



TRABALLO FIN DE GRAO
GRAO EN ENXEÑARÍA INFORMÁTICA
MENCIÓN EN TECNOLOXÍAS DA INFORMACIÓN



Ferramenta de apoio á comunicación aumentativa baseada no método PECS

Estudante: Nuria Agra Rial

Dirección: Óscar Fresnedo Arias

Paula María Castro Castro

A Coruña, xuño de 2023.

A miña familia e amigos por apoiarme todos estos anos.

Agradecementos

Comezar agradecendo aos meus titores, que propuxeron este proxecto e me axudaron durante o seu desenvolvemento. Quero mostrar tamén o meu agradecemento a miña familia, por apoiarme desde sempre e por haberme permitido chegar ata aquí, e aos meus amigos e parella polo seu apoio e por ofrecerme un descanso cando máis o necesitaba.

Resumo

O obxectivo deste traballo é o desenvolvemento dunha aplicación móbil apta para Android e iOS que use pictogramas e permita gardalos en categorías e crear frases con eles para facilitarlle a comunicación aos nenos con trastorno do espectro autista (TEA).

Abstract

The project's goal is to develop a mobile application available for Android and iOS that uses pictograms and allows users to sort them into categories and create sentences to make communication easier for children with autism spectrum disorder (ASD).

Palabras chave:

- Trastorno do espectro autista
- Pictogramas
- Categorías
- Sistemas Aumentativos e Alternativos de Comunicación
- Aplicación móbil
- Flutter
- Firebase

Keywords:

- Autism spectrum disorder
- Pictograms
- Categories
- Augmentative and Alternative Communication Systems
- Mobile application
- Flutter
- Firebase

Índice Xeral

1	Introdución	1
1.1	Trastorno do espectro autista	1
1.1.1	A comunicación nas persoas con TEA	1
1.1.2	Variedades do TEA	2
1.1.3	Sintomaloxía	3
1.1.4	Diagnóstico	4
1.1.5	Tratamento	4
1.2	Motivación	5
1.3	Obxectivos do proxecto	6
1.4	Estrutura da memoria	7
2	Estado do arte	8
2.1	Leeloo AAC	8
2.2	Soyvisual	8
2.3	Cboard	9
2.4	DictaPicto	10
2.5	Táboa comparativa	10
3	Fundamentos tecnolóxicos	13
3.1	Flutter	13
3.2	ARASAAC	13
3.3	Firebase	14
3.3.1	Firebase Authentication	14
3.3.2	Firestore Database	14
3.3.3	Firebase Storage	14
3.4	Dart	15
3.5	Android Studio	15
3.6	Git	15

3.6.1	Github	16
3.7	Balsamiq Wireframes	16
3.8	draw.io (Diagrams.net)	16
3.9	Wrike	17
3.10	Latex	17
3.11	Overleaf	18
4	Metodoloxía	19
4.1	Scrum	19
4.1.1	Sprints	20
4.1.2	Membros dun equipo Scrum	21
4.1.3	Artefactos de Scrum	21
4.2	Aprendizaxe e Servizo	22
5	Organización do proxecto	23
5.1	Tarefas	23
5.1.1	Sprint 1	23
5.1.2	Sprint 2	24
5.1.3	Sprint 3	24
5.1.4	Sprint 4	24
5.1.5	Sprint 5	25
5.1.6	Sprint 6	25
5.1.7	Sprint 7	25
5.2	Diagrama dos sprints	25
5.3	Seguemento	26
5.4	Recursos	27
5.4.1	Recursos humanos	27
5.4.2	Recursos materiais	27
5.5	Custos	28
6	Análise	30
6.1	Actores	30
6.1.1	Usuario non autenticado	30
6.1.2	Responsable	30
6.1.3	Profesional	30
6.1.4	Usuario TEA	31
6.2	Requisitos	31
6.2.1	Requisitos funcionais	31

6.2.2	Requisitos non funcionais	32
6.3	Casos de uso	33
6.3.1	Usuario non autenticado	33
6.3.2	Responsable	33
6.3.3	Profesional	35
6.3.4	Usuario TEA	36
7	Deseño	38
7.1	Patrón de deseño	38
7.2	Deseño da base de datos	39
7.2.1	Firestore Database	39
7.2.2	Firestore Storage	42
7.3	Deseño das interface do usuario	43
7.3.1	Deseño inicial	43
8	Implementación	51
8.1	Estrutura	51
8.2	Backend	51
8.2.1	Firestore Authentication	51
8.2.2	Firestore Database	52
8.2.3	Firestore Storage	52
8.2.4	ARASAAC	53
8.2.5	Uso da galería ou cámara	55
8.2.6	Reprodución dos pictogramas	55
8.3	Frontend	56
8.4	Vistas finais	56
8.4.1	Autenticación	56
8.4.2	Listado de usuarios	57
8.4.3	Listado de categorías e pictogramas	58
8.4.4	Configuración do usuario	58
8.4.5	Drawer	58
8.4.6	Perfil dos usuarios TEA	60
8.4.7	Engadir usuarios, pictogramas e categorías	60
8.4.8	Vistas dos usuarios TEA	61
8.4.9	Vistas do profesional	63
8.5	Probos	63
8.5.1	Probos unitarias	63
8.5.2	Probos de integración	64

8.5.3	Probas de aceptación	64
9	Conclusións	65
9.1	Conclusións	65
9.2	Liñas futuras	66
A	Vistas tablet	68
	Relación de Acrónimos	73
	Bibliografía	74

Índice de Figuras

2.1	Mostra do uso da aplicación Leeloo AAC.	9
2.2	Mostra do uso da páxina web de Soyvisual.	9
2.3	Mostra do uso da aplicación Cboard.	10
3.1	“Pizarra” ofrecida pola ferramenta Wrike.	17
5.1	Planificación desde o sprint 1 ao 4.	26
5.2	Planificación desde o sprint 5 ao 7.	26
6.1	Diagrama dos casos de uso do usuario non autenticado.	34
6.2	Diagrama de casos de uso do responsable.	35
6.3	Diagrama dos casos de uso do profesional.	36
6.4	Diagrama de casos de uso para o usuario TEA.	37
7.1	Fluxo unidireccional do patrón BLoC.	39
7.2	Arquitectura do patrón BLoC.	40
7.3	Pantallas do móbil.	44
7.4	Pantallas da tablet usando o Master Detail.	44
7.5	Pantallas de autenticación da aplicación.	45
7.6	Listados de usuarios.	46
7.7	Listado de usuarios TEA na <i>tablet</i>	47
7.8	Listado de pictogramas e categorías.	48
7.9	Configuración do usuario.	48
7.10	Configuración do usuario na <i>tablet</i>	49
7.11	Editar e eliminar o perfil do neno.	49
7.12	Pantallas de creación dos elementos da aplicación.	50
7.13	Vista da creación de frases dos usuarios TEA.	50
8.1	API para a obtención dos pictogramas.	54

8.2	API para a obtención dos identificadores dos pictogramas.	54
8.3	Pantallas de autenticación finais.	57
8.4	Versión final do listado de nenos.	58
8.5	Listados de categorías e pictogramas.	59
8.6	Configuración do usuario.	59
8.7	Drawer da aplicación.	60
8.8	Vista final do perfil do neno.	61
8.9	Pantallas finais para a creación de elementos.	62
8.10	Vista final do perfil do neno.	62
8.11	Vistas desde un usuario coa categoría de profesional.	63
A.1	Vista das pantallas dos nenos na <i>tablet</i>	68
A.2	Listado de usuarios TEA na <i>tablet</i>	69
A.3	Engadir categoría na <i>tablet</i>	69
A.4	Engadir pictograma <i>tablet</i>	70
A.5	Vista das categorías na <i>tablet</i>	70
A.6	Vista dos pictogramas na <i>tablet</i>	71
A.7	Pantalla de rexistro na <i>tablet</i>	71
A.8	Pantalla de inicio de sesión na <i>tablet</i>	72
A.9	Configuración do usuario na <i>tablet</i>	72

Índice de Táboas

2.1	Táboa comparativa considerando as funcionalidades de cada aplicación.	11
2.2	Táboa comparativa de prezos e sistemas.	12
5.1	Táboa de seguemento do proxecto.	27
5.2	Táboa de salarios.	28
5.3	Táboa de custos en recursos humanos.	29
5.4	Táboa de custos en recursos materiais.	29
5.5	Táboa de custos totais.	29
7.1	Táboa coa organización da colección “admins”.	41
7.2	Táboa coa organización da subcolección “teaUsers”.	41
7.3	Táboa coa organización da subcolección “categories”.	42
7.4	Táboa coa organización da subcolección “pictos”.	42

Introdución

Neste primeiro capítulo da memoria, presentarase unha ferramenta que permita facilitar a comunicación dos nenos e nenas con **Trastorno Espectro Autista (TEA)**, expoñendo a motivación detrás do proxecto, os seus obxectivos e a estrutura completa da memoria.

1.1 Trastorno do espectro autista

O **TEA** é un conxunto de trastornos neurolóxicos e de desenvolvemento que afectan as formas de interactuar, comunicarse, aprender e comportarse das persoas [1]. Chámase trastorno de espectro porque distintas persoas con **TEA** poden ter unha gran variedade de síntomas. Dentro deste espectro, tamén se inclúe o que se coñece como síndrome de Asperger e o trastorno xeralizado do desenvolvemento non especificado [2].

As causas deste trastorno son actualmente descoñecidas, aínda que hai estudos que indican que hai máis de 100 posibles xenes que poden estar implicados e que os factores medioambientais afectan á evolución e desenvolvemento desta condición. Nestes últimos anos, a porcentaxe de persoas diagnosticadas con **TEA** aumentou significativamente debido ao maior coñecemento que se ten e interese da comunidade médica e científica. Só en España, hai máis de 45.0000 persoas con autismo e estímase que nacen 4.500 bebés con **TEA** ao ano [3].

1.1.1 A comunicación nas persoas con TEA

Este trastorno acompaña á persoa ao longo da súa vida e afecta principalmente a comunicación e interacción social, así como á flexibilidade de comportamento e pensamento. Neste sentido, fai que estas persoas poidan ter problemas para comprender ou empregar a linguaxe oral, que teñan uns intereses moi restrinxidos e específicos, ou que precisen de rutinas moi estruturadas. Todos estes aspectos fan que sexa preciso traballar de forma específica este tipo de habilidades con eles e, para iso, habitualmente faise uso dos **Sistemas Aumentativos e Alternativos de Comunicación (SAAC)**, que son formas de expresión diferentes da linguaxe

falada que teñen como obxectivo aumentar o nivel de expresión e compensar as dificultades de comunicación. Estes sistemas baséanse no uso de diferentes tipos de apoios que axudan a mellorar a comprensión nas interaccións sociais e nos procesos comunicativos. Os SAAC poden usar como apoios distintos tipos de símbolos, incluídos os gráficos e os xestuais, como serían os pictogramas, que son debuxos que representan o obxecto real ao que se refiren.

Este proxecto céntrase no sistema de uso de pictogramas denominado **Picture Exchange Communication System (PECS)**, que consta de diferentes fases nas que o neno vai avanzando sempre apoiándose no uso de pictogramas para comunicar aos seus familiares/coiadores o que quere ou precisa en cada momento [4]. Nos niveis máis avanzados deste sistema, é posible cubrir diferentes funcións comunicativas, incluíndo as peticións ou proporcionar información relevante.

O obxectivo principal de PECS é ensinar comunicación funcional e para iso consta de seis fases [5]:

1. Fase 1: Aprenden a intercambiar/seleccionar imaxes (pictogramas) para conseguir certas cousas que desexan ou acceder a certas actividades.
2. Fase 2: Seguen usando as imaxes pero agora en distintos entornos e con outras persoas (xeralización da aprendizaxe).
3. Fase 3: Empézase coa discriminación entre varias imaxes para elixir as súas favoritas.
4. Fase 4: Aprenden a construír oracións simples facendo uso dun pictograma que indique “Quero”, seguido do que solicitan. Durante esta fase, tamén aprenden a expandir as oracións engadindo adxectivos, verbos e preposicións.
5. Fase 5: Aprenden a usar PECS para responder ás preguntas.
6. Fase 6: Aprenden a compoñer oracións máis complexas e seguen respondendo a preguntas.

1.1.2 Variedades do TEA

A pesar de que resulta moi complexo realizar unha clasificación dos diferentes tipos de TEA debido a gran variedade de manifestacións que pode presentar nas persoas afectas, pódense diferenciar 5 grandes variedades dentro deste espectro [6]:

1. Autismo: Este trastorno comeza nos primeiros tres anos de vida e algúns dos síntomas máis significativos son unha nula ou moi escasa comunicación verbal, que o neno sexa pouco sociable ou que non mostre interese en chamar a atención dos pais.

2. Síndrome de Rett: A principal característica é que se presenta principalmente en nenas. Ao redor dos dous anos, as persoas afectadas empezan a sufrir un proceso dexenerativo do sistema nervioso, manifestado en alteracións na comunicación, a cognición e a motricidade.
3. Síndrome de Asperger: As persoas afectadas por esta variedade non presentan ningún tipo de alteración intelectual nin física que as identifique, dificultando notablemente o seu diagnóstico. O problema neste caso está no campo das capacidades sociais, podendo chegar a comprometer a súa integración social e laboral. As características máis habituais do Asperger son a falta de empatía, unha pouca coordinación psicomotriz, non entender as ironías e a obsesión con certos temas.
4. Trastorno desintegrado infantil ou síndrome de Heller: Esta variedade pode non ser evidente ata pasados os dez anos. A pesar de afectar ás mesmas áreas que outros tipos, ten un carácter regresivo e repentino que pode causar que ata o propio neno se dea conta do problema.
5. Trastorno xeralizado do desenvolvemento non especificado: Úsase para os casos onde os síntomas son demasiado heteroxéneos como para ser incluído nos outros tipos.

1.1.3 Sintomaloxía

Con frecuencia, obsérvanse algúns tipos de comportamentos que son bastante recurrentes en persoas diagnosticadas con este tipo de trastornos. Aínda así, non todo o mundo vai ter os seguintes síntomas, pero é bastante probable que teñan varios deles [1]:

- Facer pouco contacto visual ou facelo de forma inconsciente.
- Parecer que non mira ou escoita ás persoas que están falando.
- Compartir con pouca frecuencia intereses, emocións ou actividades que disfrutan.
- Non responder ou tardar cando os chaman.
- Dificultade para seguir conversacións.
- Falar moi seguidamente dun tema sen permitir responder á outra persoa.
- Expresións faciais, movementos e xestos que non coinciden co que están dicindo.
- Ton de voz que pode ser similar a cantar ou moi monótono.
- Dificultades para comprender o punto de vista doutra persoa.

- Problemas para adaptar o seu comportamento a diferentes situacións sociais.
- Dificultades para compartir nun xogo imaxinativo ou para facer amigos.

Con respecto aos comportamentos repetitivos, poden incluírse:

- Repetir palabras ou frases.
- Mostrar especial interés en temas específicos como números ou datos.
- Mostrar demasiado interés en cousas específicas como obxectos en movemento.
- Molestarse por cambios pequenos nunha rutina.
- Ser máis sensibles á luz, roupa, son ou temperatura.

Ademais dos síntomas previamente mencionados, os nenos con **TEA** tamén poden ter dificultades para durmir. Como se pode observar, unha gran cantidade destes síntomas afectan ás áreas relacionadas coa comunicación e interacción social.

1.1.4 Diagnóstico

O diagnóstico en nenos pequenos divídese en dúas etapas [1]. A etapa inicial é a avaliación xeral durante as revisións periódicas do médico dos nenos. Se durante esta avaliación detéctanse problemas no desenvolvemento funcional ou da conduta, pasarase á segunda etapa. Esta segunda avaliación pode incluír exames neurolóxicos, unha avaliación sobre as habilidades de fala dos nenos e das súas capacidades cognitivas, observación do seu comportamento, unha avaliación das habilidades apropiadas da súa idade e preguntas sobre a historia familiar.

No caso de nenos pequenos e adolescentes, o equipo de educación especial da súa escola ou os familiares, que son os que máis tempo pasarán con el, poderían facer unha avaliación inicial e, no caso de detectar indicios deste trastorno, xa se trataría cun especialista.

Con respecto ao **TEA** en adultos, isto é máis difícil de detectar xa que os síntomas poden coincidir cos doutros trastornos de saúde mental. Neste caso, pódese mandar a un psicólogo que preguntará sobre os retos nas relacións sociais ou comportamentos repetitivos, e este tamén poderá falar cos familiares e persoas que coidaron ao adulto cando era pequeno para poder obter un diagnóstico.

1.1.5 Tratamento

O tratamento dos **TEA** debe empezar tan pronto como se faga o diagnóstico para diminuír as dificultades destas persoas e aproveitar ao máximo as súas fortalezas. Non hai unha

tratamento xeral para as persoas con TEA, hai que atopar para cada persoa a combinación de tratamentos que mellor lle vaia. O TEA afecta a cada persoa de forma distinta, polo que non hai unha vía única de tratamento que seguir, senón que depende da persoa. A maioría de tratamentos pódense dividir segundo as seguintes categorías [7]:

- Enfoques condutais: Son os que contan coa maior cantidade de evidencia para o tratamento dos síntomas do TEA. Céntranse en cambiar os comportamentos ao entender o que pasa antes e despois destes.
- Enfoques de desenvolvemento: Estes tratamentos céntranse en mellorar as destrezas que teñen que ver co desenvolvemento. A terapia máis común é a terapia da fala e da linguaxe, que axuda a mellorar a comprensión e uso da lingua e fala dunha persoa. A terapia ocupacional ensina destrezas que lle axudan á persoa a vivir da forma máis independente posible. Adoitan combinarse cos enfoques condutais.
- Educacionais: Estes tratamentos realízanse nas clases. Ofréceselle aos maestros formas de adaptar a estrutura da aula e mellorar os resultados académicos e outros resultados.
- Socio-relacionais: Céntranse en mellorar as destrezas sociais e crear vínculos emocionais.
- Farmacolóxicos: Non hai medicamentos que poidan reducir os síntomas do TEA, pero si poden usarse para reducir outros síntomas como son a irritabilidade, os comportamentos disruptivos, a hiperactividade ou a ansiedade.
- Psicolóxicos: A terapia cognitiva-condutal é un enfoque psicolóxico centrado en aprender as conexións entre os pensamentos, os sentimentos e os comportamentos.
- Complementarios e alternativos: Tratamentos que non encaixan dentro das outras categorías.

En calquera caso, hai varios aspectos importantes á hora de implementar calquera tratamento orientado á mellorar as condicións das persoas con TEA: debe comezarse o antes posible, as intervencións teñen que abarcar diferentes áreas co obxectivo de mellorar a calidade de vida e autonomía destas persoas, e precísase dun mecanismo efectivo de comunicación para conseguir o resultado desexado.

1.2 Motivación

Traballar co sistema PECS pode presentar unha serie de incomodidades e limitacións, relacionadas con ter que crear fisicamente os pictogramas, crear as súas carpetas correspondentes

e organizalos. Para cada pictograma que se queira empregar, é necesario imprimir a imaxe, plastificalo, engadirlle o velcro..., e ademais a súa manipulación pode facer que se degrade rapidamente. Por iso, resultaría moi vantaxoso o feito de poder dispoñer dunha ferramenta tecnolóxica que permita automatizar todo este proceso, de forma que os responsables/profesionais poidan crear e organizar de forma sinxela e cómoda todo o material necesario para comunicarse/traballar cos nenos con TEA usando o sistema PECS.

Do mesmo xeito, estes nenos soen sentirse moi atraídos polos estímulos visuais das aplicacións para móbiles e tabletas, o que lles axuda a centrar a súa atención. Por esa razón, moitas das intervencións e interaccións con estes nenos adoitan canalizarse a través do uso destes dispositivos.

Persoalmente, resulta moi interesante a creación dunha aplicación deste tipo desde cero, usando tecnoloxías modernas e tendo a opción de aplicar as metodoloxías que se aplicarían no día a día nun entorno laboral. Ademais, resulta moi motivador que este traballo poida servir para mellorar a calidade de vida dun grupo de persoas, permitindo aplicar o método de traballo *Aprendizaxe e Servicio (ApS)* para poder usar os coñecementos adquiridos durante a formación nunha práctica solidaria.

1.3 Obxectivos do proxecto

O principal obxectivo deste proxecto é a creación dunha aplicación para móbiles que permita aos responsables e profesionais crear facilmente as carpetas PECS, os pictogramas e organizalo todo segundo as necesidades do neno. Ademais, tamén se busca que os nenos poidan utilizar a aplicación para comunicarse e desenvolver novas habilidades comunicativas poñendo en práctica as fases máis temperás do sistema PECS, permitindo seleccionar e discriminar pictogramas, como as fases máis avanzadas que permitan compoñer frases a partir da selección de varios pictogramas de xeito secuencial para solicitar algo ou responder a preguntas doutras persoas. Co desenvolvemento desta aplicación, búscase que os nenos dispoñan dunha ferramenta útil que poidan usar no seu día a día para comunicarse, pero que tamén poida ser empregada nas diferentes intervencións que se leven a cabo por parte dos profesionais e no propio ámbito educativo.

Así mesmo, preténdese adquirir uns coñecementos tanto no desenvolvemento de aplicacións móbiles como no uso dunha metodoloxía áxil, como é Scrum. Finalmente, este proxecto tamén está orientado a coñecer a realidade social das persoas con TEA e das súas familias, co obxectivo de contribuír a mellorar a súa calidade de vida.

1.4 Estrutura da memoria

A memoria deste proxecto está estruturada da seguinte forma:

1. **Introdución:** Capítulo actual onde se fala do **TEA**, dos sistemas **PECS**, da motivación e obxectivo do proxecto, e se presenta a estrutura da memoria.
2. **Estado do arte:** Capítulo onde se mencionan algunhas das aplicacións e ferramentas xa existentes no mercado con funcións similares as que se buscan desenvolver aquí.
3. **Fundamentos tecnolóxicos:** Listado e explicacións das ferramentas de software e tecnoloxías usadas neste proxecto para o desenvolvemento da aplicación móbil.
4. **Metodoloxía:** Capítulo onde se explica a metodoloxía elexida para o desenvolvemento.
5. **Organización do proxecto:** Explicación da organización que se estableceu para o proxecto, o diagrama temporal coa organización das tarefas nos diferenets sprints e a correspondente estimación de custos.
6. **Análise:** Capítulo onde se especifican os requisitos que debe cumprir a aplicación, os actores que a usarán e os seus casos de uso.
7. **Deseño:** Capítulo onde se expoñen os patróns de deseño seguidos, o deseño da base de datos e das vistas da aplicación.
8. **Implementación:** Capítulo onde se amosarán as partes máis relevantes do código e as probas realizadas na aplicación.
9. **Conclusións:** Reflexión final e valoración do estado final do proxecto, e das ferramentas e coñecementos adquiridos.

Estado do arte

Agora mesmo, debido aos avances nas tecnoloxías, dispónse dunha gran variedade de aplicacións creadas para nenos con TEA, que poden servir para axudalos a comunicarse usando o sistema PECS. Neste capítulo, farase unha breve revisión das principais ferramentas existentes orientadas a mellorar a comprensión e comunicación de nenos con TEA empregando SAACs.

2.1 Leeloo AAC

Esta aplicación está baseada en pictogramas e xa conta con categorías preestablecidas, polo que o propio profesional non podería engadilas en función do neno. Ademais de que, para poder obter máis carpetas das iniciais, deberían pagar a versión premium. Un punto positivo desta aplicación é que conta xa con frases preestablecidas, ademais de dispor da reprodución das palabras e frases podendo variar as voces de reprodución [8].

Na imaxe 2.1 (páxina 9), obsérvase unha das categorías preestablecidas ofrecidas por esta aplicación e os pictogramas dentro desta, que ademais, son específicos da aplicación.

2.2 Soyvisual

Esta ferramenta está dispoñible tanto para iOS como para Android, ademais de contar cunha versión web. Dispón de 3 seccións principais, incluíndo unha parte con pictogramas que representan distintos obxectos, unha de láminas con situacións cotiás e outra con distintos tipos de exercicios. Entre estes exercicios, podemos atopar algúns de asociación de elementos ou de construír frases [9]. Podemos ver como se verían os pictogramas desta aplicación na imaxe 2.2 (páxina 9).

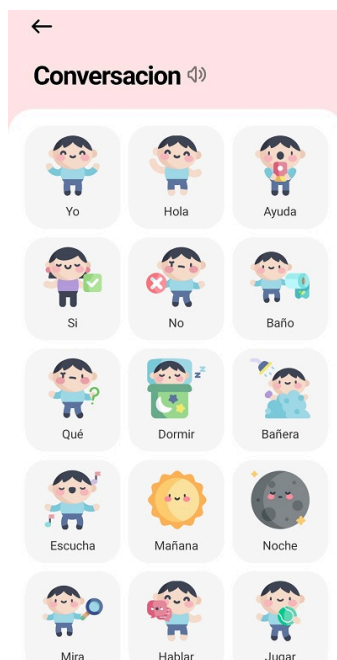


Figura 2.1: Mostra do uso da aplicación Leeloo AAC.

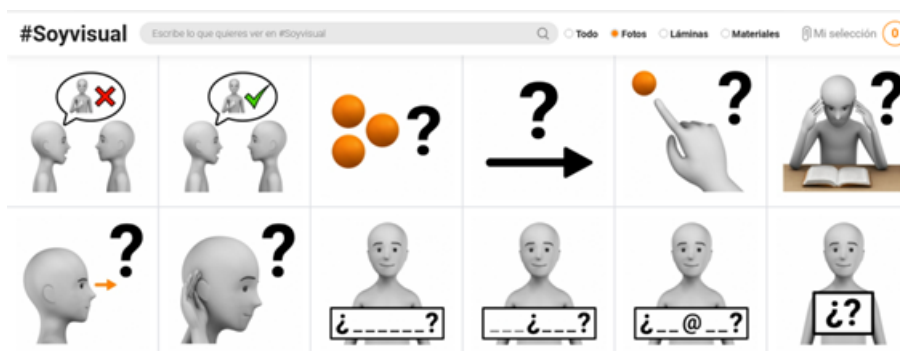


Figura 2.2: Mostra do uso da páxina web de Soyvisual.

2.3 Cboard

Cboard conta con pictogramas de todo tipo e permite construír frases con eles a medida que son seleccionados. Ademais, ten un bo índice de personalización e permite engadir novos pictogramas personalizados con imaxes propias, código de cores ou audio. A parte negativa desta aplicación é que non permite reproducir as frases unha vez creadas e que as imaxes dos pictogramas necesitan ser engadidas manualmente polo usuario, non hai forma de obtelas directamente dunha [Application Programming Interface \(API\)](#).

Ademais de para Android, esta aplicación está tamén dispoñible en navegadores web co

mesmo funcionamento exacto para aquelas persoas que non dispoñan dun dispositivo Android [10].

Na figura 2.3 (páxina 10), pode verse unha captura da aplicación coa demostración da creación de frases a partir dos diferentes pictogramas.



Figura 2.3: Mostra do uso da aplicación Cboard.

2.4 DictaPicto

DictaPicto é unha aplicación que permite pasar de voz ou escrito a pictogramas de forma inmediata. Inclúe recoñecemento de voz e un vocabulario baseado na biblioteca de pictogramas creada polo goberno rexional de Aragón, e que se denomina *Aragón Sistemas Aumentativos e Alternativos de Comunicación (ARASAAC)*. Este conxunto de pictogramas son de libre utilización (baixo a licenza correspondente) e acadaron gran popularidade entre os profesionais que traballan con persoas con *TEA*. A aplicación DictaPicto permite ampliar e actualizar o vocabulario inicial co que conta, eliminando ou editando termos [11].

2.5 Táboa comparativa

Na táboa 2.1 (páxina 11), pódese ver a comparación entre as ferramentas previamente mencionadas de acordo ás funcionalidades que inclúen. Por outra banda, na táboa 2.2 (páxi-

na 12), recóllese a dispoñibilidade destas aplicacións nos distintos dispositivos, ademais dos prezos de cada ferramenta.

Podemos observar que, con respecto ás funcionalidades, a máis completa sería Cboard, xa que permitiría a agrupación dos pictogramas por categorías, a opción de reproducilos e a creación de frases. Aínda así, algúns dos seus puntos negativos con respecto a isto, estarían na opción do audio debido a que, para que esta opción funcione correctamente, o usuario ten que grabarse a si mesmo dicindo o nome do pictograma, en lugar de permitir reproducilos todos sen o esforzo adicional de ter que grabalos. Así mesmo, esta aplicación non dispón de versión para iOS, e non usa os pictogramas de *ARASAAC*, o que é unha gran desvantaxe, debido a que moitos nenos xa están acostumados a istos pictogramas. O resto de aplicacións, aínda que todas teñen as opcións de pictogramas e creación de frases, non todas ofrecen as posibilidades de agrupar os pictogramas por carpetas ou a reprodución dun *feedback* de audio asociado á pulsación do pictograma.

Funcionalidade Aplicación	Pictogramas	Categorías	Audio	Frases
Leeloo AAC	Si	Non	Si	Si
Soyvisual	Si	Non	Non	Si
Cboard	Si	Si	Si	Si
DictaPicto	Si	Non	Non	Si

Táboa 2.1: Táboa comparativa considerando as funcionalidades de cada aplicación.

Neste proxecto, intentarase crear unha aplicación, que sexa válida tanto para Android como para iOS e que permita solucionar os problemas atopados nas outras aplicacións previamente estudadas, incluíndo todas as funcionalidades necesarias, como serían a reprodución de frases e de todos os pictogramas.

Característica Aplicación	Prezo	Sistema Operativo
Leeloo AAC	Gratuito con con- tido de pago	iOS e Android
Soyvisual	Gratuito	Navegadores web, iOS, Android
Cboard	Gratuito	Navegadores web, Android
DictaPicto	Gratuito	iOS e Android

Táboa 2.2: Táboa comparativa de prezos e sistemas.

Fundamentos tecnolóxicos

Para o desenvolvemento da aplicación e da memoria, foron necesarias varias ferramentas e tecnoloxías que serán listadas a continuación, e que foron elixidas valorando outras opcións tamén viables.

3.1 Flutter

Flutter [12] é un *framework* de código aberto creado por Google que serve para construír interfaces de usuario para varias plataformas, entre as que destacan Android, iOS, web, Windows, MacOS e Linux. Flutter baséase no uso de *widgets*, que se poden combinar para conseguir distintas funcionalidades de tal forma que, ao final, toda a aplicación está composta de *widgets* que están a súa vez formados doutros máis pequenos. O feito de elixir esta tecnoloxía para o desenvolvemento da aplicación fundaméntase en que permite xerar aplicacións nativas tanto para Android como para iOS, sendo estes os sistemas operativos máis usados, a partir dun único código fonte.

Á parte de Flutter, outro *framework* que foi considerado para o desenvolvemento da aplicación foi React Native e, aínda que ten un alto rendemento, descartouse debido a que Flutter tiña a gran vantaxe de ser multiplataforma e de contar cun alto grao de personalización debido á posibilidade de usar unha gran variedade de *widgets*.

3.2 ARASAAC

ARASAAC [4] é un proxecto que ofrece recursos gráficos e materiais para facilitar a comunicación e a accesibilidade cognitiva das persoas con TEA ou con outros tipos de diversidade que presenten dificultades nestes ámbitos. Entre outras funcionalidades, esta ferramenta conta cunha importante colección de pictogramas e cunha API REST que permite acceder ao contido desde a aplicación, o que resultou moi vantaxoso e cómodo xa que facilita que se poi-

dan incorporar pictogramas. [ARASAAC](#) é un proxecto coordinado e financiado polo Goberno de Aragón e recibiu numerosos premios e recoñecementos.

Neste proxecto, implementouse a opción de engadir os pictogramas podendo buscalos na [API](#) que nos ofrece [ARASAAC](#). Decidiuse usar esta ferramenta, debido a que, ademais de que moitos nenos e nenas, xa están familiarizados cos seus pictogramas, tamén ofrece a [API](#) que permite acceder a eles desde a aplicación.

3.3 **Firestore**

Firestore [13] é unha plataforma de desenvolvemento de aplicacións móbiles que ofrece distintos módulos para o *backend* como son a autenticación, estatísticas, bases de datos ou almacén de arquivos, entre outras. Todas estas funcionalidades que se ofrecen desde Firestore reducen considerablemente o traballo do programador, diminuindo a cantidade de esforzo que se precisa realizar no *backend* da aplicación. As seguintes subseccións céntranse en falar das funcionalidades concretas de Firestore empregadas neste proxecto.

3.3.1 **Firestore Authentication**

Firestore Authentication [14] proporciona servizos *backend* para autenticar aos usuarios dunha app. Admite a autenticación por contrasinais, números de teléfono ou incluso con contas de Google, Twitter ou Facebook. Esta ferramenta permite implementar de forma sinxela as operacións de rexistro e autenticación dos usuarios que farán uso da aplicación, ou a recuperación de contrasinais, entre outras funcións.

3.3.2 **Firestore Database**

Firestore [15] é unha base de datos NoSQL flexible e escalable. Esta ferramenta permite crear coleccións, onde se almacenarán os documentos cos datos dos usuarios da aplicación, e dos pictogramas e categorías, xunto coas referencias ás súas imaxes. Foi escollida porque permite sincronizar os datos entre distintos dispositivos de forma eficiente. Outra consideración para o almacenaxe de datos, foi Realtime Database, pero descartouse debido a que en Firestore, era máis sinxelo a organización dos datos, con subcoleccións dentro dos documentos. Ademais, Firestore Database permite máis escalabilidade e confianza.

3.3.3 **Firestore Storage**

Firestore Storage [16] é un servizo de almacenamento para gardar o contido xerado polos usuarios da aplicación, como son as fotos de perfil, ou as imaxes dos pictogramas e das carpetas (categorías).

3.4 Dart

Dart [17] é unha linguaxe de programación usada para o desenvolvemento de aplicacións en calquera plataforma. Foi elixida principalmente porque é a linguaxe que se emprega para a implementación usando o *framework* de Flutter. Está orientada a obxectos e é de tipado estático.

Dart dispón de moitas vantaxes [18], entre as que se encontran:

- É de acceso gratuíto para todos.
- Os desenvolvedores simplificaron as características complicadas doutras linguaxes, polo que Dart é bastante sinxelo de aprender. No seu sitio web, pódense atopar varios tutoriais para iniciarse.
- Funciona en todos os navegadores móbiles e de escritorio actuais.
- Detrás da súa programación, atópase Google, polo que ofrece perspectivas a longo prazo para o desenvolvemento da linguaxe.

3.5 Android Studio

Android Studio [19] é o *Integrated development environment (IDE)* oficial para o desenvolvemento de aplicacións en Android e inclúe todo o necesario para construír, compilar e xerar os executables das aplicacións para este sistema operativo. Está baseado no editor de código IntelliJ IDEA e ofrece todas as ferramentas necesarias para xerar a lóxica e deseñar a interfaz de usuario da aplicación. Tamén inclúe un compilador, baseado en Gradle, que é unha ferramenta que permite automatizar o proceso de compilación. Ademais, dispón de unha boa cantidade de recursos e ferramentas para probas e permite modificar fragmentos de código sen a necesidade de reiniciar a aplicación.

Foi escollido para este proxecto xa que permite emular dispositivos virtuais Android para poder realizar probas en dispositivos con diferentes características a medida que se vai desenvolvendo o código.

3.6 Git

Git [20] é un sistema de control de versións distribuído que permite facer clons locais do proxecto do repositorio que permiten traballar en remoto sen conexión e sen ningún tipo de problema.

Git ofrece varias vantaxes [21]:

- Sistema distribuído: Permite o traballo incluso sen conexión.
- Rápido e lixeiro: Está optimizado para facer operacións moi rápidas.
- Creación de ramas: Dispón da opción de crear varias ramas, que permite que varias persoas poidan traballar en distintas funcionalidades para despois xuntalo todo.
- Gracias ao seu modelo de almacenamento, garantiza a integridade da información.
- É gratuíto e de código aberto.

Algunhas opcións que se valoraron para usar Git foron Gitlab e Github, pero finalmente escolleuse Github porque a estudante estaba máis familiarizada con el.

3.6.1 Github

Github [22] é unha plataforma de xestión e organización de proxectos baseada na nube e que incorpora as funcións de control de versións de Git. Toda a xestión do repositorio fíxose a través de comandos, polo que non foi necesario ningún software de interfaz gráfico adicional.

3.7 Balsamiq Wireframes

Balsamiq [23] foi a ferramenta elixida para crear os *wireframes* e *mockups* da aplicación. Conta cunha serie de *widgets*, dispoñibles directamente na páxina web, para poder realizar os bosquexos de forma sinxela e rápida. Ademais, habilita a opción de enlazar distintas partes dos *mockups*, permitindo ver máis fielmente como sería o funcionamento da aplicación. É unha aplicación de pago, pero dispón dun período de proba dun mes polo que non foi necesario recurrir ao plan de pago. Entre outras razóns, elixiuse esta alternativa pola súa facilidade de uso e debido a que xa era unha ferramenta coñecida pola estudante.

3.8 draw.io (Diagrams.net)

A ferramenta draw.io [24] é unha aplicación informática deseñada para a creación dunha gran variedade de diagramas. Neste proxecto, foi usada para crear os diagramas correspondentes aos casos de uso durante a fase do análise, e elixiuse pola súa simplicidade de uso e por poder ser usada nun navegador web, en lugar de ter que descargar e instalar en local un programa como pode ser Dia.

3.9 Wrike

Wrike [25] é un software de xestión de proxectos que permite organizar proxectos en Scrum, ofrecendo unha pizarra para poder ver as tarefas divididas nas seccións de Nova, En curso, Finalizada e en Revisión, ademais de dispoñer da creación de diagramas de Gantt, entre outras funcionalidades. Neste proxecto, usuose esta ferramenta para a organización visual dos sprints.

Esta é unha plataforma de pago, pero conta cunha versión de proba gratuíta e pódese usar todo no navegador web sen necesidade de descargar ningunha aplicación. Na imaxe 3.1, podemos ver a “pizarra” que nos ofrece Wrike, no estado final do proxecto e con todas as tarefas en completado.

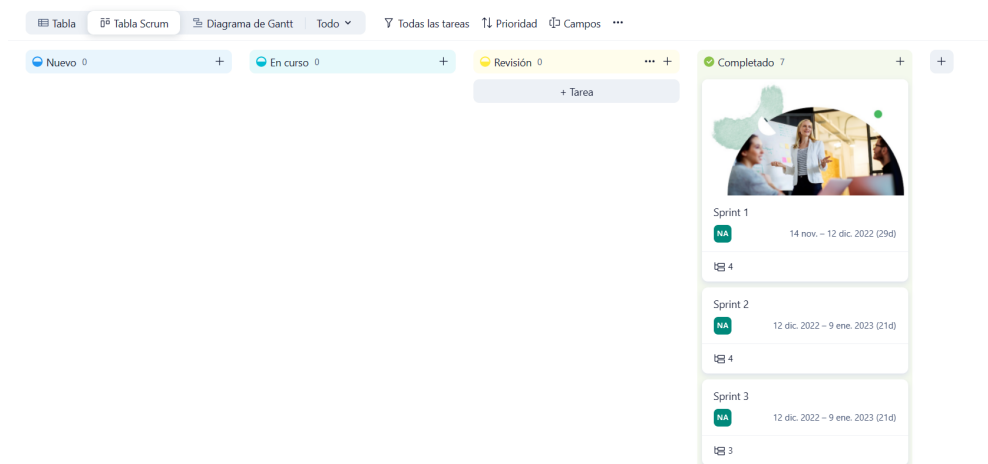


Figura 3.1: “Pizarra” ofrecida pola ferramenta Wrike.

3.10 Latex

Latex [26] é un sistema de composición de textos, orientado á creación de artigos e documentos científicos que inclúen, entre outras cousas, expresións matemáticas. Elixiuse debido a que é o estándar para documentos científicos e permite elaborar documentación cunha alta calidade tipográfica, ademais de que xa se ofrece un modelo en Latex para o desenvolvemento da memoria do TFG.

3.11 Overleaf

Overleaf [27] é unha ferramenta de edición de textos en Latex que permite facer este sistema accesible desde a web para a redacción de documentos científicos. Foi escollida para este proxecto debido ás facilidades que dá no uso de Latex, á posibilidade de editar colaborativamente documentos, á opción de engadir comentarios sobre o texto e á vantaxe de producir directamente os ficheiros en formato pdf sen necesidade de ter que instalar nada no ordenador local.

Metodoloxía

O proceso e a metodoloxía de traballo seguido é un factor moi importante para poder entregarlle un traballo de calidade ao cliente. O método de traballo elixido para este proxecto foi Scrum para o desenvolvemento do produto software combinado coa metodoloxía de ApS para a elaboración do proxecto e a súa validación.

4.1 Scrum

Unha metodoloxía áxil [28] é un conxunto de técnicas aplicadas en ciclos de traballo curtos co obxectivo de que o proceso de entrega dun proxecto sexa máis eficiente. Isto é así porque, con cada etapa completada, pódense entregar avances e xa non é necesario esperar ata a finalización completa do proxecto. Hai bastantes tipos de metodoloxías áxiles, entre as que se encontran Kanban, Scrum, Extreme Programming, Lean, Scaled Agile Framework ou Nexus. Neste proxecto decidiuse usar Scrum.

Scrum [29] é un marco de xestión de proxectos que axuda aos equipos a estruturar e xestionar o traballo a realizar mediante un conxunto de valores, principios e prácticas. Inclúe un conxunto de reunións, ferramentas e funcións que, de forma coordinada, axudan a xestionar o traballo e mellorar a produtividade. O marco de traballo desta metodoloxía baséase na aprendizaxe continua e na adaptación a factores cambiantes. Neste sentido, recoñécese que o equipo evolucionará a través da experiencia.

Esta metodoloxía permite fragmentar proxectos longos e complexos en desenvolvementos máis pequenos con prazos de entrega curtos, o que ofrece unha serie de vantaxes [30]:

1. Claridade nos obxectivos e alcances: O cliente establece os requisitos e datas nas que quere que esté completado, e despois realízanse entregas para comprobar que efectivamente estánse cumprindo os obxectivos.

2. Flexibilidade: A metodoloxía é flexible e ofrece unha alta capacidade de reacción ante os cambios solicitados polo cliente. Isto é así porque, se algunha modificación é solicitada, só se aplicará a unha pequena parte do proxecto. Do mesmo xeito, anticipa a detección de erros no desenvolvemento e minimiza o impacto destes.
3. Maior control de imprevistos: O cliente xa coñece as funcionalidades máis importantes do proxecto antes de que este finalice, polo que se reduce a posibilidade de sorpresas ao final.
4. Equipos altamente produtivos: Os equipos teñen completa liberdade para organizarse, planear tempos e formas de traballo, o que ofrece unha gran motivación.
5. Predicibilidade: É fácil predicir a velocidade coa que traballa o equipo, podendo determinar con certa precisión cando se terá unha certa funcionalidade e as posibles desviacións que poidan aparecer.
6. Fácil de implementar: É unha metodoloxía fácil de aprender. Os roles, eventos e artefactos son claros.

4.1.1 Sprints

Os proxectos en Scrum execútanse en ciclos temporais curtos e de duración fixa, denominados sprints [31]. Adoitan durar 2 semanas, aínda que nalgúns equipos poden chegar ata 3 ou 4 semanas. Cada iteración ten que proporcionar un resultado completo e funcional que poida ser entregado se o cliente así o solicita. Para este proxecto, decidiuse que as iteracións serían de 4 semanas, debido a incapacidade do equipo de poder alcanzar os obxectivos en tan só dúas semanas ao ser un equipo formado por unha soa persoa.

As actividades levadas a cabo en cada iteración nun proceso Scrum son as seguintes:

1. Planificación da iteración: Consiste nunha reunión na que se determinará a selección de requisitos que se prevee que se poderán completar nesa iteración, e na elaboración da lista de tarefas para cumprimentar os requisitos previamente establecidos.
2. Execución da iteración: Nunha metodoloxía Scrum, débese realizar unha reunión de sincronización diaria de aproximadamente 15 minutos, onde se analizará o traballo feito polo resto de membros do equipo para poder realizar as adaptacións necesarias que permitan cumprir coa previsión de obxectivos a conseguir ao final da iteración. Como xa foi mencionado anteriormente, ao contar cun equipo de desenvolvemento de tan só unha persoa, estas reunións non se consideraron necesarias.

3. Inspección e adaptación: O último día da iteración, realízase unha reunión de revisión para poder demostrar que se cumpriron os obxectivos previamente establecidos, e unha retrospectiva para poder comentar cales son os problemas que poderían impedir un progreso adecuado e posibles estratexias de mellora.

4.1.2 Membros dun equipo Scrum

Un equipo Scrum adoita estar composto por grupos de entre 3 a 9 membros, sen contar ao *Scrum Master* e o *Product Owner* [32]. Neste caso e debido á natureza do proxecto, o equipo de desenvolvemento só contou cun membro. A continuación, detállanse os diferentes roles que establece esta metodoloxía, xunto coas súas principais funcións:

1. *Product Owner*: Ten a responsabilidade de decidir o traballo que é necesario facer e de maximizar o valor do produto. Adicionalmente, tamén realiza o traballo de interlocutor cos *stakeholders*, persoas ou empresas cuxo apoio permiten que unha organización exista [33], e actúa de intermediario entre o cliente e os desenvolvedores. No caso deste proxecto, os *Product Owners* foron os titores.
2. *Scrum Master*: Ten dúas responsabilidades principais que son, por un lado, xestionar o proceso Scrum asegurando que se seguen os preceptos da metodoloxía e, por outro, axudar a eliminar os impedimentos que poidan afectar á entrega do produto. Esta función tamén foi desempeñada polos titores.
3. Equipo de desenvolvemento: Está formado polos profesionais encargados de desenvolver o produto, auto-organizándose para que estea listo para o final do ciclo de desenvolvemento correspondente. Como se mencionou previamente, este equipo está conformado só pola alumna.

4.1.3 Artefactos de Scrum

Os artefactos en Scrum [34] son aqueles elementos que garanten a transparencia e o rexistro de información fundamental no proceso Scrum. Son os seguintes:

1. *Product backlog*: É unha lista ordenada con todo o que necesita un proxecto para cumprir coas necesidades do cliente. O responsable deste artefacto é o *Product Owner*, encargado de engadir e ordenar os ítems da lista, que necesitan contar cunha descrición, ordenación, estimación e valor. Os elementos que están máis arriba son os que están máis preto de ser desenvolvidos.
2. *Sprint backlog*: É un subconxunto de elementos do *Product backlog*, elixidos para ser abordados no sprint co propósito de lograr o obxectivo desa iteración. Os membros do equipo de desenvolvemento son os encargados de modificalo.

3. Incremento: Trátase do resultado do sprint, debe estar rematado e listo para ser usado. Poderíase dicir que é unha nova versión máis completa e funcional do produto.

4.2 Aprendizaxe e Servizo

O ApS [35] [36] é unha metodoloxía que consiste en aprender facendo un servizo pola comunidade. Parte da fusión entre unha intencionalidade pedagóxica e unha intencionalidade solidaria, onde se aprende sobre necesidades reais da contorna co obxectivo de melloralas. É un método de traballo bastante útil xa que permite traballar con necesidades reais do entorno, ademais de ser un dos mellores métodos de aprendizaxe, xa que permite aplicar os coñecementos adquiridos durante a formación nunha práctica solidaria que pode contribuír a facer un mundo mellor e mellorar a calidade de vida de colectivos vulnerables.

O ApS conta con numerosas vantaxes, entre as que se atopan:

- Fomenta a aprendizaxe cooperativa ao implicar un traballo en equipo coa comunidade.
- Favorece o desenvolvemento persoal e o progreso social xa que permite desenvolver habilidades de comunicación ou de resolución de problemas entre outras, á vez que permite construír relacións positivas coa comunidade e adquirir unha maior empatía coas demais persoas.
- Permite desenvolver un compromiso solidario ao estar destinado a atender as necesidades reais dunha comunidade de persoas.
- Fomenta unha aprendizaxe máis activa e significativa debido a que permite aplicar os coñecementos adquiridos durante a formación dun xeito que poida mellorar a sociedade e que permita encontrarlle o sentido ao que se estudou.

Neste proxecto, utilizouse esta metodoloxía porque, como xa se indicou no capítulo 1, o obxectivo é poder crear unha aplicación que permita facilitarlle a profesionais e familiares de nenos con TEA o uso de pictogramas e categorías sen ter que complicarse coa creación física destes pictogramas. Neste sentido, o feito de aplicar unha metodoloxía baseada en ApS, na que se colabora con entidades sociais da contorna que traballan con este colectivo, resulta fundamental para deseñar e implementar unha ferramenta adaptada e que sexa realmente útil na vida cotiá destas persoas e nas intervencións dos profesionais.

Organización do proxecto

A organización dun proxecto é unha parte moi importante deste. Esta organización permite anticiparnos a problemas que poidan xurdir e axuda a establecer a prioridade de cada tarefa para ter mellor control do tempo. Ademais, é un aspecto fundamental para poder executar con éxito o proxecto e obter resultados finais satisfactorios. Neste capítulo, esbozaremos como se organizou temporalmente o proxecto de acordo aos principios de Scrum.

5.1 Tarefas

Unha tarefa é a unidade de traballo na que se divide todo proxecto, e débese levar a cabo nun tempo establecido para poder obter os obxectivos finais. Dado que o proxecto utiliza a metodoloxía Scrum, as diferentes tarefas que se necesitaron realizar para cumprir os obxectivos do mesmo foron organizadas en sprints de 4 semanas de duración. Estes sprints incluían ademais unha reunión ao final cos titores para valorar o traballo realizado e planificar o seguinte sprint, cunha duración aproximada de 2 horas cada unha. Previamente, realizouse unha análise onde se determinaron os actores e os requisitos do proxecto, unha fase de deseño onde se detallaron as vistas da aplicación e, para finalizar, fóra dos sprints propiamente de desenvolvemento, tamén se realizou a redacción desta memoria.

Debido a ter que compaxinar o desenvolvemento deste proxecto co traballo, a estudante dedicou unha media de 2 horas diarias as tarefas que agora se mencionarán.

5.1.1 Sprint 1

Este primeiro sprint comprendeu os días entre o 14 de novembro e o 12 de decembro, e intentouse que incluíra as primeiras tarefas relacionadas coa creación e xestión dos usuarios da aplicación:

- T1.1 Creación de usuarios.

- T1.2 Inicio de sesión.
- T1.3 Recuperar contraseña.
- T1.4 Cerrar sesión.

5.1.2 Sprint 2

O seguinte sprint estendeuse desde o 12 de decembro ata o 9 de xaneiro. Nesta iteración, intentouse incluír a implementación dun módulo, que a partir de agora será o do responsable (do neno), que permitira xestionar os distintos perfís que interactúan coa aplicación: creando usuarios [TEA](#), asignándolle profesionais que traballan con eles, etc. A lista de tarefas asignadas a este sprint foron:

- T2.1 Dar de alta usuarios [TEA](#).
- T2.2 Listado de usuarios [TEA](#).
- T2.3 Asignar/designar profesionais.
- T2.4 Eliminar usuarios [TEA](#).

5.1.3 Sprint 3

Comprende desde o 9 de xaneiro ata o 6 de febreiro. Aquí xa se incluíron a creación e xestión das distintas categorías dispoñibles nas que se organizan os pictogramas:

- T3.1 Engadir carpetas.
- T3.2 Listado de carpetas.
- T3.3 Eliminar carpetas.

5.1.4 Sprint 4

Tivo lugar desde o 6 de febreiro ata o 6 de marzo. Neste sprint, abordouse todo o relacionado coa creación de pictogramas e coa súa xestión:

- T4.1 Engadir pictogramas.
- T4.2 Listado de pictogramas.
- T4.3 Eliminar pictogramas.

5.1.5 Sprint 5

Este sprint foi programado para durar desde o 6 de marzo ata o 3 de abril. Nel impleméntouse a parte da aplicación correspondente ao usuario [TEA](#), onde se ofrezan as posibilidades comunicativas establecidas polas diferentes fases do método [PECS](#):

- T5.1 Creación de frases a partir de pictogramas.
- T5.2 Reprodución de audio asociado a pictogramas e frases.

5.1.6 Sprint 6

Esta iteración estendeuse desde o 3 de abril ata o 1 de maio, e incluíu a parte da configuración da aplicación para permitir un maior grao de personalización:

- T6.1 Configuración de usuario.
- T6.2 Personalización do perfil do usuario [TEA](#).

5.1.7 Sprint 7

Desde o 1 de maio ata o 29 de maio, nesta etapa fixéronse os retoques solicitados ou suxeridos na aplicación e as vistas adaptadas para *tablet*:

- T7.1 Activación e desactivación de pictogramas.
- T7.2 Retoques visuais.
- T7.3 Vistas para *tablet*.

5.2 Diagrama dos sprints

Nas imaxes [5.1](#) (páxina 26) e [5.2](#) (páxina 26), pódese observar máis visualmente a organización final dos diferentes sprints nos que se dividiu o ciclo de desenvolvemento, así como as súas datas de inicio e final, mediante o emprego da plataforma Wrike.

Tabla Tabla Scrum Diagrama de Gantt Todo 0 filtros Asignado Campos Expandir						
	Nombre	Asignado	Estado	Fecha de inici	Fechas de ven	+
1	▼ Sprint 1	NA Nuria Agra Rial	Completado	14/11/2022	12/12/2022	
2	Cerrar sesión	NA Nuria Agra Rial	Completado			
3	Creación de usuarios	NA Nuria Agra Rial	Completado			
4	Inicio de sesión	NA Nuria Agra Rial	Completado			
5	Recuperar contrasinal	NA Nuria Agra Rial	Completado			
6	▼ Sprint 2	NA Nuria Agra Rial	Completado	12/12/2022	09/01/2023	
7	Asignar/Desasignar profesionais	NA Nuria Agra Rial	Completado			
8	Dar de alta usuarios TEA	NA Nuria Agra Rial	Completado			
9	Eliminar usuarios TEA	NA Nuria Agra Rial	Completado			
10	Listado de usuarios TEA	NA Nuria Agra Rial	Completado			
11	▼ Sprint 3	NA Nuria Agra Rial	Completado	12/12/2022	09/01/2023	
12	Eliminar carpetas	NA Nuria Agra Rial	Completado			
13	Engadir carpetas	NA Nuria Agra Rial	Completado			
14	Listado de carpetas	NA Nuria Agra Rial	Completado			
15	▼ Sprint 4	NA Nuria Agra Rial	Completado	06/02/2023	06/03/2023	
16	Eliminar pictogramas	NA Nuria Agra Rial	Completado			
17	Engadir pictogramas	NA Nuria Agra Rial	Completado			
18	Listado de pictogramas	NA Nuria Agra Rial	Completado			

Figura 5.1: Planificación desde o sprint 1 ao 4.

19	▼ Sprint 5	NA Nuria Agra Rial	Completado	06/03/2023	03/04/2023	
20	Creación de frases	NA Nuria Agra Rial	Completado			
21	Reprodución de audio	NA Nuria Agra Rial	Completado			
22	▼ Sprint 6	NA Nuria Agra Rial	Completado	03/04/2023	01/05/2023	
23	Configuración de usuario	NA Nuria Agra Rial	Completado			
24	Personalización do perfil do usuario TEA	NA Nuria Agra Rial	Completado			
25	▼ Sprint 7	NA Nuria Agra Rial	Completado	01/05/2023	29/05/2023	
26	Activación e desactivación de pictogram	NA Nuria Agra Rial	Completado			
27	Retoques visuais	NA Nuria Agra Rial	Completado			
28	Vistas tablet	NA Nuria Agra Rial	Completado			

Figura 5.2: Planificación desde o sprint 5 ao 7.

5.3 Seguemento

O seguemento do proxecto realizouse mediante as reunións ao final de cada sprint. Nestas reunións, comentouse o traballo realizado durante o sprint, analizando e probando as funcionalidades engadidas e suxerindo posibles cambios a estas. Todos estes cambios solicitados, leváronse a cabo durante o seguinte sprint, sempre mantendo a duración previamente establecida para este. Desde o principio, intentouse que cada sprint non acaparara demasiadas tarefas xa que, como se mencionou previamente, a estudante tiña que compaxinar este proxecto co seu traballo, polo que non houbo ningún retraso significativo na execución do proxecto.

Na táboa 5.1 (páxina 27), podemos ver as datas de inicio e fin, o número de días e o total de horas traballadas polo equipo de desenvolvemento no proxecto.

Duración	
Data inicio	14/11/2022
Data fin	29/05/2023
Nº de días	155 días
Horas de traballo	310 horas

Táboa 5.1: Táboa de seguemento do proxecto.

5.4 Recursos

Un dos primeiros pasos na planificación dun proxecto é determinar cales serán os recursos que se precisan para a súa elaboración, determinando o que será necesario para poder levalo a cabo. Os recursos dun proxecto non só inclúen os recursos materiais, senón que tamén hai que considerar os recursos humanos.

5.4.1 Recursos humanos

Os recursos humanos serán os encargados de realizar as diferentes tarefas do proxecto. Neste proxecto, identificáronse os seguintes roles asociados aos recursos humanos:

1. Xefe do proxecto: Ten varias funcións principais, entre as que se encontran a xestión do proxecto e a de supervisar que todo funcione segundo o plan establecido. Ademais, tamén son os encargados de informar ao cliente dos avances do proxecto. Neste caso, este rol foi desempeñado polos titores do traballo.
2. Analista: O seu traballo é o de identificar as necesidades e requisitos do cliente para poder comunicarllos despois aos programadores e deseñadores. Este rol foi desempeñado pola estudante.
3. Deseñador: É o encargado de deseñar o sistema en función dos requisitos obtidos polo analista. Este rol tamén é desempeñado pola estudante.
4. Programador: O seu obxectivo é implementar o sistema definido previamente polo deseñador, así como encargarse das probas necesarias para o proxecto. Este rol tamén é desempeñado pola estudante.

5.4.2 Recursos materiais

Nesta categoría, inclúense todos os recursos materiais necesarios para o desenvolvemento do proxecto, incluíndo os programas de *software* utilizados. No caso deste proxecto, os

recursos materiais necesarios foron os seguintes:

1. Portátil: Ordenador portátil HP 15s-fq2038ns con 8GB de RAM, procesador Intel Core i5-1135G7 de 11ª xeración e tarxeta gráfica integrada Intel Iris Xe. Foi usado durante o desenvolvemento do código da aplicación e supuxo un custo de aproximadamente 700€.
2. Móbil: Redmi 8 co sistema operativo Android e unha pantalla de 6,2 pulgadas. Usouse para levar a cabo as probas da aplicación nun móbil físico Android e tivo un custo de 115€.
3. Recursos software: Foron xa especificados no capítulo 3 e, ao ser todos gratuítos ou usados nos períodos de proba, como é o caso de Balsamiq, non supuxeron ningún custo adicional no proxecto.

5.5 Custos

Para a estimación dos custos asociados ao proxecto, tivéronse en conta tanto os recursos humanos como os materiais.

En canto aos salarios dos distintos roles, podemos ver na táboa 5.2 os salarios medios para cada tipo de traballador [37] [38] [39].

Rol	Salario anual	Salario por hora
Xefe de proxecto	37.761€	18,87€
Analista	28.151€	14,06€
Deseñador	23.815€	11,89€
Programador	19.000€	9€

Táboa 5.2: Táboa de salarios.

Nas táboas 5.3 (páxina 29) e 5.4 (páxina 29), pódese observar o custo total respecto aos recursos humanos e materiais empregados no proxecto, respectivamente, e no caso dos humanos, tamén se mostra o total de horas realizadas por cada tipo de traballador. Como se pode apreciar, o gasto necesario en recursos humanos sería de 3.189,67€, mentres que para os recursos materiais necesitaríanse investir 815€. Para finalizar, na táboa 5.5 (páxina 29), móstrase o custo final total do proxecto.

Recurso	Horas	Total
Xefe de proxecto 1	16	301,92€
Xefe de proxecto 2	16	301,92€
Analista	8	112,48€
Deseñador	15	178,35€
Programador	255	2295€
TOTAL	310	3.189,67€

Táboa 5.3: Táboa de custos en recursos humanos.

Recurso	Total
Portátil	700€
Móbil	115€
TOTAL	815€

Táboa 5.4: Táboa de custos en recursos materiais.

Recurso	Horas	Total
Recursos humanos	310	3189,67€
Recursos materiais		815€
TOTAL	310	3189,67€

Táboa 5.5: Táboa de custos totais.

Capítulo 6

Análise

A primeira fase dun proxecto software é a de análise. Nesta etapa inténtase deducir o que a aplicación necesita, quen usará a aplicación e como o fará.

6.1 Actores

Os actores serán as persoas alleas á aplicación que interactuarán con ela. Encontráronse catro actores diferenciados neste proxecto: o usuario non autenticado, o responsable, o profesional e o usuario [TEA](#).

6.1.1 Usuario non autenticado

Será o actor que aínda non se autenticou na aplicación. As funcionalidades ás que terá acceso serán a creación dunha conta, a posibilidade de iniciar sesión e a de recuperar o contrasinal.

6.1.2 Responsable

O responsable é a persoa que se encarga dos nenos e nenas que usarán a aplicación. En principio, será o pai, nai ou titor desa persoa con [TEA](#). Ten dispoñibles as funcionalidades de crear os nenos, de xestionar os seus perfís e de asignarlles os profesionais que poden traballar con eles. Este actor tamén contará coas funcionalidades do profesional, incluídas a creación de pictogramas e categorías. Por último, tamén pode xestionar e modificar os seus propios datos.

6.1.3 Profesional

Os profesionais serán aquelas persoa designadas polo responsable que realizan algún tipo de terapia ou intervención cos nenos, podendo ser un titor da escola, o logopeda, etc. As súas

funcionalidades, como xa se indicou previamente, serían a xestión dos cartafóis e pictogramas, incluíndo a súa creación, modificación ou eliminación. Adicionalmente, unha das suxerencias que saíron para os cartafóis, foi que o responsable os poidera marcar como editables ou non editables, de forma que nos non editables, os profesionais non poderían engadir pictogramas ou modificalos. Desta forma aumentaríase a seguridade destes para o responsable. Esta funcionalidade non chegou a implementarse por falta de tempo, pero tívose en consideración para un futuro.

6.1.4 Usuario TEA

Este actor será o propio neno ou nena que use a aplicación. As súas funcionalidades serán a interacción coa aplicación, usando os pictogramas e creando frases para comunicarse ou responder a preguntas.

6.2 Requisitos

Os requisitos son as condicións que deben de cumprirse para validar un sistema. Existen dous tipos de requisitos no desenvolvemento *software*: os funcionais e os non funcionais. Estes requisitos, foron previamente establecidos nunha reunión entre os titores e as asociacións da contorna, onde se estableciu como querían que se comportase a aplicación e as funcionalidades que debería de ter.

6.2.1 Requisitos funcionais

Os requisitos funcionais describen especificamente as accións que debe poder realizar a aplicación. Cada actor ten os seus propios requisitos funcionais como se foi introducindo na sección anterior. A continuación, detallamos estes requisitos para cada tipo de usuario:

- Usuario non autenticado:
 1. Darse de alta no sistema.
 2. Iniciar sesión.
 3. Solicitar a recuperación do contrasinal.
- Responsable:
 1. Xestión dos usuarios **TEA**: o responsable debe encargarse da creación destes usuarios, da súa eliminación se se require, da personalización dos seus perfís e da asignación dos profesionais que traballarán con eles.
- Profesional:

1. Xestión das categorías: O profesional pode encargarse da creación de categorías e da súa eliminación.
 2. Xestión dos pictogramas: Tamén pode crear pictogramas dentro das correspondentes categorías, encargarse da súa activación e desactivación para permitir que os nenos e nenas poidan ou non poidan velo, e da súa eliminación se o considera necesario.
- Usuario TEA:
 1. A función principal deste actor é interactuar cos pictogramas e ten a posibilidade de crear frases mediante a combinación de diferentes pictogramas.

6.2.2 Requisitos non funcionais

Os requisitos non funcionais indican as características que debe ter a aplicación. Estes requisitos non teñen ningún impacto na funcionabilidade da aplicación, xa que esta seguirá funcionando sen eles, pero mellora bastante a súa calidade, xa que conseguen que o produto sexa atractivo, útil, rápido, seguro ou fiable.

Estes requisitos poden dividirse en varias categorías:

- Requisitos de produto: Incluirían a usabilidade, a fiabilidade, a seguridade ou a escalabilidade, entre outros.
- Requisitos relacionados coa organización: Nesta categoría entrarían a entrega ou a implementación.
- Requisitos externos: Serían, por exemplo, a lexislación ou a privacidade.

Neste proxecto, o requisito non funcional máis claro foi a aplicación da seguridade. Ademais de implementar un sistema de autenticación para os usuarios, fíxose énfase na restrición de certas funcionalidades a algúns actores, como serían a xestión dos usuarios [TEA](#), para que só o responsable poida modificar o perfil dos nenos e nenas que creou, ademais de dicir que profesionais poden traballar con eles.

Ademais, tamén é interesante destacar a usabilidade nas interfaces, xa que se intentou facer o máis simple e fácil de comprender para que os nenos e nenas que a usen non teñan ningún problema. Cabe destacar tamén que se usou unha linguaxe do mundo real para que todos os usuarios poidan entender a aplicación sen usar ningún tipo de xerga interna, tanto nos mensaxes de erros como nas funcionalidades da aplicación, e que se deu un certo control

nas accións que poden realizar os usuarios, sempre preguntando se está seguro, por exemplo, no caso de intentar borrar unha categoría, para evitar posibles erros que se poidan cometer. As interfaces ademais, teñen un deseño minimalista, evitando conter información irrelevante que poida distraer aos usuarios TEA.

Tamén destacar que se buscou desenvolver unha aplicación con certo grao de escalabilidade, o cal se logrou grazas ao uso do patrón *Business Logic Component* (BLoC), como se explicará no seguinte capítulo.

6.3 Casos de uso

Os casos de uso son as accións que poden realizar os actores e son habitualmente representados nun diagrama de casos de uso. Nas seguintes seccións, detallaremos os casos de uso para os diferentes actores da aplicación que se derivan dos requisitos funcionais identificados na fase de análise.

6.3.1 Usuario non autenticado

Os casos de uso do usuario non autenticado son os seguintes:

1. **Rexistrarse:** O actor ten unha vista dispoñible para poder rexistrarse na aplicación indicando o seu nome, correo electrónico, número de teléfono, unha foto de perfil, un contrasinal, e se o usuario será o responsable de nenos e nenas con TEA ou un profesional.
2. **Iniciar sesión:** Este actor pode iniciar sesión no sistema se xa tiña unha conta creada previamente, indicando o seu correo electrónico e o contrasinal. Devolve excepcións no caso de que algún destes valores sexa erróneo.
3. **Recuperar contrasinal:** No caso de que non recorde o seu contrasinal, pode indicar o seu correo electrónico e enviaráselle unha mensaxe cunha ligazón para poder cambiar o contrasinal.

Na figura 6.1 (páxina 34), pode verse o diagrama de casos de uso para este actor.

6.3.2 Responsable

Os casos de uso dun usuario responsable son os seguintes:

1. **Ver usuarios TEA:** Poderá ver o listado dos usuarios creados por el.

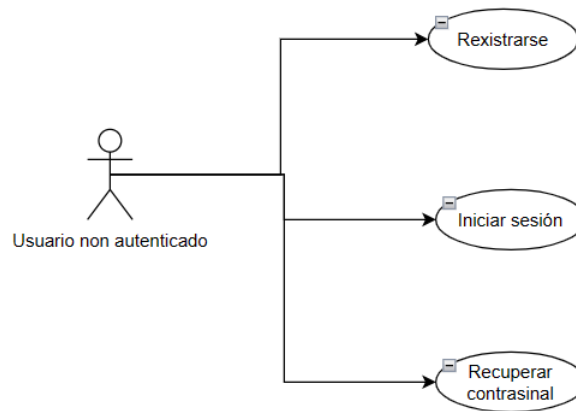


Figura 6.1: Diagrama dos casos de uso do usuario non autenticado.

2. Crear usuario **TEA**: Os responsables son os encargados de crear aos nenos e nenas que usarán a aplicación, indicando o seu nome e unha foto de perfil.
3. Eliminar usuario **TEA**: Tamén poden eliminar calquera usuario creado por eles mesmos.
4. Asignar profesional a usuario: Poden buscar na base de datos global aos profesionais para asignarllos ao neno ou nena e que estes tamén teñan acceso ao seu perfil, e a colección de categorías e pictogramas.
5. Desasignar profesional a usuario: Do mesmo xeito que poden asignar profesionais, tamén poden quitarlles o acceso.
6. Editar usuario **TEA**: Teñen dispoñible a opción de editar os usuarios dos que son responsables, podendo modificar o seu nome, a foto de perfil e a cor de fondo que verá o neno ou nena na aplicación.
7. Ver perfil: Poden ver os seus propios datos.
8. Editar perfil: Poden modificar o seu propio perfil, cambiando a súa foto de perfil, o nome, o correo electrónico, o número de teléfono ou o contrasinal.
9. Seleccionar usuario **TEA**: Poden seleccionar o usuario e serán conducidos ao perfil deste.
10. Ver perfil usuario **TEA**: Poden ver os datos do neno.
11. Cerrar sesión: Poden desconectarse do seu usuario.

Pódense ver os casos de uso mencionados previamente no diagrama da figura 6.2 (páxina 35).

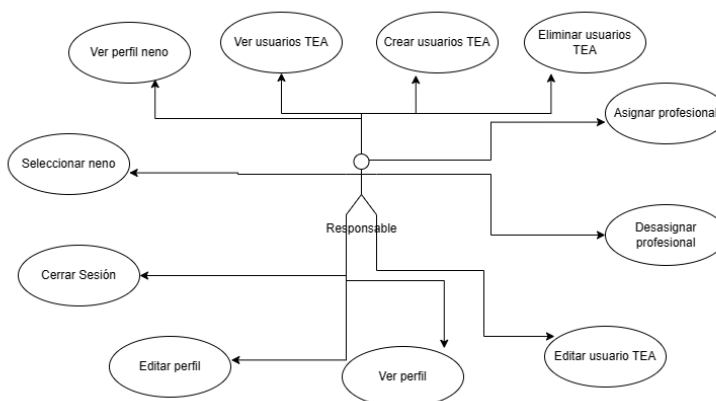


Figura 6.2: Diagrama de casos de uso do responsable.

6.3.3 Profesional

O profesional terá os seguintes casos de uso:

1. Ver categorías: Poden ver o listado de categorías creadas para cada neno.
2. Crear categoría: Os profesionais poden crear categorías indicando o seu nome e unha imaxe.
3. Eliminar categoría: Tamén poden eliminar as categorías.
4. Ver pictogramas: Poden ver os pictogramas engadidos en cada categoría.
5. Crear pictograma: Poden crear pictogramas indicando o nome que terá asociado e unha imaxe, que poderán obter da súa propia galería, sacando unha foto no momento ou da [API](#) de pictogramas de [ARASAAC](#).
6. Eliminar pictograma: Tamén poden eliminar os pictogramas creados.
7. Desactivar pictograma: Poden desactivar os pictogramas que consideran que non van ser necesarios para o usuario [TEA](#) e que, como consecuencia, deixarán de estar dispoñibles (visibles) para o neno.

8. Activar pictograma: Do mesmo xeito que poden desactivar un pictograma, tamén poden volver a activalo.
9. Ver perfil: Poden ver os seus propios datos.
10. Editar perfil: Poden modificar o seu propio perfil, cambiando a foto de perfil, o nome, o correo electrónico, o número de teléfono ou o contrasinal.
11. Ver usuarios TEA: Poderá ver o listado dos usuarios aos cales foi asignado polo responsable.
12. Seleccionar usuario TEA: Poden seleccionar o usuario e serán conducidos ao perfil deste.
13. Ver perfil usuario TEA: Aínda que non poidan modificalos, si poden ver os datos do neno.
14. Cerrar sesión: Poden desconectarse do seu usuario.

Chegados a este punto, é importante destacar que os oito primeiros casos de uso relacionados coa xestión de categorías e pictogramas están tamén dispoñibles para o usuario responsable.

Pódense ver estes casos de uso do usuario profesional na figura 6.3 (páxina 36).

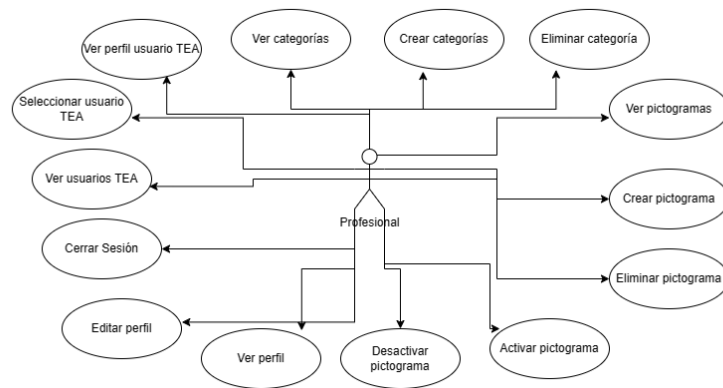


Figura 6.3: Diagrama dos casos de uso do profesional.

6.3.4 Usuario TEA

Os casos de uso do usuario TEA serán os seguintes:

1. Ver categorías: Os usuarios **TEA** poden ver todas as categorías creadas para eles.
2. Ver pictogramas: Tamén poden ver o listado de pictogramas que o responsable/profesional activou en cada categoría.
3. Seleccionar pictogramas: Poden interactuar con estes pictogramas e, no caso de que os pulsen, reproducirá o nome do pictograma mediante un *feedback* de audio.
4. Crear frases: Teñen dispoñible a posibilidade de crear frases de 3 pictogramas na versión vertical de móbil, ao ser máis reducida, e de 5 na versión horizontal. Ademais, no caso de equivocarse, poderán eliminar toda a frase creada, eliminar o último pictograma engadido á frase e reproducir o audio completo da frase que acaban de construír.

O diagrama cos casos de uso dispoñibles para o usuario **TEA** pode verse na imaxe 6.4 (páxina 37).

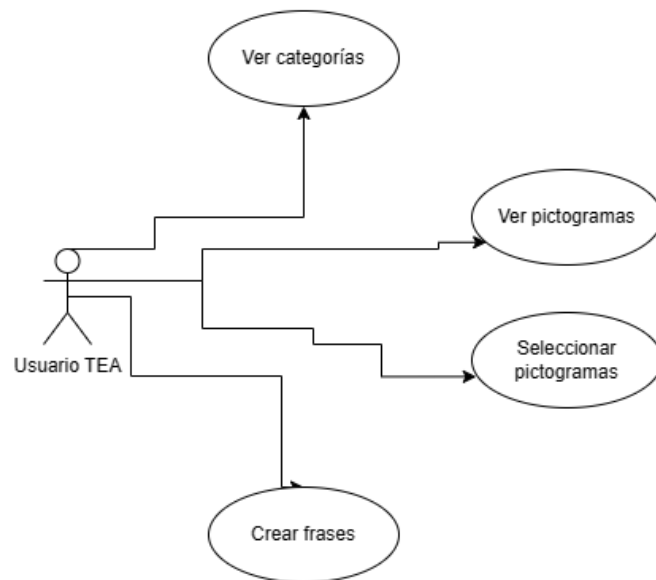


Figura 6.4: Diagrama de casos de uso para o usuario **TEA**.

Capítulo 7

Deseño

Na fase de desenvolvemento dun produto software, despois da análise, realízase o deseño da aplicación. Nesta etapa, establécese un patrón de deseño que se debe seguir na implementación do código, a organización da base de datos e deséñanse uns bosquejos iniciais para as interfaces da aplicación.

7.1 Patrón de deseño

Os patróns de deseño son técnicas usadas para resolver problemas comúns no desenvolvemento do código. Neste proxecto, usouse o patrón **BLoC** que foi creado polo equipo de Google co obxectivo de reutilizar o código nas aplicacións móbiles, usando Flutter con Dart [40]. **BLoC** baséase na programación reactiva usando o patrón *observer*, pero é máis simple e versátil. Este patrón baséase nun compoñente intermediario entre as vistas e o modelo que se encarga de xestionar os cambios de estado na aplicación.

O patrón **BLoC** ten varios obxectivos:

- Centralizar a lóxica de negocio: A idea disto é extraer todo o que non sexa renderización da vista nunhas clases chamadas **BLoC**, alleas a calquera librería, xa que non teñen ningunha dependencia de librerías concretas e só dependen de abstraccións. Desta forma, usando o principio de inversión de dependencia, unha clase **BLoC** pode escalar mellor a futuros cambios ou a usarse noutras tecnoloxías.
- Centralizar os cambios de estado: As clases **BLoC** serán as encargadas de recibir os eventos que se producen na aplicación, e que modifican o estado, e de notificar aos compoñentes correspondentes da aplicación deses cambios de estado.
- *Mapear* os datos ao formato que necesita a vista: Os elementos **BLoC** tamén se encargan de adaptar os datos do modelo ao formato que necesitan as vistas.

Este patrón está baseado nun fluxo de datos unidireccional, como podemos observar na imaxe 7.1. Para modificar o estado, os eventos do *widget* na vista invocan as accións do BLoC, que despois modifica o seu estado en función da resposta que se produce a ese evento.

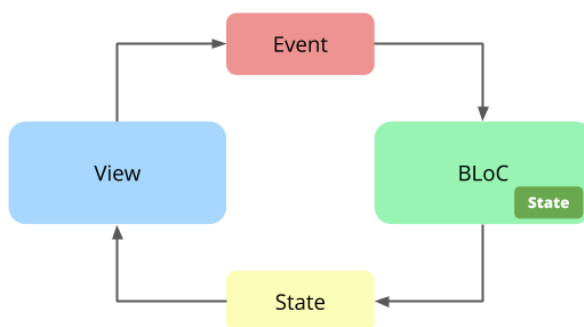


Figura 7.1: Fluxo unidireccional do patrón BLoC.

O patrón BLoC conta cos compoñentes que se poden observar na figura 7.2. Estes elementos son os seguintes:

- Vistas: Contén todas as interaccións coas vistas.
- BLoC: Aquí estará a capa de negocio.
- Repositorio: Neste compoñente concentraranse as clases que conectan coas APIs e bases de datos.
- Modelo: Representan os modelos de datos.

7.2 Deseño da base de datos

O deseño da base de datos consiste na organización dos datos da aplicación de acordo cun modelo de datos. Nesta sección, explicarase a forma na que os datos son almacenados, organizados e manipulados.

7.2.1 Firestore Database

Como xa se mencionou en capítulos anteriores, Firestore é unha base de datos NoSQL que foi elixida para almacenar os datos dos usuarios e pictogramas. Esta ferramenta está baseada na creación de coleccións que, así mesmo, conterán documentos cos datos almacenados.

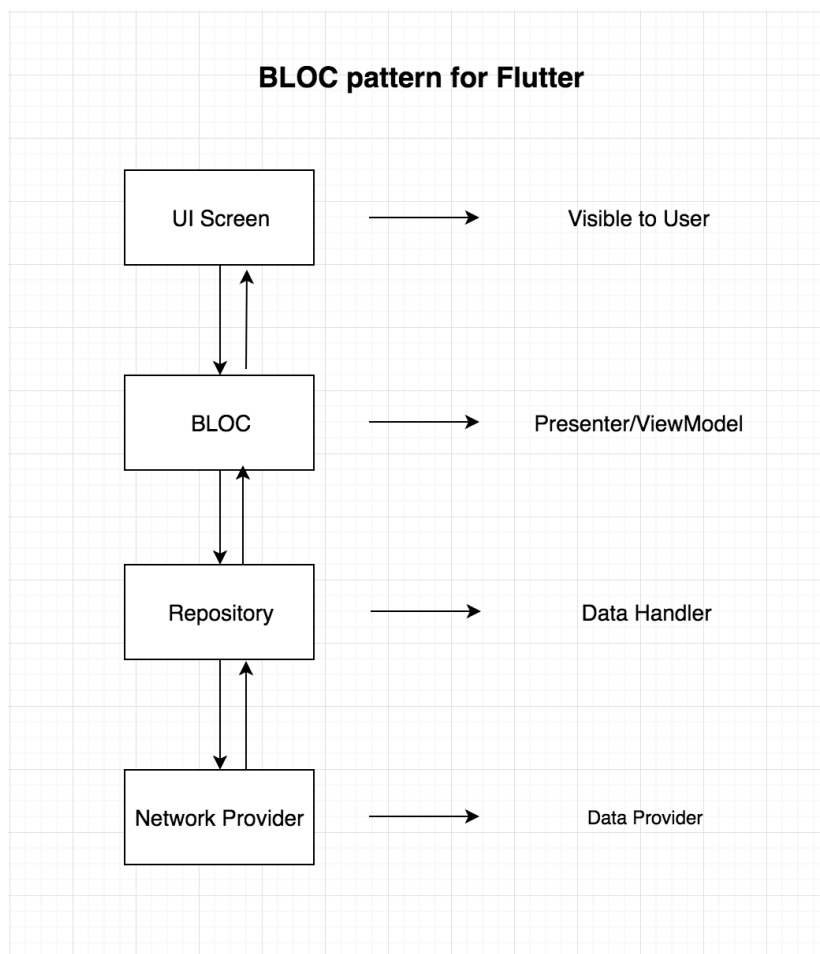


Figura 7.2: Arquitectura do patrón BLoC.

Neste proxecto, dispónse da colección “admins”, na que se encontran todos os responsables e profesionais, identificados polo seus respectivos identificadores de usuario. En cada documento, almacénase o identificador do usuario, un correo electrónico, un nome, un booleano que indica se é responsable ou non, un número de teléfono e a referencia á foto de perfil do usuario. Na táboa 7.1 (páxina 41), pódense ver os campos desta colección, co seu tipo e unha breve descrición.

Dentro da colección de admins, encóntrase a subcolección dos usuarios TEA. Nesta subcolección, gárdanse os valores correspondentes ao nome do neno ou nena, a cor de fondo do seu perfil, a referencia á imaxe de perfil e o identificador do usuario. Os diferentes campos desta subcolección pódense observar na táboa 7.2 (páxina 41).

Despois, foi creada unha colección para as categorías, cuxos documentos son identificados

Admins (Responsables/Profesionais)		
Dato	Tipo	Descrición
email	string	Email de contacto
isSuperadmin	booleano	Indica se é responsable
name	string	Nome do usuario
phoneNumber	string	Teléfono do usuario
profilePicture	string	Referencia á foto de perfil
uid	string	Valor identificativo do usuario

Táboa 7.1: Táboa coa organización da colección “admins”.

Usuarios TEA		
Dato	Tipo	Descrición
color	número	Cor de fondo para o neno
name	string	Nome do neno
profilePicture	string	Referencia á foto de perfil deste usuario
uid	string	Valor identificativo do usuario

Táboa 7.2: Táboa coa organización da subcolección “teaUsers”.

polo identificador do usuario **TEA**. Nesta colección encóntanse as categorías que, á súa vez, dentro conteñen unha subcolección de pictogramas que pertencen a cada categoría creada. A colección de categorías inclúe os datos do nome da categoría e a referencia á súa foto ou imaxe identificativa. A subcolección de pictogramas inclúe o nome do pictograma, un valor que representa se estará activo ou inactivo para o neno ou nena, unha imaxe e o seu identificador. Nas táboas 7.3 e 7.4, pódese ver a representación destes datos correspondentes ás categorías e pictogramas, respectivamente.

Categorías		
Dato	Tipo	Descrición
name	string	Nome identificativo da categoría
photo	string	Referencia á foto da categoría

Táboa 7.3: Táboa coa organización da subcolección “categories”.

Pictogramas		
Dato	Tipo	Descrición
active	booleano	Indica se o pictograma será visible para o neno
name	string	Nome do pictograma
photo	string	Referencia á foto deste pictograma
id	string	Valor identificativo do pictograma

Táboa 7.4: Táboa coa organización da subcolección “pictos”.

7.2.2 Firebase Storage

Storage é o servizo de almacenamento de Firebase onde se gardarán os arquivos xerados polos usuarios da aplicación. Nesta aplicación, almacénanse os arquivos da seguinte forma:

- Con respecto ás fotos de perfil dos distintos usuarios, gárdanse no cartafol *profilePictures*, de tal forma que, para acceder á imaxe dun determinado usuario, accederíase da seguinte forma:

profilePictures/UID_USUARIO.

- Para as imaxes das categorías e dos pictogramas, cada categoría é gardada baixo o identificador do usuario **TEA**, e, á súa vez, os pictogramas están gardados dentro das catego-

rías. Polo tanto, se se está intentando obter a imaxe dunha categoría, farase da seguinte maneira:

```
UID_USUARIO/NOME_CATEGORÍA.
```

Pola contra, se se intenta obter a imaxe do pictograma, sería así:

```
UID_USUARIO/NOME_CATEGORÍA/ID_PICTOGRAMA.
```

7.3 Deseño das interface do usuario

En canto ao deseño das interfaces desta aplicación, intentouse manter un deseño sinxelo, claro e minimalista debido aos usuarios aos que vai dirixida a aplicación. Esta céntrase no uso dos pictogramas sen engadir nada adicional que poida chegar a ser unha distracción para os nenos e nenas que a usarán. As partes da aplicación que non usarán os usuarios con TEA, si que xa inclúen máis elementos, pero aínda así, o obxectivo é que sexa simple e fácil de usar.

Para as vistas do móbil, usouse Material [41], que é unha guía de deseño definida por Google que busca axudar na creación de aplicacións. En Flutter, existen unha serie de *widgets* predefinidos que seguen as directrices de deseño de Material e que ademais contan cun gran nivel de personalización e comportamento. Algúns exemplos de *widgets* pertencentes a Material que foron usados no desenvolvemento do proxecto poderían ser os botóns *ElevatedButton* ou *IconButton*, as tarxetas ou os *TextField*.

Con respecto ás vistas da *tablet*, xorde o problema de que hai demasiado espazo, algo que nas vistas do móbil era todo o contrario, polo que o código necesita actualizarse a estas pantallas de maior tamaño. Para isto, decidiuse usar a plantilla *Master Detail* [42], que permite dividir a pantalla para que non quede tanto espazo en branco. Para exemplificar o uso deste patrón, na imaxe 7.3 (páxina 44), pódese ver a implementación dunha lista de datos. No móbil, esta lista queda ben debido a que a pantalla non é demasiado ancha pero, se se intenta aplicar o mesmo deseño nunha *tablet*, sobraría moito espazo pola dereita. Na imaxe 7.4 (páxina 44), obsérvase como coa aplicación do patrón, teríase a lista no lado esquerdo e os datos de cada ítem no dereito, aproveitando o espazo e non deixando tanto oco en branco.

7.3.1 Deseño inicial

Aínda que non son deseños totalmente fieis, porque durante o desenvolvemento da aplicación houbo cambios nas ideas iniciais do proxecto, nesta sección presentaranse os bosquejos iniciais realizados antes da implementación da aplicación.



Figura 7.3: Pantallas do móbil.

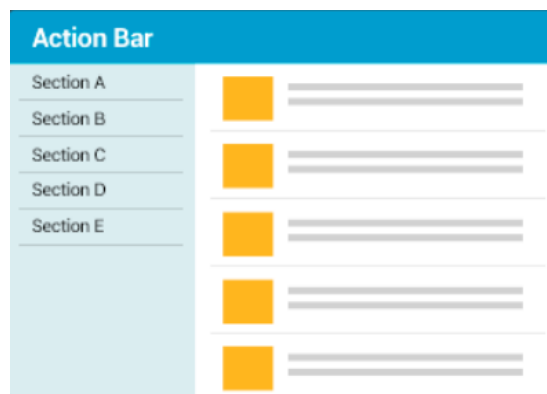


Figura 7.4: Pantallas da tablet usando o Master Detail.

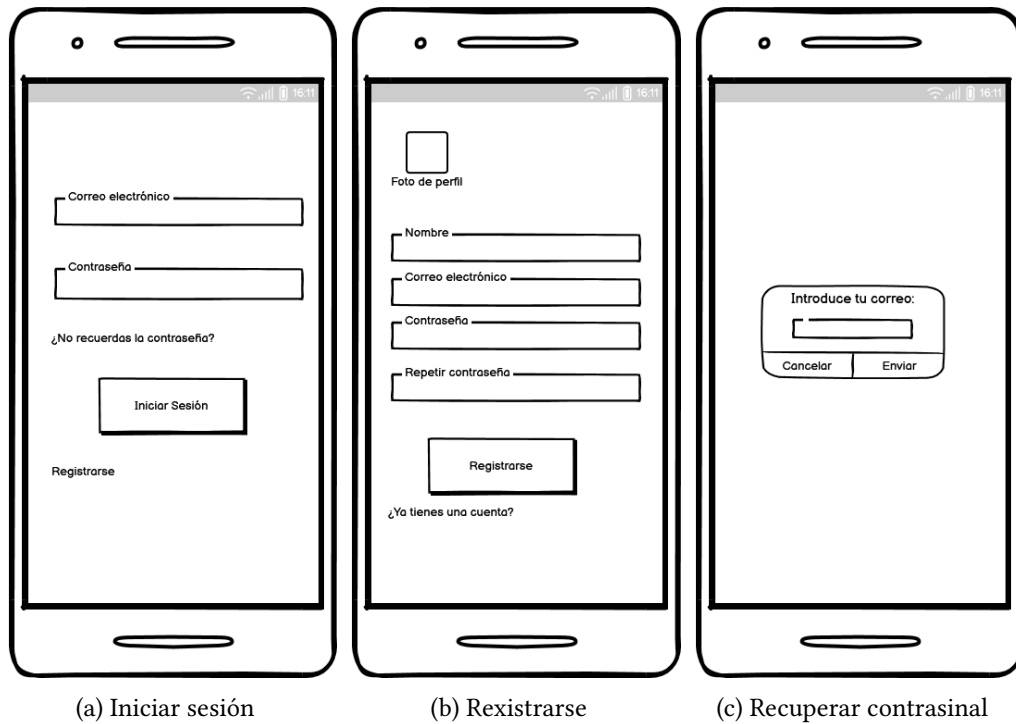


Figura 7.5: Pantallas de autenticación da aplicación.

Pantallas iniciais

Nos primeiros *mockups*, nas imaxes da figura 7.5 (páxina 45), mostráanse as pantallas de autenticación e rexistro dos usuarios. Estas pantallas non sufriron moitas modificacións na implementación final. A que máis cambios tivo foi a de rexistrarse, xa que se engadiron novos campos, como foron o do número de teléfono e un *check* para indicar se o usuario que se está a rexistrar é responsable ou non.

Listado de usuarios

Nas imaxes da figura 7.6 (páxina 46), pódense ver os listados de usuarios TEA e de profesionais. A pantalla de usuarios TEA non sufriu ningún cambio relevante, pero a de profesionais non chegou a implementarse. Isto foi debido a que a idea inicial era que o responsable crease os profesionais para despois asignarllos aos distintos nenos que tivera ao seu cargo, pero isto descartouse máis adiante debido a que non resultaba demasiado práctico e decidiuse cambiar para facer que o profesional se poidera rexistrar da mesma forma que o responsable. Para asignalos logo aos nenos, simplemente se buscarán entre os usuarios rexistrados na base de datos. Na figura 7.7 (páxina 47), tamén se pode ver esta pantalla na súa versión para *tablet*, onde se pode apreciar a aplicación do patrón *Master Detail*.

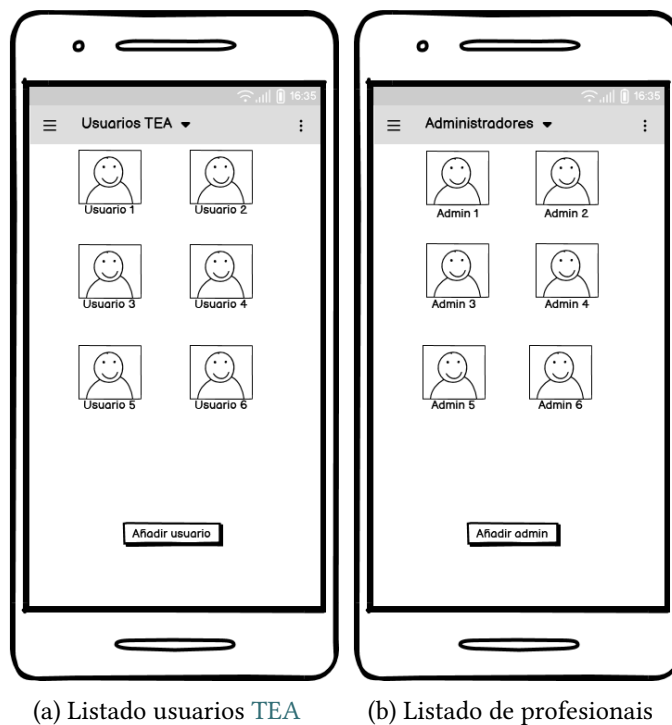


Figura 7.6: Listados de usuarios.

Listado de categorías e pictogramas

Nas imaxes da figura 7.8 (páxina 48), obsérvanse as pantallas correspondentes ao listado de pictogramas e de categorías, tal e como as verían os responsables e profesionais, é dicir, cos botóns de configuracións coas diferentes opción para personalizalos. Estas interfaces non sufriron ningún cambio relevante. Nas vistas do neno, na versión da *tablet*, tomouse a decisión de que neste caso non se aplicaría o patrón *Master Detail* para reducir ao máximo posible as distraccións que poidan chegar a ter. O importante é que simplemente teñen acceso ás categorías, pictogramas e opcións para construír as súas frases.

Configuración do usuario

Na figura 7.9 (páxina 48), encóntrase a pantalla que permite a actualización do perfil de usuario. A esta pantalla, ao igual que a de rexistrarse, tamén lle foi engadido posteriormente o campo do número de teléfono. A imaxe 7.10 (páxina 49) mostra esta mesma pantalla pero no caso da *tablet*, aplicando de novo o deseño baseado en *Master Detail*.

Editar usuarios TEA

Na figura 7.11 (páxina 49), pode verse a pantalla onde se mostra a información do perfil do neno, incluíndo a súa foto de perfil, o seu nome e a cor de fondo que se seleccionou para el.

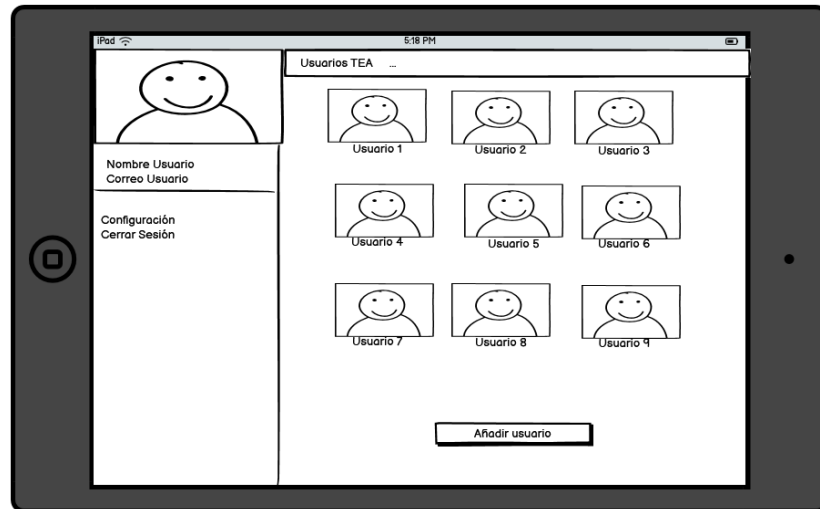


Figura 7.7: Listado de usuarios TEA na tablet.

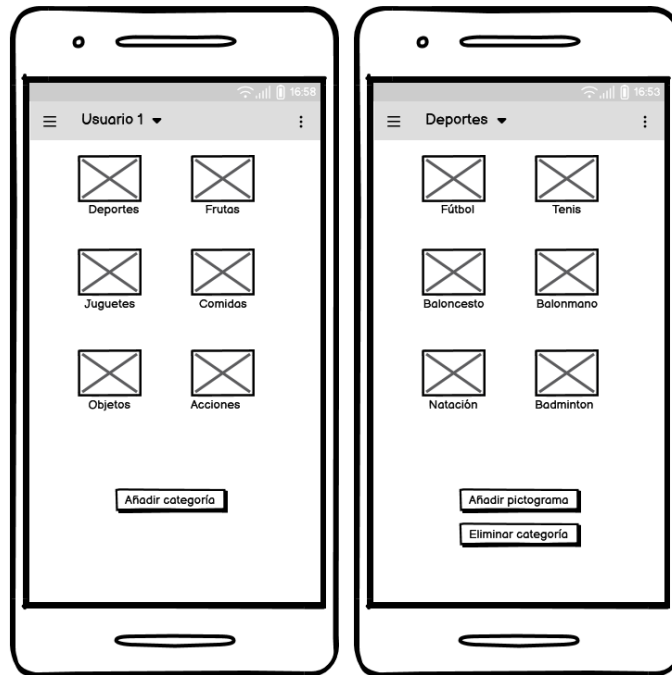
Inicialmente, como xa se explicou previamente, o responsable tiña que engadir os profesionais e despois asignalos ao neno, polo que tamén se incluía unha lista cos profesionais creados por ese responsable para poder marcar cales terían acceso ao neno.

Engadir usuarios, pictogramas, e categorías

Nas imaxes da figura 7.12 (páxina 50), poden verse as interfaces pensadas para a creación de categorías, de pictogramas e de usuarios TEA. Na vista dos pictogramas, débese especificar o nome do pictograma e a imaxe correspondente, que pode ser seleccionada tanto sacando unha foto, como seleccionándoa da galería ou buscándoa a través da API. A pantalla de categorías ten o mesmo funcionamento, só que non permite buscar na API a imaxe representativa da mesma. Para os usuarios, debíase especificar o nome, unha imaxe de perfil e os responsables que estarían ao seu cargo. Como finalmente se desbotou a idea da lista de profesionais, esta pantalla tamén sufriu cambios significativos.

Vistas do usuario TEA

Na figura 7.13 (páxina 50), represéntase o que verán os nenos cando interactúen coa aplicación. Como se pode apreciar, terán dispoñible un campo de creación de frases na parte superior, con botóns para reproducir a frase completa, borrarla completamente ou borrar simplemente o último pictograma engadido.



(a) Listado de categorías

(b) Listado de pictogramas

Figura 7.8: Listado de pictogramas e categorías.

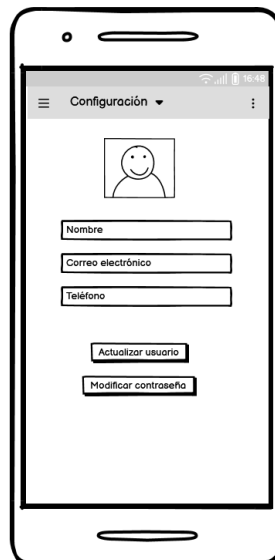


Figura 7.9: Configuración de usuario.

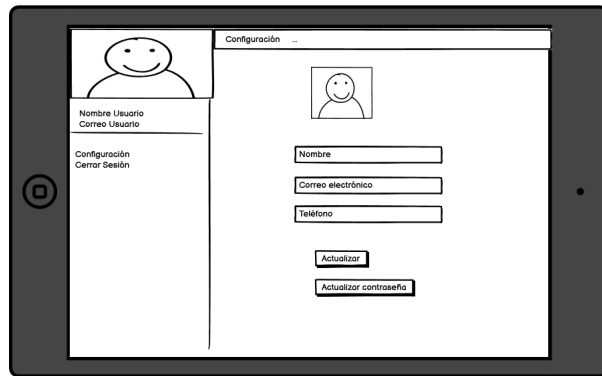
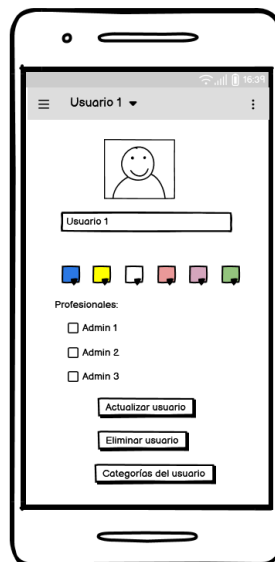
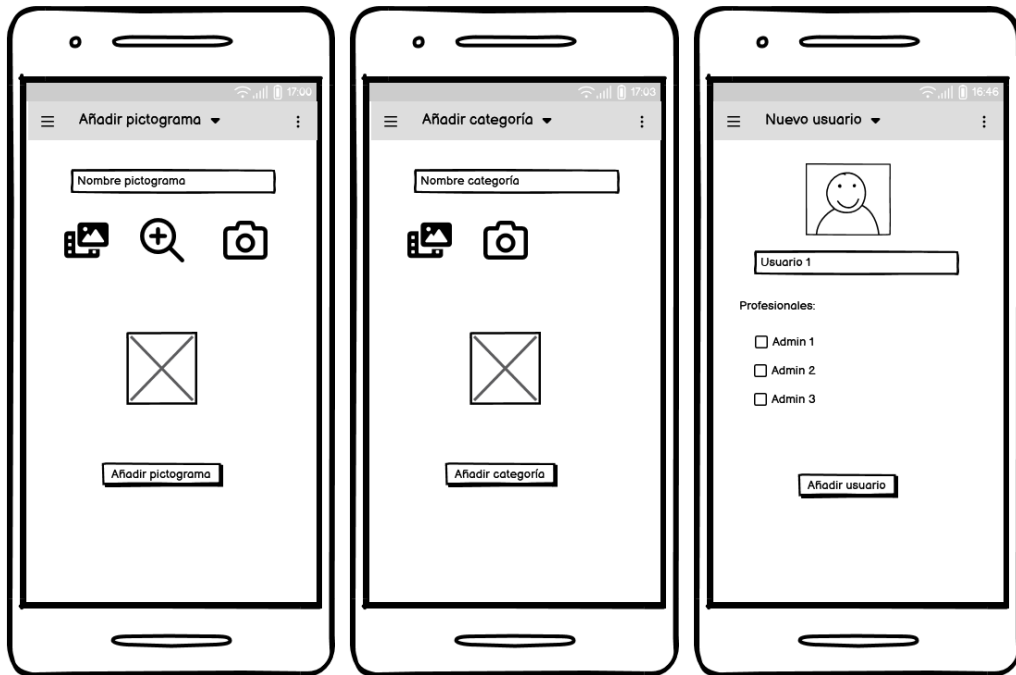
Figura 7.10: Configuración do usuario na *tablet*.

Figura 7.11: Editar e eliminar o perfil do neno.



(a) Creación de pictogramas (b) Creación de categorías (c) Creación de usuarios TEA

Figura 7.12: Pantallas de creación dos elementos da aplicación.

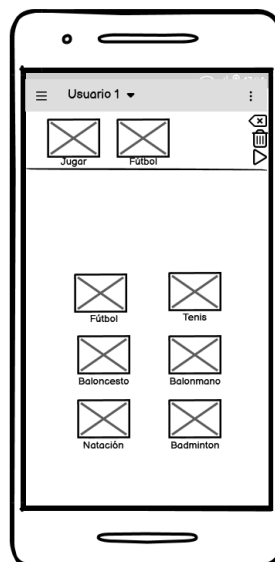


Figura 7.13: Vista da creación de frases dos usuarios TEA.

Implementación

Neste capítulo, explicarase a estrutura do proxecto e as partes máis relevantes do código, tanto do *backend* como do *frontend*. Tamén se amosará o produto final que se obtivo.

8.1 Estrutura

O proxecto segue o patrón de deseño **BLoC**, polo que está organizado como este especifica, contendo un cartafol chamado *repository* que é onde se farán as chamadas a **API** de **ARASAAC** e a **Firestore**, o cartafol *bloc*, que contén os estados, eventos e o propio compoñente **BLoC**, o directorio *models*, no que se encontran os modelos de datos usados na aplicación e *views*, que é onde están as interfaces das pantallas.

8.2 Backend

Para a implementación do *backend* da aplicación, tiveron que usarse ferramentas externas a Flutter, como foron **Firestore** ou a **API** de **ARASAAC**. Por esta razón, gran parte da aplicación tívose que facer asíncrona, usando *FutureBuilder*, para poder esperar a obter os datos da nube, antes de cargar os compoñentes.

8.2.1 Firebase Authentication

Firebase Authentication foi usado para a autenticación dos usuarios da aplicación, permitindo que se rexistren, inicien sesión e poidan recuperar o contrasinal. Para poder empregar esta ferramenta, fíxose uso da librería *firebase_auth.dart*. Algún exemplo do uso desta librería podería ser o inicio de sesión, como se observa no seguinte fragmento de código:

```
1 Future<void> signIn({  
2   required String email,
```

```
3     required String password,
4   }) async {
5     try {
6       await FirebaseAuth.instance
7         .signInWithEmailAndPassword(email: email, password:
password);
8     } on FirebaseAuthException catch (e) {
9       if (e.code == 'user-not-found') {
10        throw ('No se ha encontrado ningún usuario con el
correspondiente correo electrónico.');
```

8.2.2 Firestore Database

Esta ferramenta foi necesaria para almacenar os datos dos usuarios, e das categorías e pictogramas que estes teñan. Para acceder ás funcionalidades que proporciona desde o código, simplemente se necesita incorporar a librería *cloud_firestore.dart*. Algún exemplo do uso desta librería podería encontrarse na obtención dos datos dalgún usuario, tal e como se mostra nas seguintes liñas de código:

```
1 Future<Admin> getAdminData({required String admin}) async {
2   DocumentSnapshot<Map<String, dynamic>> snapshot =
3     await
4     _firebaseFirestore.collection('admins').doc(admin).get();
5   return Admin.fromSnapshot(snapshot);
6 }
```

8.2.3 Firebase Storage

Mediante o uso da librería *firebase_storage.dart*, almacénanse as imaxes de perfil, e das categorías e pictogramas dos usuarios en Storage da seguinte maneira:

```
1   await _firebaseStorage
2     .ref()
3     .child('profilePictures/' + admin)
4     .putFile(file);
```

8.2.4 ARASAAC

ARASAAC ofrece moitas opcións para acceder aos seus pictogramas desde as aplicacións, nas figuras 8.1 (páxina 54) e 8.2 (páxina 54) encóntranse os que serán usados neste proxecto. Para poder obter a imaxe dun pictograma, habería que usar unha petición GET a `/pictograms/idPictogram`. Sen embargo, para poder facer isto, hai que obter primeiro o identificador do pictograma que se está buscando. Este identificador pode obterse mediante outra petición GET pasándolle como parámetro a `/pictograms/{language}/search/{searchText}` un texto de búsqueda que introducirá o usuario e a linguaxe, castelán neste caso. Con esta petición, obterase a resposta en formato JSON, para despois pasarlle o identificador que ven nese JSON á outra petición para obter a imaxe. Este proceso pode observarse nas seguintes liñas de código:

```
1 Future<List<dynamic>?> getPictograms(String locale, String keyword)
  async {
2   String aux = "";
3   if (keyword.contains(" ")) {
4     var keywords = keyword.split(" ");
5     for (var k in keywords) {
6       if (k != "") {
7         aux += k + " ";
8       }
9     }
10    aux = aux.substring(0, aux.length - 1);
11    keyword = "\"" + aux + "\"";
12  }
13
14  var url = Uri.https(
15    "api.arasaac.org", "api/pictograms/" + locale + "/search/" +
16    keyword);
17
18  var response = await http.get(url);
19
20  List<dynamic> responseBody = jsonDecode(response.body);
21
22  if (response.statusCode == 200) {
23    return responseBody;
24  } else {
25    return null;
26  }
27
28 Future<String> getPictogramImage(int id) async {
29   var urlImage =
30     Uri.https("api.arasaac.org", "api/pictograms/" +
```

```

    id.toString());
31 var imageResponse = await http.getImage(urlImage);
32
33 final bytes = imageResponse.bodyBytes;
34
35 var image = base64.encode(bytes);
36
37 return image;
38 }

```

GET `/pictograms/{idPictogram}`

Obtain pictogram image file in png format based on its key. Pictogram can be customized with color, resolution, plural...

Parameters

Name	Description
idPictogram * required integer (path)	Pictogram Identifier

Figura 8.1: API para a obtención dos pictogramas.

GET `/pictograms/{language}/search/{searchText}`

Obtain a list of pictograms based on the searchText. The pictogram url can be generated based on pictogram id value and the desired resolution (300px, 500px or 2500px). There are some variations available for each pictogram using filename parameters (see below). You can see them in use and some more (available using API endpoint `GET /pictograms/{idPictogram}`) at <https://arasaac.org/pictograms/en/2349>

Filename parameters

action for verbal tense (*past, future*)
hair for hair color (brown A65E28, blonde FDD700, red ED4120, black 020100, gray EFEFEF, darkGray AAABAB, darkBrown 6A2703)
skin for skin color (white F5E5DE, black A65C17, assian F4ECAD, mulatto E3AB72, aztec CF9D7C)
plural
nocolor

Examples

https://static.arasaac.org/pictograms/2349/2349_2500.png
https://static.arasaac.org/pictograms/2349/2349_action-past_300.png
https://static.arasaac.org/pictograms/2349/2349_nocolor_action-past_500.png
https://static.arasaac.org/pictograms/2349/2349_plural_nocolor_500.png
https://static.arasaac.org/pictograms/2349/2349_action-past_hair-FDD700_500.png
https://static.arasaac.org/pictograms/2349/2349_action-future_hair-020100_skin-A65C17_500.png

Parameters

Name	Description
language * required string (path)	Languages than we currently use for pictograms searches. <i>Available values</i> : an, ar, bg, br, ca, de, el, en, es, et, eu, fa, fr, gl, he, hr, hu, it, ko, lt, lv, mk, nl, pl, pt, ro, ru, sk, sq, sv, sr, val, uk, zh
searchText * required string (path)	Keywords for searches. By default, it makes an OR search on all the specified keywords. In case you would like to perform an exact phrase search (logical AND), you can do so by specifying double quotes in the search text. Prefixing a search keyword with - (minus sign) excludes all the documents that contain the negated term.

Figura 8.2: API para a obtención dos identificadores dos pictogramas.

8.2.5 Uso da galería ou cámara

Para poder obter imaxes da galería ou da cámara, usouse a librería *image_picker.dart*. No seguinte código, pode verse un exemplo de como se habilitaría a opción de escoller unha imaxe da galería ou de sacar unha fotografía coa cámara do dispositivo móbil:

```
1 Future getImageFromGallery() async {
2   try {
3     final image =
4       await ImagePicker.platform.getImage(source:
5         ImageSource.gallery);
6     if (image == null) return;
7     final imageTemp = File(image.path);
8     setState(() => this.image = imageTemp);
9   } on PlatformException catch (e) {
10    print('Failed to pick image: $e');
11  }
12 }
13 Future getImageFromCamera() async {
14   try {
15     final image =
16       await ImagePicker.platform.getImage(source:
17         ImageSource.camera);
18     if (image == null) return;
19     final imageTemp = File(image.path);
20     setState(() => this.image = imageTemp);
21   } on PlatformException catch (e) {
22    print('Failed to pick image: $e');
23  }
24 }
```

8.2.6 Reprodución dos pictogramas

Para poder reproducir o son dos pictogramas, tomouse a decisión de usar a librería *flutter_tts.dart*. Isto é un *plugin* que permite ler en voz alta o texto que se lle pasa, neste caso, o nome do pictograma a reproducir. Esta librería permite establecer a linguaxe, o volume ou a voz que se usará, entre outras configuracións. O seguinte código mostra como se pode configurar e empregar para reproducir o texto asociado aos pictogramas:

```
1 Future sayWord(String name) async {
2   await flutterTts.setSharedInstance(true);
```

```
3   await flutterTts
4
5   .setIosAudioCategory(IosTextToSpeechAudioCategory.playAndRecord,
6   [
7     IosTextToSpeechAudioCategoryOptions.allowBluetooth,
8     IosTextToSpeechAudioCategoryOptions.allowBluetoothA2DP,
9     IosTextToSpeechAudioCategoryOptions.mixWithOthers
10  ]);
11
12  await flutterTts.awaitSpeakCompletion(true);
13  await flutterTts.getVoices;
14  await flutterTts.setLanguage("es-ES");
15  await flutterTts.setSpeechRate(0.5);
16  await flutterTts.setVolume(1.0);
17  await flutterTts.setPitch(1);
18  await flutterTts.speak(name);
19  }
```

8.3 Frontend

Como se especificou na fase de deseño, o *frontend* da aplicación segue a guía creada por Google, Material. Empezouse usando a estrutura *Scaffold*, que conta cunha cabeceira, chamada *AppBar* que se coloca na parte superior da pantalla e que, no caso deste proxecto, indica onde se encontra o usuario en cada momento. Por exemplo, no caso de estar na categoría “Froitas”, na parte superior da pantalla aparecerá o nome indicándoo. Ademais, tamén se usou un *Drawer* que conta cos datos do usuario e dun índice no que se pode acceder ás outras partes da aplicación ou cerrar a sesión. Dentro do *Scaffold*, fóronse engadindo o resto de *wid-gets* que formaron a versión final da aplicación, como foi o *GridView.builder* para organizar os elementos nas listas de usuarios, de categorías e de pictogramas ou as *Cards*, para organizar os elementos cun *Inkwell* para acceder ás outras partes da aplicación.

8.4 Vistas finais

A continuación, presentarase o produto final da aplicación, coas vistas do móbil. As vistas finais da tablet, poden encontrarse no apéndice A.

8.4.1 Autenticación

A parte dos cambios mencionados, cando se presentaron os bosquexos iniciais, as pantallas de autenticación da figura 8.3 (páxina 57) non sufriron ningún outro cambio remarcable.

Na vista para Rexistrarse, usáronse varios *widgets* entre os que se encontran o *CheckBoxList-Tile* para comprobar se é responsable ou non e varios *TextFormField* para poder introducir os datos do usuario que se está a rexistrar. Este *widget*, dispón dunha opción para verificar se o texto introducido coincide con uns requisitos que especifica o desenvolvedor e devolve un mensaxe de error se non se cumpren. Por exemplo, usando a librería *email_validator.dart*, pode verificarse se o que se introduce ten un formato correcto para un correo electrónico. Para as imaxes, ao pulsar sobre o icono da cámara, ou sobre a imaxe xa cargada no caso de que se queira modificar, saltará un *AlertDialog* para indicar se se quere abrir a galería ou sacar unha foto coa cámara.

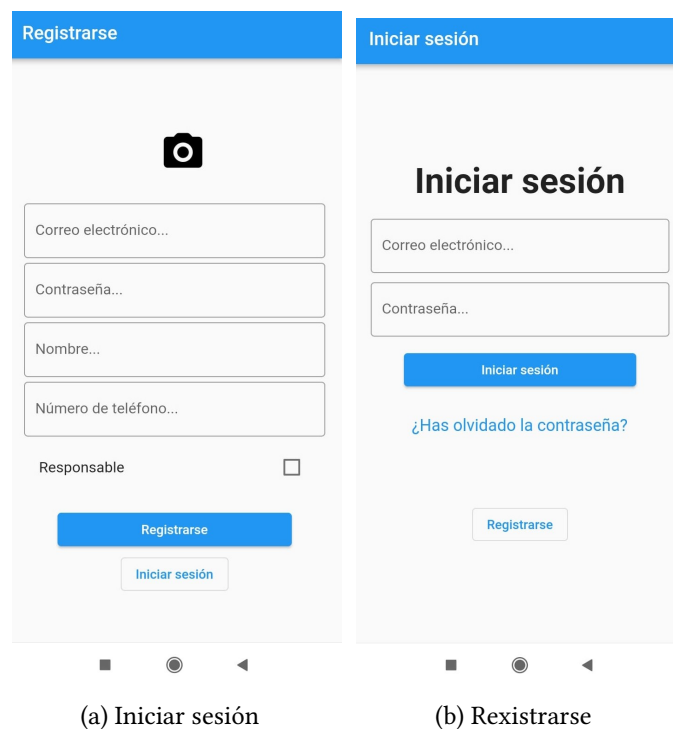


Figura 8.3: Pantallas de autenticación finais.

8.4.2 Listado de usuarios

Na figura 8.4 (páxina 58), pódese apreciar o listado dos nenos a cargo do usuario que iniciou sesión, no caso de ser un usuario rexistrado como responsable. Pulsar sobre a imaxe de calquera neno, levaríanos ao seu perfil e tamén se inclúe debaixo a opción para engadir máis usuarios TEA. No caso de ser un profesional, aparecería na pantalla principal, trala autenticación, unha lista similar cos nenos cos que pode traballar, é dicir, que lle foron “asignados” polo seu responsable correspondente.

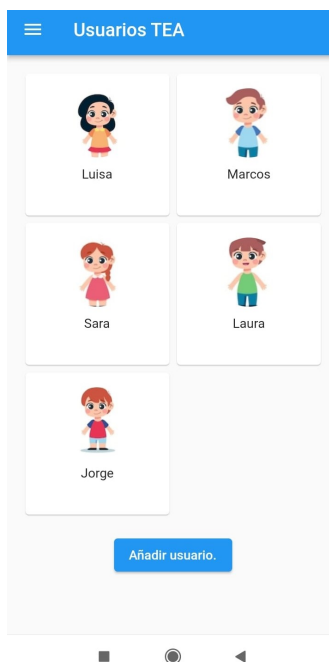


Figura 8.4: Versión final do listado de nenos.

8.4.3 Listado de categorías e pictogramas

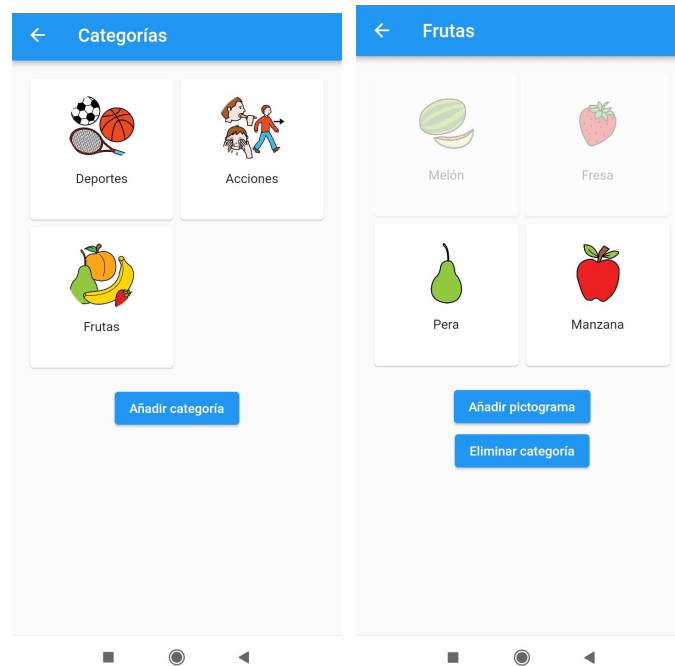
En canto ao listado das categorías, encóntranse as que foron engadidas para o usuario TEA seleccionado. Ao entrar en cada unha delas, están os pictogramas que se engadiron especificamente para esa categoría. No caso de que se atopen inactivos, móstranse máis opacos, como se pode apreciar nas imaxes da figura 8.5 (páxina 59). Nas vistas das categorías, hai un *ElevatedButton* que leva ao usuario a pantalla de creación de categorías e o mesmo cos pictogramas. Ao pulsar sobre os pictogramas, no modo de edición deles, ofrécense as opcións de eliminalo ou de desactivalo e activalo para o neno ou nena. Todos estes elementos foron engadidos nun *Card* dentro dun *GridView.builder*, que dependendo do tamaño da pantalla, mostra entre 2 ou 4 elementos por fila.

8.4.4 Configuración do usuario

A imaxe 8.6 (páxina 59) representa o perfil do usuario que iniciou sesión para que este poida modificalo no caso de que así o desexe.

8.4.5 Drawer

A aplicación tamén conta cun *drawer* na marxe esquerda, onde se pode ver o nome do usuario que iniciou sesión, a súa imaxe de perfil e o correo electrónico (ver figura 8.7). Ade-



(a) Listado de categorías disponibles para un usuario TEA (b) Listado de pictogramas dentro dunha categoría

Figura 8.5: Listados de categorías e pictogramas.

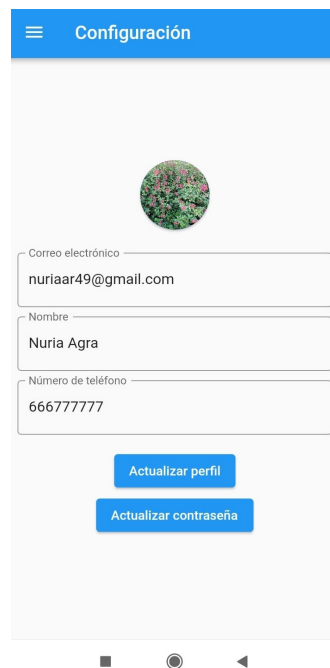


Figura 8.6: Configuración do usuario.

mais, este elemento permite acceder aos usuarios TEA, á configuración e pechar sesión.

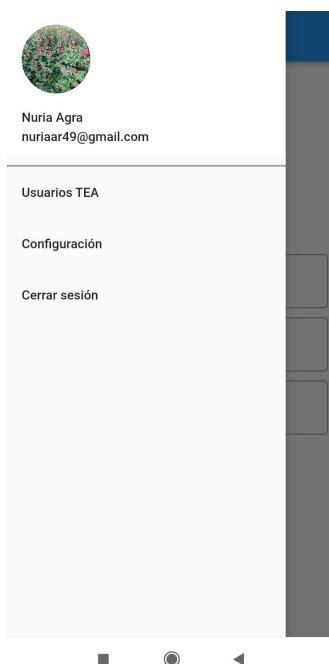


Figura 8.7: Drawer da aplicación.

8.4.6 Perfil dos usuarios TEA

Na figura 8.8 (páxina 61), pódese apreciar o perfil do neno desde as vistas do responsable, podendo editalo, eliminalo ou asignarlle un profesional. A cor de fondo que verá o neno tamén é personalizable e depende da elección do seu responsable. Esta opción é importante xa que as persoas con TEA poden ser altamente sensible a difentes estímulos, entre eles, determinados tipos de cores.

8.4.7 Engadir usuarios, pictogramas e categorías

As pantallas de creación de usuarios, pictogramas e categorías non sufriron ningún cambio importante e pódense ver nas imaxes da figura 8.9 (páxina 62). No caso da creación de usuarios, atópase un *TextFormField* para introducir o nome deste e o botón para imaxe co mesmo funcionamento explicado previamente para a pantalla de Rexistro. Tamén inclúe un botón para asignar os profesionais. Ao pulsar este botón, saltará un cadro de texto no cal se deberá introducir o correo electrónico do profesional ao que se lle queira asignar o neno que se creará. Se non se encontra un usuario con este correo na base de datos, saltará un aviso para o usuario, se se encontra mostrarase na pantalla con un botón ao lado para eliminalo en caso de erro. Para engadir os pictogramas, inclúense tres botóns e dependendo de cal se

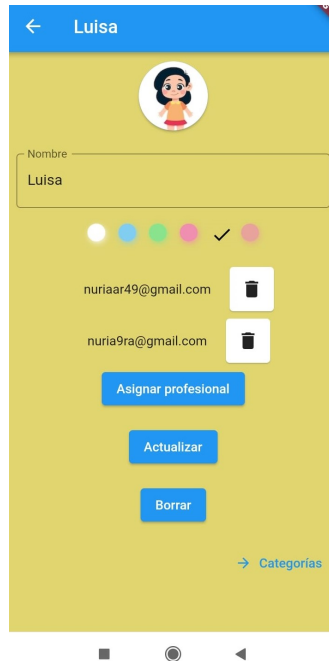
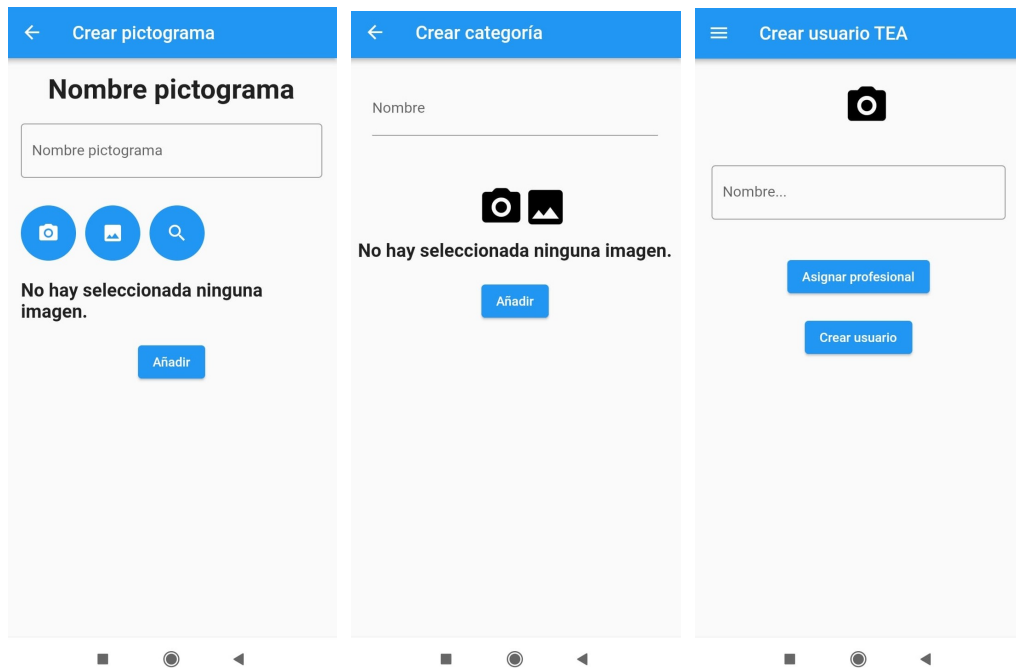


Figura 8.8: Vista final do perfil do neno.

pulse abre a galería, a cámara, ou un cadro de texto no cal se introducirá o pictograma que se quere buscar na [API de ARASAAC](#). Unha vez cargada esta imaxe, substituirá o texto que inicialmente indica que non hai seleccionada ningunha imaxe.

8.4.8 Vistas dos usuarios TEA

Nas vistas dos usuarios [TEA](#), a cor de fondo é a elixida polo responsable na súa configuración de perfil. Os pictogramas que pode ver o neno tamén dependen de se o responsable/profesional que iniciou sesión os activa ou non, e o tamaño da frase que se pode formar depende do tamaño da pantalla. No caso da imaxe [8.10](#) (páxina [62](#)), podería incluír ata 3 pictogramas, pero en horizontal e nas vistas da *tablet* poderían formarse frases con máis pictogramas. Nesta vista, téñese enriba un *Container*, inicialmente vacío, no cal se irán engadindo os pictogramas a medida que o neno os vai pulsando. Está separado dos botóns por un *Spacer*. O primeiro botón usaríase para eliminar toda a frase e que o neno poida volver a empezar, o segundo só borraría o último pictograma engadido no caso de que se confundira ao pulsalo, e o terceiro reproduciría toda a frase creada. Na imaxe, ao ser na versión vertical do móbil, só deixaría engadir ata tres pictogramas porque son os que collen. No caso de que o neno siga pulsando noutros, estes non se engadirían, aínda que o resto do funcionamento do pictograma como é que reproduza o seu nome ao pulsalo seguiría funcionando sen problema. Ademais, nesta pantalla, incluíuse que ao saír desa categoría e meterse noutra, os pictogramas da anterior



(a) Creación de pictogramas (b) Creación de categorías (c) Creación de usuarios TEA

Figura 8.9: Pantallas finais para a creación de elementos.

manteránse, permitindo crear frases máis variadas con pictogramas de distintas categorías.

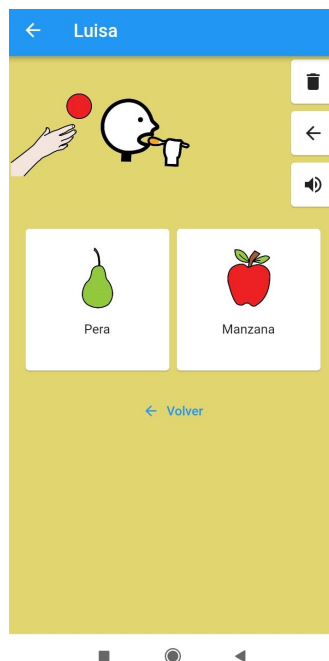
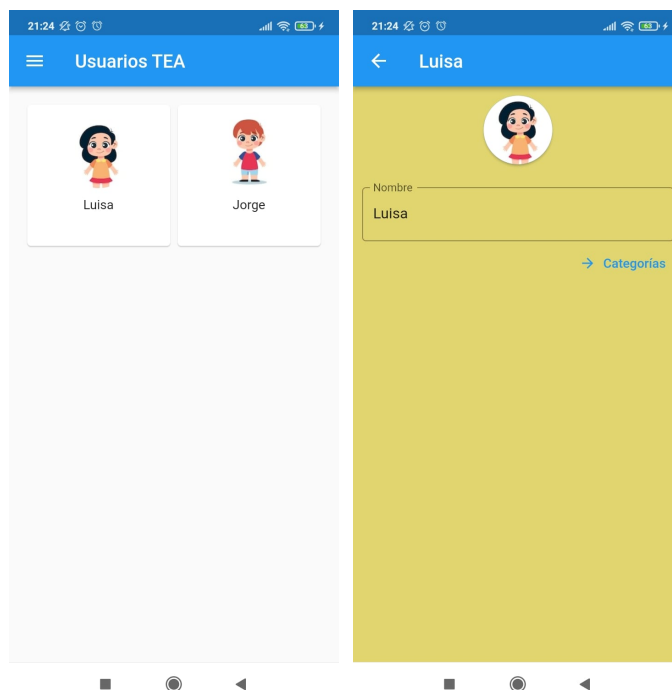


Figura 8.10: Vista final do perfil do neno.

8.4.9 Vistas do profesional

Previamente, indicouse que o profesional non podería engadir usuarios TEA e só tería dispoñibles para seleccionar os que lle asigne o responsable correspondente. Tampouco pode editar o perfil deses usuarios, aínda que si consultalo. Nas imaxes da figura 8.11 (páxina 63), obsérvase como se o usuario autenticado é un profesional, non tería acceso a esas funcionalidades.



(a) Listado de nenos do profesio- (b) Perfil do neno (sen posibili-
nal. dade de edición).

Figura 8.11: Vistas desde un usuario coa categoría de profesional.

8.5 Probas

Finalmente, para poder asegurarse do correcto funcionamento da aplicación, desenvólvense unha serie de probas.

8.5.1 Probas unitarias

Este tipo de probas permiten asegurarse do correcto funcionamento das partes máis pequenas da aplicación. A idea para este tipo de probas é asegurar que cada unidade funcione correctamente por separado. Estas probas realizáronse ao final de cada tarefa de cada *sprint*.

Para estas probas, utilizouse un móbil físico con Android e comprobouse que todas as interaccións do usuario con esa parte da aplicación eran correctas, funcionaban do xeito esperado e producían os resultados establecidos.

8.5.2 Probas de integración

Unha vez foron aprobadas as probas unitarias, realízanse as probas de integración, que se encargan de comprobar que todos os elementos unitarios do *software* funcionan en conxunto e obteñen o resultado que se espera. Estas probas foron realizadas ao final do ciclo de desenvolvemento da aplicación. Para estas probas, usouse o mesmo móbil físico Android, mencionado previamente, e ademais tamén se usou o emulador de Android Studio para poder obter unha *tablet* que permitiu probalo neste tipo de dispositivos.

8.5.3 Probas de aceptación

As probas de aceptación son as últimas probas que se realizan para comprobar se o produto cumpre cos requisitos e especificacións previamente establecidos. Estas probas adoitan ser realizadas polo equipo e posteriormente polos clientes. Neste caso, só foron realizadas nas reunións cos titores debido a falta de tempo. Posteriormente, tamén serán realizadas probas polas asociacións da contorna, para poder obter *feedback* por parte de profesionais que tratan con nenos con estas condicións máis a miúdo.

Aínda que non poideron verificar a versión final, si que se mantivo unha reunión para mostrarlles a aplicación nunha fase intermedia aos profesionais das asociacións da contorna, permitindo aplicar a metodoloxía ApS. Nesta reunión valoráronse as funcionalidades implementadas e ofrecéronse unha serie de melloras para a aplicación. Estas melloras foron posteriormente implementadas e entre elas encóntrase o tema de que os responsables non tiveran que crear os profesionais, se non que estes se rexistraran de forma normal, a personalización do usuario permitíndolles elixir entre varias cores de fondo ou a opción de reproducir toda a frase creada e non só os pictogramas pulsados.

Conclusiones

Finalmente, no derradeiro capítulo desta memoria, a modo de reflexión final, valórase o traballo realizado, os coñecementos adquiridos e as posibles melloras á aplicación para un futuro.

9.1 Conclusiones

Despois de finalizar todo o desenvolvemento, podemos concluír que se conseguiu obter unha aplicación funcional e que cumpre cos requisitos previamente establecidos. Esta aplicación dá a opción de agrupar por categorías, algo que xa tiñan todas as aplicacións estudadas durante a redacción desta memoria, pero tamén permite reproducir os pictogramas e crear frases e reproducilas, sendo estas funcionalidades importantes e que non estaba presentes en moitas das aplicacións máis populares actualmente. Ademais, tamén permite a creación de pictogramas usando os que ofrece [ARASAAC](#), unha vantaxe xa que os nenos e nenas con [TEA](#) adoitan estar familiarizados con eles.

Un traballo destas dimensións e características pode aportar moito coñecemento ou reforzar o aprendido durante a formación. Persoalmente, a realización deste proxecto serviu para afianzar os coñecementos que tiña sobre algunhas das ferramentas usadas, como foron Flutter e Dart, que foron tratadas brevemente durante a carreira. Tamén permitiu coñecer outra serie de ferramentas moi útiles, como son as distintas funcionalidades que ofrece Firebase para a autenticación e almacenamento de datos. Ademais, a posibilidade de realizar un proxecto deste tamaño, con todo o que iso conleva, en termos de realizar a análise e o deseño da aplicación, tamén afianza coñecementos adquiridos brevemente durante a formación. A aplicación dunha metodoloxías de traballo en contornas reais, como Scrum, deu a posibilidade de coñecer como sería levar a cabo un proxecto real.

Con respecto á aplicación do método de traballo [ApS](#), isto permitiu poder informarse sobre a situación das persoas con [TEA](#) e aprender máis sobre esta condición mentras que se

aplicaban os coñecementos para poder facer máis sinxela a súa vida. Aínda que, a adaptación das vistas e a aplicación para que fose algo sinxelo de usar para eles supuxo un reto debido a toda a investigación que se tivo que facer para atender as súas necesidades, a realización deste traballo foi moi satisfactoria para a estudante.

9.2 Liñas futuras

Aínda que se conseguiron os obxectivos establecidos no proxecto, propóñense unha serie de posibles melloras que se poderían facer nun futuro.

Algunhas destas melloras incluírían:

- Incorporar unha opción que permitiría que os responsables bloqueasen a edición de certas categorías para que os profesionais non poidan desactivar ou eliminar os seus pictogramas, o que aumentaría a seguridade da aplicación para o responsable.
- No momento de engadir os pictogramas de [ARASAAC](#), incluír a posibilidade de que o responsable ou profesional poida distinguir entre varios dos que ofrece a [API](#), dependendo do nivel de abstracción que necesite o neno.
- Opción de poder distinguir entre as diferentes fases do sistema [PECS](#). Segundo o nivel do neno, o responsable podería habilitar a opción das frases ou deixar que só poidan discriminar pictogramas.
- Discernir entre os pictogramas cunha cor de fondo que indicaría o tipo de palabra, para así poder axudar ao neno a discriminar entre eles.
- Internacionalización: Engadir a opción de incluír outras linguas para que poida ser utilizada por máis persoas.
- Implementación dunha serie de melloras visuais e estéticas da aplicación, así como do seu rendemento.

Apéndices

Apéndice A

Vistas tablet

NAS vistas da *tablet*, como xa se mencionou en capítulos anteriores, aplícase o patrón *Master Detail*. Nas seguintes imaxes A.1, A.2, A.3, A.4, A.5, A.6, A.8, A.7 e A.9, pódense consultar algunhas das pantallas da aplicación para o caso dunha *tablet*.

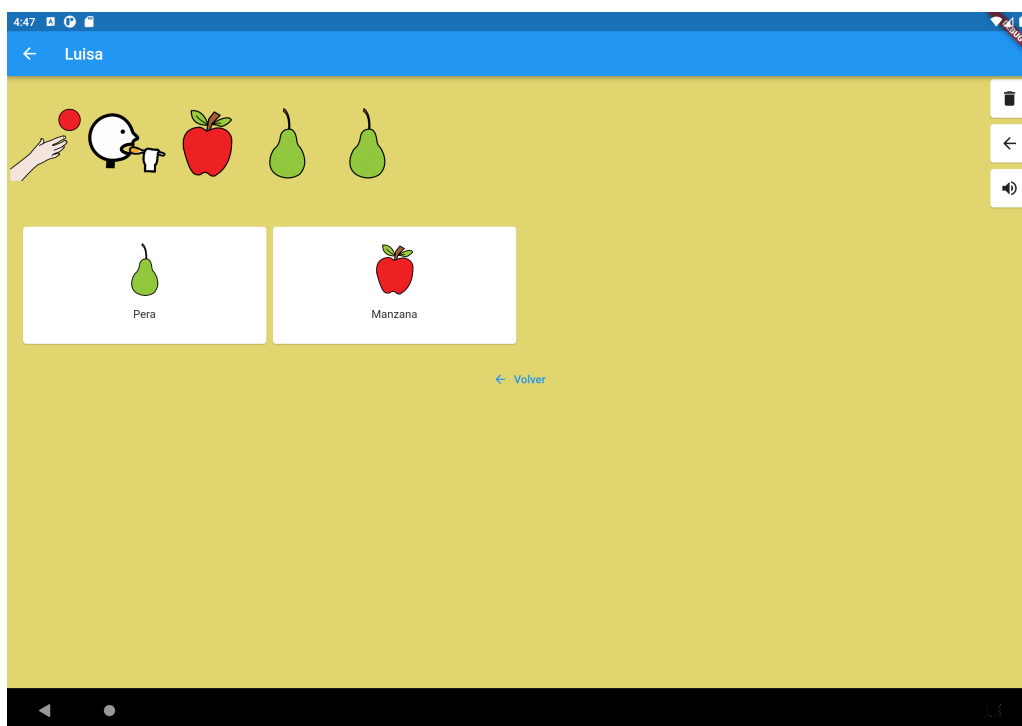


Figura A.1: Vista das pantallas dos nenos na *tablet*.

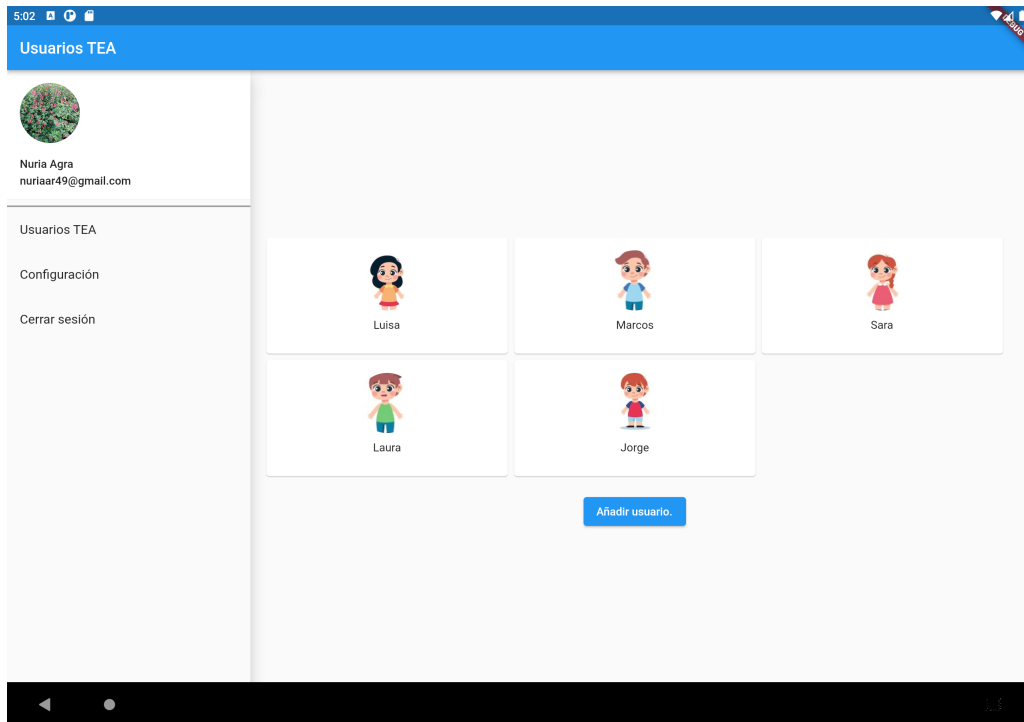


Figura A.2: Listado de usuarios TEA na *tablet*

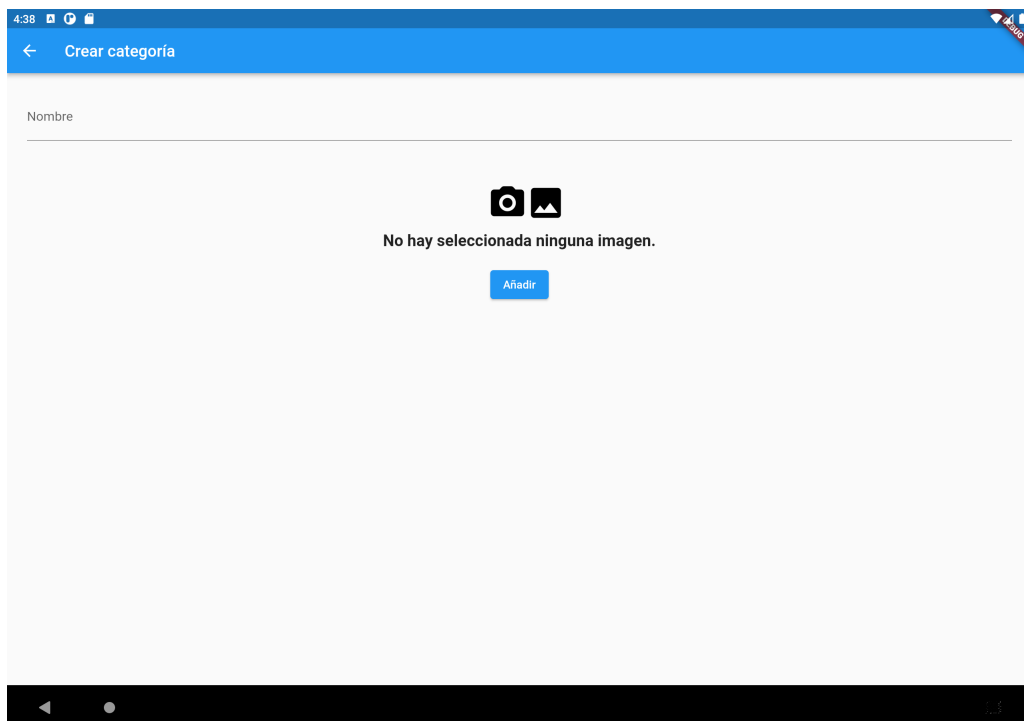


Figura A.3: Engadir categoría na *tablet*.

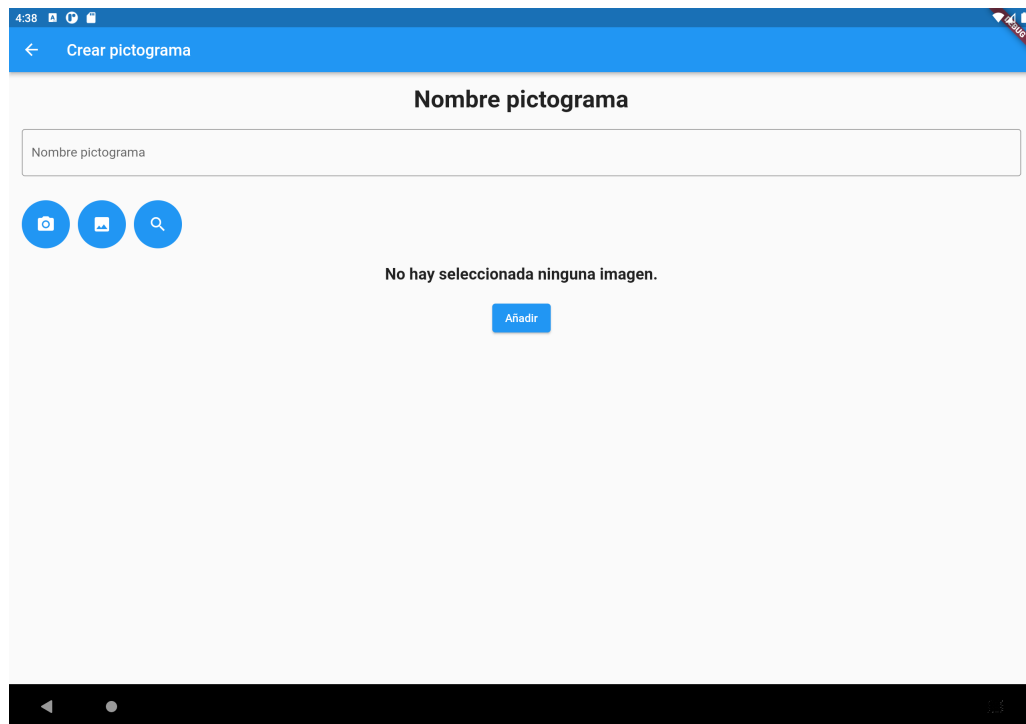


Figura A.4: Engadir pictograma *tablet*.

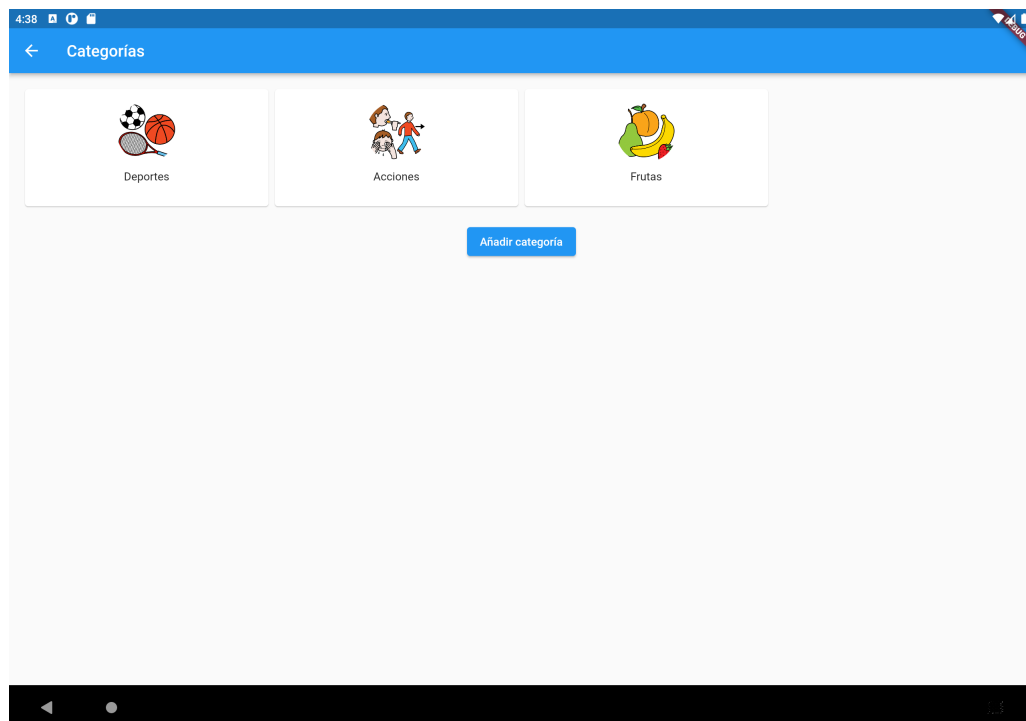


Figura A.5: Vista das categorías na *tablet*.

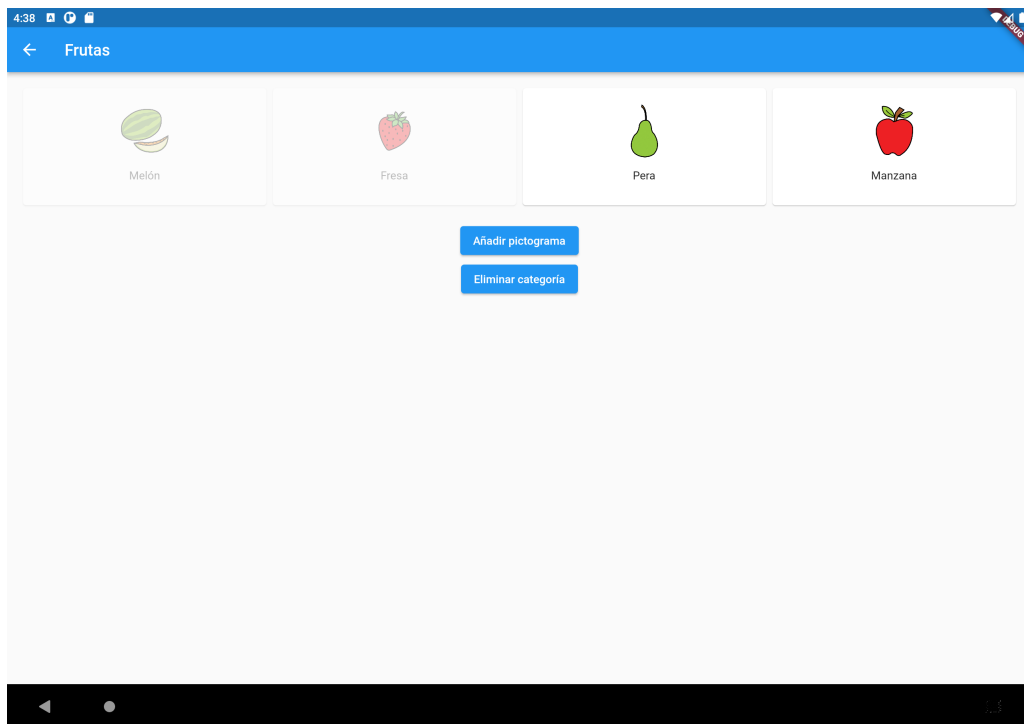


Figura A.6: Vista dos pictogramas na *tablet*.

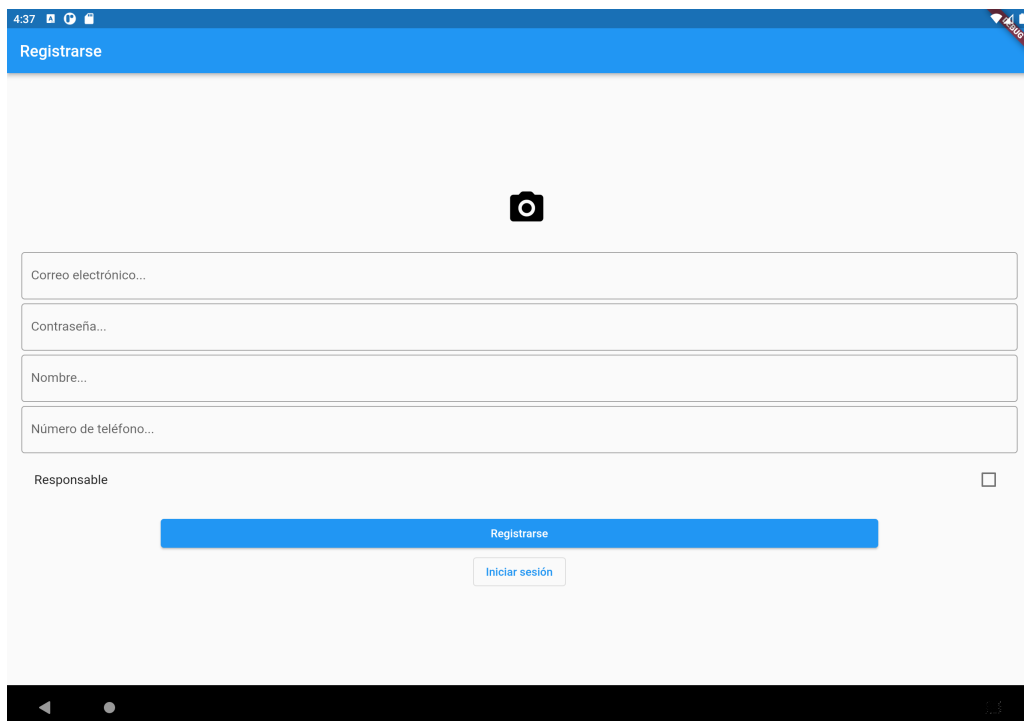


Figura A.7: Pantalla de rexistro na *tablet*.

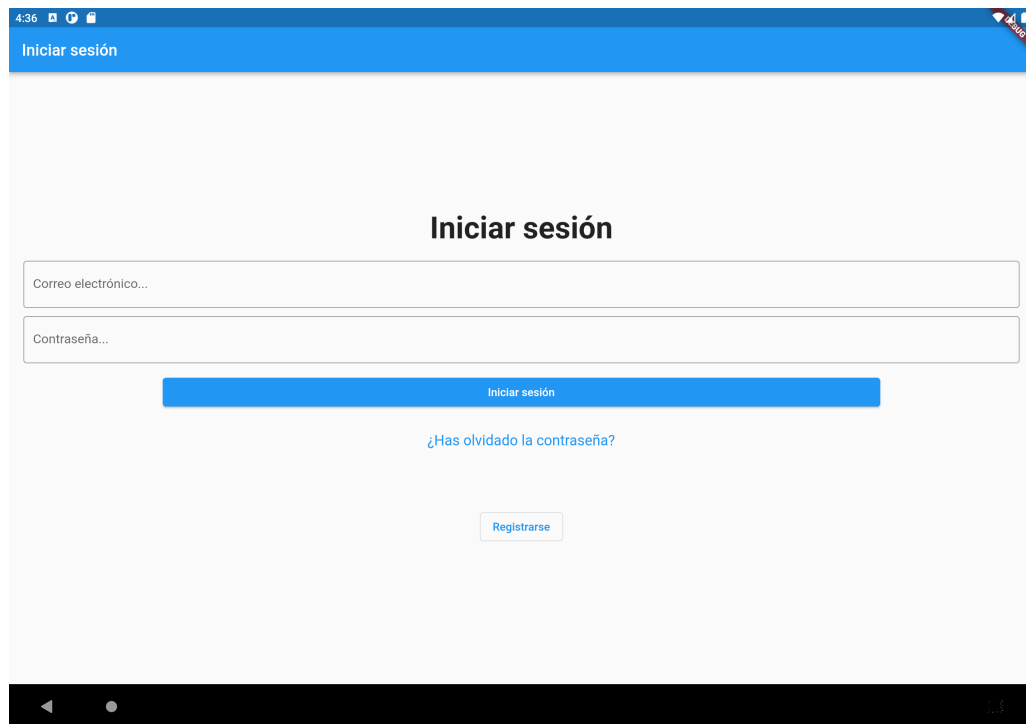


Figura A.8: Pantalla de inicio de sesión na *tablet*

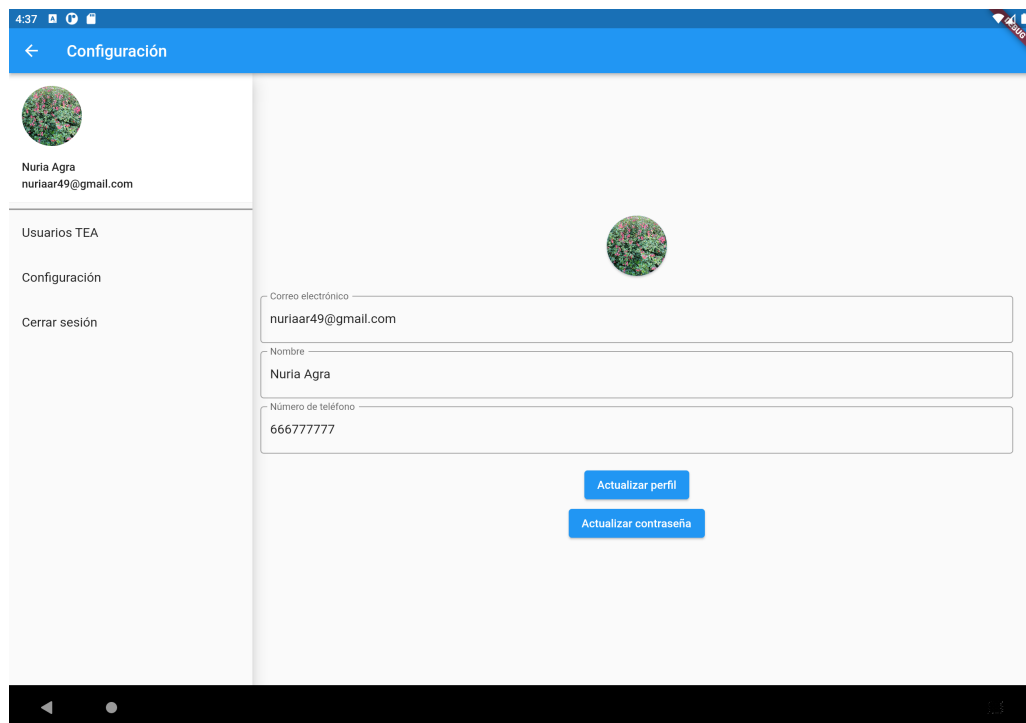


Figura A.9: Configuración do usuario na *tablet*

Relación de Acrónimos

API Application Programming Interface. v, vi, 9, 13, 14, 35, 39, 47, 51, 54, 61, 66

ApS Aprendizaxe e Servicio. 6, 19, 22, 64, 65

ARASAAC Aragón Sistemas Aumentativos e Alternativos de Comunicación. 10, 11, 13, 14, 35, 51, 53, 61, 65, 66

BLoC Business Logic Component. v, 33, 38–40, 51

IDE Integrated development environment. 15

PECS Picture Exchange Communication System. 2, 5–8, 25, 66

SAAC Sistemas Aumentativos e Alternativos de Comunicación. 1, 2, 8

TEA Trastorno Espectro Autista. v, vi, 1, 2, 4–8, 10, 13, 22, 24, 25, 30–37, 40, 42, 43, 45–47, 50, 57–61, 63, 65, 69

Bibliografía

- [1] N. I. of Mental Health, “Trastornos del espectro autista.” [En línea]. Disponible en: <https://www.nimh.nih.gov/health/publications/espanol/trastornos-del-espectro-autista>
- [2] N. I. N. de Salud Infantil y Desarrollo Humano, “Trastorno del espectro autista.” [En línea]. Disponible en: <https://medlineplus.gov/spanish/autismspectrumdisorder.html>
- [3] A. España, “Qué es el autismo.” [En línea]. Disponible en: <https://autismo.org.es/el-autismo/que-es-el-autismo/>
- [4] C. Basil and U. de Tècniques Augmentatives de Comunicació (UTAC), “¿Qué son los SAAC?” [En línea]. Disponible en: <https://arasaac.org/aac>
- [5] P. España, “El sistema de comunicación por el intercambio de imágenes (PECS).” [En línea]. Disponible en: <https://pecs-spain.com/el-sistema-de-comunicacion-por-el-intercambio-de-imagenes-pecs/>
- [6] U. I. de Valencia, “Los distintos tipos de trastorno del espectro autista (TEA): características y formas de intervención en el aula.” [En línea]. Disponible en: <https://www.universidadviu.com/es/actualidad/nuestros-expertos/los-distintos-tipos-de-trastorno-del-espectro-autista-tea>
- [7] C. para el Control y la Prevención de Enfermedades, “Tratamiento y servicios de intervención para el trastorno del espectro autista.” [En línea]. Disponible en: <https://www.cdc.gov/ncbddd/spanish/autism/treatment.html>
- [8] R. MP, “App que ayuda a la comunicacion para niños con autismo – Leeloo AAC.” [En línea]. Disponible en: <https://www.accesibles.org/app-que-ayuda-a-la-comunicacion-para-ninos-con-autismo-leeloo-aac/>
- [9] —, “SoyVisual, proyecto de comunicación aumentativa para la estimulación del lenguaje.” [En línea]. Disponible en: <https://www.accesibles.org/soyvisual-proyecto-de-comunicacion-aumentativa-para-la-estimulacion-del-lenguaje/>

- [10] —, “Comunicador AAC con pictogramas que facilita la comunicacion – Cboard.” [En línea]. Disponible en: <https://www.accesibles.org/comunicador-aac-con-pictogramas-que-facilita-la-comunicacion-cboard/>
- [11] F. Orange, “Dictapicto.” [En línea]. Disponible en: <https://fundacionorange.es/aplicaciones/dictapicto-tea/>
- [12] “Flutter.” [En línea]. Disponible en: <https://flutter.dev/>
- [13] “Firebase.” [En línea]. Disponible en: <https://firebase.google.com/?hl=es>
- [14] “Firebase authentication.” [En línea]. Disponible en: <https://firebase.google.com/docs/auth?hl=es-419>
- [15] “Firestore database.” [En línea]. Disponible en: <https://firebase.google.com/docs/firestore?hl=es-419>
- [16] “Firebase storage.” [En línea]. Disponible en: <https://firebase.google.com/docs/storage?hl=es-419>
- [17] “Dart.” [En línea]. Disponible en: <https://dart.dev/overview>
- [18] M. J. Martínez Cerdón, “¿Qué es el lenguaje de programación Dart?” [En línea]. Disponible en: <https://www.hiberus.com/crecemos-contigo/que-es-el-lenguaje-de-programacion-dart/>
- [19] “Android studio.” [En línea]. Disponible en: <https://developer.android.com/studio/intro?hl=es-419>
- [20] “¿Qué es Git?” [En línea]. Disponible en: <https://learn.microsoft.com/es-es/devops/develop/git/what-is-git>
- [21] J. M. Alarcón, “Qué es git, ventajas e inconvenientes y por qué deberías aprenderlo (bien).” [En línea]. Disponible en: <https://www.campusmvp.es/recursos/post/que-es-git-ventajas-e-inconvenientes-y-por-que-deberias-aprenderlo-bien.aspx>
- [22] G. B., “Qué es github y cómo empezar a usarlo.” [En línea]. Disponible en: https://www.hostinger.es/tutoriales/que-es-github#%C2%BFQue_es_GitHub
- [23] “Balsamiq.” [En línea]. Disponible en: <https://balsamiq.com/>
- [24] “draw.io (diagrams.net).” [En línea]. Disponible en: <https://app.diagrams.net>
- [25] “Wrike.” [En línea]. Disponible en: <https://www.wrike.com/es/>

- [26] Wikipedia, "Latex." [En línea]. Disponible en: <https://es.wikipedia.org/wiki/LaTeX>
- [27] "Overleaf." [En línea]. Disponible en: <https://es.overleaf.com/about>
- [28] B. de Zendesk, "¿Qué es la metodología ágil y cuáles son las más utilizadas?" [En línea]. Disponible en: <https://www.zendesk.com.mx/blog/metodologia-agil-que-es/#:~:text=La%20metodolog%C3%ADa%20%C3%A1gil%20es%20un,hasta%20el%20t%C3%A9rmino%20del%20proyecto.>
- [29] Atlassian, "Qué es Scrum y cómo empezar." [En línea]. Disponible en: <https://www.atlassian.com/es/agile/scrum>
- [30] Eslabón, "Ventajas de trabajar con el método Scrum." [En línea]. Disponible en: <https://www.eslabon.com.mx/es/articulo/32-ventajas-de-trabajar-con-el-metodo-scrum>
- [31] P. Ágiles, "Qué es Scrum." [En línea]. Disponible en: <https://proyectosagiles.org/que-es-scrum/>
- [32] J. Roche, "Scrum: roles y responsabilidades." [En línea]. Disponible en: <https://www2.deloitte.com/es/es/pages/technology/articles/roles-y-responsabilidades-scrum.html>
- [33] Alcibíades, "¿Qué es un Stakeholder?" [En línea]. Disponible en: <https://www.deconstruyendoscrum.com/que-es-un-stakeholder/>
- [34] P. Ágiles, "Artefactos Scrum ¿Qué son y para qué sirven?" [En línea]. Disponible en: <https://www.viewnext.com/artefactos-scrum/>
- [35] M. de Educación y Formación Profesional, "Aprendizaje y Servicio." [En línea]. Disponible en: <https://www.educacionyfp.gob.es/mc/sgctie/educacion-para-sostenibilidad/rec-aps.html#:~:text=%C2%BFQu%C3%A9%20es%20el%20Aprendizaje%20Servicio,con%20la%20finalidad%20de%20mejorarlo.>
- [36] R. E. Aprendizaje-Servicio, "¿Qué es? - Aprendizaje-Servicio." [En línea]. Disponible en: <https://www.aprendizajeservicio.net/que-es-el-aps/>
- [37] Indeed, "Sueldo de jefe de proyecto en A Coruña." [En línea]. Disponible en: <https://es.indeed.com/career/jefe-de-proyecto/salaries/A-Coru%C3%B1a--A-Coru%C3%B1a-provincia>
- [38] —, "Sueldo de analista funcional en A Coruña." [En línea]. Disponible en: <https://es.indeed.com/career/analista-funcional/salaries/A-Coru%C3%B1a--A-Coru%C3%B1a-provincia>

- [39] —, “Sueldo de diseñador/a en A Coruña.” [En línea]. Disponible en: <https://es.indeed.com/career/dise%C3%B1ador-web/salaries/A-Coru%C3%B1a--A-Coru%C3%B1a-provincia>
- [40] J. Sánchez Fernández, “Introducción al patrón BLoC.” [En línea]. Disponible en: <https://xurxodev.com/introduccion-al-patron-bloc/>
- [41] A. Cruz, “Material 3 en Flutter.” [En línea]. Disponible en: <https://www.desarrollolibre.net/blog/flutter/material-3-en-flutter#:~:text=En%20Flutter%2C%20Material%20es%20un,crear%20cualquier%20aplicaci%C3%B3n%20en%20Flutter.>
- [42] develou, “Patrón Master-Detail en Android.” [En línea]. Disponible en: <https://www.develou.com/patron-master-detail-android/>