
Editor de Pictogramas



**Trabajo de Fin de Grado
Curso 2020–2021**

Autores

**Alfonso Tercero López
Jorge García Cerros**

Directores

**Raquel Hervás Ballesteros
Gonzalo Méndez Pozo**

Grado en Ingeniería Informática

Facultad de Informática

Universidad Complutense de Madrid

Editor de Pictogramas

Trabajo de Fin de Grado en Ingeniería Informática
Departamento de Ingeniería del Software e Inteligencia
Artificial

Autores
Alfonso Tercero López
Jorge García Cerros

Directores
Raquel Hervás Ballesteros
Gonzalo Méndez Pozo

Convocatoria: Mayo 2021
Calificación:

Grado en Ingeniería Informática
Facultad de Informática
Universidad Complutense de Madrid

22 de mayo de 2021

Autorización de difusión

El abajo firmante, matriculado en el Máster en Ingeniería en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: "TITULO DEL TRABAJO", realizado durante el curso académico CURSO bajo la dirección de DIRECTORES en el Departamento de XXXXXXXXXXXXXXXXXXXXXXX, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Nombre Del Alumno

22 de mayo de 2021

Dedicatoria

Agradecimientos

Texto de los agradecimientos

Resumen

Resumen en español del trabajo

Palabras clave

Máximo 10 palabras clave separadas por comas

Abstract

Abstract in English.

Keywords

10 keywords max., separated by commas.

Índice

1. Introduction	1
1. Introducción	3
1.1. Motivación	3
1.2. Objetivos	5
1.3. Estructura de la memoria	5
2. Estado de la Cuestión	7
2.1. Sistemas Aumentativos y Alternativos de Comunicación	7
2.2. Pictogramas	8
2.2.1. Sistema Pictográfico de Comunicación - SPC	8
2.2.2. Blissymbolics	9
2.2.3. Sclera	10
2.2.4. Mulberry Symbols	11
2.2.5. Minspeak	12
2.2.6. ARASAAC	13
2.3. Aplicaciones basadas en pictogramas	15
2.3.1. Pictoselector	17
2.3.2. Editor ARASAAC	18
2.3.3. Piktoplus	20
2.3.4. Pictar	21
2.3.5. PicTableros	23
2.3.6. Symbo Talk	25
2.3.7. LetMe Talk	26
2.3.8. Jocomunico	27
3. Tecnologías	29
3.1. Introducción	29
3.2. API Arasaac	29

3.3. React	30
3.4. Drag and Drop	31
3.5. JSZip	31
4. PictUp!	33
4.1. Introducción	33
4.2. Prototipos Tecnológicos	34
4.2.1. API ARASAAC	34
4.2.2. React Drag and Drop:	34
4.3. Prototipo de papel interactivo	35
4.4. Prototipado de la interfaz	44
4.5. Herramientas	44
4.6. Componentes	45
5. Arquitectura	47
5.1. Cuadrícula e Ítems	47
5.2. Buscador de pictogramas	48
5.2.1. Búsqueda	49
5.2.2. Preajuste	50
5.2.3. Historial de pictos	52
5.2.4. PictoItem	53
5.3. Traducir frase a pictogramas	54
5.3.1. FraseItem	56
5.3.2. NIL WS Group	57
5.3.3. CORS	58
5.4. Añadir imagen al tablero	59
5.4.1. PhotoItem	60
5.4.2. Opciones investigadas y descartadas	61
5.5. Listas de pictogramas	62
5.5.1. Añadir y crear a una lista	62
5.5.2. Listas de Pictogramas creadas	64
5.6. Texto	65
5.6.1. TextoItem	65
5.7. Iconos	66
5.8. Descargar Tablero	67
6. Evaluación	69
6.1. Preparación de la evaluación	69
6.2. Diseño de la evaluación	70
6.3. Resultados de la evaluación	72
6.4. Análisis de los resultados	72

7. Conclusiones y Trabajo Futuro	73
7. Conclusions and Future Work	75
A. Título	77
B. Título	79

Índice de figuras

1.1. Tablero pictográfico en el que el usuario señala lo que quiere comunicar	4
2.1. Ejemplo de categorías en SPC	9
2.2. Ejemplo de Bliss-Characters	10
2.3. Ejemplo de Bliss-Words	10
2.4. Ejemplo de acciones en Sclera	11
2.5. Ejemplo de pictogramas de Mulberry	11
2.6. Ejemplo de Minspeak	12
2.7. ChatBox	13
2.8. Opciones de configuración de un pictograma	14
2.9. Acción de sacar la mascarilla	14
2.10. Pictograma que representa mascarilla mal colocada	14
2.11. Ejemplo de pictogramas más típicos de ARASAAC	15
2.12. Ejemplo de tablero para pictogramas en papel	16
2.13. Uso de tablero ETRAN	16
2.14. Ventana donde se edita el tamaño de la cuadrícula en el programa Pictoselector	17
2.15. Barra de inserción de pictogramas	17
2.16. Buscador y editor de Pictogramas	18
2.17. Ejemplo con el generador de frases	19
2.18. Ejemplo con el generador de horarios	19
2.19. Ejemplo con el generador de calendarios	19
2.20. Ejemplo con el generador de juegos	20
2.21. Pictoplus tablero	21
2.22. Subtablero Piktoplus	21
2.23. Funcionalidad en la aplicación Pictar de traducir frase	22
2.24. Funcionalidad en la aplicación Pictar de buscador	22
2.25. Funcionalidad en la aplicación Pictar de editor	23

2.26. Menú de Pictableros para utilizar una plantilla pública.	24
2.27. Ejemplo de creación de un tablero con un campo de texto y pictogramas.	24
2.28. Pantalla principal de la aplicación Symbo Talk.	25
2.29. Menú de la aplicación en Android de LetMe Talk.	26
2.30. Pantalla principal de la aplicación Jocomunico.	27
 3.1. Ejemplo de cómo se puede modificar el color de pelo y tono de piel de un pictograma entre otras opciones, en la web de ARASAAC	30
 4.1. Boceto de pantalla inicial	35
4.2. Boceto pantalla de plantillas	36
4.3. Boceto pantalla de edición de tableros	37
4.4. Boceto de las componentes canjón de picto y espacio picto.	37
4.5. Boceto de actividad con cajón de pictos y hueco picto	38
4.6. Boceto de las piezas que enlazan dos pictogramas al compartir la misma figura	38
4.7. Boceto de una escena de un cuento, con pictogramas en la zona inferior y botones en los laterales para pasar o retroceder la escena o tablero	39
4.8. Boceto del compositor de actividades	39
4.9. Pantalla de inicio con botones que muestran las otras pantallas de la aplicación.	40
4.10. Pantalla de configuración de tablero.	42
4.11. Pantalla de configuración de normas.	42
4.12. Pantalla de configuración de juego.	43
4.13. Pantalla de juego en ejecución.	44
 5.1. Vista del componente de búsqueda de pictogramas	49
5.2. Vista del modal del preajuste de pictos con las opciones color rubio y piel morena marcada.	51
5.3. Vista de la interfaz de búsqueda de pictogramas al aplicar un preajuste. En este caso piel morena y pelo rubio	52
5.4. Vista de un PictoItem en la cuadrícula con preajuste de pelo y piel.	54
5.5. Vista de la interfaz del modal que muestra una edición de pictogramas con más opciones. En este caso, se ha seleccionado el tiempo futuro y el borde verde.	55
5.6. Interfaz inicial Traducción frase a pictograma.	55
5.7. Resultado de la traducción de una frase	56
5.8. Vista de FraseItem en la cuadrícula	56

5.9.	Vista del modal de edición del FraseItem	57
5.10.	FraseItem tras aplicar los cambios	57
5.11.	Diagrama de conexiones entre aplicaciones para traducir una frase	59
5.12.	Interfaz inicial al añadir una imagen al tablero	59
5.13.	Vista de la Interfaz al cargar una imagen	60
5.14.	Ejemplod de PhotoItem con y sin texto	61
5.15.	Ejemmplo de PhotoItemn si no mantuviera la proporción . .	61
5.16.	Vistas del modal con ninguna lista y al menos una, respectivamente	63
5.17.	Alerta tras añadir satisfactoriamente un pictograma a la lista	63
5.18.	Alerta de acción no permitida	63
5.19.	Vista de las listas de pictogramas	64
5.20.	Vista de la selección de fuente y campo de texto	65
5.21.	Representación del TextoItem en la cuadricula	65
5.22.	Vista del modal para editar un TextoItem	66
5.23.	Vista de los botones que añaden iconos	66
5.24.	Ejemplo de uso de iconos sobre algunos pictogramas	67
5.25.	Vista del modal de edición de iconos.	67
5.26.	Vista del modal para descargar tablero	68
6.1.	Ejemplo de pista en el formulario: Buscar un pictograma. . .	71

Índice de tablas

- | | |
|--|----|
| 2.1. Tabla comparativa entre los distintos editores de tableros basados en pictogramas | 28 |
|--|----|

Chapter 1

Introduction

Introduction to the subject area.

Capítulo 1

Introducción

“La inteligencia es la habilidad de adaptarse a los cambios”
— Stephen Hawking

RESUMEN: En este capítulo se explicará la motivación de este trabajo (Sección 4), los objetivos que se quieren lograr (Sección 1.2) y la estructura de esta memoria (Sección 1.3).

1.1. Motivación

Los seres humanos siempre hemos tenido la necesidad inherente de comunicarnos, y quien no es capaz de hacerlo generalmente acaba excluido. Y esa es la palabra clave, comunicación. Su origen proviene del latín, “*communicare*” difundir, y este de “*communis*” común. Gracias a ello, hemos podido llegar hasta donde estamos actualmente, una era donde todos pueden tener una voz. Por eso es más importante que nunca no olvidarse de los que tienen dificultades. Eliminar la barrera de la comunicación a la que estamos acostumbrados como es el habla o el lenguaje natural, y hacerla comprensible para todos es un reto que se ha ido superando. [Quiero que darle un par de vueltas a esto]

Sin embargo, en las últimas décadas ha habido un avance sin precedentes en el estudio e investigación sobre las discapacidades comunicativas. Estos avances vinieron acompañados de herramientas y sistemas para facilitar la comunicación muchos de los cuales siguen vigentes a día de hoy. Uno de los principales perfiles que utilizan estos sistemas son las personas con Trastorno del Espectro Autista (TEA)

Sin entrar en gran detalle, podemos encontrar que la gente con TEA tienen dificultades en la comunicación verbal, pues a menudo la comunicación no es recíproca o no se realiza en el contexto social adecuado. Respecto a la

comunicación no verbal, también sufren dificultades al entender el significado de gestos faciales o expresión corporal de otras personas. Todo esto causa a menudo malentendidos, pues generalmente no se comprende correctamente el contexto y se dificulta la comunicación.

Para facilitar la comunicación de estos colectivos se utilizan otros medios alternativos como los sistemas pictográficos, que permiten comunicarse mediante imágenes. Estos sistemas pictográficos, al estar compuestos por cientos de pictogramas habitualmente, están agrupados en **tableros pictográficos**. Estos tableros son superficies donde se colocan pictogramas para formar mensajes. Un ejemplo de tablero es el que vemos en la Figura 1.1. Hasta hace poco, dichos tableros eran creados a mano recortando y pegando los pictogramas pero con el tiempo se han desarrollado herramientas informáticas enfocadas a trabajar con tableros y pictogramas.



Figura 1.1: Tablero pictográfico en el que el usuario señala lo que quiere comunicar.

Todas estas herramientas suelen tener cada una sus características y funcionalidades únicas. Aun así no existe una alternativa clara que reúna las herramientas necesarias para poder realizar materiales con amplia libertad. A menudo se abusan de tableros de tamaño fijo que limitan la creatividad de quienes crean estos materiales. Por ejemplo, en las aplicaciones mencionadas previamente podemos ver que falta algún tipo de cuadrícula en el tablero para que a la hora de insertar los pictogramas estos queden perfectamente colocados y el tablero tenga una mejor apariencia visual.

En este contexto, la finalidad es crear una herramienta que unifique las mejores características de cada aplicación, además de permitir crear y editar tableros en un mismo lugar con la mayor facilidad y libertad posible. Afortunadamente, ya existe una base sobre la que apoyarse, gracias a proyectos anteriores.

1.2. Objetivos

Teniendo en cuenta todos los temas que hemos tratado en la introducción, el objetivo principal de este trabajo es desarrollar una aplicación web multiplataforma que permitirá la creación y edición de tableros interactivos. En la aplicación también se incorporarán herramientas de búsquedas de pictogramas por palabras y traducción de texto a pictograma.

Respecto a la interacción con los tableros, se hará por medio de componentes configurables, los cuales aportarán algún tipo de interacción con el usuario final, como desplegar un tablero de pictogramas o colocar un pictograma en un hueco.

Para poder cumplir todos los objetivos mencionados se tendrá como referencia los TFG y TFM de Pictar y PicTableros. También se hará una labor de investigación en busca de nuevas tecnologías y herramientas con las que trabajar para desarrollar la aplicación.

La forma en la que se comprobará el estado de los objetivos y su evolución será por medio de reuniones con los directores del TFG donde se analizará el trabajo realizado para ver el progreso, la forma en la que se implementan los objetivos, y sugerencias para no pausar el progreso.

[¿Podemos poner el inestimable apoyo que nos proporcionáis para animarnos a continuar?](#)

1.3. Estructura de la memoria

La estructura para memoria se encuentra dividida en [x] capítulos, a continuación se explicará brevemente su contenido. –según avancemos habrá que ir completándolo–

- En los capítulos 1 y 2 se expondrá el contexto bajo el cual se ha realizado el trabajo junto a la motivación y objetivos para realizarlo.
- En el capítulo 3 se explicará qué es un pictograma y los distintos sistemas de comunicación basados en ellos. Además se analizarán las distintas herramientas relacionadas con pictogramas haciendo énfasis en la edición de tableros.

Capítulo 2

Estado de la Cuestión

RESUMEN: En este capítulo se dará una idea general sobre los Sistemas Aumentativos y Alternativos de Comunicación (Sección 2.1) y las características de los pictogramas y los distintos sistemas pictográficos que existen (Sección 2.2). Finalmente, se verán distintas herramientas con las que se construyen tableros de pictogramas (Sección 2.3).

Falta: Introducción SAACs

2.1. Sistemas Aumentativos y Alternativos de Comunicación

Los Sistemas Aumentativos y Alternativos de Comunicación¹ (*SAAC*) son las distintas formas de expresión sin tener en cuenta el lenguaje hablado que tiene como finalidad aumentar y/o compensar los problemas de comunicación de personas con discapacidad como por ejemplo trastornos del espectro autista, discapacidad intelectual, deficiencia auditiva, parálisis cerebral entre otros.

En ocasiones puede hacer falta el uso de recursos para poder comunicarse, es por ello que podemos distinguir dos tipos de *SAAC*, los sistemas sin ayuda y los sistemas con ayuda.

- **Sistemas sin ayuda:** no utilizan ningún recurso externo para establecer la comunicación, únicamente usan su propio cuerpo. En los sistemas sin ayuda podemos observar dos tipos de grupos, los métodos gestuales (lengua de signos) y los métodos oralistas (lectura labiofacial).

¹https://aulaabierta.arasaac.org/que_son_los_sistemas_aumentativos_y_alternativos_de_comunicacion

- **Sistemas con ayuda:** utilizan recursos externos para establecer la comunicación. Los más utilizados suelen ser pictogramas, imágenes o símbolos.

Las *SAAC* utilizan múltiples recursos para poder comunicarse con personas con discapacidades cognitivas y entre todos ellos destacan los sistemas pictográficos. Se trata de uno de los sistemas más utilizados y esto es debido a su fácil comprensión ya que representan gráficamente lo que se desea transmitir como palabras o conceptos.

2.2. Pictogramas

Los pictogramas son imágenes o símbolos de rápida comprensión que expresan acciones, objetos, emociones, etc. Un conjunto de pictogramas en un cierto orden pueden representar una oración. Todos ellos deben cumplir las siguientes características²:

1. **Referencialidad:** relación del pictograma con el referente.
2. **Ítems gráficos:** imágenes que representen de manera sencilla aquello que se toma como modelo.
3. **Comprensión:** deben ser fácilmente entendibles independientemente de la formación, idioma o discapacidad.
4. **Legibilidad:** deben mantener una coherencia visual entre pictogramas.
5. **Sencillez:** mostrar únicamente los elementos relevantes sin elementos distractores o adornos insignificantes.

Existen numerosos sistemas pictográficos. A continuación hablaremos de algunos de los más relevantes.

2.2.1. Sistema Pictográfico de Comunicación - SPC

El Sistema Pictográfico de Comunicación³ (*SPC*) fue creado en 1981 por Roxana Mayer Johnson, con la intención de facilitar la comunicación a quienes tienen un nivel de lenguaje expresivo simple o vocabulario limitado. Gracias a la diferenciación por colores, facilita la comprensión de la estructura sintáctica. Actualmente cuenta con más de 3.000 iconos. Está organizado por seis colores según su función gramatical, como podemos ver en la Figura 2.1.

²http://aularagon.catedu.es/materialesaularagon2013/arasaac/ZIPs/Modulo_1/contenidos.html

³<https://www.uv.es/bellochc/logopedia/NRTLogo8.wiki?8>

- **Personas** (Amarillo): representan a familiares y pronombres. Ejemplo: mamá, familia, yo, ellos.
- **Verbos** (Verde): representan acciones. Ejemplo: abrir, agarrar, comer, ir.
- **Descriptivos** (Azul): representan descripciones, adjetivos y adverbios. Ejemplo: bonito, triste, vacío, lleno.
- **Nombre** (Naranja): representan objetos u otros elementos que no aparecen en otra categoría. Ejemplo: gato, almohada o casa.
- **Miscelánea** (Blanco): representa números, letras y colores
- **Social** (Rosa): vocabulario relacionado con relaciones sociales. Ejemplo: buenos días, sí, gracias, no lo sé.

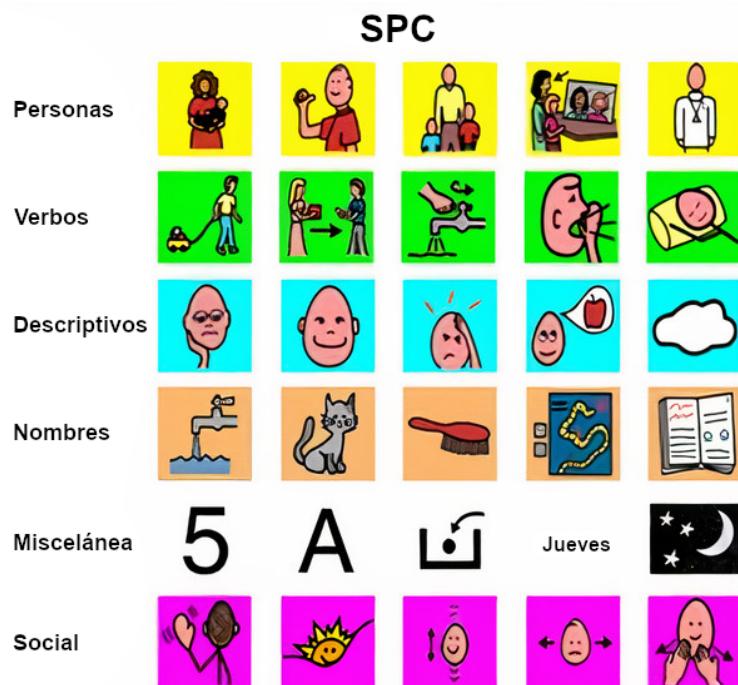


Figura 2.1: Ejemplo de categorías en SPC.

2.2.2. Blissymbolics

Byssimboldics⁴ es un sistema de comunicación que fue usado por primera vez en 1971 para facilitar la comunicación con niños que padecían algu-

⁴<https://www.blissymbolics.org/index.php/about-blissymbolics>

na discapacidad. Actualmente está compuesto por más de 5.000 símbolos o *Bliss-Words*, los cuales a su vez están compuestos por uno o más Caracteres-Bliss o *Bliss-Characters*. A pesar de que los 150 *Bliss-Characters* que hay son sencillos de dibujar, requieren un periodo de aprendizaje para comprender su significado y así también el de las *Bliss-Words*. En la Figura 2.2 vemos algunos *Bliss-Characters* con su significado.



Figura 2.2: Ejemplo de Bliss-Characters.

Una vez comprendidos los *Bliss-Characters*, en la Figura 2.3 vemos cómo se han combinado para generar *Bliss-Words*. Destacar que el orden, el tamaño o la posición de los los Bliss-Characters puede alterar su significado. Adicionalmente pueden estar agrupados usando los mismos colores vistos en SPC.



Figura 2.3: Ejemplo de Bliss-Words.

2.2.3. Sclera

La principal característica de Sclera⁵ frente a otros sistemas pictográficos es que sus pictogramas son menos coloridos, pero cuentan con pictogramas más avanzados en cuanto a las acciones representadas. Un ejemplo de ello

⁵<https://www.sclera.be/en/picto/overview>

se puede ver en la Figura 2.4 donde la acción de pedir atención se puede realizar de dos maneras posibles. Cuenta con un total de 11.497 pictogramas en español. En la actualidad el desarrollo de Sclera está paralizado desde 2015.



Figura 2.4: Ejemplo de acciones en Sclera.

2.2.4. Mulberry Symbols

Mulberry Symbols⁶ se creó con el propósito de ser un sistema pictográfico orientado a adultos, ya que un gran porcentaje de dichos sistemas estaban pensados principalmente para niños y dificultaban la comunicación por falta de pictogramas. Como se observa en la Figura 2.5 podemos ver ejemplos de pictogramas enfocados a adultos como cerveza o fumar.



Figura 2.5: Ejemplo de pictogramas de Mulberry.

Todos los pictogramas pueden descargarse gratuitamente desde su página web en formato ZIP, y las imágenes se encuentran en formato SVG o formato CSV donde están categorizados según su nombre y categoría. Los pictogramas de Mulberry cuentan con 118 categorías incluyendo sustantivos,

⁶<https://mulberrysymbols.org/>

pronombres y verbos sumando un total de 3.436 pictogramas. A diferencia de otros sistemas pictográficos, Mulberry sigue en activo añadiendo constantemente nuevos pictogramas.

Los pictogramas de Mulberry son utilizados por muchas aplicaciones como BoradBuilder⁷ (aplicación web para diseñar tableros de comunicación), Cboard⁸ (aplicación web de comunicación que usa pictogramas y conversión de texto a voz) o CommuniKate⁹ (página web que ofrece tableros diseñados en Powerpoint o Impress). Una de las últimas herramientas creadas que hace uso de los pictogramas de Mulberry es la diseñada por Eliada Pampoulou y Maria Constanta de la Universidad Tecnológica de Chipre la cual usa unas plantillas imprimibles en inglés¹⁰ o griego para ayudar a la comunicación de los pacientes con la COVID-19 que tienen dificultades para comunicarse.

2.2.5. Minspeak

Minspeak¹¹, es un sistema de comunicación alternativo creado por Bruce Baker en 1982. Difiere con los vistos anteriormente en que el significado de los iconos no viene preestablecido sino que es acordado entre usuario y logopeda. Es por ello que cada ícono acordado tenga un significado distinto según la secuenciación de íconos. Por ejemplo, en el caso de la Figura 2.6 la asociación del ícono casa junto a la cama en un orden podría ser interpretado como dormitorio.



Figura 2.6: Ejemplo de Minspeak.

Como cada pictograma puede tener un significado distinto según su orden, se crearon Programas de Comunicación Minspeak (*PAM*). Éstos se usan para que cuando una casilla o ícono haya sido seleccionada, se activen los posibles íconos con los que pueda tener algún tipo de relación. Inicialmente se creó hardware específico (como aparece en la Figura 2.7) con 16 celdas las cuales podía generar hasta 1.024 mensajes, estos teclados evolucionaron con

⁷<https://globalsymbols.com/boardbuilder/boardsets>

⁸<https://www.cboard.io/>

⁹<http://communikate.equalitytime.co.uk/>

¹⁰https://mulberrysymbols.org/assets/COVID19/Covid-19_AAC-EN.pdf

¹¹<http://ares.cnice.mec.es/informes/18/contenidos/94.htm>

más celdas y combinaciones, hasta pasar a teclados digitales implementados por software como PortaVoz¹².



Figura 2.7: Panel de ChatBox

2.2.6. ARASAAC

El portal Aragonés de Comunicación Aumentativa y Alternativa (*ARASAAC*¹³) surge en 2007 gracias a la colaboración entre el personal del CATEDU, el colegio público de educación Especial Alborada y del Centro Politécnico Superior de la Universidad de Zaragoza. Su objetivo era la creación de un sistema pictográfico de libre distribución que ayudara en el ámbito de la comunicación a todas aquellas personas que lo necesitasen. Actualmente el portal de *ARASAAC* incluye fotografías, vídeos y cuenta con más de 3.9000 pictogramas tanto a color como en blanco y negro y con traducciones a 20 idiomas. También ofrece herramientas online con las que poder generar materiales como por ejemplo calendarios, tableros, creador de símbolos, etc.

A diferencia de otros sistemas pictográficos, *ARASAAC* permite una gran cantidad de opciones configurables como el color de fondo, el marco o el tiempo verbal. Esto lo podemos ver en la Figura 2.8 donde además de las opciones vistas, se puede modificar cambiar la posición del texto, la resolución o su posición.

En la actualidad *ARASAAC* es uno de los sistemas pictográficos más utilizados a nivel de educación especial en España. Sus pictogramas se utilizan en colegios, universidades e incluso se han creado asociaciones para facilitar su implantación. CreaTea [PONER LINK] es una asociación en la Comunidad de Madrid cuyo objetivo es habilitar lugares como clínicas, restaurantes o ayuntamientos, para ayudar a la inclusión de personas con dificultades en la comunicación y concienciar a la sociedad.

¹²http://www.terapia-ocupacional.com/articulos/Portavoz_JMLedesma.shtml

¹³<http://www.arasaac.org/>



Figura 2.8: Opciones de configuración de un pictograma.

También cabe destacar que *ARASAAC* ha recibido varios premios por su labor y que es una herramienta utilizada en varios países por lo que la cantidad de usuarios que colaboran es muy alta. Esto queda reflejado en la cantidad de pictogramas que se publican a la plataforma por parte de colaboradores sin ánimo de lucro y dichos pictogramas están constantemente actualizándose. Un ejemplo de ellos son los pictogramas que se han publicado debido a la pandemia como por ejemplo los que podemos ver en las Figuras 2.9 y 2.10.

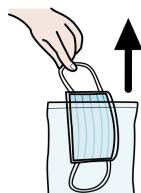


Figura 2.9: Acción de sacar la mascarilla.



Figura 2.10: Pictograma que representa mascarilla mal colocada.

En la Figura 2.11 podemos ver algunos ejemplos de pictogramas de *ARASAAC* en situaciones o casos más cotidianos. Destacan la manera clara y concisa en la que están representados.

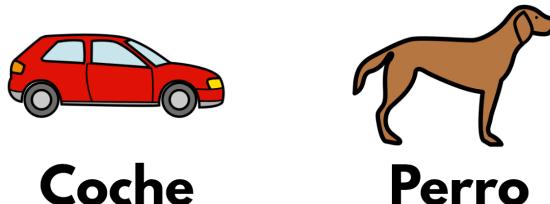


Figura 2.11: Ejemplo de pictogramas más típicos de *ARASAAC*.

La licencia de *ARASAAC* es de tipo *Creative Commons (BY-NC-SA)*, por lo que se podrá utilizar el material elaborado por ellos de cara a la implementación del Trabajo Fin de Grado. Utilizaremos sus pictogramas publicados en su página web y la API que han desarrollado para acceder a pictogramas de su base de datos.

2.3. Aplicaciones basadas en pictogramas

Ya hemos visto multitud de sistemas pictográficos. Pero para trabajar con ellos es necesario que sean fáciles de acceder, pues lo normal encontrar cientos de pictogramas en cada sistema pictográfico. La solución a este problema, son los tableros pictográficos. Los tableros son un área en la que se pueden colocar pictogramas, fotografías o palabras que la persona indicará para comunicarse. Pueden tener distintas funciones, como por ejemplo hacer un horario, normas, o elegir entre distintas opciones mediante pictogramas. A menudo estos tableros se realizan mediante piezas de papel recortadas, como podemos ver en la Figura 2.12.

Estos tableros¹⁴ no tienen por qué limitarse a un espacio rectangular, sino que se pueden usar de maneras más creativas dependiendo de las discapacidades de la persona que lo use. Por ejemplo los *ETRAN*¹⁵ (“Eye-Transfer”) son usados por gente con baja movilidad y que apuntan al pictograma con la mirada, de manera que otra persona está al otro lado del *ETRAN* para ver qué pictograma está mirando como podemos ver en la Figura 2.13. Otro ejemplo son los cuadernos de comunicación, que como su nombre indica son un conjunto de hojas o tableros con símbolos básicos.

¹⁴<http://burbujadelenguaje.blogspot.com/2016/05/tablero-de-comunicacion-tea.html>

¹⁵<http://psicosociosanitario.blogspot.com/2016/05/tableros-de-comunicacion.html>



Figura 2.12: Ejemplo de tablero para pictogramas en papel.



Figura 2.13: Uso de tablero ETRAN.

Como trabajar con piezas de papel puede resultar engorroso y a menudo poco eficiente, los tableros pictográficos han sufrido una evolución natural al medio digital, y con ello ha aparecido gran cantidad de software de edición de tableros pictográficos. Gracias a ello, las personas que usan estos nuevos sistemas ahorran mucho tiempo, como buscar pictogramas rápidamente, alinearlos perfectamente, editarlos o incluso poder compartir los tableros.

Para crear y editar tableros se han creado multitud de aplicaciones. A continuación estudiaremos algunas de sus características.

2.3.1. Pictoselector

Pictoselector¹⁶ una herramienta gratuita para crear agendas visuales. Recopila más de 28.000 pictogramas provenientes de *Sclera*, *Mulberry*, *ARA-SAAC*, etc. Al crear un proyecto, permite cargar una plantilla o crear una desde cero. Se puede modificar el número de filas y columnas, la posición del texto y el tamaño del borde de los pictogramas como podemos ver en la Figura 2.14 .



Figura 2.14: Ventana donde se edita el tamaño de la cuadrícula en el programa Pictoselector.

Una vez creado el tablero, podemos insertar en su cuadrícula distintos elementos, muchos de ellos en forma de pictograma. Para facilitar esta tarea, la cabecera de la aplicación contiene acceso directo a la inserción de pictogramas, como podemos ver en la Figura 2.15. De izquierda a derecha, existe un buscador de pictogramas que incluye la función de filtrar por juego de pictogramas como se ve en la Figura 2.16. Además permite editar ligeramente el picto ya sea coloreándolo o añadiendo un signo de pasado, presente o plural. Para el marcaje de tiempo pueden incluirse con facilidad pictogramas de reloj que marcan la hora, y de duración que marcan un intervalo de tiempo. Adicionalmente se puede importar imágenes propias, códigos QR, texto o emoticonos.



Figura 2.15: Barra de inserción de pictogramas.

A pesar de ser una herramienta muy completa respecto a búsqueda y edición de pictogramas, la colocación de éstos está limitada a una cuadrícula.

¹⁶<https://www.pictoselector.eu/es/>

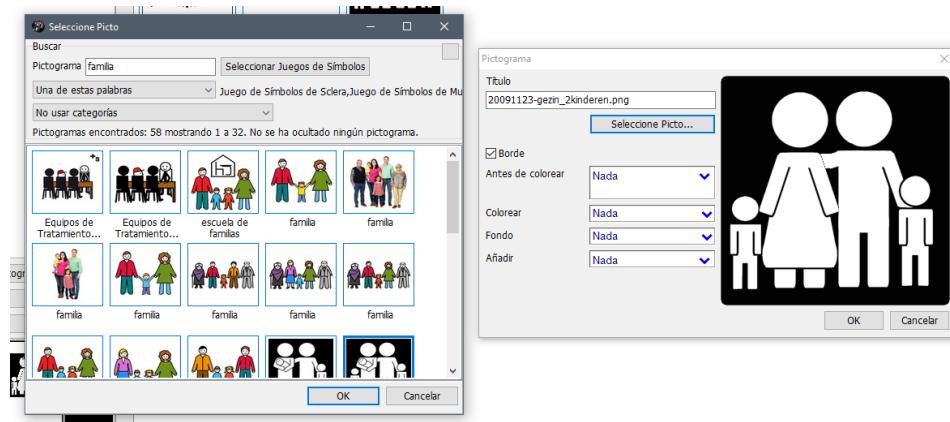


Figura 2.16: Buscador y editor de Pictogramas.

Por ello, no se pueden colocar pictogramas de distintos tamaños con facilidad o en posiciones intermedias.

2.3.2. Editor ARASAAC

La página web de ARASAAC¹⁷ cuenta con diversas herramientas online que se pueden usar para generar materiales:

- **Creador de animaciones:** genera una animación con los pictogramas que queramos en formato GIF o SWF. También permite configurar el intervalo entre los pictogramas y el número de repeticiones que hará.
- **Creador de símbolos:** permite la personalización de pictogramas donde podremos cambiar el nombre del pictograma, poner su traducción, modificar la fuente del texto, poner un marco, ampliar la imagen y cambiar el fondo.
- **Generador de frases:** consta de un total de tres pasos a seguir. El primero de ellos consiste en seleccionar las palabras que queramos traducir a pictogramas. El segundo paso nos mostrará todos los pictogramas asociados para cada palabra introducida y deberemos seleccionar el que más nos guste. Y en el tercer paso aparecerán todos los pictogramas colocados en una tabla en la cual podremos añadir un marco, insertar el nombre de los pictogramas y hacer más grande la tabla resultante. En la Figura 2.17 podemos ver la traducción de la frase: "Me gustan los helados".
- **Generador de horarios:** genera un horario donde previamente tendremos que configurar una plantilla con los días, horas, el formato

¹⁷ <http://www.arasaac.org/herramientas.php>



Figura 2.17: Ejemplo con el generador de frases.

(horizontal o vertical), idioma, bordes del horario, texto para cada día y hora, y la opción de insertar un pictograma en función de su día y hora. En la Figura 2.18 vemos un horario generado por esta herramienta en el que se especifica para cada día y hora una acción a realizar.

	lunes reunión	martes estudiar	miércoles pintar	jueves jugar
	pintar	jugar	reunión	estudiar

Figura 2.18: Ejemplo con el generador de horarios.

- **Generador de calendarios:** genera un calendario donde tendremos que especificar el mes, año e idioma deseado. Al igual que en el generador de horarios, permite la opción de modificar el texto, colores, bordes y la posibilidad de poner un pictograma para cada día del mes. En la Figura 2.19 vemos el calendario generado para el mes de Noviembre de 2020

noviembre 2020						
lunes	martes	miércoles	jueves	viernes	sábado	domingo
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Figura 2.19: Ejemplo con el generador de calendarios.

- **Generador de tableros:** crea un tablero con el número de filas y columnas deseado donde para cada casilla podremos insertar un pictograma. Al igual que en otras herramientas permite la modificación de colores, bordes y texto del tablero.
- **Creador de juegos:** genera una plantilla en formato .rtf para poder jugar al bingo, oca, dominós y dominós encadenados con los pictogramas.

mas que deseemos. En la Figura 2.20 vemos la creación de un tablero para jugar al bingo.

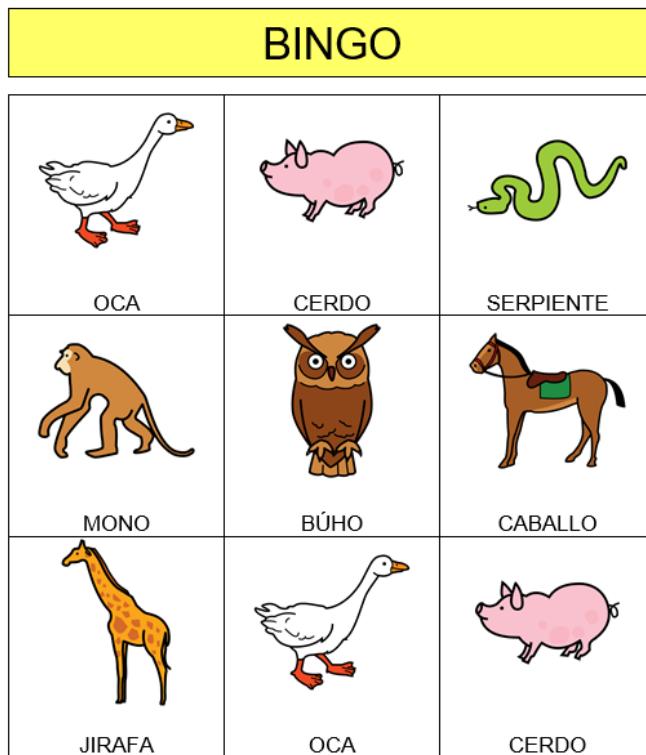


Figura 2.20: Ejemplo con el generador de juegos.

2.3.3. Piktoplus

Piktoplus¹⁸ es una aplicación para dispositivos Android que tiene como particularidad la creación de un avatar tridimensional personalizable. Dicho avatar será usado en los tableros pictográficos pues será quien protagonice las acciones. Permite registrar múltiples usuarios, cada uno con su propio avatar. Otra particularidad de Piktoplus son los sub-tableros¹⁹. Por ejemplo, en la Figura 2.21 podemos ver un tablero con dos pictogramas: “Estoy” y “Me duele”.

La función de dicho sub-tablero es la de mostrar pictogramas relacionados con el picto seleccionado para formar una frase de manera más natural. Por ejemplo, si seleccionamos el pictograma de “Estoy” como en la Figura 2.22, aparecería el sub-tablero con cuatro posibles pictogramas relacionados. En

¹⁸<http://www.aulautista.com/2013/12/05/piktoplus-un-comunicador-android-muy-especial/>

¹⁹<https://fatimamikel.wordpress.com/2014/04/17/piktoplus-2/>

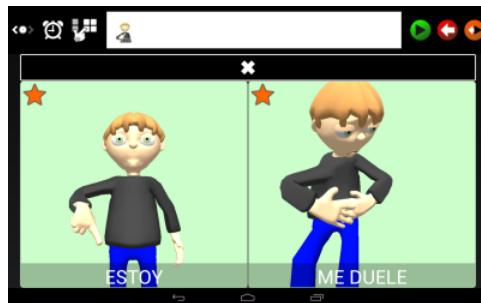


Figura 2.21: Ejemplo de tablero en Piktoplus

este caso aparecen relacionados con el estado anímico, facilitando la creación de la frase “*Estoy + Contento*”.

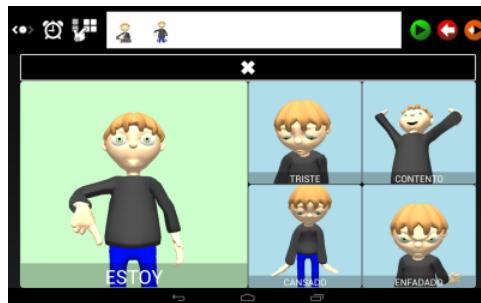


Figura 2.22: Subtablero en Piktoplus

Respecto a la creación y edición de tableros, se trata de una cuadrícula sobre la cual se colocan los pictogramas, además permite aumentar el tamaño de los pictos. Por ejemplo “*Estoy*” ocupa cuatro celdas más que “*Contento*”.

Actualmente esta aplicación no está disponible para descargar en tiendas de aplicaciones habituales y su desarrollo ha cesado desde 2018. Aunque no esté disponible, plantea una idea muy interesante como la posibilidad de desplegar un sub-tablero a partir de un pictograma para mostrar pictogramas relacionados entre ellos.

2.3.4. Pictar

Pictar²⁰ es una aplicación web desarrollada por Alejandro Martín Guerrero de la Universidad Complutense de Madrid como Trabajo de Fin de Máster (?). El propósito de Pictar es poder tener una aplicación web accesible desde cualquier dispositivo con conexión a internet para facilitar la comunicación a personas con TEA.

Pictar ofrece tres herramientas en su página web:

²⁰<http://hypatia.fdi.ucm.es/pictar/>

- **Traducir frase:** permite generar una secuencia de pictogramas asociados a una frase o texto introducido por el usuario. Ofrece la posibilidad, tras haber generado la secuencia de pictogramas, de poder cambiarlos por otros pictogramas del mismo significado mediante unas flechas que se encuentran tanto encima como debajo de cada pictograma. También incluye un ícono que tiene como finalidad copiar la secuencia de pictogramas generados al tablero (ver la Figura 2.23).

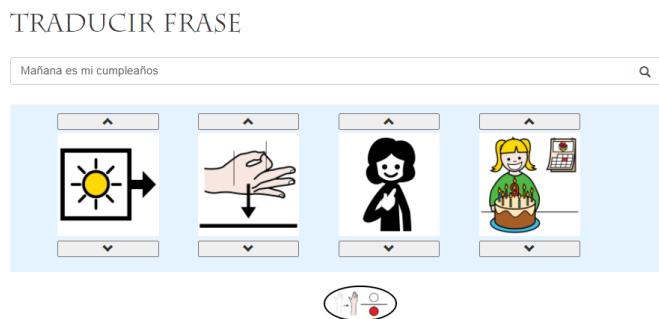


Figura 2.23: Funcionalidad en la aplicación Pictar de traducir frase.

- **Buscador:** al introducir una palabra en el campo de búsqueda nos mostrará todos aquellos pictogramas con un significado igual o similar a la palabra introducida. Podemos ver un ejemplo de ello en la Figura 2.24. El buscador ofrece la posibilidad de poder arrastrar los pictogramas para insertarlos al tablero que se muestra en la Figura 2.25.



Figura 2.24: Funcionalidad en la aplicación Pictar de buscador.

- **Editor:** permite generar un tablero de pictogramas en el que debere-

mos seleccionar cuantos elementos va a tener en total y el número de columnas en los que se divide. Para añadir pictogramas al tablero tenemos dos opciones: la primera de ella es copiar la secuencia generada al traducir una frase a pictogramas y la segunda buscar un pictograma en el buscador para poder arrastrar el pictograma deseado al tablero. Por cada pictograma insertado en el tablero tendremos dos opciones debajo de éste situadas en las esquinas inferiores izquierda y derecha que permiten eliminar o dejar la casilla en blanco. El editor ofrece la posibilidad de añadir el nombre a cada pictograma, poner todos los pictogramas a color o blanco y negro y exportar o importar el tablero. Todas estas características se pueden observar en la Figura 2.25.



Figura 2.25: Funcionalidad en la aplicación Pictar de editor.

2.3.5. PicTableros

PicTableros²¹ es una aplicación web desarrollada por Carmen López Gonzalo de la Universidad Complutense de Madrid en el Grado de Ingeniería Informática como Trabajo de Fin de Grado (?). PicTableros tiene como objetivo ayudar a la comunicación de personas con discapacidades cognitivas y poder realizar de una forma ágil plantillas y tableros. La aplicación tiene dos partes diferenciadas:

- **Plantillas:** en la sección de plantillas se permite seleccionar tres modalidades: públicas, crear nuevas y privadas. En el apartado de públicas permite elegir diferentes plantillas ya creadas que las podremos usar como tableros, (ver Figura 2.26). En el apartado de crear nuevas nos permitirá añadir en la plantilla áreas para insertar pictogramas, texto y formas como triángulos, estrellas, flechas, etc. Y por último en la sección de privadas podremos exportar o importar plantillas que tengamos creadas en nuestro equipo.
- **Tableros:** al igual que en el apartado anterior, la sección de tableros consta de tres secciones: públicos, crear nuevos y privados. En el apartado de públicos aparecen tableros ya creados que podremos duplicar

²¹<https://holstein.fdi.ucm.es/picto-tableros/>



Figura 2.26: Menú de Pictableros para utilizar una plantilla pública.

como copia privada y modificarlo, o ver el tablero. En crear nuevo ofrece la posibilidad de insertar en el tablero áreas para añadir pictogramas, insertar texto y formas geométricas, se puede ver un ejemplo de creación de un tablero en la Figura 2.27. Por último en la sección de tableros privados podremos importar o exportar tableros que tengamos en nuestro equipo.

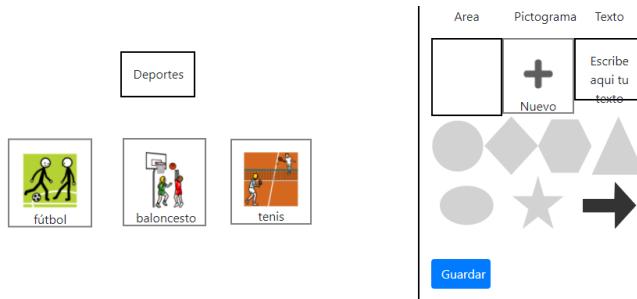


Figura 2.27: Ejemplo de creación de un tablero con un campo de texto y pictogramas.

2.3.6. Symbo Talk

Symbo Talk²² permite la creación de tableros de comunicación aumentativa y locución de tableros y pictogramas mediante su aplicación web o dispositivos móviles como Android e iOS.

SymboTalk ofrece dos modos de usuario:

- **Modo edición:** permite la creación de pictogramas, construir tableros y buscar pictogramas en un buscador. También ofrece la opción de crear un perfil y poder guardar todos los tableros que hayamos realizado.
- **Modo usuario:** pensado para que el usuario pueda comunicarse de una forma más fácil e intuitiva. Permite usar los pictogramas de la biblioteca o los que hayamos creado y reproducir por voz la secuencia de pictogramas seleccionada. En la Figura 2.28 podemos ver la pantalla principal de este modo. En el modo usuario permite editar el perfil para hacerlo lo más afín al usuario que lo utilice y que éste se sienta más cómodo a la hora de usar la aplicación.

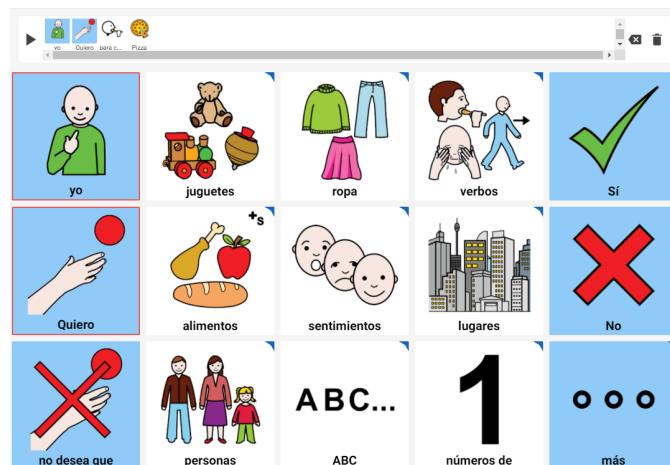


Figura 2.28: Pantalla principal de la aplicación Symbo Talk.

²²<https://civat.es/app/symbo-talk/>

2.3.7. LetMe Talk

LetMe Talk²³ es una aplicación para dispositivos Android e iOS que permite generar frases a partir de una lista de pictogramas seleccionados. Al ser descargada contiene una pequeña selección de pictogramas separadas por categorías (ver Figura 2.29) como “General”, “Comida” o “Juguetes”. Al pulsar en cualquiera de estos pictogramas, se desplegará otro tablero donde aparecen pictogramas con juguetes como “muñeca” o “pelota”. Ofrece un total de 9.000 pictogramas de ARASAAC y la posibilidad de añadir imágenes del móvil para ser incluidas como pictogramas con texto personalizado.



Figura 2.29: Menú de la aplicación en Android de LetMe Talk.

²³<https://www.letmetalk.info/es>

2.3.8. Jocomunico

Jocomunico²⁴ es una aplicación basada en el uso de pictogramas para ayudar a personas con dificultad en el habla. Su aplicación se puede usar tanto en su página web como en dispositivos Android e iOS y Windows y Mac OS X.

Jocomunico se creó con el objetivo de poder ayudar a los logopedas a trabajar con personas con dificultades comunicativas y poder facilitar las tareas de aprendizaje de los tiempos verbales, creación de distintos tipos de frases (preguntas, negaciones, etc) o estructurar de manera correctas los textos.

La aplicación cuenta con múltiples ajustes de accesibilidad que permiten la manera en la que se puede seleccionar un pictograma dependiendo del dispositivo ya sea de manera táctil, con uno o dos toques, o con el ratón del ordenador. También ofrece servicios de síntesis de voz para poder reproducir una secuencia de pictogramas creada por medio de los altavoces de nuestros dispositivos.

Una característica muy importante de esta aplicación es que aprende del uso que le da el usuario a los pictogramas y permite predecir pictogramas que le pueden ser útiles según está generando una frase, como se puede ver en el recuadro azul situado a la izquierda de la Figura 2.30.



Figura 2.30: Pantalla principal de la aplicación Jocomunico.

Comparativa de las herramientas analizadas

Tras haber analizado las aplicaciones mencionadas anteriormente se han recopilado aspectos que muchas de ellas tienen en común y funcionalidades que no tienen en la Tabla 2.1.

Podemos observar que muchas de ellas se pueden descargar a través de

²⁴<http://joanpahisa.com/es/jocomunico/>

Programas	Disponible	Se mantiene actualizado	Dispositivos	Permite editar pictogramas	Precio	Colocación de pictogramas
Pictoselector	Sí	Sí	PC y MAC	Sí	Gratis	No
Editor ARASAAC	Sí	Sí	PC, MAC, Android, iOS y Web	Sí	Gratis	No
Pikt oplus	No	No	Android	No	139€	No
Pictar	Sí	La web no, los pictogramas sí	Web	No	Gratis	Sí
Pictableros	Sí	La web no, los pictogramas sí	Web	Sí	Gratis	Sí
Jocomunico	Sí	Sí	Web	No	Gratis	No

Tabla 2.1: Tabla comparativa entre los distintos editores de tableros basados en pictogramas

la Play Store, App Store o utilizar su servicio desde su página web. Los más completos o que ofrecen más opciones, están disponibles para ordenadores aunque en dispositivos móviles hay mayor oferta de aplicaciones, generalmente ofrecen pocas opciones. Además de las aplicaciones de dispositivos, las más actualizadas son las que se pueden acceder en formato web.

Aunque a pesar de todo, el factor determinante es el precio. Las aplicaciones que no son gratuitas suelen tener un precio desorbitado o que muchas familias o docentes no pueden permitirse, por ello que una aplicación sea gratuita es determinante. Respecto a los idiomas, destacan el español y el inglés como idiomas predominantes entre las aplicaciones.

Muchas de estas aplicaciones no permiten la creación de tableros en los que poder colocar los pictogramas libremente ni insertar texto o iconos, simplemente tienen una cuadrícula donde se van insertando en orden los pictogramas seleccionados por el usuario. Un aspecto muy importante que se va a desarrollar en nuestra aplicación y que muy pocas ofrecen la posibilidad es permitir que el usuario pueda cargar sus propias imágenes y utilizarlas como si fueran pictogramas, con su imagen y texto correspondiente.

Como conclusión, creemos que la herramienta que desarrollemos debería cumplir las siguientes características. Debe ser gratuita, disponible desde navegador web, con edición de pictogramas, una base de datos de éstos actualizada y que ofrezca la posibilidad de insertar texto, figuras, iconos e imágenes.

Capítulo 3

Tecnologías

RESUMEN: En este capítulo se enumerarán las tecnologías utilizadas y su utilidad en el proyecto.

3.1. Introducción

Para llevar a cabo el proyecto, fue necesario investigar sobre tecnologías existentes que faciliten el desarrollo del mismo [COMPLETAR]

3.2. API Arasaac

ARASAAC facilita la obtención de recursos pictográficos mediante su API¹. Dicha API está exclusivamente disponible para aplicaciones no comerciales, tal y como indica su licencia Creative Commons. Mediante los métodos de la API , podemos obtener materiales y pictogramas. Los materiales son actividades, calendarios o agendas hechas con pictogramas. Respecto a los pictogramas, los métodos que hemos utilizado son.

- **BestSearch:** A partir de una palabra dada retorna el identificador y otros datos como el significado o la categoría del pictograma que más se asemeja al término buscado. Además, permite buscar palabras en multitud de idiomas.
- **searchText:** Igual que *BestSearch*, diferenciándose en que retorna todos los pictogramas en vez de uno solo.
- **idPictogram:** A partir del identificador obtenido anteriormente, retorna la dirección de la imagen del pictograma asociado al identificador.

¹<https://arasaac.org/developers/api>



Figura 3.1: Ejemplo de cómo se puede modificar el color de pelo y tono de piel de un pictograma entre otras opciones, en la web de ARASAAC

La API de ARASAAC está pensada para ser utilizada por múltiples aplicaciones en distintos idiomas por lo que ofrece la posibilidad, tanto en los métodos de búsqueda como los pictogramas en sí, de seleccionar qué idioma queremos usar por defecto. Otra característica a destacar, es que todos aquellos métodos en los que se retornen pictogramas podrán ser modificados desde la propia llamada a la API. Podremos modificar desde el color de fondo, rasgos físicos o tiempos verbales entre otras muchas opciones. Podemos ver un ejemplo de ello en la Figura 3.1 que muestra la edición del pictograma "Niño" dentro de la web de ARASAAC ²

3.3. React

React es una librería de Javascript pensada para desarrollar interfaces de usuario. Esta librería fue desarrollada por Facebook buscando la creación de interfaces de una manera sencilla y con un alto rendimiento. React permite desarrollar tanto aplicaciones web como aplicaciones móviles de una manera ordenada y con menos líneas de código respecto a otros lenguajes como Javascript puro. El hecho de que React permita estructurar los distintos componentes de una aplicación ayuda tanto al desarrollo como mantenimiento de esta. Otra característica importante de React es que ya tiene implementado muchas funcionalidades por lo que puede ayudar a reducir el tiempo de desarrollo de la aplicación, un ejemplo de esto es el hecho de tener las vistas asociadas a los datos. Esto permite que los datos mostrados se actualicen automáticamente sin necesidad de escribir nuevos fragmentos de código.

Respecto a otros Frameworks como Angular, vemos que React no ofrece todas las funcionalidades de un framework completo. Esto no supone ningún

²<https://arasaac.org/pictograms/es/7176/ni~no>

impedimento para desarrollar la aplicación con React pero habrá que adaptar nuestro código al ecosistema que ofrece.

3.4. Drag and Drop

React Drag and Drop³ Es una biblioteca que permite arrastrar componentes en React. Su principal uso en la aplicación, es la recolocación de los componentes del tablero de manera cuadriculada. Esto permite colocar con precisión los elementos en el tablero, lo cual era un requisito indispensable para poder hacer tableros organizados. De otra manera podrían quedar algunos componentes unos pocos píxeles por encima de otro, dando un resultado poco profesional.

3.5. JSZip

JSZip⁴ es una biblioteca de javascript que mediante una API permite crear y cargar archivos comprimidos en formato *ZIP*. Será el método para importar y exportar proyectos a la aplicación.

- **Generar ZIP:** Crea un zip con todo lo que haya creado el usuario, como las fotos que haya subido, la posición de los pictogramas colocados y su información asociada (Por ejemplo, si se ha cambiado la descripción de un pictograma)
- **Cargar ZIP:** Al subir un ZIP, vuelve al estado cuando fue generado.

³<https://react-dnd.github.io/react-dnd/about>

⁴<https://stuk.github.io/jszip/>

PictUp!

EN DESARROLLO: Éstá bastante desordenado, podríais indicarnos si está en el orden correcto, o qué podemos hacer para cohesionarlo.

El nombre es provisional, aunque suena bastante bien y ayuda a que no nos refiramos al proyecto como *El proyecto* o *Lo de los pictos*

RESUMEN:

En este capítulo se explicará el proceso de creación de la aplicación PictUp. Ésta se trata de una aplicación web de creación y edición de tableros pictográficos interactivos. El capítulo se dividirá en las distintas fases del desarrollo: 1. Estudio de requisitos

4.1. Introducción

El propósito de la aplicación es la de solventar los problemas que otras aplicaciones traían y aunar las herramientas necesarias para facilitar la creación de tableros pictográficos. Otro objetivo es la de ofrecer interacción a los tableros, al estar éstos en un medio digital.

PictUp está ideada para que sea útil tanto para padres o profesores que son los que crearán contenido pictográfico, como para las personas que puedan beneficiarse del uso de los tableros creados. Es por ello, que se pueden distinguir dos tipos de usuarios, el editor y el alumno.

Las características que de estos han sido obtenidas del trabajo futuro de otros TFG, comentarios de los usuarios que utilizaban estas aplicaciones y sugerencias de los tutores.

- **Crear tableros con precisión.**
- **Utilizar fotografías como material.**

- **Agilizar el proceso buscar pictogramas que sean usados de manera recurrente.**
- **Añadir interacción a los tableros:** Como hemos visto, muchas aplicaciones son rígidas, por lo que una vez creado el tablero no se puede hacer nada más, al añadir alguna clase de componente que permita no solo interactuar al alumno, sino que además el editor pueda dar distintos usos al componente para poder crear distintas actividades con el mismo elemento.

[COMPLETAR CON LA TABLA DE EL ESTADO DE LA CUESTIÓN]

4.2. Prototipos Tecnológicos

¿Esto iría aquí o complementa el capítulo de Tecnologías?

Durante las primeras etapas del desarrollo, creamos algunos prototipos para familiarizarnos con las tecnologías imprescindibles para crear la aplicación. Estas fueron el acceso a la API de ARASAAC¹ y desplazar elementos en una aplicación web.

4.2.1. API ARASAAC

Para probar el acceso a la API se creó un buscador de pictogramas independiente para comprender el funcionamiento de la API y las posibilidades que ésta ofrece. Aunque inicialmente simplemente muestra el pictograma de una palabra introducida. Pero más adelante en el desarrollo, explotamos la posibilidad de cambiar el color de pelo y tono de piel de los pictogramas que lo permiten.

4.2.2. React Drag and Drop:

Otro objetivo es el de la interacción de los componentes en una superficie cuadriculada. Tras experimentar con distintas bibliotecas de JavaScript como Interact JS² no obtuvimos el resultado esperado pues el desplazamiento de los objetos no era lo suficientemente preciso.

React tenía la biblioteca Drag and Drop que permite desplazar los componentes. La principal diferencia entre Drag and Drop y Interact JS, es que al mover un elemento, “Interact JS” deja unos píxeles de diferencia y “Drag and Drop.” permite mover objetos en intervalos definidos, como poder mover un objeto de 10 píxeles en 10 píxeles.

¹<https://arasaac.org/developers/api>

²<https://interactjs.io/>

4.3. Prototipo de papel interactivo

Después de probar distintas tecnologías y estudiar la situación actual de los tablero pictográficos, se procedió a bocetar la idea de la aplicación y el formato de la interfaz. Al ser dos personas, se realizaron dos bocetos diferentes, de los cuales se obtuvieron los primeros componentes y asentaron las bases del proyecto.

Habría que poner mejor Prototipo A y Prototipo B? No estamos seguros si es correcto poner nuestros nombres ¿Deberíamos editar las imágenes? Para facilitar su lectura A la hora de redactarlo ¿sería mejor estructurar como en la parte de Jorge (por las distintas pantallas) o mejor como esta puesto en la parte de Alfonso?

Prototipo realizado por Alfonso

Teniendo en cuenta, los requisitos de los usuarios, las aplicaciones existentes y el trabajo futuro de Pictableros empezamos a bocetar una primera idea del proyecto.

En la Figura 4.1 podemos ver el boceto de la pantalla de inicio, la cual estaría compuesta por cuatro secciones bien diferenciadas. **Creación de tablero libre** serviría para crear un tablero donde se pueden colocar pictogramas, texto o figuras. **Creación de actividad** añadiría interacción a los tableros mediante la sucesión de los mismos. Más tarde profundizaremos en esas dos secciones.

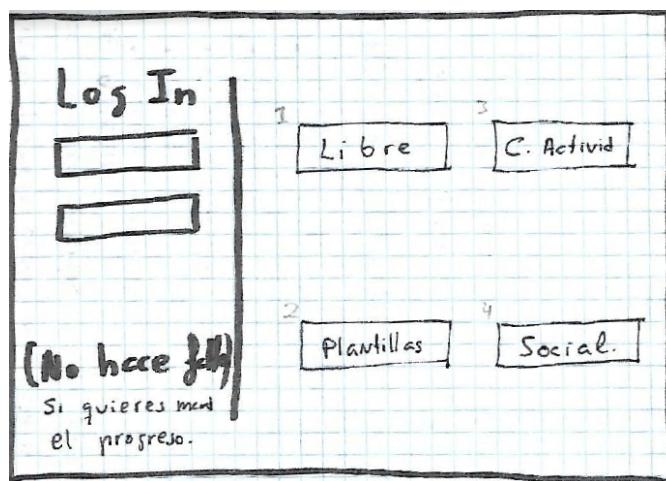


Figura 4.1: Boceto de pantalla inicial

Las **plantillas** permitirían crear un tablero mediante el uso de una plantilla que cuenta con espacios donde colocar pictogramas. Las plantillas facilitan la creación del material laborioso de crear.

Un ejemplo de ello es la creación de un horario, donde pueden haber decenas de huecos a llenar y se estructuran generalmente de la misma manera. Al tener una plantilla, el usuario se puede despreocupar de que los elementos queden bien centrados o añadir los días de la semana. En la Figura 4.2 podemos ver un ejemplo de algunas ideas de plantillas disponibles. La sección de Social tendría la función de compartir tableros, actividades y plantillas con la comunidad de usuarios.

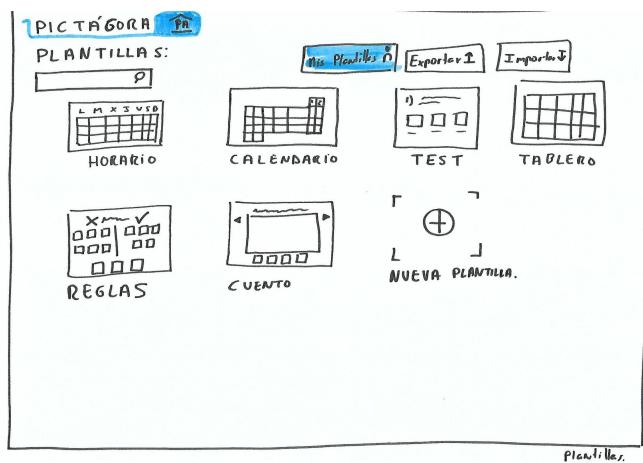


Figura 4.2: Boceto pantalla de plantillas

Por último, existe un inicio de sesión opcional para mantener el progreso entre dispositivos. Si profundizamos en la creación de un tablero en la Figura 4.3, podemos ver los componentes que se pueden colocar sobre el tablero. Éstos son los pictogramas, texto, figuras como flechas o rectángulos y la posibilidad de subir imágenes. También comenzamos a plantear la idea de almacenar colecciones de pictogramas, éstas almacenan distintos conjuntos de pictogramas que el usuario agrupa según su criterio. Por ejemplo, si un profesor tiene que crear varios tableros respecto a un tema, como "Animales de la granja", puede ser útil tener una colección con los pictogramas de gallina, cabra, oveja, etc agilizando el proceso de creación.

El apartado de creación de actividad no deja de ser una extensión de creación libre pero con más componentes que permiten interacción con el usuario para crear distintos tipos de actividad. Éstos componentes que a partir de ahora denominaremos como **componentes interactivos** son:

Cajón de pictos y espacio picto: Son dos componentes que van ligados, en primer lugar está el cajón de pictogramas que es un **espacio al margen del tablero donde aparecen un conjunto de pictogramas**, podemos ver cómo son los componentes en la Figura 4.4. Los elementos que aparecen en el cajón, pueden ser desplazadas a un **espacio picto** que es un hueco inicialmente vacío donde puede ser colocado un pictograma. Ambos

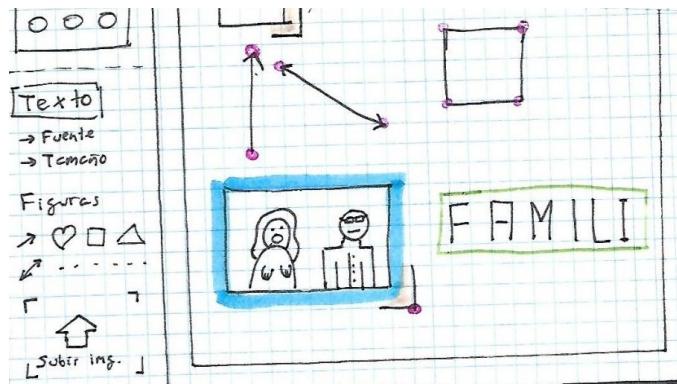


Figura 4.3: Boceto pantalla de edición de tableros

componentes son configurables, lo que permite establecer los pictogramas que aparecen en el cajón de pictogramas, y los pictogramas que acepta el espacio picto. Estos dos elementos se beneficiarían de las colecciones, pues son conjuntos ya hechos por el usuario.

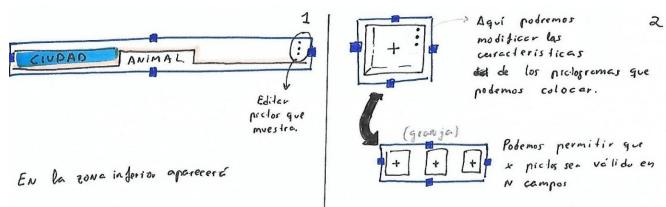


Figura 4.4: Boceto de las componentes canjón de picto y espacio picto.

Retomando el ejemplo de la granja, podríamos construir un ejercicio que permita al usuario mover los pictogramas según al espacio al que pertenezca un animal, como vemos en la Figura 4.5. En el tablero, aparecen representados algunos espacios picto en azul, que solo aceptan animales de la colección “*Granja*” y otro espacio para la colección “*Selva*”, dejando abajo el cajón de pictos con todos los animales mezclados. El objetivo, es dar la máxima flexibilidad al usuario que cree una actividad.

El componente **subtablero** permite desplegar un tablero de pictogramas. Los pictogramas que componen dicho tablero pueden venir dados por una colección de pictogramas creada por el usuario o indicarse en el propio componente. Su finalidad es la de añadir más pictogramas en el mismo espacio y agilizar la comunicación. La idea ha sido rescatada de Piktoplus que también permitía crear subtableros.

Por último el componente **enlace** que permite ligar dos pictogramas diferentes. Está compuesto por dos “piezas” las cuales se asignan a dos pictogramas para ser enlazados. Su finalidad es la de crear actividades como “hacer parejas”.

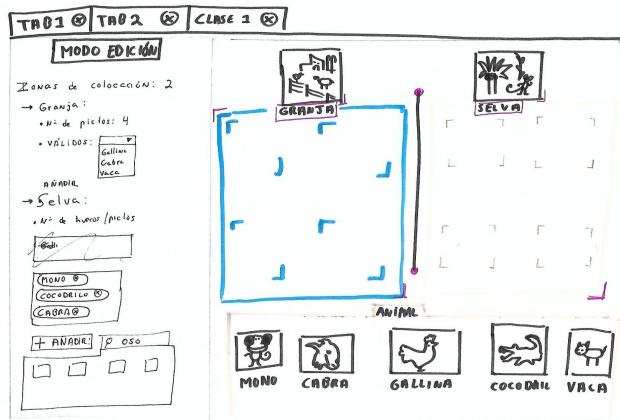


Figura 4.5: Boceto de actividad con cajón de pictos y hueco picto

En la Figura 4.6 podemos ver el ejemplo del pictograma hueso y perro, a los cuales se les asignan la misma pieza identificada por un símbolo de pica con fondo verde. El motivo por el que las piezas tienen una forma y color asociado facilita al usuario que cree la actividad identificar las piezas ya ligadas. El usuario final al pulsar sobre los pictos permitirá hacer parejas, y completar la actividad.

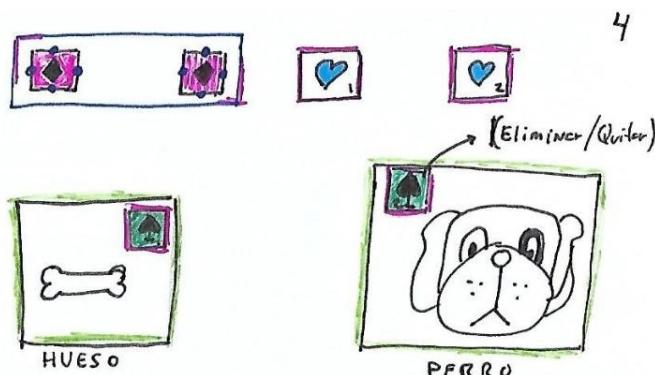


Figura 4.6: Boceto de las piezas que enlazan dos pictogramas al compartir la misma figura

Con todos los tableros creados también es posible crear una actividad de mayor tamaño secuenciando los tableros. Se podría pasar de uno a las siguientes escenas mediante flechas, como se puede ver en la Figura 4.7, que está compuesta por una fotografía y algunos pictogramas que la describen. Pero también se podrían intercalar estos tableros con otros que aporten interacción, por ejemplo podría añadirse un test mediante el cajón de pictos

para comprobar si se está comprendiendo la lectura.

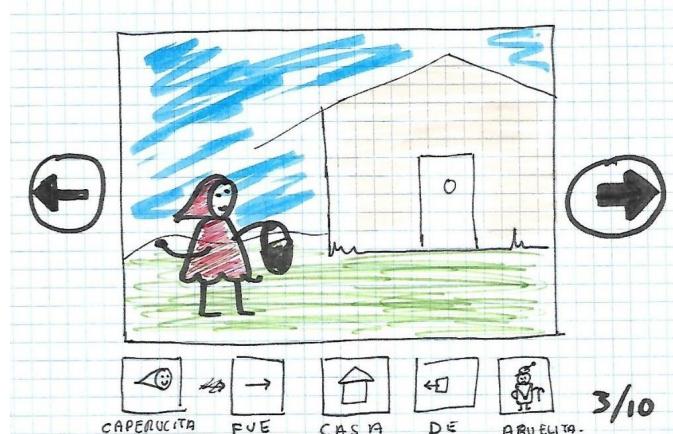


Figura 4.7: Boceto de una escena de un cuento, con pictogramas en la zona inferior y botones en los laterales para pasar o retroceder la escena o tablero

En la Figura 4.8 podemos ver cómo se compondrían este tipo de actividades, que resulta muy familiar a la construcción de una presentación de diapositivas.

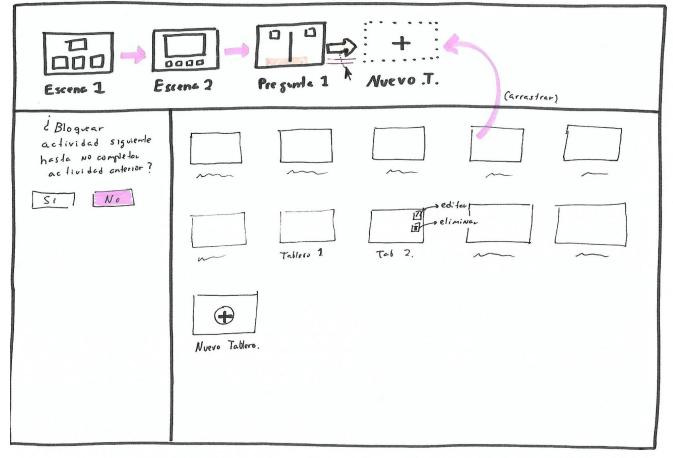


Figura 4.8: Boceto del compositor de actividades

Prototipo realizado por Jorge

■ Pantalla de inicio

En la Figura 4.9 podemos ver la pantalla principal que nos encontraríamos al entrar a la aplicación. Podemos distinguir 4 botones que cada

uno de ellos nos llevaría a su respectiva pantalla. También podemos ver que en la parte superior de la pantalla tendríamos el nombre de la aplicación a la izquierda, que al pulsarlo volveríamos a esta pantalla, y un selector de idioma en la parte de la derecha. Esta parte sería común en las distintas pantallas de la aplicación.



Figura 4.9: Pantalla de inicio con botones que muestran las otras pantallas de la aplicación.

■ Pantalla del tablero

En la Figura 4.10 podemos ver pantalla de los tableros. En ella podríamos diferenciar claramente dos zonas, la parte de la izquierda correspondiente a la personalización del tablero y la parte de la derecha donde se mostraría el tablero que se está editando.

En la parte de personalización del tablero podemos ver dos recuadros que englobarían diferentes posibilidades. Las funcionalidades que encontraríamos en el primer recuadro serían las siguientes:

- **Traducir frase:** dada una frase, mostraría toda la secuencia de pictogramas que tuviera ese significado. El botón de insertar que vemos insertaría toda la secuencia de pictogramas en el tablero.
- **Búsqueda simple de un pictograma:** dada una palabra concreta mostraría todos los pictogramas que tuvieran ese significado. Al igual que en el apartado anterior también tendríamos un botón de insertar el pictograma deseado al tablero.

En el segundo recuadro tendríamos las siguientes opciones:

- **Insertar iconos:** al pulsarlo mostraría un modal con todos los

iconos que se pueden añadir al tablero, al pulsar sobre uno de ellos se añadiría al tablero.

- **Insertar figuras geométricas:** al igual que con la funcionalidad anterior, al pulsarlo se abriría un modal donde se podrían seleccionar diferentes figuras geométricas para añadirlas al tablero.
- **Insertar imágenes:** esta herramienta permitiría al usuario insertar una fotografía que tuviera en su ordenador y a la hora de insertarla en el tablero tuviera las mismas propiedades que un pictograma, la imagen y un texto descriptivo.
- **Insertar un campo de texto:** permite añadir un campo en el tablero donde poner textos.
- **Editar el tamaño de letra del campo de texto:** esta funcionalidad solo estará disponible si hemos seleccionado un campo de texto en el tablero. Nos permite ajustar el tamaño del texto de un campo específico.
- **Editar la fuente del campo de texto:** al igual que con la funcionalidad anterior, deberemos seleccionar qué campo queremos editar. Permite cambiar la fuente del texto por otra.

En la zona inferior izquierda hay dos botones que permitirían guardar el estado de la página y volver a cargar el estado para posteriormente seguir trabajando con nuestro proyecto.

En la parte superior tendríamos un botón llamado “*Guardar como pdf*” que generaría un archivo con extensión pdf a partir del tablero inferior.

En la zona de la derecha encontramos un tablero donde se insertaría todos los pictogramas, iconos, formas, imágenes y campos de texto. Todos los elementos que se inserten en el tablero podrán ser ampliados de tamaño pulsando sobre una de las esquinas del elemento.

Debajo de dicho tablero encontramos un botón que nos permitiría añadir un nuevo tablero con el que seguir trabajando.

■ Pantalla de normas y cuentos

En la Figura 4.11 al igual que en la pantalla del tablero distinguimos dos zonas, la parte de la izquierda perteneciente a la edición y la parte de la derecha donde veríamos el visionado de las normas o el cuento.

También tenemos la posibilidad de guardar como pdf lo que tenemos en el tablero.

La parte de la derecha sería distinta respecto a la pantalla de tableros ya que aquí el tablero no está cuadriculado. Para añadir una norma o una nueva sección en nuestro tablero tendríamos que pulsar sobre el botón “*Nueva norma*” o “*Nueva sección*” y se añadirían dos campos.

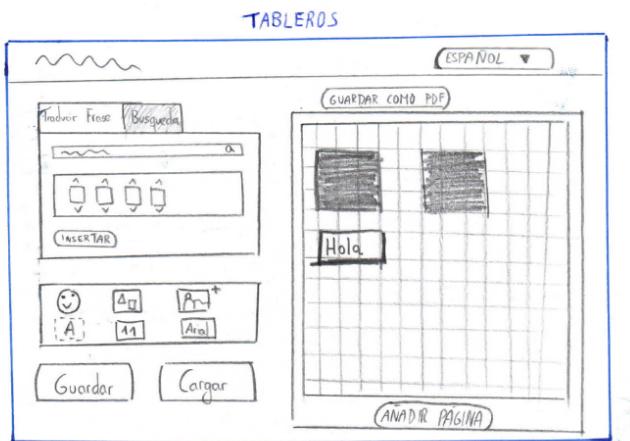


Figura 4.10: Pantalla de configuración de tablero.

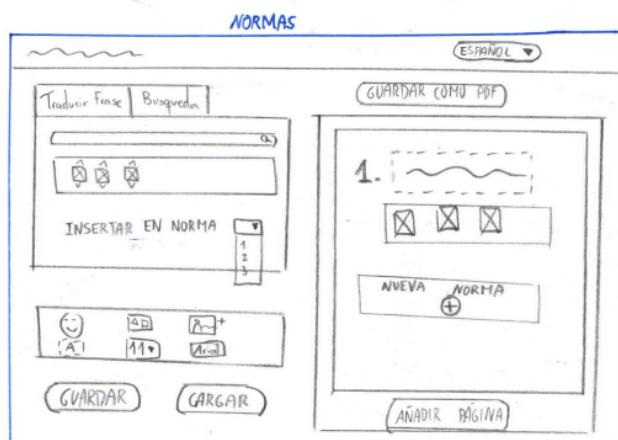


Figura 4.11: Pantalla de configuración de normas.

El primero de ellos estaría numerado y en él se insertaría el texto correspondiente a la norma o a la sección del cuento. El segundo sería un campo en donde poder insertar los pictogramas que queramos que hicieran alusión al campo de texto superior. Además hay un botón para añadir un nuevo tablero y seguir trabajando.

■ Pantalla de juegos

Para el apartado de juegos hay que distinguir dos pantallas, la primera de ellas sería la configuración del juego y la segunda sería la pantalla del juego.

- **Pantalla de configuración del juego**

Al igual que en las pantallas anteriores la parte de la izquierda, ver la Figura 4.12, es la que se utilizará de cara a la edición del tablero para buscar y añadir pictogramas, imágenes, iconos, etc.

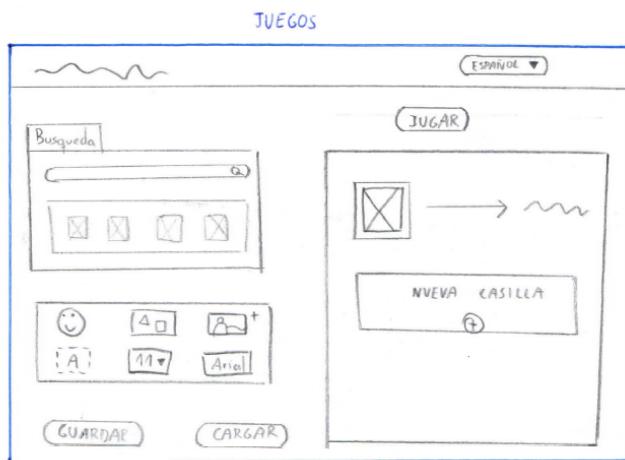


Figura 4.12: Pantalla de configuración de juego.

También encontramos dos botones para guardar el estado de la configuración del juego y otro para cargar dicho estado y reanudar el trabajo realizado.

Las funcionalidades que podemos encontrar en la parte del tablero son las siguientes:

- **Añadir una nueva casilla:** al pulsar sobre este botón se añadirá sobre el tablero sin cuadricular una casilla para insertar un pictograma, imagen, ícono o figura geométrica, una flecha y un campo de texto. Esto permitirá crear una asociación entre un pictograma y su texto correspondiente para posteriormente ejecutar el juego.
- **Jugar:** en la parte superior tendremos un botón que al pulsarlo ejecutará el juego con las normas que estén creadas. Este botón al pulsarlo nos llevará a la pantalla de juego, ver Figura 4.13.

- **Pantalla de ejecución del juego**

En esta pantalla, ver la Figura 4.13, podremos ver todos los pictogramas que se han seleccionado en la pantalla de configuración y los textos asociados a cada uno de ellos.

Todos los pictogramas seleccionados aparecerán en el recuadro superior de la pantalla. Estos pictogramas se podrán seleccionar y arrastrar a la casilla correspondiente de dicho pictograma.

Si al arrastrar un pictograma y soltarlo en una casilla coincide con la norma definida por el usuario la fecha que une la casilla del pictograma y el texto se pondrá en verde indicando que es correcta esa relación. En caso de que no corresponda el pictograma con el texto, el pictograma volverá a la parte superior donde se encuentran todos los pictogramas indicando de esta manera que la asociación entre el pictograma y el texto no es la correcta.

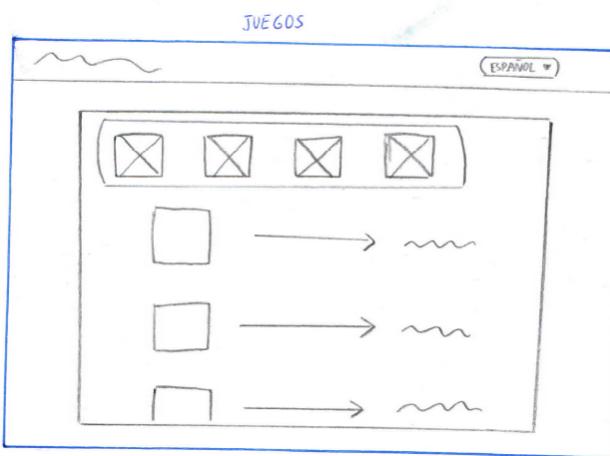


Figura 4.13: Pantalla de juego en ejecución.

4.4. Prototipado de la interfaz

4.5. Herramientas

En este apartado se incluirán todas aquellas funcionalidades que ayuden a la edición del tablero. Una de estas herramientas es la **búsqueda de pictogramas**, esta herramienta permitirá al usuario realizar una búsqueda a partir de una palabra e incluir el pictograma deseado en el tablero. También estará disponible la **herramienta de traducción de una frase** para facilitar al usuario la elaboración del tablero.

- **Colecciones:** la herramienta colecciones ofrecen al usuario la opción de poder tener varias agrupaciones de los pictogramas que desee, según su conveniencia. Éstas están compuestas por un nombre que lo identifique y uno o varios pictogramas que el usuario elija según su criterio.
- **Importar y exportar:** estas herramientas ayudarán al usuario a poder guardar el estado de la página y poder editarla posteriormente.

4.6. Componentes

En esta sección desarrollaremos los principales componentes que pueden colocarse sobre el tablero.

Un componente es **un elemento que puede ser añadido al tablero** e incluso añadir interacción para al usuario final. Distinguiremos dos tipos de componentes, los **básicos** que no añaden ninguna interacción y los **componentes interactivos**, que pueden ser modificados y añadir comportamientos específicos al usuario final.

- **Picto:** El elemento picto representa un pictograma junto a su nombre asociado. Cuenta con la opción de poder modificar algunas características del pictograma, como indicar un tiempo verbal, el plural. Si el pictograma muestra a una persona, también se puede cambiar el color de pelo y tono de piel.
- **Foto:** El elemento foto, permite añadir imágenes tanto a partir de una URL como las que suba el propio usuario. Esta ha sido una de las características más demandadas por los usuarios. Al permitir añadir fotos abre multitud de posibilidades, como la de poner fotos de la familia, mostrar localizaciones habituales como la cocina e identificar objetos personales que no se representan tan fielmente mediante un pictograma (Por ejemplo, un juguete específico o la portada de su libro favorito). Esto facilita al usuario final relacionar conceptos al mostrar figuras que le sean familiares.
- **Figuras:** Las figuras, sirven para ordenar, enfatizar o decorar el tablero. Un ejemplo, sería una línea que puede ser usada para dividir el espacio de trabajo en secciones, relacionar dos pictogramas o incluso marcar un espacio donde escribir una respuesta si se va a imprimir el tablero. Pese a la simplicidad de las figuras, sus posibilidades son muy amplias, según la creatividad de quien cree el tablero.
- **Campos de texto:** Los campos de texto podrán ser insertados en el tablero de la aplicación. Estos campos de texto podrán ser editados permitiendo cambiar el color del texto y aumentar la fuente.
- **Cajón de Pictogramas:** El cajón de pictos es un apartado al margen del tablero donde aparecen un conjunto de pictogramas que el usuario debe mover a alguna posición. En el hueco donde vayan los pictogramas que se encuentren en el cajón de pictogramas puede ser modificado y aceptar unos u otros. Esto puede ser utilizado como test sencillo de hacer y usar.
- **Subtablero:** El Subtablero es un componente que a simple vista parece un pictograma pero al ser pulsado, despliega un tablero que contiene

otros pictogramas. Este concepto la ha sido rescatada de Piktoplus, la cual actualmente no cuenta con soporte y puede ser de utilidad para añadir más pictogramas en el mismo espacio.

Respecto a los elementos **descartados**, a continuación se detallarán los elementos que fueron descartados y sus motivos:

- **Pantallas de normas y cuentos predefinidos:** El principal motivo es su falta de flexibilidad, es decir que si la aplicación no permite crear un cuento o un listado de reglas con sus propias herramientas, tampoco permitirá crear otro tipo de material. Por ello se ha decantado por herramientas que ofrecieran más posibilidades al usuario. Otro motivo era que no todos los usuarios querían por ejemplo que las normas aparecen de una misma manera, como si fuese un listado. Por lo que es imprescindible ofrecer la mayor libertad posible al usuario.
- **Plantillas:** Tras comentarlo en las reuniones, no se centró mucha atención en este apartado, pues ya estaba muy explorado por la aplicación Pictableros.
- **Log In:** El principal motivo de su descarte, fue el no poder garantizar en un primer momento la total seguridad de los tableros creados. Otra de las características del proyecto es la inclusión de imágenes que pudiera subir el usuario, con la responsabilidad añadida de haber imágenes de menores de edad por medio. Estos fueron los motivos por los que optamos por una modalidad sin necesidad de servidor, donde los documentos generados se guardan en el ordenador.
- **Selección de idioma:** Fue una idea descartada al principio del desarrollo, en vista a que la gran mayoría de las aplicaciones existentes ofrecían soporte en multitud de idiomas. Pero no se llevó a cabo para centrarnos en otros aspectos con mayor relevancia.

Capítulo 5

Arquitectura

RESUMEN: En este capítulo se especificará la estructura de la aplicación a nivel interno, desde el funcionamiento de la cuadrícula, las herramientas y el ítem que estas añaden al tablero y el esquema de conexiones con funcionalidades externas.

5.1. Cuadrícula e Ítems

En cuanto al funcionamiento de la cuadrícula y la cuadrícula fue clonado del proyecto dnd-a11y-patterns¹ que contaba con el material para trabajar con estos elementos. Cuenta con una licencia BSD 3-Clause que permite libertad de modificación y distribución.

Empezar desde este proyecto facilitó el tratamiento de eventos de desplazamiento (Drag and Drop), pues ya están integrados en estos componentes. Además de gestionar estos eventos, cuentan con muchos parámetros que han sido modificados.

La cuadrícula o canvas a es una cuadrícula, de la cual se puede modificar su tamaño, mediante la cantidad e intervalo. Siendo la cantidad, el número de filas y columnas que se representan. Y el intervalo, la distancia en píxeles entre cada fila y columna. Está ajustado para que abarque el mayor ancho y alto posible, dejando espacio para la barra de herramientas.

Los ítems son los elementos que se pueden colocar y desplazar por la cuadrícula, es sobre ellos donde se representarán los distintos elementos de la aplicación.

Los ítems, al ser una clase que hereda de la cuadrícula, toman los valores del intervalo para desplazarse en consecuencia. Por ejemplo, al aplicar

¹<https://github.com/salesforce-ux/dnd-a11y-patterns>

un evento de desplazamiento, el ítem se desplazará en función del tamaño del intervalo. Dando así una sensación de control al usuario, ya que al hacer movimientos cortos sabrá que el ítem se desplazará una casilla en una dirección. Este mismo principio se aplica al modificar el tamaño del ítem, modificando la anchura y altura en función del intervalo de la cuadrícula. El proyecto incluye dos botones dentro de los ítems que al ser pulsados, permite al usuario desplazar y redimensionar un ítem mediante las teclas direccionales. De esta manera el usuario tiene un control más preciso de la aplicación, por cada pulsación el ítem se mueve exactamente en una fila o columna de la cuadrícula.

Otro ajuste muy útil es el parámetro que permite mantener la proporción de un ítem. Esto ha servido para que la acción de modificar el tamaño de un ítem no resulte frustrante, pues al modificar varios ítems algunos acababan con un ancho o alto diferente al resto.

Retomando con la sección anterior, la información de los pictogramas añadidos se guarda en una lista que es posteriormente renderizada en la cuadrícula en forma de ítem. Al existir distintos elementos que se pueden colocar (Pictos, fotos, figuras, texto...), cada uno cuenta con su propio ítem que los representa en la cuadrícula (PictoItem, FotoItem, FigutrItem...).

Cada ítem cuenta con un renderizado y funcionalidades adecuadas para cada elemento. La mayoría de estos ítems tienen en común los botones vistos anteriormente de desplazar y redimensionar, a los cuales se ha añadido otro botón en la esquina superior derecha para eliminar el pictograma de la cuadrícula. Cuando se pulsa, se envía al componente padre, quien gestiona los elementos que aparecen en la cuadrícula el identificador del ítem a eliminar. Al recibirla, se quita de la lista donde está guardado el elemento y desaparece de la cuadrícula.

Visto cómo funciona a nivel interno la aplicación, podemos pasar a ver el funcionamiento de las distintas herramientas, el ítem que genera en la cuadrícula y las posibilidades que ofrecen.

A continuación se expondrá el funcionamiento de cada herramienta y el ítem que genera en la cuadrícula.

5.2. Buscador de pictogramas

Ésta herramienta muestra los distintos pictogramas que se asocian a una palabra mediante la API de *ARASAAC*.

Como se puede ver en la Figura 5.1 la componente no sólo es un buscador, además tiene un historial de pictogramas añadidos y la opción de preajuste de pictogramas. Dos utilidades muy convenientes de cara a añadir pictogramas al tablero.

Para empezar se va a explicar cómo está estructurado el componente de



Figura 5.1: Vista del componente de búsqueda de pictogramas

búsqueda de pictogramas. En el estado del componente se almacenará:

- La palabra de la que se quiere obtener sus pictogramas
- Los pictogramas encontrados por la consulta
- El preajuste de los pictos, donde se especifican cómo se van a mostrar los pictogramas.
- Flags de control: Para saber si se ha completado la búsqueda, está abierto el modal o si se ha seleccionado el historial.

Este componente abarca el acceso a la API de búsqueda de pictos, gestión de eventos y los métodos de renderización. En este caso hay dos, el primero que podemos ver en la Figura 5.1 muestra la barra de búsqueda y sus opciones. El otro método de renderización, muestra el resultado de la consulta, debajo del espacio de búsqueda como se puede ver en la misma figura.

A continuación, analizaremos las distintas características de este componente:

5.2.1. Búsqueda

Al pulsar sobre el ícono de la lupa o pulsar intro se lanza la consulta a la API mediante Fetch o get. La URL para hacer la petición tiene dos parámetros, el idioma que por defecto es español y el término a buscar. Tras finalizar la búsqueda, se guarda la respuesta de la API para posteriormente procesarla.

Los pictogramas ya buscados, pasan a ser renderizados. Aunque antes de ello es importante explicar el contenido de un pictograma de ARASAAC. De entre todos los parámetros que incluye, destacar:

- **_id:** Número único que lo identifica, imprescindible para cargar la imagen.
- **Keyword:** La palabra asociada al pictograma.
- **hasLocution:** Indica si cuenta con locución. La locución está grabada por una persona y no se trata de una voz robótica típica de un “text to speech”.
- **hair:** Indica si el pictograma tiene pelo, en caso afirmativo se puede cambiar su color.
- **skin:** Indica si el pictograma es una persona, en caso afirmativo se podrá cambiar el tono de piel.

Existen otros muchos parámetros, pero son éstos con los que se han trabajado a lo largo de la aplicación.

En la sección Preajuste 5.2.2 se explica en detalle cómo se obtiene y modifica la url. Con la url creada, se mostrarán los pictogramas devueltos por la API. Cada pictograma cuenta con dos botones, uno para añadirlo a la cuadrícula y otro para guardarlos en una lista de pictos. Las listas de pictos se verán en la Sección Lista de pictos 5.5.

En caso de añadirlo a la cuadrícula, se guardará en el historial de pictos y enviará la información de éste al componente padre. Éste recibiría el el pictograma que ha devuelto la API de ARASAAC junto a sus parámetros y el preajuste aplicado a dicho picto. Antes de ver el funcionamiento del nuevo ítem que aparecerá en la cuadrícula, es importante conocer el funcionamiento del preajuste e historial.

5.2.2. Preajuste

La manera de obtener la dirección URL de una imagen tiene dos maneras de abordarse. En un primer lugar, podría poner simplemente la url “<https://api.arasaac.org/api/pictograms/>” junto al identificador del pictograma. De esta manera simplemente se mostraría el pictograma pero perderíamos la valiosa posibilidad de editar un pictograma que ofrece ARASAAC.

Para poder optar a las modificaciones, se ha de modificar la URL de una manera concreta. Desde la propia API se especifican los atributos modificables que se pueden aplicar a todos los pictogramas:

- **action:** Añade una flecha para indicar que una acción se da en el futuro o en el pasado.
- **plural:** Añade un “+” en la esquina superior derecha.
- **nocolor:** Muestra el pictograma en blanco y negro.

Existen otras dos modificaciones que solo son aplicables a los pictogramas cuyos parámetros “hair” y “skin” estén marcados como “true“.

- **hair:** Si un pictograma tiene pelo su color puede ser uno de entre 6 colores posibles asociado a su valor hexadecimal. Moreno, rubio, pelirrojo, etc.
- **skin:** Si el pictograma contiene una persona se puede modificar el tono de piel de entre 5 posibles, especificados mediante su valor hexadecimal.

Es aquí donde entra en acción el preajuste de pictogramas, ofreciendo al usuario una interfaz donde modificar la apariencia de los pictogramas que se buscan y posteriormente se añadan. Al pulsar sobre el botón [Dónde está] se abre el modal que se ve en la Figura 5.2. El modal en sí, se trata de otra componente que recibe como parámetro de entrada los preajustes ya establecidos, y devuelve los nuevos preajustes. Los preajustes son los atributos vistos anteriormente: color de pelo, tono de piel, color y plural.

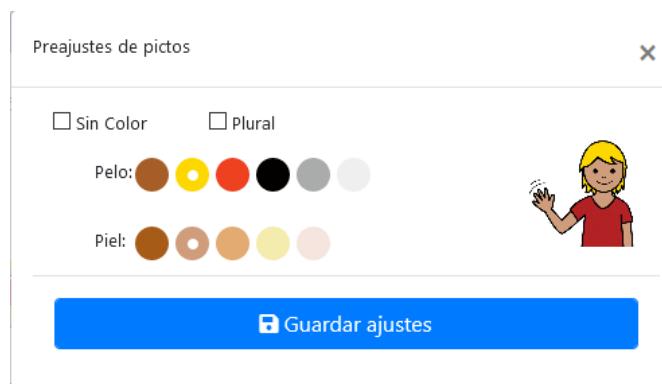


Figura 5.2: Vista del modal del preajuste de pictos con las opciones color rubio y piel morena marcada.

La interfaz del modal consiste en dos filas. La primera para los checkbox que marcan las opciones de plural y color, las cuales pueden estar activadas o desactivadas. La segunda fila son para elegir una de entras distintas opciones de color disponibles tanto para el tono de piel como para el color de pelo.

Está dispuesto de esa manera ya que si se marca algún checkbox, los botones radiales desaparecen. Esto se debe a que no es posible tener por ejemplo, un pictograma sin color y color de pelo rubio. Previiniendo así combinaciones que no estén disponibles.

A la izquierda cuenta con una pre visualización para que el usuario conozca cómo afectan los atributos seleccionados al pictograma. Una vez el usuario esté conforme con los cambios, se envían al componente búsqueda de pictogramas las opciones seleccionadas.

Respecto a la construcción de las URLs se realiza de la siguiente manera.

1. A partir de la url: https://static.arasaac.org/pictograms/id/id + {options} + _500.png. En la variable *options* se añadirán las opciones que modifiquen el pictograma.
2. Dichas opciones deben estar en un orden concreto utilizando los valores obtenidos del preajuste y el pictograma a crear. Por ejemplo, si se va a crear la url del pictograma “avion”, los ajustes relacionados con el color de pelo o piel no se aplicarían, ya que el picto no lo permite y por tanto se saltaría.
3. Un resultado posible sería: “_action-future_hair-020100_skin-A65C17” Brevemente, cada etiqueta que se añade significa “_action-future” que añade la flecha en la esquina superior derecha indicando futuro, “hair” y “skin” van asociados a su color hexadecimal correspondiente obtenido del preajuste. Si la etiqueta de “hair” fuera antes que la de “skin”, la url creada no funciona. De ahí la importancia del orden.
4. La URL creada se usa en cada pictograma mostrado en función de los preajustes, tal y como se puede observar en la Figura 5.2.

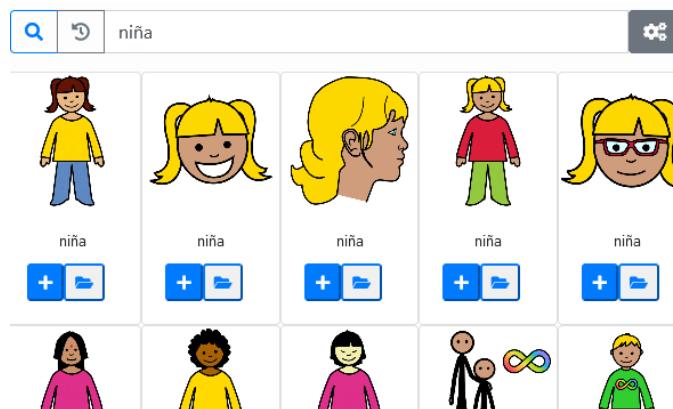


Figura 5.3: Vista de la interfaz de búsqueda de pictogramas al aplicar un preajuste. En este caso piel morena y pelo rubio

5.2.3. Historial de pictos

Al añadir un pictograma al tablero, dicho pictograma se guarda en el historial que se mantiene entre sesiones. Para ello se ha utilizado LocalStorage², el cual almacena una lista en la variable “recentPictos”. Dicha lista está compuesta por el objeto devuelto por la API de ARASAAC parseado a

²<https://developer.mozilla.org/en-US/docs/Web/API/Window/localStorage>

string, ya que los valores almacenados en LocalStorage solo pueden estar en formato UTF-16 DOMString que utiliza dos bytes por carácter.

Al pulsar sobre el botón de historial se cargarán y mostrarán todos los pictogramas encontrados en “recentPictos”. En caso que el historial esté vacío, se mostrará un mensaje indicándolo. Al igual que la búsqueda, los pictogramas del historial se podrán añadir al tablero o a una lista, y son susceptibles del preajuste de pictogramas.

Es importante destacar que al realizar las pruebas no se puso un límite de pictogramas que se pudieran guardar en el historial. A partir de 300 aparecían pérdidas de rendimiento llegando a detener momentáneamente la página, motivo por el cual se redujo a 50.

5.2.4. PictoItem

Al añadir el ítem a la cuadrícula, tiene un aspecto similar al de la Figura 5.2. Éste ítem ha recibido como parámetros de entrada, el objeto de pictograma devuelto por la API de ARASAAC y el preajuste con el que ha sido añadido. En el caso de la Figura 5.4, ha sido el pictograma “hola” colocado con preajuste de color de pelo rubio y piel morena.

A continuación se especificarán todas las posibilidades que ofrece un PictoItem colocado en el tablero.

Editar Picto

Tiene un funcionamiento igual al de preajuste de picto aunque se han añadido dos opciones más. Tal y como se puede ver en la Figura 5.5, éstas son el selector de tiempo verbal y borde. El motivo de aparición de estas dos opciones en este modal y no en el preajuste es la de no abrumar al usuario. El Borde puede ser de utilidad para los usuarios que quieran resaltar el tipo de pictograma mediante un color, tal y como se vio en capítulos anteriores (Estado de la cuestión). El selector de tiempo verbal sirve para indicar si una acción toma lugar en presente, pasado o futuro.

Reproducir sonido

La función del botón es la de reproducir mediante audio el nombre del pictograma. Como se ha visto anteriormente, los pictogramas devueltos por la API pueden contar con locución si se consulta el parámetro “*hasLocution*”. En caso de contar con locución, se accede a la URL³ que contiene la pista de voz asociada y se reproduce. En caso de no contar con locución se ha im-

³<https://privateapi.arasaac.org/api/locutions/es/>



Figura 5.4: Vista de un PictoItem en la cuadrícula con preajuste de pelo y piel.

plementado mediante *SpeechSynthesisUtterance*⁴ para que haga una función similar en el resto de pictogramas. Está configurada para que sintetize la voz en español y es compatible con todos los navegadores modernos.

5.3. Traducir frase a pictogramas

La herramienta de traducción de frase a pictograma, se encuentra en la segunda pestaña que agrupa las herramientas para añadir pictogramas e imágenes. Cuenta con un ítem propio llamado FraseItem, que agrupa los pictogramas de la traducción para ser desplazados con facilidad por la cuadrícula.

La traducción de frase a pictogramas, como su nombre indica ofrece una

⁴<https://developer.mozilla.org/en-US/docs/Web/API/SpeechSynthesisUtterance>

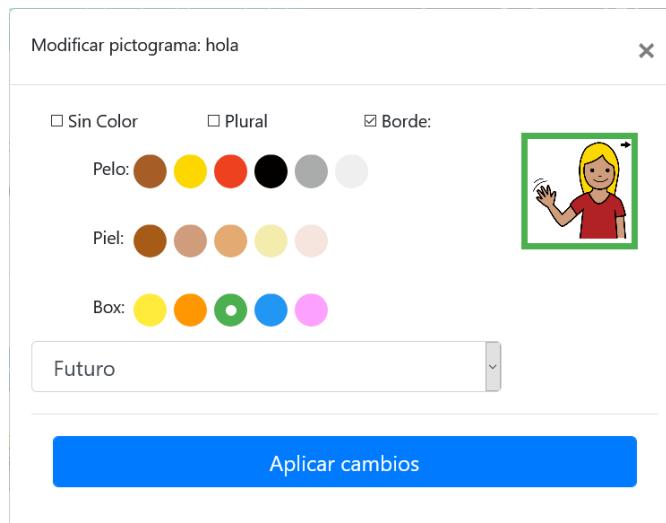


Figura 5.5: Vista de la interfaz del modal que muestra una edición de pictogramas con más opciones. En este caso, se ha seleccionado el tiempo futuro y el borde verde.

interfaz que permite al usuario escribir una frase y recibir la traducción en pictogramas. En la Figura 5.6 se puede ver la interfaz inicial de la traducción a frase, muy similar a la vista en la búsqueda de picto en la Figura 5.1.

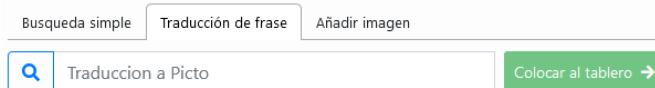


Figura 5.6: Interfaz inicial Traducción frase a pictograma.

La traducción es realizada mediante la API de *ARASAAC* o *NIL Group*.

- **API de ARASAAC:** Su API no cuenta con una traducción dedicada a frases, por ello se traduce el texto palabra a palabra de la frase escrita. Por cada palabra devolverá entre 0 o más pictogramas.
- **NIL WS API:** Cuenta con una función dedicada a la traducción de frases a pictogramas. La traducción resulta mucho más fiel a nivel sintáctico que la planteada mediante la API de ARASAAC. Las características de NIL WS API será vista más adelante en la Sección 5.3.2.

El resultado de la traducción no está fijado a un pictograma concreto por palabra, sino que puede haber varios. Es por ello que ambas APIs devuelven de un modo u otro varios pictogramas posibles por cada palabra de la

frase. En la Figura 5.6 se puede ver cómo la interfaz permite navegar entre los pictogramas alternativos mediante dos botones. Estos botones permiten avanzar y retroceder entre los pictogramas alternativos que se muestran por cada palabra. Por último destacar que el botón “Colocar al tablero” no se activa hasta que no hay una frase traducida, como se puede ver en las figuras 5.6 y 5.7.

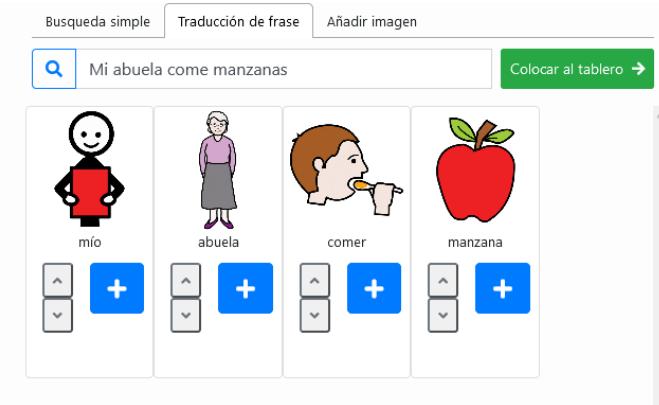


Figura 5.7: Resultado de la traducción de una frase

Antes de añadir la frase a la cuadrícula, es posible añadir los pictogramas en forma individual. En caso que se pulse sobre “Colocar al tablero”, se colocarán en el tablero todos los pictogramas juntos en un único elemento `FraseItem`.

5.3.1. `FraseItem`

Como se puede ver en la Figura 5.8, el ítem agrupa todos los pictogramas recibidos de la traducción. De esta manera el usuario para desplazarlo por el tablero no tiene que mover cada pictograma de manera individual. Este ítem, cuenta con la posibilidad de ocultar pictogramas ya que el usuario podría no querer que se muestre alguno. Para ello se ha de presionar en el botón de edición con el icono del lápiz.



Figura 5.8: Vista de `FraseItem` en la cuadrícula

Al ser presionado aparecerá un nuevo modal como se ve en la Figura

5.9. Está compuesto por una previsualización de la frase, los pictogramas que componen la frase junto al botón que permite esconderlo o mostrarlo. Al presionar alguno de ellos, se actualiza la previsualización. En la parte inferior cuenta con un campo de texto para modificar la frase que se muestra en el ítem. En la Figura 5.10 se puede ver un ejemplo de cómo quedaría el ítem tras aplicar los cambios.

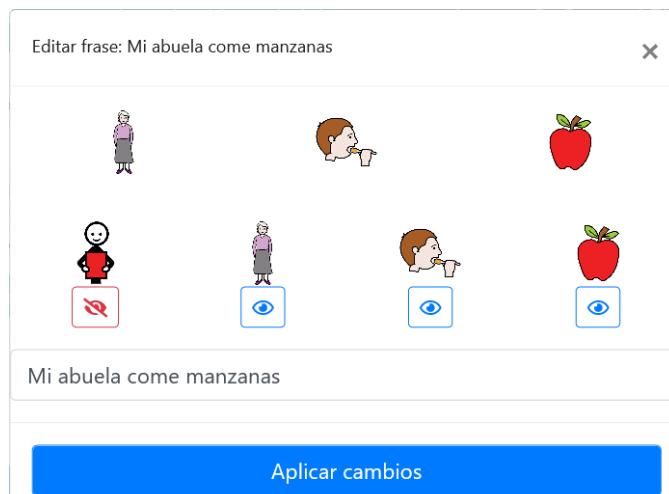


Figura 5.9: Vista del modal de edición del FraseItem



Figura 5.10: FraseItem tras aplicar los cambios

5.3.2. NIL WS Group

NIL Word Search⁵ es una API que devuelve información relativa a palabras y textos. Respecto al tratamiento de palabras, permite buscar sinónimos, antónimos, emoción, etc. Para el tratamiento de textos, permite traducir un texto a pictogramas, listar las emociones de un texto o resumir.

⁵<https://holstein.fdi.ucm.es/nil-ws-api/>

Originalmente la traducción de pictogramas en esta aplicación fue creada específicamente para trabajar con esta API. No obstante también ha causado varios inconvenientes, relativos a los pictogramas que usa la API y los errores causados por CORS.

La API cuenta con una base de datos propia con una gran cantidad de pictogramas. Pero al traducir un texto a pictogramas. Ésta devuelve por cada palabra o lema, una lista de identificadores. En su mayoría dichos identificadores coinciden con los de los pictogramas de ARASAAC, pero existen otros que no. Estos pictogramas que no comparten identificador con los de ARASAAC por lo que no se puede obtener toda la información deseada, como por ejemplo el texto, audio y modificaciones. Es por ello que PictoItem se modificó para incluir estos pictogramas suprimiendo la posibilidad de personalizar o reproducir el audio.

5.3.3. CORS

Cross Origin Resource Sharing⁶, es un mecanismo de cabeceras http que permite a un servidor dar permiso a otros dominios para cargar recursos del mismo. Por razones de seguridad los navegadores restringen las peticiones de dominio cruzado desde los scripts del navegador.

Como el proyecto de la web sigue operativo pero descontinuado, no cuenta con la configuración CORS y por tanto no puede recibir peticiones POST de otras webs desde el frontend. Después de comprender el problema se estudió que la solución es crear un backend para la aplicación. Dicho backend está creado exclusivamente para resolver esta petición. El componente de traducción envía la petición al backend, el cual la resolverá gestionando los CORS. En la próxima Sección 5.3.3 se explicará cómo fue creado.

Después de crear y configurar el backend de manera local y ver que el componente funcionaba sin complicaciones apareció otro error con la biblioteca html2canvas, encargada de crear una imagen a partir de la cuadrícula y sus ítems. Debido a que las imágenes que retorna la API de NIL son un recurso externo y la biblioteca html2canvas desconoce su fuente. Por ello al descargar la imagen del tablero estos pictogramas no aparecían representados dejando un hueco en blanco.

Backend

El backend fue creado mediante Express⁷. Se trata de un framework de Node.js que permite gestionar llamadas a API de manera sencilla. Express cuenta con una dependencia que gestiona los CORS⁸, la cual fue usada para

⁶<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

⁷<http://expressjs.com/en/starter/installing.html>

⁸<https://www.npmjs.com/package/cors>

realizar el método post a la API de NIL.

Para esquematizar cómo se obtienen los pictogramas de la API de NIL, se puede ver en la Figura 5.11 un diagrama de las llamadas que se realizan entre las aplicaciones.

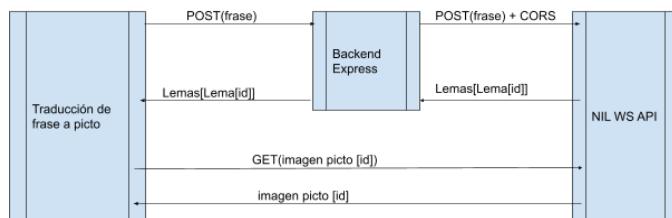


Figura 5.11: Diagrama de conexiones entre aplicaciones para traducir una frase

Después de varias semanas investigando todos estos problemas, finalmente se detuvo el desarrollo de este componente ya que la fecha para mostrar la web a los usuarios se acercaba. Al tener un prototipo funcional de traducción mediante API de ARASAAC, asumiendo la baja calidad sintáctica de ésta serviría para estudiar la facilidad de uso de la interfaz por parte de los usuarios.

5.4. Añadir imagen al tablero

Añadir una imagen al tablero se encuentra en la tercera pestaña que agrupa todas las herramientas que añaden pictogramas o fotos a la cuadrícula. Como se puede ver en la Figura 5.12, su interfaz inicial consiste en un simple botón que carga una imagen del explorador de archivos del dispositivo. Está especificado que solo acepte archivos de tipo imagen, previniendo así posibles errores del usuario.

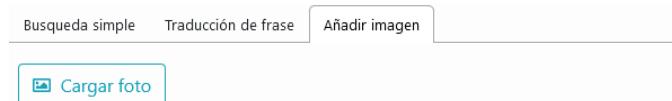


Figura 5.12: Interfaz inicial al añadir una imagen al tablero

Al cargar la imagen correctamente, tal y como se puede ver en la Figura 5.13 se mostrará:

- Una previsualización de la imagen.
- El campo de texto para añadirlo debajo de la imagen.
- Botón para añadir la imagen a la cuadrícula.

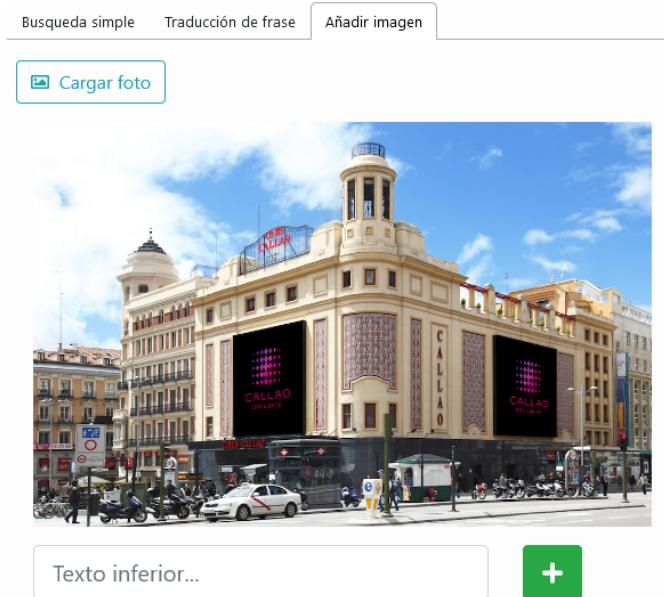


Figura 5.13: Vista de la Interfaz al cargar una imagen

Al presionar el botón de añadir, antes de enviar la información de la foto, se obtiene su valor de alto y ancho en píxeles. Estos datos serán utilizados para conocer la proporción de la imagen. Respecto al valor de la foto se guarda en una URL que funciona únicamente en la sesión vigente mediante “URL.createObjectURL”⁹.

Al enviar la información de la foto a la capa superior se envían los siguientes parámetros:

- URL de la imagen.
- Ancho y alto en píxeles.
- Texto inferior, en caso de no haber sido escrito nada se enviará vacío.

Al tratar con imágenes y éstas al poder contener material muy sensible, como fotografías personales. Es por este motivo que todos los archivos que pueda cargar el usuario son tratados en el lado del cliente.

5.4.1. PhotoItem

Su representación en el tablero es la imagen dentro del ítem. El ancho y alto del ítem es proporcional a la imagen original, como se calculó anteriormente.

⁹<https://developer.mozilla.org/en-US/docs/Web/API/URL/createObjectURL>

Respecto al texto inferior, se puede ver en la Figura 5.14 puede aparecer o no. Si se ha recibido un texto inferior, éste aparecerá en un espacio reservado para él debajo de la imagen. En caso contrario la imagen abarcará la totalidad del ítem.

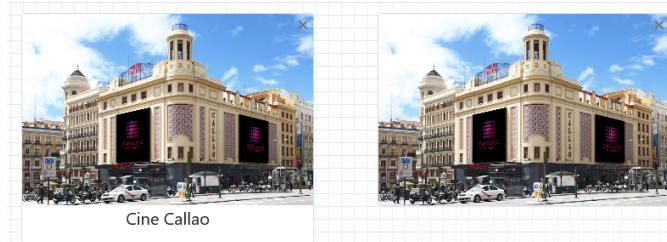


Figura 5.14: Ejemplod de PhotoItem con y sin texto

En la Figura 5.15 se ejemplifica qué pasaría si en vez de calcular la proporción de la imagen, se dejara siempre con un ancho y alto fijo apareciendo espacios en blanco.

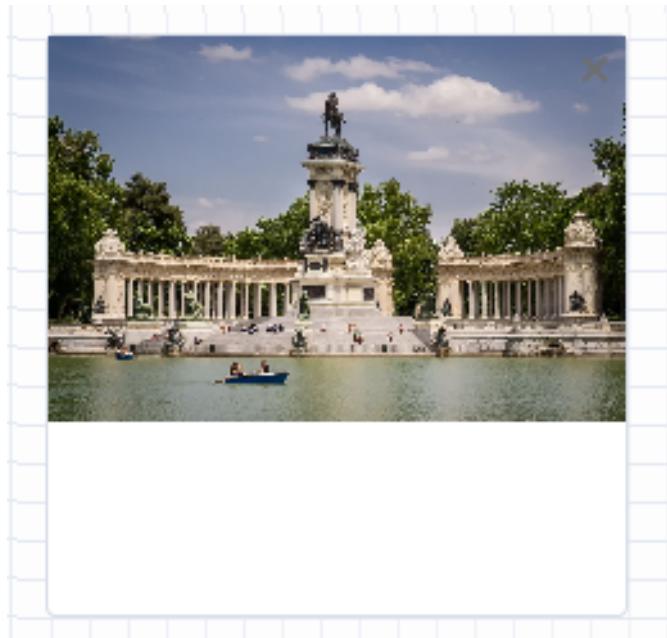


Figura 5.15: Ejemmplo de PhotoItemn si no mantuviera la proporción

5.4.2. Opciones investigadas y descartadas

Originalmente hubo dos aproximaciones hacia cómo cargar imágenes. La primera consistía en poder cargar un archivo comprimido zip y mediante la

librería JSZip¹⁰ que contaba con funciones para descomprimir archivos. El objetivo era automatizar la colocación de imágenes sobre el tablero, mediante un zip que almacenara éstas junto a un documento con la información de cada una. Dicho documento incluiría el nombre, la posición y el tamaño de cada imagen colocada en la cuadrícula. Descomprimiendo dicho zip, las imágenes volverían a ser colocadas a la posición original.

La otra aproximación ofrecía la posibilidad de conectarse mediante la API de Google Drive¹¹ para que el usuario pudiese acceder a sus fotos almacenadas en su cuenta.

Pese a haber dedicado varias semanas a cada una de estas opciones en proyectos independientes, fueron abandonadas por no obtener resultados satisfactorios. En el caso de JSZip, aunque la compresión de las fotos y descarga del zip funcionara correctamente, surgieron problemas con la librería a la hora de descomprimir los archivos. Respecto a la API de Google Drive, debido a la falta de documentación sobre cómo usarla en React, apenas se logró un prototipo funcional en el tiempo establecido.

Hubo una última opción que se planteó. Realizar un historial de imágenes cargadas muy similar al historial de pictogramas visto anteriormente. Sin embargo, LocalStorage apenas permitía unos pocos megabytes de almacenamiento. La alternativa a LocalStorage fue indexedDB.

IndexedDB¹² es una API que permite almacenar información en el cliente, en este caso el ordenador del usuario. En ella se pueden crear bases de datos con la capacidad de almacenar archivos de gran tamaño como en este caso imágenes y fotos. Al final fue descartado por falta de tiempo.

5.5. Listas de pictogramas

Esta herramienta permite crear distintas listas donde el usuario podrá añadir los pictogramas que deseé. Para implementar esta herramienta se necesita tener una estructura donde se guarde el nombre de cada una de las listas y todos los elementos PictoItem que contenga.

5.5.1. Añadir y crear a una lista

Para poder crear una lista el usuario deberá buscar el pictograma y pulsar el ícono de la carpeta del que más le guste. Tras pulsar el ícono se desplegará un modal donde dependiendo del estado de las listas mostrará un menú u otro. En el caso de que no haya ninguna lista simplemente se mostrará la opción de crear una lista donde se deberá introducir el nombre de la lista a crear, mientras que si ya había alguna lista creada se mostrará un menú con

¹⁰<https://stuk.github.io/jszip/>

¹¹<https://developers.google.com/drive/api/v3/about-sdk>

¹²https://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API

la posibilidad de añadir el pictograma seleccionado a una lista existente o crear una nueva lista. Un ejemplo de esto lo podemos ver en la Figura 5.16 donde el modal de la izquierda aparece cuando no existe ninguna lista y el de la derecha cuando ya existe al menos una lista. Además la lista mostrada en el modal de la izquierda será la última lista modificada.

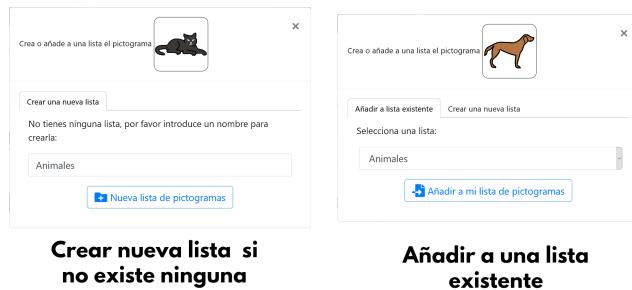


Figura 5.16: Vistas del modal con ninguna lista y al menos una, respectivamente

Para informar al usuario de las acciones tanto la de crear una nueva lista, cómo añadir un pictograma a una lista existente se han utilizado alertas (mensajes). Estas a parte de dar una cierta información al usuario también sirven para tener un cierto control de errores para que el usuario no cree dos listas con el mismo nombre o que no añada un pictograma dos veces a una lista.

- En el caso de que se haya podido realizar la acción con éxito se mostrará un pequeño mensaje descriptivo de la acción en verde, ver la Figura 5.17.

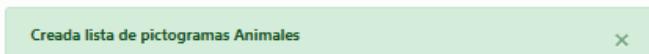


Figura 5.17: Alerta tras añadir satisfactoriamente un pictograma a la lista

- En el caso de que la acción no se haya podido realizar correctamente se informará al usuario con un mensaje en rojo, como el que se ve en la Figura 5.18, informando al usuario el porqué no ha podido realizarla.



Figura 5.18: Alerta de acción no permitida

5.5.2. Listas de Pictogramas creadas

Otras funcionalidades que podemos encontrar en esta herramienta es la posibilidad de importar y exportar las listas creadas. Para la funcionalidad de exportar guardaremos el estado actual de las listas en un fichero con extensión “.json”. Para la generación de este documento se utilizará la función `JSON.stringify()`, la cual nos permitirá crear la estructura propia de los json y posteriormente generar el documento con ese contenido.

La funcionalidad de importar cargará el estado de las listas a partir del archivo creado anteriormente. Para ello se hará uso de la función `JSON.parse()` la cual convertirá el json a la estructura de listas requerida. En caso de ya existir listas en la aplicación, las nuevas listas cargadas se añadirán a las ya existentes.

Por último, las listas existentes se encuentran en el apartado “Mis listas de pictogramas”, como se muestra en la Figura 5.19. Para visualizar una de ellas deberemos seleccionar el nombre de esta de entre todas las disponibles. A continuación, al igual que en el apartado de búsqueda de pictograma, se mostrarán todos los pictogramas con su imagen, texto correspondiente y el botón de “+” que permite añadirlo a la cuadrícula como un PictoItem visto anteriormente.



Figura 5.19: Vista de las listas de pictogramas

5.6. Texto

Esta herramienta permite añadir un cuadro de texto a la cuadrícula. Como opciones de personalización de texto se ofrece la posibilidad al usuario de seleccionar una tipografía entre las seis posibles. Las tipografías elegidas son las más utilizadas dentro del ámbito educativo.

Esta herramienta se puede encontrar en el apartado de “Personalización de tablero”, donde se podrán seleccionar distintas tipografías y escribir el texto que se va a añadir a la cuadrícula.

Como se puede ver en la Figura 5.20 en la parte superior se puede elegir una tipografía y a su derecha ver una previsualización de ésta. Debajo, está el campo de texto donde se escribirá la frase seguido del botón para añadirla a la cuadrícula.

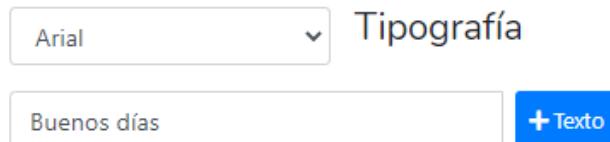


Figura 5.20: Vista de la selección de fuente y campo de texto

Para poder implementar este componente, en su estructura se guardará el texto que se vaya a añadir a la cuadrícula y la tipografía seleccionada.

A diferencia de otros elementos como los pictogramas en los que el ancho y alto del ítem era proporcional, en el texto se permite ajustar ambas propiedades de manera independiente.

5.6.1. TextoItem

Al añadir el texto a la cuadrícula, el ancho del ítem se ajustará al número de caracteres del texto. En la parte inferior del ítem como se ve en la Figura 5.21 se encuentran tres botones, dos para reducir y aumentar el tamaño de la fuente del texto y otro para editar su contenido.

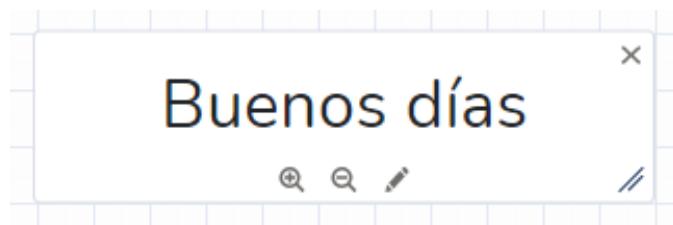


Figura 5.21: Representación del TextoItem en la cuadrícula

Al pulsar sobre el ícono del lápiz se desplegará el modal de edición de ítem. Como se ve en la Figura 5.22 cuenta con un campo donde se puede ver y editar el contenido actual del texto. Esto resultará muy útil para corregir errores ortográficos. Tras pulsar sobre el botón “Cambiar texto”, el ancho del ítem se ajustará de nuevo en función del nuevo número de caracteres.

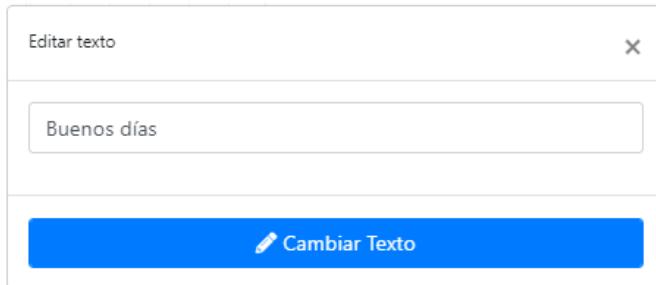


Figura 5.22: Vista del modal para editar un TextoItem

5.7. Iconos

La componente iconos permite añadir a la cuadrícula iconos y personalizarlos. Los iconos disponibles son: cuadrado, tick, barra y corazón, como se muestra en la Figura 5.23. La componente está formada por una estructura donde se guarda el tipo de ícono, el color y la opacidad. Al ser añadidos a la cuadrícula tiene un color por defecto en función del tipo de ícono, verde para el tick y rojo para la barra y el corazón.



Figura 5.23: Vista de los botones que añaden iconos

A la hora de mostrar los iconos en la cuadrícula hay que hacer una distinción entre el cuadrado y el resto de los iconos ya que la forma en la que se muestran es distinta.

Para generar un cuadrado se hace uso de las formas geométricas básicas que ya están implementadas en SVG. Este lenguaje de marcas permite crear figuras geométricas y personalizar su aspecto. Para representar el cuadrado se ha de incluir la etiqueta `<rect/>` (etiqueta para generar un cuadrado) y añadir las propiedades que queremos que tenga como por ejemplo altura, ancho, grosor de los bordes del cuadrado, color u opacidad.

Sin embargo, para el resto de los iconos se utilizan recursos obtenidos de FontAwesome¹³. Este framework tiene implementado iconos vectoriales

¹³<https://fontawesome.com/>

que al igual que con el cuadrado, permiten la personalización de los iconos respecto a el color, tamaño y opacidad. Para mostrarlos se deberá incluir la clase correspondiente de cada ícono especificada en el apartado tipo de la estructura.

En la Figura 5.24 se puede ver que estos íconos se pueden superponer a los pictogramas para realzar ideas.

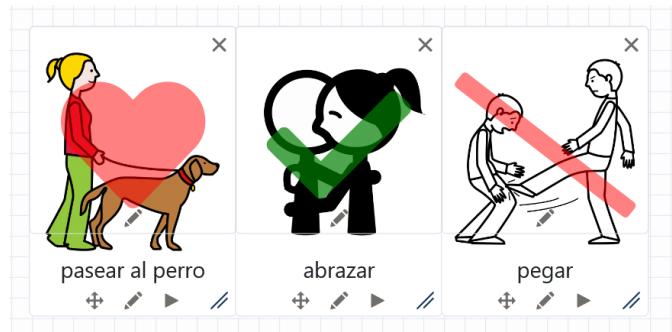


Figura 5.24: Ejemplo de uso de íconos sobre algunos pictogramas

Aunque la forma de visualizarlos se hace de manera distinta dependiendo del ícono todos ellos permiten editar ciertos aspectos, es por ello por lo que se incluyó la funcionalidad de editar. Para poder editar un ícono añadido a la cuadrícula tendremos que pulsar sobre el ícono del lápiz, al pulsarlo aparecerá un modal, como el de la Figura 5.25 donde se podrá modificar el color y la opacidad del ícono.

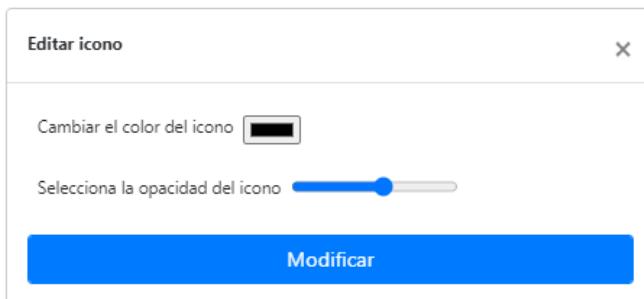


Figura 5.25: Vista del modal de edición de íconos.

5.8. Descargar Tablero

Esta herramienta permite descargar la cuadrícula como imagen. Para ello se utiliza la librería html2tocanvas¹⁴. Ésta librería además permite seleccio-

¹⁴<https://html2canvas.hertzen.com/>

nar la calidad de descarga de la imagen. Por ello que se ha creado un modal como se puede ver en la Figura 5.26 donde el usuario puede elegir la calidad deseada, siendo la calidad alta la elegida por defecto.

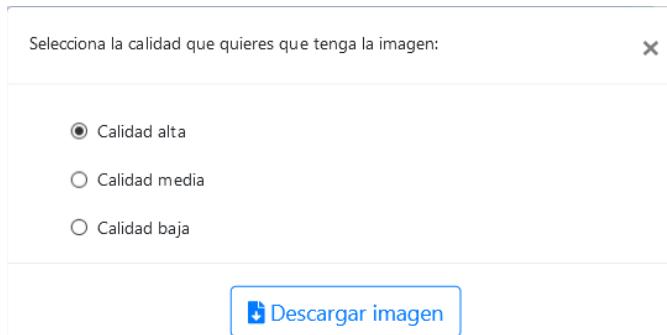


Figura 5.26: Vista del modal para descargar tablero

Capítulo 6

Evaluación

RESUMEN: En este capítulo se explicará el proceso de evaluación de la aplicación: Empezando por la preparación (Sección 6.1), el diseño de la evaluación (Sección 6.2), resultados obtenidos (Sección 6.3) y el análisis de dichos resultados (Sección 6.4).

Al tener un prototipo funcional con la suficiente antelación a la entrega, se puso en marcha el plan de realizar una evaluación con usuarios. Durante este mes se focalizó el trabajo en mejorar la usabilidad e interfaz de la aplicación, diseñar la evaluación y en dar a conocer la aplicación al público objetivo de la aplicación. Éstos son usuarios creadores de material pictográfico, generalmente profesores de educación especial y padres y tutores.

6.1. Preparación de la evaluación

Inicialmente se buscaron sitios donde se subiera material pictográfico con frecuencia para así localizar a los creadores de dicho material y contactar con ellos. La propia web de ARASAAC cuenta con una sección dedicada a la subida de material pictográfico donde se añaden nuevo material diariamente. La inmensa mayoría de este material contaba con una marca de agua indicando el perfil de Instagram del creador. Por ello nos decantamos por **crear una cuenta de Instagram¹ para crear una red de potenciales usuarios.**

En dicha cuenta, se contactó con muchos de estos creadores de material para informarles sobre el desarrollo de la aplicación. También se creó **material en forma de videos** donde se explicaban muchas funcionalidades de la web. Los vídeos contaron con una recepción positiva y de interés por

¹<https://www.instagram.com/pictupweb/>

parte de los usuarios, por lo que en el transcurso del mes se logró un grupo considerable de usuarios.

6.2. Diseño de la evaluación

Es en este punto donde la situación causada por la situación de pandemia tuvo mayor impacto sobre el proyecto. En otras circunstancias la evaluación hubiera sido en un centro escolar de manera más cercana al usuario final, es por ello que se tuvo que plantear una **evaluación a distancia y asíncrona**. **El objetivo de esta evaluación es medir la usabilidad y utilidad de la aplicación**.

Debido a estas circunstancias, la evaluación se realizó mediante un formulario para que el usuario pudiera completarlo por su cuenta. Por contraposición también estaba la posibilidad que el usuario dejara el formulario incompleto en caso de no lograr completar alguna tarea. Otro problema es el dispositivo de evaluación, ya que la aplicación está diseñada y probada para ordenadores. Por ello antes y durante la realización del formulario se aconsejó encarecidamente el uso del ordenador para el desarrollo de la evaluación.

Primero se plantean algunas **preguntas para conocer la relación de los usuarios respecto al uso de pictogramas**, la edad y ocupación. De esta manera se podría determinar el tipo de perfil de usuario que está evaluando.

Tras conocer el perfil, se plantean una serie de **tareas a realizar** dentro de la aplicación. Antes de plantear las tareas, se vuelve a recomendar al usuario el uso de ordenador para probar la web. Respecto al posible escenario en el cual el usuario no supiera continuar con la tarea y dejara el formulario incompleto, fue solucionado mediante **pistas**. Las pistas son imágenes autoexplicativas donde se especifican las acciones que se han de realizar para completar cada tarea. Como se puede ver en la Figura 6.1, se trata de la pista que indica cómo buscar un pictograma.

En el formulario las pistas vienen dadas mediante una pregunta que se podría realizar el usuario de cara a completar una tarea, seguido de una URL que redirige a la imagen de la pista asociada. De esta manera el usuario puede resolver una duda concreta y **no se desvela cómo completar otras acciones dentro de la tarea**. Otra manera de orientar al usuario a saber si ha realizado una tarea correctamente fue mediante imágenes que mostraran un posible estado final del tablero o componente al completar la tarea.

Respecto a las tareas, aunque no se abarcan todas las posibilidades, representa una muestra significativa de las funcionalidades disponibles en la aplicación. Esto se debe a que el formulario debe de completarse en un período razonable de tiempo y no extenderse más de lo debido. A continuación se detallarán las tareas escogidas y el motivo de elección:

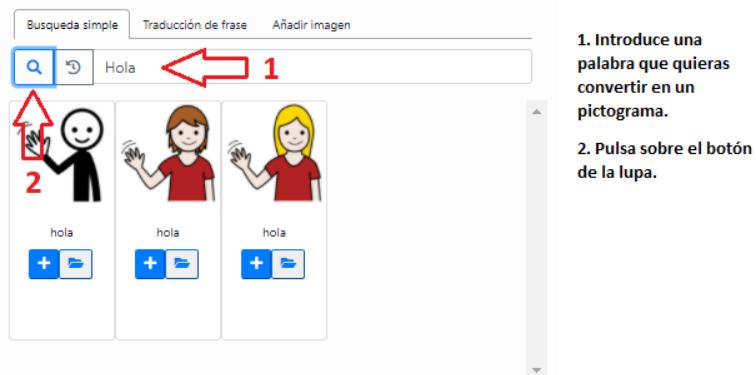


Figura 6.1: Ejemplo de pista en el formulario: Buscar un pictograma.

- **Buscar un pictograma, añadirlo al tablero y editarlo:** Se trata de la funcionalidad más básica de la aplicación.
- **Crear una lista de pictos, añadir un pictograma a una lista de pictos y añadir un pictograma de la lista al tablero:** Una funcionalidad que ha causado dudas durante el desarrollo ya que se desconocía si era lo suficientemente claro su funcionamiento.
- **Traducir una frase, añadirla al tablero y ocultar un pictograma de la frase:** Al ser una funcionalidad más compleja y más opciones era deseable conocer el desempeño del usuario con ésta.
- **Añadir fotos, figuras e iconos al tablero:** Abarcan funciones independientes más simples comparadas con las anteriores, ya que requieren menos acciones para completarlas.

El tipo de cuestionario seleccionado para medir la usabilidad de cada tarea ha sido el **ASQ²** (After Scenario Cuestionare), el cual permite al usuario evaluar la dificultad de cada una. Este cuestionario ha de ser respondido una vez finalizada cada tarea y está compuesta por tres afirmaciones. El usuario ha de indicar cuán de acuerdo está mediante una escala de 1 a 7, siendo el 1 muy en desacuerdo y el 7 muy de acuerdo. Dichas afirmaciones son:

1. En general, estoy satisfecho con la facilidad para completar la tarea en este escenario.
2. En general, estoy satisfecho con la cantidad de tiempo que tomó completar la tarea en este escenario.
3. Estoy satisfecho con la respuesta de la web al realizar una acción, sé lo que pasa en todo momento

²<https://help.qualaroo.com/hc/en-us/articles/360039070552-After-Scenario-Questionnaire-ASQ->

Esta última afirmación fue modificada para ser ajustada al contexto de la aplicación, enfocándose más al feedback que pueda aportar la tarea. Originalmente ésta era: *En general, estoy satisfecho con la información de soporte (ayuda en línea, mensajes y documentación) al completar la tarea*

Tras completar las tareas, se ofrece al usuario a que siga probando la aplicación con libertad para probar el resto de funcionalidades que no aparecieran en las tareas. También se dejan un hueco de respuesta libre en función si se ha sentido perdido o ha echado en falta alguna función durante la realización de las tareas.

Por último, de manera opcional se propone al usuario otro formulario donde puede dar su opinión sobre la aplicación de manera global. Para ello se ha utilizado las preguntas de la escala SUS³ (System Usability Scale), que mide la usabilidad del sistema. Nuevamente plantea una serie de afirmaciones, en este caso 10 donde el usuario deberá elegir en una escala de 1 a 5 cuán de acuerdo está con ellas. Dichas afirmaciones son:

1. Creo que usaría esta web frecuentemente
2. Encontré la web innecesariamente compleja
3. Creo que la web es fácil de usar
4. Creo que necesitaría la ayuda de una persona con conocimientos técnicos para usar la web
5. Las funciones de la web están bien integradas
6. Creo que la web es muy confusa
7. Creo que la mayoría de la gente aprendería a usar la web muy rápidamente
8. Encuentro la web muy complicada de utilizar
9. Me siento confiado/a al utilizar la web
10. Necesito aprender muchas cosas antes de poder utilizar la web

Una vez acabado el cuestionario, éste fue enviado, tras dos semanas, se obtuvieron más de 35 respuestas por parte de estudiantes y profesores.

Continuará...

6.3. Resultados de la evaluación

6.4. Análisis de los resultados

³<https://help.qualaroo.com/hc/en-us/articles/360039474571-System-Usability-Scale-SUS->

Capítulo **7**

Conclusiones y Trabajo Futuro

Conclusiones del trabajo y líneas de trabajo futuro.

Chapter 7

Conclusions and Future Work

Conclusions and future lines of work.

Apéndice A

Título

Contenido del apéndice

Apéndice **B**

Título

*-¿Qué te parece desto, Sancho? – Dijo Don Quijote –
Bien podrán los encantadores quitarme la ventura,
pero el esfuerzo y el ánimo, será imposible.*

*Segunda parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes*

*-Buena está – dijo Sancho –; fírmela vuestra merced.
–No es menester firmarla – dijo Don Quijote–,
sino solamente poner mi rúbrica.*

*Primera parte del Ingenioso Caballero
Don Quijote de la Mancha
Miguel de Cervantes*

