



Facultade de Informática

UNIVERSIDADE DA CORUÑA

TRABALLO FIN DE GRAO
GRAO EN ENXEÑARÍA INFORMÁTICA
MENCIÓN EN TECNOLOXÍA DE COMPUTADORES

Aplicación móbil de cuentos personalizados para niños con TEA

Estudiante: Álvaro Sánchez González
Dirección: Óscar Fresnedo Arias
Paula María Castro Castro

A Coruña, setembro de 2021.

A mi familia y amigos

Agradecimientos

Primero quiero agradecer a Óscar Fresnedo y Paula María Castro toda la ayuda que me han brindado a lo largo de todo este proyecto. También quiero agradecer a toda mi familia y amigos, por haberme acompañado durante este trayecto, y por haberme ayudado y apoyado en todo momento.

Resumen

El objetivo de este Trabajo de Fin de Grado es desarrollar una aplicación móvil enfocada a ayudar a personas con el Trastorno del Espectro Autista y otros trastornos del neurodesarrollo. Se trata de una aplicación de lectura de cuentos, desarrollada en Flutter para Android e iOS, con la que estas personas podrán practicar y mejorar aspectos en los que habitualmente presentan dificultades, como la habilidad lectora o su comprensión. Estos cuentos son personalizables y adaptados a los lectores y su progreso.

Abstract

The objective of this Final Degree Project is to develop a mobile application focused on helping people with Autism Spectrum Disorder and others neurodevelopmental disorders. It is a stories application, developed in Flutter for Android and iOS, with which these people can practice and improve aspects in which they usually have difficulties, such as reading skills or comprehension. These stories are customizable and adapted to the readers and their progress.

Palabras clave:

- Trastorno del Espectro Autista
- Aplicación móvil
- Aprendizaje-servicio
- Scrum
- Flutter
- Firebase

Keywords:

- Autism spectrum disorder
- Mobile application
- Service-learning
- Scrum
- Flutter
- Firebase

Índice general

1	Introducción	1
1.1	Motivación	1
1.2	Objetivos	2
1.2.1	Visión general de la aplicación	2
1.3	Estructura de la memoria	4
2	Trastornos del neurodesarrollo	5
2.1	Trastornos de la comunicación	5
2.2	Trastornos del aprendizaje	6
2.3	Trastornos por déficit de atención e hiperactividad	6
2.4	Trastorno del Espectro Autista	7
2.4.1	Introducción histórica	7
2.4.2	Detección y diagnóstico	8
2.4.3	Características principales y capacidades	9
2.4.4	Accesibilidad y tecnologías	11
2.4.5	Tipos de TEA	11
3	Estado del arte	13
3.1	DictaPicto	13
3.2	PictoTEA	14
3.3	José Aprende	15
3.4	TEAyudo a Jugar	15
3.5	Pictocuentos	16
3.6	Conclusiones	17
4	Fundamentos y herramientas tecnológicas	19
4.1	Flutter	19
4.2	Dart	20

4.3	Firebase	20
4.3.1	Firebase Authentication	20
4.3.2	Cloud Firestore	20
4.3.3	Cloud Storage	21
4.3.4	Cloud Functions	21
4.4	Pictogramas de ARASAAC	21
4.5	Android	21
4.6	Android Studio	22
4.7	iOS	22
4.8	XCode	22
4.9	Git	22
4.10	SourceTree	23
4.11	Microsoft Project	23
4.12	LATEX	23
4.13	Balsamiq Wireframes	23
4.14	DrawIO	24
5	Metodología de desarrollo	25
5.1	Scrum	25
5.2	Equipo Scrum	26
5.2.1	Product Owner	26
5.2.2	Scrum Master	26
5.2.3	Equipo de desarrollo	26
5.3	Eventos de Scrum	26
5.3.1	Sprint	26
5.3.2	Planificación del sprint	27
5.3.3	Daily	27
5.3.4	Revisión del sprint	27
5.3.5	Retrospectiva del sprint	28
5.4	Artefactos de Scrum	28
5.4.1	Product Backlog	28
5.4.2	Sprint Backlog	28
5.4.3	Incremento	28
5.5	Aplicación de la metodología en este proyecto	29
5.6	Apendizaje-servicio (ApS)	29

6	Planificación y seguimiento	31
6.1	Planificación	31
6.1.1	Planificación inicial	31
6.2	Desarrollo del proyecto	32
6.2.1	Seguimiento	34
6.3	Recursos	34
6.4	Costes	35
7	Análisis	37
7.1	Actores	37
7.2	Requisitos	38
7.2.1	Requisitos funcionales	38
7.2.2	Requisitos no funcionales	40
7.3	Casos de uso	41
7.3.1	Casos de uso del usuario no registrado	41
7.3.2	Casos de uso del tutor/educador	41
7.3.3	Casos de uso del tutor	47
7.3.4	Casos de uso del usuario (niño/a)	47
8	Diseño	49
8.1	Patrón de diseño	49
8.2	Diseño de la vista	50
8.2.1	Material Design	50
8.2.2	Interfaz de usuario	50
8.3	Diseño lógico de la base de datos	51
8.3.1	Colección de tutores	51
8.3.2	Colección de educadores	53
8.3.3	Colección de Usuarios	54
8.3.4	Colección de Cuentos	54
8.3.5	Subcolección de páginas	54
8.3.6	Subcolección de palabras	55
8.4	Estructura del proyecto	56
9	Implementación y pruebas	59
9.1	Implementación de las diferentes funcionalidades donde usamos Firebase	59
9.2	Uso del API ARASAAC	62
9.3	Uso de imágenes del propio dispositivo	63
9.4	Implementación de gestión de usuarios	65

9.5	Implementación de gestión de cuentos	66
9.5.1	Creación de cuentos	66
9.5.2	Edición de cuentos	70
9.5.3	Borrado de cuentos	71
9.6	Implementación de la lectura de cuentos	71
9.7	Pruebas	74
9.7.1	Pruebas funcionales	74
9.7.2	Pruebas de revisión de <i>sprint</i>	75
9.7.3	Pruebas no funcionales	76
10	Conclusiones	77
10.1	Conclusiones	77
10.2	Lecciones aprendidas	78
10.3	Líneas futuras	78
A	Acta de reunión inicial	83
A.1	Resumen del acta de reunión inicial	83
B	Acta de reunión de seguimiento	85
C	Acta de reunión final	87
D	Material adicional	91
	Lista de acrónimos	99
	Bibliografía	101

Índice de figuras

3.1	Pantallas de DictaPicto	13
3.2	Pantallas de PictoTEA	14
3.3	Pantallas de José Aprende	15
3.4	Pantallas de TEAyudo a Jugar	16
3.5	Pantalla de lectura de PictoCuentos	17
6.1	Esquema temporal de la planificación	32
7.1	Casos de uso del usuario no registrado	41
7.2	Casos de uso para gestión de sesión (Tutor y Educador)	44
7.3	Casos de uso en la pantalla principal de la aplicación (Tutor y Educador)	44
7.4	Casos de uso de la gestión de cuentos (Tutor y Educador)	44
7.5	Casos de uso de la edición de cuentos (Tutor y Educador)	45
7.6	Casos de uso de la creación de cuentos (Tutor y Educador)	46
7.7	Casos de uso de la gestión de usuarios niño/a (Tutor)	47
7.8	Casos de uso de la lectura de cuentos (Niño/a)	48
8.1	Ejemplo gráfico del patrón de diseño MVVM	50
8.2	Esquema de datos de la aplicación	52
8.3	Estructura y organización del proyecto	57
9.1	Funcionalidad de selección de imágenes o búsquedas de pictogramas	64
9.2	Formulario de registro en la aplicación	65
9.3	Pantallas principales del tutor y del educador	67
9.4	Pantallas con los diferentes pasos de la creación	70
9.5	Pantallas de edición desde la vista de lectura.	71
9.6	Pantallas del niño/a para seleccionar género y cuentos	72
9.7	Pantalla de lectura del cuento	73

D.1	<i>Mockup</i> de la vista de Registro	91
D.2	<i>Mockup</i> de la vista de Inicio de Sesión	92
D.3	<i>Mockup</i> de la vista de Añadir niño (<i>Tutor</i>)	92
D.4	<i>Mockup</i> de la vista de Asignar Responsable a un niño/a (<i>Tutor</i>)	93
D.5	<i>Mockup</i> de la vista de la pantalla principal de un usuario <i>Educador</i>	94
D.6	<i>Mockup</i> de la vista de Lectura de cuentos	95
D.7	<i>Mockup</i> de la vista de Creación de Cuentos. Segunda versión.	96
D.8	<i>Mockup</i> de la vista de Creación de cuentos. Primera versión.	96
D.9	<i>Mockup</i> de la vista de Edición de Cuentos	97
D.10	<i>Mockup</i> de la vista de Borrado de Cuentos	97

Índice de tablas

3.1	Tabla comparativa de las aplicaciones del mercado	18
6.1	Costes de los recursos humanos	36
6.2	Costes totales de los recursos humanos	36
6.3	Costes de los recursos materiales	36
6.4	Coste total del proyecto	36
7.1	Actor Tutor	37
7.2	Actor Educador	38
7.3	Actor Usuario	38
7.4	Actor Usuario no registrado	38
8.1	Atributos de la colección <i>Tutors</i>	53
8.2	Atributos de la colección <i>Educators</i>	53
8.3	Atributos de la colección <i>Users</i>	54
8.4	Atributos de la colección <i>Stories</i>	55
8.5	Atributos de la subcolección <i>Pages</i>	55
8.6	Atributos de la subcolección <i>Words</i>	56

Introducción

Los cuentos son un aspecto fundamental de la infancia de cualquier niño y niña. Por lo tanto, en este trabajo se ha querido adaptar este recurso al mundo tecnológico actual, creando una aplicación de cuentos personalizados, llamada **Cuentea**. El objetivo principal del trabajo está enfocado a desarrollar las diversas dificultades que presentan las personas con **Trastorno del Espectro Autista (TEA)** u otras diversidades funcionales del neurodesarrollo. El **TEA** se suele manifestar a una edad muy temprana, de dos a tres años. Por lo que el cuento es un recurso muy útil con el que se puede trabajar la comunicación, la comprensión lectora y el desarrollo social, características en las cuales estas personas suelen presentar carencias. La principal ventaja de una aplicación móvil frente a los cuentos físicos es la capacidad de creación de nuevos cuentos y personalizarlos teniendo en cuenta las necesidades del niño o la niña.

Considerando este fin social, se aplicará la metodología **Aprendizaje-Servicio (ApS)**, que integrará un aprendizaje académico del alumno con un servicio a la comunidad, como es en este caso la ayuda a personas con diferentes diversidades funcionales y dificultades en el desarrollo. Por lo tanto, con esta aplicación de lectura de cuentos se realiza un gran aporte social a esta comunidad, proporcionándoles un servicio a los educadores y educadoras para trabajar el desarrollo de estas personas, con el que poder llegar a mejorar su calidad de vida. Al mismo tiempo, el estudiante podrá adquirir conocimientos sobre la realización de aplicaciones adaptadas a las necesidades de estas personas.

1.1 Motivación

En los últimos años ha aumentado considerablemente el número de personas diagnosticadas con el **TEA** y se disponen de limitados recursos para trabajar con ellos. Por ello, se ha decidido apoyarlas con este proyecto, tanto a ellas como a sus familiares y especialistas. La personalización es la propiedad clave de este trabajo, dado que permite adaptar cada cuento

o historia a cualquier persona, dependiendo de sus dificultades, con lo que les proporcionará motivaciones con elementos con los que se sienten familiarizados, como los pictogramas. Por otro lado, facilitará la interacción de los especialistas y personas cercanas mediante una aplicación flexible y accesible para cualquier usuario.

En lo que concierne a la motivación personal, el alumno seleccionó un proyecto social para poder conocer las diferentes dificultades con las que se enfrentan estas personas en su día a día y sus familias, y con ello poder llegar a realizar un aporte a la comunidad. Por último, en la misma medida y siguiendo la metodología ApS, este proyecto se enfocó en aumentar los conocimientos del estudiante sobre el desarrollo de aplicaciones móviles, para poder ampliar su mercado laboral en un futuro cercano.

1.2 Objetivos

El objetivo principal del proyecto es la creación de una aplicación de lectura de cuentos enfocada y adaptada para personas que presentan varias dificultades en el desarrollo, centrándose sobre todo en personas de temprana edad afectadas por el TEA. Dentro del mismo encontramos objetivos más concretos, que se tuvieron en cuenta para el desarrollo:

- Desarrollar el aprendizaje, las habilidades comunicativas y sociales de estas personas, mediante recursos visuales, como los pictogramas, y auditivos, como la reproducción de palabras y eventos de forma comprensible y sencilla para ellos.
- Analizar y comprender los diferentes aspectos que debe tener una aplicación enfocada a personas con estas diversidades funcionales.
- Proporcionar un recurso de ayuda para los familiares y especialistas, con una personalización singular para cada niño y niña.
- Contribuir al desarrollo e investigación para mejorar las condiciones y calidad de vida de estas personas.
- Familiarizarse con las tecnologías actuales para el desarrollo de aplicaciones móviles.

1.2.1 Visión general de la aplicación

Para desarrollar estos objetivos se expondrá la visión general que debe tener nuestro trabajo y qué debe cumplir. La aplicación estará dividida en tres partes principales, las dos primeras serán muy similares, que son la del **tutor** y la del **educador** y, por último, la del **usuario o niño/a**.

Por un lado, los perfiles de **tutor** y **educador** representan el rol principal de la aplicación, encargándose de gestionar los cuentos y de asignarlos a los niños.

- Realizarán la creación personalizada de los cuentos para su correcta adaptación a cada niño, y la edición o borrado.
- Para la creación del cuento y sus diferentes páginas se realizará la importación de un documento con un formato específico pero sencillo, para evitar escribir un cuento extenso en un dispositivo móvil, lo que puede ser incómodo y laborioso.
- Tanto en la creación como en la edición se permitirá la asignación de pictogramas o imágenes a las palabras, para realizar un aporte visual en la posterior lectura.
- Ambos usuarios deben disponer de un registro e inicio de sesión seguro dentro de la aplicación.

La gestión de usuarios o niños/as se englobará en un único perfil, el del **tutor**. Esta acción se realiza por motivos de seguridad y para delegar la información sensible de un niño o niña en un único usuario, en este caso, un familiar o un tutor. Esta parte de la aplicación debe cumplir los siguientes objetivos:

- Dar de alta un niño o una niña en el sistema para su posterior asignación de cuentos, y que pueda realizar la lectura.
- Asignación controlada de los niños o las niñas a otros tutores o especialistas que estén dados de alta dentro del sistema.

Por último, la parte fundamental de proyecto es la lectura de cuentos, que es la que realizarán los **niños** y las **niñas** que utilicen la aplicación. Por eso, debemos tener en cuenta las siguientes características:

- La lectura debe contar con un diseño adaptado hacia estas personas.
- Los cuentos contemplarán la opción de lectura libre y de audio-lectura, para mejorar así su comprensión lectora.
- La selección de cuentos y géneros mostrará pictogramas o imágenes atractivas, para que los niños/as realicen la elección que más les interese.
- La lectura de cuentos se apoyará en recursos visuales, como los pictogramas, y auditivos, mediante la reproducción de audio claro y pausado.

Todos estos objetivos y características se han tenido en cuenta a lo largo de todo el desarrollo del proyecto, sobre todo en la fase de análisis y definición de requisitos.

1.3 Estructura de la memoria

En esta sección se explicarán de manera concisa los diferentes apartados que componen la memoria. Se divide en los siguientes capítulos:

- **Introducción.** El primer apartado del Trabajo de Fin de Grado (TFG), que introduce el tema a tratar.
- **Trastornos del neurodesarrollo.** Capítulo donde se explicarán los diferentes trastornos en los que se centra la aplicación, y en el que se detallará con mayor profundidad el TEA.
- **Estado del arte.** Capítulo que recoge aquellas aplicaciones disponibles similares a la desarrollada en este trabajo, comparándolas y tratando diversos aspectos de las mismas.
- **Fundamentos y herramientas tecnológicas.** Explicación de las herramientas utilizadas para el desarrollo de este trabajo.
- **Metodología de desarrollo.** Descripción de la metodología aplicada en el desarrollo del TFG y la metodología de aprendizaje, ApS.
- **Planificación y seguimiento.** Capítulo donde se expondrá la planificación y desarrollo completo del proyecto, los costes del mismo y los recursos empleados.
- **Análisis.** Exposición del análisis de requisitos, donde veremos las diferentes funcionalidades que debe tener la aplicación.
- **Diseño.** Capítulo donde se muestra el diseño detallado del proyecto y se exponen los bocetos realizados previamente.
- **Implementación y pruebas.** Capítulo donde se explican los detalles más significativos sobre la implementación de la aplicación. En este capítulo se explicarán las pruebas realizadas y se comentarán las aportaciones de los usuarios que hicieron uso de la misma.
- **Conclusiones.** Capítulo final donde se exponen las conclusiones del trabajo y se analizan las líneas futuras del mismo.

Trastornos del neurodesarrollo

EN este apartado se describirán los diferentes trastornos del neurodesarrollo para los que puede ser de utilidad nuestra aplicación. Es importante destacar que el diseño de la aplicación está pensado para personas con TEA. Sin embargo, muchos de los trastornos que se comentan en este capítulo comparten ciertas características relacionadas con problemas en la comprensión y generación del lenguaje oral y/o escrito, además de otras relacionadas con el desarrollo cognitivo. Por esta razón, es importante conocer bien estas características para realizar un desarrollo adaptado de la aplicación que resulte útil en el contexto en el que se ha planteado.

Los trastornos a los que hemos orientado nuestra aplicación son los siguientes:

2.1 Trastornos de la comunicación

El lenguaje es un sistema complejo y dinámico que forma parte del desarrollo general de cualquier persona e incide fuertemente en su calidad de vida [1]. La detección de la no aparición del lenguaje o del retraso en su adquisición suele darse en la edad temprana de los dos o tres años, por norma general antes de los cuatro. El retraso del lenguaje se basa en la demora en la aparición o desarrollo de todos los niveles del lenguaje (fonológico, morfosintáctico, semántico y pragmático), que afecta sobre todo a la expresión y a la comprensión. Según el [Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition \(DSM-5\)](#), estos trastornos se dividen en los siguientes tipos [2]:

- Trastorno del lenguaje.
- Trastorno del habla o fonológico.
- Trastorno de la fluencia de inicio en la infancia (tartamudeo).
- Trastorno de la comunicación pragmática.

- Trastorno de la comunicación no especificado.

En este trabajo se intenta ayudar a mejorar estas dificultades mediante el uso de una reproducción de audio clara y pausada del texto de los cuentos. Además, la aplicación ofrece la posibilidad de repetir lo que se ha escuchado con el objetivo de mejorar su comunicación y su forma de expresarse.

2.2 Trastornos del aprendizaje

Los **Trastornos de Aprendizaje (TA)** que afectan al desarrollo son los más extendidos en la población, llegando a afectar a alrededor del 10% de las personas en edad escolar [3]. Los **TA** afectan a aspectos básicos del aprendizaje como son la lectura, el cálculo y/o la atención. Por lo general, con una detección y diagnóstico tempranos se pueden abordar estos problemas, mediante metodologías o herramientas que permitan a estas personas continuar con sus estudios. Sin embargo, si se realiza un manejo inadecuado podría desembocar en fracaso escolar. Estos trastornos, según **DSM-5** [2], se engloban en un único tipo, **Trastorno Específico del Aprendizaje**.

Por lo tanto, la herramienta desarrollada también está enfocada a tratar este tipo de trastornos. La posibilidad de personalizar los cuentos permite reforzar el aprendizaje de la persona en aquellas áreas donde sea necesario. Además, el uso de pictogramas permite mejorar la comprensión de conceptos o estructuras del lenguaje ya que le ayudan a relacionar las palabras que aparecen en el cuento con la realidad a través de un estímulo visual.

2.3 Trastornos por déficit de atención e hiperactividad

El **Trastorno por Déficit de Atención e Hiperactividad (TDAH)** es una afección crónica que afecta a muchos niños y a menudo continúa en la edad adulta [4]. Las características principales del mismo son la falta de atención y el comportamiento hiperactivo-impulsivo que, en determinadas ocasiones, puede motivar que la persona tenga una baja autoestima y un bajo rendimiento escolar. Se suele detectar/diagnosticar en una edad temprana anterior a los doce años y, en algunos casos, se puede dar a partir de los tres años. El **TDAH** se suele dividir en tres subtipos: **falta de atención predominante**, **conducta hiperactiva/impulsiva predominante** y **combinado**, una mezcla de los dos comportamientos anteriores. Generalmente, se suele utilizar algún tipo de tratamiento que ayuda a paliar estos síntomas que, habitualmente, se basa en medicamentos para la conducta.

De esta forma, también se puede hacer uso de herramientas tecnológicas que presentan un estímulo suficientemente atractivo para mantener la atención de estas personas. Estas herra-

mientas apoyan el aprendizaje en el entorno educativo y familiar, y pueden ayudar a mejorar la conducta y a controlar su hiperactividad manteniéndoles entretenidos y ocupados con un elemento llamativo para ellos.

2.4 Trastorno del Espectro Autista

El TEA es un trastorno del neurodesarrollo de origen neurobiológico que se manifiesta en la infancia y que afecta a la configuración del sistema nervioso y al funcionamiento del cerebro, llevando a la aparición de dificultades en el desarrollo de la comunicación e interacción social y en la conducta, con la presencia de comportamientos e intereses repetitivos y restringidos [5]. El TEA tiene unas características propias y definitorias que se manifiestan de forma heterogénea a lo largo del ciclo de vida de cada persona, por lo que sus manifestaciones y necesidades cambiarán en función de su desarrollo y experiencias. Existe una gran variabilidad en las personas que manifiestan este trastorno, por ello, es muy necesario un tratamiento especializado y personalizado para cada una de ellas, con el objetivo de detectarlo y abordarlo lo antes posible. Dentro de estas terapias son fundamentales los recursos de apoyo, como el desarrollado en este proyecto.

2.4.1 Introducción histórica

En 1908, **Eugen Beuler**, un psiquiatra suizo, fue el primero en acuñar el término **autismo**. Beuler utilizaba este término para referirse a los pacientes que presentaban un comportamiento diferente al de los demás que sufrían la enfermedad con la que él trabajaba, la esquizofrenia. El término viene del griego y significaba “autos”: uno mismo e “ismos”: modo de estar, refiriéndose al hecho de estar encerrado en uno mismo [6].

Grunya Efimovna Sukhareva fue la primera psiquiatra infantil en realizar una descripción detallada de las características del autismo en 1925. También fue la primera en ofrecer un tratamiento educativo, familiar y sistemático a estas personas, aparte del psiquiátrico. Su definición de las características asociadas a este trastorno tuvo gran repercusión en el estudio futuro del mismo, llegando a que la definición actual en DSM-5 sea muy similar a la dada en su día por esta psiquiatra. Aunque realizó grandes avances, estos no tuvieron gran relevancia hasta veinte años después con las investigaciones más conocidas y destacadas de **Leo Kanner** y **Hans Asperger**.

De forma paralela y casi simultánea, **Leo Kanner** y **Hans Asperger** comenzaron sus estudios sobre el autismo.

Leo Kanner fue un psiquiatra que centró su investigación en once niños que presentaban diversas dificultades como: dificultad para desarrollar interacciones con las demás personas, en la adaptación a cambios de rutinas, en la adquisición del lenguaje, sensibilidad a los estímulos, ecolalias (imitar o repetir frases, sonidos o conversaciones de manera involuntaria)..., la mayoría de ellas equivalentes a las características del autismo conocido hoy en día. Estas investigaciones desembocaron en la defensa del autismo como cuadro específico y único.

Por otro lado, **Hans Asperger** observó que algunos niños presentaban rutinas repetitivas e intereses restringidos, pero no identificó dificultades en la comunicación, aunque sí un lenguaje centrado en un único tema de interés. También se declaró como un gran defensor de la inclusión social, educativa y laboral de estas personas.

Las investigaciones y estudio de características de estos dos últimos autores significó un gran avance en este campo, llevando a considerar esta patología como una entidad diferente a cualquier otra. Además, ayudaron a delimitar una nuevo subtipo de trastorno dentro del espectro del autismo, conocido como **Síndrome de Asperger**.

2.4.2 Detección y diagnóstico

La detección del trastorno se relaciona con la falta de adquisición de varias habilidades esperadas en un niño o niña de una edad de 12, 18 y 24 meses [7]. Aunque se manifieste de forma diferente para cada niño o niña, existen una serie de patrones comunes en el desarrollo que sirven de referencia para identificar factores relacionados con el TEA. Si estas señales se dan de forma aislada no significa que tenga que ser un diagnóstico de autismo, pero sí se recomienda una evaluación especializada, ya que la importancia de un diagnóstico precoz es muy alta y significativa. También es importante destacar que la edad anteriormente citada es orientativa y que los signos se pueden dar en otras edades.

Las señales por las que se recomienda una evaluación especializada son las siguientes:

- **Antes de los 12 meses:** poco contacto visual, que no muestre anticipación a la hora de cogerlo en brazos y poco interés en juegos interactivos.
- **A los 12 meses:** falta de comunicación mediante balbuceos o gestos.
- **Entre los 12 y los 18 meses:** no contestar cuando lo llaman por su nombre, que tenga una respuesta inusual ante determinados estímulos auditivos o que no señale objetos que deberían atraer su interés.
- **A los 18 meses:** no expresar palabras sencillas.

- **Entre los 18 y los 24 meses:** retraso en el desarrollo del lenguaje, falta de imitación o falta de interés en relacionarse con otros niños o niñas.
- **A los 24 meses:** no formular frases de dos o más palabras.

Como se ha mencionado anteriormente, un diagnóstico precoz es la clave para realizar una atención temprana y específica, para favorecer el desarrollo de estas personas y el bienestar emocional de sus familias. Dado que no es posible un diagnóstico a través de marcadores biológicos, es muy importante el apoyo mediante el desarrollo de herramientas y facilidades que favorezcan una valoración especializada.

2.4.3 Características principales y capacidades

Como ya se ha mencionado en el apartado 2.4.2, las manifestaciones del TEA varían mucho dependiendo de la persona. Sin embargo, todas comparten algunos aspectos en áreas comunes del desarrollo: la comunicación, la interacción social, y la flexibilidad del comportamiento y del pensamiento [5].

Habilidades de comunicación

Esta dificultad se puede manifestar a nivel de expresión, de comprensión o de ambos a la vez. Por ello, los subdividiremos en:

- **Comunicación verbal:** algunas personas con TEA, aunque puedan hablar adecuadamente, suelen presentar dificultades a la hora de hacerlo en un entorno social de manera adecuada. Otras personas no desarrollan lenguaje oral y precisan del apoyo de **Sistemas Alternativos y/o Aumentativos de Comunicación (SAAC)**. Por esto, es muy importante el uso de los pictogramas, ya que les proporciona una forma de comunicarse y de entender las diferentes situaciones ante las que se encuentran.
- **Comunicación no verbal:** por otro lado, estas personas también pueden tener problemas a la hora de entender gestos, expresiones faciales o corporales de otras personas, dificultándoles una comunicación e interacción adecuada.

Habilidades para la interacción social

Las personas con TEA pueden tener dificultades a la hora de relacionarse con los demás, desenvolverse en determinados momentos o comprender la situación y el entorno. Estas personas se suelen regir por una interacción social diferente, por no saber reaccionar ante determinadas situaciones como acercamientos inusuales, en los cuales tiene que iniciar una

conversación. En ocasiones, también presentan emociones o reacciones poco ajustadas o adecuadas a determinadas situaciones. Por ello, presentan bastantes inconvenientes en lo que se refiere a interacción social. Además, en determinados casos, no muestran interés por establecer o iniciar relaciones sociales con sus iguales y pueden reaccionar de forma negativa si otra persona intenta forzar esta interacción.

Flexibilidad del comportamiento y del pensamiento

Las personas con TEA suelen limitarse a intereses específicos o patrones de comportamiento rígidos y repetitivos. Por lo que pueden llegar a tener problemas a la hora de ajustar su forma de pensar y comportarse ante cambios o situaciones imprevistas. En el mismo sentido, pueden presentar dificultades para comprender indicaciones o situaciones poco estructuradas. Por otro lado, también pueden presentar alteraciones ante diferentes estímulos, reflejando un malestar o disgusto ante determinados sonidos, olores, luces intensas, colores, sabores o texturas.

Estas personas en muchas ocasiones pueden verse en situaciones en las que no saben como actuar o peligrosas por su falta de noción ante el peligro, por ello, son muy importantes las terapias y el trato con algún especialista, que les aporten ayuda para saber reaccionar de forma adecuada ante estos eventos. En este proceso se apoyan en recursos como pictogramas, como comentamos anteriormente, con los que recrean las situaciones de una forma sencilla, de tal forma que puedan llegar a entenderlas y adaptarse a ellas poco a poco.

Capacidades

Las personas con TEA no presentan únicamente dificultades dentro de sus características, sino que también se les asocian una serie de capacidades, como:

- **La meticulosidad.**
- **Gran curiosidad por temas muy específicos.**
- **Conocimiento especializado en estos temas.**
- **La sinceridad y la honestidad.**
- **Respeto y cumplimiento de reglas.**
- **Atención por detalles.**
- **Una buena adaptación y seguimiento de las rutinas.**
- **Buenas competencias en tareas mecánicas y repetitivas.**

- **Tendencias a ser muy lógicas.**
- **La capacidad de escuchar sin prejuicios.**

2.4.4 Accesibilidad y tecnologías

La accesibilidad universal es una cualidad de las aplicaciones que ayuda a que todas las personas comprendan, utilicen, participen y disfruten de los recursos con autonomía y comodidad [8]. Se divide en tres tipos: accesibilidad física, sensorial y cognitiva. En nuestro caso, nos centraremos únicamente en la accesibilidad cognitiva, ya que es la que afecta directamente a las personas con TEA.

La **accesibilidad cognitiva** es una cualidad que permite que las personas con dificultades de comprensión puedan utilizar los diferentes entornos que les rodean. Teniendo en cuenta las características de las personas con TEA, explicadas en el apartado 2.4.3, este tipo de accesibilidad les facilita adaptarse al mismo, ayudándoles a comprenderlo, utilizarlo y disfrutarlo de una forma sencilla. Existen una serie de estrategias y recursos que favorecen que un elemento sea accesible de una forma cognitiva, como por ejemplo, los sistemas de señalización o el diseño de las **Tecnologías de la Información y la Comunicación (TIC)** para todas las personas. Esta estrategia se detallará a continuación, ya que ha sido la que más se ha estudiado y analizado a la hora de desarrollar la aplicación.

Con el diseño de las TIC para todas las personas, se intenta crear software y hardware fácil de comprender y utilizar por personas que presenten diversas dificultades. Por eso, se han tenido en cuenta la mayoría de sus principios en el desarrollo de este trabajo, dado que buscamos que las personas con TEA o con otros tipos de diversidad funcional se encuentren cómodas, aprendan con la aplicación y que puedan realizar esta tarea con autonomía. Los diferentes aspectos que se han tenido en cuenta es que sea sencilla, intuitiva, predecible y adaptable. Que sea flexible es una cualidad que hemos tenido también muy en cuenta, dado que no todas las personas con TEA tienen los mismos intereses y muchas veces son muy específicos, con lo cual, se busca una aplicación personal que se adapte a las necesidades de cada usuario.

2.4.5 Tipos de TEA

Con la publicación del DSM-5 en 2013 [9], se modificó la clasificación y los criterios asociados al TEA, respecto a lo englobado en la anterior, que dividía este trastorno en: TEA común, síndrome de Asperger, síndrome de Heller y trastorno generalizado del desarrollo no especificado. El DSM-5 pasa a clasificar las características del TEA de la siguiente forma [9]:

- Deficiencias significativas en la comunicación e interacción social.
- Presencia de patrones de comportamiento, intereses y actividades restringidos y repetitivos.
- Los síntomas suelen presentarse en la etapa temprana de la infancia.
- El conjunto de síntomas crea dificultades en el desarrollo de su día a día.

Para facilitar su diagnóstico añade diferentes grados de severidad tanto para los síntomas relacionados con la comunicación social como para los síntomas relacionados con comportamientos restringidos y repetitivos. En concreto, se definen los siguientes grados de severidad o tipos de TEA [10]:

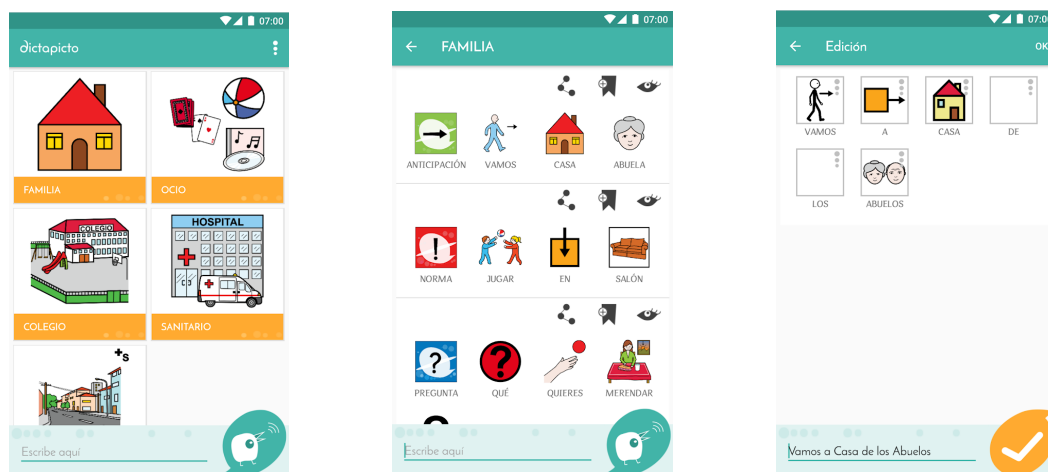
- **Grado 3, necesita ayuda muy notable:** presenta una mínima comunicación social, con inicio muy limitado de las interacciones sociales y respuesta mínima hacia la apertura social de otra persona. Por otro lado, suelen mostrar inflexibilidad de comportamiento, una extrema dificultad de adaptación a cambios en su vida diaria, limitándose a comportamientos restringidos y repetitivos, y dificultades para cambiar el foco de atención.
- **Grado 2, necesita ayuda notable:** muestra deficiencias notables en la comunicación verbal y no verbal, problemas sociales incluso con ayuda *in situ*, inicio limitado de conversaciones o interacciones y respuesta poco común ante la apertura de comunicación de otra persona. En lo referente al comportamiento, son los mismos que los del párrafo anterior, con la diferencia de que suelen tener una aparición inferior (o más leve) a la del anterior grado.
- **Grado 3, necesita ayuda:** presenta deficiencias en la comunicación social sin ayuda *in situ*. Puede parecer que tiene poco interés en las interacciones sociales, ya que tiene dificultades para iniciar estas interacciones y realiza respuestas atípicas al inicio de comunicación por parte de otras personas. En el comportamiento, muestra problemas para alternar actividades, para organizarse y planificarse por sí mismo, lo que dificulta su autonomía.

Estado del arte

ANTES de comenzar con el desarrollo de este trabajo, se analizaron las diferentes aplicaciones en el mercado actual similares a la nuestra. A continuación, se expondrán sus características y funcionalidades y, por último, se realizará una comparación general de las mismas con este trabajo.

3.1 DictaPicto

DictaPicto [11] es una aplicación que permite convertir el lenguaje oral en información visual. Fue desarrollada por BJ-Adaptaciones y Doble Equipo, y está pensada para ayudar a las personas que presentan TEA o a aquellas que utilizan pictogramas para mejorar su comunicación (ver figura 3.1).



(a) Pantalla de selección de tipos de pictogramas

(b) Pantalla de selección de pictogramas

(c) Pantalla de edición de la acción

Figura 3.1: Pantallas de DictaPicto

Ha sido desarrollada para Android e iOS y presenta diversas funcionalidades como: la traducción de voz a pictogramas, gestión de vocabulario, el reconocimiento de voz, la gestión de copias de seguridad y la personalización. Ésta presenta una gran accesibilidad y un sencillo uso, lo que facilita la participación e interacción del niño o la niña.

3.2 PictoTEA

PictoTEA [12] facilita la comunicación a personas que presentan **TEA**, **Trastorno Generalizado del Desarrollo (TGD)** o cualquier persona con problemas en las habilidades sociales y comunicativas [13]. Esta aplicación es gratuita y se ha desarrollado en Android con la colaboración del **Centro Aragonés para la Comunicación Aumentativa y Alternativa (ARASAAC)** [14], utilizando sus pictogramas para practicar y mejorar la comunicación. Una de sus características más desatacadas es su sencillez y que ofrece distintos niveles progresivos, donde se irán desbloqueando poco a poco nuevos pictogramas para facilitar el aprendizaje de la persona. Por otro lado, otro aspecto a tener en cuenta es la capacidad de poder añadir nuevos pictogramas creados por uno mismo (ver figura ??).

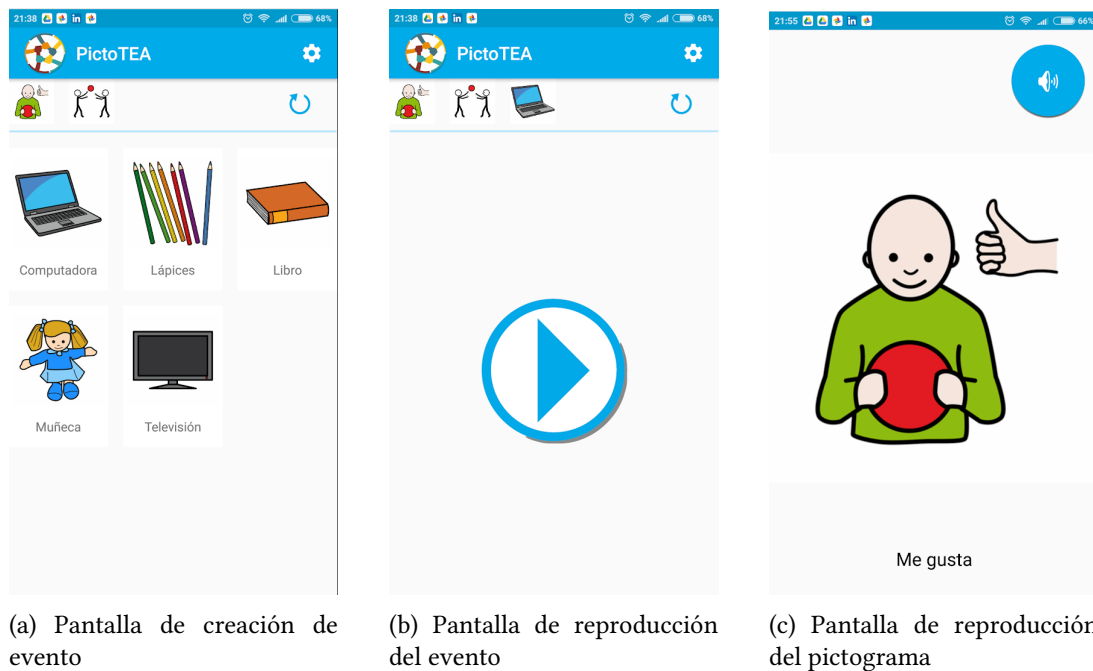


Figura 3.2: Pantallas de PictoTEA

3.3 José Aprende

José Aprende [15] es una aplicación móvil para la lectura de cuentos adaptados con pictogramas y desarrollados especialmente para niños con TEA resultando útil, al mismo tiempo, para personas con dificultades en el aprendizaje y para jóvenes lectores. Ha sido creada por Aprendizajes Visuales bajo una licencia Creative Commons para las plataformas de Android e iOS. Dispone de quince cuentos sencillos enfocados al aprendizaje de autonomía personal: auto-cuidados, rutinas y emociones (ver figura 3.3). Es un proyecto sin ánimo de lucro con el objetivo de crear herramientas que permitan desarrollar al máximo el potencial de cada usuario.

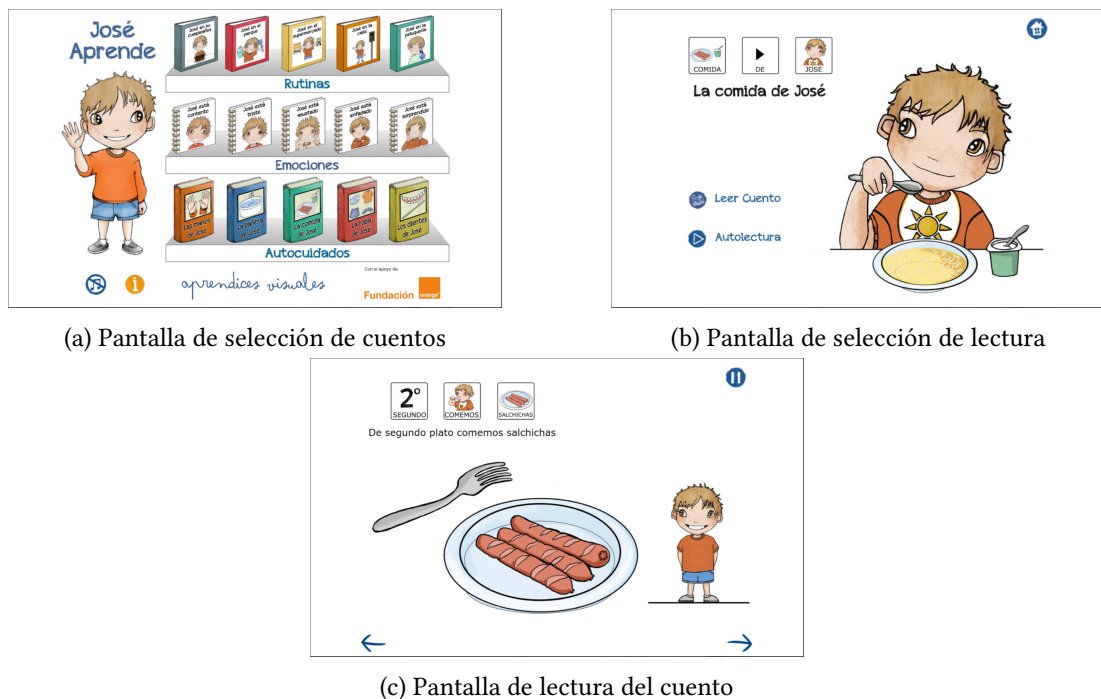
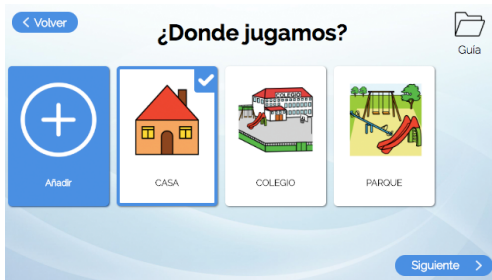


Figura 3.3: Pantallas de José Aprende

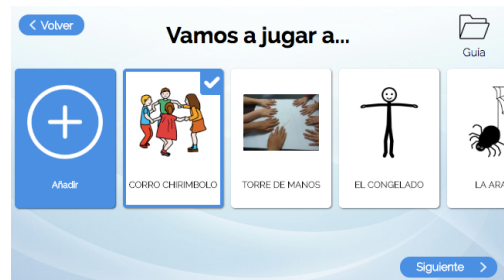
3.4 TEAyudo a Jugar

TEAyudo a Jugar está disponible para Android e iOS y tiene como objetivo la inclusión educativa y social de los niños. Por lo tanto, es una herramienta muy recomendable para personas con autismo u otras diversidades del neurodesarrollo. Ha sido creada en colaboración de la Universidad de Murcia y La Fundación Orange dentro de la convocatoria Soluciones tecnológicas aplicadas al autismo de 2016.

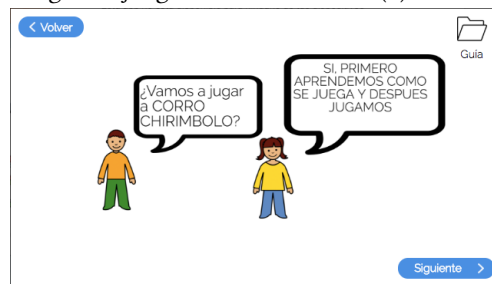
Jugar es considerada una de las actividades básicas del desarrollo infantil, por ello, esta aplicación busca mejorar la socialización y favorecer la inclusión social mediante este recurso [16]. De esta forma, la aplicación establece premisas claras a los niños para que sepan con quién van a jugar, dónde, cómo y cuándo (ver figura 3.4). Con estas indicaciones se puede estructurar el juego, haciéndolo más accesible a los usuarios y que de esta manera lleguen a aplicarlo en su día a día [13].



(a) Pantalla de selección de lugar de juego



(b) Pantalla de selección de juego



(c) Pantalla de representación del juego

Figura 3.4: Pantallas de TEAyudo a Jugar

3.5 Pictocuentos

La fundación PictoAplicaciones es una entidad sin ánimo de lucro que busca promover actividades de formación para personas con TEA que mejoren su calidad de vida. Una de estas herramientas es **Pictocuentos** [17], una página web gratuita de cuentos que ayudan a estimular y mejorar su lenguaje y, a su vez, buscan trabajar su comprensión lectora. Los cuentos se encuentran disponibles en varios idiomas y con un conjunto muy variado de pictogramas, animaciones, locuciones y música (ver figura 3.5). Esta aplicación es la que más se acerca a cumplir los objetivos de este TFG, pero en ella encontramos varias limitaciones, que comentaremos en el apartado 3.6.



Figura 3.5: Pantalla de lectura de PictoCuentos

3.6 Conclusiones

Tras analizar aplicaciones similares a la que se va a llevar a cabo en este trabajo, se ha observado que no existen demasiadas, específicamente de cuentos. Por ello, se han seleccionado aplicaciones con el mismo objetivo y características similares a la que se quiere desarrollar:

- En el caso de **PictoTEA** no comparte el mismo objetivo que el de nuestro trabajo, como es la mejora de comprensión lectora, sino más bien tiene como objetivo la creación y gestión de rutinas del día a día del niño/a. Por otro lado, el problema que nos encontramos es que la aplicación es poco llamativa y motivadora para los niños o niñas, y con ello podrían acabar prescindiendo de su uso. Algo similar pasa con **DictaPicto**, aunque esta herramienta presenta un alto grado de personalización para adaptarse a las necesidades y preferencias de los usuarios.
- **José Aprende** proporciona una serie de cuentos muy completos visuales y llamativos, que hacen uso de animaciones. También proporcionan la opción de audio-lectura para mejorar su comunicación. Su principal desventaja es que cuenta con un número fijo de cuentos que no se pueden modificar. Además, no permite trabajar otras áreas relacionadas con la autonomía, desarrollo cognitivo o social que no estén recogidas en los cuentos predeterminados.
- **TEAyudo a Jugar** completa la mayoría de limitaciones de las anteriores, pero está enfocada a crear juegos, para mejorar principalmente la interacción e inclusión social, y no a la lectura de cuentos como se busca con este proyecto.
- Por último, **Pictocuentos**, solventa la mayoría de los problemas comentados pero al ser una aplicación web no es tan manejable como una móvil. Lo que puede llegar a ser un

problema dado que la mayoría de asociaciones y educadores utilizan tabletas y dispositivos móviles por su comodidad y fácil transporte. Por otro lado, presenta un diseño que puede llegar a distraer al lector debido a la sobrecarga de elementos, animaciones y sonidos.

Nuestra aplicación busca suplir la mayoría de las carencias anteriormente identificadas. Por esta razón, el objetivo es permitir que los familiares, educadores y educadoras puedan crear diferentes tipos de cuentos con soportes visuales para trabajar diferentes áreas. Uno de los pilares de la aplicación es la posibilidad de crear cuentos más personales y específicos para cada persona en función de sus preferencias o necesidades. De esta forma, se busca darles la posibilidad a los responsables de crear cuentos y añadirles objetos o personas del entorno más cercano, mediante el uso de la cámara del dispositivo y la galería. Además se ha diseñado de una forma atractiva y motivadora, teniendo en cuenta las características de los potenciales usuarios (tanto personas con TEA u otros tipos de diversidad funcional). Se permitirá la modificación y adaptación del cuento, para una mayor interacción con los niños y niñas.

Otra de las características destacables de Cuentea es el alto grado de personalización a la hora de diseñar los cuentos que permite ir adaptándolos en función de los progresos del niño o la niña. En un principio la mayoría de palabras en el cuento se acompañarán de sus correspondientes pictogramas, y a medida que el usuario vaya mejorando, se retirarán poco a poco los apoyos visuales para adaptarse a su desarrollo.

Para ordenar todas estas ideas, en la tabla 3.1 se presenta un esquema de las diferentes características de las aplicaciones y de las conclusiones extraídas a partir de ellas.

Característica Aplicación	Uso de pictogramas	Diseño adaptado	Personalización	Lectura de cuentos	Sistema operativo
DictaPicto	✓	✗	✓	✓	Android e iOS
PictoTEA	✓	✗	✓	✗	Android
José Aprende	✓	✓	✗	✓	Android e iOS
TEAyudo a Jugar	✓	✓	✓	✗	Android e iOS
Pictocuentos	✓	✗	✗	✓	Aplicación Web

Tabla 3.1: Tabla comparativa de las aplicaciones del mercado

Como se observa no existe una aplicación actualmente que cumpla todos los objetivos definidos en el capítulo 1. En resumen, con este trabajo se intenta conseguir una mejor herramienta que cumpla todos estos objetivos y permita desarrollar la comprensión lectora, la comunicación y las habilidades sociales de estas personas y, con ello, mejorar su calidad de vida.

Fundamentos y herramientas tecnológicas

EN este capítulo se desarrollarán y analizarán brevemente los diferentes fundamentos y herramientas tecnológicas utilizadas a lo largo del proyecto.

4.1 Flutter

Flutter [18] es un *framework* de código abierto desarrollado por Google que permite crear código 100% nativo para aplicaciones móviles y web de una forma rápida y sencilla. Una de las mayores ventajas de este *framework* radica en que al disponer un código 100% nativo se van a poder desarrollar aplicaciones con un único proyecto. Se podrán implementar para los sistemas operativos de **Android**, **iOS**, **Mac**, **Windows**, **Google Fuchsia** y **web**. Esto supone una gran ventaja si se desea desarrollar una aplicación multiplataforma.

Las principales funcionalidades que ofrece esta herramienta son: la calidad nativa, nombrada en el anterior párrafo, una experiencia de usuario óptima dado que sus diseños son idénticos a los de Google (Material Design) y a los de Apple (Cupertino), tiempos de carga por debajo de un segundo y un desarrollo ágil y rápido, gracias a muchas de sus características.

Por lo tanto, teniendo en cuenta todas estas ventajas, Flutter fue el *framework* que se decidió utilizar para el desarrollo de este proyecto, ya que permite crear una aplicación tanto para el sistema operativo Android como para iOS. Esta característica permitirá llegar a un mayor número de usuarios y adaptarla a las diferentes asociaciones que disponen tanto de móviles de las dos plataformas como iPads para trabajar con estas personas.

4.2 Dart

Dart [19] es un lenguaje de programación desarrollado por Google y orientado a la creación de aplicaciones multiplataforma. Principalmente, se ha seleccionado este lenguaje porque es el utilizado en el desarrollo de aplicaciones en Flutter. Se trata de un lenguaje multiparadigma, aunque principalmente se sigue el de orientación a objetos, permitiendo así definir clases y tener un “recolector de basura”. Por lo tanto, tendrá una sintaxis muy similar a Java, Javascript y C++.

Dart se diseñó pensando en un desarrollo rápido y cómodo, y por ello, su máquina virtual y compilación *Just-in-Time* ejecutan los cambios en el código de manera inmediata [20] y nos permiten ver estos cambios al momento en el dispositivo donde se está probando la aplicación.

4.3 Firebase

Firebase [21] es una plataforma para el desarrollo de aplicaciones web, ubicada en la nube, que permite crear y sincronizar proyectos de forma sencilla. Al ser de Google también está integrada con Google Cloud Platform y preparada para funcionar con Flutter, el *framework* utilizado para el desarrollo de esta aplicación. Esta herramienta ha sido empleada en la implementación del *backend* de la aplicación, utilizando sus funcionalidades de autenticación y almacenamiento de usuarios, alojamiento de la base de datos del proyecto, almacenamiento de archivos y utilización de funciones en la nube. Estas tecnologías se expondrán a continuación.

4.3.1 Firebase Authentication

Firebase Authentication [22] permite almacenar y autenticar usuarios mediante varios servicios de una forma segura. Dispone de varias formas de autenticación, mediante contraseñas y correo, número de teléfono, proveedores de identidad federada (Google, Twitter, Facebook y Github), de manera anónima, y permite personalizar el proceso de autenticación con la base de datos del sistema de Firebase. En este trabajo se ha utilizado este servicio porque permite gestionar los usuarios de la aplicación de una forma sencilla y segura.

En el desarrollo de la aplicación se ha empleado el servicio de autenticación mediante correo y contraseña, almacenando la información de los usuarios a su vez en Firestore (ver sección 4.3.2) con un identificador único para poder establecer una diferencia clara entre los distintos usuarios.

4.3.2 Cloud Firestore

Cloud Firestore [23] es una base de datos *NoSQL*, flexible y escalable en la nube, que almacena y sincroniza datos para la parte tanto del cliente como del servidor. Almacena los

datos en forma de documentos que a su vez se organizan en colecciones. Este servicio realiza actualizaciones en tiempo real, manteniendo sincronizados todos los dispositivos conectados, y también permite realizar consultas para recuperar datos específicos o para recuperar documentos completos, pudiendo aplicar filtros en caso de que fuese preciso.

4.3.3 Cloud Storage

Cloud Storage [24] es un servicio de Firebase potente, sencillo y escalable, pensado para almacenar y entregar contenido generado por usuarios como, por ejemplo, audio, vídeo o fotos. En esta aplicación se ha utilizado para el almacenamiento y posterior obtención de las imágenes de la cámara y la galería que seleccione el usuario.

4.3.4 Cloud Functions

Cloud functions [25] es un *framework* de Firebase que permite ejecutar código *backend*, de una forma segura, en respuesta a peticiones [Hypertext Transfer Protocol \(HTTP\)](#). Este código se puede desarrollar en JavaScript o en Typescript y se almacenará en la nube. En este caso, se ha implementado en Javascript y será utilizado para realizar un borrado recursivo de colecciones almacenadas en la base de datos, Firestore, explicada en la sección 4.3.2.

4.4 Pictogramas de ARASAAC

ARASAAC [14] es un proyecto del Gobierno de Aragón bajo una licencia Creative Commons, que ofrece recursos para facilitar la comunicación y accesibilidad cognitiva a personas que presentan dificultades en estas áreas como son, en este caso, las personas con TEA. Este proyecto ha sido financiado por el Departamento de Educación, Cultura y Deporte del Gobierno de Aragón y es coordinado por la Dirección General de Innovación y Formación Profesional de dicho departamento.

En este trabajo se ha hecho uso de su colección de pictogramas para que puedan ser incorporados durante la creación de los cuentos y facilitar así su posterior lectura por parte de los usuarios. Se han seleccionado los pictogramas de este proyecto porque son los más utilizados por las asociaciones que trabajan con personas con TEA y en las aplicaciones creadas para ellos, por lo que estarán más familiarizados y tendrán una mejor adaptación.

4.5 Android

Android [26] es un sistema operativo móvil basado en un núcleo Linux y otro software de código abierto. Este sistema fue desarrollado por Android Inc que, posteriormente, fue

adquirida por Google. En un comienzo, fue diseñado para teléfonos móviles inteligentes, pero posteriormente se integró para *tablets*, relojes inteligentes y automóviles (Android Auto). Actualmente, junto a iOS, abarca gran parte del mercado de sistemas operativos móviles, llegando a ser el más utilizado en el mundo.

4.6 Android Studio

Android Studio [27] es un **Integrated Development Environment (IDE)** basado en IntelliJ IDEA y enfocado al desarrollo de aplicaciones Android.

Esta herramienta fue seleccionada para el desarrollo del proyecto porque permite compilar nuestra aplicación a código nativo Android y dispone de emuladores que cargan estos cambios realizados en el código sin necesidad de reiniciar la aplicación. Por otro lado, también dispone de la posibilidad de conectar dispositivos físicos para desarrollar o realizar las pruebas. También está integrado con Git, lo que nos facilita el control de versiones de nuestro código.

4.7 iOS

iOS [28] es un sistema operativo móvil desarrollado por Apple Inc. Originalmente, sólo fue desarrollado para móvil (iPhone) pero, posteriormente, se extendió tanto a iPad como iPod Touch. Este sistema ofrece una interfaz muy fluida con respuestas inmediatas a las acciones de los usuarios. También se incluyen gestos diferentes según el contexto de la interfaz. Junto a Android es uno de los sistemas operativos móviles más utilizados.

4.8 XCode

Xcode [29] es un **IDE** para MacOS que contiene un conjunto de herramientas para desarrollar software tanto para este sistema como para iOS, watchOS y tvOS. En este proyecto, se ha utilizado para compilar el código de la aplicación y poder ejecutarla en los emuladores de la propia herramienta y comprobar su funcionamiento para iPhone y iPad.

4.9 Git

Git [30] es un sistema de control de versiones de código abierto y gratuito. Esta herramienta fue diseñada por Linus Torval pensando en la eficiencia, compatibilidad y manejo de proyectos de diferentes tamaños. Para este trabajo se ha seleccionado este sistema debido a que es un software libre y está integrado con los IDEs utilizados, lo que facilita y agiliza su uso.

Se ha utilizado junto a un cliente gráfico, Sourcetree (ver sección 4.10), para el desarrollo del proyecto y control del código de las diferentes funcionalidades, pudiendo realizar la revisión del mismo y revertiendo cambios en caso de que fuese necesario.

4.10 SourceTree

Sourcetree [31] es un cliente gráfico creado por Atlassian y orientado a manejar repositorios Git y Mercurial. Ofrece las mismas funcionalidades que Git (commit, push, pull, clone, ...) pero de una forma gráfica, lo que hace este proceso mucho más sencillo y rápido. En este caso se ha utilizado para gestionar el repositorio Git del proyecto.

4.11 Microsoft Project

Microsoft Project [32] es la herramienta de administración de proyectos que se ha utilizado para planificar y gestionar este proyecto. Este programa proporciona muchas funcionalidades que hacen la tarea de gestión mucho más eficaz y sencilla como, por ejemplo, crear diagramas de Gantt, realizar cálculos de costes, tiempos o el cálculo de caminos críticos.

4.12 LATEX

Latex [33] es una herramienta para la edición y composición de documentos con una alta calidad tipográfica. Por sus características, su uso está enfocado a la creación de artículos científicos, artículos académicos o tesis o, como en este caso, para la memoria del TFG. Por otro lado, permite la creación de documentos mediante un determinado lenguaje de comandos que permite ajustar el diseño a los estándares tipográficos. En el caso de este trabajo, se ha utilizado junto a la aplicación web, Overleaf [34], que nos permite la compilación del código Latex, la compartición del documento y la edición conjunta del código, que permitió realizar la revisión de la memoria a los tutores.

4.13 Balsamiq Wireframes

Balsamiq Wireframes [35] es una herramienta de creación de *wireframes* y *mockups* o bocetos mediante *widgets* definidos dentro de la aplicación. La creación de estos elementos es muy sencilla, mediante el desplazamiento de componentes presentes en las vistas y bocetos simples. Este programa se utilizó para realizar los primeros diseños que se mostraron a los tutores.

4.14 DrawIO

DrawIO [36] es una herramienta de creación de diagramas, esquemas o bocetos. En ella, se pueden crear diagramas de todo tipo debido a la gran variedad de elementos gráficos que proporciona. En este trabajo se ha utilizado para crear los diagramas de casos de uso.

Metodología de desarrollo

EN un proyecto de desarrollo software es muy importante trabajar en equipo de manera organizada, por ello, se usan distintas metodologías. Las metodologías de desarrollo son un conjunto de técnicas y procesos que se aplican para desenvolver un producto software. La aplicación de estos métodos permitirá organizar las tareas entre el equipo, reducir el nivel de dificultad del proyecto y entregar un buen producto en plazo.

En este proyecto se ha seleccionado la metodología ágil de desarrollo, **Scrum**. Generalmente esta metodología está enfocada al trabajo colaborativo entre varios miembros de un equipo, aunque en el caso de este trabajo la totalidad del equipo de desarrollo será representado por una única persona. A su vez, se combinará con la propuesta educativa, **ApS**, que permitirá desarrollar un producto software al mismo tiempo que se adquieren nuevos conocimientos y se realiza un aporte a la comunidad, en este caso a las personas que presentan dificultades en el neurodesarrollo.

5.1 Scrum

Scrum [37] es un marco de trabajo para el desarrollo ágil y mantenimiento de productos software. Su enfoque principal se basa en un correcto trabajo en equipo para abordar problemas complejos y poder resolverlos de la forma más productiva. Se suele ejecutar en ciclos de duración fija de unas dos o tres semanas llamados *Sprints*, aunque este intervalo de tiempo puede variar dependiendo del equipo de trabajo. Durante estos periodos se realizan reuniones diarias, llamadas *Dailys*, de unos quince minutos, en las que los miembros del equipo exponen sus tareas y los avances o dudas en las mismas. En los proyectos más completos donde se busca obtener resultados rápidos, con requisitos cambiantes o poco definidos, se suele aplicar este marco de trabajo, dado que se realizan entregas parciales y regulares del producto al cliente, consiguiendo una reacción rápida ante la competencia y aportando flexibilidad al desarrollo.

Scrum define un conjunto de componentes básicos y esenciales para garantizar una adecuada aplicación de la metodología. Entre ellos, destacan el equipo Scrum y sus roles, los eventos durante el ciclo de desarrollo, los artefactos y las reglas asociadas. En los siguientes apartados, vamos a profundizar en cada uno de ellos.

5.2 Equipo Scrum

5.2.1 Product Owner

El Dueño del producto (*Product Owner*) es el encargado de definir el valor del producto y las tareas del equipo de desarrollo. Esta persona es la encargada de gestionar el *Product Backlog* (ver sección 5.4.1), definiendo los diferentes elementos de éste y ordenándolos de una manera óptima. Será a su vez el responsable de presentar el *Product Backlog* al equipo de una forma clara y sencilla para su posterior desarrollo.

5.2.2 Scrum Master

El *Scrum Master* es el responsable de que el equipo aplique correctamente los principios de Scrum. Esta persona trabaja tanto con el *Product Owner* como con el equipo de desarrollo, ayudando y aclarando sus diferentes funciones y organizando su trabajo. El *Scrum Master* coordina a los diferentes miembros de una forma eficiente y productiva.

5.2.3 Equipo de desarrollo

El equipo de desarrollo está formado por un conjunto de profesionales encargados de realizar las tareas, que tienen el objetivo de entregar un incremento del producto finalizado que se pueda lanzar a producción al finalizar el *sprint*. Es un grupo auto-organizado a la hora de realizar las tareas que se le han asignado, con miembros multifuncionales, dado que se trata de desarrolladores especializados en resolver las diferentes características que presenta el producto. La sincronía de este equipo es fundamental para optimizar la eficiencia y efectividad del mismo.

5.3 Eventos de Scrum

5.3.1 Sprint

Una de las bases de Scrum es dividir el ciclo de desarrollo del software en iteraciones breves de tiempo en las que el equipo trabaja para completar una parte del desarrollo completo de

un producto [38]. Estos períodos de tiempo se denominan *sprints* y suelen durar de dos a tres semanas. En nuestro caso concreto, se ha establecido la duración en tres y, al finalizar cada uno de estos, se da comienzo al siguiente. Durante cada *sprint*, se divide el trabajo a realizar de una manera equitativa entre los miembros del equipo, lo que reduce los riesgos y permite asegurar y controlar la entrega del producto en plazo. En un *sprint* existen diferentes conceptos y eventos que se ejecutan durante su desarrollo. Dentro de ellos se encuentran la planificación del *sprint*, las *dailys*, la revisión del *sprint* y la retrospectiva del *sprint*, los cuales desarrollaremos en los siguientes puntos.

5.3.2 Planificación del sprint

La planificación del *sprint* consiste en planificar el trabajo que será llevado a cabo durante un *sprint* para completar el correspondiente incremento del producto [38]. Este plan será creado de forma colaborativa por el equipo Scrum al completo, en un intervalo variable de tiempo, dependiendo de la duración del *sprint*. El *Scrum Master* será el encargado de coordinar este evento y de que los demás miembros del equipo entiendan el propósito del *sprint*. Durante este evento se responderá a las preguntas de qué puede entregarse en el incremento (ver sección 5.4.3) del *sprint* que comienza y cómo será llevado a cabo el trabajo para entregarlo.

5.3.3 Daily

La *daily* es una reunión de aproximadamente quince minutos que sincroniza todo el equipo de desarrollo. En esta reunión se expone el trabajo realizado por cada miembro del grupo desde la última *daily* y el que se espera realizar antes de la siguiente. Al mismo tiempo, se comentarán las dudas surgidas o algún impedimento que haya aparecido durante el desarrollo. Este evento mejora la comunicación de los miembros y agiliza su desarrollo, identificando los impedimentos y promoviendo las tomas de decisiones rápidas ante ellos. La *daily* se usa para evaluar el progreso de las tareas planeadas para ese *sprint* y si podrá cumplirse el objetivo definido durante la planificación.

5.3.4 Revisión del sprint

Al final de cada *sprint*, es necesario llevar a cabo una revisión del mismo donde se analiza el trabajo realizado durante este proceso y se valora si el incremento alcanzado cumple con lo esperado. Esta revisión supone en ocasiones adaptar el *Product Backlog* para refinar el producto desarrollado. Se trata de una reunión informal, no de seguimiento, en la que los componentes del equipo de desarrollo comentan y analizan el trabajo realizado, con el objetivo de facilitar la retroalimentación de información y fomentar la colaboración.

5.3.5 Retrospectiva del sprint

La retrospectiva del *sprint* es una reunión que tiene lugar después de la revisión y antes de la planificación del siguiente *sprint* [38]. Se trata de un evento de análisis del *sprint*, evaluando el trabajo de las personas involucradas y los objetivos logrados, identificando los aciertos y errores durante el transcurso del *sprint* con el objetivo de mejorar de cara al siguiente. Con esta reunión se busca aumentar la efectividad y planificar estrategias para aumentar la calidad del producto. Con la finalización de esta retrospectiva se definirán las mejoras que serán llevadas a cabo en el siguiente *sprint*.

5.4 Artefactos de Scrum

5.4.1 Product Backlog

La lista de producto (*Product Backlog*) es un conjunto dinámico de requisitos en forma de historias de usuario necesarios para llevar a cabo el desarrollo de un producto [38]. El equipo de desarrollo ordena, prioriza y estima el esfuerzo de cada tarea de la lista en función de la importancia y la información sobre la misma. Posteriormente, en la planificación se descompondrán y organizarán las historias de usuario del *Product Backlog* que se abordarán durante ese *sprint*.

5.4.2 Sprint Backlog

El *Sprint Backlog* es el conjunto de elementos del *Product Backlog* seleccionados para ser realizados durante el *sprint* [38]. Este elemento se define en la fase de planificación, donde se tendrá en cuenta la información y estimaciones del esfuerzo de las historias de usuario del *Product Backlog*. A su vez se determina el incremento del producto para esa iteración y el objetivo del mismo. El equipo de desarrollo definirá esta lista teniendo en cuenta las diferentes funcionalidades que se tienen que cumplir para completar el incremento.

5.4.3 Incremento

El incremento es la entrega al cliente de las tareas completadas al final de cada *sprint* [39]. Se corresponde con una versión funcional del producto que incorpora las nuevas funcionalidades que se han desarrollado. De esta forma, el incremento en el último *sprint* da como resultado el producto final, cumpliendo con los requisitos establecidos por el *Product Owner*.

Cada incremento es definido en la reunión correspondiente a la planificación del *sprint*. Al mantener todo el trabajo en un *sprint* coordinado con los elementos del *Sprint Backlog*, se puede analizar y tener un control de estas entregas realizadas a los clientes.

5.5 Aplicación de la metodología en este proyecto

En este proyecto se ha aplicado Scrum de una manera diferente, dado que se trata de una metodología enfocada a equipos, y en este caso, el desarrollo ha sido realizado por una única persona, aún así se han tenido en cuenta la mayoría de sus características. Al comienzo del proyecto se estableció la duración de los diferentes *sprints* que se iban a llevar a cabo en un intervalo de tiempo de tres semanas. Posteriormente se definieron en aspectos generales los diferentes requisitos de la aplicación, que compusieron los elementos del *Product Backlog*, y se establecieron los distintos roles del equipo Scrum. El *Scrum Master* y el equipo de desarrollo fueron representados por el alumno y el *Product owner*, por los tutores, que ayudaron a definir las tareas y transmitieron al alumno los diferentes aportes realizados por el cliente, en este caso la asociación con la que se trabajó.

Al comienzo de cada *sprint*, el alumno y los tutores realizaron la planificación, en la que se definió el trabajo que iba a ser realizado durante el mismo. Durante el transcurso del *sprint*, no se realizaron *daily*s, dado que eran innecesarias con un único integrante en el equipo de desarrollo. Con la finalización, se realizó la revisión, donde se analizaron y se refinaron las tareas realizadas, y una pequeña retrospectiva en la que se observó si se habían cumplido los objetivos establecidos previamente.

5.6 Aprendizaje-servicio (ApS)

La metodología [Aprendizaje-Servicio \(ApS\)](#) es una propuesta educativa en la que se combina el aprendizaje académico y el servicio a la comunidad en el desarrollo de un proyecto [40]. Con este proceso, las personas participantes adquieren nuevos conocimientos para su formación profesional trabajando sobre necesidades reales del día a día, con el objetivo principal de mejorarlas. Por esto último, no es lo mismo que un voluntariado, ya que se realiza con el fin de lograr determinadas competencias específicas de la materia, en este caso el [TFG](#), mientras a su vez se realiza una contribución a la comunidad.

Este [TFG](#) se ha centrado en la realización de un servicio que ayude en el desarrollo de personas con diferentes diversidades funcionales, como es el caso de las personas con [TEA](#). Esta aplicación aporta un recurso para una etapa fundamental como es la infancia, en la que se desarrollan muchos aspectos claves para su posterior vida adulta. Por lo tanto, una aplicación de cuentos personalizados puede resultar de gran ayuda para la comunidad, tanto para los niños y niñas como para sus familiares o educadores y terapeutas, es decir, tanto en el contexto familiar como educativo. Dado que actualmente la mayoría de personas disponen de

un dispositivo móvil o una *tablet*, se ha adaptado un recurso tradicional como es un cuento al mundo tecnológico actual, facilitando la interacción con estas personas y mejorando su calidad de vida. Además, con la situación actual del COVID-19, es una gran ventaja disponer de estas herramientas, ya que la relación con los niños y niñas no puede ser la misma que antes, teniendo que llegar a veces a ser de forma no presencial, con lo que disponer de este tipo de recursos es de gran ayuda para ellos y, a su vez, se aporta un elemento de entretenimiento y aprendizaje para estos niños y niñas. Muchas de las terapias que antes se desarrollaban de forma presencial han tenido que ser substituidas por sesiones no presenciales u otros tipos de intervenciones, por lo que el uso de este tipo de recursos puede resultar de gran ayuda en esta nueva realidad social para mantener las sesiones de trabajo con los niños y niñas.

Durante el trascurso del TFG se ha colaborado con una asociación del entorno de A Coruña que realiza intervenciones específicas para personas con diferentes tipos de diversidades funcionales, entre ellas TEA, y cuyos profesionales nos han aportado su experiencia en los tipos de recursos que sus usuarios necesitan. Siguiendo la propuesta de la metodología ApS, se ha mantenido una comunicación activa con los profesionales de la asociación que nos han ayudado a establecer los requisitos de la aplicación, nos han proporcionado retroalimentación para mejorar diferentes aspectos de la misma considerando su mayor experiencia en este campo y han validado el resultado final como un recurso útil tanto en el entorno familiar como en sus sesiones de trabajo. De esta forma, el alumno tuvo en cuenta todas las aportaciones realizadas no solo por los tutores del TFG sino también por dichos profesionales para la implementación de las funcionalidades de la aplicación, las mejoras de la misma y la forma de abordarlas.

Finalmente, se puede concluir que la metodología ApS se integró exitosamente con el marco de desarrollo utilizado, Scrum, y ha permitido un aprendizaje eficaz del alumno, proporcionándole nuevos conocimientos y concienciación acerca de esta realidad social. Con ello, también se ha demostrado que se pueden tratar aspectos sociales mediante la tecnología y se ha destacado la importancia de este tipo de proyectos para atender necesidades reales en el contexto actual.

Planificación y seguimiento

EN este capítulo se desarrollará la planificación que se estableció al comienzo del trabajo y el seguimiento del mismo, para poder observar cómo han ido cada uno de los *sprints* y si se han cumplido las estimaciones realizadas. También se realizará un cálculo de los costes totales de todo el proyecto.

6.1 Planificación

La planificación del proyecto especifica y organiza las tareas que van a ser llevadas a cabo durante el transcurso del mismo, con el fin de conseguir un objetivo. Como se comentó en el capítulo 5, seguiremos el marco de trabajo Scrum, que divide el desarrollo del proyecto en *sprints*. En cada uno de ellos, detallaremos las tareas que se deben abordar, cómo se realizaron y cuánto tiempo se invirtió en el mismo.

6.1.1 Planificación inicial

En la planificación inicial se dividieron las tareas y el desarrollo del proyecto en seis *sprints*, de una duración aproximada de tres semanas cada uno. Estas tareas se determinaron a partir de los requisitos iniciales del proyecto establecidos en colaboración con la entidad especializada en intervenciones para niños con TEA (ver apéndice A). Para cada *sprint* en concreto, en la fase de planificación, se definieron las subtareas para distribuir equitativamente el trabajo y conseguir el objetivo definido para ese *sprint*. Durante el desarrollo del proyecto, se dedicaron alrededor de tres a cuatro horas diarias, teniendo en cuenta la disponibilidad del alumno. Este aspecto fue clave para determinar las tareas que se podían abordar en cada *sprint* en función de la estimación de esfuerzo que requerían. Aproximadamente, antes de la finalización del sexto *sprint*, se comenzó con la realización de la documentación, que es la que constituye este trabajo, la cual tendrá una duración aproximada de un mes, con una dedicación diaria de tres horas. Finalmente, el desarrollo del proyecto concluirá con una reunión final para determinar

el grado de consecución de los objetivos teniendo en cuenta la valoración de los profesionales de la entidad colaboradora.

La realización de este proyecto se estimó inicialmente en 7 meses, teniendo en cuenta las limitaciones temporales del estudiante debido a que tenía que compaginar su desarrollo con las horas de trabajo en la empresa. La fecha que se estableció para el inicio del proyecto fue el 1 de marzo de 2021, después de una reunión informal con los tutores del mismo, y la fecha de fin se programó para el 23 de agosto de 2021. En la figura 6.1, se puede ver un esquema temporal de la planificación realizada para este proyecto.

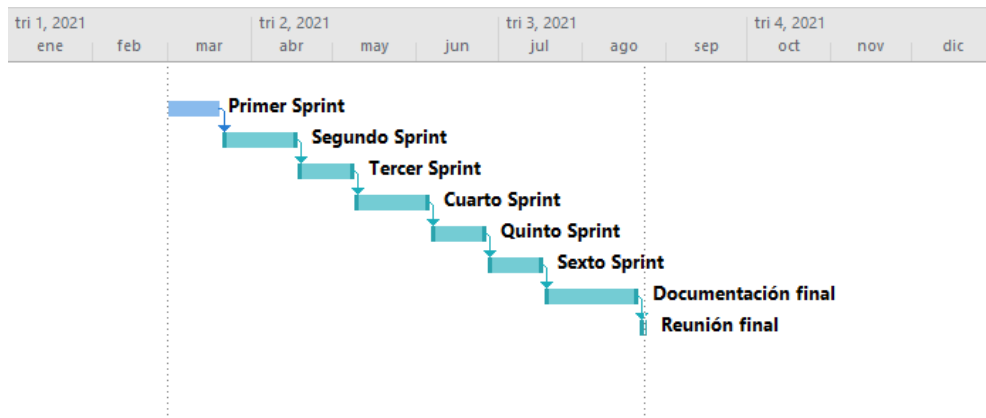


Figura 6.1: Esquema temporal de la planificación

6.2 Desarrollo del proyecto

En esta sección, se detalla cada uno de los *sprints* que conformaron el ciclo de desarrollo de la aplicación. En ellos, se explicará brevemente el trabajo realizado y las diferentes dificultades o adaptaciones que fueron necesarias tras evaluar el resultado de los *sprints*.

Primer Sprint

Durante este **primer sprint**, se realizó un análisis del proyecto y se estudiaron las diferentes tecnologías con las que se iba a trabajar. Se realizaron unos primeros *mockups* o bocetos de la aplicación para tener una idea general del diseño de la misma. Y, por último, se definieron de una forma general las principales funcionalidades y entidades que iba a tener la aplicación.

Segundo Sprint

En el **segundo sprint**, se desarrollaron las funcionalidades correspondientes a la gestión de usuarios. Para ello, se comenzó con la configuración de Firebase para la creación de la base

de datos de la aplicación en Firestore y la administración de usuarios con Firebase Authentication. Tras esta tarea, se realizó la implementación de los formularios de registro y las funcionalidades de inicio y de cerrar sesión. Con el alta de niños/as implementada, se realizó la acción de asignación de tutores o educadores a estos usuarios. Como ocurre al finalizar cada *sprint*, se realizó una reunión para observar y estudiar cómo había ido y para dar comienzo al siguiente. En este análisis, se aportó una nueva funcionalidad que no se había tratado y que se implementó en el siguiente *sprint*. Esta funcionalidad consistía en un filtro de búsqueda en la lista de usuarios.

Tercer Sprint

En este **tercer sprint**, se implementó la nueva funcionalidad comentada en el anterior *sprint* y se comenzó con el desarrollo de una parte fundamental de la aplicación, la lectura de cuentos. Se decidió realizar esta funcionalidad antes que la de gestión de cuentos para tener una primera vista de esta parte, ya que se trata de una pieza muy importante. Se realizó un primer diseño con textos y pictogramas estáticos, para poder establecer el diseño y añadir las funcionalidades principales, como son la navegación del cuento y la reproducción de audio. Para finalizar el sprint, se hicieron pruebas unitarias de las diferentes funcionalidades implementadas, lo que se hará siempre al finalizar cada uno de los *sprints*.

Cuarto Sprint

Durante el **cuarto sprint**, se gestionó la parte correspondiente a los cuentos, comenzando por la implementación de la creación, que es la acción principal. En esta creación se realizó el diseño de varias pantallas y varias funcionalidades que suponían una pequeña dificultad, como la importación y el “parseado” del cuento o la búsqueda de pictogramas. Una vez creado y asignado un cuento, ya podría ser leído por el usuario correspondiente. Para finalizar este sprint se realizaron las pruebas de todas estas funcionalidades.

Quinto Sprint

Una vez finalizada la creación de los cuentos, el **quinto sprint** se dedicó a la edición y al borrado de estos cuentos. La edición adquirió un aspecto y una funcionalidad similar a la de la creación. En el borrado, se hizo uso de las Cloud Functions para poder hacer una correcta eliminación de toda la información de un cuento en base de datos. Para finalizar el sprint, se realizaron pruebas de todas estas funcionalidades y se integraron con lo realizado en el anterior.

Sexto Sprint

Este **sexto sprint** se utilizó para mejorar varios aspectos de la vista y ,con ello, adaptarla a las necesidades de las personas con **TEA**. Además, en él se gestionaron diferentes aportaciones que se realizaron desde la asociación colaboradora después de mostrarles una versión ya avanzada de la aplicación.

Fase de documentación

En esta parte del proyecto se realizó la memoria de este trabajo.

Reunión final

Se realizó una reunión final con los tutores para comprobar y analizar la correcta realización del trabajo. Además, se les presentó la versión final de la aplicación a varios profesionales de la entidad colaboradora para valorar su utilidad dentro del contexto de la metodología **ApS** y sugerir mejoras que puedan completar la aplicación en un futuro.

6.2.1 Seguimiento

En el desarrollo del proyecto se ha intentado seguir la planificación inicial y cumplir con los objetivos establecidos para cada *sprint*. Durante el desarrollo de cada uno de ellos, se llevaron a cabo el conjunto de tareas que se han comentado en el apartado anterior (ver 6.1). El segundo *sprint* ha sido quizás el más problemático, debido al desconocimiento y la poca práctica con las tecnologías que se han utilizado en este proyecto, lo que provocó una desviación significativa de una semana, que afectó a los demás *sprints* y tareas del proyecto debido a sus relaciones Fin-Comienzo. En el cuarto *sprint* se realizó una pequeña pausa de cinco días debido a una enfermedad del alumno, lo que afectó a los siguientes *sprints*, con un retraso de estos días. Los demás *sprints* se realizaron cumpliendo los plazos establecidos al comienzo de cada uno. La documentación se realizó en el transcurso de aproximadamente un mes, a la vez que se finalizaba con la implementación de la aplicación.

6.3 Recursos

En esta sección describiremos los recursos humanos y técnicos necesarios para la elaboración de este trabajo.

Recursos humanos

Los recursos humanos que nos encontramos en este proyecto son los comunes en un proyecto Scrum y se dividen en el **Scrum Master**, **Product Owner** y **equipo de desarrollo**.

Dentro del equipo de desarrollo vamos a tener como mínimo un analista/programador y un diseñador.

Todos estos perfiles serán representados por el estudiante, excepto el de *Product Owner*, cuyo papel será desempeñado por los tutores, que transmitirán las aportaciones realizadas por el cliente, en este caso la asociación con la que se trabajó. Tanto el *Scrum Master* como el *Product Owner* serán los encargados de definir las diferentes tareas y funcionalidades del proyecto y de organizarlas dentro de los *sprints*. Por otro lado, el analista/programador se encargará del análisis de las diferentes tareas y de su desarrollo. Y por último, el diseñador que, junto al analista/programador, será el encargado del diseño de la aplicación.

Recursos técnicos

En cuanto a los recursos técnicos, debemos distinguir entre los recursos materiales y los recursos software.

Los recursos materiales que se han utilizado para el desarrollo del proyecto han sido el ordenador personal del estudiante, un portátil Asus con procesador Intel Core i7 y 16GB de RAM, para la implementación de todo el código. Por otro lado, para la compilación para iOS, se ha utilizado un Macbook proporcionado por la empresa en la que trabaja el alumno, al cual se conectaba remotamente. Para las pruebas de la aplicación, se utilizó un dispositivo móvil con sistema operativo Android, Xiaomi Mi 10 Lite.

Los recursos software que se utilizaron son todos los nombrados en el capítulo 4, para los que se disponían de licencias gratuitas. En el caso de **Microsoft Project**, se utilizó una licencia proporcionada por la Universidade da Coruña. En **Firebase** se tuvo que contratar la versión Premium, debido al uso de las Cloud Functions, pero aún así no se tuvo que realizar ningún pago ya que permiten su uso dentro de unos límites de manera gratuita.

6.4 Costes

Para la estimación de los costes asociados al proyecto, se realizó el cálculo de los sueldos de los recursos humanos y los costes de los recursos materiales.

Para los costes ligados a los recursos humanos, se consultaron los salarios medios anuales del *Scrum Master* [41], del analista/programador [42] y del diseñador [43], en España. Estos salarios anuales se pueden ver en la tabla 6.1, junto a los correspondientes salarios por hora aproximados, dato que se utilizó para el cálculo de costes.

Recursos humanos / Salarios	Salario medio/año	Salario medio/hora
Scrum Master	34 963 €	20 €
Analista/Programador	28 487 €	16 €
Diseñador	18 782 €	11 €

Tabla 6.1: Costes de los recursos humanos

Teniendo en cuenta estos datos y la horas trabajadas de cada persona en el proyecto, los costes totales de los recursos humanos del proyecto serán los de la siguiente tabla 6.2:

Recursos humanos/Salarios	Horas totales	Salario medio/hora	Salario total
Scrum Master	104	20 €	2 800 €
Analista/Programador	1 008	16 €	16 128 €
Diseñador	296	11 €	3 256 €
Coste total			22 184 €

Tabla 6.2: Costes totales de los recursos humanos

Por otro lado, en la tabla 6.3 podemos encontrar reflejados los costes de los recursos materiales. En cuanto a los recursos software, como se comentó en la sección anterior, no supusieron ningún gasto adicional, por lo que no se han tenido en cuenta para la estimación de costes.

Recursos materiales / Coste	Coste
Portátil personal	800 €
Smartphone	220 €

Tabla 6.3: Costes de los recursos materiales

Por lo tanto teniendo en cuenta, toda esta información podemos concluir que el coste total del proyecto serían 23 204 euros, como se ve reflejado en la tabla 6.4.

Recursos \ Coste	Coste total
Humanos	22 184 €
Materiales	1 020 €
Total	23 204 €

Tabla 6.4: Coste total del proyecto

EN este capítulo se va a detallar el análisis de requisitos funcionales y no funcionales que deben cumplir nuestra aplicación. Los diferentes requisitos de la aplicación se han definido siguiendo la metodología ApS, previamente introducida en el capítulo 5, gracias a las aportaciones de los profesionales de la asociación con la que se ha colaborado, recogidas en el apéndice A. De esta forma, en los siguientes apartados se detallarán los diferentes actores que pueden interactuar con la aplicación, los requisitos identificados para cumplir con las necesidades especificadas por la asociación y los distintos casos de uso que se derivan de estos requisitos.

7.1 Actores

En nuestro sistema podemos encontrar cuatro actores: el tutor (ver tabla 7.1), el educador (ver tabla 7.2), el usuario (el niño o la niña) (ver tabla 7.3) y el usuario no registrado (ver tabla 7.4).

ACT-01	Tutor
Descripción	Este actor representa al usuario administrador o familiar y, por tanto, será el que más permisos tiene dentro de la aplicación. Este usuario podrá dar de alta niños y niñas en el sistema, asignarlos a otros tutores o educadores, gestionar cuentos, iniciar sesión en la aplicación, cerrar esta sesión y restablecer su contraseña. En definitiva, representa a la persona responsable de un niño o niña que va a utilizar la aplicación.

Tabla 7.1: Actor Tutor

ACT-02	Educador
Descripción	Representa al usuario educador del niño/a (terapeuta / personal educativo de apoyo) y podrá realizar los mismos casos de uso que el actor tutor (ver tabla 7.1), excepto dar de alta a niños y niñas dentro del sistema y asignarlos a otros usuarios.

Tabla 7.2: Actor Educador

ACT-03	Usuario (niño o niña)
Descripción	Este actor engloba a los distintos niños y niñas con TEA u otro trastorno de los mencionados en el capítulo 2, que utilizarán la aplicación. Estos usuarios podrán seleccionar entre varios géneros literarios el que desea leer y, luego, elegir un cuento de ese género y leerlo. Dentro de la lectura dispondrá de la opción de audio-lectura o lectura manual y, en ambos casos, podrá interactuar con la historia.

Tabla 7.3: Actor Usuario

ACT-04	Usuario no registrado
Descripción	Se trata del usuario sin registrar todavía en la aplicación como, por ejemplo, un familiar del niño o de la niña que se acaba de descargar la aplicación.

Tabla 7.4: Actor Usuario no registrado

7.2 Requisitos

7.2.1 Requisitos funcionales

Los requisitos funcionales describen cómo debe responder la aplicación frente a la interacción con los distintos actores u otros sistemas. Estos requisitos los dividiremos en los requisitos correspondientes a los tutores o educadores y los correspondientes a los usuarios (niño o niña).

Requisitos funcionales (tutor/educador)

Como ya se ha comentado en el apartado de la definición de actores 7.1, estos dos tipos de usuario podrán acceder a las mismas funcionalidades, excepto por el requisito de **Dar de**

alta niño/a y Asignar niño/a a otro tutor o educador, que sólo pueden ser realizados por el tutor (administrador).

- **Registro del tutor/educador:** permite a estos usuarios darse de alta en el sistema mediante un formulario en el que podrán seleccionar entre los dos roles principales de la aplicación: tutor o educador. Dispondrán de un correo y contraseña únicos para iniciar sesión y entrar posteriormente en la aplicación.
- **Cambiar contraseña del tutor/educador:** tras haberse registrado en la aplicación, en el momento que lo precise el usuario, podrá cambiar su contraseña para entrar en la misma.
- **Gestión de usuarios (solo tutor):** se permitirá dar de alta los niños/as dentro del sistema, aportando información relevante acerca de ellos o ellas. Posteriormente, el tutor, como responsable del niño/a, podrá asignarles nuevos tutores o educadores registrados en la aplicación para que puedan trabajar con él a través de los cuentos que vayan creando. Por otra parte, el tutor también podrá crear sus propios cuentos y asignarlos. Además, podrá modificar los datos acerca de cada niño/a del que es responsable.
- **Gestión de cuentos:** la gestión de cuentos permite crearlos para que sean más personales y adaptados a las personas con TEA. Se utilizarán historias creadas por el autor, con pictogramas o imágenes de la cámara o galería con las que se quieran trabajar o, simplemente, para que sea atractiva y motivadora. Dispondrá de un amplio número de pictogramas entre los que podrá seleccionar el más adecuado para una correcta adaptación. Se ofrecerá además la posibilidad de importar un fichero de texto con las frases del cuento para evitar que el usuario tutor o educador tenga que escribirlo directamente con el dispositivo móvil. Al finalizar la creación del cuento, será necesario asignarlo a uno o varios niños/as. Por otro lado, se permite la edición de estos cuentos previamente creados desde la perspectiva de lectura del niño/a y desde una vista menos visual, pero con una modificación más intuitiva. Esta funcionalidad sirve para corregir una simple equivocación o una posible reacción negativa del niño/a hacia el cuento, lo que implica la necesidad de modificarlo. También, se podrá eliminar cualquiera de los cuentos previamente creados.

Requisitos funcionales (usuario niño/a)

- **Lectura de cuentos:** este requisito es una de las partes fundamentales de la aplicación, ya que deberá mostrar los diferentes cuentos que han creado y asignado los tutores o educadores para un niño/a, ordenados por géneros o categorías. Tanto los géneros como los cuentos se mostrarán con imágenes o pictogramas representativos y atractivos

para ellos, de forma que atraigan su atención y les motive a leerlos. Habrá variedad de géneros y cuentos, aunque también se podría llegar a tener un único género con cuentos muy similares, debido a los gustos muy restringidos y únicos que pueden tener ciertas personas con TEA, como se comentó en la sección 2.4.3.

Para la lectura, se ofrecerán dos opciones: la audio-lectura y la lectura normal o manual. Ambas opciones deben mostrar el texto del cuento y pictogramas o imágenes fácilmente comprensibles y adaptadas para ellos. Además, se podrán reproducir las palabras y los pictogramas seleccionándolos, con un audio claro y pausado.

7.2.2 Requisitos no funcionales

Los requisitos no funcionales de la aplicación representan las propiedades del sistema, en lo referente a rendimiento, seguridad, accesibilidad, ... Estos requisitos se originan en las necesidades de los propios usuarios. En este proyecto, se han identificado los siguientes requisitos no funcionales:

- **Requisito de accesibilidad:** como se trató en la sección de accesibilidad para las personas con TEA (apartado 2.4.4), es muy importante que la aplicación sea accesible para todas las personas. Sobre todo en el caso de las personas con TEA u otro tipo de trastorno del neurodesarrollo debe tener unas determinadas características, como ser simple e intuitiva, con una transición entre pantallas fluida y con tiempos de respuesta cortos, dado que este tipo de personas tiende a estresarse ante eventos inesperados o demasiado lentos.

En lo que concierne a los tutores y educadores, su vista debe presentarse de una forma fácil de usar por todo tipo de personas. Un educador como, por ejemplo, un logopeda, puede estar familiarizado con este tipo de aplicaciones, pero también puede ser usada por un familiar del niño o niña con poca experiencia en el uso de estas tecnologías. Por ello, debe implementarse una aplicación intuitiva y clara.

- **Requisito de diseño:** siguiendo el estándar Material Design, mencionado posteriormente en el apartado 8.2.1, la aplicación debe tener un diseño minimalista y no muy cargado. Considerando las personas a las que está enfocada, se deberán resaltar los pictogramas o el contenido visual frente a las partes con texto.
- **Requisito de hardware:** para utilizar esta aplicación se precisa de un *smartphone* o *tablet* con Android superior a la versión 8 o un dispositivo iOS con versión superior a la 10.
- **Requisito de seguridad:** como se comentó anteriormente en el apartado 7.1, se deben dividir los permisos de los usuarios dentro de la aplicación, delegando la capacidad

administrativa a un único tipo de usuario. Esto es especialmente relevante ya que el control del acceso a los datos del niño/a y de la posibilidad de interactuar con él a través de los cuentos en la aplicación debería recaer en un único responsable (padre, madre, tutor legal, ...).

7.3 Casos de uso

En esta sección, vamos a describir los casos de uso para los diferentes actores de la aplicación que derivan de los requisitos presentados en la sección anterior.

7.3.1 Casos de uso del usuario no registrado

- **CU-01 Registrar usuario:** un usuario no registrado en el sistema puede darse de alta mediante el formulario de registro, en el que proporcionará su información, como el tipo de usuario, es decir, si será **tutor** o **educador**. Este caso de uso podemos verlo representado en la figura 7.1 junto a su actor.

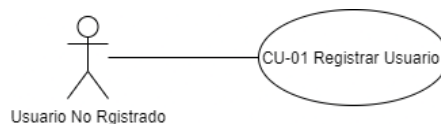


Figura 7.1: Casos de uso del usuario no registrado

7.3.2 Casos de uso del tutor/educador

- **CU-02 Iniciar sesión:** una vez registrado en el sistema, el tutor/educador podrá iniciar sesión con su cuenta de correo y contraseña, previamente establecidas en el registro.
- **CU-03 Cambiar la contraseña del tutor/educador:** el tutor o educador podrá cambiar su contraseña de usuario en el momento en que lo desee.
- **CU-04 Visualizar niños/as asignados:** tras haber iniciado sesión (CU-02), el usuario visualizará los niños/as que tiene asignados o los que ha dado de alta, en el caso del tutor. Si esta persona no tiene ningún niño/a asignados, verá un mensaje informativo de esa situación.
- **CU-05 Seleccionar un niño/a:** si esta persona tiene niño/as asignados, podrá seleccionar uno para entrar en su vista y allí podrá visualizar los cuentos que tiene asignados este niño/a, en caso de que tuviese alguno.
- **CU-06 Visualizar cuentos creados:** en la vista de gestión de cuentos, el tutor/educador podrá observar los cuentos que ha creado. En caso de que no hubiese creado ninguno, se mostrará un mensaje informativo reflejando esta situación.

- **CU-07 Crear cuento:** el tutor/educador crea un nuevo cuento en el sistema.
- **CU-08 Editar cuento:** el tutor/educador puede modificar determinada información de uno de los cuentos que ha creado anteriormente.
- **CU-09 Editar cuento de forma manual:** el tutor/educador modifica de forma manual uno de los cuentos que ha creado anteriormente.
- **CU-10 Editar cuento desde la vista de lectura:** el tutor/educador modifica un cuento desde la vista que tendría el niño o la niña en la lectura.
- **CU-11 Eliminar cuento:** el tutor/educador elimina uno de los cuentos del sistema.
- **CU-12 Establecer datos generales del cuento:** en la creación del cuento, el tutor/educador establece los datos generales del cuento: el título, el género y el autor.
- **CU-13 Establecer portada del cuento:** en la creación del cuento, el tutor/educador debe establecer una portada para el cuento. Ésta podrá ser un pictograma o una imagen.
- **CU-14 Visualizar ayuda de creación:** en la creación, el tutor/educador visualiza un texto de ayuda para la creación del cuento, en el cual se especifica en qué formato se debe importar la historia.
- **CU-15 Importar un fichero:** durante la creación, el tutor/educador importa un fichero de tipo *txt* que contendrá la historia en el formato que se precisa.
- **CU-16 Asignar una pictograma/imagen a una palabra:** en el segundo paso de la creación, se permite asociar un pictograma o una imagen a una palabra del cuento importado, que será mostrada en la lectura del mismo.
- **CU-17 Asignar una pictograma/imagen a una página:** durante la creación del cuento, se permite asociar opcionalmente un pictograma o una imagen a la página que se está gestionando. Esta imagen busca representar el sentido o la emoción de la frase correspondiente a esa página.
- **CU-18 Establecer un texto de escucha al pictograma/imagen de la página:** se permitirá también asociar un texto cualquiera a una página del cuento, pero únicamente si se ha asociado un pictograma/imagen previamente.
- **CU-19 Desasociar un pictograma/imagen de una palabra:** se desasocia (elimina) el pictograma/imagen que se añadió anteriormente a una palabra.
- **CU-20 Ver pictograma/imagen asignado a una palabra:** el tutor/educador puede visualizar el pictograma/imagen tras asociarlo a una palabra.

- **CU-21 Modificar datos generales del cuento:** en la edición del cuento, se pueden modificar los datos generales del mismo: la portada, el título, el género y el autor/a.
- **CU-22 Modificar una palabra:** en el segundo paso de la edición, se puede modificar una palabra de la historia.
- **CU-23 Asignar un cuento a un niño/a:** tras completar todos los datos necesarios para la creación del cuento, el tutor/educador deberá asignar el cuento a uno o varios niños y niñas. Al final de la edición también se llegará a este paso pero, en este caso, será opcional asignar el cuento.
- **CU-24 Buscar niño/a por apellidos:** en la vista de visualización de niños/as, se podrá filtrar esta lista por apellidos mediante un buscador.
- **CU-25 Buscar un pictograma:** a la hora de asociar un pictograma o una imagen, se podrá buscar en la [Application Programming Interface \(API\)](#) de ARASAAC (ver apartado 4.4) el pictograma que se precise.
- **CU-26 Seleccionar un pictograma de la búsqueda:** tras la búsqueda anterior, podemos seleccionar un pictograma de la lista.
- **CU-27 Seleccionar imagen de la galería:** el tutor/educador puede seleccionar una imagen de su galería, en vez de un pictograma, para después asociarla a una palabra o a la portada del cuento.
- **CU-28 Sacar una foto:** el tutor/educador podrá sacar una foto a modo de imagen para asociarla posteriormente a una palabra o a la portada del cuento.
- **CU-29 Ver los datos del niño/a:** el tutor/educador puede ver información personal acerca de un niño/a del que es responsable / tiene asignado.
- **CU-30 Cerrar sesión:** el tutor/educador cierra sesión y vuelve a la pantalla de inicio de la aplicación.

En los siguientes diagramas, podemos observar los casos de uso enumerados anteriormente, divididos en las diferentes funcionalidades y pantallas de la aplicación.

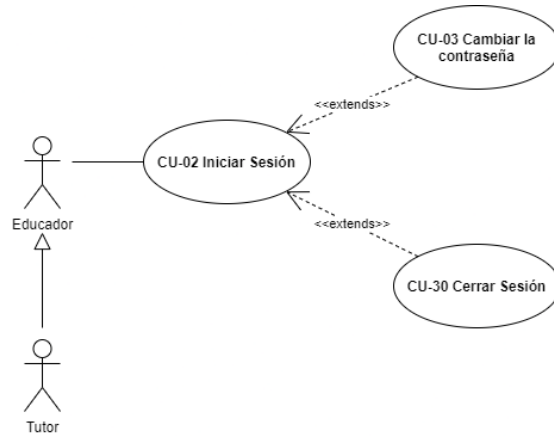


Figura 7.2: Casos de uso para gestión de sesión (Tutor y Educador)

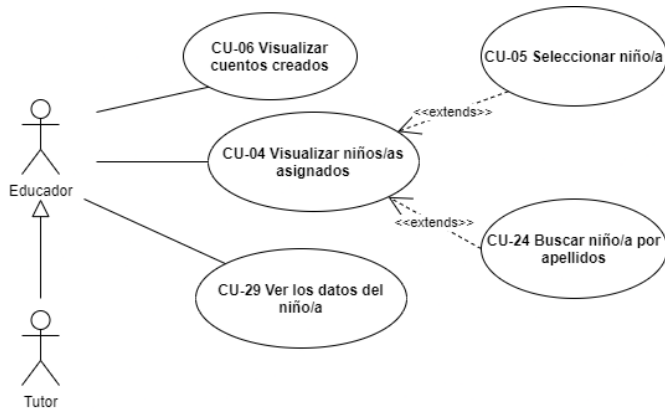


Figura 7.3: Casos de uso en la pantalla principal de la aplicación (Tutor y Educador)

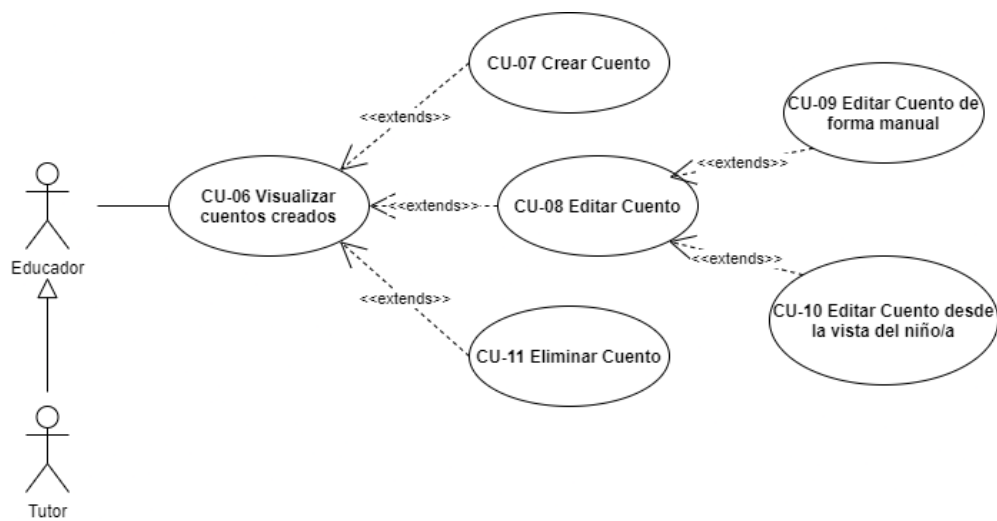


Figura 7.4: Casos de uso de la gestión de cuentas (Tutor y Educador)

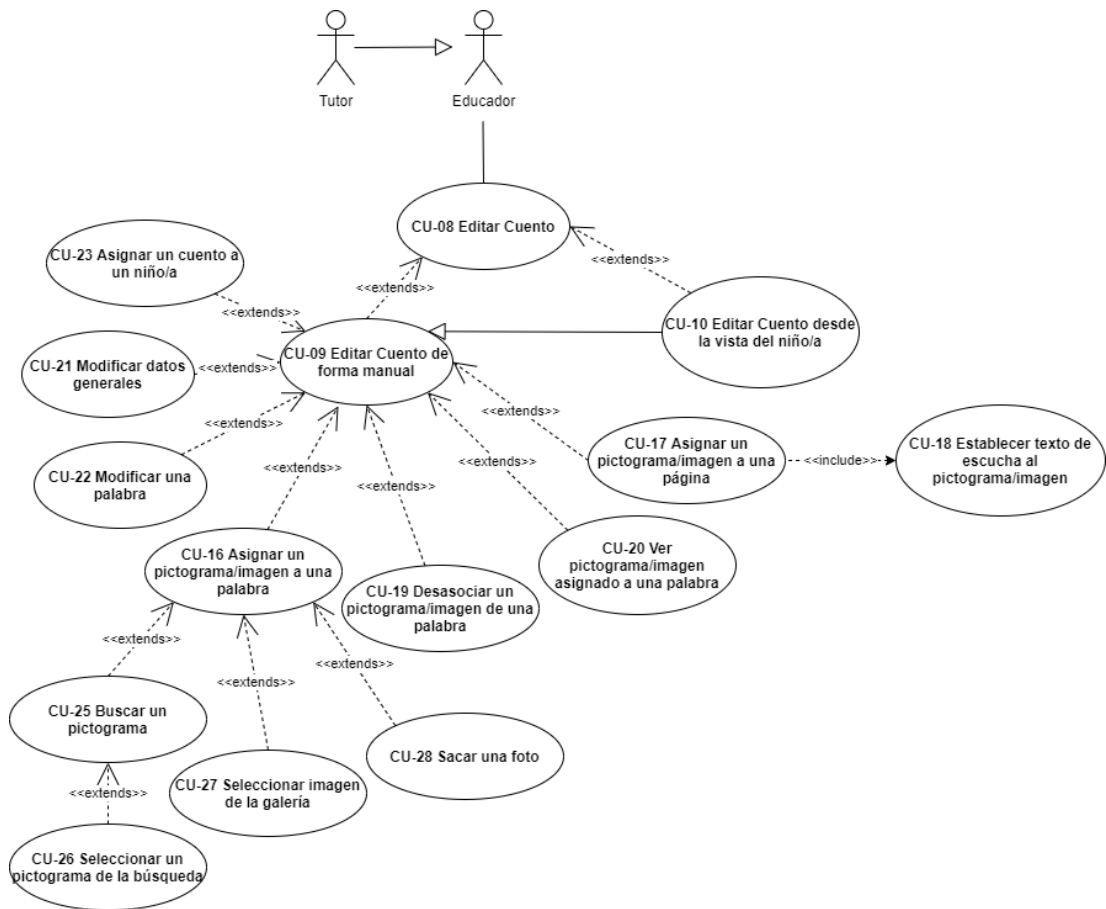


Figura 7.5: Casos de uso de la edición de cuentos (Tutor y Educador)

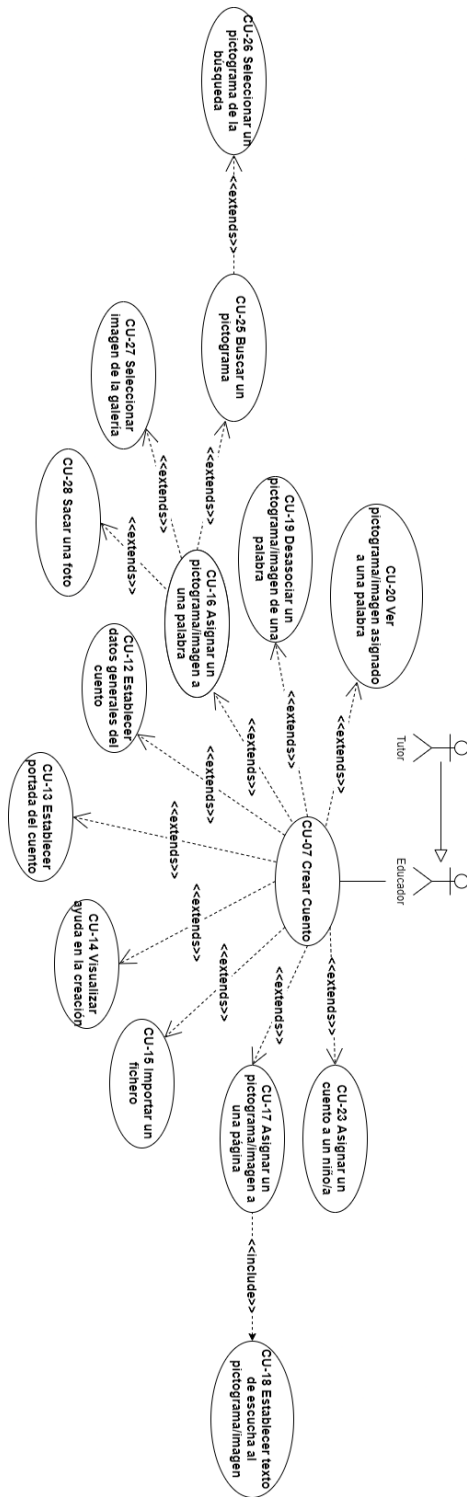


Figura 7.6: Casos de uso de la creación de cuentas (Tutor y Educador)

7.3.3 Casos de uso del tutor

En este apartado se recogen los casos de uso específicos para el tutor:

- **CU-31 Dar de alta un niño/a:** el tutor podrá dar de alta un niño o una niña en el sistema mediante un formulario.
- **CU-32 Asignar un tutor o educador a un niño/a:** el tutor podrá asignarle a un niño o a una niña, uno o varios educadores o tutores.
- **CU-33 Modificar datos del niño/a:** el tutor puede modificar la información personal relacionada con el niño/a.

Estos casos de uso para un tutor se pueden apreciar en la figura 7.7.

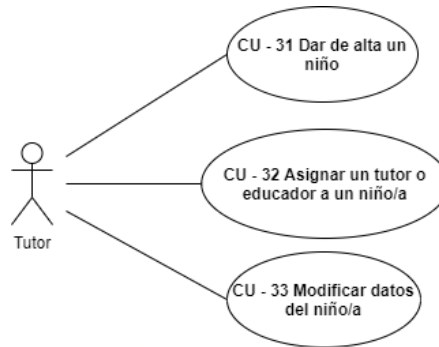


Figura 7.7: Casos de uso de la gestión de usuarios niño/a (Tutor)

7.3.4 Casos de uso del usuario (niño/a)

Los casos de uso asociados a los niños/as que van a leer el cuento en la aplicación son:

- **CU-34 Seleccionar género:** el niño o la niña podrá seleccionar entre los géneros de lectura disponibles.
- **CU-35 Seleccionar cuento:** podrá seleccionar un cuento dentro del género seleccionado en el caso de uso anterior (CU-34).
- **CU-36 Leer cuento de forma manual:** una vez seleccionado el cuento, podrán leer el cuento de forma normal.
- **CU-37 Audio-lectura:** podrá seleccionar la opción de audio-lectura que leerá el cuento a medida que avanzan las páginas.
- **CU-38 Seleccionar palabra:** dentro de cualquiera de las dos lecturas anteriores, se podrá seleccionar una palabra para reproducirla de una forma sencilla y clara.

- **CU-39 Seleccionar pictograma/imagen:** dentro de cualquiera de las dos lecturas anteriores, se podrá seleccionar un pictograma o una imagen para reproducir la palabra o texto asociado.
- **CU-40 Pasar página:** el niño o la niña podrá navegar por las diferentes páginas del cuento.

A continuación, podemos ver en la figura 7.8 el diagrama de casos de uso del usuario lector.

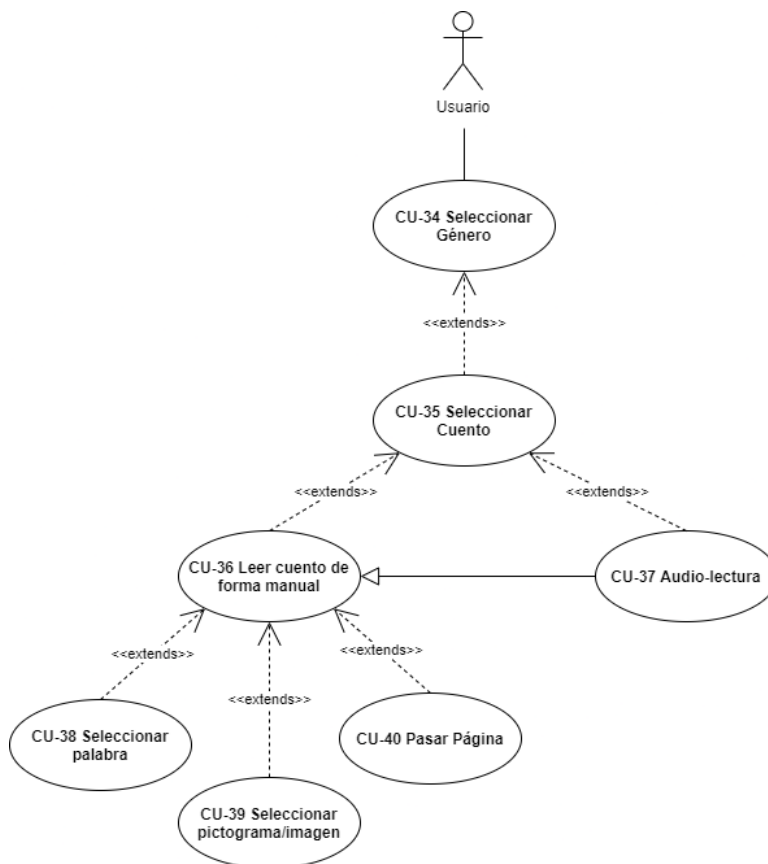


Figura 7.8: Casos de uso de la lectura de cuentos (Niño/a)

Capítulo 8

Diseño

EN este capítulo de la memoria, se explicará el patrón de diseño utilizado en el desarrollo de la aplicación y el por qué de su elección. También se definirá el estándar de diseño que se ha seguido para la vista y el diseño lógico de la base de datos.

8.1 Patrón de diseño

Los patrones de diseño se utilizan en proyectos software para desarrollar un código limpio y organizado que sea fácil de comprender frente a otros desarrolladores. En este caso, se ha seguido el patrón de diseño *Model-View-ViewModel (MVVM)*, muy común en estos tipos de proyectos. Este patrón permite separar la lógica de negocio de la interfaz de usuario [44], lo que nos facilita la escalabilidad y el mantenimiento de nuestro proyecto. Para ello, establece los siguientes componentes:

- **Modelo (*Model*):** es el componente donde se gestionan los datos de la aplicación y su lógica. En esta capa es donde se realizan las consultas a la base de datos y se implementan las diferentes funcionalidades de la aplicación. Será la primera en tenerse en cuenta al iniciar la aplicación.
- **Vista-Modelo (*ModelView*):** es la capa fundamental para la comunicación de la vista con el modelo. Implementa ciertos enlaces que permitirán conectar nuestra interfaz de usuario con el modelo, donde están almacenadas las funcionalidades y los datos.
- **Vista (*View*):** representa la capa de interfaz de usuario, en la que se muestra la información y funcionalidades mediante componentes visuales. Este es el componente con el que interactuarán los usuarios finales de la aplicación.

Podemos ver un ejemplo gráfico de la comunicación entre las diferentes capas en la figura 8.1.

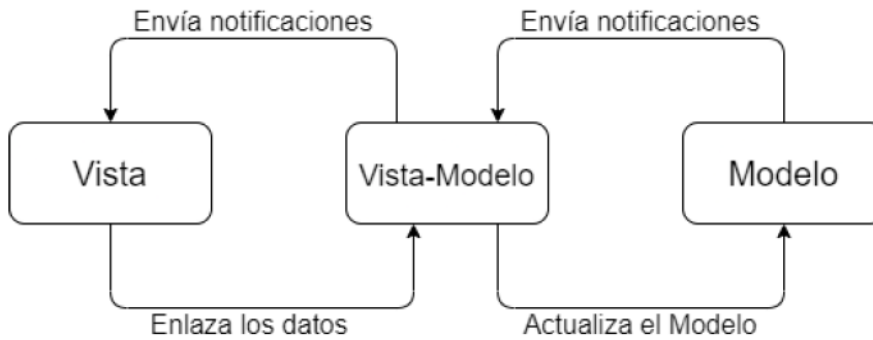


Figura 8.1: Ejemplo gráfico del patrón de diseño MVVM

8.2 Diseño de la vista

En este apartado, se explicará el estándar que se ha seguido para realizar el diseño de la aplicación y la interfaz de usuario.

8.2.1 Material Design

Material Design es un estándar de diseño creado por Google y que se centró en la visualización para el sistema operativo Android. Fue integrada en Android Lollipop y se aplicó primero en Google Drive, Docs, Sheet y Slides, y posteriormente de manera progresiva a todos los productos de Google.

Material Design se inspira en el mundo físico para sus texturas, exponiendo un diseño general sencillo, sin elementos sobrecargados que puedan afectar o distraer al usuario, y hace uso de componentes muy sencillos e intuitivos.

8.2.2 Interfaz de usuario

Siguiendo el principio definido en el apartado anterior 8.2.1, para las pantallas que usarán los niños y niñas con TEA, se ha creado un diseño muy sencillo, con interacciones fluidas y rápidas, y con elementos intuitivos y poco sobrecargados para favorecer su accesibilidad y evitar que se pongan nerviosos a la hora de utilizar la aplicación. La interfaz de usuario en este caso se centrará en las imágenes y pictogramas más que, por ejemplo, en el propio texto del cuento, ya que estos elementos resultarán mucho más atractivos, motivadores y comprensibles para estas personas.

Por otro lado, la vista de los tutores y educadores se centra en las diferentes funcionalidades que encontramos en la aplicación, es decir, en que tanto la gestión de usuarios como de los cuentos sea ágil y sencilla. Una de las partes fundamentales de la aplicación es la creación y edición de cuentos, que debe permitir crear cuentos en cualquier momento y completamente

personalizados. Por ello, esta creación y edición se realizará en tres pasos con una interfaz eficaz y simple. A su vez, esta parte de la aplicación debe ser accesible para todas las personas cercanas al niño o la niña, de forma que, con un mínimo conocimiento en aplicaciones móviles, puedan ser capaces de utilizarla. En el apéndice D, podemos ver representados y explicados los bocetos de esta interfaz que se realizaron al comienzo de este proyecto.

8.3 Diseño lógico de la base de datos

Como se ha explicado en el capítulo 4, para almacenar los datos de la aplicación se utilizará Firestore en la nube. Firestore es una base de datos no relacional y, por lo tanto, almacenará sus datos en formato *JavaScript Object Notation (JSON)*, como las demás base de datos de este tipo. El diseño lógico de nuestra aplicación representará, por tanto, una estructura no relacional donde los datos estarán organizados en colecciones y documentos, que contendrán a su vez varios atributos. A continuación, en la figura 8.2, podemos ver representado nuestro esquema de datos que incluye las diferentes entidades que tenemos en el sistema y sus atributos con sus tipos.

Para definir esta estructura de la base de datos, se analizaron las diferentes entidades que hay dentro de la aplicación y las relaciones que iba a haber entre ellas. Por lo tanto, se llegó a la conclusión de que nuestra base de datos iba a contener cuatro entidades principales: *Users*, *Tutors*, *Educators* y *Stories*, la cual contiene dos subcolecciones, *Pages* y *Words*. Además, todas estas colecciones o entidades tienen relaciones entre ellas, exceptuando las subcolecciones. A continuación, se definirán en mayor profundidad estas colecciones y sus diferentes atributos. Para las colecciones y sus atributos se decidió utilizar nombres en inglés.

8.3.1 Colección de tutores

La colección *Tutors* de nuestra base de datos representa al usuario administrador de la aplicación que será el encargado de la gestión de usuarios. Se identifica por un código único generado por Firebase al almacenarlo en el sistema. A su vez, se almacenará información del usuario como, por ejemplo, sus credenciales para iniciar sesión en la aplicación y el atributo *tutorType* que almacena si el usuario es tutor o educador, para posteriormente poder identificarlo dentro de la aplicación, ya que disponen de diferentes permisos dentro de la misma y podrán realizar diferentes tareas (ver tabla 8.1).

```

{
  "educators":{
    "educatorID":"<String>"{
      "name":"<String>",
      "fullname":"<String>",
      "email":"<String>",
      "password":"<String>",
      "phoneNumber":"<Integer>",
      "tutorType":"<String>",
      "stories":"<Collection>",
      "users":"<Collection>"
    }
  },
  "tutors":{
    "tutorID":"<String>"{
      "name":"<String>",
      "fullname":"<String>",
      "email":"<String>",
      "password":"<String>",
      "phoneNumber":"<Integer>",
      "tutorType":"<String>",
      "stories":"<Collection>",
      "users":"<Collection>"
    }
  },
  "users":{
    "userID":"<String>"{
      "name":"<String>",
      "fullname":"<String>",
      "email":"<String>",
      "age":"<Integer>",
      "teaType":"<String>",
      "birthDate":"<String>",
      "favGenre":"<String>",
      "interest":"<String>",
      "stories":"<Collection>",
      "educators":"<Collection>",
    }
  },
  "stories":{
    "storyID":"<String>"{
      "title":"<String>",
      "author":"<String>",
      "front":"<String>",
      "genre":"<String>",
      "genreImage":"<String>",
      "numPages":"<Integer>",
      "pages":"<Collection>",
      "users":"<Collection>",
    }
  },
  "pages":{
    "pageID":"<String>"{
      "image":"<String>",
      "imageText":"<String>",
      "words":"<Collection>"
    }
  },
  "words":{
    "wordID":"<String>"{
      "word":"<String>",
      "picto":"<String>",
    }
  }
}

```

Figura 8.2: Esquema de datos de la aplicación

Tutors		
Nombre	Tipo	Descripción
<u>tutorID</u>	String	Es la clave primaria de la colección. Este código sirve para identificar unívocamente al tutor.
name	String	Nombre del tutor.
fullname	String	Apellidos del tutor.
phoneNumber	Integer	Número de teléfono del tutor.
tutorType	String	Tipo del usuario, si es tutor o educador.
email	String	Correo del tutor que, posteriormente, utilizará para iniciar sesión en la aplicación.
password	String	Contraseña cifrada que se utilizará en el inicio de sesión.
users	Collection	Colección de los usuarios que tiene asignados.
stories	Collection	Colección de las historias que ha creado.

Tabla 8.1: Atributos de la colección *Tutors*

8.3.2 Colección de educadores

La colección *Educators* representa a los usuarios educadores de los niños y niñas con **TEA**, quienes tendrán la opción de crearles los cuentos. La información guardada en esta colección será la misma que la de la colección anterior 8.3.1, a excepción de la clave primaria (ver tabla 8.2).

Educators		
Nombre	Tipo	Descripción
<u>educatorID</u>	String	Es la clave primaria de la colección. Este código sirve para identificar unívocamente al educador.
name	String	Nombre del educador.
fullname	String	Apellidos del educador.
phoneNumber	Integer	Número de teléfono del educador.
tutorType	String	Tipo del usuario, si es tutor o educador.
email	String	Correo del educador que posteriormente utilizará para iniciar sesión en la aplicación.
password	String	Contraseña cifrada que se utilizará en el inicio de sesión.
users	Collection	Colección de los usuarios que tiene asignados.
stories	Collection	Colección de las historias que ha creado.

Tabla 8.2: Atributos de la colección *Educators*

8.3.3 Colección de Usuarios

La colección *Users* almacena la información referente a los niños o niñas con TEA que utilicen la aplicación. Se guardará información identificativa que luego se mostrará dentro de la aplicación, como su edad, el grado de su trastorno, su nombre y sus apellidos (ver tabla 8.3).

Users		
Nombre	Tipo	Descripción
<u>userID</u>	String	Es la clave primaria de la colección. Este código sirve para identificar al niño o la niña.
name	String	Nombre del niño o la niña.
fullname	String	Apellidos del niño o la niña.
email	String	Correo, será opcional.
age	Integer	Edad del niño o la niña.
teaType	String	Tipo de TEA o trastorno que manifiesta.
birthDate	String	Fecha de nacimiento.
favGenre	String	Género de lectura preferido.
interest	String	Intereses del niño o la niña.
educators	Collection	Colección de los educadores y tutores a los que están asignados.
stories	Collection	Colección de los cuentos que tienen asignados.

Tabla 8.3: Atributos de la colección *Users*

8.3.4 Colección de Cuentos

Esta colección, llamada *Stories*, contiene la información de los cuentos creados por los educadores o tutores. A su vez, contendrá la subcolección de las páginas (*pages*) que forman parte del cuento. Dentro de los datos que contiene esta colección, están los datos de la portada, su título, el autor, el género y la imagen que representará el género en la vista. En este último atributo y el de la portada se guardaran las direcciones de su correspondiente servicio (API de ARASAAC o Cloud Storage) en forma de *string*, para posteriormente poder acceder a él y mostrarlo en la vista (ver tabla 8.4).

8.3.5 Subcolección de páginas

La subcolección *Pages* guardará la información de las diferentes páginas que forman un cuento. En ella, se guardará a su vez la subcolección de palabras (*words*) que formarán la frase de la historia para esa página y la imagen asociada a esa página (ver tabla 8.5).

Stories		
Nombre	Tipo	Descripción
<u>storyID</u>	String	Es la clave primaria de la colección. Este código sirve para identificar unívocamente el cuento.
title	String	Título del cuento.
author	String	Autor del cuento.
front	String	Dirección del API de ARASAAC de un pictograma o de una imagen del módulo Cloud Storage que representa la portada.
genre	String	Género dentro del que se encuentra el cuento.
genreImage	String	Dirección del API de ARASAAC de un pictograma o de una imagen del módulo Cloud Storage que representa el género.
numPages	Integer	Número de páginas que tiene el cuento.
users	Collection	Colección de los usuarios a los que está asignado el cuento.
pages	Collection	Colección de páginas que contiene el cuento.

Tabla 8.4: Atributos de la colección *Stories*

Pages		
Nombre	Tipo	Descripción
<u>pageID</u>	String	Es la clave primaria de la subcolección. Este código nos servirá para identificar cada página dentro de cada historia.
image	String	Imagen asociada a la página.
imageText	String	Texto que se reproducirá una vez que se pulse sobre la imagen asociada.
words	Collection	Colección de palabras que forman la frase en cada página.

Tabla 8.5: Atributos de la subcolección *Pages*

8.3.6 Subcolección de palabras

La subcolección *Words* guardará las palabras que forman el cuento y las imágenes o pictogramas en caso de haber sido asociadas. El atributo *picto* guardará la dirección del pictograma en el API de ARASAAC o del módulo Cloud Storage, en caso de que sea una imagen del dispositivo (ver tabla 8.6).

Words		
Nombre	Tipo	Descripción
wordID	String	Es la clave primaria de la subcolección. Este código nos servirá para identificar las palabras en cada página.
word	String	Texto que representa la palabra en sí misma.
picto	String	Dirección que referencia en el API de ARASAAC (ver 4.4) el pictograma asociado o la dirección del la imagen del módulo Cloud Storage (ver 9.1).

Tabla 8.6: Atributos de la subcolección *Words*

8.4 Estructura del proyecto

La estructura y organización de archivos y directorios de nuestro proyecto se ha realizado siguiendo los diferentes componentes del patrón de diseño utilizado, *MVVM*. Por ello, como se puede observar en la figura 8.3, los diferentes archivos del proyecto se han distribuido en tres directorios: *Model*, *View_Model* y *View*. El directorio ***Model*** contiene las clases que representan las diferentes entidades de base de datos y los archivos que almacenan el código referente al servicio de autenticación, *auth_service.dart*, y a la implementación de las diferentes funcionalidades, *database.dart*. El directorio ***View_Model*** contiene los archivos que realizan la comunicación con la vista y el modelo, en este caso separados según las tres partes diferenciadas de la aplicación: la parte de gestión de usuarios, de gestión de cuentas y de la lectura de estos cuentas. Por último, en el directorio ***View*** se encuentran los archivos que representan las diferentes partes de las pantallas de la aplicación y los *widgets* utilizados en ellas.

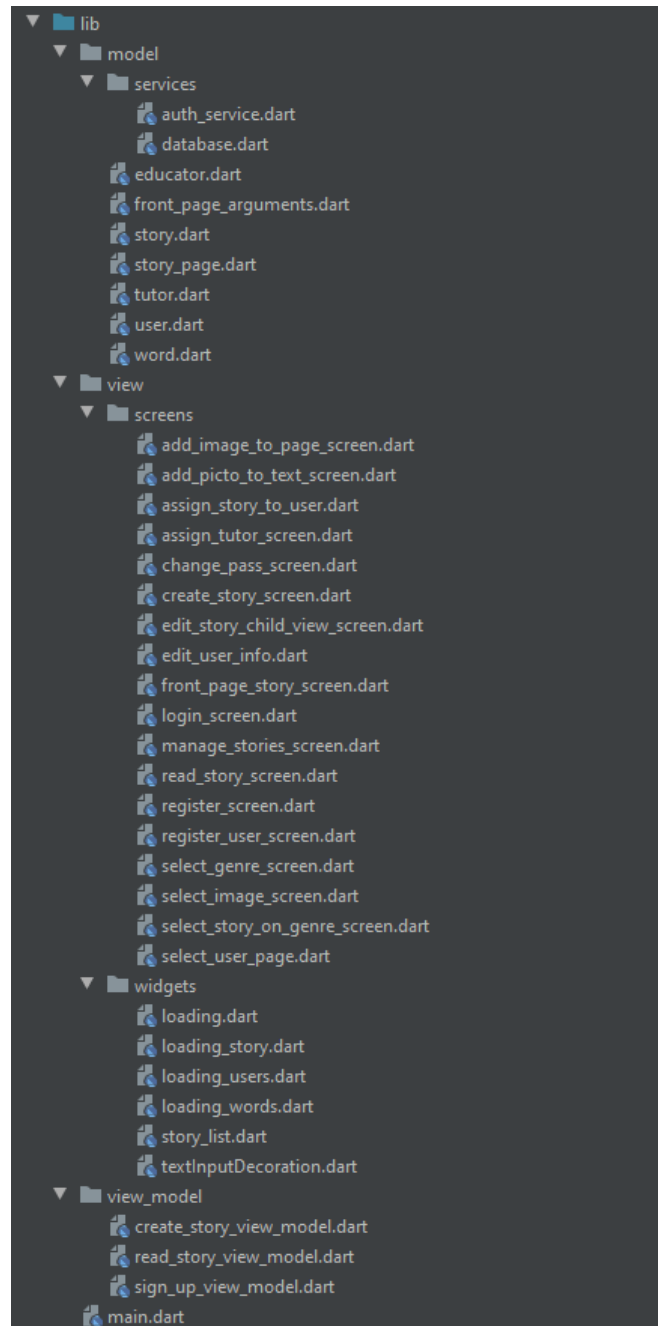


Figura 8.3: Estructura y organización del proyecto

Implementación y pruebas

EN este capítulo de la memoria, se definirán los aspectos más relevantes de la implementación de la aplicación y se expondrán las diferentes pruebas que se realizaron sobre la misma.

9.1 Implementación de las diferentes funcionalidades donde usamos Firebase

Firebase es una de las herramientas fundamentales para este desarrollo, como se vio en el capítulo 4, ya que permitirá implementar el *backend* y la base de datos de esta aplicación. En esta sección, se comentará cómo y para qué se han utilizado los diferentes módulos que proporciona este *framework*. Para ello, a continuación, se expondrá un ejemplo de cada una de las distintas herramientas de Firebase que se han utilizado:

- **Firestore Authentication** se encargará de la gestión de usuarios, utilizando funciones predefinidas en la propia librería del módulo. Su uso permitió simplificar y ahorrar código en la implementación de las funcionalidades que habitualmente se requieren para una correcta gestión de los usuarios.

A continuación, se muestra el código correspondiente a la funcionalidad de **inicio de sesión**, en el que se puede observar que se llama a la función `signInWithEmailAndPassword` de la librería, a la cual se le pasan dos parámetros que proporciona la vista de la aplicación.

```
1 Future signInWithEmailAndPassword(String email, String  
   password) async {  
2   try{
```

```
3     AuthResult result = await
    _auth.signInWithEmailAndPassword(email: email, password:
password);
4     FirebaseUser user = result.user;
5     return user.uid;
6   } catch(e) {
7     print(e.toString());
8     return null;
9   }
10 }
```

- El módulo de **Cloud Firestore** será el encargado del almacenamiento y gestión de la base de datos. Con este módulo, se implementarán las diferentes funciones que guardarán, obtendrán y borrarán datos de la aplicación. Estas funcionalidades se componen de un código más complejo que el del módulo anterior, dado que trabajan con una mayor cantidad de información y se desarrollan casos de uso más complejos.

Por ejemplo, a continuación se puede observar el código necesario para obtener la información relativa a un cuento que se quiere mostrar.

```
1 // Parsea los datos y los convierte en una entidad Story
2 Story _storyDataFromSnapshot(DocumentSnapshot snapshot) {
3   return Story(
4     title: snapshot.data['title'],
5     author: snapshot.data['author'],
6     genre: snapshot.data['genre'],
7     storyID: snapshot.data['storyID'],
8     numPages: snapshot.data['numPages'],
9     front: snapshot.data['front'],
10    genreImage: snapshot.data['genreImage']);
11 }
12
13 // Obtiene la información de cuento específico
14 Future<Story> getStoryData(String storyID) async {
15   var story = null;
16
17   await storiesCollection.document(storyID).get().then(
18     (DocumentSnapshot documentSnapshot) =>
19       {story = _storyDataFromSnapshot(documentSnapshot)});
20
21   return story;
22 }
```

- Dentro de este mismo módulo (**Cloud Firestore**), también se almacenarán los diferentes usuarios y cuentas de la aplicación. Los identificadores de estas entidades se calcularán de una forma diferente dado que, en el caso de los tutores y educadores, se utilizará el identificador único que devuelve la función de registro de **Firebase Authentication**. Por el contrario, como los niño/as y los cuentos no se registran dentro de Firebase, se utilizará un paquete de Dart y Flutter llamado *uuid*, que genera identificadores de forma unívoca y aleatoria, para los usuarios, y basado en el tiempo, para los cuentos. A continuación, se expondrá el ejemplo de la acción de guardar los tutores en la base de datos, la cual es prácticamente idéntica al del guardado de los educadores.

```
1 // Guarda un tutor en BD
2 Future setTutorDataOnRegister(String id, Tutor tutor) async {
3   try {
4     return await tutorsCollection.document(id).setData({
5       'tutorID': id,
6       'name': tutor.name,
7       'fullname': tutor.fullname,
8       'email': tutor.email,
9       'password': generateMd5(tutor.password),
10      'phoneNumber': tutor.phoneNumber,
11      'tutorType': tutor.tutorType
12    });
13   } catch (e) {
14     print('ERROR: $e');
15   }
16 }
```

Como se puede ver en este código de inserción en base de datos, las contraseñas de los diferentes usuarios serán cifradas con [Message-Digest Algorithm 5 \(MD5\)](#) para ser almacenadas de forma segura mediante el uso de la función *generateMd5* que utiliza el paquete *crypto.dart* de Flutter.

- Por otro lado, **Cloud Storage** almacenará las imágenes de la galería del dispositivo y las fotografías de la cámara, utilizadas en la gestión y lectura de cuentos. Las imágenes serán almacenadas en este módulo junto a toda su información, y para su identificación se utilizará el *path* que tiene este archivo en el dispositivo. Para dar soporte a esta funcionalidad, se implementó el siguiente código.

```
1 // Función para guardar imágenes
2 Future<String> uploadFile(File _image) async {
3   String returnUrl;
4 }
```

```

5   StorageReference storageReference =
      FirebaseStorage.instance.ref().child('images/${_image.path}');
6   StorageUploadTask uploadTask = storageReference.putFile(_image);
7   await uploadTask.onComplete;
8
9   await storageReference.getDownloadURL().then((fileURL) {
10      returnURL = fileURL;
11  });
12  return returnURL;
13 }

```

- Por último, con el módulo **Cloud Functions** se implementó el borrado recursivo de documentos y colecciones de Firestore ya que, con un borrado simple, no se eliminaban las subcolecciones que contenían las colecciones principales y quedaba información “basura” en base de datos. Las funciones de este módulo deben ser implementadas en JavaScript o TypeScript y luego deben ser desplegadas en la herramienta. En este caso, se ha implementado una función de borrado recursivo en JavaScript que fue desplegada en Firebase Cloud Functions para, posteriormente, realizar una llamada [HTTP](#) desde el código de la aplicación y utilizarla para realizar el borrado de cuentos.

```

1 // Borra recursivamente un cuento
2 void delete(String storyID) async {
3   var path = '/stories/' + storyID;
4   var data = {'path': path};
5   var deleteFn = CloudFunctions.instance
6     .getHttpsCallable(functionName: 'recursiveDelete');
7
8   await deleteFn.call(data);
9 }

```

9.2 Uso del API ARASAAC

Una de las partes fundamentales de este proyecto es el uso de los pictogramas. Estos recursos nos permitirán crear cuentos mucho más atractivos y motivadores para estos niños y niñas y, de esta forma, poder mejorar su aprendizaje y comunicación. Además, el uso de soportes visuales es fundamental en el caso de niños/as que tienen problemas para comprender el lenguaje escrito o que están trabajando en áreas específicas de desarrollo del lenguaje.

Para conseguir integrar los pictogramas en la aplicación, se hizo uso del API de [ARASAAC](#), con el que se puede acceder a una colección de pictogramas, los cuales son utilizados

por la mayoría de asociaciones que trabajan con este colectivo y en las aplicaciones actuales enfocadas a atender sus necesidades. Este [API](#) se encuentra en una versión bastante temprana, 0.0.1, pero ofrece el servicio de obtención de pictogramas que necesitamos para nuestra aplicación.

Esta aplicación hará uso de este servicio al que se harán peticiones de pictogramas por palabras claves. Estas peticiones se realizarán mediante [HTTP](#), de forma que devuelve los pictogramas que tiene almacenados en el servidor de [ARASAAC](#). Esto se realizó de esta manera para evitar almacenar los pictogramas usados en el almacenamiento interno del dispositivo o en el módulo de Firebase, Cloud Storage, que dispone de espacio gratuito limitado. Esta petición devolverá un resultado en formato [JSON](#), del cual se obtendrá el identificador que almacenaremos en una lista. Posteriormente será obtenida para mostrar todos los pictogramas obtenidos en la pantalla de búsqueda (ver figura 9.1). Una vez seleccionado un pictograma, se guardará la dirección [URL](#) del [API](#) de [ARASAAC](#) junto al identificador del pictograma, para poder realizar una petición al servidor en el momento en que se precise mostrarlo. A continuación, se muestran las líneas de código de la búsqueda de pictogramas:

```
1 // Obtiene los ids de los pictogramas
2 Future<List<String>> getPictogramsByKeywords(String keywords)
  async {
3   List<String> idsList = List<String>();
4   if (keywords != null && keywords != '') {
5
6     var picto = jsonDecode((await http.get(
7
8     'https://api.arasaac.org/api/pictograms/es/search/$keywords'))
9     .body);
10
11    for (int i = 0; i < picto.length; i++) {
12      idsList.add(picto[i]['_id'].toString());
13    }
14  } else {
15    return null;
16  }
17  return idsList;
18 }
```

9.3 Uso de imágenes del propio dispositivo

La personalización es una de las características principales de esta aplicación. Por ello, se decidió permitir utilizar imágenes personales de la galería y fotografías tomadas con la cámara

del dispositivo. Esta funcionalidad se desarrolló para la creación de los cuentos, donde se pueden crear historias de la vida cotidiana de esta persona con TEA e introducir estas imágenes para que le resulte más familiar y cercana. Esta característica se proporciona como alternativa o complemento a los pictogramas de ARASAAC, mencionados en la sección anterior 9.2.

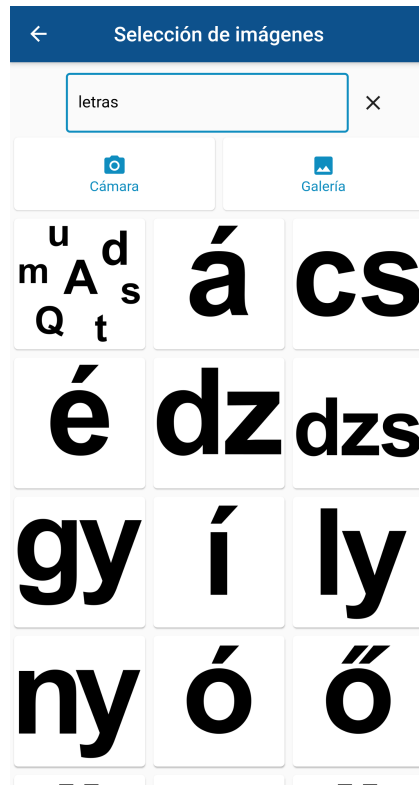


Figura 9.1: Funcionalidad de selección de imágenes o búsquedas de pictogramas

Como podemos observar en la figura 9.1, a la hora de añadir una imagen en el cuento, se permite seleccionar cualquiera de las tres opciones. Por un lado, se encuentran las opciones de seleccionar imágenes del propio dispositivo mediante la galería o la cámara y, por otro, mediante el buscador de pictogramas.

Para implementar esta funcionalidad, ha sido necesario añadir varios permisos para el sistema operativo de iOS. Esto es debido a que la forma de gestionar el acceso a los archivos, en este caso las imágenes, es diferente dependiendo del sistema operativo móvil donde es ejecutado. En este caso, se han añadido en el archivo `/ios/Runner/Info.plist` de este proyecto Flutter las siguientes líneas de petición de permisos de acceso a la galería y a la cámara del dispositivo:

```

1 <key>NSPhotoLibraryUsageDescription</key>
2 <string>Allow access to photo library</string>
3 <key>NSCameraUsageDescription</key>

```

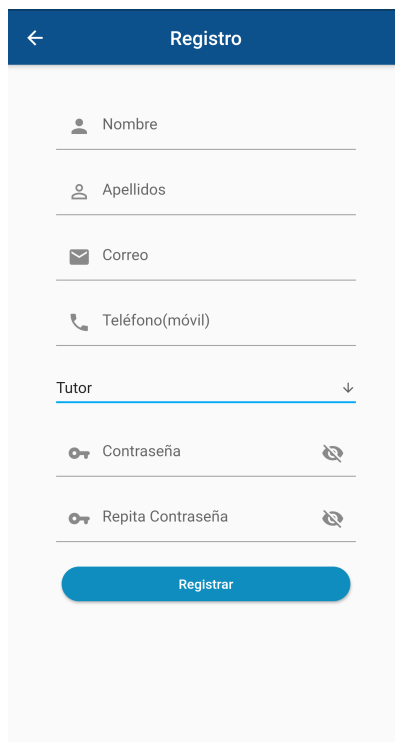


```
4 <string>Allow access to camera to capture photos</string>
```

En Android, la solicitud de permisos de acceso a archivos está soportada de forma nativa a partir de la versión 10. En el caso de que la aplicación sea utilizada en versiones anteriores, habría que añadir `android:requestLegacyExternalStorage="true"` como un atributo `<application>` en el fichero `AndroidManifest.xml`.

9.4 Implementación de gestión de usuarios

La gestión de usuarios, como se comentó anteriormente en la memoria, se organiza en función de los tres tipos de usuarios en el sistema: los tutores, los educadores y los niños o niñas que utilicen la aplicación. Los niños o niñas no realizarán registro dentro de la aplicación por ellos mismos, ya que lo realizará su tutor. Por lo tanto, estos usuarios solo podrán acceder a la lectura de los cuentos. En el caso de los tutores y educadores, realizarán su registro en la pantalla de inicio de sesión de la aplicación. En este formulario, mostrado en el figura 9.2, seleccionarán el tipo concreto de usuario y cubrirán sus datos de registro. Como se puede observar, el formulario sigue un diseño similar al realizado en la fase de prototipado (apéndice D).



The image shows a mobile application registration screen. At the top, there is a blue header bar with a white back arrow on the left and the word 'Registro' in white text on the right. Below the header, the form consists of several input fields, each with a small icon to its left: 'Nombre' (person icon), 'Apellidos' (person icon), 'Correo' (envelope icon), and 'Teléfono(móvil)' (phone icon). Below these is a dropdown menu labeled 'Tutor' with a downward arrow. The next two fields are 'Contraseña' and 'Repita Contraseña', both with a key icon and a visibility toggle icon (an eye with a slash) to their right. At the bottom of the form is a large, rounded blue button with the white text 'Registrar'.

Figura 9.2: Formulario de registro en la aplicación

Después de su registro e inicio de sesión en la aplicación, se observarán los diferentes permisos que tiene cada usuario. En el caso del tutor, se ha decidido que tenga permisos para dar de alta niños o niñas dentro del sistema, modificar sus datos en caso de que sea necesario y asignarlos a otros tutores o educadores para que puedan trabajar con ellos. Esta división de permisos se debe a que se quiere que la gestión de los datos de un niño o niña recaiga sobre un único responsable y, con ello, proporcionar también una capa de seguridad. Por otro lado, los educadores solo podrán ver los datos y gestionar cuentos de los niños o niñas que tengan asignados. En las siguientes figuras 9.3a y 9.3b, se muestran las vistas que tienen un tutor y un educador tras iniciar sesión y las diferentes acciones que pueden realizar en cada caso.

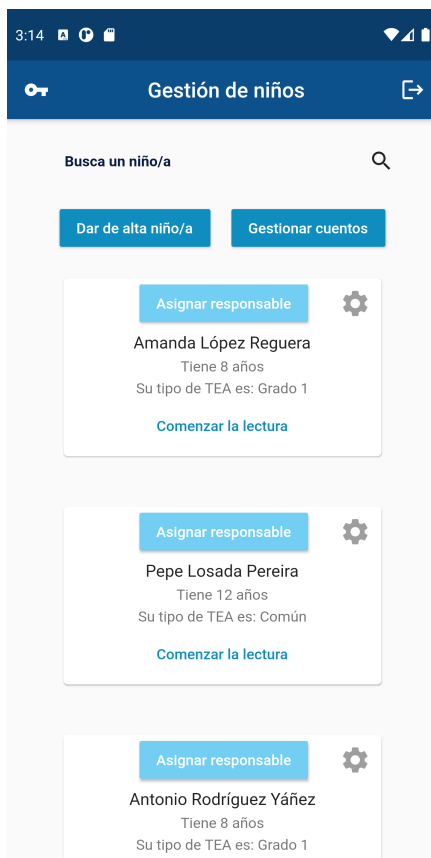
Es importante aclarar que si un usuario quiere dar de alta a un niño/a, necesita registrarse en la aplicación como tutor. De esta forma, en la base de datos quedará almacenado con ese rol y podrá gestionar la información de niños y niñas que estén a su cargo y crear cuentos personalizados para ellos. Sin embargo, esto no impide que ese usuario tutor puede ser asignado a otros niños/as (que no estén a su cargo) y desempeñar en ese caso un rol de educador. El ejemplo más claro es un profesional que tiene un hijo/a con TEA u otro tipo de diversidad. En ese caso, se registrará como tutor para poder gestionar a su hijo/a, pero también trabajará con otros niños/as de los que no es responsable. En el primer caso, puede gestionar todo lo relacionado con su hijo/a mientras que, en el segundo caso, sólo podrá crear cuentos para trabajar con los otros niños/as que tiene asignados. La aplicación ha sido desarrollada para dar soporte a este tipo de situaciones si bien, en la mayoría de los casos, los roles de tutor y educador estarán claramente separados y no habrá este tipo de solapamientos.

9.5 Implementación de gestión de cuentas

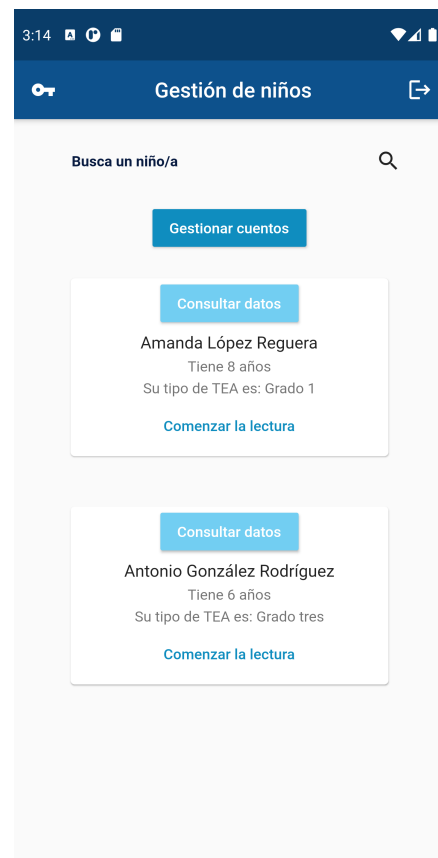
La tarea principal de los tutores y educadores es la gestión de cuentas, y serán ellos los encargados de crear estos cuentos personalizados para los niños y niñas con TEA. Conforme a sus conocimientos, ya sea porque es un familiar del niño/a o es el o la especialista del mismo, podrá crear historias más personales, orientadas a los intereses del niño/a, o con el objetivo de trabajar algún aspecto de su aprendizaje. En cuanto a la edición o borrado del cuento, se proporcionan estas alternativas para poder modelar y adaptar los cuentos de una forma correcta según la reacción del niño/a.

9.5.1 Creación de cuentas

La **creación de cuentas** consta de tres pasos: la elección de datos generales y la importación del cuento, la asignación de pictogramas a palabras y la asignación del cuento a los niños/as.



(a) Pantalla principal del tutor.



(b) Pantalla principal del educador.

Figura 9.3: Pantallas principales del tutor y del educador

Elección de datos generales e importación

En este primer paso, se seleccionarán los datos principales del cuento, como la portada, género, autor y título (ver subfigura 9.4a). En el caso de la portada, se seleccionará un pictograma o una imagen desde una pantalla como la de la figura 9.1, que vimos anteriormente. Esta elección debe ser adecuada para captar la atención del niño/a para que desee seleccionar el cuento cuando se trabaje con la aplicación. La selección del pictograma correspondiente al género establecido se realiza conforme al nombre introducido en esta pantalla para el género al que pertenece el cuento, con el que se realizará una búsqueda en el [API ARASAAC](#) y se guardará el primer pictograma que coincida con la palabra. En caso de que no coincidiese ninguno, se establece un pictograma por defecto.

La importación del cuento se realizó mediante la ayuda del paquete de Flutter *file_picker*, con el que se accede al sistema de archivos del dispositivo. Se decidió limitar este acceso a archivos *txt*, dado que es el tipo de archivo más fácil de manipular. Este fichero deberá tener un formato predefinido, cada línea del archivo será una página del cuento y podrá contener un máximo de diez palabras, para no crear frases extensas que puedan llegar a distraer al lector. Este formato será facilitado al usuario de la aplicación mediante un botón de ayuda.

Tras importar el fichero, se recorre el *txt* mediante la función *readAsLines* del paquete *File* de Dart. Como se comentó en el párrafo anterior, esto se realiza para dividir las líneas del fichero de texto en las páginas del cuento. Tras dividir las se almacenarán cada una como un documento de Firestore en la subcolección *pages*. Dentro de cada página se trocearán las palabras mediante la función *split*, haciendo uso de los espacios entre cada una, y se guardarán cada una en un documento de la subcolección *words*. Este último paso se realiza para poder asignar en un futuro un pictograma específico a cada palabra individualmente, como podemos observar en la subfigura 9.4b. A continuación, se representa el código de este paso.

```

1  try {
2      pages = await file.readAsLines(encoding: utf8);
3  } catch (e) {
4      print('ERROR: Fallo en la lectura del archivo');
5  }
6
7  if (pages != null) {
8      for (int i = 0; i < pages.length; i++) {
9          var page = pages.elementAt(i).split(' ');
10         await _createStoryViewModel.addPageToStory(page, (i +

```

```
1), storyID);  
11     }  
12 }
```

Como ocurría en el caso del acceso a la cámara, para iOS se tuvieron que añadir de forma explícita permisos en el fichero `/ios/Runner/Info.plist` para que el componente `file_picker` pudiese acceder al sistema de archivos. En este caso, se añadieron las siguientes líneas:

```
1 <key>UIBackgroundModes</key>  
2 <array>  
3 <string>fetch</string>  
4 <string>remote-notification</string>  
5 </array>  
6 <key>NSAppleMusicUsageDescription</key>  
7 <string>Allow access to music files</string>
```

Asignación de pictogramas a las palabras

En este paso de la creación, se seleccionarán las palabras que se quieren mostrar con pictogramas en la lectura (ver subfigura 9.4b). Esto se realizará asociando una imagen o un pictograma desde la pantalla de la figura 9.1 a la palabra seleccionada. Después de este paso, se puede observar que se ha añadido correctamente la imagen o eliminarla en caso de equivocación. En esta pantalla, también se realizará la asignación de un pictograma o de una imagen a una página del cuento, acción la cual es opcional. El código de la asignación, tanto del pictograma a la página como a la palabra, será sencillo, y se realizará haciendo uso de la función de Firestore, `updateData`, a la que se le pasará el pictograma o la imagen a guardar.

Asignación del cuento

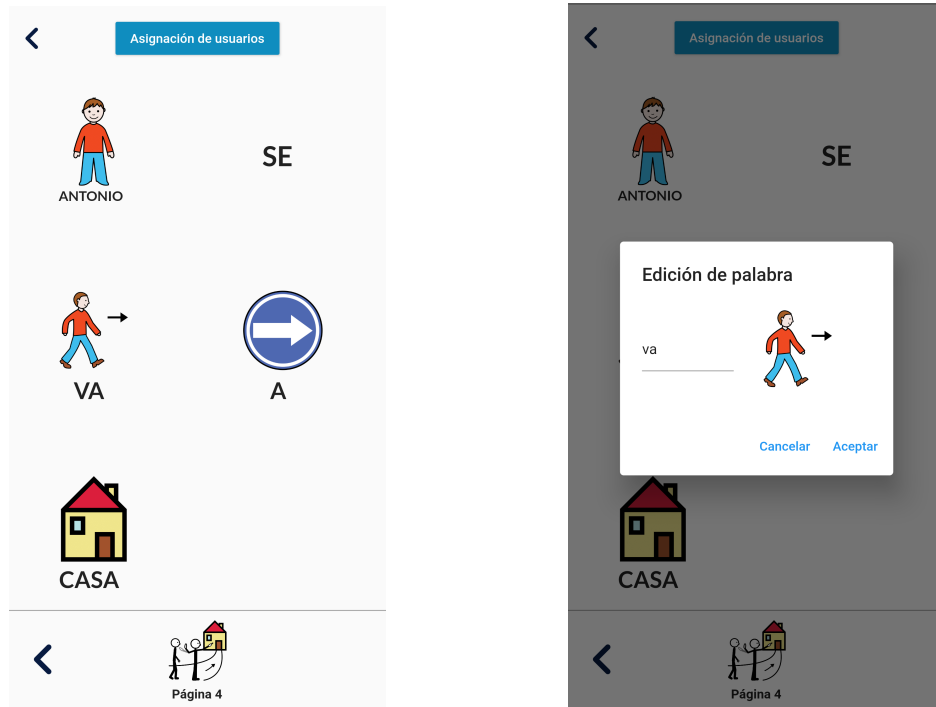
La asignación del cuento a uno o varios niños/as será el último paso en la creación del cuento. En este paso, seleccionaremos mediante un buscador el niño o la niña al que se le quiere asignar el cuento que se acaba de crear. Podemos ver este paso representado en la subfigura 9.4c.



Figura 9.4: Pantallas con los diferentes pasos de la creación

9.5.2 Edición de cuentos

Las pantallas para la edición de cuentos serán las mismas que en la fase de creación, con la diferencia de que los campos de texto estarán cubiertos y que se permite editar de dos formas: como en la creación, es decir, de manera manual, o desde la vista del niño, para poder ver los cambios en tiempo real. Esta vista para el niño o la niña será horizontal, pero en la edición se ha decidido que sea vertical ya que su manejo y edición es mucho más cómodo y sencillo. Este proceso de edición se ilustra en la figura 9.5 donde se observa que, al seleccionar una palabra, la aplicación permitirá modificar esa palabra y añadirle un pictograma o, en caso de que tuviese ya uno asignado, podría ser modificado o eliminado.



(a) Pantalla de edición (vista de lectura).

(b) Modificación de una palabra.

Figura 9.5: Pantallas de edición desde la vista de lectura.

9.5.3 Borrado de cuentos

La funcionalidad de borrado de cuentos se realizará desde la pantalla general de gestión de cuentos. En esta vista, disponemos de varias opciones como acceder, editar o borrar un cuento. Si se pulsa el botón de eliminar cuento, se mostrará un diálogo de confirmación que, tras aceptarlo, llamará a la función recursiva de borrado comentada en el apartado de las Cloud Functions en la sección 9.1. Esta función borrará el documento correspondiente al cuento en base de datos, tanto de su creador como de los niños a los que esté asignado.

9.6 Implementación de la lectura de cuentos

El desarrollo de la lectura de los cuentos es la parte fundamental de la aplicación y la que utilizarán los niños y las niñas con TEA. El diseño de esta parte ha cambiado bastante con respecto al prototipado inicial de la aplicación, en el que se presentaba la imagen representativa de la página con un tamaño considerable. Sin embargo, se decidió reducir significativamente su importancia en la vista del cuento, dado que centraría la atención del niño/a en la figura y no en el cuento, que es lo que se busca. También se decidió cambiar el formato de elección de cuentos para que el niño/a primero seleccione un género y, dentro del mismo, uno de los

cuentos, mostrando las imágenes de ambos con un tamaño adecuado. Tanto en el título de la selección de cuentos y géneros como en la lectura de la historia, se mostrarán las palabras en mayúsculas y negrita con una tipografía bien definida, para adaptarla a las personas con TEA y no lleguen a tener ningún tipo de confusión. Podemos observar este último detalle en la figura 9.6.



Figura 9.6: Pantallas del niño/a para seleccionar género y cuentos

Dentro de la lectura de cuentos, se permite elegir entre leer el cuento de forma manual o mediante la audio-lectura, según lo que se desee trabajar con el niño o la niña en cada momento. En ambas opciones, se mostrará el texto de la página y los pictogramas asociados a palabras con un tamaño destacable. Lo que al principio supuso un problema, ya que la longitud de las palabras de la página era muy variable y dependiendo del tamaño de pantalla del dispositivo no funcionaba correctamente. Por ello, se decidió utilizar el *widget* de Flutter, *GridView*, que nos permite controlar cuántas palabras mostrar por cada línea de la página, para una correcta lectura se estableció en cinco. En el caso de los términos de extensa longitud, se redujo su tamaño para poder adaptarlo al espacio predeterminado del *Grid*.

Por otro lado, en la parte inferior de la pantalla se muestra una imagen seleccionada por el creador del cuento para representar el sentido de la frase en la página. Principalmente, se intenta practicar con ella las emociones del niño o la niña utilizando pictogramas que representen esta acción conforme al texto del cuento. Esta última característica será opcional, a elección del creador, por lo que puede haber páginas en las que no se muestre. Se expone la vista correspondiente a una página de ejemplo en un cuento en la figura 9.7.



Figura 9.7: Pantalla de lectura del cuento

Por último, una de las funcionalidades que se han implementado en la lectura de cuentos es la reproducción de las palabras y pictogramas cuando se pulsa sobre ellos. Se ha utilizado un paquete de Flutter llamado *flutter_tts* que reproduce textos mediante una voz, algo robótica, pero con una entonación limpia y un sonido claro a la hora de reproducir las palabras. El módulo que se ha utilizado permite modificar parámetros del audio, como la velocidad, el tono, el volumen o el lenguaje. Para iOS, se deberán añadir a mayores las siguientes líneas de código para activar la compartición de audio y establecer las opciones y categorías del mismo.

```

1  await flutterTts.setSharedInstance(true);
2  await flutterTts
3
4  .setIosAudioCategory(IosTextToSpeechAudioCategory.playAndRecord,
5  [
6      IosTextToSpeechAudioCategoryOptions.allowBluetooth,
7      IosTextToSpeechAudioCategoryOptions.allowBluetoothA2DP,
8      IosTextToSpeechAudioCategoryOptions.mixWithOthers
9  ]);

```

En la parte de la audio-lectura, se utilizará una función que recorre el texto de la página y reproduce las palabras de una forma pausada y sencilla para que el niño/a pueda comprenderlo. Si esta página tuviese más de una línea de lectura y no se viese reflejada en la pantalla, la audio-lectura desplazará la pantalla automáticamente hacia abajo una vez llegue a la última palabra mostrada en la pantalla. De esta forma, se garantiza que el niño/a siempre está visualizando la línea correcta con las palabras que se están leyendo en cada momento. La función

que se encarga de implementar esta funcionalidad se puede ver en el siguiente código.

```

1 // Función para la reproducción en la audio-lectura
2 if (currentPageInfo != null) {
3     audioCuento = '';
4     for (int i = 0; i < currentPageInfo.length; i++) {
5         await speak(currentPageInfo.elementAt(i).word);
6         if (i == 4 && !tablet) {
7             _gridController.animateTo(200,
8                 duration: Duration(milliseconds: 400),
9                 curve: Curves.easeInOut);
10        }
11    }
12 }

```

9.7 Pruebas

En este apartado, se detallarán las diferentes pruebas realizadas para la aplicación. Se dividirán en funcionales, donde se realizarán pruebas unitarias, de integración, de aceptación y de revisión de *sprint*, y no funcionales, en las cuales encontraremos, pruebas de usabilidad y de compatibilidad.

9.7.1 Pruebas funcionales

Pruebas unitarias

Las pruebas unitarias validan el correcto funcionamiento de un bloque concreto de código. Se realizaron pruebas de cada funcionalidad por separado, antes de integrarlas entre ellas. Entre estas pruebas, destacó la de lectura de los cuentos, en la que nos encontramos el problema de que, con palabras demasiadas largas, en ciertas ocasiones la palabra podía aparecer cortada en la vista de lectura. Este problema se solucionó utilizando otro tipo de *widget* en Flutter, el cual permitía pasar las palabras hacia otra línea, y se decidió reducir el tamaño de fuente para este tipo de palabras. Por otro lado, se observó que con el borrado simple de cuentos, si la colección contenía subcolecciones, se quedaban documentos "basura" en base de datos. La solución a este problema fue la utilización de las Cloud Functions para realizar un borrado recursivo. A parte de estos pequeños problemas, las demás pruebas fueron satisfactorias.

Pruebas de integración

Las pruebas de integración se encargan de verificar que los diferentes componentes y funcionalidades de la aplicación se comuniquen bien entre ellas y funcionen de una forma

correcta. En este caso, se realizaron tras finalizar las pruebas unitarias de la aplicación y consistieron, por ejemplo, en comprobar que si la creación de un cuento finaliza correctamente, posteriormente se pueda realizar su lectura. Una de las pruebas a destacar fue la de integración de la creación de cuentos con la pantalla de gestión de cuentos, la cual, no se actualizaba y cuando se retrocedía en el dispositivo no funcionaba correctamente. Esto fue un problema en el manejo de la navegación entre pantallas que se solucionó mediante el uso de una función de la clase *Navigator*. Las demás pruebas que se realizaron en esta parte del proyecto fueron la mayoría exitosas, sin ningún inconveniente destacable.

9.7.2 Pruebas de revisión de *sprint*

En la revisión de cada *sprint*, se realizaron las pruebas necesarias para comprobar el funcionamiento de las funcionalidades implementadas durante el mismo. Al final del quinto *sprint*, se proporcionó a los tutores y a los profesionales de la asociación colaboradora una primera versión de la aplicación (ver acta de la reunión en el apéndice B). Tras realizar las pruebas correspondientes, se encontraron algunos detalles que se comentaron al estudiante con el objetivo de mejorar la aplicación y que este resolvió tras la reunión como, por ejemplo: un error de duplicados en la creación de cuentos, el hecho de que la creación de cuentos era muy poco intuitiva y no contaba con ayuda, y un problema en la lectura de cuentos. Por otro lado, se realizaron aportaciones desde la asociación para mejorar la aplicación y se recomendaron nuevas funcionalidades para la misma. Estas aportaciones se tuvieron en cuenta y se llevaron a cabo alguna de ellas: la edición desde la vista de lectura, la ayuda al importar cuentos, la opción de cambiar la contraseña del usuario y la posibilidad de modificar los datos de los niños y las niñas.

Pruebas de aceptación

Las pruebas de aceptación determinan si se han cumplido los requisitos definidos en la fase de análisis y, con ello, si se cubren las necesidades del usuario.

En esta fase, se realizó una prueba de aceptación con la entidad con la que se trabajó en el proyecto. El acta de esta reunión se puede ver reflejada en el apéndice C. Se entregó una versión casi definitiva de la aplicación a los tutores que, debido a la situación actual, fueron los que se la proporcionaron a la asociación. La valoración fue positiva, mostrándose conformes con los cambios incorporados de acuerdo a sus recomendaciones. También se realizaron nuevas sugerencias para ayudar a mejorar la herramienta y su adaptación a las personas a la que está destinada:

- Prescindir de añadir cierta información acerca del niño/a como el tipo de TEA/diversidad.

- Mostrar más *feedback* al usuario.
- Proporcionar la posibilidad de colaboración entre usuarios a la hora de crear los cuentos.
- Añadir una melodía relajante durante la lectura.
- Habilitar actividades de refuerzo para permitir que el niño/a pueda mejorar su aprendizaje en una determinada área de trabajo.

En esta reunión, también se hicieron pruebas preliminares de la aplicación por parte de un niño con TEA, para comprobar que la interfaz resultase intuitiva para él, aunque se comprobó que el niño necesitaba de cierto tiempo para familiarizarse con la aplicación.

9.7.3 Pruebas no funcionales

Estas pruebas se realizaron para comprobar que nuestra aplicación cumpliera los requisitos no funcionales mencionados en el capítulo 7. Se realizaron las siguientes pruebas:

Pruebas de usabilidad

La aplicación se ejecutó en diferentes tipos de dispositivos con diferentes dimensiones y resoluciones. Se probó en dispositivos móviles de como mínimo 5,5 pulgadas y en tabletas de como mínimo 8 pulgadas, en las que, como era previsible, se observó que la lectura de cuentos es más manejable ya que se pueden ver distintas líneas de texto de forma simultánea. Tras realizar estas pruebas, se observaron problemas con los componentes para dispositivos de pequeñas dimensiones, en los que algunas frases se cortaban y se debía desplazar la pantalla. Este problema se resolvió adaptando los componentes de la vista para pantallas pequeñas y que se pudieran trabajar con ellos sin tener que desplazarla.

Pruebas de compatibilidad

En este tipo de pruebas, se probó la compatibilidad de la aplicación con distintos sistemas operativos móviles, Android e iOS. En estas pruebas, se observó que para dispositivos iOS con versión anterior a la 10, nuestra aplicación no funciona correctamente, debido a errores con el acceso al sistema de archivos. En las pruebas con dispositivos Android, se encontraron problemas con un dispositivo Android 7, en el que la aplicación no funcionaba como debería, pero para versiones superiores y en el dispositivo físico con Android 10 funcionó correctamente.

Conclusiones

En este capítulo se expondrán las conclusiones y lecciones aprendidas tras el desarrollo de este proyecto y, a su vez, las posibles mejoras de la aplicación para un futuro.

10.1 Conclusiones

Tras la finalización del proyecto, se puede concluir que la aplicación desarrollada cumple con todos los objetivos definidos al comienzo del mismo.

La realización de este trabajo permitió aplicar los conocimientos técnicos adquiridos a lo largo del grado, incorporando además los que se han aprendido en la realización de este proyecto. Estos conocimientos se han enfocado además a desarrollar una aplicación que puede resultar útil o proporcionar un servicio a un determinado colectivo como es, en este caso, la atención de las dificultades de una persona con TEA u otros trastornos del neurodesarrollo. Por lo tanto, fue necesario realizar un estudio cuidadoso de las diferentes características que presentan estas personas y se analizó la mejor forma de atenderlas con el diseño de nuestra aplicación.

Debido a las buenas valoraciones recibidas por parte de los profesionales de la entidad colaboradora que realizaron las pruebas de la misma, este proyecto intentará seguir mejorando y añadiendo nuevas funcionalidades en el futuro. En este sentido, se busca aportar un recurso básico en la educación de todos los niños y niñas, como son los cuentos, pero de una forma personalizada y adaptada.

A día de hoy, no existe una aplicación en el mercado que cumpla todos los objetivos/-características que reúne la aplicación desarrollada, con lo que se puede concluir que se ha implementado una aplicación móvil que contribuye a la mejora de la calidad de vida de las personas que presentan dificultades en el desarrollo social y cognitivo, aportando una nueva herramienta con potencial para ser empleada por las educadoras y los educadores.

10.2 Lecciones aprendidas

De acuerdo a la metodología educativa aplicada, ApS, el estudiante ha adquirido conocimientos académicos al mismo tiempo que realizaba un aporte a la sociedad. La aplicación de esta metodología ha supuesto algo novedoso para el alumno, pero con unos resultados muy positivos. Ha adquirido conocimientos al mismo nivel en las dos dimensiones del proyecto (académica y social) gracias a ella. En este sentido, me gustaría destacar como aspecto más importante en este tipo de proyectos, la concienciación social que se adquiere mediante la investigación que se realiza para enfocar el desarrollo.

En relación con el servicio realizado, se han tratado las diferentes dificultades con las que se encuentran las personas con TEA en su vida cotidiana, un trastorno que está muy invisibilizado y del que todavía falta mucho por conocer. Durante la realización de este trabajo se ha contado con la ayuda de una asociación que nos ha dado a conocer la escasez de recursos destinados a las personas con TEA. Debido a esto, el objetivo de este trabajo ha sido crear una aplicación para estos niños y niñas, y que puedan ver que este trastorno no debería suponer ninguna limitación.

En el aspecto tecnológico, se ampliaron y adquirieron nuevos conocimientos sobre las tecnologías usadas para el desarrollo móvil, como son Dart y Flutter. A su vez, se han aprendido herramientas novedosas para el desarrollo como Firebase. En el proceso de desarrollo se aprendió a realizar una planificación y aplicar una metodología de desarrollo completa, lo que permitió acercarse a la realización de un proyecto software real.

10.3 Líneas futuras

Tras las últimas reuniones con los tutores y la asociación, se definieron y analizaron algunas mejoras que sería interesante introducir en el futuro en nuestra aplicación. Las mejoras que se han planteado y recogido tras la última reunión con la asociación son las siguientes:

- Permitir introducir audio propio en la reproducción del cuento y las palabras. Con esto, se conseguiría una mayor personalización de los cuentos y una lectura mucho más cercana para el niño o la niña con una voz que le puede resultar más familiar.
- Mejorar la creación de cuentos, habilitando la opción de introducir el texto desde el dispositivo móvil o tableta, sin la necesidad de importar un documento. Por otro lado, en la finalización mostrar un mensaje de correcta creación para proporcionar un mayor *feedback* al usuario.
- Permitir el trabajo colaborativo para la creación y uso de un cuento entre varios usuarios (tutores / educadores) en la aplicación.

- Mejorar la lectura de cuentos mediante la inclusión de una melodía relajante y enlaces con actividades adicionales para reforzar el aprendizaje durante la lectura.

Apéndices

Acta de reunión inicial

A.1 Resumen del acta de reunión inicial

Fecha: 03 de noviembre de 2020.

Asistentes:

- Óscar Fresnedo Arias.
- Tres profesionales de la asociación Dismacor (psicóloga, psicopedagoga, logopeda).

Reunión presencial en la que se abordan las siguientes cuestiones:

- Se analizan posibles necesidades de la asociación que puedan ser cubiertas con el desarrollo de aplicaciones para dispositivos móviles que ayuden a mejorar la calidad de vida de las personas a las que ofrecen terapias (especialmente niños y niñas con TEA) y que puedan resultar útiles tanto en las terapias como en su entorno familiar.
- Se identifican varias necesidades especialmente relacionadas con facilitar la comunicación de los niños y niñas con TEA con su entorno familiar, así como con el desarrollo del lenguaje oral y de la comprensión de conceptos sencillos que pueden contribuir a mejorar su autonomía.
- Se proponen dos aplicaciones que pueden cubrir parcialmente las necesidades identificadas en el punto anterior y se esbozan los requisitos que deberían incorporar en una primera versión funcional con la que comenzar a trabajar.
- Se abordan temas relacionados con el diseño de la interfaz gráfica de la aplicación considerando las características de las personas que se puedan beneficiar de estas aplicaciones, prestando especial atención a las recomendaciones propuestas por los profesionales de la asociación.

- Considerando los requisitos identificados por los profesionales de la asociación y las sugerencias aportadas con respecto a la parte gráfica de la aplicación, se elabora un documento inicial que sirve de base para la elaboración de la documentación correspondiente al anteproyecto de este trabajo.

Acta de reunión de seguimiento

Fecha: 12 de julio de 2021.

Asistentes:

- Óscar Fresnedo Arias.
- Dos profesionales de la asociación Dismacor (logopeda, psicopedagoga).

Reunión presencial de seguimiento en la que se realizaron las siguientes actividades:

- Se presenta a los profesionales de la asociación una primera versión funcional de la aplicación desarrollada en el proyecto.
- Se hace una demostración de las principales funcionalidades de la aplicación, tanto de la parte de administración, creación y personalización de cuentos, más orientada a los padres/educadores, como de la parte orientada a la lectura de los cuentos por los niños y niñas con **TEA**.
- Las profesionales valoran en general la aplicación y, en particular, las distintas funcionalidades, aportando un *feedback* muy valioso para corregir y mejorar ciertos detalles que puedan ayudar a construir una aplicación más usable y que resulte más útil tanto a los educadores/tutores como a niños.

Las principales conclusiones alcanzadas tras la reunión se pueden resumir en los siguientes puntos:

- La valoración global de la aplicación por parte de los profesionales ha sido muy positiva, destacando que la personalización de cuentos puede resultar muy útil para trabajar diferentes aspectos del desarrollo de los niños y niñas con **TEA**. Se ha destacado especialmente la interfaz sencilla y clara de la aplicación, y el hecho de que pueda ser

completamente personalizable la inclusión de pictogramas ya que esto permite adaptarse a los progresos de los niños y niñas con los que se trabaja en las sesiones.

- Además, han destacado lo adecuadas que resultan este tipo de aplicaciones para niños con estas características, ya que representan un estímulo visual mucho más atractivo que los tradicionales cuentos en papel. Por último, se ha valorado positivamente que los pictogramas (o palabras) en el cuento tengan un apoyo auditivo cuando se pulsa en ellos, ya que esto ayuda a los niños y niñas a reforzar su lenguaje oral y la conexión de la palabra con su representación visual y el concepto que representa. En la misma línea, se ha valorado positivamente la opción de incluir un pictograma que represente la emoción asociada a cada página del cuento.
- Por último, se han recogido una serie de sugerencias proporcionadas por los profesionales de la asociación:
 - Añadir algunas indicaciones o instrucciones para guiar el proceso de creación de los cuentos.
 - Mejorar o hacer más intuitiva el menú que permite buscar pictogramas para ser añadidos a determinadas palabras en el cuento.
 - Incorporar una vista similar a la que pueden ver los niños y niñas en la lectura de cuentos para editar de forma más sencilla los cuentos.
 - Hacer más accesible el botón para asignar un cuento a un niño y para finalizar la edición de un cuento.
 - Modificar la vista para que sea más intuitivo el acceso a la parte donde el niño o niña puede leer los cuentos creados.
 - Conseguir que una frase del cuento se pueda visualizar en la misma pantalla, aunque sea necesario hacer algún tipo de *scroll*.
 - Mejorar la voz con la que se reproduce el audio correspondiente a las palabras del cuento, especialmente en la audio-lectura, ya que resulta bastante fría.
 - En esta misma línea, introducir la opción de que un tutor o educador pueda grabar el sonido correspondiente a una palabra con cierta entonación y con una voz familiar.
 - Posibilidad de establecer un conjunto de categorías fijas para los cuentos.

Acta de reunión final

Fecha: 31 de agosto de 2021.

Asistentes:

- Óscar Fresnedo Arias.
- Tres profesionales de la asociación Dismacor (psicóloga, psicopedagoga, logopeda).
- Niño con TEA y su madre.

Reunión presencial en la que se realizaron las siguientes actividades:

- Se presenta a los tres profesionales de la asociación DISMACOR una versión casi definitiva de la aplicación desarrollada con todas sus funcionalidades.
- Se explica en detalle todas las funcionalidades de la aplicación, centrándose especialmente en aquellas que se han incorporado desde la última reunión.
- Los profesionales comentan sus dudas sobre el uso e instalación de la aplicación, que son aclaradas por uno de los tutores del proyecto.
- Los profesionales prueban las diferentes funcionalidades de la aplicación utilizando un móvil Android y una tableta iPad.
- Al final de la reunión, se le muestra la aplicación a un niño con TEA y, bajo la supervisión de su madre y los profesionales, el niño interactúa durante un periodo de tiempo breve con la aplicación.

Una vez completada la reunión se establecen las siguientes conclusiones:

- De nuevo, las profesionales han valorado de forma positiva la aplicación y han expresado que puede resultar muy útil tanto en el ámbito más familiar como a la hora de realizar intervenciones específicas sobre niños con TEA.

-
- De igual modo, han mostrado su conformidad con los cambios incorporados en la aplicación siguiendo sus recomendaciones en la reunión de seguimiento. Además, han sido muy comprensivas con el hecho de que ciertos cambios no se hayan podido incorporar debido a la complejidad técnica que implicarían y que superaría el esfuerzo que el estudiante debe dedicar a su TFG.
 - Además, las profesionales han propuesto una serie de sugerencias relacionadas con las nuevas funcionalidades añadidas en la aplicación desde la última reunión y han matizado algunos detalles sobre las recomendaciones de la reunión anterior. Entre estas sugerencias, se puede destacar:
 - El hecho de incorporar información sobre el tipo de TEA/diversidad del niño o niña puede no ser demasiado adecuado al representar una información de tipo personal que no es estrictamente necesario compartir con todos los profesionales/educadores que trabajan con el niño o niña.
 - Sería conveniente indicar con un mensaje que un cuento se ha creado correctamente tras finalizar el proceso de creación/edición.
 - Podría resultar interesante en ciertas situaciones habilitar una cooperación entre distintos educadores que trabajan con el niño o la niña, por ejemplo, si ambos están trabajando una misma área de desarrollo. De esta forma, un educador podría crear un cuento y compartirlo con otros profesionales que trabajen con la misma persona.
 - Sería interesante añadir la opción (personalizable) de incluir una melodía de música suave de fondo mientras el niño o la niña lee el cuento. Eso puede ayudar a relajar a ciertos niños.
 - En determinadas partes del cuento, se podría habilitar una especie de enlace para que el niño o niña pueda reforzar ciertos aprendizajes que se presentan en el cuento. Por ejemplo, discriminar ciertos objetos que aparecen en el cuento o para entender la emoción de una sección del mismo.
 - Por último, se ha definido en detalle cómo se realizarán las pruebas de la aplicación con un pequeño grupo de niños con TEA.

En la reunión inicial, ya se estableció que no resultaría demasiado útil que los niños probarán la aplicación en una única sesión y de forma puntual debido al alcance del proyecto y a las características de los niños con TEA. Este hecho se ha comprobado de forma práctica en la reunión cuando se le ha dejado interactuar a un niño con la aplicación. El objetivo de esta prueba era simplemente comprobar si la interfaz resultaba agradable e intuitiva, pero se ha comprobado que un niño afectado con TEA necesita

un cierto tiempo para familiarizarse con la aplicación, además de estar en un entorno controlado y tranquilo.

Teniendo en cuenta esto y siguiendo la idea inicial, se ha decidido enviar la aplicación a las familias de los niños que van a probarla para que, a lo largo de las próximas semanas, prueban y valoren el uso de la herramienta en un entorno familiar. Además, las profesionales se han comprometido a utilizar la aplicación también en sus intervenciones durante las dos próximas semanas con esos niños para valorar su utilidad en un entorno educativo y desde un punto de vista profesional.

Material adicional

En esta parte del trabajo, se explicarán los bocetos iniciales que se hicieron antes de comenzar con la implementación de la aplicación.

En las figuras D.1 y D.2, se puede ver representado el registro e inicio de sesión de nuestra aplicación. Primero se accederá a esta pantalla de inicio de sesión, pero no se podrá avanzar hasta que registre su usuario en el sistema. En el registro, el usuario se encontrará con un simple formulario donde introducirá varios datos como son sus credenciales, con los que posteriormente iniciará sesión, y su tipo de usuario, es decir, si es un tutor o un educador. Con el registro realizado, podrá iniciar sesión dentro de la aplicación.



Figura D.1: *Mockup* de la vista de Registro

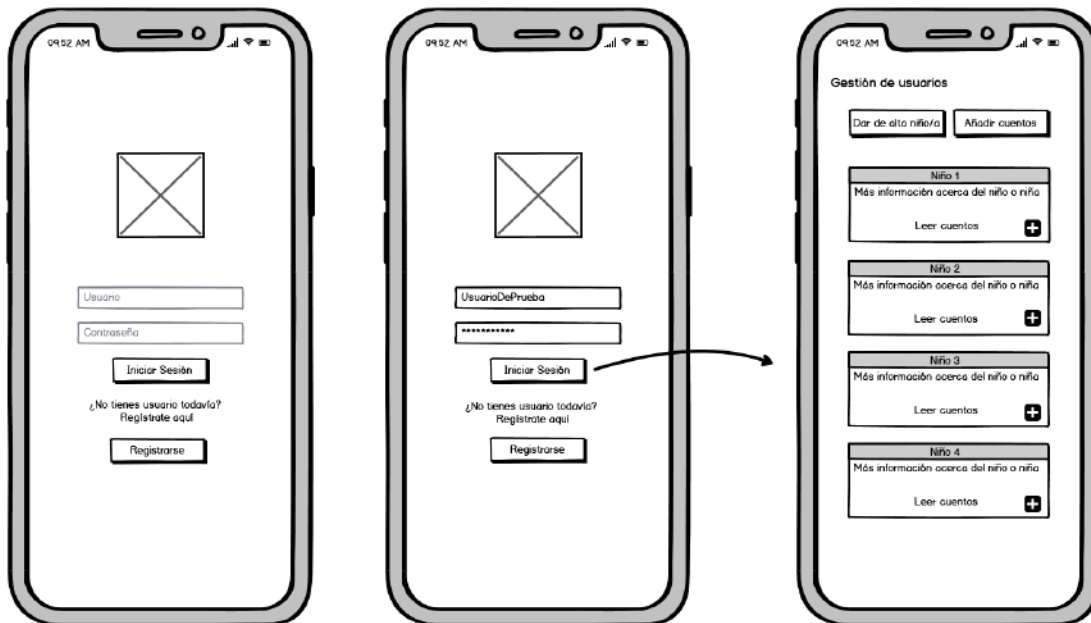


Figura D.2: *Mockup* de la vista de Inicio de Sesión

Si se inicia sesión por primera vez como un usuario **tutor**, que es el que representamos en este caso, se mostrará la primera pantalla de la figura D.3, donde se puede observar que indica que ese usuario no tiene niños/as asignados. Por lo tanto, este tutor podrá dar de alta un niño o niña en el sistema con el formulario que se representa en la imagen central de la figura D.3.

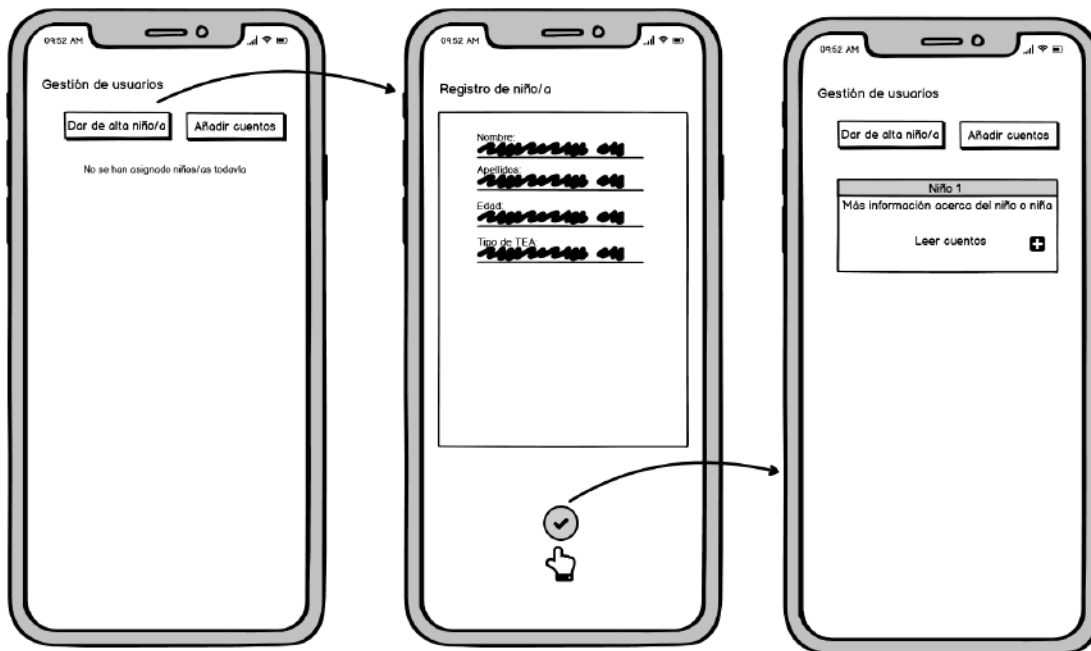


Figura D.3: *Mockup* de la vista de Añadir niño (*Tutor*)

Un vez dado de alta este niño, el usuario tutor podrá asignar este niño o niña a los demás tutores o educadores. Por ello, como se puede observar en la figura D.4, seleccionando el niño o la niña al que se le quiere añadir un tutor o educador, se accede a la pantalla donde se puede ver a todos los tutores o educadores del sistema y se podrá asignar uno o varios de ellos.

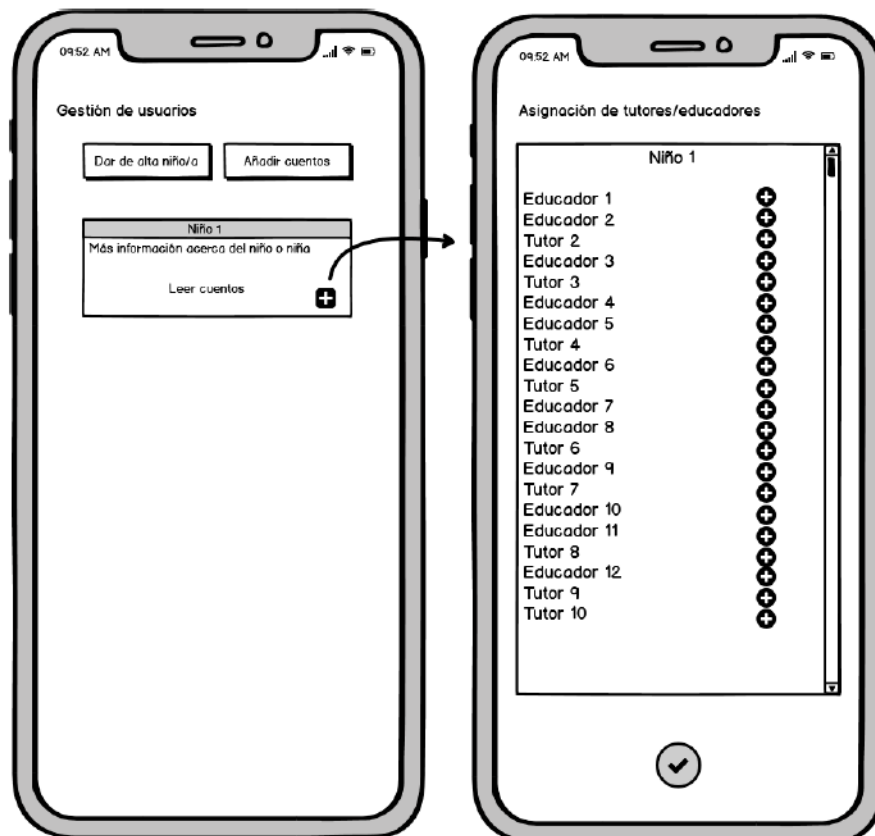


Figura D.4: *Mockup* de la vista de Asignar Responsable a un niño/a (*Tutor*)

La vista de un usuario **educador** será similar a la anterior, omitiendo estas dos últimas funcionalidades comentadas para el tutor, ya que este usuario no podrá hacer este tipo de gestión de usuarios, solo de la parte de cuentas. La figura D.5 muestra el boceto para la vista de la pantalla principal de un educador al que le han asignado varios niños o niñas y, como se puede observar, no se muestran los botones para dar de alta usuarios (niños/as) o para asignarlos.

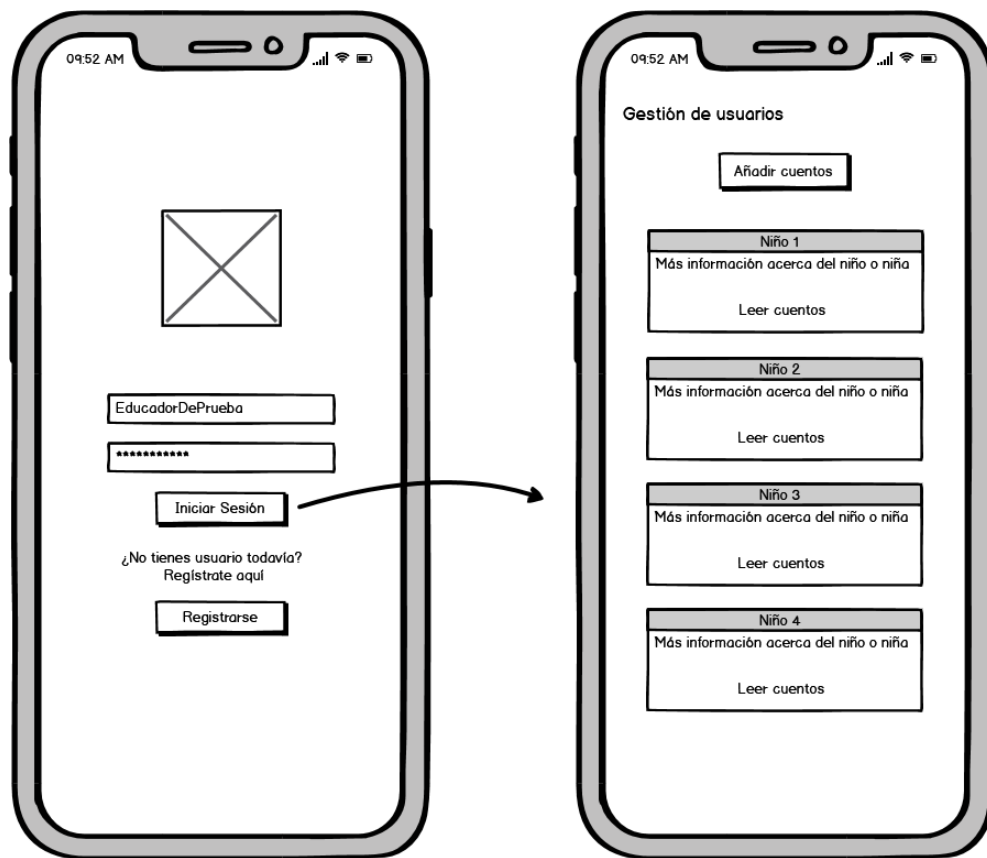


Figura D.5: *Mockup* de la vista de la pantalla principal de un usuario *Educador*

A continuación, se puede observar en la figura D.6 cómo sería la pantalla que va a ver el niño o la niña y con la que va a trabajar mediante los cuentos personalizados. En ella, se puede observar cómo se destacan las diferentes imágenes y pictogramas tanto a la hora de escoger cuento como cuando realizan la lectura. Dentro de cada historia, el niño o la niña podrá “interactuar” con el cuento seleccionando las palabras o los pictogramas para reproducirlos y podrá navegar entre sus páginas en los dos sentidos. El texto de los cuentos será con una tipografía recta y grande para que puedan leerlos de una forma sencilla y facilitar su comprensión. Actualmente, en el boceto se muestra con un tipo de letra predeterminado del programa pero la idea es que en la aplicación se muestre de la forma descrita.

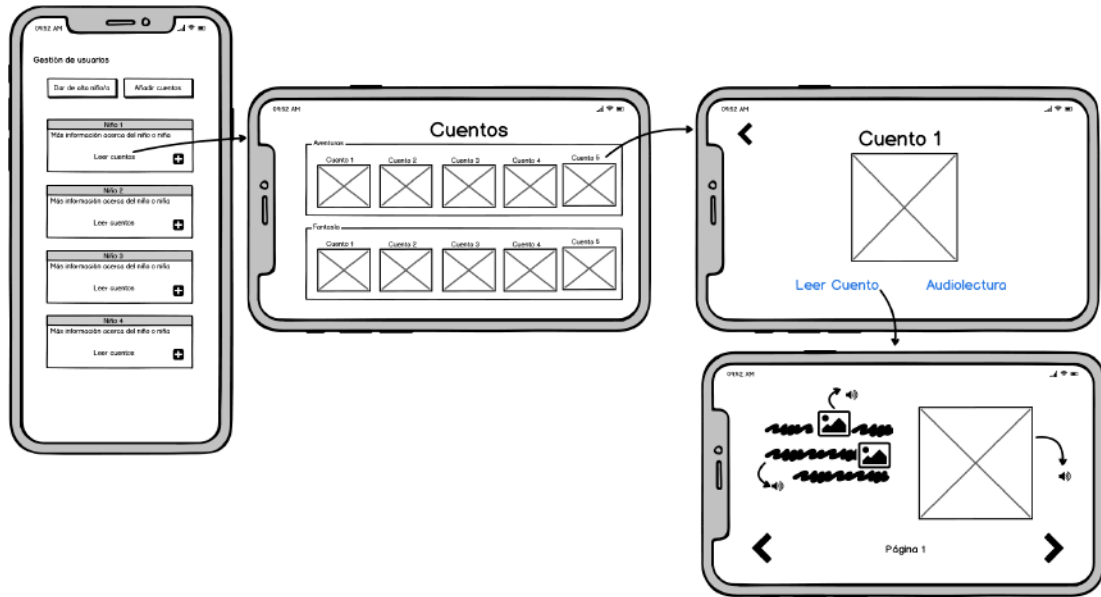


Figura D.6: *Mockup* de la vista de Lectura de cuentos

A la gestión de cuentos se puede acceder seleccionando el botón **Añadir cuentos** que se observa, por ejemplo, en la figura D.6. En esta vista, que se muestra en la figura D.7, se puede observar los diferentes pasos que conlleva la creación de un cuento. Aunque primero se había pensado realizarlo como en las pantalla de la figura D.8, tras la primera reunión se concluyó que era más óptimo y rápido hacerlo como se indica en la figura anterior. Esta conclusión fue tomada debido a que el proceso de creación era mucho más fluido y no requería hacerlo de manera manual en un dispositivo móvil, lo que puede llegar a ser muy tedioso, por lo que se decidió realizarlo mediante la importación de un fichero en formato *txt*.

Como se observa en la figura D.7, también se podrán seleccionar las palabras que serán sustituidas por pictogramas en el cuento y las imágenes que se asociarán a cada página. Como último paso de esta funcionalidad, se asociará el cuento a un niño o niña que tenga asignado ese usuario.

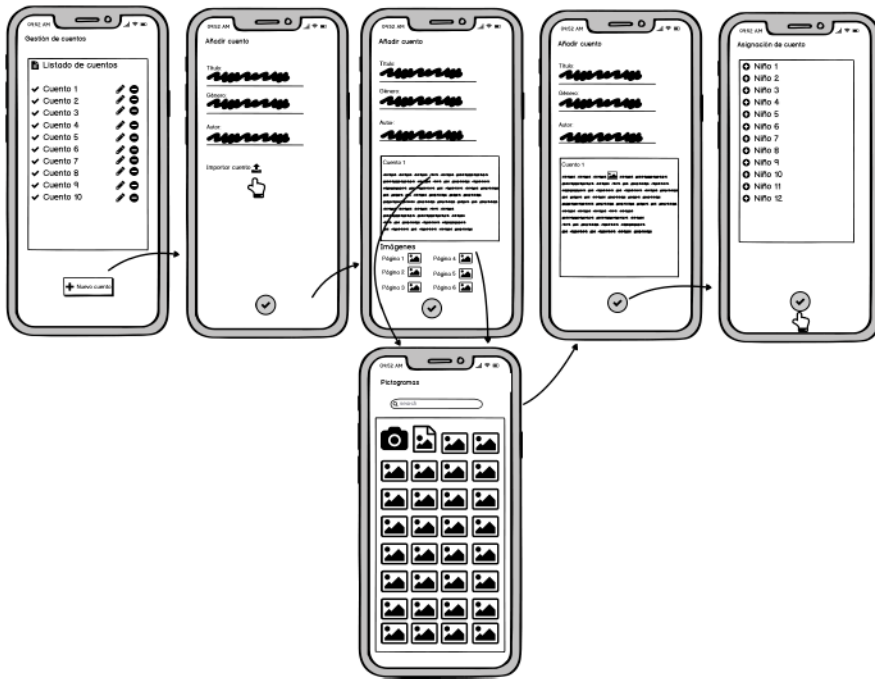


Figura D.7: *Mockup* de la vista de Creación de Cuentos. Segunda versión.

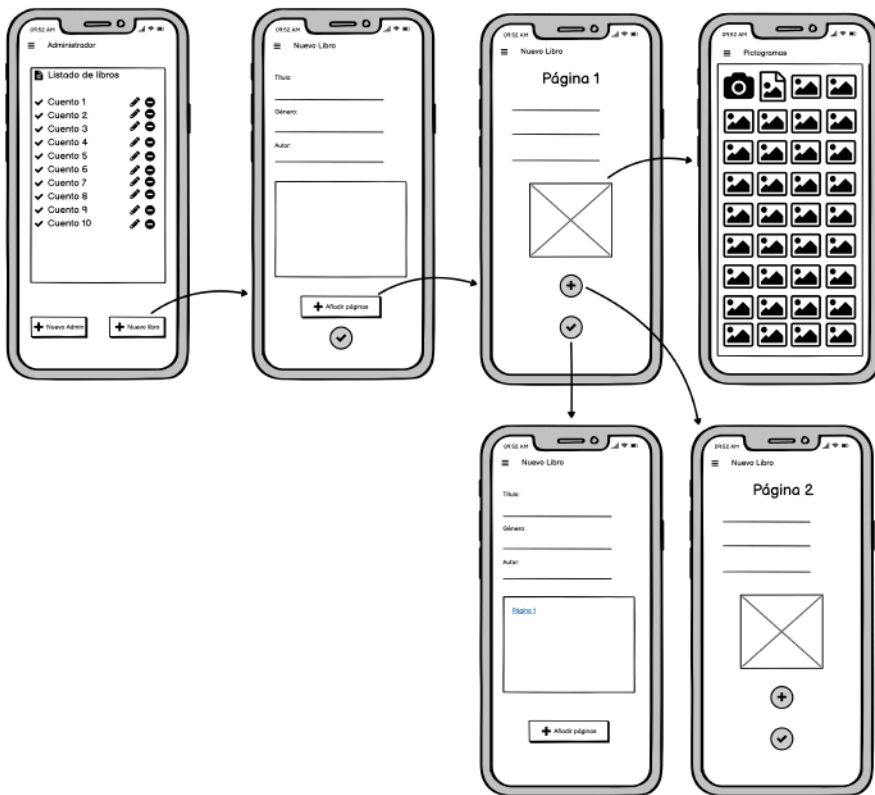


Figura D.8: *Mockup* de la vista de Creación de cuentos. Primera versión.

La edición de cuentas se puede ver representada en la figura D.9. Esta vista será muy similar a la de la creación. Las únicas diferencias que encontraremos son que los campos con lo datos iniciales del cuento ya estarán cubiertos y que habrá imágenes o pictogramas ya asociados a palabras. También se podrá asignar el cuento a otros niños o niñas.

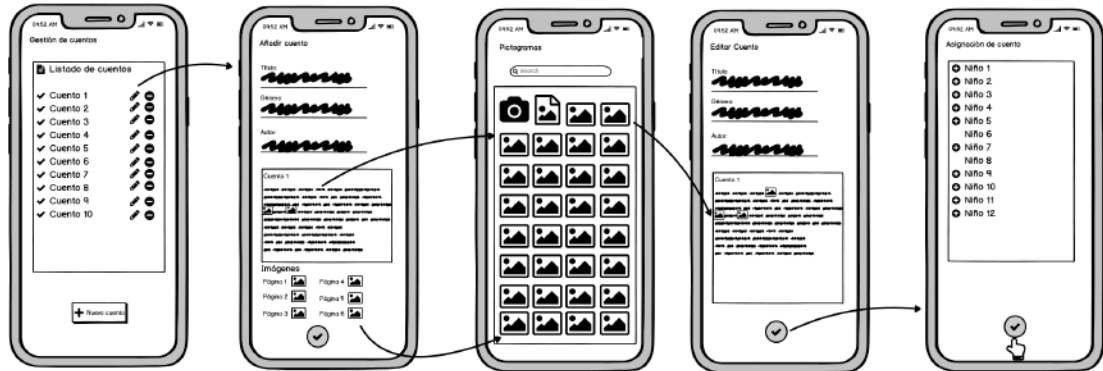


Figura D.9: Mockup de la vista de Edición de Cuentos

El borrado de los cuentos se realizará de una forma muy simple, como se representa en la figura D.10. Se seleccionará el botón de borrado del cuento que se quiere eliminar, lo que mostrará un mensaje para confirmar la acción, de forma que, si aceptamos, borrará el cuento y, si cancelamos, no lo realizará.

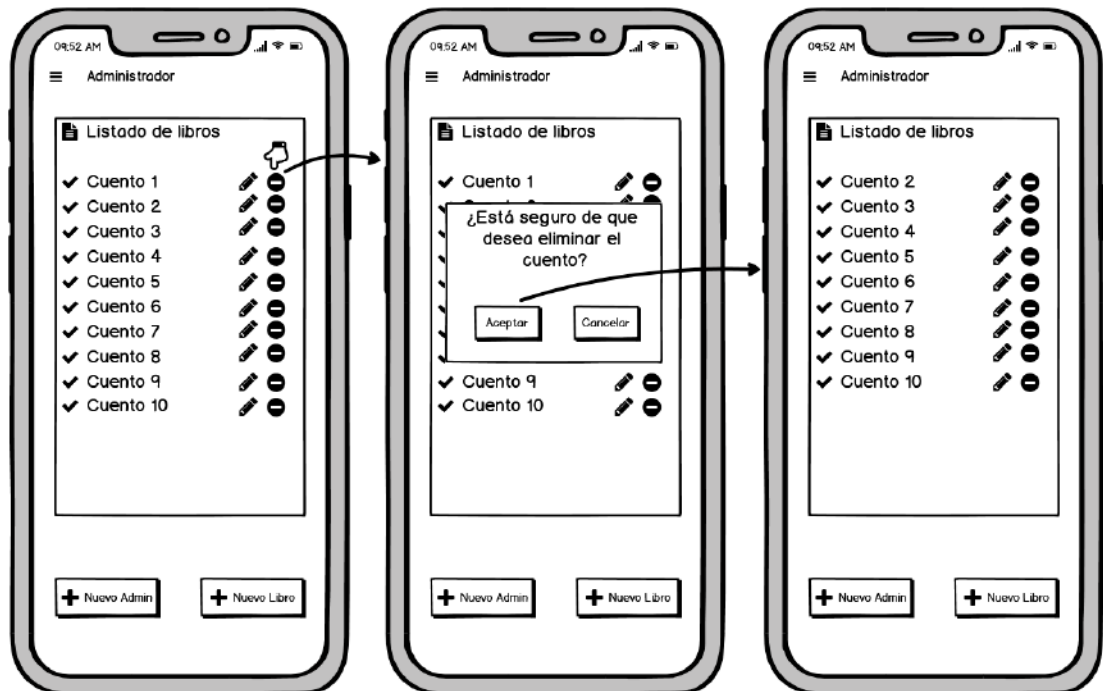


Figura D.10: Mockup de la vista de Borrado de Cuentos

Lista de acrónimos

- API** Application Programming Interface. iii, 43, 54–56, 62, 63, 68
- ApS** Aprendizaje-Servicio. 1, 2, 4, 25, 29, 30, 34, 37, 78
- ARASAAC** Centro Aragonés para la Comunicación Aumentativa y Alternativa. ii, iii, 14, 21, 43, 54–56, 62–64, 68
- DSM-5** Diagnostic and Statistical Manual of Mental Disorders, Fifth Edition. 5–7, 11
- HTTP** Hypertext Transfer Protocol. 21, 62, 63
- IDE** Integrated Development Environment. 22
- JSON** JavaScript Object Notation. 51, 63
- MD5** Message-Digest Algorithm 5. 61
- MVVM** Model-View-ViewModel. v, 49, 50, 56
- SAAC** Sistemas Alternativos y/o Aumentativos de Comunicación. 9
- TA** Trastornos de Aprendizaje. 6
- TDAH** Trastorno por Déficit de Atención e Hiperactividad. 6
- TEA** Trastorno del Espectro Autista. i, 1, 2, 4, 5, 7–16, 18, 21, 29–31, 34, 38–40, 50, 53, 54, 64, 66, 71, 72, 75–78, 83, 85, 87, 88
- TFG** Trabajo de Fin de Grado. 4, 16, 23, 29, 30
- TGD** Trastorno Generalizado del Desarrollo. 14
- TIC** Tecnologías de la Información y la Comunicación. 11

URL Uniform Resource Locator. 63

Bibliografía

- [1] G. de Navarra, “Definición/clasificación trastornos de la comunicación.” 2021. [En línea]. Disponible en: <https://creena.educacion.navarra.es/web/necesidades-educativas-especificas/dif-de-lenguaje-oral/definicion/>
- [2] C. de documentación de estudios y oposiciones, “DSM-5: Novedades y criterios diagnósticos,” 2021. [En línea]. Disponible en: <http://www.codajic.org/sites/www.codajic.org/files/DSM%205%20Novedades%20y%20Criterios%20Diagn%C3%B3sticos.pdf>
- [3] S. B. C. L.-S. y Sanguinetti, “Trastornos del aprendizaje,” 2017. [En línea]. Disponible en: https://www.pediatriaintegral.es/wp-content/uploads/2017/xxi01/03/n1-023-031_AnnaSans.pdf
- [4] M. Clinic, “El TDAH en niños,” 2021. [En línea]. Disponible en: <https://www.mayoclinic.org/es-es/diseases-conditions/adhd/symptoms-causes/syc-20350889#:~:text=El%20trastorno%20por%20d%C3%A9ficit%20de,atenci%C3%B3n%20hiperactividad%20y%20comportamiento%20impulsivo.>
- [5] C. A. España, “Sobre el TEA. TEA: Trastorno del espectro del autismo.” 2021. [En línea]. Disponible en: <http://www.autismo.org.es/sobre-los-TEA>
- [6] A. Madrid, “Un poco de historia sobre el TEA,” 2021. [En línea]. Disponible en: <https://autismomadrid.es/articulo-especializado/un-poco-de-historia-sobre-el-tea-parte-1/>
- [7] C. A. España, “Sobre el TEA. Detección y diagnóstico,” 2021. [En línea]. Disponible en: <http://www.autismo.org.es/sobre-los-TEA/deteccion>
- [8] —, “Sobre el TEA. Accesibilidad.” 2021. [En línea]. Disponible en: <http://www.autismo.org.es/sobre-los-TEA/accesibilidad>
- [9] B. y. M. S. Hervás Zuñiga, “Los trastornos del espectro autista (TEA).” 2017. [En línea]. Disponible en: <https://www.pediatriaintegral.es/publicacion-2017-03/los-trastornos-del-espectro-autista-tea/>

- [10] R. P. Seldas, “DSM-5: la nueva clasificación de los TEA.” Octubre 2014. [En línea]. Disponible en: <http://apacu.info/wp-content/uploads/2014/10/Nueva-clasificaci%C3%B3n-DSMV.pdf>
- [11] F. Orange, “Dictapicto,” 2021. [En línea]. Disponible en: <https://www.fundacionorange.es/aplicaciones/dictapicto-tea/>
- [12] R. M. Accesibles.org, “Pictotea, app de comunicacion para niños con autismo,” Marzo 2019. [En línea]. Disponible en: <https://www.accesibles.org/pictotea-app-de-comunicacion-para-ninos-con-autismo/>
- [13] Familiados, “Mejores apps para autismo en español,” Agosto 2019. [En línea]. Disponible en: <https://famiados.com/blog/mejores-apps-para-autismo-en-espanol/>
- [14] G. de Aragón, “¿Qué es ARASAAC?” 2021. [En línea]. Disponible en: <https://arasaac.org/about-us>
- [15] F. Orange, “Cuentos visuales: José aprende.” 2021. [En línea]. Disponible en: <https://www.fundacionorange.es/aplicaciones/cuentos-visuales-jose-aprende/>
- [16] —, “Teayudo a jugar.” 2021. [En línea]. Disponible en: <https://www.fundacionorange.es/aplicaciones/teayudo-a-jugar/>
- [17] F. Pictoaplicaciones, “Pictocuentos.” [En línea]. Disponible en: <https://www.pictocuentos.com/>
- [18] Flutter, “Flutter,” 2021. [En línea]. Disponible en: <https://flutter.dev/>
- [19] Dart, “Dart overview,” 2021. [En línea]. Disponible en: <https://dart.dev/overview>
- [20] inLAB FIB, “¿Qué es el lenguaje de programación Dart?” 2021. [En línea]. Disponible en: <https://inlab.fib.upc.edu/es/blog/que-es-el-lenguaje-de-programacion-dart>
- [21] Firebase, “Firebase,” 2021. [En línea]. Disponible en: <https://firebase.google.com/docs?hl=es-419>
- [22] —, “Firebase Authentication,” 2021. [En línea]. Disponible en: <https://firebase.google.com/docs/auth>
- [23] —, “Cloud Firestore,” 2021. [En línea]. Disponible en: <https://firebase.google.com/docs/firestore?hl=es>
- [24] —, “Cloud storage para Firebase,” 2021. [En línea]. Disponible en: <https://firebase.google.com/docs/storage>

- [25] —, “Cloud Functions para Firebase,” 2021. [En línea]. Disponible en: <https://firebase.google.com/docs/functions>
- [26] Wikipedia, “Android,” 2021. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Android>
- [27] G. Developers, “Introducción a Android Studio,” 2021. [En línea]. Disponible en: <https://developer.android.com/studio/intro?hl=es-419>
- [28] Wikipedia, “iOS,” 2021. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/IOS>
- [29] —, “Xcode,” 2021. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Xcode>
- [30] Git, “Git, everithing is local,” 2021. [En línea]. Disponible en: <https://git-scm.com/>
- [31] Atlassian, “Sourcetree,” 2021. [En línea]. Disponible en: <https://www.sourcetreeapp.com/>
- [32] M. 365, “Administración de proyectos,” 2021. [En línea]. Disponible en: <https://www.microsoft.com/es-es/microsoft-365/project/project-management>
- [33] Overleaf, “What is Latex?” 2021. [En línea]. Disponible en: https://es.overleaf.com/learn/latex/Learn_LaTeX_in_30_minutes#What_is_LaTeX.3F
- [34] —, “Overleaf,” 2021. [En línea]. Disponible en: <https://es.overleaf.com>
- [35] B. Studios, “Blasamiq Wireframes,” 2021. [En línea]. Disponible en: <https://balsamiq.com/>
- [36] DrawIO, “Página web oficial,” 2021. [En línea]. Disponible en: <https://app.diagrams.net/>
- [37] Atlassian, “¿Qué es Scrum?” 2021. [En línea]. Disponible en: <https://www.atlassian.com/es/agile/scrum>
- [38] K. S. y Jeff Sutherland, “La guía de Scrum,” Julio 2016. [En línea]. Disponible en: <https://scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-Spanish.pdf>
- [39] C. Harris, “Artefactos del Scrum ágil.” [En línea]. Disponible en: <https://www.atlassian.com/es/agile/scrum/artifacts>
- [40] A. Hebert and P. Hauf, *Student learning through service learning: Effects on academic development, civic responsibility, interpersonal skills and practical skills. Active Learning in Higher Education*, 2015.
- [41] Indeed, “¿Cuánto se gana en españa de Scrum Master/a?” [En línea]. Disponible en: <https://es.indeed.com/career/scrum-master/salaries>

- [42] —, “¿Cuánto se gana en España de analista programador/a?” [En línea]. Disponible en: https://es.indeed.com/career/analista-programador/salaries?from=top_sb
- [43] —, “¿Cuánto se gana en España de analista diseñador?” [En línea]. Disponible en: https://es.indeed.com/career/dise%C3%B1ador-web/salaries?from=top_sb
- [44] L. Reyes, “Aplicando el patrón de diseño MVVM,” 2018. [En línea]. Disponible en: <https://medium.com/@reyes.leomaris/aplicando-el-patr%C3%B3n-de-dise%C3%B1o-mvvm-d4156e51bbe5>