-002	-	AuthState con resultado de registro, cargando falso y el id del usuario
CP-009	Registro exitoso	AuthController / register	HU-003	Instancia de Usuario	AuthState con resultado exitoso, cargando falso y el id del usuario
CP-010	Fallo en el registro	AuthController / register	HU-003	Instancia de Usuario	AuthState con resultado de fallo, cargando falso y userId nulo
CP-011	Éxito en el cierre de sesión	AuthController / logOut	HU-004	-	AuthState desconocido

**Tabla 28***Pruebas unitarias para la clase AuthService*

Id	Nombre	Clase/Método	HU	Entrada	Resultado esperado
CP-012	Éxito en el inicio de sesión con Google (nuevo usuario)	AuthService / login	HU-001	AuthMethod. google	AuthResult .registering
CP-013	Éxito en el inicio de sesión con Google (usuario existente)	AuthService / login	HU-001	AuthMethod. google	AuthResult .success
CP-014	Fallo en el inicio de sesión con Google	AuthService / login	HU-001	AuthMethod. google	AuthResult .failure
CP-015	Aborto en el inicio de sesión con Google	AuthService / login	HU-001	AuthMethod. google	AuthResult .aborted
CP-016	Éxito en el inicio de sesión con Facebook (nuevo usuario)	AuthService / login	HU-002	AuthMethod. facebook	AuthResult .registering
CP-017	Éxito en el inicio de sesión con Facebook (usuario existente)	AuthService / login	HU-002	AuthMethod. facebook	AuthResult .success
CP-018	Fallo en el inicio de sesión con Facebook	AuthService / login	HU-002	AuthMethod. facebook	AuthResult .failure

CP-019	Aborto en el inicio de sesión con Facebook	AuthService / login	HU-002	AuthMethod. facebook	AuthResult .aborted
CP-020	Éxito en el cierre de sesión	AuthService / logOut	HU-004	-	AuthRepository.logOut llamado una vez
CP-021	Éxito en el registro	AuthService / register	HU-003	Instancia de Usuario	true
CP-022	Fallo en el registro	AuthService / register	HU-003	Instancia de Usuario	false
CP-023	Obtener userId existente	AuthService / getUserId	HU-001, HU-002	-	Id de usuario en sesión
CP-024	Obtener userId nulo	AuthService / getUserId	HU-001, HU-002	-	null

Es posible que al ejecutar las pruebas existan fallos en algunas de ellas, esto alerta al desarrollador y hace que se revise la clase y el método específico donde se encuentra el error. También es posible que la prueba esté mal elaborada. En ambos casos, se requerirá de varias revisiones hasta que el código de la aplicación y las pruebas unitarias coincidan y verifiquen que existe un correcto funcionamiento.

### Figura 25

*Resultados de pruebas unitarias para la clase AuthController*

```
Carlos@ZENBOOK MINGW64 /c/Projects/Luciapp (feature/auth)
$ flutter test test/auth/unit_testing/auth_controller_test.dart -r expanded
00:00 +0: loading C:/Projects/Luciapp/test/auth/unit_testing/auth_controller_test.dart
00:00 +0: Prueba Unitaria [CP-001] Autenticación exitosa con Google
00:00 +1: Prueba Unitaria [CP-002] Fallo en la Autenticación con Google
00:00 +2: Prueba Unitaria [CP-003] Aborto en Autenticación con Google
00:00 +3: Prueba Unitaria [CP-004] Redirección a Registro desde Google
00:00 +4: Prueba Unitaria [CP-005] Autenticación exitosa con Facebook
00:00 +5: Prueba Unitaria [CP-006] Fallo en Autenticación con Facebook
00:00 +6: Prueba Unitaria [CP-007] Aborto en Autenticación con Facebook
00:00 +7: Prueba Unitaria [CP-008] Redirección a Registro desde Facebook
00:00 +8: Prueba Unitaria [CP-009] Registro exitoso
00:00 +9: Prueba Unitaria [CP-010] Fallo en el registro
00:00 +10: Prueba Unitaria [CP-011] Éxito en el cierre de sesión
00:00 +11: All tests passed!
```

**Figura 26**

*Resultado de pruebas unitarias para la clase AuthService*

```
Carlos@ZENBOOK MINGW64 /c/Projects/Luciapp (feature/auth)
$ flutter test test/auth/unit_testing/auth_service_test.dart -r expanded
00:00 +0: loading C:/Projects/Luciapp/test/auth/unit_testing/auth_service_test.dart
00:00 +0: Prueba Unitaria [CP-012] Éxito en el inicio de sesión con Google (nuevo usuario)
00:00 +1: Prueba Unitaria [CP-013] Éxito en el inicio de sesión con Google (usuario existente)
00:00 +2: Prueba Unitaria [CP-014] Fallo en el inicio de sesión con Google
00:00 +3: Prueba Unitaria [CP-015] Aborto en el inicio de sesión con Google
00:00 +4: Prueba Unitaria [CP-016] Éxito en el inicio de sesión con Facebook (nuevo usuario)
00:00 +5: Prueba Unitaria [CP-017] Éxito en el inicio de sesión con Facebook (usuario existente)
00:00 +6: Prueba Unitaria [CP-018] Fallo en el inicio de sesión con Facebook
00:00 +7: Prueba Unitaria [CP-019] Aborto en el inicio de sesión con Facebook
00:00 +8: Prueba Unitaria [CP-020] Éxito en el cierre de sesión
00:00 +9: Prueba Unitaria [CP-021] Éxito en el registro
00:00 +10: Prueba Unitaria [CP-022] Fallo en el registro
00:00 +11: Prueba Unitaria [CP-023] Obtener userId existente
00:00 +12: Prueba Unitaria [CP-024] Obtener userId nulo
00:00 +13: All tests passed!
```

Por otro lado, las pruebas de integración examinan cómo estos componentes se combinan y trabajan juntos, verificando que la aplicación en su conjunto cumpla con los requisitos establecidos. Flutter ofrece la posibilidad de realizar estas automáticas en un dispositivo móvil real, simulando acciones del usuario como toques, entradas de texto, desplazamientos, entre otros, de manera automática. Esto permite automatizar procesos que anteriormente se realizaban de manera manual, ejecutando pruebas complejas en cuestión de segundos.

Si bien es posible ejecutar la aplicación completa en este tipo de pruebas, se realizó mocking para las dependencias externas de la aplicación, concretamente los servicios de Firebase, para simular inicios de sesión, registros y otras acciones. Así, se evita la necesidad de consumir recursos de Firebase de manera innecesaria, además de lograr un ambiente de prueba controlado y predecible.

La siguiente tabla muestra las pruebas de integración construidas para la funcionalidad de autenticación. Cabe mencionar que las pruebas integración no solo van asociadas a historias de usuario, sino que además se asocian directamente a los criterios de aceptación que las validan.

**Tabla 29**

*Pruebas de integración para funcionalidad de autenticación*

<b>Id</b>	<b>Nombre</b>	<b>CID</b>	<b>Descripción escenario</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-025	Usuario autenticado y registrado	HU001-C1 HU002-C1	El usuario tiene una sesión activa y una cuenta creada	-	Mostrar pantalla de Inicio
CP-026	Usuario en proceso de registro	HU001-C2 HU002-C2	El usuario completa el proceso de autenticación, pero aún no tiene una cuenta creada	-	Mostrar pantalla de Registro
CP-027	Fallo en proceso de autenticación	HU001-C3 HU002-C3	El proceso de autenticación falló	-	Mostrar pantalla de Login
CP-028	Proceso de autenticación cancelado	HU001-C4	El usuario cancela el	-	Mostrar pantalla de Login

		HU002-C4	proceso de autenticación		
CP-029	Registro exitoso	HU003-C1	Todos los campos están diligenciados en la pantalla de Registro	Usuario presiona botón de crear cuenta	Método register de AuthController llamado una vez
CP-030	Registro con campos vacíos	HU003-C2	Existen campos sin diligenciar en la pantalla de Registro	Usuario presiona botón de crear cuenta	Mostrar mensaje indicando que el campo faltante es requerido
CP-031	Cierre de sesión	HU004-C1	El usuario cierra sesión	Usuario presiona botón de cerrar sesión	Mostrar pantalla de Login. Pantalla de Home no debe mostrarse

Al final, se obtienen los siguientes resultados al ejecutar las pruebas de integración. Como se observa, la ejecución de este tipo de pruebas requiere una mayor cantidad de tiempo para ejecutarse, en comparación de las pruebas unitarias. Mientras un grupo de pruebas unitarias se ejecuta en alrededor de un segundo, un grupo de pruebas de integración puede tardar hasta un minuto en ejecutarse.

Esto sucede debido a que la ejecución de las pruebas de integración incluye procesos de compilación e instalación de la aplicación, los cuales requieren un tiempo que varía dependiendo de la complejidad del proyecto; a mayor cantidad de funcionalidades incluidas en la prueba, mayor es el tiempo requerido para realizar estos procesos. Nótese como en la siguiente figura los procesos de compilación e instalación toman alrededor de 27 segundos, mientras que en posteriores pruebas este tiempo asciende hasta los 40 segundos (Ver Figura 65.

).

**Figura 27**

### Resultados de pruebas de integración para funcionalidad de autenticación

```
Carlos@ZENBOOK MINGW64 /c/Projects/Luciapp (feature/auth)
$ flutter test integration_test/test.dart -d ZT322K96PJ -r expanded
00:00 +0: loading C:/Projects/Luciapp/integration_test/test.dart
Running Gradle task 'assembleDebug'...                               20.6s
✓ Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app-debug.apk...          7.1s
00:00 +0: (setUpAll)
00:00 +0: Prueba de Integración [CP-025] Usuario autenticado y registrado
00:01 +1: Prueba de Integración [CP-026] Usuario en proceso de registro
00:02 +2: Prueba de Integración [CP-027] Fallo en proceso de autenticación
00:02 +3: Prueba de Integración [CP-028] Proceso de autenticación cancelado
00:02 +4: Prueba de Integración [CP-029] Registro exitoso
00:03 +5: Prueba de Integración [CP-030] Registro con campos vacíos
00:04 +6: Prueba de Integración [CP-031] Cierre de sesión
00:05 +7: (tearDownAll)
00:05 +7: All tests passed!
```

Así concluye la primera iteración del proceso de desarrollo, donde se trabajó en las historias de usuario HU-001, HU-002, HU-003 y HU-004, logrando la funcionalidad de autenticación con los proveedores Google y Facebook, así como la conexión a la base de datos de Firebase, permitiendo así un proceso fluido de registro e inicio de sesión.

**2.2.3.2. Segunda iteración.** Para la segunda iteración se seleccionó las historias de usuario HU-010, relacionada con el cambio de tamaño de fuente y HU-011, HU-012, relacionada con el cambio de temas de color. Adicionalmente, como en este punto del desarrollo ya se cuenta con una interfaz gráfica considerable, se inició con la implementación del atributo de calidad Accesibilidad, concretamente, con el requerimiento no funcional IRNF-003 (Distinguishable) Se decidió desarrollarlas en la segunda iteración porque de estas depende la apariencia de la aplicación entera, incluyendo funcionalidades posteriores.

**2.2.3.2.1. Etapa de análisis.** Se realizó la especificación de las historias de usuario, que se muestran en la siguiente tabla

**Tabla 30**

*Especificación historia de usuario 10*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-010</b>		
<b>Nombre:</b>	Modificar tamaño de fuente		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero modificar el tamaño de la fuente de la aplicación para visualizar los contenidos de manera clara		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el usuario seleccione la opción “reducir tamaño de fuente” y el tamaño actual es mayor al mínimo tamaño permitido	Se debe reducir el tamaño de fuente a lo largo de toda la aplicación
	2	Cuando el usuario seleccione la opción “reducir tamaño de fuente” y el tamaño actual es igual al mínimo tamaño permitido	El tamaño de fuente no debe cambiar a lo largo de la aplicación
	3	Cuando el usuario seleccione la opción “aumentar tamaño de fuente” y el tamaño actual es menor al máximo tamaño permitido	Se debe ampliar el tamaño de fuente a lo largo de toda la aplicación

4	Cuando el usuario seleccione la opción “aumentar tamaño de fuente” y el tamaño actual es igual al máximo tamaño permitido	El tamaño de fuente no debe cambiar a lo largo de la aplicación
---	---------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------

**Tabla 31**

*Especificación historia de usuario 11*

<b>Historia de Usuario</b>		
<b>Código:</b>	<b>HU-011</b>	
<b>Nombre:</b>	Alternar tema oscuro	
<b>Actor:</b>	Usuario con discapacidad visual	
<b>Descripción:</b>	Como usuario con discapacidad visual quiero alternar el tema oscuro de la aplicación para visualizar los contenidos de manera cómoda.	
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>
	1	Cuando el usuario seleccione la opción de alternar tema oscuro y el tema actual sea claro.
	2	Cuando el usuario seleccione la opción de alternar tema oscuro y el tema actual sea claro de alto contraste.
	3	Cuando el usuario seleccione la opción de alternar tema oscuro y el tema actual sea oscuro
		<b>Resultado</b>
		El tema de color debe cambiar a oscuro a lo largo de toda la aplicación
		El tema de color debe cambiar a oscuro de alto contraste a lo largo de toda la aplicación
		El tema de color debe cambiar a claro a lo largo de toda la aplicación

4	Cuando el usuario seleccione la opción de alternar tema oscuro y el tema actual sea oscuro de alto contraste	El tema de color debe cambiar a claro de alto contraste a lo largo de toda la aplicación
---	--------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------

**Tabla 32**

*Especificación historia de usuario 12*

<b>Código:</b>	<b>HU-012</b>	
<b>Nombre:</b>	Alternar tema de alto contraste	
<b>Actor:</b>	Usuario con discapacidad visual	
<b>Descripción:</b>	Como usuario con discapacidad visual quiero modificar el tema de alto contraste de la aplicación para visualizar los contenidos de manera cómoda.	
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>
	1	Cuando el usuario seleccione la opción de alternar tema de alto contraste y el tema actual sea claro.
	2	Cuando el usuario seleccione la opción de alternar tema alto contraste y el tema actual sea claro de alto contraste.
	3	Cuando el usuario seleccione la opción de alternar tema alto contraste y el tema actual sea oscuro
	4	Cuando el usuario seleccione la opción de alternar tema alto
		<b>Resultado</b>
		El tema de color debe cambiar a claro de alto contraste a lo largo de toda la aplicación
		El tema de color debe cambiar a claro a lo largo de toda la aplicación
		El tema de color debe cambiar a oscuro de alto contraste a lo largo de toda la aplicación
		El tema de color debe cambiar a oscuro a lo largo de toda la aplicación

contraste y el tema actual sea  
oscuro de alto contraste

**Tabla 33**

*Especificación requerimiento no funcional 3*

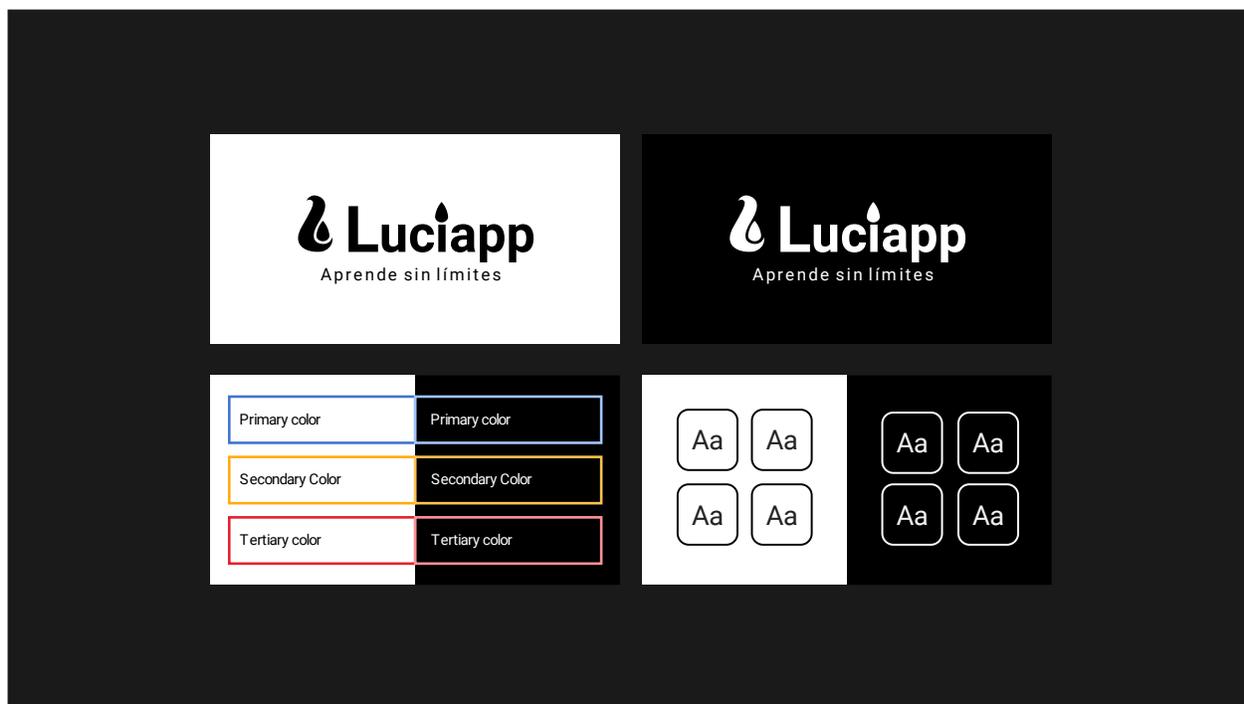
<b>Requerimiento no funcional</b>															
<b>Código:</b>	<b>IRNF-003</b>														
<b>Nombre:</b>	Distinguible														
<b>Normas asociadas</b>	<table border="1"> <thead> <tr> <th><b>Nombre</b></th> <th><b>Descripción</b></th> </tr> </thead> <tbody> <tr> <td>1.4.1. Uso del color (A)</td> <td>El color no es usado como único método para distinguir un elemento visual</td> </tr> <tr> <td>1.4.3. Contraste (AA)</td> <td>El texto (e imágenes de texto) tiene un contraste de al menos 4.5:1 con respecto al color de fondo</td> </tr> <tr> <td>1.4.4. Redimensionar texto (AA)</td> <td>El texto puede ser redimensionado sin tecnologías asistivas hasta en un 200% sin perder contenido ni funcionalidad.</td> </tr> <tr> <td>1.4.6. Contraste (AAA)</td> <td>El texto (e imágenes de texto) tiene un contraste de al menos 7:1 con respecto al color de fondo</td> </tr> <tr> <td>1.4.9. Imágenes de texto (AAA)</td> <td>Sin excepción, las imágenes de texto son meramente decorativas. De otro modo, no deben ser imágenes</td> </tr> <tr> <td>1.4.12. Espaciado de texto (AA)</td> <td>                     El texto debe tener:                     <ul style="list-style-type: none"> <li>• Altura de línea: al menos 1.5 veces el tamaño de fuente</li> <li>• Espaciado de letra: al menos 0.12 veces el tamaño de fuente</li> </ul> </td> </tr> </tbody> </table>	<b>Nombre</b>	<b>Descripción</b>	1.4.1. Uso del color (A)	El color no es usado como único método para distinguir un elemento visual	1.4.3. Contraste (AA)	El texto (e imágenes de texto) tiene un contraste de al menos 4.5:1 con respecto al color de fondo	1.4.4. Redimensionar texto (AA)	El texto puede ser redimensionado sin tecnologías asistivas hasta en un 200% sin perder contenido ni funcionalidad.	1.4.6. Contraste (AAA)	El texto (e imágenes de texto) tiene un contraste de al menos 7:1 con respecto al color de fondo	1.4.9. Imágenes de texto (AAA)	Sin excepción, las imágenes de texto son meramente decorativas. De otro modo, no deben ser imágenes	1.4.12. Espaciado de texto (AA)	El texto debe tener: <ul style="list-style-type: none"> <li>• Altura de línea: al menos 1.5 veces el tamaño de fuente</li> <li>• Espaciado de letra: al menos 0.12 veces el tamaño de fuente</li> </ul>
	<b>Nombre</b>	<b>Descripción</b>													
	1.4.1. Uso del color (A)	El color no es usado como único método para distinguir un elemento visual													
	1.4.3. Contraste (AA)	El texto (e imágenes de texto) tiene un contraste de al menos 4.5:1 con respecto al color de fondo													
	1.4.4. Redimensionar texto (AA)	El texto puede ser redimensionado sin tecnologías asistivas hasta en un 200% sin perder contenido ni funcionalidad.													
	1.4.6. Contraste (AAA)	El texto (e imágenes de texto) tiene un contraste de al menos 7:1 con respecto al color de fondo													
	1.4.9. Imágenes de texto (AAA)	Sin excepción, las imágenes de texto son meramente decorativas. De otro modo, no deben ser imágenes													
1.4.12. Espaciado de texto (AA)	El texto debe tener: <ul style="list-style-type: none"> <li>• Altura de línea: al menos 1.5 veces el tamaño de fuente</li> <li>• Espaciado de letra: al menos 0.12 veces el tamaño de fuente</li> </ul>														

- Espaciado de palabras: al menos 0.16 veces el tamaño de fuente.

**2.2.3.2.2. Etapa de diseño.** Teniendo en cuenta que esta iteración incluye funcionalidades relacionadas al cambio de tema de color, se procedió a desarrollar una nueva paleta de colores para los temas de alto contraste. Para garantizar la accesibilidad, se evaluaron los colores mediante la herramienta Adobe Color Wheel, asegurándose de cumplir con los estándares de nivel AAA de la WCAG. Este proceso permitió establecer una combinación de colores que optimiza la legibilidad y la experiencia de usuario para aquellos con necesidades de accesibilidad visual. Así, se cumple con las normas 1.4.6 y 1.4.3. que se relacionan al contraste AAA y AA, respectivamente.

## Figura 28

*Selección de colores y diseño de logotipo en alto contraste*

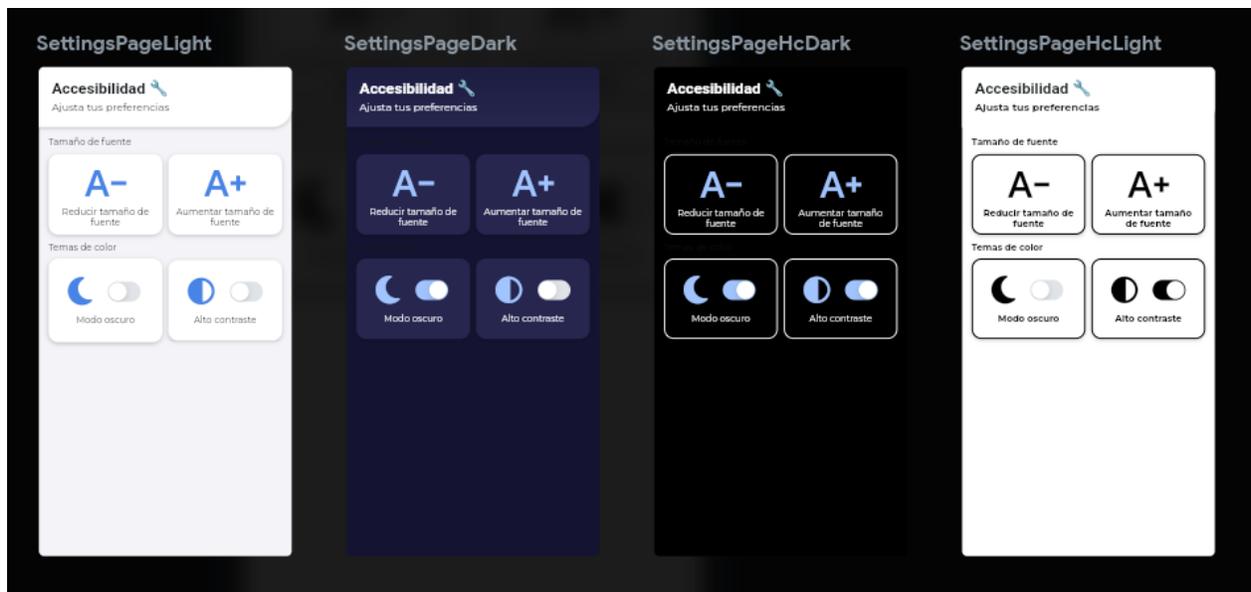


De esta manera, la aplicación cuenta con colores para cuatro temas de color: claro, oscuro, claro de alto contraste y oscuro de alto contraste. Los dos primeros temas de color cumplen con un nivel de accesibilidad AA, mientras que los dos últimos cumplen con el nivel AAA, asegurando así que la aplicación sea utilizable por una amplia gama de usuarios, incluyendo aquellos con necesidades específicas de accesibilidad visual. De esta manera, se brinda la posibilidad de modificar la apariencia de la aplicación de acuerdo con las preferencias y necesidades individuales de cada usuario.

Con los colores seleccionados, se prosiguió con la elaboración de las interfaces de usuario para las funcionalidades contempladas en la segunda iteración. Estas funcionalidades fueron integradas en una única página, la cual fue diseñada teniendo en cuenta los cuatro temas de color previamente definidos: claro, oscuro, claro de alto contraste y oscuro de alto contraste. También, se siguió la norma 1.4.1, que dice que el color no debe ser la única característica que diferencie un componente. Teniendo en cuenta lo anterior, se incluyen varios factores, como la forma, el tamaño, la posición y el contraste.

## Figura 29

*Diseño de interfaces de usuario para configuraciones de accesibilidad*

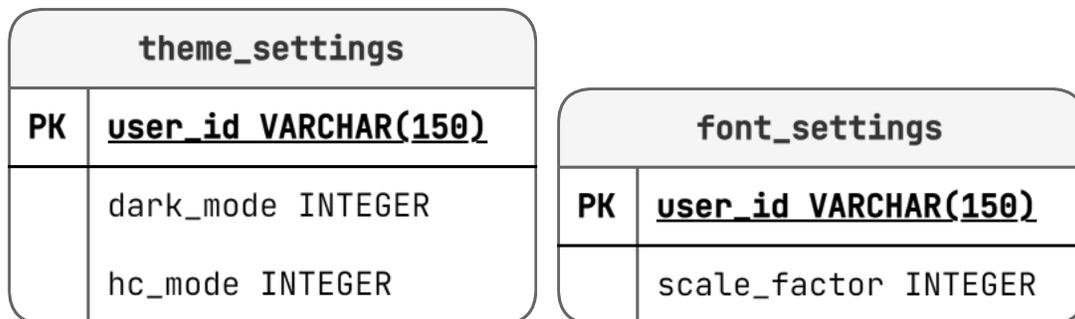


Para esta iteración, se optó por utilizar SQLite como base de datos para almacenar las configuraciones de tema y fuente. SQLite es una base de datos local que almacena los datos directamente en el dispositivo del usuario. Dado que las configuraciones de tema y fuente son específicas de cada usuario y dispositivo, resulta más eficiente realizar consultas a una base de datos local en lugar de una base de datos remota.

Una ventaja significativa de SQLite para esta funcionalidad es su rapidez para acceder a los datos almacenados en el dispositivo del usuario. No es necesario realizar consultas a través de la red ni iterar sobre múltiples usuarios o dispositivos para recuperar las configuraciones individuales. Esto mejora la velocidad de carga y rendimiento de la aplicación, proporcionando una experiencia más fluida para el usuario final. En ese contexto, se diseñaron las siguientes tablas para almacenar las configuraciones de tema y fuente:

**Figura 30**

*Diseño de base de datos para funcionalidades de tema y fuente*

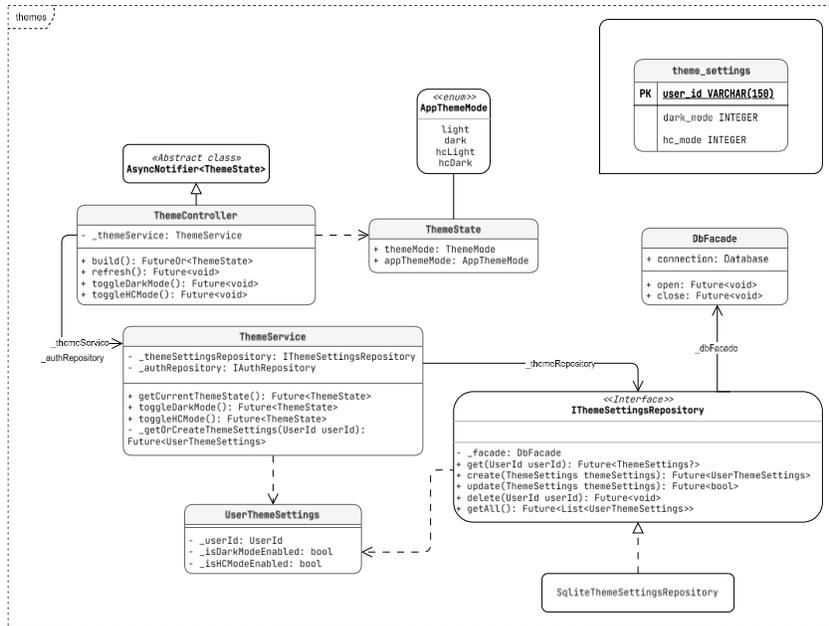


Del anterior diagrama es importante destacar que, la llave primaria de ambas tablas corresponde al id del usuario ya existente dentro de la base de datos de Firebase. Por este motivo no se crea una tabla local de usuarios, ni se referencia a la misma dentro de las tablas de configuraciones. En su lugar, se utiliza este campo como llave primaria para que la búsqueda sea aún más rápida.

En cuanto a los diagramas UML, se separó la lógica en dos funcionalidades: temas de color y tamaño de fuente. Por lo tanto, se realizó un diagrama para cada funcionalidad.

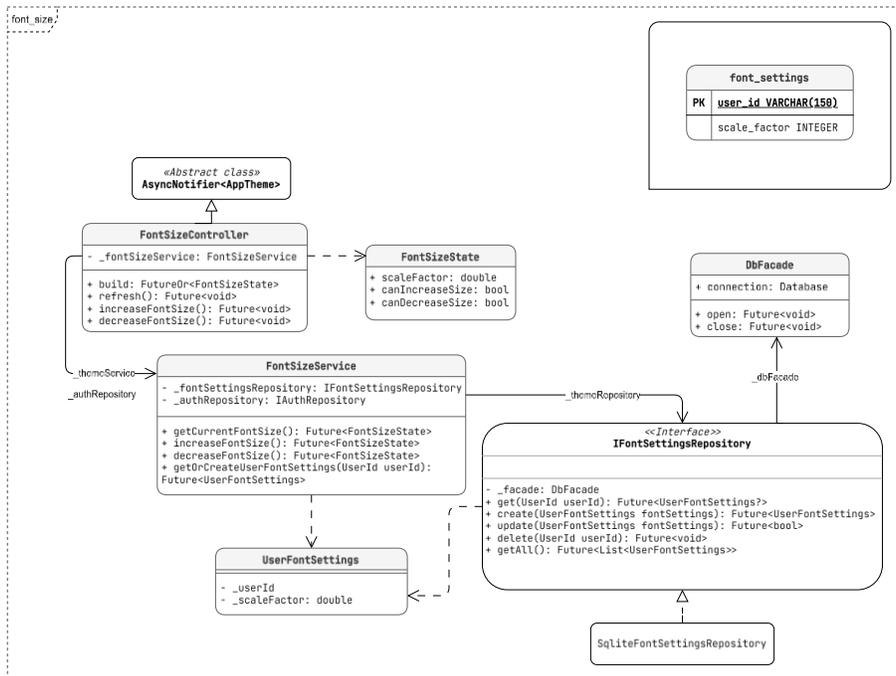
**Figura 31**

Diagrama UML para funcionalidad de temas



**Figura 32**

Diagrama UML para funcionalidad de tamaño de fuente



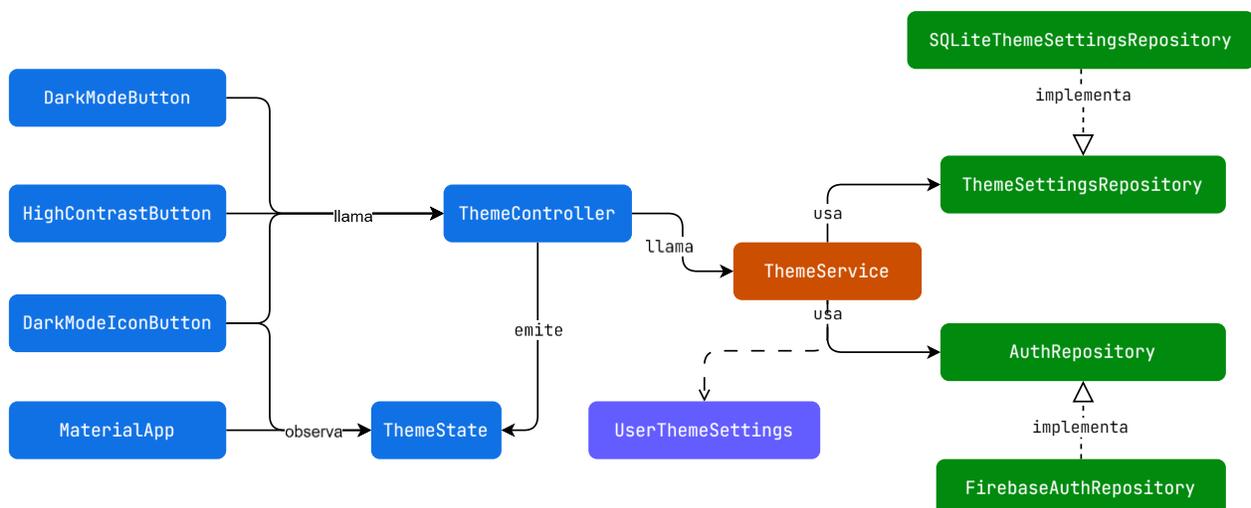
Como se observa, ambos diagramas tienen funcionalidades bastante similares teniendo en cuenta que ambas funcionalidades cambian la apariencia de la aplicación entera, pues el widget MaterialApp reacciona a los cambios del estado de ambas funcionalidades. Al igual que la funcionalidad de autenticación, se utilizan interfaces para desacoplar los repositorios del resto de la aplicación. Además, se utiliza el patrón de diseño “fachada”, que provee una interfaz sencilla para interactuar con clases o servicios más complejos. En este caso, la fachada DbFacade expone métodos para abrir y cerrar una conexión hacia la base de datos SQLite, además de un getter para acceder a la instancia de la base de datos.

Así se concluye con la etapa de diseño de la segunda iteración del proceso de desarrollo, en la que se obtiene una nueva paleta de colores para temas de alto contraste, el diseño de la base de datos local para almacenar configuraciones del usuario, y el diseño del diagrama UML para las funcionalidades de temas de color y tamaño de fuente.

**2.2.3.2.3. Etapa de codificación.** Se trabajó en las ramas “feature/themes” y “feature/font\_size”; cada una en su respectivo directorio dentro del proyecto. La comunicación entre componentes de la funcionalidad de cambio de temas se ilustra en el siguiente diagrama:

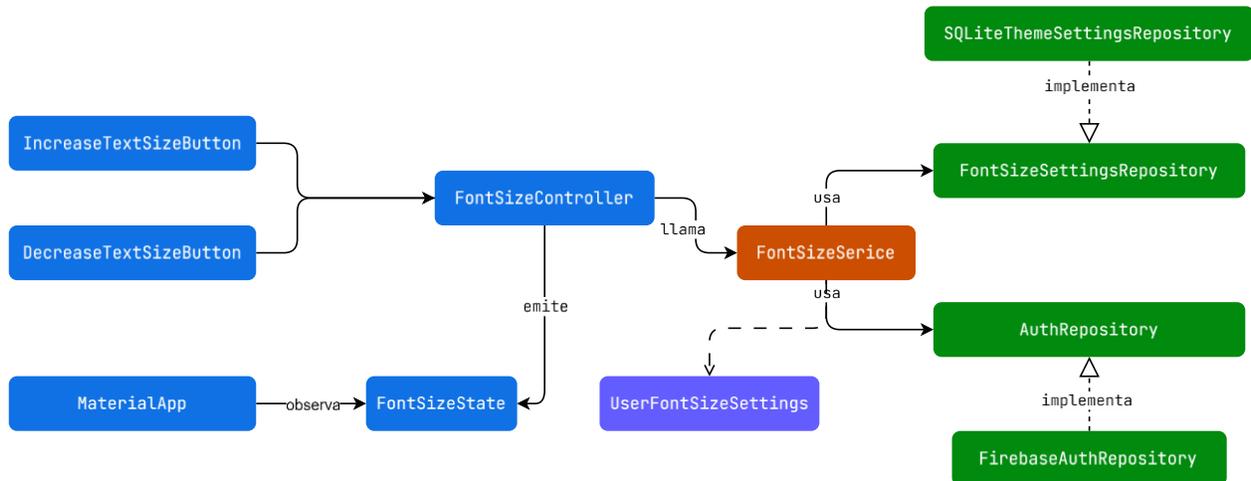
**Figura 33**

*Comunicación de componentes - Temas*



**Figura 34**

*Comunicación de componentes – Tamaño de fuente*



Al igual que la funcionalidad de autenticación, el estado es observado constantemente por MaterialApp, es decir que la aplicación entera reacciona a los cambios que ocurren en el estado. En el caso de la funcionalidad de temas, el estado contiene el tipo de tema seleccionado. Para definir estos temas se hizo uso de la clase AppTheme, incluida en Flutter, que permite asignar colores diferentes a cada tipo de Widget. De esta manera se realizaron cuatro instancias de esta clase, una por cada tema de color a incluir en la aplicación.

Por otro lado, el estado de la funcionalidad de tamaño de fuente contiene tres atributos. El primero corresponde al factor de escala del texto, un valor de tipo flotante que permite modificar el tamaño de la fuente multiplicando su tamaño y obteniendo uno nuevo. Cuando se tiene el tamaño de texto por defecto, significa que el factor de escala es 1.0. Así mismo, un factor de escala 2.0 hace que el tamaño de fuente se duplique, y así sucesivamente. Este valor puede ser alterado por el usuario desde las configuraciones de su dispositivo. Sin embargo, en esta funcionalidad se sobrescribe el factor de escala de texto para que la configuración sea aplicada únicamente dentro de la aplicación y no sea alterada por configuraciones propias del dispositivo. Esto también hace posible que en un mismo dispositivo existan distintas configuraciones de tamaño de texto asociadas a distintas cuentas, sin que estas interfieran entre sí.

El segundo y tercer atributo del estado para esta funcionalidad corresponden a valores de límite para el factor de escala de texto, que se usan para evitar que el texto tenga tamaños exorbitantes. De esta forma, el texto podrá ser ampliado hasta el doble y reducido hasta en un 20% de su tamaño original.

**Figura 35**

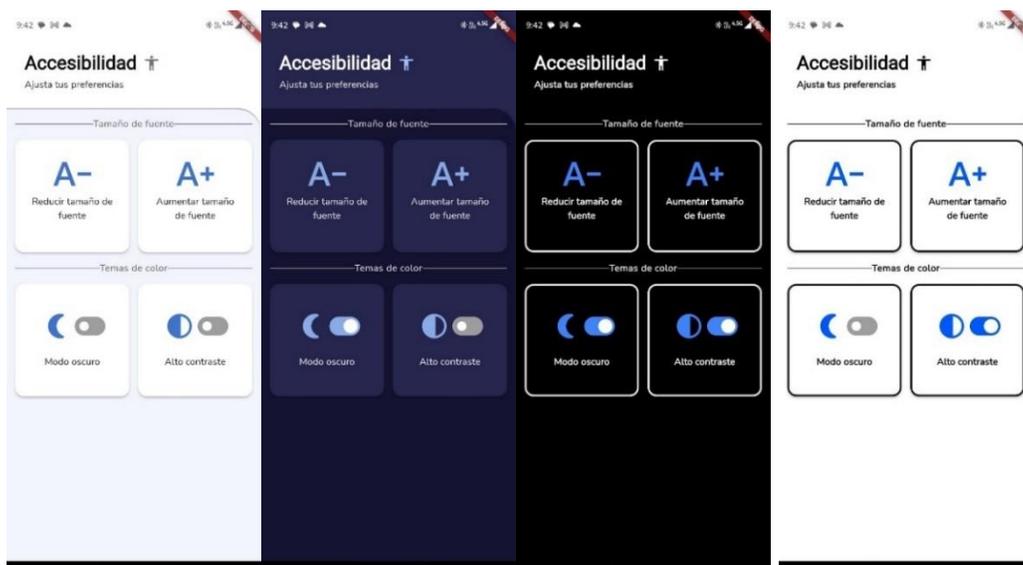
*Límites en factor de escala de texto*

```
FontSizeState.fromUserFontSettings(UserFontSettings fontSettings)
: scaleFactor = fontSettings.scaleFactor,
  canDecreaseSize = fontSettings.scaleFactor > .8,
  canIncreaseSize = fontSettings.scaleFactor < 2;
```

Posteriormente, se terminó de implementar el código para estas dos funcionalidades. Al implementarlas, se cumple con las normas 1.4.4. (Redimensionar texto) y 1.4.3 (Contraste AA), 1.4.6 (Contraste AAA).

**Figura 36**

*Cambio de temas en aplicación*



**Figura 37**

*Cambio de tamaño de fuente en aplicación*



Adicionalmente, se cumple con las normas 1.4.9 (imágenes de texto AAA), pues la única imagen de texto es el logotipo de la aplicación, la cual constituye un elemento netamente decorativo, tal y como lo establece la norma. El resto de texto es completamente legible por lectores de pantalla.

**2.2.3.2.4. Etapa de pruebas.** Al igual que la primera iteración, se realizaron pruebas unitarias y de integración a las clases de tipo controlador y servicio. Se elaboraron los mocks necesarios para simular el comportamiento esperado necesario en cada prueba.

Cabe mencionar que se separaron las pruebas por funcionalidades, de la misma manera que se hizo con los diagramas UML y de componentes. En ese contexto, para la funcionalidad de cambio de temas, se realizaron las siguientes pruebas unitarias para las clases ThemeController y ThemeService.

**Tabla 34**

*Pruebas unitarias para la clase ThemeController*

<b>Id</b>	<b>Nombre</b>	<b>Clase / Método</b>	<b>HU</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-032	Alternar modo oscuro	ThemeController / toggleDarkMode	HU-011	-	ThemeState con modo oscuro alternado
CP-033	Alternar modo alto contraste	ThemeController / toggleHCMMode	HU-012	-	ThemeState con modo de alto contraste alternado
CP-034	Obtener tema de color actual	ThemeController / refresh	HU-011, HU-012	-	ThemeState por defecto

**Tabla 35**

*Pruebas unitarias para la clase ThemeService*

<b>Id</b>	<b>Nombre</b>	<b>Clase / Método</b>	<b>HU</b>	<b>Entra da</b>	<b>Resultado esperado</b>
CP-035	Obtener configuraciones de tema existentes en base de datos	ThemeService / getOrCreateUser ThemeSettings	HU-011, HU-012	UserId	Instancia de UserTheme Settings (existente)
CP-036	Crear nuevas configuraciones de temas en base de datos	ThemeService / getOrCreateUser ThemeSettings	HU-011, HU-012	UserId	Instancia de UserTheme Settings (nuevo)

CP-037	Obtener tema de color actual sin autenticación	ThemeService / getCurrentThemeState	HU-011, HU-012	-	ThemeState.light()
CP-038	Obtener tema de color actual con autenticación	ThemeService / getCurrentThemeState	HU-011, HU-012	-	ThemeState con configuraciones del usuario autenticado
CP-039	Alternar modo oscuro	ThemeService / toggleDarkMode	HU-011	-	ThemeState con modo oscuro alternado
CP-040	Alternar modo alto contraste	ThemeService / toggleHCMMode	HU-012	-	ThemeState con modo de alto contraste alternado

Las anteriores pruebas unitarias buscan que, al ocurrir un cambio de tema de color, se aplique el tema de color correcto a lo largo de la aplicación. Como se manejan dos configuraciones (tema oscuro y de alto contraste), puede haber cuatro combinaciones posibles que derivan en cuatro temas de color distintos. Además, se busca probar la persistencia de estas configuraciones, es decir que al cerrar la aplicación y volverla a abrir, permanezca el último tema de color seleccionado.

Por otro lado, se manejan los escenarios relacionados con la autenticación. Por ejemplo, se define un tema por defecto para usuarios no autenticados (tema claro), y se mantiene este tema hasta que el usuario cree una cuenta. Cuando esto sucede, se registra la configuración por defecto en la base de datos para ese usuario en específico, los casos de prueba CP-035 y CP-036 manejan esta situación, garantizando que un usuario siempre tenga una configuración asociada.

**Figura 38**

*Resultados de pruebas unitarias para la clase ThemeController*

```

Carlos@ZENBOOK MSYS /c/Projects/Luciapp (feature/themes)
$ flutter test test/themes/unit_testing/theme_controller_test.dart -r expanded
00:00 +0: loading C:/Projects/Luciapp/test/themes/unit_testing/theme_controller_test.dart
00:00 +0: Prueba Unitaria [CP-032] Alternar modo oscuro
00:00 +1: Prueba Unitaria [CP-033] Alternar modo alto contraste
00:00 +2: Prueba Unitaria [CP-034] Obtener tema de color actual
00:00 +3: All tests passed!
    
```

**Figura 39**

*Resultados de pruebas unitarias para la clase ThemeService*

```

Carlos@ZENBOOK MSYS /c/Projects/Luciapp (feature/themes)
$ flutter test test/themes/unit_testing/theme_service_test.dart -r expanded
00:00 +0: loading C:/Projects/Luciapp/test/themes/unit_testing/theme_service_test.dart
00:00 +0: Prueba Unitaria [CP-035] Obtener configuraciones de tema existentes en base de datos
00:00 +1: Prueba Unitaria [CP-036] Crear nuevas configuraciones de temas en base de datos
00:00 +2: Prueba Unitaria [CP-037] Obtener tema de color actual sin autenticación
00:00 +3: Prueba Unitaria [CP-038] Obtener tema de color actual con autenticación
00:00 +4: Prueba Unitaria [CP-039] Alternar modo oscuro
00:00 +5: Prueba Unitaria [CP-040] Alternar modo alto contraste
00:00 +6: All tests passed!
    
```

Para las pruebas de integración, se simuló un inicio de sesión exitoso en cada prueba, seguido de una serie de gestos del usuario para llegar a la sección de accesibilidad. Una vez dentro de esta sección, se probó cada cambio de tema en la aplicación con distintas configuraciones previamente establecidas, como se muestra en la siguiente tabla.

**Tabla 36**

*Pruebas de integración para funcionalidad de cambio de tema*

Id	Nombre	CID	Descripción escenario	Entrada	Resultado esperado
----	--------	-----	-----------------------	---------	--------------------

CP-041	Alternar modo oscuro con modo de alto contraste activado	HU11-C2, HU11-C4	El usuario tiene el modo de alto contraste activado y presiona el botón de modo oscuro	Usuario presiona el botón de modo oscuro	Alternar entre AppThemeMode.hcLight y AppThemeMode.hcDark
CP-042	Alternar modo oscuro con modo de alto contraste desactivado	HU11-C1, HU11-C3	El usuario tiene el modo de alto contraste desactivado y presiona el botón de modo oscuro	Usuario presiona el botón de modo oscuro	Alternar entre AppThemeMode.light y AppThemeMode.dark
CP-043	Alternar modo de alto contraste con modo oscuro desactivado	HU12-C1, HU12-C2	El usuario tiene el modo oscuro desactivado y presiona el botón de modo de alto contraste	Usuario presiona el botón de modo de alto contraste	Alternar entre AppThemeMode.hcLight y AppThemeMode.light
CP-044	Alternar modo de alto contraste con modo oscuro activado	HU12-C3, HU12-C4	El usuario tiene el modo oscuro activado y presiona el botón de modo de alto contraste	Usuario presiona el botón de modo de alto contraste	Alternar entre AppThemeMode.hcDark y AppThemeMode.dark

Estas pruebas se ejecutaron en el mismo dispositivo móvil utilizado en la anterior iteración, obteniendo los resultados mostrados a continuación.

**Figura 40***Resultados de pruebas de integración para funcionalidad de cambio de tema*

```

Carlos@ZENBOOK MSYS /c/Projects/Luciapp (feature/themes)
$ flutter test integration_test/themes.dart -r expanded -d ZT322K96PJ
00:00 +0: Loading C:/Projects/Luciapp/integration_test/themes.dart
Running Gradle task 'assembleDebug'...                               21.9s
✓ Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app-debug.apk...         7.3s
00:00 +0: (setUpAll)
00:00 +0: Prueba de Integración [CP-041] Alternar modo oscuro con modo de alto contraste activado
00:03 +1: Prueba de Integración [CP-042] Alternar modo oscuro con modo de alto contraste desactivado
00:05 +2: Prueba de Integración [CP-043] Alternar modo de alto contraste con modo oscuro desactivado
00:06 +3: Prueba de Integración [CP-044] Alternar modo de alto contraste con modo oscuro activado
00:07 +4: (tearDownAll)
00:08 +4: All tests passed!

```

En cuanto a la funcionalidad de cambio de tamaño de fuente, se realizaron pruebas unitarias para las clases `FontSizeController` y `FontSizeService`.

**Tabla 37***Pruebas unitarias para la clase `FontSizeController`*

<b>Id</b>	<b>Nombre</b>	<b>Clase / Método</b>	<b>HU</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-045	El tamaño de fuente aumenta y no alcanza el límite	<code>FontSizeController / increaseFontSize</code>	HU-010	-	Estado de tamaño de fuente con factor de escala aumentado
CP-046	El tamaño de fuente aumenta y alcanza el límite	<code>FontSizeController / increaseFontSize</code>	HU-010	-	Estado de tamaño de fuente con factor de escala aumentado y

					canIncreaseSize falso
CP-047	El tamaño de fuente no aumenta su tamaño	FontSizeController / increaseFontSize	HU-010	-	El estado de tamaño de fuente permanece sin cambios y canIncreaseSize falso
CP-048	El tamaño de fuente disminuye y no alcanza el límite	FontSizeController / decreaseFontSize	HU-010	-	Estado de tamaño de fuente con factor de escala disminuido
CP-049	El tamaño de fuente disminuye y alcanza el límite	FontSizeController / decreaseFontSize	HU-010	-	Estado de tamaño de fuente con factor de escala disminuido y canDecreaseSize falso
CP-050	El tamaño de fuente no disminuye su tamaño	FontSizeController / decreaseFontSize	HU-010	-	El estado de tamaño de fuente permanece sin cambios y canDecreaseSize falso
CP-051	Obtener estado de tamaño de	FontSizeController / refresh	HU-010	-	Estado de tamaño de fuente después

fuentes actuales	de la operación de actualización
---------------------	-------------------------------------

**Tabla 38**

*Pruebas unitarias para la clase FontSizeService*

<b>Id</b>	<b>Nombre</b>	<b>Clase / Método</b>	<b>HU</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-052	Obtener estado actual del tamaño de fuente para no autenticados	FontSizeService / getCurrentFontSizeState	HU-010	-	Estado de tamaño de fuente inicial (FontSizeState.initial())
CP-053	Obtener estado actual del tamaño de fuente para autenticados	FontSizeService / getCurrentFontSizeState	HU-010	-	Estado de tamaño de fuente desde configuraciones del usuario autenticado
CP-054	Obtener tamaño de fuente aumentado cuando no se ha alcanzado el límite	FontSizeService / increaseFontSize	HU-010	-	Estado de tamaño de fuente con factor de escala aumentado
CP-055	Obtener tamaño de fuente cuando ya se ha alcanzado el límite	FontSizeService / increaseFontSize	HU-010	-	Estado de tamaño de fuente permanece sin cambios

CP-056	Obtener tamaño de fuente reducido cuando no se ha alcanzado el límite	FontSizeService / decreaseFontSiz e	HU-010	-	Estado de tamaño de fuente con factor de escala disminuido
CP-057	Obtener tamaño de fuente cuando ya se ha alcanzado el límite	FontSizeService / decreaseFontSiz e	HU-010	-	Estado de tamaño de fuente permanece sin cambios
CP-058	Obtener configuracion es de fuente de usuario existentes desde la base de datos	FontSizeService / getOrCreateUser FontSettings	HU-010	-	Configuraciones de fuente de usuario existentes
CP-059	Crear nuevas configuracion es de fuente de usuario en la base de datos	FontSizeService / getOrCreateUser FontSettings	HU-010	-	Nuevas configuraciones de fuente de usuario

Luego, se procede a ejecutar las pruebas diseñadas. En las siguientes figuras se puede apreciar los resultados de la ejecución de pruebas unitarias para la clase FontSizeController y FontSizeService.

## Figura 41

### Resultados de pruebas unitarias para la clase *FontSizeController*

```
Carlos@ZENBOOK MINGW64 /c/Projects/luciapp (feature/font_size)
$ flutter test test/font_size/unit_testing/font_size_controller_test.dart -r expanded
00:00 +0: Loading C:/Projects/luciapp/test/font_size/unit_testing/font_size_controller_test.dart
00:00 +0: Prueba Unitaria [CP-045] El tamaño de fuente aumenta y no alcanza el límite
00:00 +1: Prueba Unitaria [CP-046] El tamaño de fuente aumenta y alcanza el límite
00:00 +2: Prueba Unitaria [CP-047] El tamaño de fuente no aumenta su tamaño
00:00 +3: Prueba Unitaria [CP-048] El tamaño de fuente disminuye y no alcanza el límite
00:00 +4: Prueba Unitaria [CP-049] El tamaño de fuente disminuye y alcanza el límite
00:00 +5: Prueba Unitaria [CP-050] El tamaño de fuente no disminuye su tamaño
00:00 +6: Prueba Unitaria [CP-051] Obtener estado de tamaño de fuente actuales
00:00 +7: All tests passed!
```

## Figura 42

### Resultados de pruebas unitarias para la clase *FontSizeService*

```
Carlos@ZENBOOK MINGW64 /c/Projects/luciapp (feature/font_size)
$ flutter test test/font_size/unit_testing/font_size_service_test.dart -r expanded
00:00 +0: Loading C:/Projects/luciapp/test/font_size/unit_testing/font_size_service_test.dart
00:00 +0: Prueba Unitaria [CP-052] Obtener estado actual del tamaño de fuente para no autenticados
00:00 +1: Prueba Unitaria [CP-053] Obtener estado actual del tamaño de fuente para autenticados
00:00 +2: Prueba Unitaria [CP-054] Obtener tamaño de fuente aumentado cuando no se ha alcanzado el límite
00:00 +3: Prueba Unitaria [CP-055] Obtener tamaño de fuente cuando ya se ha alcanzado el límite
00:00 +4: Prueba Unitaria [CP-056] Obtener tamaño de fuente reducido cuando no se ha alcanzado el límite
00:00 +5: Prueba Unitaria [CP-057] Obtener tamaño de fuente cuando ya se ha alcanzado el límite
00:00 +6: Prueba Unitaria [CP-058] Obtener configuraciones de fuente de usuario existentes desde la base de datos
00:00 +7: Prueba Unitaria [CP-059] Crear nuevas configuraciones de fuente de usuario en la base de datos
00:00 +8: All tests passed!
```

Con las anteriores pruebas se verifica el cambio de tamaño de fuente en distintos escenarios, así como el funcionamiento de los límites superior e inferior del tamaño de fuente. En el caso del controlador, se prueba que los valores booleanos que indican si se puede o no modificar el tamaño de fuente, sean correctos en cada escenario posible. Así mismo, se prueba que en caso de haber alcanzado cualquiera de los límites, no se realicen cambios en el tamaño de fuente. Por otro lado, en el caso de la clase servicio, se verifica que se manejen los escenarios donde el usuario tenga o carezca de configuraciones previas, realizando la respectiva creación de configuraciones si se presenta este último caso.

Estos mismos escenarios se manejaron mediante pruebas de integración, realizando la misma simulación utilizada en las pruebas de integración de la Tabla 36.

para el inicio de sesión exitoso.

**Tabla 39**

*Pruebas de integración para funcionalidad de tamaño de fuente*

<b>Id</b>	<b>Nombre</b>	<b>CID</b>	<b>Descripción escenario</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-060	Aumentar tamaño de fuente cuando no se alcanza el límite	HU10-C3	El tamaño de fuente actual es 1.0	Usuario presione el botón de aumentar tamaño de fuente	El tamaño de fuente se incrementa a 1.1 si no se ha alcanzado el límite
CP-061	Aumentar tamaño de fuente cuando se alcanza el límite	HU10-C4	El tamaño de fuente actual es 2.0, ha alcanzado el límite	Usuario presione el botón de aumentar tamaño de fuente	El tamaño de fuente permanece sin cambios si ya se alcanzó el límite
CP-062	Disminuir tamaño de fuente cuando no se alcanza el límite	HU10-C1	El tamaño de fuente actual es 1.0	Usuario presione el botón de reducir tamaño de fuente	El tamaño de fuente se reduce a 0.9 si no se ha alcanzado el límite
CP-063	Disminuir tamaño de	HU10-C2	El tamaño de fuente actual	Usuario presione	El tamaño de fuente permanece sin

fuente	es 0.8, ha	el botón	cambios si ya se
cuando se	alcanzado el	de reducir	alcanzó el límite
alcanza el	límite	tamaño de	
límite		fuente	

Para concluir la etapa de pruebas de esta iteración, se ejecutan las pruebas de integración de la funcionalidad de cambio de tamaño de fuente:

### Figura 43

*Resultados de pruebas de integración - cambio de tamaño de fuente*

```

Carlos@ZENBOOK MINGW64 /c/Projects/luciapp (feature/font_size)
$ flutter test integration_test/font_size.dart -r expanded -d ZT322K96PJ
00:00 +0: loading C:/Projects/luciapp/integration_test/font_size.dart
Running Gradle task 'assembleDebug'...                25.5s
✓ Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app-debug.apk... 7.4s
00:00 +0: (setUpAll)
00:00 +0: Prueba de Integración [CP-060] Aumentar tamaño de fuente cuando no se alcanza el límite
00:03 +1: Prueba de Integración [CP-061] Aumentar tamaño de fuente cuando se alcanza el límite
00:04 +2: Prueba de Integración [CP-062] Disminuir tamaño de fuente cuando no se alcanza el límite
00:05 +3: Prueba de Integración [CP-063] Disminuir tamaño de fuente cuando se alcanza el límite
00:07 +4: (tearDownAll)
00:07 +4: All tests passed!
    
```

**2.2.3.3. Tercera iteración.** En esta iteración se desarrolló las historias de usuario HU-005, relacionada con la funcionalidad de cursos y HU-006 relacionada con los contenidos de curso; y con los requerimientos no funcionales IRNF-001 (Alternativas de texto) e IRNF-002 (Adaptable).

**2.2.3.3.1. Etapa de análisis.** Se especificó la historia de usuario y requerimiento no funcional de la iteración. Las siguientes tablas muestra el resultado de este proceso.

### Tabla 40

*Especificación historia de usuario 5*

---

<b>Historia de Usuario</b>
----------------------------

---

<b>Código:</b>	<b>HU-005</b>			
<b>Nombre:</b>	Consultar cursos			
<b>Actor:</b>	Usuario con discapacidad visual			
<b>Descripción:</b>	Como usuario con discapacidad visual quiero consultar todos los cursos disponibles, para poder cursar los que sean de mi interés.			
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>		<b>Resultado</b>
	1	Cuando existan cursos publicados		Se deben mostrar todos los cursos que han sido publicados por el usuario administrador
	2	Cuando no existan cursos publicados		Se debe mostrar un aviso que indique que no hay cursos disponibles
	3	Cuando no haya una conexión activa de internet		Se debe mostrar un aviso que indique que es necesario estar conectado a internet para acceder a los cursos.

**Tabla 41**

*Especificación historia de usuario 6*

<b>Historia de Usuario</b>				
<b>Código:</b>	<b>HU-006</b>			
<b>Nombre:</b>	Consultar contenidos de curso			
<b>Actor:</b>	Usuario con discapacidad visual			
<b>Descripción:</b>	Como usuario con discapacidad visual quiero consultar todos contenidos de un curso para poder cursar los que sean de mi interés.			
	<b>CID</b>	<b>Condición</b>		<b>Resultado</b>

<b>Criterios de Aceptación</b>	1	Cuando existan contenidos en el curso	Se deben mostrar todos los contenidos del curso en una lista
	2	Cuando no existan contenidos	Se debe mostrar una lista vacía

Por otro lado, para los requerimientos no funcionales se utilizó el siguiente formato:

**Tabla 42**

*Especificación requerimiento no funcional 1*

<b>Requerimiento no funcional</b>		
<b>Código:</b>	<b>IRNF-001</b>	
<b>Nombre:</b>	Alternativas de texto	
	<b>Nombre</b>	<b>Descripción</b>
<b>Normas asociadas</b>	1.1.1 Contenido no textual	Todo contenido no textual presentado al usuario debe tener una alternativa de texto que sirva para el propósito equivalente

**Tabla 43**

*Especificación requerimiento no funcional 2*

<b>Requerimiento no funcional</b>		
<b>Código:</b>	<b>IRNF-002</b>	
<b>Nombre:</b>	Adaptable	
	<b>Nombre</b>	<b>Descripción</b>
<b>Normas asociadas</b>	1.3.2. Secuencia significativa	Cuando la secuencia en que un contenido es presentado puede cambiar su significado, se

---

	debe determinar la secuencia de lectura de manera programática.
1.3.3. Características sensoriales	Las instrucciones para operar el contenido no dependen únicamente en características sensoriales como forma, color, tamaño, orientación o sonido.

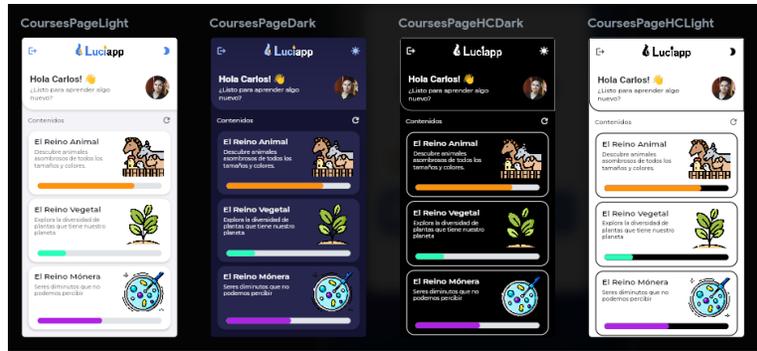
---

**2.2.3.3.2. Etapa de diseño.** Se realizaron las interfaces gráficas con base en los cuatro temas de color de la segunda iteración. Las interfaces corresponden a la página de cursos y la página de contenidos de curso.

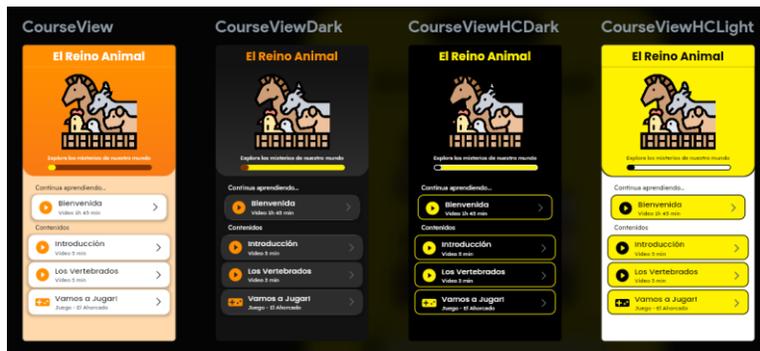
Es importante destacar que, desde la etapa de diseño se comienzan a aplicar las normas de accesibilidad definidas en el requerimiento no funcional IRNF-002. Por ejemplo, la norma 1.3.3, dice que las opciones para operar el contenido no deben depender exclusivamente de una sola característica sensorial del componente (forma, color, tamaño, ubicación, orientación o sonido). En su lugar, debe haber más de un indicador que muestre cómo utilizar el componente. Un ejemplo de la aplicación de esta norma es la forma de identificar un curso en concreto; pues se pueden utilizar diversas características, como su nombre, color, ícono, sonido (en lectores de pantalla). De esta manera, se ofrecen múltiples posibilidades según las preferencias de cada usuario. Al aplicar esta norma, también se cumple con la norma 1.4.1, pues el color ya no es el único medio para identificar el funcionamiento de un componente.

#### **Figura 44**

*Diseño de interfaces pantalla de cursos.*

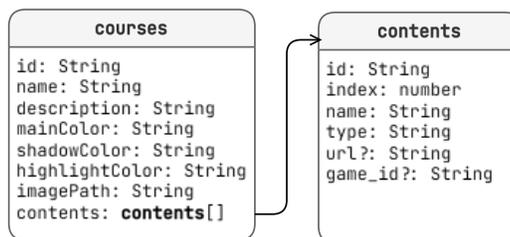


**Figura 45**  
*Diseño de interfaces para pantalla de curso*



Para las funcionalidades relacionadas a cursos, se decidió utilizar Firebase Firestore para el almacenamiento de datos, al igual que sucedió en la funcionalidad de autenticación. Se toma esta decisión debido a que los cursos deben ser accesibles para todos los usuarios en general, por lo que es necesario que la base de datos sea remota. El esquema utilizado es el siguiente:

**Figura 46**  
*Colección de cursos y contenidos de curso*



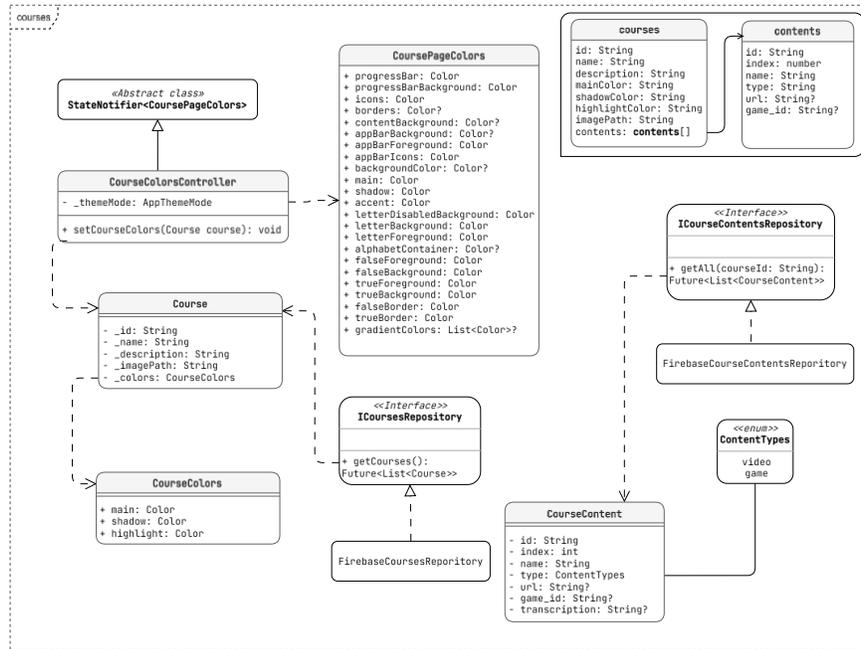
Como se muestra en el esquema, la colección de cursos tiene una subcolección que almacena la información de los contenidos del curso. En este contexto, para consultar todos los contenidos de un curso no es necesario realizar ningún tipo de filtro, pues basta con acceder a la colección entera. En contraste, si se utilizara un esquema relacional, para acceder a los contenidos de un curso habría que establecer una relación de uno a muchos y realizar una consulta con al menos una cláusula “where”, lo cual hace que la consulta sea más lenta conforme aumenta la cantidad de cursos y contenidos. Otro punto positivo de utilizar colecciones es que, al consultar un curso en concreto, la base de datos no retorna sus subcolecciones a menos que se esto sea especificado. Por lo tanto, aunque una colección tenga una o varias subcolecciones, esto no afecta el rendimiento al momento de realizar una consulta a dicha colección.

El diagrama UML para la funcionalidad de cursos presenta algunas diferencias a los realizados previamente. En primer lugar, esta funcionalidad solo requiere opciones de lectura, lo cual simplifica bastante el proceso. Según Bizzotto, cuando una funcionalidad consista solo en lectura de datos, se puede prescindir de las clases de tipo controlador y/o servicio. En este enfoque, los widgets se comunican directamente con el repositorio.

De este modo, el diagrama UML posee únicamente un controlador encargado de exponer y modificar los colores de la página de curso, los cuales son asignados dependiendo del tema de color presente en la aplicación. Para ello se utiliza una clase que tiene múltiples colores como atributo, similar a la clase `AppTheme` de Flutter.

#### **Figura 47**

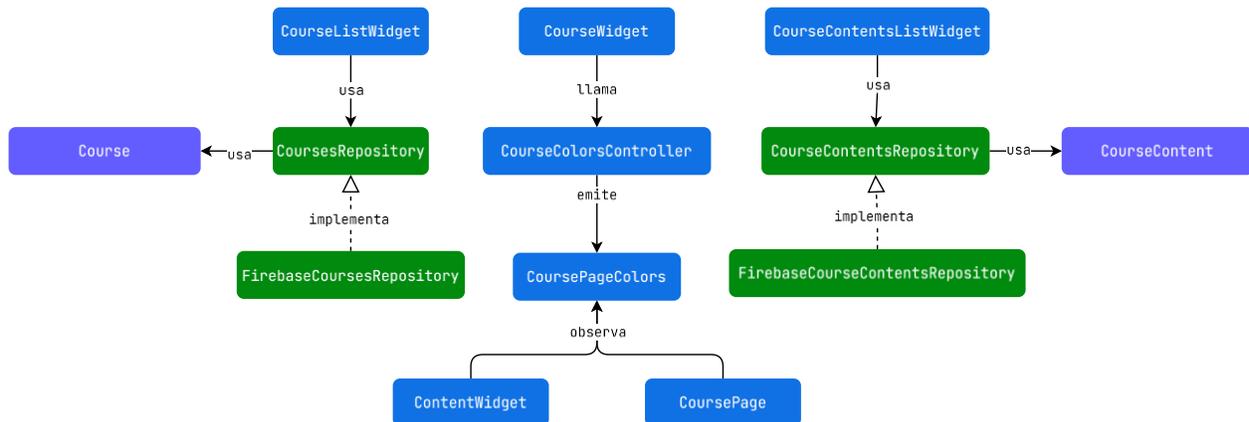
*Diagrama UML para funcionalidad de cursos*



**2.2.3.3.3. Etapa de codificación.** Siguiendo los diagramas e interfaces elaborados en la etapa de diseño, se procedió a implementar la funcionalidad de cursos. Como esta funcionalidad requiere de consultas a la base de datos para obtener la información a mostrar en la aplicación, se comenzó codificando las interfaces con datos estáticos de prueba, para posteriormente realizar la conexión a la base de datos por medio de repositorios, al igual que en funcionalidades anteriores. Se inició también con la implementación de la norma 1.3.2 (Secuencia significativa) del requerimiento IRNF-002. Para ello, se definió de manera programática la jerarquía de componentes en la aplicación, haciendo uso del widget Semantics de Flutter, que ofrece múltiples herramientas para interactuar con lectores de pantalla. En este contexto, se definieron los widgets de encabezado y de texto. De esta manera, se garantiza que la navegación sea en un solo sentido.

Por otro lado, como se muestra en la **Figura 47**, en esta funcionalidad no se implementaron clases de servicio; en su lugar, los widgets que muestran los cursos y los contenidos de curso usarán directamente su respectivo repositorio. La comunicación de los componentes se da de la siguiente manera.

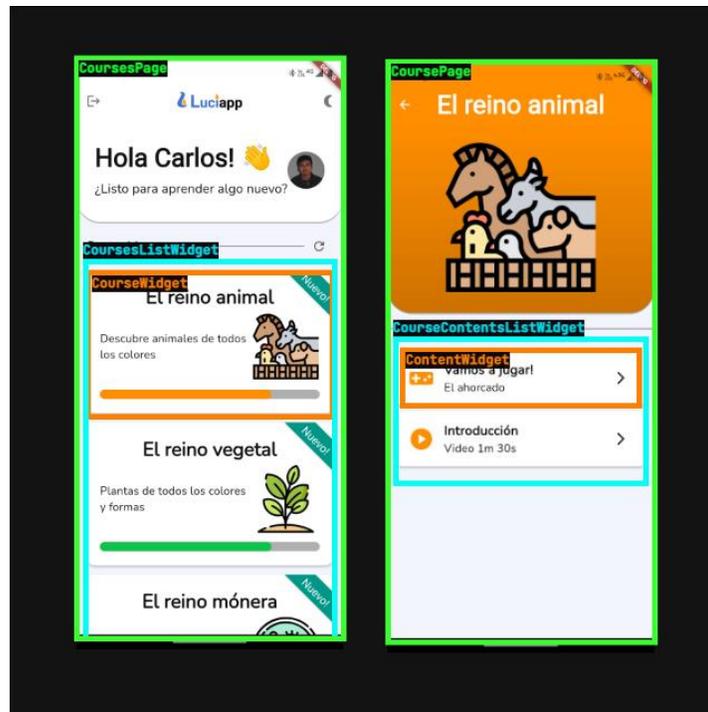
**Figura 48**  
*Comunicación de componentes – cursos y contenidos.*



El flujo comienza al momento de renderizar la página CoursesPage. Esta página contiene el widget CourseListWidget, que usa el repositorio de cursos para obtener los cursos de la base de datos. Cada curso es mostrado mediante el widget CourseWidget, que contiene la información de cada curso, incluyendo sus colores. En el momento que el usuario presiona este widget, se llama al controlador, quien modifica y emite la clase CoursePageColors con los colores del curso en cuestión. Luego, se renderiza la página CoursesPage, que observa los cambios en el estado para asignar sus colores. CoursesPage contiene el widget CourseContentsListWidget, que tiene un funcionamiento similar al de CourseListWidget, con la diferencia de que en este caso se consulta a un repositorio distinto para acceder a los contenidos de un curso en concreto. Cada contenido es mostrado con el widget ContentWidget, que a su vez observa al estado para asignar sus colores. La siguiente Figura muestra los componentes visuales mencionados.

### Figura 49

*Selección de componentes en interfaz gráfica de cursos*



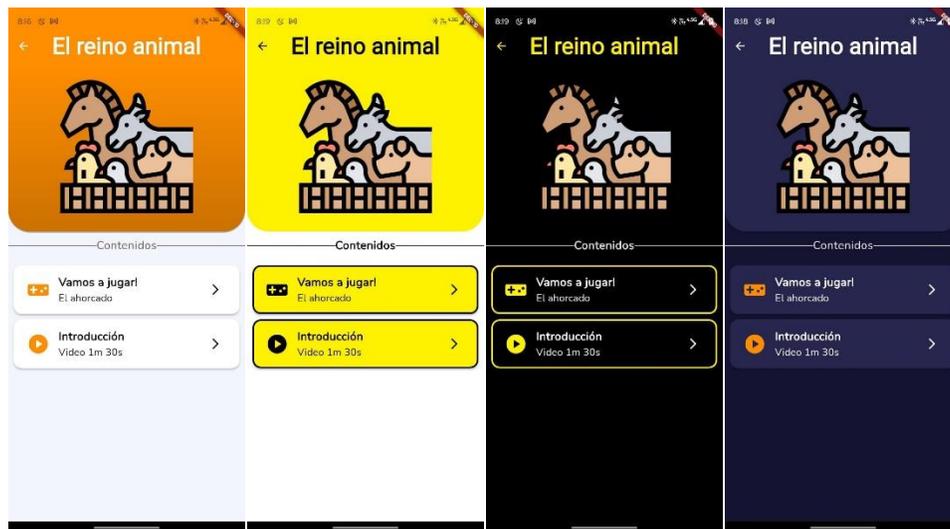
Al finalizar la etapa de codificación se agrega la funcionalidad de acceder a los cursos alojados en la base de datos de Firebase, así como a los contenidos de cada curso. Es importante mencionar que posteriormente se establecerá que un curso solo puede ser publicado si este tiene al menos un contenido, por lo tanto, el criterio 2 de la historia de usuario 6 no será visible en la aplicación final. Por otro lado, cada curso tiene su propia paleta de colores, que se integra en la aplicación para cada tema de color que pueda presentarse.

## Figura 50

*Página de cursos en aplicación*



**Figura 51**  
*Página de curso en aplicación*



Para el requerimiento no funcional IRNF-001, se hizo uso del widget Semantics para establecer la alternativa de texto para contenidos no textuales, es decir, cómo los lectores de pantalla muestran al usuario final el componente que está siendo seleccionado. Este widget permite, entre otras cosas, asignar una etiqueta a cada componente dentro de la aplicación y determinar si éste debe estar o no enfocado por defecto.

Esto resulta especialmente útil cuando el componente no tiene texto visible para el usuario. Por ejemplo, cuando un botón consta únicamente de un ícono, sin un texto que le indique al usuario para que sirve ese botón. Un ejemplo dentro de la aplicación es el botón de activar o desactivar el modo oscuro. Este botón consta únicamente de un ícono de sol o de luna. Si no se asigna una etiqueta elocuente para este tipo de iconos, el lector de pantalla le indicará al usuario que simplemente se trata de un icono, sin especificar su función ni utilidad.

## Figura 52

*Implementación de norma 1.1.1. – contenido no textual*

```
...
class DarkModeIconButton extends ConsumerWidget {
  const DarkModeIconButton({super.key = Keys.darkModeIconButton});

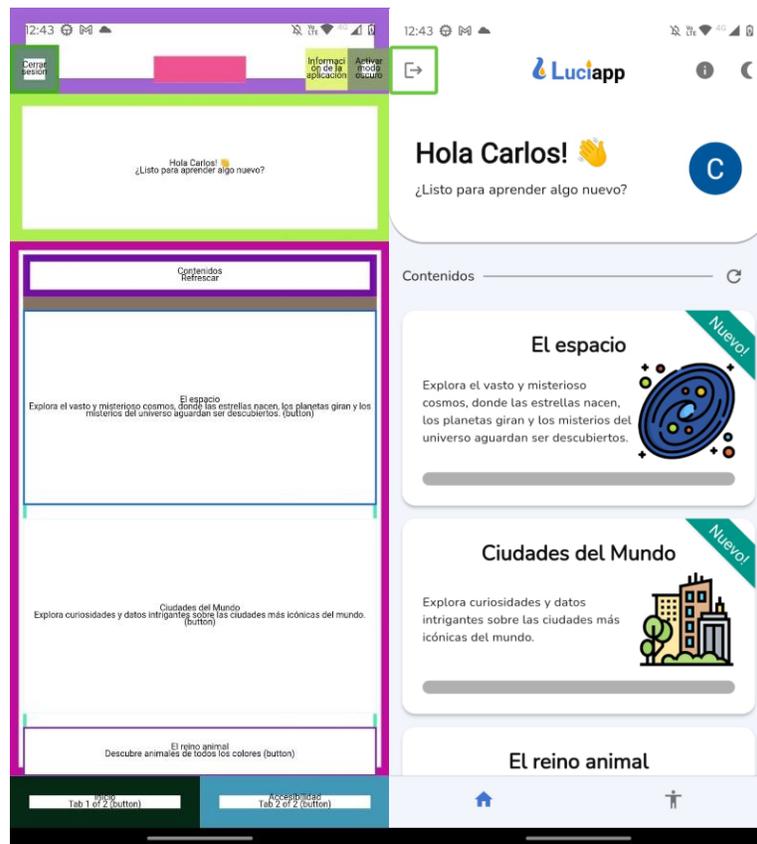
  @override
  Widget build(BuildContext context, WidgetRef ref) {
    final bool isDarkModeEnabled = ref.watch(isDarkModeEnabledProvider);

    return Semantics(
      label: '${isDarkModeEnabled ? 'Desactivar' : 'Activar'} modo oscuro',
      child: ExcludeSemantics(
        child: IconButton(
          icon: Icon(isDarkModeEnabled ? Icons.sunny : Icons.nightlight_round),
          onPressed: ref.read(themeControllerProvider.notifier).toggleDarkMode,
        ), // IconButton
      ), // ExcludeSemantics
    ); // Semantics
  }
}
```

En este caso dependiendo del estado del tema de color activo, el lector de pantalla leerá “activar” o “desactivar” el modo oscuro. De este mismo modo, otros botones similares fueron adaptados para ser compatibles con lectores de pantalla. Para probar el requerimiento no funcional IRNF-001 se activó el modo de depuración semántico de Flutter. Al compilar la aplicación con este modo, se mostrará de manera visual todos los componentes legibles para el lector de pantalla. Si el requerimiento está bien implementado, no debería haber ningún componente sin etiquetar (unlabeled).

## Figura 53

*Aplicación con modo de depuración semántica*



**2.2.3.3.4. Etapa de pruebas.** Teniendo en cuenta que en esta iteración solamente se hizo uso de operaciones de lectura en la base de datos, la única clase a probar será el controlador de los colores de curso. Esa clase se encarga de asignar los colores a la aplicación dependiendo del tema de color activo y de los colores específicos de cada curso punto de esta manera es posible obtener cuatro combinaciones con una misma paleta de colores asignada a cada curso. A continuación, se muestran las pruebas unitarias realizadas en esa clase.

**Tabla 44**

*Pruebas unitarias para la clase `courseColorsController`*

Id	Nombre	Clase / Método	HU	Entrada	Resultado esperado
----	--------	----------------	----	---------	--------------------

CP-064	Establecer colores del curso para tema claro	courseColors Controller/ setCourseColors	HU-005, HU-006	AppTheme Mode.light	Colores del curso en tema claro
CP-065	Establecer colores del curso para tema oscuro	courseColors Controller/ setCourseColors	HU-005, HU-006	AppTheme Mode.dark	Colores del curso en tema oscuro
CP-066	Establecer colores del curso para tema alto contraste claro	courseColors Controller/ setCourseColors	HU-005, HU-006	AppTheme Mode.hcLight	Colores del curso en tema alto contraste claro
CP-067	Establecer colores del curso para tema alto contraste oscuro	courseColors Controller/ setCourseColors	HU-005, HU-006	AppTheme Mode.hcDark	Colores del curso en tema alto contraste oscuro

Por otro lado, las pruebas de integración se realizaron en la aplicación entera para validar cada uno de los criterios de aceptación que se definieron en la especificación de cada historia de usuario.

**Tabla 45**

*Pruebas de integración para funcionalidad de cursos*

<b>Id</b>	<b>Nombre</b>	<b>CID</b>	<b>Descripción escenario</b>	<b>Entrada</b>	<b>Resultado esperado</b>
-----------	---------------	------------	------------------------------	----------------	---------------------------

CP-068	Mostrar lista de cursos cuando existen registros en la base de datos	HU05-C1	La base de datos contiene una lista de cursos	-	Lista de cursos mostrada en pantalla de cursos
CP-069	Obtener mensaje cuando no existen cursos en la base de datos	HU05-C2	La base de datos no contiene ningún curso	-	Se muestra el mensaje "No se han publicado cursos aún"
CP-070	Mostrar aviso cuando no se cuenta con una conexión a internet	HU05-C3	El dispositivo no cuenta con una conexión activa de internet	-	Se muestra el mensaje "Es necesario estar conectado a internet para acceder a los cursos"

De esta manera, se realiza la ejecución de las pruebas de integración relacionadas con la funcionalidad de temas de color en la aplicación. En la Figura 54 se muestra el resultado de la ejecución de dichas pruebas.

### Figura 54

*Resultados de pruebas unitarias para la clase courseColorsController*

```

Carlos@ZENBOOK MSYS /c/Projects/Luciapp (feature/courses)
$ flutter test test/courses/unit_testing/course_colors_controller_test.dart -r expanded
00:00 +0: loading C:/Projects/Luciapp/test/courses/unit_testing/course_colors_controller_test.dart
00:00 +0: Prueba Unitaria [CP-064] Establecer colores del curso para el tema claro
00:00 +1: Prueba Unitaria [CP-065] Establecer colores del curso para el tema oscuro
00:00 +2: Prueba Unitaria [CP-066] Establecer colores del curso para el tema de alto contraste claro
00:00 +3: Prueba Unitaria [CP-067] Establecer colores del curso para el tema de alto contraste oscuro
00:00 +4: All tests passed!
    
```

**Figura 55**

*Resultados de pruebas de integración para funcionalidad de cursos*

```

Carlos@ZENBOOK MSYS /c/Projects/Luciapp (feature/courses)
$ flutter test integration_test/courses.dart -r expanded -d ZT322K96PJ
00:00 +0: loading C:/Projects/Luciapp/integration_test/courses.dart
Running Gradle task 'assembleDebug'...                               32.6s
✓ Built build/app/outputs/flutter-apk/app-debug.apk.
Installing build/app/outputs/flutter-apk/app-debug.apk...         5.2s
00:00 +0: (setUpAll)
00:00 +0: Prueba de Integración [CP-068] Mostrar lista de cursos cuando existen registros en la base de datos
00:04 +1: Prueba de Integración [CP-069] Obtener mensaje cuando no existen cursos en la base de datos
00:06 +2: Prueba de Integración [CP-070] Mostrar aviso cuando no se cuenta con una conexión a internet
00:09 +3: (tearDownAll)
00:09 +3: All tests passed!
    
```

**2.2.3.4. Cuarta iteración.** El objetivo de esta iteración es desarrollar las historias de usuario relacionadas con los contenidos digitales y minijuegos adaptados para personas con discapacidad visual. En cuanto a los requerimientos no funcionales, se trabajó en el IRNF-004 (Navegable), IRNF-005 (Modalidades de entrada), IRNF-006 (Predecible) e IRNF-007 (Compatible). Con esta iteración, se concluye el desarrollo de la estrategia computacional basada en tecnologías móviles propuesta. Las siguientes historias de usuario corresponden a la aplicación web de administrador utilizada para agregar contenidos digitales a la aplicación móvil desarrollada.

**2.2.3.4.1. Etapa de análisis.** Se realizó la especificación de las historias de usuario HU-007, HU-008, IRNF-004, IRNF-005, IRNF-006 y IRNF-007.

**Tabla 46**

*Especificación historia de usuario 7*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-007</b>		
<b>Nombre:</b>	Reproducir videos		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero reproducir videos didácticos para aprender temáticas nuevas.		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el usuario seleccione un video	Se debe reproducir el video completo
	2	Cuando el usuario termine el video	Se debe redirigir a la pantalla de curso

**Tabla 47**

*Especificación historia de usuario 8*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-008</b>		
<b>Nombre:</b>	Jugar al ahorcado		
<b>Actor:</b>	Usuario con discapacidad visual		
<b>Descripción:</b>	Como usuario con discapacidad visual quiero jugar al ahorcado con niveles relacionados a la temática de cada curso para afianzar mis conocimientos		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el usuario inicie el juego	Se debe mostrar los espacios de la palabra del nivel y una pista sobre dicha palabra
	2	Cuando el usuario realice un acierto	Se debe revelar la letra que ha acertado en la palabra del nivel

3	Cuando el usuario realice un fallo	Se debe dibujar una parte del ahorcado
4	Cuando el usuario llegue al límite de fallos permitidos	Se debe completar el dibujo del ahorcado y se debe mostrar un mensaje de juego terminado
5	Cuando el usuario acierte todas las letras de la palabra del nivel	Se debe mostrar un mensaje de éxito

**Tabla 48**

*Especificación historia de usuario 9*

<b>Historia de Usuario</b>		
<b>Código:</b>	<b>HU-009</b>	
<b>Nombre:</b>	Jugar a la trivia	
<b>Actor:</b>	Usuario con discapacidad visual	
<b>Descripción:</b>	Como usuario con discapacidad visual quiero jugar a la trivia con niveles relacionados a la temática de cada curso para afianzar mis conocimientos.	
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>
	1	Cuando el usuario inicie el juego
	2	Cuando el usuario seleccione una respuesta incorrecta
	3	Cuando el usuario seleccione una respuesta correcta
		<b>Resultado</b>
		Se deben mostrar la pregunta y sus opciones de respuesta, junto a una imagen alusiva
		Se debe reproducir un sonido de fallo y mostrar un mensaje
		Se debe reproducir un mensaje de éxito y mostrar un mensaje con la explicación de la respuesta

**Tabla 49**

*Especificación requerimiento no funcional 4*

<b>Requerimiento no funcional</b>		
<b>Código:</b>	<b>IRNF-004</b>	
<b>Nombre:</b>	Navegable	
	<b>Nombre</b>	<b>Descripción</b>
<b>Normas asociadas</b>	2.4.6. Encabezados y etiquetas (AA)	Los encabezados y etiquetas describen temas o propósito
	2.4.10. Encabezados de sección (AAA)	Los encabezados son utilizados para organizar el contenido

**Tabla 50**

*Especificación requerimiento no funcional 5*

<b>Requerimiento no funcional</b>		
<b>Código:</b>	<b>IRNF-005</b>	
<b>Nombre:</b>	Modalidades de entrada	
	<b>Nombre</b>	<b>Descripción</b>
<b>Normas asociadas</b>	2.5.3. Etiqueta en nombre (A)	Para componentes con etiquetas que incluyen texto, o imágenes de texto, el nombre contiene el texto que está siendo presentado.
	2.5.5. Tamaño de objetivo (AAA)	El tamaño del objetivo para las entradas de puntero es de al menos 44 por 44 píxeles.

**Tabla 51**

*Especificación requerimiento no funcional 6*

<b>Requerimiento no funcional</b>		
-----------------------------------	--	--

<b>Código:</b>	<b>IRNF-006</b>	
<b>Nombre:</b>	Predecible	
	<b>Nombre</b>	<b>Descripción</b>
<b>Normas asociadas</b>	3.2.3. Navegación consistente (AA)	Mecanismos de navegación que se repiten en múltiples páginas ocurren en el mismo orden relativo cada vez que se repiten.
	3.2.4. Identificación consistente (AA)	Componentes que tienen la misma funcionalidad en múltiples páginas, son identificados constantemente.

**Tabla 52**

*Especificación requerimiento no funcional 7*

<b>Requerimiento no funcional</b>		
<b>Código:</b>	<b>IRNF-007</b>	
<b>Nombre:</b>	Compatible	
	<b>Nombre</b>	<b>Descripción</b>
<b>Normas asociadas</b>	4.1.2. Nombre, rol, valor (A)	El nombre, rol y valor de todos los componentes de interfaz de usuario deben ser programáticamente determinados.
	4.1.3. Mensajes de estado (AA)	Los cambios en contenido fuera de foco deben ser notificados al usuario sin interrumpir su trabajo.

**2.2.3.4.2. Etapa de diseño.** Se realizó el diseño de las interfaces de usuario para ambos minijuegos, en los cuatro temas de color que maneja la aplicación.

**Figura 56**

*Diseño de interfaces de usuario para juego de trivia*

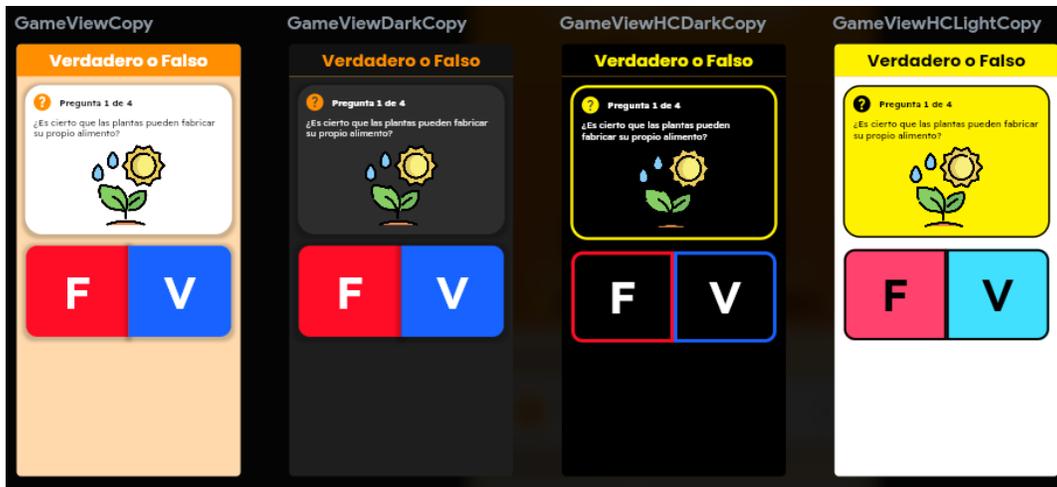
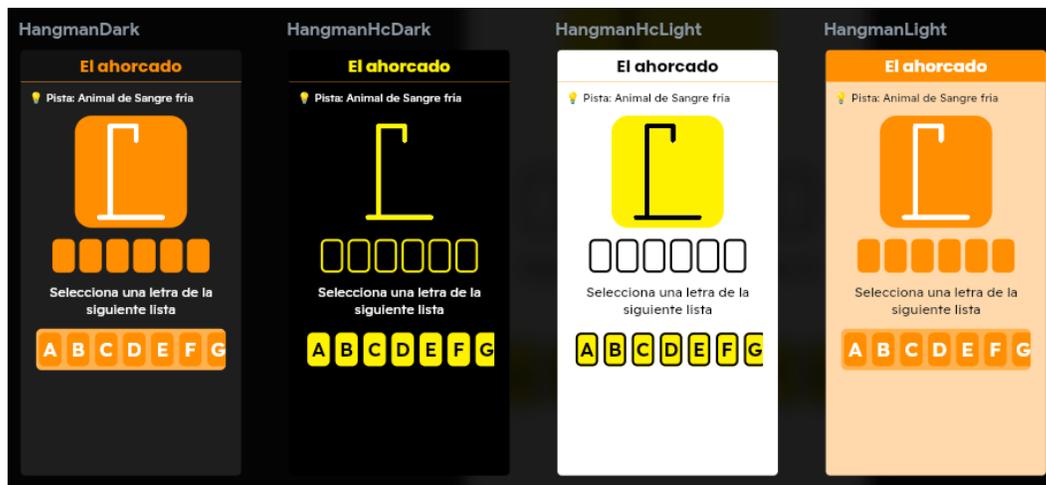


Figura 57

Diseño de interfaces de usuario para juego de ahorcado



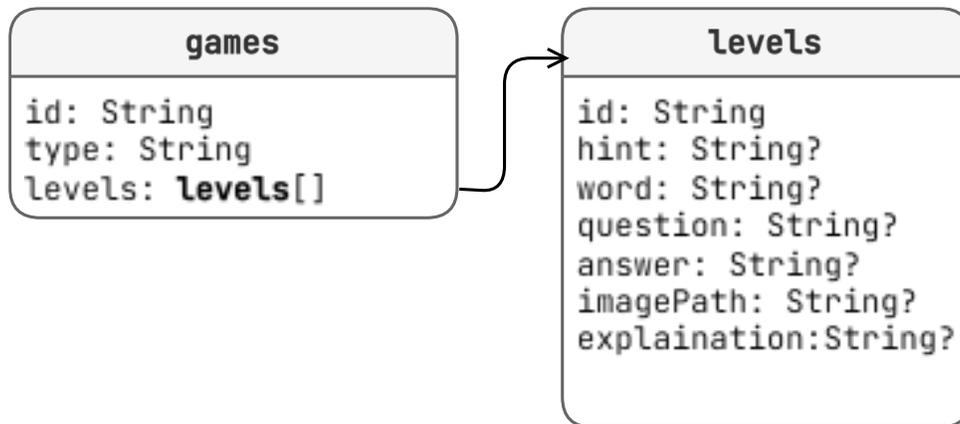
Luego, se continuó con el diseño de la base de datos para manejar la persistencia en la funcionalidad de juegos. Como se mencionó anteriormente, cada curso puede tener varios juegos, que pueden ser de tipo “ahorcado” o “trivia”. La Figura 46.

muestra la estructura de la colección contenidos de curso; nótese que existen dos campos opcionales, que son “url” y “game\_id”.

En esta iteración se utilizará a la propiedad “game\_id” para consultar una colección dedicada a almacenar los juegos y sus niveles correspondientes. Siendo así, las colecciones “games” y “levels” tienen la siguiente estructura:

### Figura 58

*Estructura de colección de juegos*

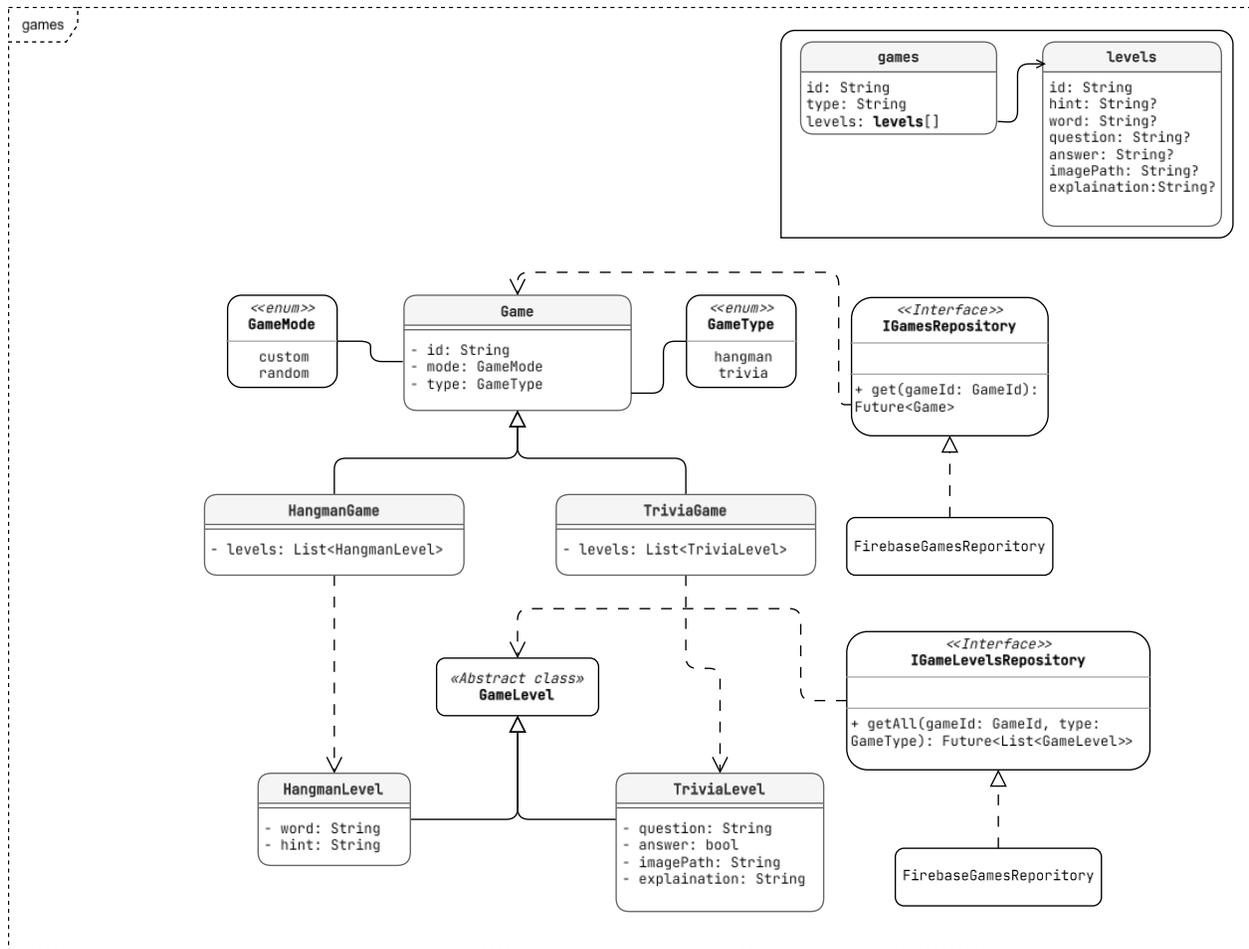


En esta ocasión también se utilizaron subcolecciones para acceder a los niveles de cada juego, los cuales almacenan información como las preguntas, respuestas, pistas y explicaciones, dependiendo del tipo de juego al que pertenezcan. Cabe destacar que, debido a que las funcionalidades de los juegos únicamente requieren operaciones de lectura de datos, se optó por implementar una arquitectura simplificada que prescinde de clases de controlador y servicio, dando lugar a un sistema más ligero y eficiente.

La siguiente figura ilustra la arquitectura simplificada descrita anteriormente. En ella, se observa cómo las subcolecciones de niveles interactúan directamente con la capa de datos.

### Figura 59

*Diagrama UML para funcionalidad de juegos*



Del anterior diagrama se destaca que, se hace uso de herencia para definir las clases HangmanGame y TriviaGame a partir de la clase padre Game. Esto debido a que, a nivel de base de datos todos los juegos van dentro de la colección games, simplificando el proceso de búsqueda a la hora de consultar un juego determinado. Lo mismo sucede con la clase GameLevel, que tiene como clases hijas a HangmanLevel y TriviaLevel, para definir la estructura de los niveles de ahorcado y trivia, respectivamente.

**2.2.3.4.3. Etapa de codificación.** Para la codificación de las historias de usuario, se inició con la implementación de las normas relacionadas al IRNF-004, IRNF-005, e IRNF-006.

Para las normas 3.2.3 y 3.2.4, correspondientes a la navegación e identificación consistente, se realizó una abstracción de las páginas relacionadas a los cursos y contenidos de curso, de manera que puedan ser reutilizadas para distintos cursos o contenidos. De esta manera se garantiza que la

navegación es consistente, pues se utiliza la misma plantilla para mostrar cada curso y cada minijuego, sin importar del curso al que pertenezca. Por ende, también se cumple con la normativa 3.2.4, pues cada elemento se identifica de la misma manera en cada una de las páginas, independientemente del contenido que sea presentado. Lo anterior supone una ventaja para el usuario, pues facilita la navegación y uso de la aplicación en general; y para el desarrollador, pues evita la repetición de código y fomenta la reutilización de este.

Con respecto a la norma 2.4.6 (Encabezados y etiquetas) y 2.4.10 (Organizar contenido), se seleccionaron y etiquetaron los encabezados a lo largo de la aplicación, lo cual cumple la primera norma mencionada. La segunda, hace referencia a casos más específicos. Uno de ellos, descrito en la documentación oficial como ejemplo, se presenta cuando en una página de ajustes existen varias secciones con distintas configuraciones agrupadas. Esto es aplicable en la página de accesibilidad de la aplicación, donde existen dos secciones (ajustes de texto y ajustes de temas de color). Por lo tanto, se aplican los encabezados en dichas secciones.

## Figura 60

*Implementación de norma 2.4.6 y 2.4.10 en sección de accesibilidad*

```
// font size section
Semantics(
  header: true,
  child: const TextDivider(Strings.fontSize),
), // Semantics
const Row(
  children: [
    Expanded(child: DecreaseTextSizeButton(),
      SizedBox(width: 15),
      Expanded(child: IncreaseTextSizeButton(),
    ],
  ), // Row
// text theme section
Semantics(
  header: true,
  child: const TextDivider(Strings.colorThemes),
), // Semantics
const Row(
  children: [
    DarkModeButton(),
    SizedBox(width: 15),
    Expanded(child: HighContrastButton(),
  ],
), // Row
```

Continuando con la norma 2.5.3 (Etiqueta en nombre), Flutter por defecto incluye el nombre del componente o texto del componente en cada etiqueta legible por defecto, incluyendo

componentes como campos de texto o botones. Las únicas excepciones son los íconos que no presentan texto, los cuales fueron abordados en la tercera iteración. Por lo tanto, no fue necesario realizar modificaciones para cumplir con esta normativa.

Para el cumplimiento de la norma 2.5.5 (Tamaño objetivo), Flutter trae por defecto una prueba que verifica que todo componente presionables tenga un área activa de al menos 44x44 píxeles. Para probar esto, se agregaron las siguientes líneas en todas las pruebas de integración realizadas previamente, y se la continuó agregando para pruebas posteriores.

### Figura 61

*Prueba de accesibilidad para norma 2.5.5*

```
// ... Otros tests ...

testWidgets('[CP-XXX] Prueba de integracion', (tester) async {
  final SemanticsHandle handle = tester.ensureSemantics();

  await tester.pumpWidget(const MyApp());

  // ... Contenido del test ...

  // ? Verifica un area activa de 48x48 px
  await expectLater(tester, meetsGuideline(androidTapTargetGuideline));

  // ? Verifica un area activa de 44x44 px
  await expectLater(tester, meetsGuideline(iOSTapTargetGuideline));

  handle.dispose();
});
```

Finalmente, la norma 4.1.2 (Nombre, rol, valor) se cumple por defecto debido al sistema de accesibilidad de Flutter. Esto se puede observar al activar el modo de depuración semántica, por ejemplo, la siguiente figura muestra cómo el botón de inicio de la barra de navegación incluye su nombre (Inicio), rol (botón) y valor (pestaña 1 de 2).

### Figura 62

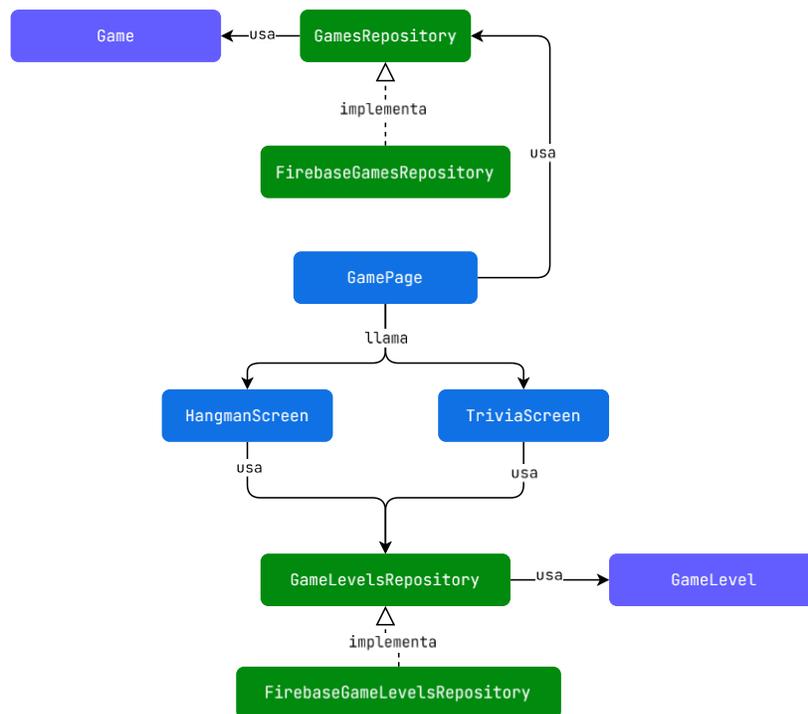
### Implementación de norma 4.1.2. mediante depuración semántica



Por otro lado, una vez implementadas estas normas, se continuó con la codificación de las funcionalidades de la iteración. Así, se aplicó el diagrama UML elaborado en la etapa de diseño, obteniendo la siguiente comunicación de componentes:

**Figura 63**

*Comunicación de componentes - juegos*



Para esta funcionalidad, el flujo comienza cuando se abre la página GamePage a partir del id de un juego (presente en la propiedad game\_id del contenido). En este punto, se hace uso del repositorio de juegos para obtener la información del juego en cuestión. Dentro de esta información

se consulta si se trata de un juego de ahorcado o de trivia, para mostrar la página correspondiente. En cualquiera de los casos, se llama usa el repositorio de niveles para traer la lista de niveles correspondiente al juego, y finalmente se renderiza el juego normalmente.

Debido a que el funcionamiento de ambos juegos es sencillo, no se aplicó la arquitectura para cada juego en particular. En lugar de ello, se desacopla la lógica del juego de manera que la clase HangmanGame o TriviaGame únicamente se encargan de ejecutar el juego dada una lista de niveles, sin importar de donde venga dicha lista. Esto permite que el proceso de testing sea más sencillo, como se verá a continuación.

No obstante, en el caso particular del juego del ahorcado, fue pertinente aplicar la norma 4.1.3 sobre mensajes de estado, pues el juego incluye funciones que no pueden ser leídas por defecto por un lector de pantalla. Por ejemplo, cuando el usuario realiza un fallo, el cambio reflejado en el dibujo del ahorcado es ignorado por el lector de pantalla a menos que se especifique lo contrario, llamando al servicio de accesibilidad del dispositivo e ingresando manualmente la frase que el lector de pantalla debe leer. En ese orden de ideas, se debe utilizar el lector de pantalla para alertar al usuario sobre la cantidad de intentos disponibles, así como de la posición de las letras acertadas en pantalla. De esta manera, se hace posible que personas con discapacidad visual total puedan acceder a este tipo de contenidos.

Finalmente, para la historia de usuario HU-007 (Reproducir videos) se hizo uso de una librería externa que usa a YouTube como reproductor de video dentro de Flutter. De esta manera se hace más eficiente el proceso de carga de video, pues de hacerlo por medio del servicio Firebase Storage, el usuario debería descargar todo el video cada vez que abre el contenido, lo cual resulta costoso en términos de rendimiento. Usando a YouTube, se aprovecha su sistema de “buffering” (cargar pequeños extractos del video progresivamente) para optimizar este proceso. Además, de esta manera se evita ocupar más espacio en los servicios de Firebase, y se ahorra el proceso de implementación de pruebas, ya que se trata de un paquete externo que ya ha sido testeado previamente.

**2.2.3.4.4. Etapa de pruebas.** Teniendo en cuenta que en la funcionalidad de juegos no se hizo uso de clases de controlador ni servicio, tampoco se realizan pruebas unitarias para estas clases. En su lugar, se pasa directamente a las pruebas de integración para cada juego por separado. Esto es posible debido a que las clases de cada juego están desacopladas del resto de la aplicación, lo cual permite realizar pruebas con datos ficticios (en el caso de los niveles).

**Tabla 53**

*Pruebas de integración para funcionalidad de juegos*

<b>Id</b>	<b>Nombre</b>	<b>CID</b>	<b>Descripción escenario</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-071	Inicio de nivel - Ahorcado	HU07-C1	El usuario abre el juego del ahorcado	-	Se muestran los campos de letras correspondientes a la palabra del primer nivel
CP-072	El usuario acierta - Ahorcado	HU07-C2	El usuario realiza un acierto, pero no gana aún	El usuario selecciona una letra correcta	No se muestran cambios en el ahorcado
CP-073	El usuario falla - Ahorcado	HU07-C3	Usuario realiza un fallo, pero no pierde el juego aún	El usuario selecciona una letra incorrecta	Se dibuja una parte del ahorcado en pantalla
CP-074	El usuario pierde - Ahorcado	HU07-C4	El usuario alcanza el límite de fallos permitido	El usuario menciona múltiples letras incorrectas hasta	Se muestra un mensaje indicando que se ha perdido

				alcanzar el límite de fallos permitido	
CP-075	El usuario gana – Ahorcado	HU07-C5	El usuario gana el juego	El usuario completa las letras de la palabra del nivel	Se muestra un mensaje indicando que se ha ganado
CP-076	Inicio de nivel - Trivia	HU08-C1	El usuario abre el juego de trivia		Se muestra la pregunta del nivel y sus opciones de respuesta
CP-077	El usuario acierta - Trivia	HU08-C3	El usuario acierta la pregunta	El usuario selecciona una respuesta correcta	Se mostró un mensaje indicando que se ha completado el nivel
CP-078	El usuario falla - Trivia	HU08-C2	El usuario falla la pregunta	El usuario selecciona una respuesta incorrecta	Se muestra un mensaje indicando que se ha perdido el nivel

En la siguiente figura se indican los resultados de las pruebas de integración para el juego del ahorcado.

### Figura 64

*Resultados de pruebas de integración para juego de ahorcado*

```
Carlos@ZENBOOK MINGW64 /c/Projects/Luciapp (feature/games)
$ flutter test integration_test/hangman.dart -r expanded -d ZT322K96PJ
00:00 +0: loading C:/Projects/Luciapp/integration_test/hangman.dart
Running Gradle task 'assembleDebug'... 40.9s
✓ Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app-debug.apk... 8.3s
00:00 +0: (setUpAll)
00:00 +0: Integration Test [CP-071] Inicio de nivel - Ahorcado
00:01 +1: Integration Test [CP-072] El usuario acierta - Ahorcado
00:02 +2: Integration Test [CP-073] El usuario falla - Ahorcado
00:02 +3: Integration Test [CP-074] El usuario pierde - Ahorcado
00:03 +4: Integration Test [CP-075] El usuario gana - Ahorcado
00:08 +5: (tearDownAll)
00:08 +5: All tests passed!
```

**Figura 65**

*Resultados de pruebas de integración para juego de trivia*

```
Carlos@ZENBOOK MINGW64 /c/Projects/Luciapp (feature/games)
$ flutter test integration_test/trivia.dart -r expanded -d ZT322K96PJ
00:00 +0: loading C:/Projects/Luciapp/integration_test/trivia.dart
Running Gradle task 'assembleDebug'... 33.5s
✓ Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app-debug.apk... 8.3s
00:00 +0: (setUpAll)
00:00 +0: Integration Test [CP-076] Inicio de nivel - Trivia
00:01 +1: Integration Test [CP-077] El usuario acierta - Trivia
00:05 +2: Integration Test [CP-078] El usuario falla - Trivia
00:09 +3: (tearDownAll)
00:09 +3: All tests passed!
```

**2.2.3.5. Quinta iteración.** A partir de esta iteración, se inicia el desarrollo de la aplicación web de administrador, cuyo objetivo es proporcionar una interfaz gráfica amigable para interactuar con la base de datos que almacena los contenidos digitales de la aplicación móvil. Esta aplicación de administrador también se realizó con el framework Flutter, lo cual permite aprovechar componentes previamente desarrollados para la aplicación móvil. Siendo así, se seleccionan las historias de usuario HU-013, HU-014 y HU-015.

**2.2.3.5.1. Etapa de análisis.** Se llevó a cabo el proceso de especificación de historias de usuario.

**Tabla 54**

*Especificación historia de usuario 13*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-013</b>		
<b>Nombre:</b>	Crear curso		
<b>Actor:</b>	Usuario administrador		
<b>Descripción:</b>	Como usuario administrador quiero crear un nuevo curso para poder agregar contenidos a dicho curso		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando se ingrese el nombre del curso, la imagen de curso y la descripción correctamente	Se debe mostrar un mensaje de éxito y se debe guardar el curso
	2	Cuando el nombre del curso exceda los 50 caracteres	Se debe mostrar un mensaje de alerta indicando la restricción de caracteres
	3	Cuando la descripción exceda los 150 caracteres	Se debe mostrar un mensaje de alerta indicando la restricción de caracteres

4	Cuando alguno de los campos está vacío	Se debe mostrar un mensaje de alerta indicando que todos los campos son requeridos
5	Cuando no se seleccione ninguna imagen para el curso	Se debe mostrar un mensaje de alerta indicando que la imagen del curso es obligatoria

**Tabla 55**

*Especificación historia de usuario 14*

<b>Historia de Usuario</b>		
<b>Código:</b>	<b>HU-014</b>	
<b>Nombre:</b>	Crear juego [Ahorcado]	
<b>Actor:</b>	Usuario administrador	
<b>Descripción:</b>	Como usuario administrador quiero crear un nuevo juego de ahorcado para asociarlo a un curso existente	
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>
	1	Cuando no se ingrese ningún nivel
	2	Cuando los datos (palabra, pista) de todos los niveles ingresados sean correctos
	3	Cuando la palabra de un nivel exceda los 15 caracteres
		<b>Resultado</b>
		Se deben mostrar un mensaje de error indicando que no se puede crear un juego sin niveles
		Se debe mostrar un mensaje de éxito, y se debe agregar el juego al curso.
		Se debe mostrar un mensaje de alerta indicando la restricción de caracteres

4	Cuando la pista de un nivel exceda los 80 caracteres	Se debe mostrar un mensaje de alerta indicando la restricción de caracteres
5	Cuando alguno de los campos esté vacío	Se debe mostrar un mensaje de alerta indicando que los campos son requeridos

**Tabla 56**

*Especificación historia de usuario 15*

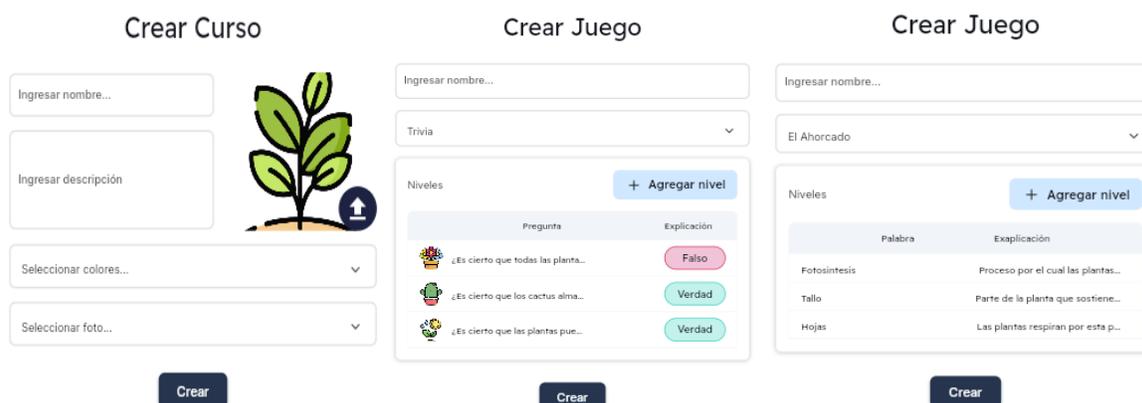
Historia de Usuario			
<b>Código:</b>	<b>HU-015</b>		
<b>Nombre:</b>	Crear juego [Trivia]		
<b>Actor:</b>	Usuario administrador		
<b>Descripción:</b>	Como usuario administrador quiero crear un nuevo juego de trivia para asociarlo a un curso existente		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando no se ingrese ningún nivel	Se deben mostrar un mensaje de error indicando que no se puede crear un juego sin niveles
	2	Cuando los datos (pregunta, respuesta, imagen, explicación) de todos los niveles ingresados sean correctos	Se debe mostrar un mensaje de éxito, y se debe agregar el juego al curso.
	3	Cuando la pregunta de un nivel exceda los 150 caracteres	Se debe mostrar un mensaje de alerta indicando la restricción de caracteres

4	Cuando la explicación de un nivel exceda los 250 caracteres	Se debe mostrar un mensaje de alerta indicando la restricción de caracteres
5	Cuando alguno de los campos de algún nivel esté vacío	Se debe mostrar un mensaje de alerta indicando que todos los campos deben estar llenos para seguir

**2.2.3.5.2. Etapa de diseño.** Se realizaron las interfaces de usuario en la herramienta FlutterFlow, esta vez con un tamaño de pantalla de escritorio. A pesar de ser una aplicación separada, se reutilizó componentes y diseños de la aplicación móvil, como la pantalla de autenticación y logotipos. Para el diseño de tablas se utilizó componentes por defecto de FlutterFlow.

**Figura 66**

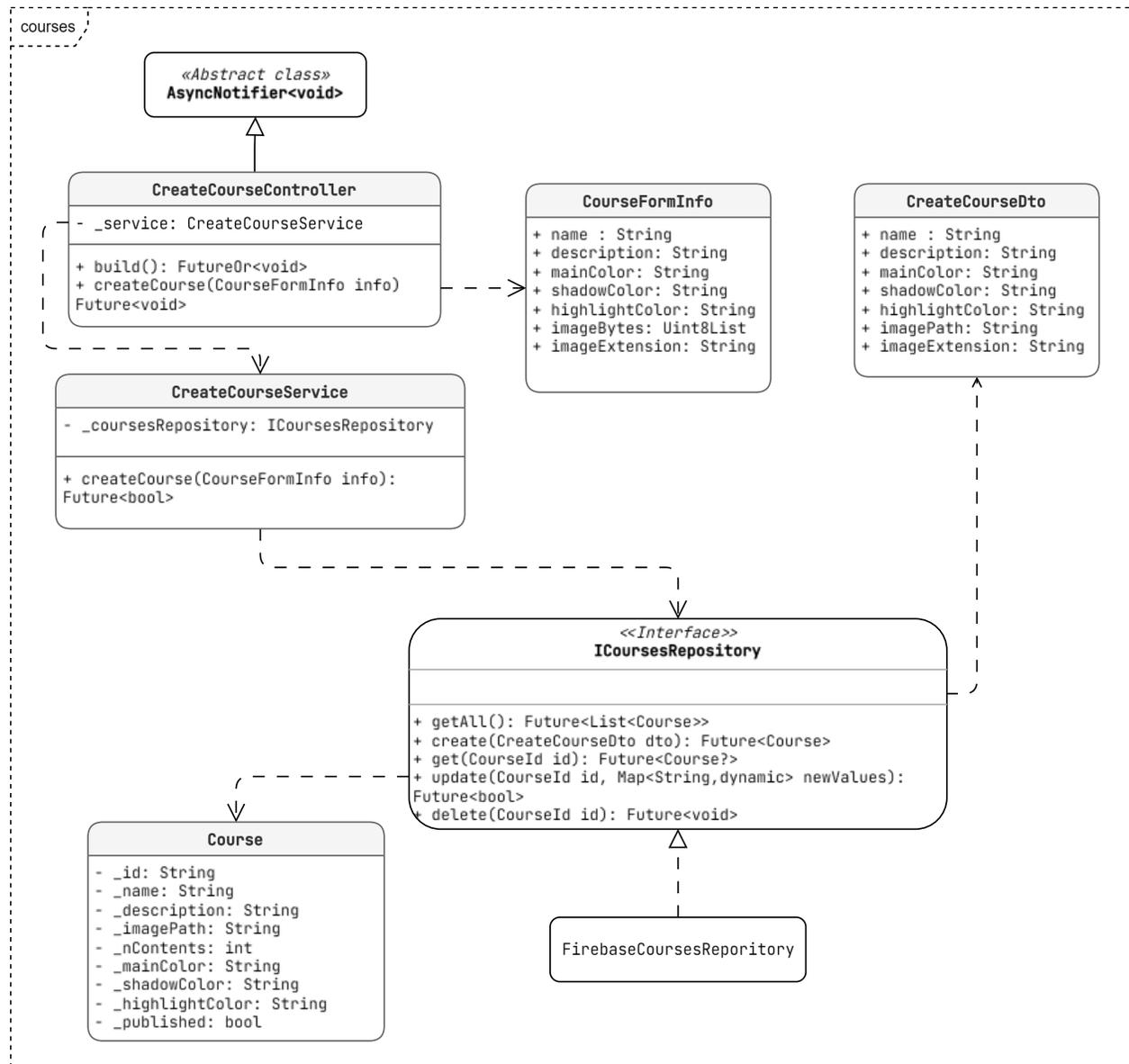
*Diseño de interfaz de usuario – creación de cursos y juegos*



Como se utiliza la misma base de datos que la aplicación móvil, se utilizó también el mismo diseño ilustrado en la Figura 46. Por consiguiente, el diagrama UML utiliza la misma interfaz para definir el repositorio de cursos, agregando el resto de los métodos para completar las operaciones CRUD.

**Figura 67**

Diagrama UML para funcionalidad de cursos (administrador)

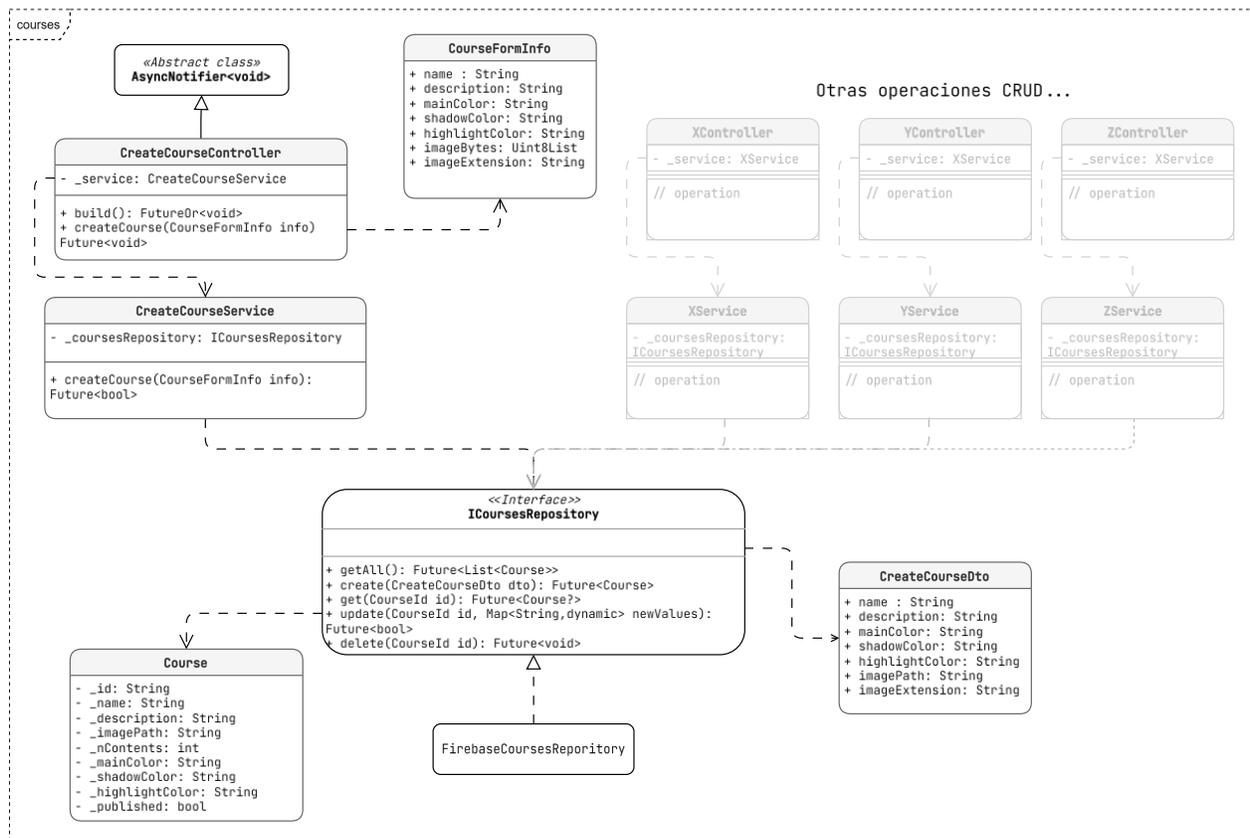


Para la construcción de la aplicación web de administrador, se implementó una estrategia distinta para el manejo del estado. En vista de que las operaciones a realizar son operaciones CRUD sencillas, los controladores extienden a la clase abstracta AsyncNotifier (en lugar de StateNotifier), de manera que exponen un estado que puede tener tres valores por defecto: éxito, cargando y error. Con esta estrategia, los componentes de interfaz de usuario que observen o llamen al controlador

solo tendrán acceso a estos tres tipos de valores, simplificando el proceso de renderización. Las clases de servicio se mantienen igual, conservando su única función de ejecutar lógica de negocio y comunicarse con la capa de datos. Así, para agregar más funcionalidades CRUD, simplemente se agrega el controlador y el servicio correspondiente, como se muestra a continuación:

**Figura 68**

*Diagrama UML para funcionalidad de cursos – operaciones CRUD*

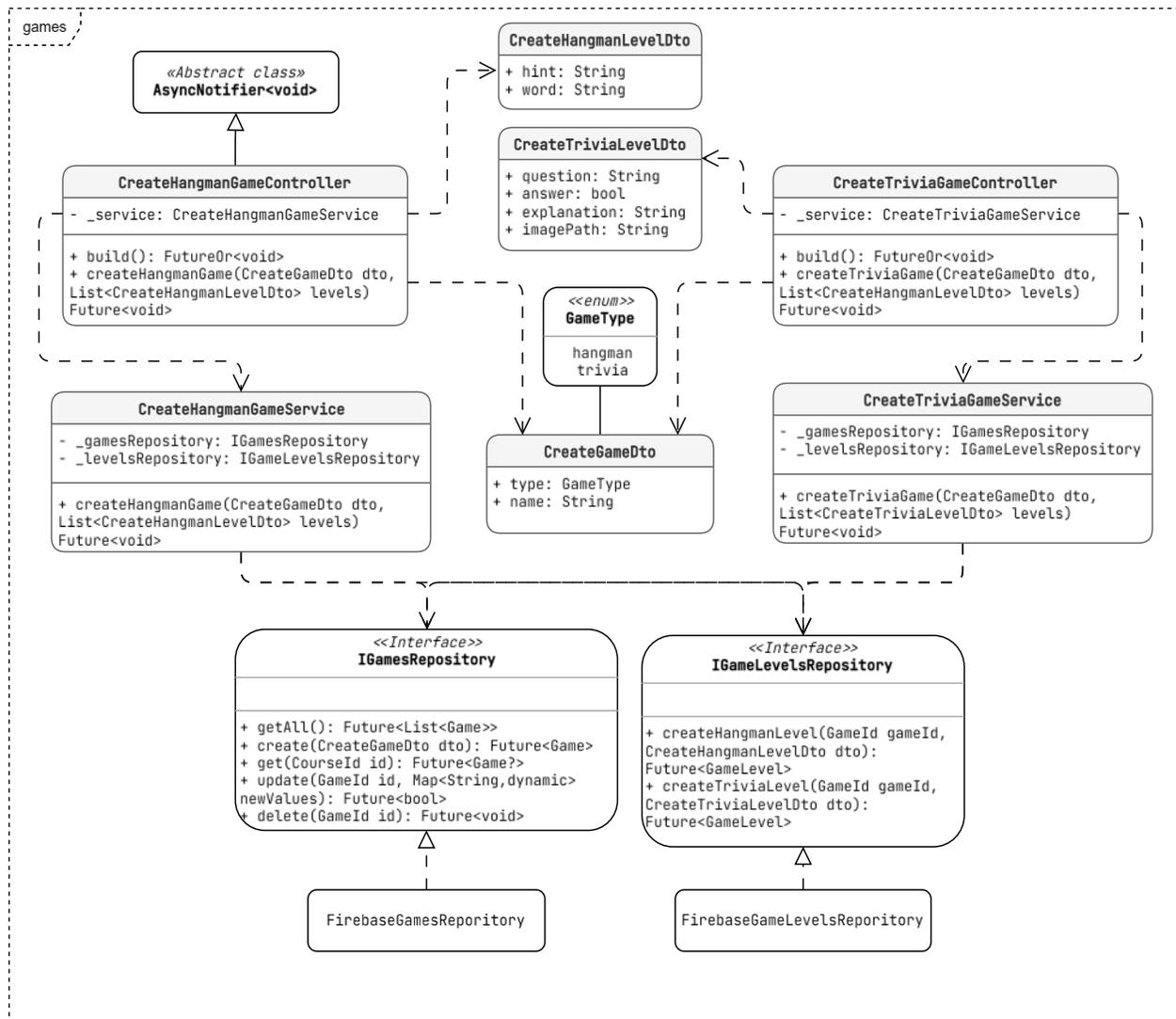


Otro punto por destacar es que se hace uso dos clases extra para manejar el intercambio de información entre capas. La primera es CourseFormInfo, que se encarga de almacenar la información obtenida al diligenciar el formulario de creación de curso. Nótese que la propiedad imageBytes es de tipo Uint8List, es decir que se trata de la imagen en bytes. En contraste, la propiedad que hace referencia a la imagen del curso en la clase CreateCourseDto es un String que corresponde al URL de la imagen, una vez almacenada en el servicio de Firebase Storage. Por lo

tanto, CreateCourseDto es la clase que se usa para insertar datos directamente en la colección de cursos. Siguiendo el mismo principio, se elaboró el diagrama UML para la funcionalidad de juegos.

**Figura 69**

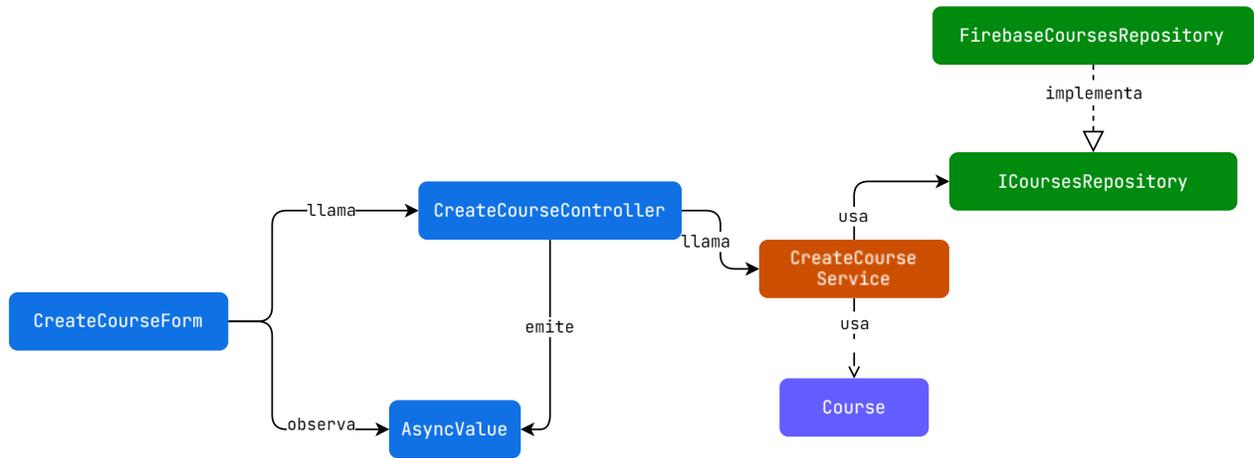
*Diagrama UML para funcionalidad de juegos (administrador)*



**2.2.3.5.3. Etapa de codificación.** Como se mencionó previamente, se simplificó la estrategia para el manejo de estado debido a que las funcionalidades del administrador son operaciones CRUD sencillas. De la misma manera, se simplifica la comunicación de componentes.

**Figura 70**

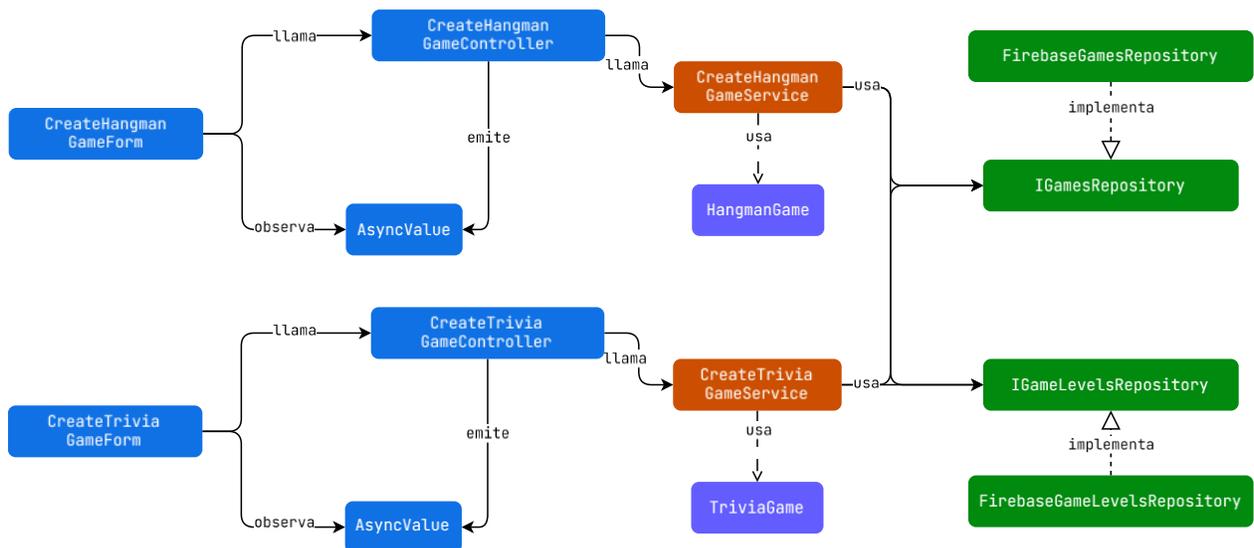
*Comunicación de componentes – Creación de cursos*



En este caso, un mismo componente se hará cargo de llamar y observar al estado del controlador. Para la operación de creación de cursos, el componente encargado de esto será el formulario correspondiente. Lo mismo sucede en el caso de los formularios para crear juegos de ahorcado y trivia.

**Figura 71**

*Comunicación de componentes – Creación de cursos*



Además, para agilizar el proceso de codificación se utilizó el mismo esquema de autenticación desarrollado para la aplicación móvil. No obstante, esta vez se utilizó únicamente la estrategia de correo-contraseña y no se incluyó autenticación por Google ni Facebook. Esto debido a que la aplicación de administrador limita el acceso a determinada cantidad de usuarios.

**2.2.3.5.4. Etapa de pruebas.** La aplicación de administrador posee operaciones que incluyen la subida y descarga de imágenes del servicio de Firebase Storage. Este proceso requiere realizar peticiones HTTP a la API correspondiente. Sin embargo, al realizar testing con Flutter, el cliente HTTP se bloquea por defecto. Esto dificulta el proceso de testing para operaciones como crear cursos, pues hace necesario sobrescribir el cliente HTTP con cada imagen que se esté manejando, lo cual resulta ineficiente al realizar este tipo de pruebas. Por este motivo, se optó por realizar pruebas de caja negra para las funcionalidades de la aplicación administrador.

Como lo menciona (Jovanović, 2009), las pruebas de caja negra se basan en requerimientos de salida, sin tener en cuenta la estructura interna del programa. Es decir, que se realizan de manera similar a las pruebas de integración, pero centrándose únicamente en las entradas y salidas esperadas. Así, se elaboraron los siguientes casos de prueba.

**Tabla 57**

*Pruebas de caja negra - Iteración 5*

<b>Id</b>	<b>Nombre</b>	<b>CID</b>	<b>Descripción escenario</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-079	Creación de curso exitosa	HU13-C1	El administrador ingresa los datos del curso correctamente	Nombre, Descripción, Imagen	Se muestra un mensaje de éxito y se crea el curso

CP-080	Creación de curso con campos vacíos	HU13-C4	El administrador deja campos vacíos en la creación del curso	-	Se muestra un mensaje de alerta en cada campo vacío
CP-081	Creación de curso sin imagen	HU13-C5	El administrador no selecciona ninguna imagen	-	Se muestra un mensaje de alerta indicando que se debe subir una imagen
CP-082	Creación de curso excediendo el límite de caracteres	HU13-C2, HU13-C3	El administrador excede el número máximo de caracteres en algún campo	Nombre con más de 50 caracteres – Descripción con más de 150 caracteres	Se muestra un mensaje indicando que se ha excedido el número máximo de caracteres en cada campo que incumpla con la condición
CP-083	Creación de juego sin niveles - Ahorcado	HU14-C1	El administrador no ingresa ningún nivel al juego del ahorcado	-	Se muestra un mensaje de alerta indicando que debe haber mínimo un nivel en el juego
CP-084	Creación de juego con campos excediendo el límite de	HU14-C3, HU14-C4	El administrador excede el número máximo de	Nombre con más de 50 caracteres – Palabra con más de 15 caracteres –	Se muestra un mensaje indicando que se ha excedido el número máximo de caracteres en cada campo que

	caracteres - Ahorcado		caracteres en algún campo	Pista con más de 80 caracteres	incumpla con la condición
CP- 085	Creación de juego con campos vacíos - Ahorcado	HU14- C5	El administrador deja algún campo vacío en la creación del juego de ahorcado	-	Se muestra un mensaje de alerta indicando que todos los campos son requeridos
CP- 086	Creación de juego exitosa - Ahorcado	HU14- C2	El administrador crea el juego con todos los datos correctos	Lista de niveles de ahorcado con al menos un nivel	Se muestra un mensaje de éxito y se crea el juego
CP- 087	Creación de juego sin niveles - Trivia	HU15- C1	El administrador no ingresa ningún nivel al juego de trivia		Se muestra un mensaje de alerta indicando que debe haber mínimo un nivel en el juego
CP- 088	Creación de juego exitosa - Trivia	HU15- C2	El administrador crea el juego con todos los datos correctos	Lista de niveles de trivia con al menos un nivel	Se muestra un mensaje de éxito y se crea el juego
CP- 089	Creación de juego con	HU15- C5	El administrador deja algún	-	Se muestra un mensaje de alerta indicando que

campos vacíos - Trivia	campo vacío en la creación del juego de trivia	todos los campos son requeridos
Creación de juego con campos excediendo el límite de caracteres - Trivia	El administrador excede el número máximo de caracteres en algún campo	Nombre con más de 50 caracteres – Pregunta con más de 150 caracteres – Explicación con más de 250 caracteres
CP-090	HU15-C3, HU15-C4,	Se muestra un mensaje indicando que se ha excedido el número máximo de caracteres en cada campo que incumpla con la condición

Posteriormente, se realizan las pruebas de caja negra con la aplicación en ejecución.

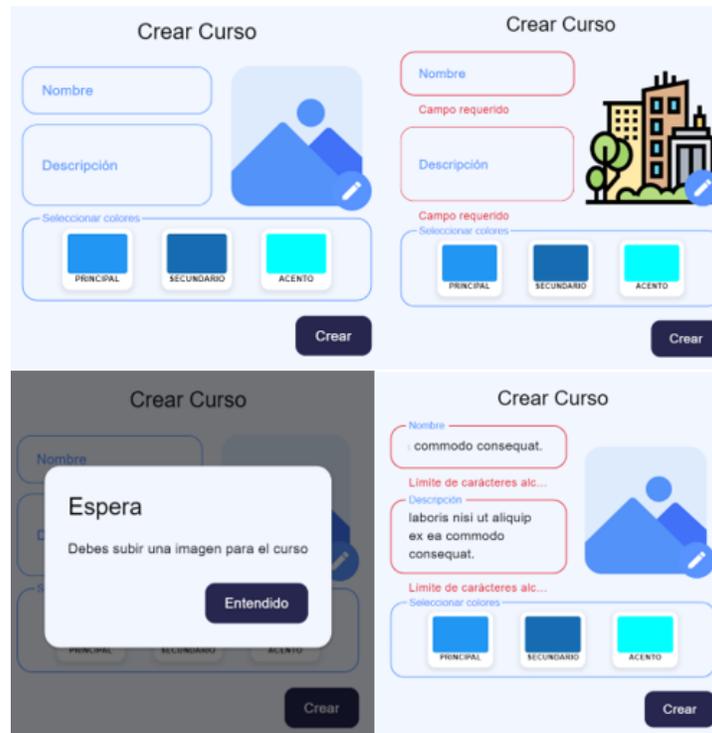
**Figura 72**

*Prueba de creación de cursos exitosa (CP-079)*



**Figura 73**

Pruebas de creación de cursos fallida (CP-080, CP-081, CP-082)



**Figura 74**

Pruebas de creación de juego de ahorcado fallida (CP-083, CP-084, CP-085)



**Figura 75**

*Prueba de creación de juego de ahorcado exitosa (CP-086)*



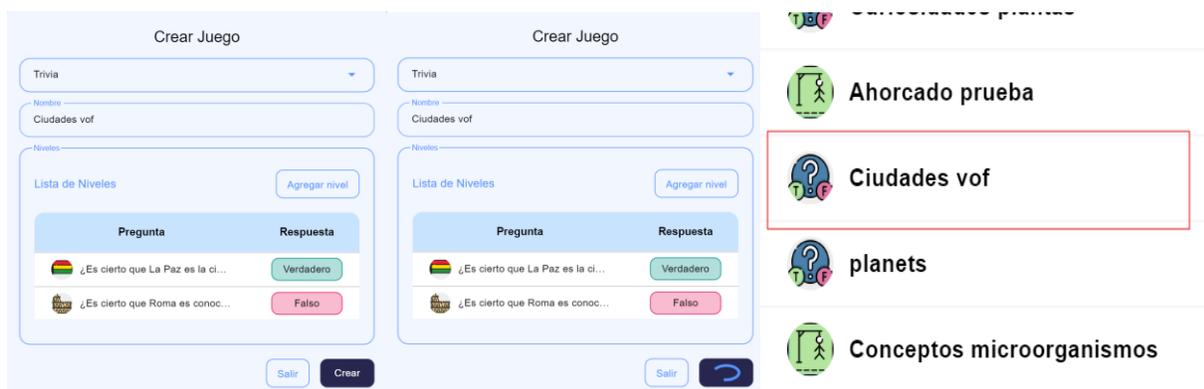
**Figura 76**

*Prueba de creación de juego de trivia fallida (CP-087, CP-088, CP-089)*



**Figura 77**

*Prueba de creación de juego de trivia exitosa (CP-090)*



**2.2.3.6. Sexta iteración.** Finalmente, en la sexta iteración se desarrolló las historias de usuario restantes: HU-016, HU-017 y HU-018, referentes a agregar videos, agregar juegos y publicar curso, respectivamente.

**2.2.3.6.1. Etapa de análisis.** Se especificó las historias de usuario seleccionadas para esta iteración.

**Tabla 58**

*Especificación historia de usuario 16*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-016</b>		
<b>Nombre:</b>	Agregar video a curso		
<b>Actor:</b>	Usuario administrador		
<b>Descripción:</b>	Como usuario administrador quiero agregar un nuevo video a un curso existente para que los usuarios con discapacidad visual puedan acceder a dicho video mediante el curso existente.		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando los datos se ingresen correctamente	Se debe agregar el video al curso
	2	Cuando el enlace del video no sea válido	Se deben mostrar un mensaje de alerta indicando que el enlace ingresado no es válido
	3	Cuando alguno de los campos esté vacío	Se debe mostrar un mensaje de alerta en los campos requeridos
	4	Cuando el nombre del video supere los 50 caracteres	Se debe mostrar un mensaje de alerta en el campo que ha

		superado el límite de caracteres permitido
5	Cuando la descripción del video supere los 50 caracteres	Se debe mostrar un mensaje de alerta en el campo que ha superado el límite de caracteres permitido

**Tabla 59**

*Especificación historia de usuario 17*

<b>Historia de Usuario</b>		
<b>Código:</b>	<b>HU-017</b>	
<b>Nombre:</b>	Agregar juego a curso	
<b>Actor:</b>	Usuario administrador	
<b>Descripción:</b>	Como usuario administrador quiero agregar un nuevo video a un curso existente para que los usuarios con discapacidad visual puedan acceder a dicho video mediante el curso existente.	
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>
	1	Cuando los datos se ingresen correctamente
	2	Cuando alguno de los campos esté vacío
	3	Cuando el nombre del juego supere los 50 caracteres
	4	Cuando la descripción del juego supere los 50 caracteres
		<b>Resultado</b>
		Se debe agregar el video al curso
		Se debe mostrar un mensaje de alerta en los campos requeridos
		Se debe mostrar un mensaje de alerta en el campo que ha superado el límite de caracteres permitido
		Se debe mostrar un mensaje de alerta en el campo que ha

superado el límite de caracteres permitido

**Tabla 60**

*Especificación historia de usuario 18*

<b>Historia de Usuario</b>			
<b>Código:</b>	<b>HU-018</b>		
<b>Nombre:</b>	Publicar curso		
<b>Actor:</b>	Usuario administrador		
<b>Descripción:</b>	Como usuario administrador quiero publicar un curso ya creado para que esté disponible en la aplicación.		
<b>Criterios de Aceptación</b>	<b>CID</b>	<b>Condición</b>	<b>Resultado</b>
	1	Cuando el curso tenga al menos un contenido (vídeos o juegos)	Se debe mostrar un mensaje de éxito, y se debe publicar el curso
	2	Cuando el curso no tenga contenidos (vídeos o juegos)	Se deben mostrar un mensaje de alerta, indicando que debe haber al menos un contenido para publicar el curso

**2.2.3.6.2. Etapa de diseño.** Se inició esta etapa con la elaboración de las interfaces de usuario para agregar videos y juegos a un curso existente.

**Figura 78**

*Diseño de interfaz de usuario para creación de cursos*

The image shows two identical forms side-by-side, both titled "Agregar Contenido".

The left form contains the following fields from top to bottom:

- A dropdown menu with "Juego" selected and a downward arrow.
- A text input field with "Nombre" as a placeholder.
- A text input field with "Descripción" as a placeholder.
- A dropdown menu with "Seleccionar Juego" selected and a downward arrow.
- A dark blue button with the text "Crear" in white.

The right form contains the following fields from top to bottom:

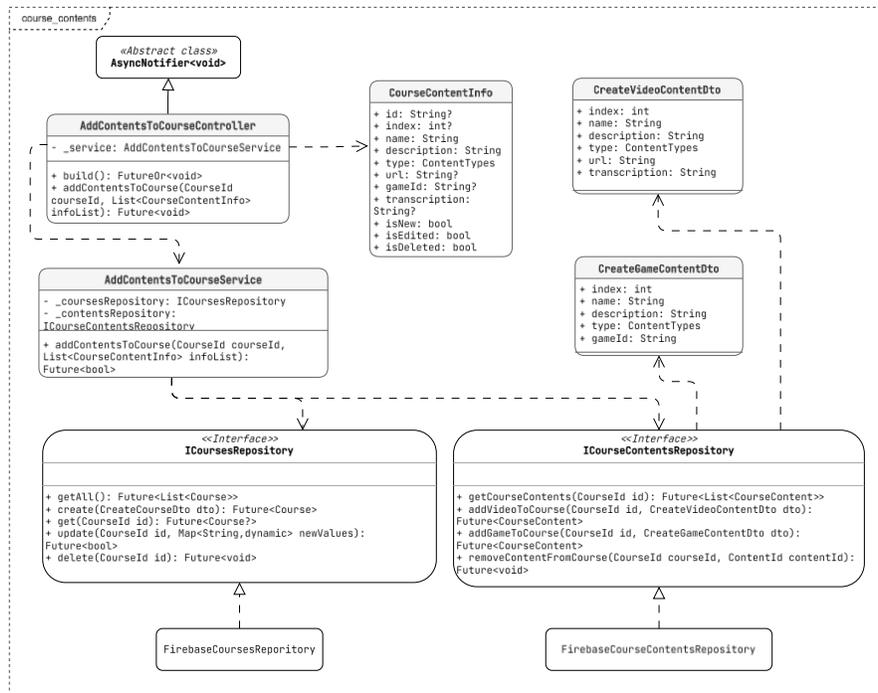
- A dropdown menu with "Video" selected and a downward arrow.
- A text input field with "Nombre" as a placeholder.
- A text input field with "Descripción" as a placeholder.
- A text input field with "Link" as a placeholder.
- A text input field with "Transcripción" as a placeholder.
- A dark blue button with the text "Crear" in white.

Al igual que en la anterior iteración, se utiliza la misma base de datos de la aplicación móvil desarrollada, en este caso, corresponde al diseño descrito en la Figura 46.

y la Figura 58. Estructura de colección de juegos. Con base en ello, se realiza el diagrama UML, que reutiliza y extiende los repositorios de cursos y contenidos de curso mostrados en la Figura 47.

### **Figura 79**

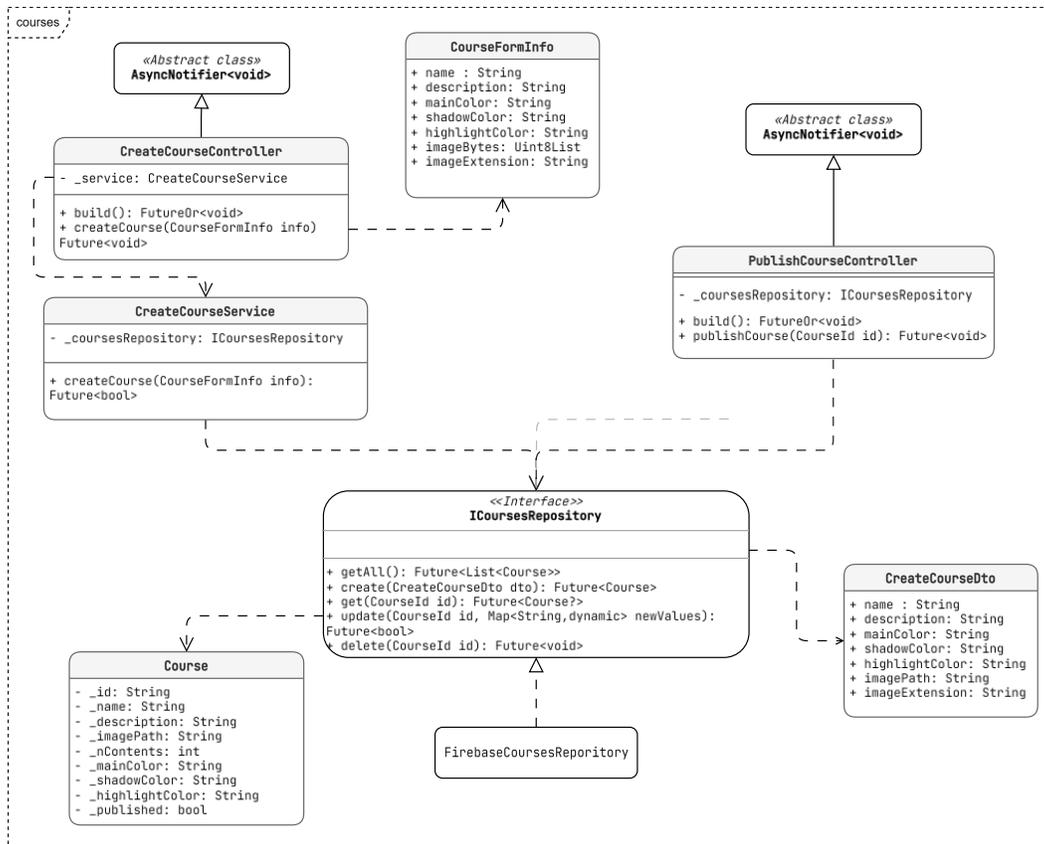
*Diagrama UML para funcionalidad de contenidos de cursos*



Como lo indica el diagrama, se utilizan los repositorios de cursos y contenidos de curso utilizados en la aplicación móvil. En adición, se utiliza la clase llamada CourseContentInfo para almacenar la información de cada contenido, independientemente del tipo de contenido (video o juego). Para cada tipo de contenido, existe un DTO (Data Transfer Object) que es utilizado para representar los datos de cada formulario una vez diligenciado.

Por otro lado, la funcionalidad de publicar cursos se agregó como una operación CRUD más en el diagrama de la Figura 68, pues en esencia es una operación de actualización para la propiedad “published” de un curso. Por lo tanto, el diagrama UML de cursos completo resulta de la siguiente manera.

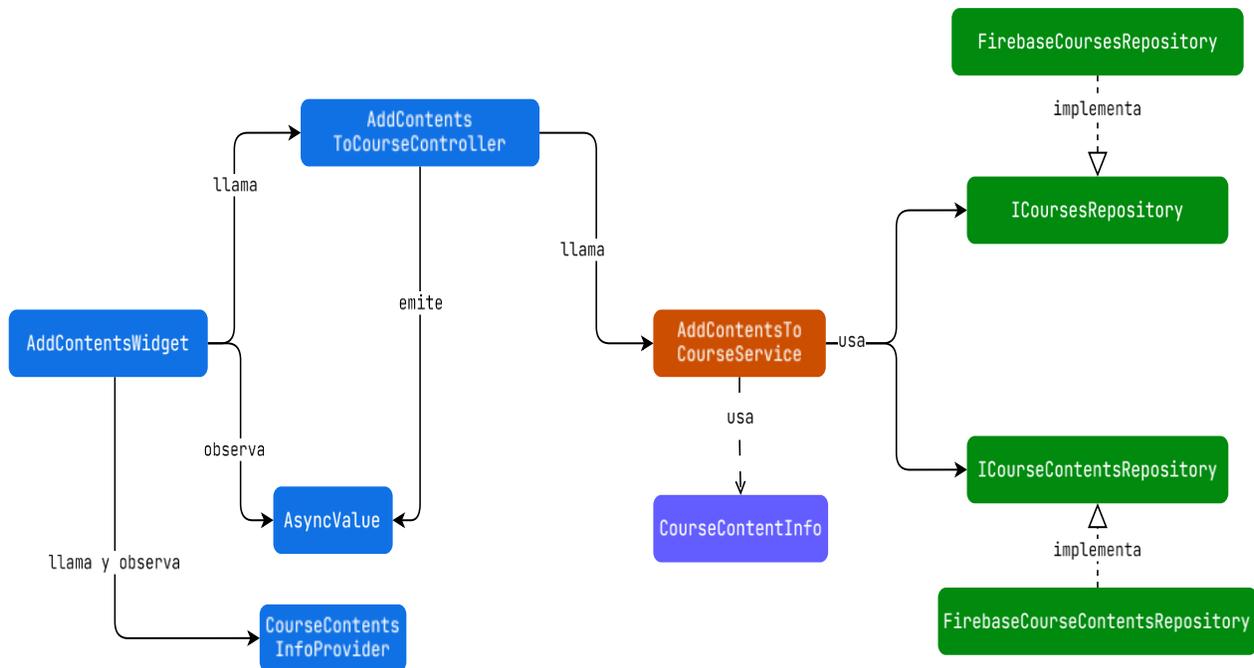
**Figura 80**  
 Diagrama UML para funcionalidad de cursos (Administrador)



2.2.3.6.3. *Etapa de codificación.* Siguiendo el diagrama elaborado en la etapa de diseño, se procede a implementar las funcionalidades al proyecto.

**Figura 81**

*Comunicación de componentes – Agregar contenido a curso*

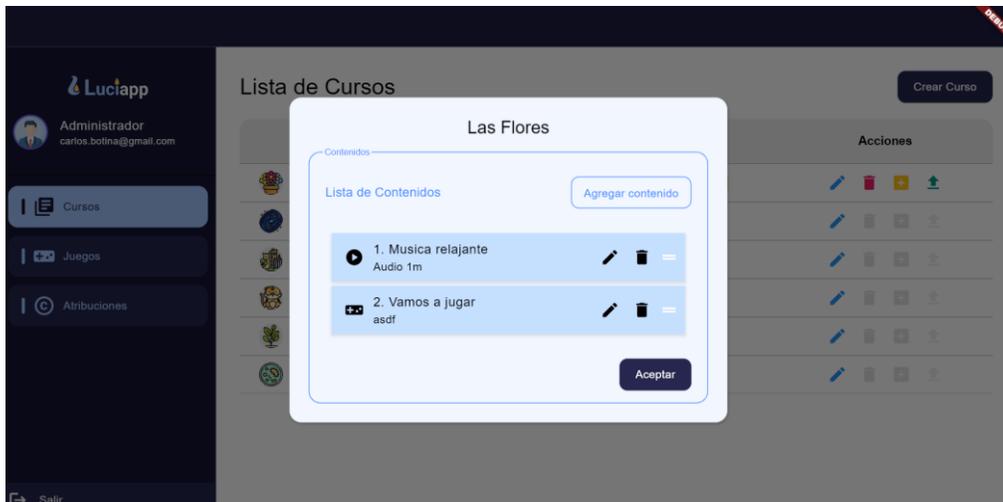


En este caso, se utilizó un funcionamiento diferente al utilizado anteriormente. Para agregar contenidos al curso, primero se almacena la información de dichos contenidos de manera local, por medio de la clase `CourseContentsInfoProvider`. Esto con el fin de realizar escrituras a la base de datos únicamente cuando la lista de contenidos esté completa.

Así, para agregar un contenido primero se agrega a la lista en memoria, y posteriormente, al presionar el botón de guardar cambios, se llama al controlador `AddContentsToCourseController` para que realice el proceso para cada uno de los contenidos. De este modo, el administrador podrá visualizar los contenidos de un curso antes de agregarlos a éste. Además, haciendo uso de componentes especiales de Flutter, es posible reordenar la lista de contenidos de manera sencilla con un sistema sencillo de arrastrar y soltar.

### Figura 82

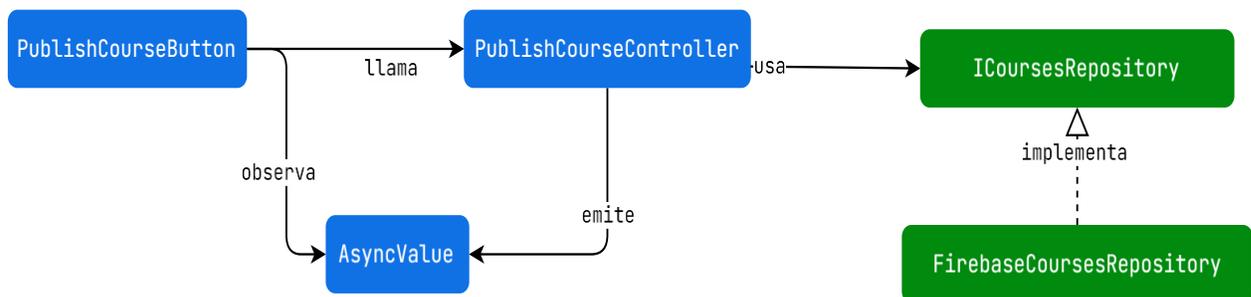
*Panel de contenidos de cursos en aplicación de administrador*



Por otro lado, la comunicación de componentes en el proceso de publicación de curso es más sencillo, pues el controlador se comunica directamente con el repositorio para actualizar el curso en cuestión.

**Figura 83**

*Comunicación de componentes – Publicación de cursos*



**2.2.3.6.4. Etapa de pruebas.** Al igual que en la quinta iteración, se realizaron pruebas de caja negra en la aplicación de administrador.

**Tabla 61**

*Pruebas de caja negra – Iteración 6*

<b>Id</b>	<b>Nombre</b>	<b>CID</b>	<b>Descripción escenario</b>	<b>Entrada</b>	<b>Resultado esperado</b>
CP-091	Agregar video a curso exitosamente	HU17-C1	El administrador ingresa los datos del video correctamente	Nombre, Descripción, Link, Transcripción	Se muestra un mensaje de éxito y se agrega el video
CP-092	Agregar video a curso con enlace no válido	HU17-C2	El administrador ingresa un enlace no válido	Enlace no válido	Se muestra un mensaje de alerta indicando que se debe ingresar un enlace de YouTube
CP-093	Agregar video a curso con campos vacíos	HU17-C3	El administrador deja campos vacíos al agregar el video	-	Se muestra un mensaje de alerta indicando aquellos campos que son requeridos
CP-094	Agregar video a curso excediendo el límite de caracteres	HU17-C4, HU17-C5	El administrador excede el límite de caracteres permitido en un campo de texto	Nombre con más de 50 caracteres, Descripción con más de 50 caracteres	Se muestra un mensaje de alerta indicando que se ha excedido el número de caracteres permitido
CP-095	Agregar juego a	HU18-C1		-	Se muestra un mensaje de

	curso exitosamente				éxito y se agrega el juego
CP-096	Agregar juego a curso con campos vacíos	HU18-C2	El administrador ingresa los datos del juego correctamente	Nombre, Descripción, Juego	Se muestra un mensaje de alerta indicando aquellos campos que son requeridos
CP-097	Agregar juego a curso sin seleccionar un juego	HU18-C2	El administrador no selecciona ningún juego en el formulario	-	Se muestra un mensaje de alerta indicando que es necesario seleccionar un juego
CP-098	Agregar juego a curso excediendo el límite de caracteres	HU18-C3, HU18-C4	El administrador deja campos vacíos al agregar el juego	-	Se muestra un mensaje de alerta indicando aquellos contenidos que son requeridos
CP-099	Publicar curso con al menos un contenido	HU19-C1	El administrador publica un curso que contiene al menos un contenido	-	Se muestra un mensaje de éxito y se publica el curso.
CP-100	Publicar curso sin contenidos	HU19-C2	El administrador publica un	-	Se muestra un mensaje de alerta indicando que para

curso sin  
contenidos

publicar un  
curso este debe  
tener al menos  
un contenido.

**Figura 84**

*Prueba de agregar video a curso exitosamente (CP-091)*

The screenshot shows a form titled 'Agregar contenido' for a course named 'Ciudades del Mundo'. The form has several fields: 'Video' (set to 'Video'), 'Nombre' (containing 'Datos sobre ciudades'), 'Description' (containing 'Video 1m 1s'), 'Link' (containing 'https://www.youtube.com/watch?v=sca4VG9b0NY'), and 'Transcripcion' (containing a paragraph about cities). A dark blue 'Agregar' button is at the bottom. To the right, a 'Contenidos' panel shows a 'Lista de Contenidos' with two items: '1. Vamos a jugar! El ahorcado' and '2. Vamos a jugar! Trivia'. Each item has edit and delete icons. An 'Agregar contenido' button is at the top right of the list, and an 'Aceptar' button is at the bottom right.

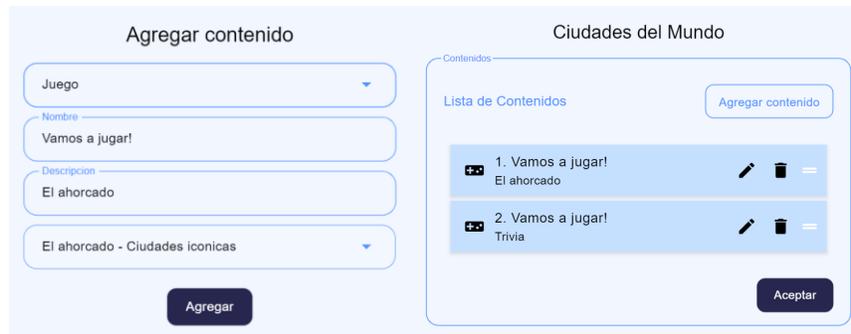
**Figura 85**

*Pruebas de agregar video a curso con fallos (CP-092, CP-093, CP-094)*

Three screenshots of the 'Agregar contenido' form showing validation errors. The first shows 'Campo requerido' for 'Nombre', 'Description', 'Link', and 'Transcripcion'. The second shows 'Limite de caracteres alcanzado' for 'Description' and 'Link', and 'Debe ingresar un link' for 'Transcripcion'. The third shows 'El link debe ser de youtube' for 'Link'.

**Figura 86**

*Prueba de agregar juego a curso exitosamente (CP-095)*



**Figura 87**

*Prueba de agregar juego a curso con fallos (CP-096, CP-097, CP-099)*



**Figura 88**

*Prueba de publicar curso sin contenidos (CP-099)*



**Figura 89**

*Prueba de publicar curso con contenidos (CP-100)*



#### 2.2.4. Fase de producción

Una vez desarrolladas todas las funcionalidades del Product Backlog, la aplicación móvil está lista para ser desplegada, haciéndola disponible al público. Como se trata de una aplicación para la plataforma Android, se usa a Google PlayStore como medio de difusión. Para ello, fue necesario crear una cuenta de desarrollador independiente y verificarla. Este proceso tuvo una duración de tres días.

Posteriormente, se inicia con la creación de la aplicación, registrando su nombre, eslogan y adjuntando recursos gráficos obligatorios como el ícono de la aplicación, imágenes de publicitarias, capturas de pantalla en dispositivos de distintos tamaños y más. Luego, la aplicación fue sometida a una clasificación de rangos de edad para varias regiones del mundo. Esto se llevó a cabo diligenciando varios formularios sobre su contenido. Como resultado de este proceso, la aplicación fue catalogada como apta para mayores de 3 años en la mayoría de las regiones disponibles.

#### Figura 90

*Clasificación de la aplicación en PlayStore*

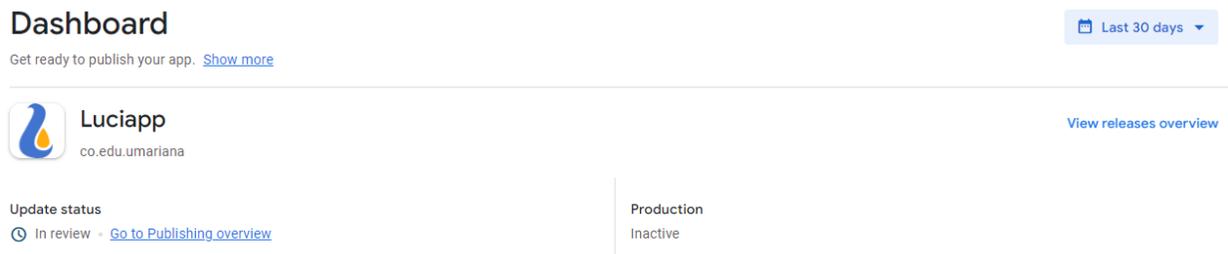
Your ratings		
<b>Brazil</b> Rating authority: Classificação Indicativa (Classind)	<b>Europe</b> Rating authority: Pan-European Game Information (PEGI)	<b>Rest of world</b> Rating authority: IARC Generic
Rating:  All ages	Rating:  PEGI 3	Rating:  Rated for 3+
Content descriptors: -	Content descriptors: -	Content descriptors: -
<b>North America</b> Rating authority: Entertainment Software Rating Board (ESRB)	<b>Germany</b> Rating authority: Unterhaltungssoftware Selbstkontrolle (USK)	<b>Russia</b> Rating authority: Google Play
Rating:  Everyone	Rating:  USK: All ages	Rating:  Rated for 3+
Content descriptors: -	Content descriptors: -	Content descriptors: -

Nota: Captura tomada de la consola de Google Play

Luego de este proceso, se agregó el “bunde” de la aplicación, un archivo que incluye todo el código compilado de la aplicación y sus recursos. Con este paso completado, se generó la primera versión de la aplicación, la cual fue enviada a Google para revisión. En esta revisión, Google instala la aplicación en varios dispositivos Android, para luego rastrear distintos tipos de problemas y generar un informe detallado con los resultados obtenidos.

## Figura 91

*Primera versión de la aplicación en revisión*



The screenshot shows the Google Play Console dashboard for the application 'Luciapp'. At the top, it says 'Dashboard' and 'Get ready to publish your app. [Show more](#)'. There is a filter for 'Last 30 days'. Below this, the app's logo and name 'Luciapp' are shown, along with the developer 'co.edu.umariana' and a link to 'View releases overview'. The 'Update status' section shows 'In review' with a link to 'Go to Publishing overview'. The 'Production' section shows 'Inactive'.

Nota: Captura tomada de la consola de Google Play

Cinco días después, la versión enviada fue aprobada por Google. Esta validación se hace para identificar problemas de rendimiento, seguridad o contenidos. Como resultado, la aplicación cumplió con todos los estándares necesarios para ser publicada. No se detectó ningún problema de seguridad ni rendimiento. De hecho, se obtuvieron métricas muy positivas en las pruebas de rendimiento realizadas por Google en distintos dispositivos móviles.

## Figura 92

### Resultados de pruebas de rendimiento en la aplicación

Device	Avg. CPU	Avg. network sent	Avg. network received	Avg. memory	Cold start time	
<span>✔</span> Test devices without issues						
google Pixel 5 64-bit only	2.39%	0 B	0 B	127 MB	-	→
google Pixel 6	2.51%	0 B	0 B	143 MB	436 ms	→
google Pixel 5	1.74%	0 B	0 B	141 MB	705 ms	→
motorola Motorola G20	3.12%	0 B	0 B	107 MB	940 ms	→
google Pixel 7	1.88%	0 B	0 B	133 MB	329 ms	→

Nota: Captura tomada de la consola de Google Play

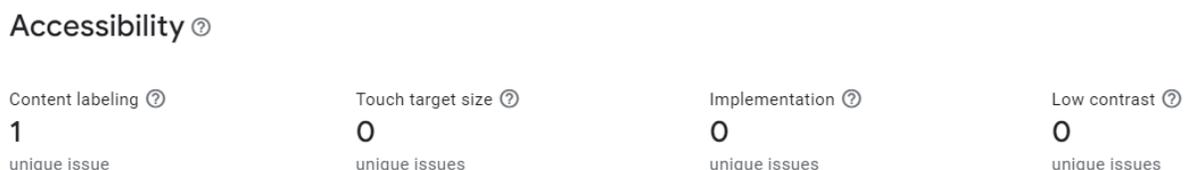
Como se aprecia en la Figura 92, el consumo en promedio de CPU va desde un 1.74% hasta un 3.12%, lo cual representa un uso muy bajo, que se ve reflejado en el tiempo de arranque “en frío” de la aplicación (iniciar la aplicación sin que se esté siendo ejecutada en segundo plano), el cual en ninguno de los dispositivos supera los 1000 milisegundos de duración.

En el apartado de accesibilidad, se identificó un solo problema de accesibilidad. Al, al revisar su reporte, este se refería a un único componente sin etiquetar, que correspondía a la pantalla de

carga al iniciar sesión. Por lo tanto, este problema no trae inconvenientes para el usuario final, debido a que esta pantalla de carga se muestra en un periodo de tiempo muy corto, insuficiente para ser leído por un lector de pantalla.

### Figura 93

*Resultados de pruebas de accesibilidad en aplicación*



Nota: Captura tomada de la consola de Google Play

Con estos resultados, la aplicación fue aprobada por Google, lo cual permitía continuar con el siguiente paso, el cual fue la prueba cerrada. En 2023, se produjo una serie de cambios en las políticas para publicar nuevas aplicaciones en Play Store (Google, 2023). Debido a esto, antes de poder lanzar la aplicación al mercado fue necesario ejecutar una prueba cerrada durante 14 días con 20 testers, quienes tuvieron acceso anticipado a la aplicación directamente desde la PlayStore.

### Figura 94

*Aplicación en acceso anticipado de PlayStore*

**Luciap**  
Carlos Botina

10+ Downloads | Everyone

Install | Share | Add to wishlist

This app is available for your device

**Contenidos digitales para todos**  
Temas de color (Para todos los gustos!)  
Variedad de contenidos (Audios, videos, minijuegos...)  
Minijuegos interactivos (¡Aprende mientras te diviertes!)  
Apariencia personalizable (Adaptado a tus preferencias)

**El reino animal**  
Bienvenida (Video en 3D)  
Introducción (Video en 3D)  
Los Vertebrados (Video en 3D)  
Vamos a jugar! (El ahorcado)

**App support**  
Support email: carlosbotti78@gmail.com  
Privacy Policy

**About the developer**  
Carlos Alberto Botina Carpio  
carlosbotti78@gmail.com  
Colombia

**About this app** →

Luciap is a mobile application developed as part of the project titled "Accessibility applied to the creation of digital content for people with visual disabilities" of the Systems Engineering program at the Mariana University. The application offers various digital content

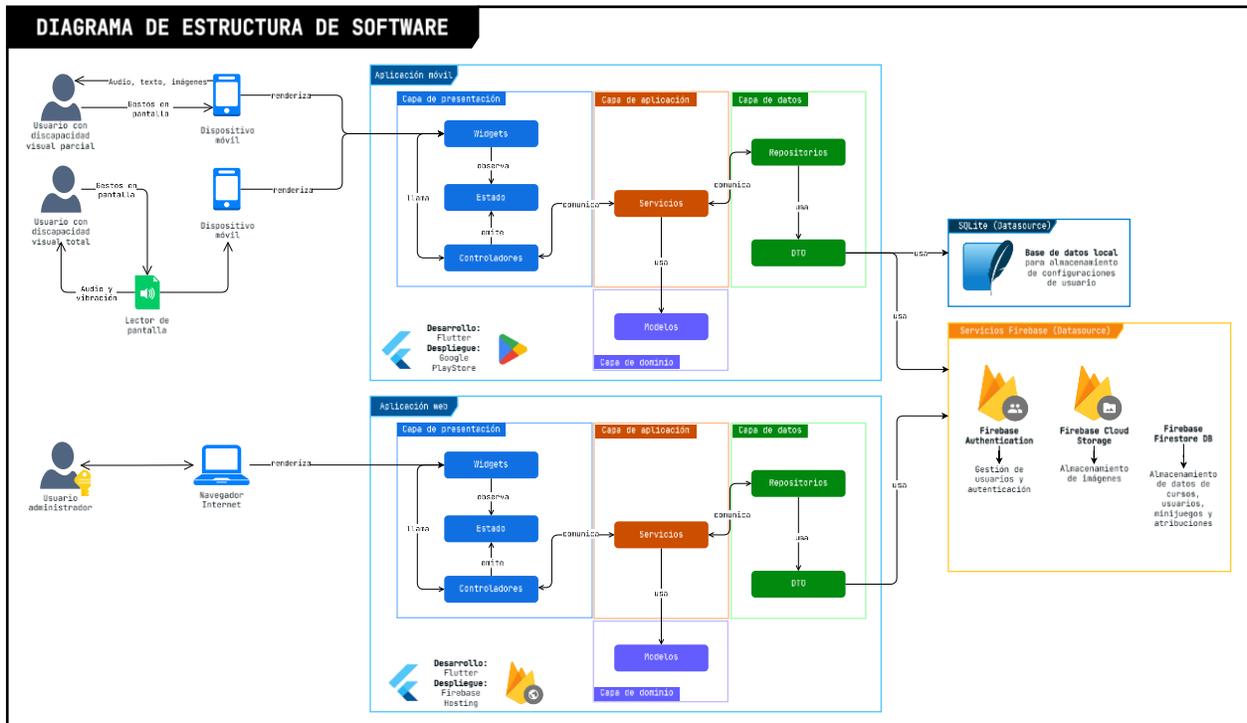
Nota: Captura tomada de Play Store

### 2.2.5. Fase de muerte

En esta fase se realizó la entrega final al cliente, quien mostró satisfacción con la estrategia computacional desarrollada (Ver Anexo I. Carta de entrega del desarrollo tecnológico). Esto indica que se cumplió con las expectativas del cliente, desarrollando todas las funcionalidades requeridas y cumpliendo con los plazos establecidos. Finalmente, se realiza el proceso de documentación del sistema desarrollado, mediante un archivo README.md en el repositorio del proyecto (Ver Anexo J. Otros enlaces). El objetivo del archivo README.md es brindar un contexto del proyecto a otros desarrolladores que deseen explorar el proyecto. Finalmente, se realiza el diagrama de estructura de software que muestra el funcionamiento global del sistema completo, incluyendo la aplicación web y la aplicación móvil, así como los componentes involucrados, el flujo de datos y el modo de despliegue de cada aplicación.

**Figura 95**

*Diagrama de estructura de software*



### 2.2.6. Discusión

El proceso de desarrollo se llevó a cabo de manera exitosa, cumpliendo con todos los requerimientos del cliente en los plazos acordados. Se destaca que el uso de la metodología híbrida propuesta (solo eXtreme Programming y SCRUM) se ajustó satisfactoriamente al presente proyecto con un solo desarrollador, siguiendo un proceso iterativo e incremental con etapas concretas (análisis, diseño, codificación y pruebas). Además, definir y seguir una arquitectura específica de inicio a fin hizo que el proyecto cuente con una estructura ordenada, de manera que agregar o modificar funcionalidades sea sencillo. Particularmente, la arquitectura Riverpod se ajustó perfectamente al proyecto, permitiendo modularizarlo, facilitando la implementación de pruebas unitarias y de integración, entre otros procesos.

Utilizar un control de versiones como Git fue crucial a la hora de gestionar los cambios ocurridos en el proyecto, pues en varias ocasiones fue necesario recurrir a esta herramienta para restaurar el

proyecto a su versión más estable. Adicionalmente, se recomienda el uso de estándares como GitFlow para la nomenclatura de las ramas a utilizar en el proyecto.

Por otro lado, la elección de tecnologías fue acertada, ofreciendo una experiencia de desarrollo eficiente y eficaz. Flutter como Framework para desarrollo multiplataforma y FlutterFlow como herramienta para diseño de interfaces permitieron agilizar los procesos de diseño y codificación, pues es posible integrar ambas tecnologías al utilizar el mismo lenguaje de programación (Dart). Además, para estos dos procesos fue de gran utilidad la elaboración de diagramas UML, EERR, y de comunicación de componentes; especialmente al diseñar las pruebas unitarias y de integración.

La comunicación constante con el cliente y la especialista en baja visión de FUNDONAR permitió que la aplicación cumpla con las expectativas de ambos involucrados, obteniendo retroalimentación y sugerencias que ayudaron a perfilar detalles importantes de la aplicación sin alterar los requerimientos definidos al inicio del proceso de desarrollo. El recibimiento de la aplicación por parte de la especialista en baja visión de la fundación fue positivo y expresó que la aplicación será utilizada constantemente sus pacientes al incluir determinados contenidos de la aplicación dentro de sus historias clínicas. Por otro lado, se cumplió con las expectativas del cliente, quien hará uso de la aplicación de administrador para generar nuevos contenidos constantemente para la aplicación.

En cuanto al proceso de despliegue en la fase de producción, fue necesario aprobar diversas pruebas por parte de Google antes de lanzar la aplicación al mercado. Esto incluyó una prueba cerrada con 20 testers, lo que retrasó el lanzamiento 14 días. Sin embargo, estas pruebas presentaron resultados satisfactorios en cuanto a rendimiento y pruebas de accesibilidad, entregando cifras y métricas concretas que aportan gran valor al evaluar la calidad de la aplicación desde una perspectiva técnica.

Finalmente, en la fase de muerte se presenta la documentación del proyecto en el repositorio de GitHub, el cual quedará abierto para servir como referencia de implementación de accesibilidad en proyectos de desarrollo móvil en general. De esta manera se hace un aporte al mundo del desarrollo de software y se apoya el desarrollo accesible.

## 2.3. Objetivo específico 3

### 2.3.1. Desarrollo

Una vez finalizado el proceso de desarrollo, se inicia el desarrollo del tercer objetivo específico del proyecto, enfocado en la evaluación de la calidad de la estrategia computacional desarrollada. Esta evaluación se realizó teniendo en cuenta las métricas accesibilidad, usabilidad, adecuación funcional y mantenibilidad. La primera métrica fue evaluada durante el proceso de desarrollo de la aplicación, como se muestra en la Tabla 20.

**Tabla 21.** , se aplicó la accesibilidad mediante la implementación de normas concretas tomadas de la WCAG 2.1. Como parte del proceso de despliegue, la aplicación fue sometida a una evaluación de accesibilidad, donde se encontró únicamente un error de bajo impacto (Ver Figura 93.

). Adicionalmente, todas las pruebas de integración incluyen las pruebas de accesibilidad de Flutter, para las plataformas Android e IOS. Por lo tanto, la métrica de accesibilidad fue alcanzada de manera satisfactoria. La siguiente tabla muestra un resumen de las normas aplicadas en el proyecto

**Tabla 62**

*Implementación de normas de accesibilidad en el proceso de desarrollo*

<b>Propiedades de accesibilidad</b>	<b>de</b>	<b>Normas asociadas</b>	<b>Implementada en Iteración #</b>
IRNF-001	Alternativas de texto	1.1.1. Contenido no textual (AA)	3
IRNF-002	Adaptable	1.3.2. Secuencia Significativa (A) 1.3.3. Características sensoriales (A)	3
IRNF-003	Distinguible	1.4.1. Uso del color (A) 1.4.3. Contraste (AA) 1.4.4. Redimensionar texto (AA)	2

		1.4.6. Contraste (AAA)	
		1.4.8. Presentación visual (AAA)	
		1.4.9. Imágenes de texto (AAA)	
		1.4.10. Reflujo (AA)	
		1.4.11. Contraste no textual (AA)	
IRNF-004 Navegable		2.4.6. Encabezados y etiquetas (AA)	4
		2.4.10. Encabezados de sección (AAA)	
IRNF-005 Modalidad de entrada		2.5.3. Etiqueta en nombre (A)	4
		2.5.5. Tamaño de objetivo (AAA)	
IRNF-006 Predecible		3.2.3. Navegación consistente (AA)	4
		3.24. Identificación consistente (AA)	
IRNF-007 Compatible		4.1.2. Nombre, Rol, Valor (A)	4
		4.1.3. Mensajes de Estado (AA)	

Luego, se procedió a evaluar la segunda métrica establecida, la cual es usabilidad. Esta evaluación se realizó en la población de FUNDONAR, seleccionando a diez pacientes, quienes atendieron la encuesta luego de realizar su consulta regular en la fundación. Para probar la aplicación, se elaboró un curso de prueba con seis videos y dos minijuegos, referentes a la temática de “El Reino Animal”. Se eligió esta temática según la caracterización realizada por Salas en la misma población.

## Figura 96

*Contenidos de prueba creados para probar la aplicación*



Durante las pruebas, los pacientes seleccionados utilizaron la aplicación durante aproximadamente cinco minutos, completando satisfactoriamente los contenidos de prueba. Para este proceso, se hizo uso de herramientas como lentes graduables, magnificadores y linternas, según el diagnóstico de cada paciente. Adicionalmente, algunos pacientes autorizaron la recolección de evidencia audiovisual de este proceso.

### Figura 97

*Evidencia de pruebas de la aplicación en la población*



Al completar los contenidos, se aplicó el cuestionario SUS, propuesto por (Brooke, 2013) (Ver Anexo E. Instrumento SUS para evaluación de usabilidad), el cual contiene dos tipos de preguntas. El primero busca identificar indicios positivos de la usabilidad de la aplicación, mientras que el segundo busca identificar lo contrario.

**Tabla 63**

*Clasificación de preguntas en instrumento de recolección*

<b>Preguntas tipo 1</b>	<b>Preguntas tipo 2</b>
Creo que me gustaría utilizar esta aplicación con frecuencia	Encontré la aplicación innecesariamente compleja
Pensé que la aplicación es fácil de usar	Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación
Encontré que las diversas funciones de esta aplicación están bien integradas	Pensé que había demasiada inconsistencia en esta aplicación
Me imagino que la mayoría de la gente aprendería a utilizar esta aplicación rápidamente	Encontré la aplicación muy complicada de usar
Me sentí muy segur@ usando la aplicación	Necesitaba aprender muchas cosas antes de empezar con esta aplicación

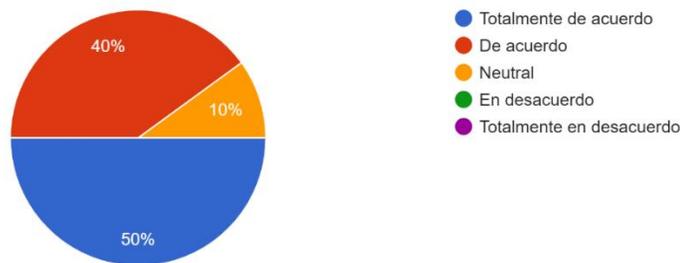
A partir de estas preguntas, se logró obtener datos valiosos que arrojaron luz sobre la experiencia de los usuarios con la aplicación en proceso de desarrollo. Este análisis permitió entender cómo los participantes interactuaron con la interfaz y qué percepciones tuvieron al respecto.

Como se observa en la siguiente figura, se observa un porcentaje significativo de usuarios interesados en seguir utilizando la aplicación una vez esté lista para su lanzamiento. Esto indica una receptividad positiva por parte de los pacientes de FUNDONAR hacia la aplicación, lo cual puede ser un factor determinante para su éxito en el mercado. Esto respalda la importancia de continuar refinando la aplicación para asegurar que cumpla con las expectativas y necesidades de los usuarios.

**Figura 98**

*Interés de los participantes por seguir utilizando la aplicación*

Creo que me gustaría utilizar esta aplicación con frecuencia  
10 responses



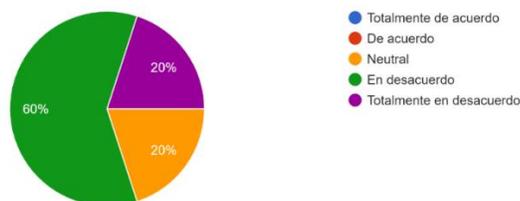
Si bien el uso de la aplicación por parte de los pacientes de la fundación será dirigido por la especialista en baja visión de FUNDONAR, la mayoría de los pacientes encuestados expresó que no necesitaría de apoyo técnico para utilizar la aplicación, como se indica en la Figura 99.

. Esto es un indicio positivo de usabilidad y abre la posibilidad de que la aplicación sea utilizada también por fuera de las instalaciones de la fundación.

### Figura 99

*Apoyo técnico para usuarios de la aplicación*

Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación  
10 responses



De la misma manera, la Figura 100. indica que los encuestados consideran que la mayoría de gente aprendería a utilizar la aplicación rápidamente, lo cual incluye al resto de pacientes de FUNDONAR que no participaron en la evaluación de usabilidad de la aplicación. Por lo tanto, es factible que la aplicación sea utilizada no solo por la población de FUNDONAR, sino por la población con discapacidad visual en general.

**Figura 100**

*Aprender a utilizar la aplicación rápidamente.*



De la misma manera, el resto de las preguntas indicaron que en general, la aplicación es fácil de utilizar para los pacientes seleccionados. Una manera de observar esto es mediante el puntaje de SUS, para calcular este puntaje, se asignaron valores numéricos para ambos tipos de pregunta, teniendo en cuenta que entre mayor sea el valor, representa un mayor grado de usabilidad. Ambos tipos de pregunta se presentaron de manera intercalada para una mayor uniformidad y un menor riesgo de sesgo en el instrumento de recolección.

**Tabla 64**

*Asignación de valores de respuesta según tipo de pregunta*

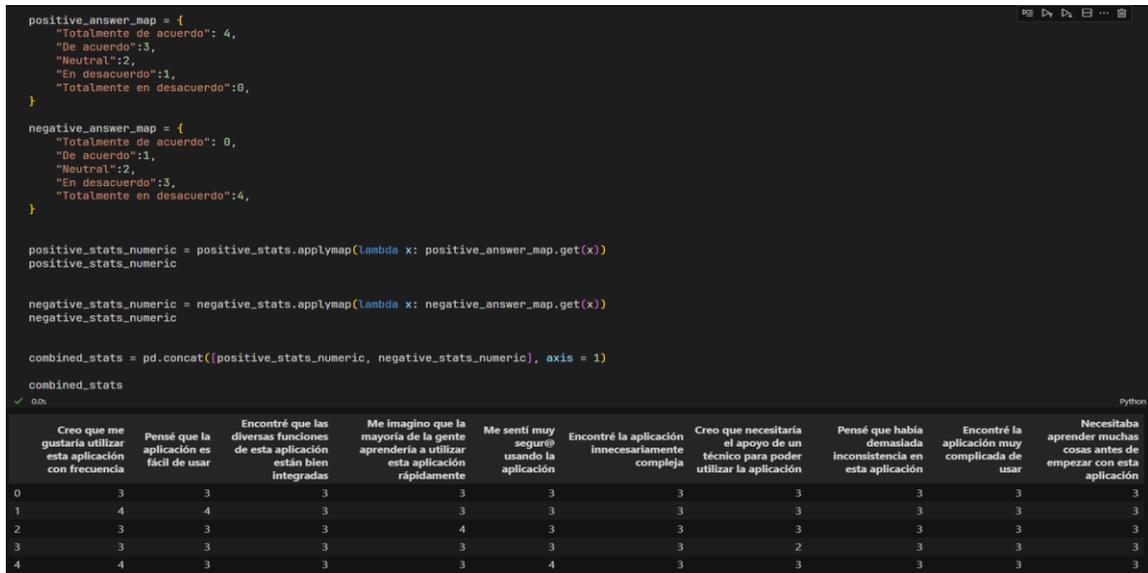
Respuesta	Valor en pregunta tipo 1	Valor en pregunta tipo 2
<b>Totalmente de acuerdo</b>	4	0
<b>De acuerdo</b>	3	1
<b>Neutral</b>	2	2
<b>En desacuerdo</b>	1	3
<b>Totalmente en desacuerdo</b>	0	4

Posteriormente se realiza un proceso de análisis de datos en Python, haciendo uso de la biblioteca numpy para el manejo del archivo csv que contenía los resultados del instrumento de

recolección (Ver Anexo F. Análisis de datos - Cuestionario SUS). Como resultado, se obtuvo un DataFrame con los valores numéricos correspondientes a cada pregunta

**Figura 101**

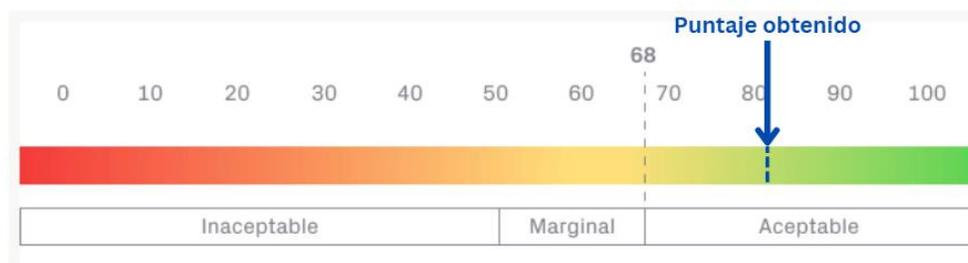
*Resultados de instrumento de recolección después del mapeo de datos.*



Finalmente, se efectuó una sumatoria de todos los valores, y se lo dividió entre la mayor cantidad de puntos posibles, obteniendo un puntaje final de 80.5. Este resultado indica una percepción positiva respecto a la usabilidad de la aplicación por parte de los pacientes participantes. Según (Sauro, 2011), el puntaje promedio es 68 y todo puntaje mayor a este valor es considerado aceptable.

**Figura 102**

*Representación de los resultados de cuestionario SUS*



Nota: Gráfico tomado de (Busquets, s/f)

Luego, para evaluar dos métricas restantes (adecuación funcional y mantenibilidad), se sometió a la estrategia computacional desarrollada a una evaluación por parte de expertos en el área de desarrollo de software, utilizando un instrumento elaborado con base en el cuestionario SUS (Ver Anexo G. Instrumento de evaluación - Proceso de desarrollo de software).

**Tabla 65**

*Preguntas de cuestionario para evaluación de expertos*

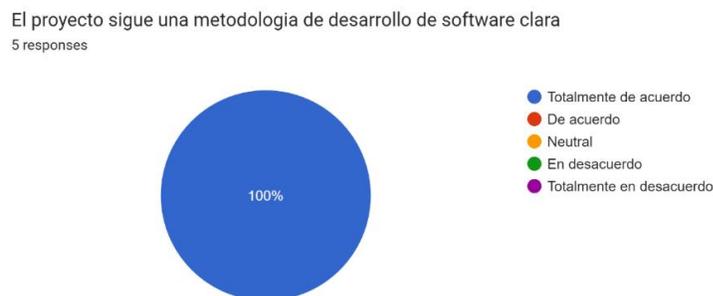
<b>Preguntas tipo 1</b>	<b>Preguntas tipo 2</b>
El proyecto sigue una metodología de desarrollo de software clara	Agregar nuevas funcionalidades al proyecto resultaría complejo y costoso
El proyecto cuenta con una estructura de directorios adecuada	El código fuente presenta repetición de código innecesaria
Las herramientas de desarrollo utilizadas fueron adecuadas para el proyecto	La aplicación experimenta problemas de rendimiento o tiempos de respuesta lentos en ciertas situaciones.
Las historias de usuario del proyecto cumplen con el criterio SMART (Specific, Measurable, Achievable, Relevant, Time-boundeds)	La aplicación tarda mucho tiempo en iniciar desde cero
La implementación programática de normas de accesibilidad es adecuada	La elección del tipo de bases de datos es injustificada

Los métodos de testing utilizados son adecuados	La aplicación presenta problemas de seguridad
Considera útil que este tipo de aplicaciones sean desarrolladas para la población con discapacidad visual	La aplicación posee una interfaz gráfica poco intuitiva
En su experiencia como desarrollador de software, considera necesario el desarrollo de este tipo de aplicaciones	La metodología de desarrollo es aplicada de manera inconsistente

Al aplicar este cuestionario se obtuvo resultados positivos en general. Particularmente, como lo indica la Figura 103, todos los expertos consideraron que el proyecto siguió una metodología clara en el proceso de desarrollo de software. Esto indica que desarrollo fue ordenado y coherente, como lo indican otras preguntas similares, relacionadas a la estructura de directorios y la consistencia de a metodología.

### Figura 103

#### *Seguimiento de metodología en proyecto*



También se evaluaron aspectos técnicos, donde los expertos indicaron que la elección de tecnologías, tanto de Frontend, como de Backend y Bases de datos son adecuadas, como se muestra en la Figura 104.

### Figura 104

## Herramientas utilizadas en el proyecto



No obstante, como se muestra en la Figura 105.

, al preguntar sobre la mantenibilidad del proyecto, algunos de los encuestados se mostraron escépticos con respecto al costo y dificultad que tendría agregar nuevas funcionalidades al proyecto. Al recibir la retroalimentación correspondiente, los expertos expresaron que esta valoración se debía principalmente a que el proyecto ya contaba con un tamaño considerable, por lo que contextualizar a un nuevo desarrollador para agregar nuevas funcionalidades puede ser demorado al principio.

### Figura 105

#### Costos y complejidad al agregar nuevas funcionalidades

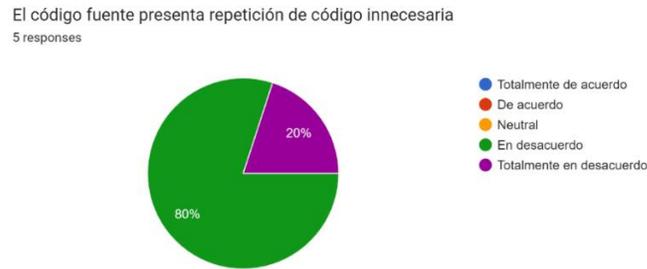


Por otro lado, los expertos consideraron que la estructura de directorios utilizada fue adecuada, además de que el código escrito no presenta repetición innecesaria, como se indica en la Figura 106. y Figura 107. . Esto indica que el proyecto se desarrolló de manera ordenada y siguiendo buenas prácticas de

programación. El modularidad y la claridad en la organización del código facilitaron su comprensión por parte de los expertos.

### Figura 106

#### *Repetición de código en el proyecto*



### Figura 107

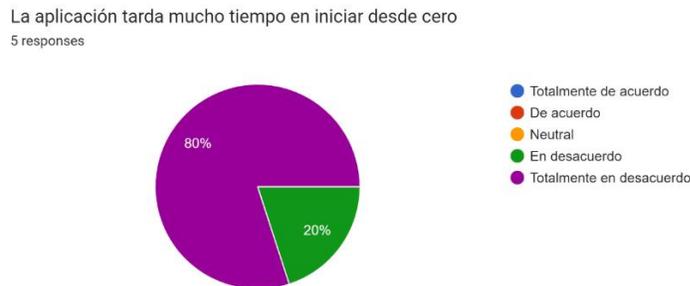
#### *Estructura de directorios del proyecto*



De la misma manera, se obtuvieron resultados positivos con respecto a los tiempos de espera en la aplicación, lo cual demuestra que se logró una óptima eficiencia en el rendimiento del sistema. Los tiempos de carga de los diferentes componentes de la aplicación se mantuvieron dentro de los límites aceptables, lo que garantiza una experiencia fluida para los usuarios finales. Además, este mismo aspecto fue evaluado al realizar el proceso de despliegue, como se muestra en la Figura 92.

### Figura 108

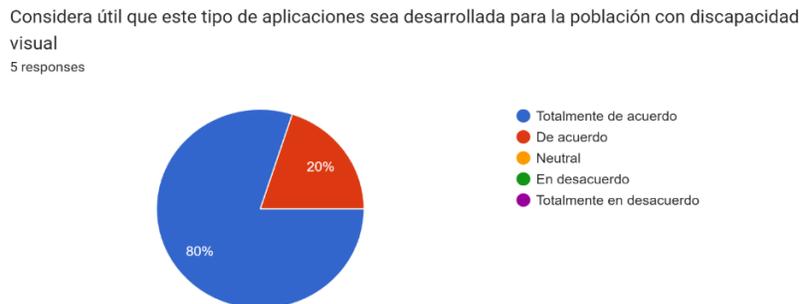
#### Tiempo de inicio en la aplicación



Como se aprecia en la Figura 108, los expertos encuestados consideran útil que este tipo de aplicaciones sea desarrollado para la población con discapacidad visual. Esto indica que existe una necesidad real de este tipo de soluciones tecnológica, lo que respalda la relevancia y pertinencia del presente proyecto. Este hallazgo refuerza la importancia de seguir desarrollando aplicaciones y recursos accesibles que promuevan la inclusión y la igualdad de oportunidades para personas con discapacidad visual.

### Figura 109

#### Utilidad del desarrollo de aplicaciones para personas con discapacidad visual



Finalmente, al realizar el respectivo proceso de análisis para los resultados de la evaluación de expertos (Ver Anexo H. Análisis de datos – Evaluación de expertos), se obtuvo un puntaje de 86.25, lo cual indica una recepción positiva en general por parte de los expertos. Adicionalmente, se

agregó una pregunta de libre respuesta para agregar comentarios adicionales, donde se destacó la importancia de haber desplegado la aplicación para el mercado, aportando como referente a futuras soluciones e investigaciones, además de resaltar la automatización de pruebas unitarias y de integración. Entre los puntos a mejorar, se recomendó una división más granular de tareas e inclusión de patrones de acceso en la base de datos.

### ***2.3.2. Discusión***

El tercer objetivo específico se centró en evaluar la calidad de la estrategia computacional desarrollada mediante cuatro métricas fundamentales: accesibilidad, usabilidad, adecuación funcional y mantenibilidad.

La evaluación de accesibilidad se realizó tanto durante el proceso de desarrollo como en etapas posteriores, implementando normas específicas derivadas de la WCAG 2.1, las cuales fueron aplicadas de manera gradual durante diferentes iteraciones del desarrollo. Se destaca la importancia de las pruebas de accesibilidad realizadas por Google, donde se obtuvo resultados muy positivos. Por otro lado, al utilizar Flutter como Framework para el desarrollo móvil, muchas de las normas seleccionadas fueron cumplidas por defecto, lo cual facilitó el cumplimiento de todos los requerimientos relacionados con accesibilidad.

Por otro lado, la evaluación de usabilidad se realizó con una muestra de diez pacientes de la fundación FUNDONAR, quienes probaron la aplicación por un periodo de aproximadamente cinco minutos, completando contenidos digitales y minijuegos elaborados para dicha prueba. La usabilidad fue estimada utilizando el cuestionario SUS, obteniendo un puntaje final de 80.5, lo cual indicó una percepción positiva por parte de los usuarios. Este resultado supera el puntaje promedio establecido como aceptable según los estudios consultados.

Finalmente, la adecuación funcional y mantenibilidad fueron evaluadas a luz de expertos en el área de desarrollo de software. Se diseñó un cuestionario que abarcaba aspectos relacionados a estas dos métricas mencionadas. Tras el análisis de los resultados, se obtuvo un puntaje de 93.75, reflejando una percepción positiva y favorable por parte de los expertos. Este paso trajo consigo

comentarios y retroalimentación de gran valor, que sirven como referencia para lograr una mejora continua en la aplicación. Como aspectos positivos se destacó la implementación acertada de una metodología de desarrollo ágil, incluyendo aspectos como pruebas unitarias y de integración a lo largo de las iteraciones del proceso. Se elogió la estructura de directorios y el alto rendimiento de la aplicación, teniendo en cuenta buenas prácticas de programación en la codificación. Por otro lado, se destacaron aspectos a mejorar, principalmente sobre experiencia de usuario en algunos minijuegos como el ahorcado. Se identificó un fallo en el juego de la trivia donde el usuario podía obtener puntuaciones perfectas sin acertar ninguna pregunta al pulsar repetidamente ambas opciones de respuesta de manera simultánea. Este error fue corregido y se lanzó una actualización con los cambios correspondientes.

### **3. Conclusiones**

Al realizar el proceso de caracterización de la población de FUNDONAR, se concluye que las principales barreras que enfrentan al acceder a contenidos digitales se relacionan con factores como el tamaño de texto, combinación de colores y etiquetas de elementos. Además, se encontró que existe diversidad en cuanto a las preferencias de cada paciente con respecto a estos factores. Por ende, se concluye que la estrategia computacional a desarrollar deberá permitir al usuario modificar estas variables para ajustar los contenidos a su preferencia.

La implementación de la metodología de desarrollo híbrida Solo eXtreme Programming se ajustó satisfactoriamente a las necesidades del proyecto. Aplicar esta metodología desde el inicio hizo que el proyecto tenga una estructura ordenada, facilitando muchas tareas relacionadas al proceso de desarrollo, tales como la elaboración de diagramas y escenarios de prueba. Igualmente, las tecnologías seleccionadas para el desarrollo fueron adecuadas. Concretamente, utilizar Flutter como Framework de desarrollo multiplataforma aceleró la construcción de ambas aplicaciones, pues permitió compartir código entre la aplicación móvil y la aplicación web, sin necesidad de volver a implementar las mismas funcionalidades en otro lenguaje de programación distinto. Así mismo, utilizar un servicio como Firebase resultó más eficiente que implementar un Backend desde cero.

Finalmente, al evaluar la accesibilidad, usabilidad, adecuación funcional y mantenibilidad de la aplicación, se garantizó que se cumple con un nivel de calidad aceptable, obteniendo resultados positivos en cada una de las pruebas realizadas para medir estas métricas. En el caso de la accesibilidad, se obtuvo resultados satisfactorios tanto en las pruebas de integración como en las pruebas de Google al lanzar la aplicación al mercado. Esto indica que la implementación de las normativas de accesibilidad fue efectiva. Esto se ve reflejado en la evaluación de usabilidad de la aplicación, donde los pacientes de FUNDONAR seleccionados utilizaron la aplicación de manera sencilla y agradable. Por último, los aspectos técnicos positivos reflejados en la evaluación de calidad de software por parte de expertos en el área sugieren que el proceso de desarrollo fue realizado de manera exitosa.

#### **4. Recomendaciones**

Para futuros proyectos de desarrollo tecnológico, se recomienda tener en cuenta la accesibilidad desde el inicio del proceso de desarrollo y sustentarla por medio de alguna normativa vigente. En concreto, se recomienda seguir la normativa internacional WCAG, debido a su buena documentación que incluyen casos de aprobación y fallo en cada una de las normas expuestas, aplicadas a distintas plataformas como web, móvil, o multimedia. Se debe tener en cuenta que la accesibilidad abarca un gran número de discapacidades distintas a la discapacidad visual, por ende, es necesario estudiar y enfocar nuevos proyectos hacia estas otras discapacidades para así lograr una inclusión social más amplia en el mundo digital.

Por otro lado, para equipos de un solo desarrollador se recomienda aplicar metodologías de desarrollo ágiles como las utilizadas en este proyecto. Cada proyecto es distinto, y dependiendo del contexto y sus necesidades, se recomienda combinar metodologías para obtener un mejor resultado, teniendo en cuenta factores como tamaño del equipo, complejidad del proyecto y plazos de entrega. Es importante que todos estos procesos sean definidos antes de dar inicio al proceso de desarrollo, de esta manera se tiene definido qué pasos seguir y dónde comenzar en cada una de las etapas del proceso.

Finalmente, si se realizan desarrollos móviles para la plataforma Android, se debe tener en cuenta que los procesos de despliegue requieren de numerosas pruebas y procesos que retrasarán el lanzamiento del software. Por lo tanto, se recomienda contar con al menos tres semanas en el cronograma destinadas a este fin. Durante ese tiempo, será necesario realizar una prueba cerrada con 20 testers durante 14 días, por lo cual es recomendado contar con este número de personas previamente para agilizar esta prueba.

## Referencias bibliográficas

*Accessible 3D Audio Maze Game* (1.1). (2022). LightOnDevs.  
<https://play.google.com/store/apps/details?id=com.lightgames.maze>

Águila, S. C. A. Del, Incio, F. A. R., & Moreno, Á. E. P. (2022). Comparative study of the digital government websites of the mayors of Peru and Colombia. *International journal of health sciences*, 1564–1577. <https://doi.org/10.53730/ijhs.v6ns3.5696>

Akhtar, A., Bakhtawar, B., & Akhtar, S. (2022). EXTREME PROGRAMMING VS SCRUM: A COMPARISON OF AGILE MODELS. *International Journal of Technology, Innovation and Management (IJTIM)*, 2. <https://doi.org/10.54489/ijtim.v2i1.77>

ATLASSIAN. (s/f). *Gitflow Workflow*. Recuperado el 3 de mayo de 2023, de <https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow>

Bakhtawar, B., Aziz, N., Khan, S., Fahiem, M. A., Aftab, S., Ahmad, M., Almotilag, A., & Sabri Elmitwally, N. (2021). Latest Transformations of XP Process Model: A Systematic Literature Review Agile software development models View project Green ICT + Cloud Computing = Digital Ecosystem Thinking? View project Latest Transformations of XP Process Model: A Systematic Literature Review. *IJCSNS International Journal of Computer Science and Network Security*, 21(6), 143. <https://doi.org/10.22937/IJCSNS.2021.21.6.19>

*Ballyland Code*. (s/f). Sonokids. Recuperado el 3 de mayo de 2023, de <https://www.sonokids.org/ballyland-early-learning/ballyland-game-apps/ballyland-code-1-say-hello/>

*BARD Mobile* (1.5.0). (2023). Library Of Congress.  
[https://play.google.com/store/apps/details?id=gov.loc.nls.dtb&hl=es\\_CO&gl=US](https://play.google.com/store/apps/details?id=gov.loc.nls.dtb&hl=es_CO&gl=US)

Beck, Kent. (2000). *Extreme programming eXplained : embrace change*. Addison-Wesley.

Benefield, G., Larman, C., & Vodde, B. (2010). *THE SCRUM PRIMER* Pete Deemer Scrum Training Institute (ScrumTI.com).

Bizzotto, A. (2023, septiembre 1). *Flutter App Architecture with Riverpod: An Introduction*.  
<https://codewithandrea.com/articles/flutter-app-architecture-riverpod-introduction/>

*Blind Explorer*. (2022). GEKO NAVSAT.  
<https://play.google.com/store/apps/details?id=com.startapp.smtrail>

Brooke, J. (2013). *SUS: A Retrospective* (Vol. 8).

Busquets, C. (s/f). *Medir la usabilidad con el Sistema de Escalas de Usabilidad (SUS)*.

Comunicaciones INCI. (2020). *INCI conoce de manera oficial la cifra del Censo de Discapacidad*.  
<https://www.inci.gov.co/blog/inci-conoce-de-manera-oficial-la-cifra-del-censo-de-discapacidad>

Cronin, G. (2001). *eXtreme Solo eXtreme Solo A Case Study in Single Developer eXtreme Programming*.

Deacon, J. (2009). *Model-View-Controller (MVC) Architecture*.  
<http://www.jdl.co.uk><http://www.johndeacon.net>

Enrique Cortés Fandiño, J. (2022). *Web accessibility in the colombian public entities Proposal of an evaluation model applied*.

Escandell Bermúdez, M. O., Fortea Sevilla, M. del S., & Castro Sánchez, J. J. (2016). LA BRECHA DIGITAL EN LAS PERSONAS CON DISCAPACIDAD VISUAL. *International Journal of Developmental and Educational Psychology. Revista INFAD de Psicología.*, 1(1), 489.  
<https://doi.org/10.17060/ijodaep.2014.n1.v1.396>

- Gaur, S. (2019). *Voice-Guided-Navigation-for-visually-impaired*.  
<https://github.com/ShivamGaurUQ/Voice-Guided-Navigation-for-visually-impaired>
- Google. (2023). *Requisitos de pruebas de aplicaciones para las nuevas cuentas personales de desarrollador*. <https://support.google.com/googleplay/android-developer/answer/14151465>
- Henry, S. (2022). *Introduction to Web Accessibility*.
- Jovanović, I. (2009). *Software Testing Methods and Techniques*. 30–41.
- Kamaghe, J., Luhanga, E., & Kisangiri, M. (2020). The challenges of adopting M-learning assistive technologies for visually impaired learners in higher learning institution in Tanzania. *International Journal of Emerging Technologies in Learning*, 15(1), 140–151.  
<https://doi.org/10.3991/ijet.v15i01.11453>
- Martin, R. C., Grenning, J., & Brown, S. (s/f). *CLEAN ARCHITECTURE*.
- Ministerio de Tecnologías de la Información y las Comunicaciones (MINTIC). (s/f). Recuperado el 14 de abril de 2023, de <https://gobiernodigital.mintic.gov.co/portal/Secciones/Preguntas-Frecuentes/76021:6-Accesibilidad>
- Niño-Vega, J. A., Giraldo-Cardona, M. T., & Fernández-Morales, F. H. (2022). Analysis of web accessibility to Colombian universities under the guidelines proposed by WCAG 2.1. *Gaceta Medica de Caracas*, 130, S618–S625. <https://doi.org/10.47307/GMC.2022.130.s3.15>
- Paniagua L., A., Bedoya R., D., & Mera, C. (2020). Un método para la evaluación de la accesibilidad y la usabilidad en aplicaciones móviles. *TecnoLógicas*, 23(48), 99–117.  
<https://doi.org/10.22430/22565337.1553>
- Salas, M., Erazo, L., & Luna, L. (2022). *Diseño de material didáctico para la población con discapacidad visual en la ciudad de Pasto*.

- Sauro, J. (2011, febrero 3). *Measuring Usability with the System Usability Scale (SUS)*.  
<https://measuringu.com/sus/>
- Shera, A., Iqbal, M. W., Shahzad, S. K., Gul, M., Mian, N. A., Naqvi, M. R., & Khan, B. A. (2021). Blind and visually impaired user interface to solve accessibility problems. *Intelligent Automation and Soft Computing*, 30(1), 285–301. <https://doi.org/10.32604/iasc.2021.018009>
- Velázquez, C., Cardona, P., Muñoz, J., Álvarez, F., & Ochoa, A. (2020). *TECNOLOGÍA INNOVACIÓN Y PRÁCTICA EDUCATIVA* (Editorial CIATA). <http://www.ciata.org>
- Villalobos Salcedo, J. Alberto. (2008). *Introducción a las estructuras de datos*. Pearson Educación.
- Villamizar Suaza, K. (2013). *Definición de equivalencias entre historias de usuario y especificaciones en UN-LENCEP para el desarrollo ágil de software*.
- Web Content Accessibility Guidelines (WCAG) 2.1*. (2018, mayo 5).  
<https://www.w3.org/TR/WCAG21/>
- World Health Organization: WHO*. (2022). <https://www.who.int/es/news-room/fact-sheets/detail/blindness-and-visual-impairment>
- World Web Consortium (W3C)*. (s/f). Recuperado el 14 de abril de 2023, de <https://www.w3.org/>
- Zambrano, L., Zambrano, D., Minaya, R., Torres, S., & Almeida, J. (2020). *SUPLEMENTO CICA, multidisciplinario* N:011-2021.  
<https://suplementocica.uleam.edu.ec/index.php/SuplementoCICA>

## Anexos

### Anexo A. Carta que justifica el desarrollo tecnológico



Universidad  
**Mariana**  
NIT. 800.092.198-5

San Juan de Pasto, 31 de marzo de 2023

Magíster  
**MAGDA MIREYA SALAZAR SUÁREZ**  
Programa de Ingeniería de Sistemas  
Directora

Apreciada Directora, saludo de paz y bien

El abajo firmante y Director del Programa de Mercadeo hago una solicitud formal para que el estudiante del programa de Sistemas Carlos Alberto Botina Carpio con C.C # 1004234727, pueda formalizar su trabajo de grado diseñando una aplicación de software para población con discapacidad visual, el cual se enmarcaría dentro de la investigación profesoral que están llevando a cabo Lorena Tatiana Luna Tobar y Edgar Mauricio Salas Leiva docentes del programa de Mercadeo que tiene como objetivo General "Diseñar un material didáctico para la población con discapacidad visual de la ciudad de Pasto".

Se hace esta solicitud debido a que el estudiante presenta interés en el tema y el docente Fabián Parra apoyará el proceso de diseño del material didáctico en cuestión.

Agradezco su atención,

  
Andrés Fernando Andrade Santander  
Programa de Mercadeo  
Director  
[aandrade@umariana.edu.co](mailto:aandrade@umariana.edu.co)  
3136845920

"Consolidación de la Excelencia Educativa para la Transformación Social"

Calle 18 No. 34 -104 - PBX (TEL.) 7244460 - SAN JUAN DE PASTO - COLOMBIA  
[www.umariana.edu.co](http://www.umariana.edu.co) - [informacion@umariana.edu.co](mailto:informacion@umariana.edu.co) VIGRADA MINEDUCACIÓN

## **Anexo B. Instrumento de recolección para prueba piloto**

El propósito de esta encuesta es identificar las necesidades y preferencias la población con discapacidad visual de la fundación FUNDONAR. La información recopilada en esta encuesta será utilizada con fines académicos en la realización de un trabajo de grado en la carrera de Ingeniería de Sistemas de la Universidad Mariana.

### **Cuestionario**

1. Nombres y Apellidos: \_\_\_\_\_
2. Cédula: \_\_\_\_\_
3. Estrato:
  - a) 1
  - b) 2
  - c) 3
  - d) 4
  - e) 5
  - f) 6
4. Género
  - a) Hombre
  - b) Mujer
  - c) No Binario
5. Edad
  - a) 1
  - b) 2
  - c) 3
  - d) 4
  - e) 5
  - f) 6
  - g) 7
  - h) 8
  - i) 9

- j) 10
- k) 11
- l) 12
- m) 13
- n) 14
- o) 15
- p) 16
- q) 17
- r) 18
- s) 19
- t) 20

6. Diagnóstico

- a) Ceguera legal
- b) Ceguera total
- c) Baja visión
- d) Otro: \_\_\_\_\_

7. Nivel de escolaridad

- a) Primaria incompleta
- b) Primaria completa
- c) Secundaria incompleta
- d) Secundaria completa
- e) Pregrado incompleto
- f) Pregrado completo
- g) Posgrado incompleto
- h) Posgrado completo

8. Ocupación: \_\_\_\_\_

9. ¿Qué dispositivos utiliza para acceder a aplicaciones y contenidos digitales? (puede seleccionar varios)
- a) Smartphone
  - b) Tablet
  - c) Computador de escritorio

- d) Computador portátil
  - e) Dumbphone
  - f) Otro: \_\_\_\_\_
10. Seleccione las tecnologías de asistencia que utiliza para acceder a contenidos digitales (puede seleccionar varios)
- a) Lectores de Pantalla
  - b) Dispositivos de Braile
  - c) No utilizo tecnología de asistencia
  - d) Otro: \_\_\_\_\_
11. ¿Qué tipo de aplicaciones suele utilizar con en sus dispositivos? (puede seleccionar varios)
- a) Redes sociales
  - b) Aplicaciones de mensajería
  - c) Navegación por Internet
  - d) Aplicaciones de lectura de libros o noticias
  - e) Otro: \_\_\_\_\_
12. Me gusta interactuar con mi dispositivo mediante gestos táctiles
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
13. Me gusta recibir respuestas de mi dispositivo mediante audio (efectos de sonido, voces, música)
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
14. Me gusta recibir respuestas de mi dispositivo mediante vibración
- a) Nunca
  - b) Casi Nunca

- c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
15. Si usted utiliza lector de pantalla, ¿ha tenido situaciones donde este sea incompatible con alguna aplicación?
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
16. Si ha utilizado internet, ¿ha presentado dificultad a la hora de navegar por sitios web?
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
17. ¿Es importante para usted que las imágenes contengan texto descriptivo?
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
18. ¿Con qué frecuencia encuentra imágenes que no tienen texto descriptivo?
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
19. ¿Ha tenido situaciones donde los colores en pantalla hagan difícil la navegación en el dispositivo?
- a) Nunca

- b) Casi Nunca
- c) Algunas Veces
- d) Casi Siempre
- e) Siempre

20. ¿Ha tenido situaciones donde el tamaño de la letra en pantalla haga difícil la navegación en el dispositivo?

- a) Nunca
- b) Casi Nunca
- c) Algunas Veces
- d) Casi Siempre
- e) Siempre

21. ¿Ha realizado algún curso educativo en internet?

- a) Si
- b) No

22. Si ha realizado algún curso educativo en internet, ¿Cómo fueron adaptados los contenidos para su diagnóstico? \_\_\_\_\_

23. ¿Hay algo más que le gustaría compartir con relación a las barreras que enfrentas al consumir contenidos digitales? \_\_\_\_\_

24. Para usted ¿Qué es importante que tenga una aplicación cuando la va a utilizar?

\_\_\_\_\_

### **Anexo C. Instrumento de recolección final para caracterización**

El propósito de esta encuesta es identificar las necesidades y preferencias la población con discapacidad visual de la fundación FUNDONAR. La información recopilada en esta encuesta será utilizada con fines académicos en la realización de un trabajo de grado en la carrera de Ingeniería de Sistemas de la Universidad Mariana.

#### **Cuestionario**

1. Nombres y Apellidos: \_\_\_\_\_
2. Cédula – Tarjeta de Identidad: \_\_\_\_\_
3. Estrato:
  - a) 1
  - b) 2
  - c) 3
  - d) 4
  - e) 5
  - f) 6
4. Género
  - a) Hombre
  - b) Mujer
  - c) No Binario
5. Edad
  - a) Menor a 10 años
  - b) De 10 a 20 años
  - c) De 20 a 30 años
  - d) De 30 a 40 años
  - e) De 40 a 50 años
  - f) Mayor de 50 años
6. Diagnóstico
  - a) Ceguera legal
  - b) Ceguera total

- c) Baja visión
7. Nivel de escolaridad
- a) Primaria incompleta
  - b) Primaria completa
  - c) Secundaria incompleta
  - d) Secundaria completa
  - e) Pregrado incompleto
  - f) Pregrado completo
  - g) Posgrado incompleto
  - h) Posgrado completo
8. Ocupación:
9. ¿Qué dispositivos utiliza para acceder a aplicaciones y contenidos digitales? (puede seleccionar varios)
- a) Smartphone
  - b) Tablet
  - c) Computador de escritorio
  - d) Computador portátil
  - e) Celular de teclas
  - f) Otro: \_\_\_\_\_
10. Seleccione las tecnologías de asistencia que utiliza para acceder a contenidos digitales (puede seleccionar varios)
- a) Lectores de Pantalla (TalkBack o VoiceOver)
  - b) Dispositivos de Braile (Teclados de braile)
  - c) Teclado
  - d) Magnificador
  - e) No utilizo tecnología de asistencia
  - f) Otro: \_\_\_\_\_
11. ¿Qué tipo de aplicaciones suele utilizar con en sus dispositivos? (puede seleccionar varios)
- a) Redes sociales (Facebook, Twitter, Instagram...)
  - b) Aplicaciones de mensajería (WhatsApp, Telegram, Line...)
  - c) Navegación por Internet (Chrome, Mozilla, Opera...)

- d) Aplicaciones de lectura de libros o noticias
  - e) Asistentes (Alexa, Ok Google)
  - f) Otro: \_\_\_\_\_
12. Me gusta interactuar con mi dispositivo mediante gestos táctiles
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
13. Me gusta recibir respuestas de mi dispositivo mediante audio (efectos de sonido, voces, música)
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
14. Me gusta recibir respuestas de mi dispositivo mediante vibración
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
15. ¿Con qué frecuencia utiliza su correo electrónico?
- a) No tengo correo electrónico
  - b) Nunca
  - c) Casi Nunca
  - d) Algunas Veces
  - e) Casi Siempre
  - f) Siempre
16. Si conoce herramientas de Inteligencia Artificial, ¿Cuál ha utilizado? (puede seleccionar varias)

- a) No he utilizado herramientas de Inteligencia Artificial
- b) ChatGPT
- c) BARD
- d) Perplexity
- e) Dall-E
- f) Bing Image

17. Si usted utiliza lector de pantalla, ¿ha tenido situaciones donde este sea incompatible con alguna aplicación?

- a) Nunca
- b) Casi Nunca
- c) Algunas Veces
- d) Casi Siempre
- e) Siempre

18. Si ha utilizado internet, ¿ha presentado dificultad a la hora de navegar por sitios web?

- a) Nunca
- b) Casi Nunca
- c) Algunas Veces
- d) Casi Siempre
- e) Siempre

19. ¿Es importante para usted que las imágenes contengan texto descriptivo?

- a) Nunca
- b) Casi Nunca
- c) Algunas Veces
- d) Casi Siempre
- e) Siempre

20. ¿Con qué frecuencia encuentra imágenes que no tienen texto descriptivo?

- a) Nunca
- b) Casi Nunca
- c) Algunas Veces
- d) Casi Siempre
- e) Siempre

21. ¿Ha tenido situaciones donde los colores en pantalla hagan difícil la navegación en el dispositivo?
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
22. ¿Ha tenido situaciones donde el tamaño de la letra en pantalla haga difícil la navegación en el dispositivo?
- a) Nunca
  - b) Casi Nunca
  - c) Algunas Veces
  - d) Casi Siempre
  - e) Siempre
23. ¿Utiliza un tamaño de letra ampliado en su dispositivo?
- a) Si
  - b) No
  - c) No Aplica
24. Al acceder a contenidos digitales. ¿Prefiere que estos tengan variedad de colores?
- a) Si
  - b) No
  - c) No Aplica
25. En los contenidos digitales, ¿Cómo prefiere que se presenten los textos?
- a) Fondo claro y texto oscuro
  - b) Fondo oscuro y texto claro
  - c) No aplica
  - d) Sin preferencia
26. En los contenidos digitales, ¿Cómo prefiere que se presenten las imágenes?
- a) Imágenes sencillas, sin muchos detalles y con pocos colores
  - b) Imágenes complejas, con muchos detalles y variedad de colores
  - c) Imágenes realistas (Fotografías)

d) No Aplica

27. ¿Ha realizado algún curso educativo en internet?

a) Si

b) No

28. Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?

a) Si

b) No

29. Si ha realizado algún curso educativo en internet, ¿En qué plataforma(s) lo realizó?

a) Si

b) No

30. Si ha realizado algún curso educativo en internet, ¿Cómo fueron adaptados los contenidos para su diagnóstico? \_\_\_\_\_

31. ¿Le gustaría aprender por medio de juegos interactivos utilizando su dispositivo?

a) Si

b) No

32. ¿Qué dificultades enfrenta al acceder a contenidos digitales?

---

---

---

33. Para usted ¿Qué es importante que tenga una aplicación cuando la va a utilizar?

---

---

---

## Anexo D. Análisis de datos – Caracterización FUNDONAR

Este Jupyter notebook se utiliza para analizar la información recolectada con el instrumento de recolección aplicado a la población con discapacidad visual de la fundación FUNDONAR, principalmente respuestas a preguntas abiertas, utilizando bibliotecas de Python orientadas a la analítica de datos y estadística. Los resultados de este análisis son parte del desarrollo de la actividad #5 del objetivo específico 1 (Caracterizar las principales barreras tecnológicas que enfrenta la población con discapacidad visual de la fundación FUNDONAR en el municipio de Pasto).

### Bibliotecas utilizadas:

- **Pandas:** Biblioteca open-source utilizada para la manipulación y análisis de datos en Python.
- **NLTK:** (Natural Language Toolkit) Biblioteca utilizada para el análisis de texto en Python.
- **Matplotlib:** (Natural Language Toolkit) Librería de visualización de datos en Python.
- **WordCloud:** Biblioteca que brinda una vista general de las palabras más frecuentes en un texto.

```
In [ ]: # Importando librerías
import pandas as pd
import matplotlib.pyplot as plt
```

Lectura de archivo .csv

```
In [ ]: data = pd.read_csv('encuesta_fundonar.csv')
```

**Pregunta: Ocupación**

**Limpieza de Datos:**

Como se observa a continuación, algunos datos que corresponden a la misma ocupación fueron escritos en mayúsculas y minúsculas (ejemplo: "Estudiante", y "ESTUDIANTE"), por lo que son considerados como registros diferentes. Para solucionar esto, se estandariza las respuestas a un solo tipo de escritura. (capitalize). Además, algunos registros cuentan con espacios en blanco (ejemplo "Estudiante" y " Estudiante "), que se removerán con el método .strip() de Python

Conteo de datos antes de la limpieza:

```
In [ ]: conteo_original = data['OCUPACIÓN'].value_counts()
        conteo_original
```

```
Out[ ]: OCUPACIÓN
        Ama de casa          4
        Servicios generales   2
        JUBILADA              2
        Vendedora             2
        VENDEDORA DE KIOSKO    1
        Sin ocupación         1
        Estudiante            1
        Pensionada            1
        Empleado              1
        Geóloga               1
        Ama de casa          1
        ESTUDIANTE            1
        Jubilado              1
        Grabados de datos     1
        Nutricionista         1
        DESEMPLEADO           1
        AMA DE CASA           1
        Estudiante            1
        Name: count, dtype: int64
```

Datos después de la limpieza:

```
In [ ]: original_values = data['OCUPACIÓN'].values
        # A cada valor se lo convierte a "Primera mayúscula" y se le retiran los espacios en blanco
        normalized = [occupation.capitalize().strip() for occupation in list(original_values)]
        normalized
```

```
Out[ ]: ['Vendedora de kiosko',
        'Estudiante',
        'Ama de casa',
        'Jubilada',
        'Jubilada',
        'Desempleado',
        'Nutricionista',
        'Ama de casa',
        'Grabados de datos',
```

```
'Jubilado',  
'Ama de casa',  
'Ama de casa',  
'Ama de casa',  
'Ama de casa',  
'Servicios generales',  
'Geóloga',  
'Empleado',  
'Vendedora',  
'Vendedora',  
'Servicios generales',  
'Pensionada',  
'Estudiante',  
'Sin ocupación',  
'Estudiante']
```

## Conteo de datos después de la limpieza

```
In [ ]: series = pd.Series(normalized)  
# Conteo de datos limpios  
count = series.value_counts()  
count
```

```
Out[ ]: Ama de casa      6  
Estudiante      3  
Jubilada        2  
Servicios generales  2  
Vendedora       2  
Vendedora de kiosko  1  
Desempleado     1  
Nutricionista   1  
Grabados de datos  1  
Jubilado        1  
Geóloga         1  
Empleado        1  
Pensionada      1  
Sin ocupación   1  
Name: count, dtype: int64
```

Ahora, existen datos que hacen referencia a una misma ocupación. Por ejemplo, Desempleado y Sin ocupación.

```
In [ ]: occupation_mapping = {  
    "Vendedora de kiosko": "Vendedora",  
    "Desempleado": "Sin ocupación",  
    "Pensionada": "Jubilado",  
    "Jubilada" : "Jubilado"  
}  
  
cleaned_data = [occupation_mapping.get(occupation, occupation) for occupation in normalized]  
series = pd.Series(cleaned_data)
```

```
# Conteo de datos limpios
count = series.value_counts()
count
```

```
Out[ ]: Ama de casa          6
        Jubilado           4
        Vendedora          3
        Estudiante         3
        Sin ocupación      2
        Servicios generales 2
        Nutricionista       1
        Grabados de datos   1
        Geóloga            1
        Empleado           1
        Name: count, dtype: int64
```

Para mantener la neutralidad en los nombres de las ocupaciones se implementa la siguiente función que reemplaza los sustantivos masculinos/femeninos con @

```
In [ ]: def make_inclusive(occupation):
        splitted = occupation.split(' ')
        length = len(splitted[0])
        if(splitted[0][length-1] == 'o' or splitted[0][length-1] == 'a'):
            inclusive_word = splitted[0][0:length-1]
            splitted[0] = inclusive_word+'@'
        return (' ').join(splitted)
```

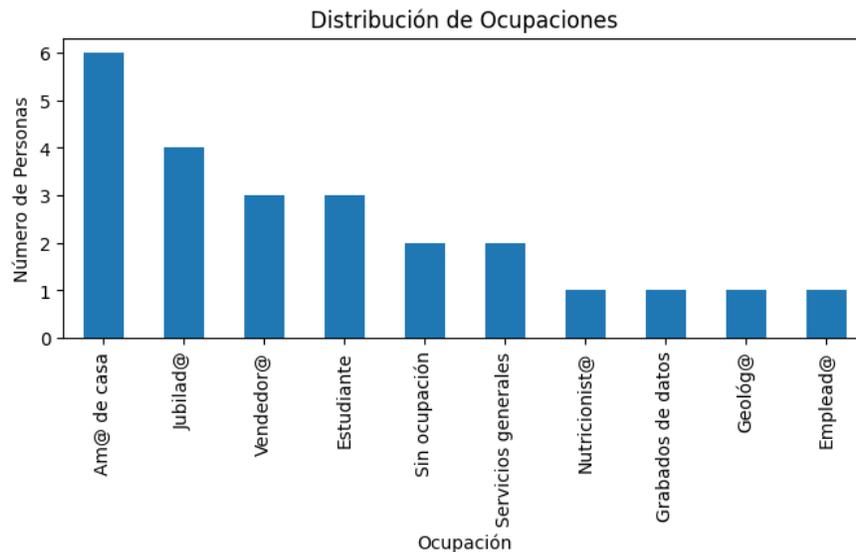
Obteniendo los siguientes resultados

```
In [ ]: final_data = [make_inclusive(occupation) for occupation in cleaned_data]
        series = pd.Series(final_data)
        # Conteo de datos limpios y neutros
        count = series.value_counts()
        count
```

```
Out[ ]: Am@ de casa          6
        Jubilad@           4
        Vendedor@          3
        Estudiante         3
        Sin ocupación      2
        Servicios generales 2
        Nutricionist@       1
        Grabados de datos   1
        Geológ@            1
        Emplead@           1
        Name: count, dtype: int64
```

Finalmente, se muestra la gráfica que indica los resultados después de la limpieza de datos:

```
In [ ]: plt.figure(figsize=(8, 3))
count.plot(kind='bar')
plt.title('Distribución de Ocupaciones')
plt.xlabel('Ocupación')
plt.ylabel('Número de Personas')
plt.show()
```



**Pregunta: Para usted ¿Qué es importante que tenga una aplicación cuando la va a utilizar?**

Esta pregunta es de libre respuesta. Para analizar sus respuestas, se hace uso de la biblioteca nltk, que provee las siguientes herramientas:

- **word\_tokenize:** Convierte las respuestas en palabras individuales (tokens), listas para ser analizadas
- **stopwords:** Identifica palabras comunes que son descartadas en el análisis de texto en un lenguaje concreto. Por ejemplo, "a", "ah", "como", etc.

```
In [ ]: import nltk
from nltk import word_tokenize
from nltk.corpus import stopwords
from collections import Counter
import matplotlib.pyplot as plt
from wordcloud import WordCloud
```

## Respuestas antes del análisis

```
In [ ]: responses = list(data['Para usted ¿Qué es importante que tenga una aplicación cuando la va a utili
responses = [response for response in responses if str(response)!='nan']
responses

Out[ ]: ['Audio y vibración',
'No aplica',
'Sonido y vibración',
'Que tenga audio',
'no aplica',
'Imágenes sencillas y voz ',
'Voz ',
'Imágenes claras y audio ',
'Imágenes claras y texto corto ',
'Imágenes claras y texto corto ',
'Imágenes grandes y sencillas ',
'Imágenes y textos claros y cortos ',
'Imágenes con pocos colores',
'Audio e imágenes grandes y coloridas ',
'Fondo oscuro y letras blancas o amarillas ',
'Imágenes claras y grandes ',
'Imágenes y letra grande ',
'Audio',
'Audio ',
'Imágenes grandes, audio, fondo oscuro ',
'Audio, más imágenes y grandes ',
'Audio y tacto únicamente ',
'Audio, letra e imágenes grandes ']
```

Se descargan complementos necesarios para realizar el análisis con nltk. Adicionalmente, se agrega la coma (",") a las stop\_words, pues esta era contada como palabra separada por la librería.

```
In [ ]: nltk.download("punkt")
nltk.download("stopwords")
nltk.download("omw")
stop_words = set(stopwords.words("spanish"))
stop_words.add(",")

[nltk_data] Downloading package punkt to
[nltk_data] C:\Users\alber\AppData\Roaming\nltk_data...
[nltk_data] Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data] C:\Users\alber\AppData\Roaming\nltk_data...
[nltk_data] Package stopwords is already up-to-date!
[nltk_data] Downloading package omw to
[nltk_data] C:\Users\alber\AppData\Roaming\nltk_data...
[nltk_data] Package omw is already up-to-date!
```

De esta manera, se realiza el proceso de tokenización en cada respuesta. Por ejemplo, obsérvese cómo la primera frase es tokenizada, extrayendo únicamente las palabras relevantes para el análisis.

```
In [ ]: tokens_first_answer = [word.lower() for word in word_tokenize(responses[5]) if word.lower() not in tokens_first_answer]
```

```
Out[ ]: ['imágenes', 'sencillas', 'voz']
```

Asimismo, se realiza el proceso para cada respuesta en la lista de respuestas de la encuesta. Con la función Counter de la biblioteca collections, se muestra la frecuencia de cada palabra.

```
In [ ]: tokens = [word.lower() for response in responses for word in word_tokenize(response) if word.lower() not in tokens]
word_freq = Counter(tokens)
word_freq
```

```
Out[ ]: Counter({'imágenes': 13,
                'audio': 10,
                'grandes': 6,
                'claras': 4,
                'vibración': 2,
                'aplica': 2,
                'sencillas': 2,
                'voz': 2,
                'texto': 2,
                'corto': 2,
                'fondo': 2,
                'oscuro': 2,
                'letra': 2,
                'sonido': 1,
                'textos': 1,
                'claros': 1,
                'cortos': 1,
                'pocos': 1,
                'colores': 1,
                'coloridas': 1,
                'letras': 1,
                'blancas': 1,
                'amarillas': 1,
                'grande': 1,
                'tacto': 1,
                'únicamente': 1})
```

Finalmente, se hace uso de la biblioteca wordcloud y matplotlib para mostrar una representación gráfica (nube de palabras) que ilustra las palabras más recurrentes de manera visual.

```
In [ ]: plt.figure(figsize=(10, 6))
wordcloud = WordCloud(width=800, height=400, background_color="white").generate_from_frequencies(word_freq)
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.title("Análisis de Frecuencia de Palabras")
plt.show()
```



Adicionalmente, se puede extraer las 5 palabras más frecuentes en las respuestas de los encuestados

```
In [ ]: top_words = word_freq.most_common(5)
print("Palabras más frecuentes y sus frecuencias:")
for word, freq in top_words:
    print(f"{word}: {freq}")
```

```
Palabras más frecuentes y sus frecuencias:
imágenes: 13
audio: 10
grandes: 6
claras: 4
vibración: 2
```

## Tablas de contingencia

Pregunta: ¿Le gustaría aprender por medio de juegos interactivos utilizando su dispositivo? El resultado a esta pregunta fue 70.8% No, 29.2% Si. Se cruza estos resultados con la variable "Edad", realizando una tabla de contingencia con la biblioteca Pandas, de Python

```
In [ ]: contingency_table = pd.crosstab(data['¿Le gustaría aprender por medio de juegos interactivos utili
contingency_table = contingency_table.rename_axis("Le gustaria aprender...")
contingency_table
```

```
Out[ ]:
```

Edad	De 10 a 20 años	De 20 a 30 años	De 30 a 40 años	De 40 a 50 años	Mayor a 50 años	Menor a 10 años
<b>Le gustaria aprender...</b>						
No	0	0	0	0	6	1
Sí	3	1	4	4	5	0

Del mismo modo, para la pregunta 'Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?' se realiza una tabla de contingencia con la variable 'Nivel de escolaridad'

```
In [ ]: contingency_table = pd.crosstab(data['Si no ha realizado cursos educativos en internet ¿Le gustaría
contingency_table = contingency_table.rename_axis("Le gustaria realizar cursos...")
contingency_table.T
```

Out[ ]: **Le gustaria realizar cursos... No Sí**

Nivel de escolaridad		
	No	Sí
<b>PREGRADO COMPLETO</b>	2	2
<b>PRIMARIA COMPLETA</b>	2	6
<b>PRIMARIA INCOMPLETA</b>	4	1
<b>SECUNDARIA COMPLETA</b>	0	4
<b>SECUNDARIA INCOMPLETA</b>	0	2
<b>Técnico</b>	0	1

Tabla de contingencia para "Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?" vs "Diagnóstico"

```
In [ ]: contingency_table = pd.crosstab(data['Si no ha realizado cursos educativos en internet ¿Le gustaría
contingency_table = contingency_table.rename_axis("Le gustaria realizar cursos...")
contingency_table.T
```

Out[ ]: **Le gustaria realizar cursos... No Sí**

Diagnóstico		
	No	Sí
<b>BAJA VISION</b>	6	14
<b>CEGUERA LEGAL</b>	1	2
<b>CEGUERA TOTAL</b>	1	0

Tabla de contingencia para "Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?" vs "¿Con qué frecuencia utiliza su correo electrónico?"

```
In [ ]: contingency_table = pd.crosstab(data['Si no ha realizado cursos educativos en internet ¿Le gustaría realizar cursos...'], data['Uso de correo electrónico'], axis=1)
contingency_table = contingency_table.rename_axis("Le gustaría realizar cursos...", axis="rows")
contingency_table = contingency_table.rename_axis("Uso de correo", axis="columns")
contingency_table.T
```

```
Out[ ]: Le gustaría realizar cursos... No Sí
```

Uso de correo		
Algunas veces	2	7
Casi nunca	0	3
Casi siempre	0	2
No tengo correo electrónico	5	3
Nunca	1	1

Tabla de contingencia para "Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?" vs "¿Le gustaría aprender por medio de juegos interactivos utilizando su dispositivo?"

```
In [ ]: contingency_table = pd.crosstab(data['Si no ha realizado cursos educativos en internet ¿Le gustaría realizar cursos...'], data['Le gustaría aprender con juegos'], axis=1)
contingency_table = contingency_table.rename_axis("Le gustaría realizar cursos...", axis="rows")
contingency_table = contingency_table.rename_axis("Le gustaría aprender con juegos", axis="columns")
contingency_table.T
```

```
Out[ ]: Le gustaría realizar cursos... No Sí
```

Le gustaría aprender con juegos		
No	7	0
Sí	1	16

Tabla de contingencia para "Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?" vs "Qué dispositivos utiliza".

```
In [ ]: contingency_table = pd.crosstab(data['Si no ha realizado cursos educativos en internet ¿Le gustaría realizar cursos...'], data['Dispositivos Utilizados'])
contingency_table = contingency_table.rename_axis("Le gustaría realizar cursos...")
contingency_table = contingency_table.rename_axis("Dispositivos Utilizados",axis="columns")
contingency_table.T
```

```
Out[ ]:
```

	Le gustaría realizar cursos...	No	Sí
<b>Dispositivos Utilizados</b>			
	<b>Celular de teclas</b>	1	1
	<b>Smartphone</b>	6	4
	<b>Smartphone, Computador de escritorio</b>	0	5
	<b>Smartphone, Computador de escritorio, Computador portátil</b>	0	4
	<b>Smartphone, Tablet, Computador de escritorio, Computador portátil</b>	0	2

Tabla de contingencia para "Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?" vs "Seleccione las tecnologías de asistencia que utiliza para acceder a contenidos digitales (puede seleccionar varios)".

```
In [ ]: columns = data[['Si no ha realizado cursos educativos en internet ¿Le gustaría hacerlo?','Seleccione las tecnologías de asistencia que utiliza para acceder a contenidos digitales (puede seleccionar varios)']]
values = list(columns.values)

all_devices = []
all_responses = []

for value in values:
    if str(value[1]) != "nan":
        for device in value[1].split(','):
            all_responses.append(value[0])
            all_devices.append(device.strip().capitalize())

contingency_table = pd.crosstab(all_devices, all_responses)
contingency_table.index.name = "Asistencia utilizada"
contingency_table.columns.name = "Le gustaría realizar cursos..."
contingency_table
```

```
Out[ ]:
```

	Le gustaría realizar cursos...	No	Sí
<b>Asistencia utilizada</b>			
	<b>Lectores de pantalla (talkback o voiceover)</b>	0	5
	<b>Magnificador</b>	4	12
	<b>No utilizo tecnología de asistencia</b>	4	2

## **Anexo E. Instrumento SUS para evaluación de usabilidad**

Este documento ha sido creado con el propósito de brindarle una comprensión detallada sobre la naturaleza y características esenciales de la investigación que se está llevando a cabo. Nuestro objetivo es proporcionar información clara y concisa que le permita tomar una decisión informada sobre su posible participación voluntaria en este estudio.

Si después de leer la información proporcionada, aún tiene preguntas o inquietudes, le invitamos cordialmente a ponerse en contacto con el equipo de investigación responsable. Nos comprometemos a garantizar que cada posible participante esté completamente informado y comprenda todos los aspectos relacionados con la investigación. Por lo tanto, estaremos disponibles para ofrecer aclaraciones detalladas y responder a todas sus preguntas para asegurar su comprensión completa y satisfacción.

### **Importancia del proyecto**

Este documento ha sido creado con el propósito de brindarle una comprensión detallada sobre la naturaleza y características esenciales de la investigación que se está llevando a cabo. Nuestro objetivo es proporcionar información clara y concisa que le permita tomar una decisión informada sobre su posible participación voluntaria en este estudio.

### **Objetivo general de la investigación**

El objetivo general de esta investigación es proponer una estrategia computacional basada en tecnologías móviles para disminuir la brecha digital existente en cuanto al acceso a contenidos digitales en personas con discapacidad visual pertenecientes a la fundación FUNDONAR en el municipio de Pasto. La estrategia computacional desarrollada proporciona un acceso equitativo a contenidos digitales educativos, eliminando barreras digitales y adaptándose a las preferencias de cada usuario, independientemente de su diagnóstico visual.

### **Descripción de la investigación**

La investigación se enfoca en la creación de contenidos digitales para personas con discapacidad visual. Se eligió la plataforma móvil Android para desarrollar una aplicación educativa que ofrece la posibilidad de acceder a contenidos digitales accesibles, incluyendo materiales educativos diversos y minijuegos adaptados para personas con discapacidad visual, con el fin de fomentar su

participación activa y su aprendizaje interactivo. En el proceso de desarrollo, se priorizó la accesibilidad y la usabilidad de la aplicación, siguiendo las pautas establecidas en la normativa WCAG 2.1. Se implementaron funcionalidades clave, como la capacidad de ajustar el tamaño de fuente y seleccionar temas de color adecuados para mejorar la legibilidad y la experiencia visual del usuario. Además, se garantiza la compatibilidad con lectores de pantalla modernos, como Talkback y VoiceOver, para facilitar la navegación y la interacción con la aplicación. Una de las ventajas destacadas de esta aplicación es su capacidad para ampliar continuamente su biblioteca de contenidos. Se ha diseñado con la flexibilidad necesaria para agregar nuevos materiales educativos y minijuegos de manera regular, lo que ofrece a los usuarios una experiencia enriquecedora y variada. Los contenidos pueden presentarse en diversos formatos, incluido el audio, lo que amplía aún más las posibilidades de accesibilidad y personalización para adaptarse a las preferencias individuales de cada usuario.

### **Responsables de la investigación**

Este estudio es liderado por el docente Fabián Parra Pay y el estudiante Carlos Alberto Botina Carpio, ambos pertenecientes al programa de Ingeniería de Sistemas de la Universidad Mariana de Pasto. Para cualquier consulta o inquietud, no dude en ponerse en contacto a través del número telefónico 3215553519.

### **Confidencialidad**

En la investigación, se prioriza la protección, el respeto y la autonomía de todos los participantes. Se garantiza un proceso de participación voluntaria, informada y libre de cualquier forma de coerción, asegurando en todo momento la privacidad y dignidad de cada individuo involucrado. Es importante destacar que este estudio no tiene motivaciones lucrativas ni busca imponer ninguna forma de subordinación sobre los participantes. La información recopilada será tratada con el máximo rigor y estará accesible únicamente para el equipo de investigación, garantizando la confidencialidad y la no divulgación de datos individuales. Todos los resultados publicados serán presentados de forma agregada, protegiendo así la anonimidad y privacidad de los participantes. Además, los datos recopilados serán almacenados de manera segura durante un período de cinco años después de la conclusión de la investigación.

### **Derechos y deberes**

Usted tiene derecho a recibir una copia de este documento y puede decidir retirarse de la investigación en cualquier momento sin necesidad de firmar ningún documento o explicar sus razones si así lo prefiere. Durante su participación en la investigación, no tendrá ningún costo y puede solicitar información sobre los resultados en cualquier momento a los responsables del estudio. Confirmando haber leído o escuchado la lectura completa de este documento y haber comprendido su contenido. Reconozco también que tengo la libertad de plantear cualquier pregunta y que todas mis inquietudes han sido satisfactoriamente abordadas. Con este entendimiento, elijo participar voluntariamente en esta investigación.

1. Declaro que he leído detenidamente el consentimiento informado y, tras comprender su contenido, acepto voluntariamente participar en el estudio propuesto.
  - a) Si
  - b) No
2. Nombres y apellidos
3. Cédula
4. Edad
5. Género
  - a) Masculino
  - b) Femenino
  - c) No Binario
6. Prefiero no decirlo

#### Preguntas de usabilidad

1. Creo que me gustaría utilizar esta aplicación con frecuencia
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
2. Encontré la aplicación innecesariamente compleja
  - a. Totalmente de acuerdo

- b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
3. Pensé que la aplicación es fácil de usar
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
4. Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
5. Encontré que las diversas funciones de esta aplicación están bien integradas
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
6. Pensé que había demasiada inconsistencia en esta aplicación
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
7. Me imagino que la mayoría de la gente aprendería a utilizar esta aplicación rápidamente
- a. Totalmente de acuerdo
  - b. De acuerdo

- c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
8. Encontré la aplicación muy complicada de usar
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
9. Me sentí muy segur@ usando la aplicación
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
10. Necesitaba aprender muchas cosas antes de empezar con esta aplicación
- a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo

## Anexo F. Análisis de datos - Cuestionario SUS

En el presente notebook se realiza un proceso de limpieza y análisis de datos para los resultados obtenidos en la encuesta de usabilidad aplicada a la fundación FUNDONAR como parte del proyecto "Accesibilidad aplicada a la creación de contenidos digitales para personas con discapacidad visual"

### Librerías por utilizar

- pandas: Utilizada para el análisis y manipulación de datos en distintos formatos

```
In [ ]: import pandas as pd
```

Se realiza una lectura del archivo csv con los resultados del instrumento de recolección y se muestra la tabla correspondiente

```
In [ ]: data = pd.read_csv('data/results.csv')
data
```

Out[ ]:

	Timestamp	Declaro que he leído detenidamente el consentimiento informado y, tras comprender su contenido, acepto voluntariamente participar en el estudio propuesto	Nombres y apellidos	Cédula	Edad	Género	Creo que me gustaría utilizar esta aplicación con frecuencia	Encontré la aplicación innecesariamente compleja	Pensé que la aplicación es fácil de usar
0	3/7/2024 11:41:45	Sí	Yelixa Samboni	1085662812	10	Femenino	De acuerdo	En desacuerdo	De acuerdo
1	3/7/2024 11:53:19	Sí	Hernán Guerrero	98381500	53	Masculino	Totalmente de acuerdo	En desacuerdo	Totalmente de acuerdo
2	3/8/2024 15:29:49	Sí	Mercedes Ramírez	27246968	77	Femenino	De acuerdo	En desacuerdo	De acuerdo
3	3/8/2024 15:34:05	Sí	Libardo Campaña	1871846	82	Masculino	De acuerdo	En desacuerdo	De acuerdo

Se muestran los nombres de las columnas para eliminar aquellas que no se utilizan en el análisis

```
In [ ]: cols = data.columns.tolist()
        cols
```

```
Out[ ]: ['Timestamp',
        'Declaro que he leído detenidamente el consentimiento informado y, tras comprender su contenido,
        acepto voluntariamente participar en el estudio propuesto',
        'Nombres y apellidos',
        'Cédula',
        'Edad',
        'Género',
        'Creo que me gustaría utilizar esta aplicación con frecuencia ',
        'Encontré la aplicación innecesariamente compleja',
        'Pensé que la aplicación es fácil de usar',
        'Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación',
        'Encontré que las diversas funciones de esta aplicación están bien integradas',
        'Pensé que había demasiada inconsistencia en esta aplicación',
        'Me imagino que la mayoría de la gente aprendería a utilizar esta aplicación rápidamente',
        'Encontré la aplicación muy complicada de usar',
        'Me sentí muy segur@ usando la aplicación',
        'Necesitaba aprender muchas cosas antes de empezar con esta aplicación']
```

```
In [ ]: cols_to_delete = [
        cols[0],
        cols[1],
        ]

df = data.drop(cols_to_delete, axis=1)
df
```

```
Out[ ]:
```

	Nombres y apellidos	Cédula	Edad	Género	Creo que me gustaría utilizar esta aplicación con frecuencia	Encontré la aplicación innecesariamente compleja	Pensé que la aplicación es fácil de usar	Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación	Encontré que las diversas funciones de esta aplicación están bien integradas	
0	Yelixa Samboni	1085662812	10	Femenino	De acuerdo	En desacuerdo	De acuerdo	En desacuerdo	De acuerdo	En c
1	Hernán Guerrero	98381500	53	Masculino	Totalmente de acuerdo	En desacuerdo	Totalmente de acuerdo	En desacuerdo	De acuerdo	En c
	Mercedes				De		De	En	De	

A partir de las nuevas columnas, se separan las preguntas de tipo 1 y de tipo 2

```
In [ ]: positives = [
    new_cols[4],
    new_cols[6],
    new_cols[8],
    new_cols[10],
    new_cols[12],
]

negatives = [
    new_cols[5],
    new_cols[7],
    new_cols[9],
    new_cols[11],
    new_cols[13],
]
```

De esta manera, se crea un nuevo DataFrame únicamente con las preguntas de tipo 1

```
In [ ]: positive_stats = df[positives]
positive_stats
```

Out[ ]:

	Creo que me gustaría utilizar esta aplicación con frecuencia	Pensé que la aplicación es fácil de usar	Encontré que las diversas funciones de esta aplicación están bien integradas	Me imagino que la mayoría de la gente aprendería a utilizar esta aplicación rápidamente	Me sentí muy segur@ usando la aplicación
0	De acuerdo	De acuerdo	De acuerdo	De acuerdo	De acuerdo
1	Totalmente de acuerdo	Totalmente de acuerdo	De acuerdo	De acuerdo	De acuerdo
2	De acuerdo	De acuerdo	De acuerdo	Totalmente de acuerdo	De acuerdo
3	De acuerdo	De acuerdo	De acuerdo	De acuerdo	De acuerdo
4	Totalmente de acuerdo	De acuerdo	De acuerdo	De acuerdo	Totalmente de acuerdo

Y un DataFrame con las preguntas de tipo 2

```
In [ ]: negative_stats = df[negatives]
negative_stats
```

Out[ ]:

	Encontré la aplicación innecesariamente compleja	Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación	Pensé que había demasiada inconsistencia en esta aplicación	Encontré la aplicación muy complicada de usar	Necesitaba aprender muchas cosas antes de empezar con esta aplicación
0	En desacuerdo	En desacuerdo	En desacuerdo	En desacuerdo	En desacuerdo
1	En desacuerdo	En desacuerdo	En desacuerdo	En desacuerdo	En desacuerdo
2	En desacuerdo	En desacuerdo	En desacuerdo	En desacuerdo	En desacuerdo
3	En desacuerdo	Neutral	En desacuerdo	En desacuerdo	En desacuerdo

Luego, se definen los valores que tendrán cada respuesta dependiendo del tipo de pregunta

```
In [ ]: positive_answer_map = {
    "Totalmente de acuerdo": 4,
    "De acuerdo":3,
    "Neutral":2,
    "En desacuerdo":1,
    "Totalmente en desacuerdo":0,
}

negative_answer_map = {
    "Totalmente de acuerdo": 0,
    "De acuerdo":1,
    "Neutral":2,
    "En desacuerdo":3,
    "Totalmente en desacuerdo":4,
}
```

Y se aplica en ambos DataFrame

```
In [ ]: positive_stats_numeric = positive_stats.applymap(lambda x: positive_answer_map.get(x))
positive_stats_numeric
```

Out[ ]:

	Creo que me gustaría utilizar esta aplicación con frecuencia	Pensé que la aplicación es fácil de usar	Encontré que las diversas funciones de esta aplicación están bien integradas	Me imagino que la mayoría de la gente aprendería a utilizar esta aplicación rápidamente	Me sentí muy segur@ usando la aplicación
0	3	3	3	3	3
1	4	4	3	3	3
2	3	3	3	4	3

```
In [ ]: negative_stats_numeric = negative_stats.applymap(lambda x: negative_answer_map.get(x))
negative_stats_numeric
```

Out[ ]:

	Encontré la aplicación innecesariamente compleja	Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación	Pensé que había demasiada inconsistencia en esta aplicación	Encontré la aplicación muy complicada de usar	Necesitaba aprender muchas cosas antes de empezar con esta aplicación
0	3	3	3	3	3
1	3	3	3	3	3
2	3	3	3	3	3
3	3	2	3	3	3

Posteriormente, se concatenan ambos DataFrame, obteniendo el DataFrame con los datos mapeados de todas las preguntas

```
In [ ]: combined_stats = pd.concat([positive_stats_numeric, negative_stats_numeric], axis = 1)
combined_stats
```

```
Out[ ]:
```

	Creo que me gustaría utilizar esta aplicación con frecuencia	Pensé que la aplicación es fácil de usar	Encontré que las diversas funciones de esta aplicación están bien integradas	Me imagino que la mayoría de la gente aprendería a utilizar esta aplicación rápidamente	Me sentí muy segur@ usando la aplicación	Encontré la aplicación innecesariamente compleja	Creo que necesitaría el apoyo de un técnico para poder utilizar la aplicación	Pensé que había demasiada inconsistencia en esta aplicación	Encont aplica compli de
0	3	3	3	3	3	3	3	3	3
1	4	4	3	3	3	3	3	3	3
2	3	3	3	4	3	3	3	3	3
3	3	3	3	3	3	3	2	3	3

Finalmente, se calcula la sumatoria de todos los valores

```
In [ ]: total_sum = combined_stats.sum().sum()
total_sum
```

```
Out[ ]: 322
```

La cual se compara con la mayor puntuación posible

```
In [ ]: rows, cols = combined_stats.shape
max_value = rows*cols*4
max_value
```

```
Out[ ]: 400
```

Obteniendo el puntaje final que indica el nivel de usabilidad percibido por los usuarios en general

```
In [ ]: average = total_sum/max_value
average*100
```

```
Out[ ]: 80.5
```

## **Anexo G. Instrumento de evaluación - Proceso de desarrollo de software**

Este documento ha sido creado con el propósito de brindarle una comprensión detallada sobre la naturaleza y características esenciales de la investigación que se está llevando a cabo. Nuestro objetivo es proporcionar información clara y concisa que le permita tomar una decisión informada sobre su posible participación voluntaria en este estudio.

Si después de leer la información proporcionada, aún tiene preguntas o inquietudes, le invitamos cordialmente a ponerse en contacto con el equipo de investigación responsable. Nos comprometemos a garantizar que cada posible participante esté completamente informado y comprenda todos los aspectos relacionados con la investigación. Por lo tanto, estaremos disponibles para ofrecer aclaraciones detalladas y responder a todas sus preguntas para asegurar su comprensión completa y satisfacción.

### **Importancia del proyecto**

Este documento ha sido creado con el propósito de brindarle una comprensión detallada sobre la naturaleza y características esenciales de la investigación que se está llevando a cabo. Nuestro objetivo es proporcionar información clara y concisa que le permita tomar una decisión informada sobre su posible participación voluntaria en este estudio.

### **Objetivo general de la investigación**

El objetivo general de esta investigación es proponer una estrategia computacional basada en tecnologías móviles para disminuir la brecha digital existente en cuanto al acceso a contenidos digitales en personas con discapacidad visual pertenecientes a la fundación FUNDONAR en el municipio de Pasto. La estrategia computacional desarrollada proporciona un acceso equitativo a contenidos digitales educativos, eliminando barreras digitales y adaptándose a las preferencias de cada usuario, independientemente de su diagnóstico visual.

### **Descripción de la investigación**

La investigación se enfoca en la creación de contenidos digitales para personas con discapacidad visual. Se eligió la plataforma móvil Android para desarrollar una aplicación educativa que ofrece la posibilidad de acceder a contenidos digitales accesibles, incluyendo materiales educativos diversos y minijuegos adaptados para personas con discapacidad visual, con el fin de fomentar su

participación activa y su aprendizaje interactivo. En el proceso de desarrollo, se priorizó la accesibilidad y la usabilidad de la aplicación, siguiendo las pautas establecidas en la normativa WCAG 2.1. Se implementaron funcionalidades clave, como la capacidad de ajustar el tamaño de fuente y seleccionar temas de color adecuados para mejorar la legibilidad y la experiencia visual del usuario. Además, se garantiza la compatibilidad con lectores de pantalla modernos, como Talkback y VoiceOver, para facilitar la navegación y la interacción con la aplicación. Una de las ventajas destacadas de esta aplicación es su capacidad para ampliar continuamente su biblioteca de contenidos. Se ha diseñado con la flexibilidad necesaria para agregar nuevos materiales educativos y minijuegos de manera regular, lo que ofrece a los usuarios una experiencia enriquecedora y variada. Los contenidos pueden presentarse en diversos formatos, incluido el audio, lo que amplía aún más las posibilidades de accesibilidad y personalización para adaptarse a las preferencias individuales de cada usuario.

### **Responsables de la investigación**

Este estudio es liderado por el docente Fabián Parra Pay y el estudiante Carlos Alberto Botina Carpio, ambos pertenecientes al programa de Ingeniería de Sistemas de la Universidad Mariana de Pasto. Para cualquier consulta o inquietud, no dude en ponerse en contacto a través del número telefónico 3215553519.

### **Confidencialidad**

En la investigación, se prioriza la protección, el respeto y la autonomía de todos los participantes. Se garantiza un proceso de participación voluntaria, informada y libre de cualquier forma de coerción, asegurando en todo momento la privacidad y dignidad de cada individuo involucrado. Es importante destacar que este estudio no tiene motivaciones lucrativas ni busca imponer ninguna forma de subordinación sobre los participantes. La información recopilada será tratada con el máximo rigor y estará accesible únicamente para el equipo de investigación, garantizando la confidencialidad y la no divulgación de datos individuales. Todos los resultados publicados serán presentados de forma agregada, protegiendo así la anonimidad y privacidad de los participantes. Además, los datos recopilados serán almacenados de manera segura durante un período de cinco años después de la conclusión de la investigación.

### **Derechos y deberes**

Usted tiene derecho a recibir una copia de este documento y puede decidir retirarse de la investigación en cualquier momento sin necesidad de firmar ningún documento o explicar sus razones si así lo prefiere. Durante su participación en la investigación, no tendrá ningún costo y puede solicitar información sobre los resultados en cualquier momento a los responsables del estudio. Confirmando haber leído o escuchado la lectura completa de este documento y haber comprendido su contenido. Reconozco también que tengo la libertad de plantear cualquier pregunta y que todas mis inquietudes han sido satisfactoriamente abordadas. Con este entendimiento, elijo participar voluntariamente en esta investigación.

1. Declaro que he leído detenidamente el consentimiento informado y, tras comprender su contenido, acepto voluntariamente participar en el estudio propuesto.
  - a) Si
  - b) No

#### Datos personales

1. Nombres y Apellidos
2. Perfil Profesional
3. Años de experiencia en construcción de software

#### Preguntas sobre el software desarrollado

1. El proyecto sigue una metodología de desarrollo de software clara
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
2. Agregar nuevas funcionalidades al proyecto resultaría complejo y costoso
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo

3. El proyecto cuenta con una estructura de directorios adecuada
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
4. El código fuente presenta repetición de código innecesaria
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
5. Las herramientas de desarrollo utilizadas fueron adecuadas para el proyecto
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
6. La aplicación experimenta problemas de rendimiento o tiempos de respuesta lentos en ciertas situaciones.
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
7. Las historias de usuario del proyecto cumplen con el criterio SMART (Specific, Measurable, Achievable, Relevant, Time-bounded)
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo

- e. Totalmente en desacuerdo
- 8. La aplicación tarda mucho tiempo en iniciar desde cero
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
- 9. La implementación programática de normas de accesibilidad es adecuada
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
- 10. La elección del tipo de bases de datos es injustificada
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
- 11. La aplicación presenta problemas de seguridad
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo
- 12. Los métodos de testing utilizados son adecuados
  - a. Totalmente de acuerdo
  - b. De acuerdo
  - c. Neutral
  - d. En desacuerdo
  - e. Totalmente en desacuerdo

13. La aplicación posee una interfaz gráfica poco intuitiva
- Totalmente de acuerdo
  - De acuerdo
  - Neutral
  - En desacuerdo
  - Totalmente en desacuerdo
14. La metodología de desarrollo es aplicada de manera inconsistente
- Totalmente de acuerdo
  - De acuerdo
  - Neutral
  - En desacuerdo
  - Totalmente en desacuerdo
15. Considera útil que este tipo de aplicaciones sean desarrolladas para la población con discapacidad visual
- Totalmente de acuerdo
  - De acuerdo
  - Neutral
  - En desacuerdo
  - Totalmente en desacuerdo
16. En su experiencia como desarrollador de software, considera necesario el desarrollo de este tipo de aplicaciones.
- Totalmente de acuerdo
  - De acuerdo
  - Neutral
  - En desacuerdo
  - Totalmente en desacuerdo
17. ¿Tiene alguna recomendación o comentario del proyecto realizado?
- Totalmente de acuerdo
  - De acuerdo
  - Neutral
  - En desacuerdo

e. Totalmente en desacuerdo

**Anexo A.** Nombre del anexo

## Anexo H. Análisis de datos – Evaluación de expertos

En el presente notebook se realiza un proceso de limpieza y análisis de datos para los resultados obtenidos en la encuesta realizada para la evaluación del proceso de desarrollo de software en el proyecto "Accesibilidad aplicada a la creación de contenidos digitales para personas con discapacidad visual" teniendo en cuenta las métricas de "Adecuación funcional" y "Mantenibilidad".

### Librerías por utilizar

- pandas: Utilizada para el análisis y manipulación de datos en distintos formatos

```
In [ ]: data = pd.read_csv('data/evaluacion_expertos.csv')
data
```

Se realiza una lectura del archivo csv con los resultados del instrumento de recolección y se muestra la tabla correspondiente

```
In [ ]: data = pd.read_csv('data/evaluacion_expertos.csv')
data
```

Out[ ]:

	Timestamp	Declaro que he leído detenidamente el consentimiento informado y, tras comprender su contenido, acepto voluntariamente participar en el estudio propuesto	Nombres y Apellidos	Perfil Profesional	Años de experiencia en construcción de software	El proyecto sigue una metodología de desarrollo de software clara	Agregar nuevas funcionalidades al proyecto resultaría complejo y costoso	El proyecto cuenta con una estructura de directorios adecuada	
0	4/5/2024 10:16:08	Sí	Wilson Andrés Castillo Castro	Docente Ingeniería de Sistemas	6	Totalmente de acuerdo	Neutral	Totalmente de acuerdo	Tc de
1	4/6/2024 11:10:17	Sí	Giovanni Hernández	Ingeniero de Sistemas	27	Totalmente de acuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	de

Se eliminan las columnas que no se utilizarán para el análisis

```
In [ ]: cols_to_delete = [
    cols[0],
    cols[1],
    cols[2],
    cols[3],
    cols[4],
]

df = data.drop(cols_to_delete, axis=1)
df
```

```
Out[ ]:
```

	El proyecto sigue una metodología de desarrollo de software clara	Agregar nuevas funcionalidades al proyecto resultaría complejo y costoso	El proyecto cuenta con una estructura de directorios adecuada	El código fuente presenta repetición de código innecesaria	Las herramientas de desarrollo utilizadas fueron adecuadas para el proyecto	La aplicación experimenta problemas de rendimiento o tiempos de respuesta lentos en ciertas situaciones.	Las historias de usuario del proyecto cumplen con el criterio SMART (Specific, Measurable, Achievable, Relevant, Time-boundeds)	La aplicación tarda mucho tiempo en iniciar desde cero	imp F d acc
0	Totalmente de acuerdo	Neutral	Totalmente de acuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	Totalmente en desacuerdo	1
1	Totalmente de acuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	En desacuerdo	De acuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	Totalmente en desacuerdo	1
2	Totalmente de acuerdo	De acuerdo	Totalmente de acuerdo	En desacuerdo	De acuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	Totalmente en desacuerdo	

A partir de las nuevas columnas, se separan las preguntas de tipo 1 y de tipo 2

```
In [ ]: positives = [
    new_cols[0],
    new_cols[2],
    new_cols[4],
    new_cols[6],
    new_cols[8],
    new_cols[11],
    new_cols[14],
    new_cols[15],
]
```

```
negatives = [
    new_cols[1],
    new_cols[3],
    new_cols[5],
    new_cols[7],
    new_cols[9],
    new_cols[10],
    new_cols[12],
    new_cols[13],
]
```

De esta manera, se crea un nuevo DataFrame únicamente con las preguntas de tipo 1

```
In [ ]: positive_stats = df[positives]
positive_stats
```

Out[ ]:

	El proyecto sigue una metodología de desarrollo de software clara	El proyecto cuenta con una estructura de directorios adecuada	Las herramientas de desarrollo utilizadas fueron adecuadas para el proyecto	Las historias de usuario del proyecto cumplen con el criterio SMART (Specific, Measurable, Achievable, Relevant, Time-boundeds)	La implementación programática de normas de accesibilidad es adecuada	Los métodos de testing utilizados son adecuados	Considera útil que este tipo de aplicaciones sea desarrollada para la población con discapacidad visual	En su experiencia como desarrollador de software, considera necesario el desarrollo de este tipo de aplicaciones
0	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo
1	Totalmente de acuerdo	Totalmente de acuerdo	De acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo	Totalmente de acuerdo

Y un DataFrame con las preguntas de tipo 2

```
In [ ]: negative_stats = df[negatives]
negative_stats
```

Out[ ]:

	Agregar nuevas funcionalidades al proyecto resultaría complejo y costoso	El código fuente presenta repetición de código innecesaria	La aplicación experimenta problemas de rendimiento o tiempos de respuesta lentos en ciertas situaciones.	La aplicación tarda mucho tiempo en iniciar desde cero	La elección del tipo de bases de datos es injustificada	La aplicación presenta problemas de seguridad	La aplicación posee una interfaz gráfica poco intuitiva	La metodología de desarrollo es aplicada de manera inconsistente
0	Neutral	Totalmente en desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo	En desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo

1	Totalmente en desacuerdo	En desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo	En desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo
2	De acuerdo	En desacuerdo	Totalmente en desacuerdo	Totalmente en desacuerdo	Totalmente de acuerdo	Totalmente en desacuerdo	De acuerdo	En desacuerdo

Luego, se definen los valores que tendrán cada respuesta dependiendo del tipo de pregunta

```
In [ ]: positive_answer_map = {
    "Totalmente de acuerdo": 4,
    "De acuerdo":3,
    "Neutral":2,
    "En desacuerdo":1,
    "Totalmente en desacuerdo":0,
}

negative_answer_map = {
    "Totalmente de acuerdo": 0,
    "De acuerdo":1,
    "Neutral":2,
    "En desacuerdo":3,
    "Totalmente en desacuerdo":4,
}
```

Y se aplica en ambos DataFrame

```
In [ ]: positive_stats_numeric = positive_stats.applymap(lambda x: positive_answer_map.get(x))
positive_stats_numeric
```

```
Out[ ]:
```

	El proyecto sigue una metodología de desarrollo de software clara	El proyecto cuenta con una estructura de directorios adecuada	Las herramientas de desarrollo utilizadas fueron adecuadas para el proyecto	Las historias de usuario del proyecto cumplen con el criterio SMART (Specific, Measurable, Achievable, Relevant, Time-boundeds)	La implementación programática de normas de accesibilidad es adecuada	Los métodos de testing utilizados son adecuados	Considera útil que este tipo de aplicaciones sea desarrollada para la población con discapacidad visual	En su experiencia como desarrollador de software, considera necesario el desarrollo de este tipo de aplicaciones
0	4	4	4	4	4	4	4	4
1	4	4	3	4	4	4	4	4
2	4	4	3	4	3	3	3	3

```
In [ ]: negative_stats_numeric = negative_stats.applymap(lambda x: negative_answer_map.get(x))
negative_stats_numeric
```

Out[ ]:

	Agregar nuevas funcionalidades al proyecto resultaría complejo y costoso	El código fuente presenta repetición de código innecesaria	La aplicación experimenta problemas de rendimiento o tiempos de respuesta lentos en ciertas situaciones.	La aplicación tarda mucho tiempo en iniciar desde cero	La elección del tipo de bases de datos es injustificada	La aplicación presenta problemas de seguridad	La aplicación posee una interfaz gráfica poco intuitiva	La metodología de desarrollo es aplicada de manera inconsistente
0	2	4	4	4	4	3	4	4
1	4	3	4	4	3	4	4	4
2	1	3	4	4	0	4	1	3
3	2	3	3	3	3	3	3	3
4	1	3	3	4	3	3	4	4

Posteriormente, se concatenan ambos DataFrame, obteniendo el DataFrame con los datos mapeados de todas las preguntas

```
In [ ]: combined_stats = pd.concat([positive_stats_numeric, negative_stats_numeric], axis = 1)
combined_stats
```

Out[ ]:

	El proyecto sigue una metodología de desarrollo de software clara	El proyecto cuenta con una estructura de directorios adecuada	Las herramientas de desarrollo utilizadas fueron adecuadas para el proyecto	Las historias de usuario del proyecto cumplen con el criterio SMART (Specific, Measurable, Achievable, Relevant, Time-bounded)	La implementación programática de normas de accesibilidad es adecuada	Los métodos de testing utilizados son adecuados	Considera útil que este tipo de aplicaciones sea desarrollada para la población con discapacidad visual	En su experiencia como desarrollador de software, considera necesario el desarrollo de este tipo de aplicaciones
0	4	4	4	4	4	4	4	4
1	4	4	3	4	4	4	4	4
2	4	4	3	4	3	3	3	3
3	4	4	4	3	3	3	4	4
4	4	3	4	4	4	3	4	4

Finalmente, se calcula la sumatoria de todos los valores

```
In [ ]: total_sum = combined_stats.sum().sum()  
total_sum
```

```
Out[ ]: 276
```

La cual se compara con la mayor puntuación posible

```
In [ ]: rows, cols = combined_stats.shape  
max_value = rows*cols*4  
max_value
```

```
Out[ ]: 320
```

Obteniendo el puntaje final que indica el nivel de usabilidad percibido por los usuarios en general

```
In [ ]: average = total_sum/max_value  
average*100
```

```
Out[ ]: 86.25
```

## Anexo I. Carta de entrega del desarrollo tecnológico



San Juan de Pasto, 22 de Abril de 2024

Magister  
**MAGDA MIREYA SALAZAR SUAREZ**  
Programa de Ingeniería de Sistemas  
Directora

Cordial Saludo de paz y bien

Por la presente, me dirijo a ustedes en calidad de docente del programa de Mercadeo de nuestra institución para comunicar oficialmente que el estudiante Carlos Alberto Botina Carpio, identificado con CC. 1004234727 ha completado satisfactoriamente el desarrollo y entrega de la aplicación requerida como parte del proyecto de investigación titulado "Diseño de material didáctico para personas con discapacidad visual en la ciudad de Pasto".

La aplicación, diseñada para atender las necesidades de accesibilidad de las personas con discapacidad visual, ha sido entregada dentro del plazo establecido y cumple con todos los requisitos y especificaciones establecidos en el proyecto. Durante el proceso de desarrollo, el estudiante ha demostrado un alto nivel de compromiso, dedicación y competencia técnica, lo cual ha sido fundamental para el éxito de la iniciativa.

Asimismo, deseo expresar mi agradecimiento al Comité de Investigación de Ingeniería de Sistemas por su continuo apoyo y orientación durante la ejecución de este proyecto. Su retroalimentación y sugerencias han sido de gran valor para el estudiante y para mí como supervisor del proyecto.

Quedo a su disposición para cualquier consulta adicional que puedan tener sobre este proyecto o sobre el desempeño del estudiante.

Atentamente,

Edgar Mauricio Salas Leiva  
Programa de Mercadeo  
Docente  
ensalas@umariana.edu.co  
3016571285

## **Anexo J. Otros enlaces**

- Repositorio de la aplicación (GitHub): <https://github.com/cbotina/Luciapp>
- Enlace de la aplicación (PlayStore):  
<https://play.google.com/store/apps/details?id=co.edu.umariana>