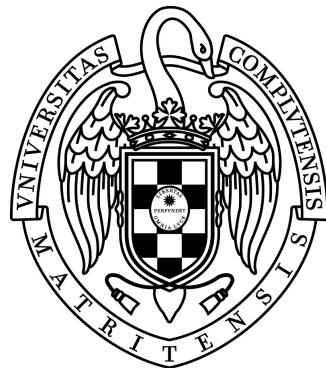

AdaptaMaterialEscolar2.0: Mejorando una herramienta para la adaptación de asignaturas a necesidades educativas especiales

AdaptaMaterialEscolar2.0: Improving a tool for adapting subjects to special educational needs



Trabajo de Fin de Grado
Curso 2022–2023

Autores

Álvaro Gómez Sittima, Dunia Namour Doughani, Alberto Alejandro Rivas Fernández, Johan Sebastian Salvatierra Gutierrez

Directoras

Virginia Francisco Gilmartín
Raquel Hervás Ballesteros

Trabajo de Fin de Grado en Ingeniería del Software

Facultad de Informática

Universidad Complutense de Madrid

AdaptaMaterialEscolar2.0:
Mejorando una herramienta para la
adaptación de asignaturas a
necesidades educativas especiales

AdaptaMaterialEscolar2.0:
Improving a tool for adapting
subjects to special educational
needs

Trabajo de Fin de Grado en Ingeniería del Software

Autores

**Álvaro Gómez Sittima, Dunia Namour Doughani, Alberto
Alejandro Rivas Fernández, Johan Sebastian Salvatierra
Gutierrez**

Directoras

**Virginia Francisco Gilmartín
Raquel Hervás Ballesteros**

Trabajo de Fin de Grado en Ingeniería del Software

**Facultad de Informática
Universidad Complutense de Madrid**

9 de noviembre de 2022

Autorización de difusión

Los abajo firmante, matriculados en el Grado en Ingeniería del Software de la Facultad de Informática, autorizan 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: AdaptaMaterialEscolar2.0, realizado durante el cuarto curso académico bajo la dirección de Virginia Francisco Gilmartín y Raquel Hervás Ballesteros en el Departamento de Ingeniería del Software e Inteligencia Artificial, 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.

Alberto Alejandro Rivas Fernández, Álvaro Gómez Sittima, Dunia Namour Doughani, Johan Sebastian Salvatierra Gutierrez

9 de noviembre de 2022

Dedicatoria

Texto de la 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. Introducción	1
1.1. Motivación	1
1.2. Objetivos	2
1.3. Estructura del proyecto	2
2. Introduction	5
3. Estado del arte	7
3.1. Adaptación curricular	7
3.1.1. Adaptaciones de acceso al currículum	8
3.1.2. Adaptaciones curriculares individualizadas	10
3.1.3. Ejemplificación de algunas adaptaciones curriculares asociadas a diferentes necesidades	11
3.2. Herramientas existentes para adaptaciones curriculares	13
3.2.1. SymWriter	13
3.2.2. EducaPlay	13
3.2.3. AdaptaMaterialEscolar 1.0	14
4. Metodología	19
4.1. Metodología de desarrollo	19
4.1.1. Tablero Kanban	19
4.1.2. Políticas explícitas	20
4.1.3. Clases de servicios	20
4.2. Plan de Pruebas	21
4.2.1. Pruebas Unitarias	21
4.2.2. Pruebas de integración	22
4.2.3. Integración continua	23
5. AdaptaMaterialEscolar 2.0	25

5.1. Requisitos	25
5.2. Diseño de la aplicación	28
6. Conclusiones y Trabajo Futuro	37
7. Conclusions and Future Work	39
8. Trabajo Individual	41
8.1. Álvaro Gómez Sittima	41
8.2. Dunia Namour Doughani	41
8.3. Alberto Alejandro Rivas Fernandez	41
8.4. Johan Sebastian Salvatierra Gutierrez	42
Bibliografía	43

Índice de figuras

3.1. Interfaz de SymWriter	14
3.2. Actividad de EducaPlay	14
3.3. Impresión de actividad de EducaPlay	15
3.4. Subir PDF	16
3.5. Busquedad de pictogramas	16
3.6. Ejercicios de completar huecos	16
3.7. Ejercicios de definiciones	16
3.8. Ejercicios de desarrollo	17
3.9. Sopas de letras	17
3.10. Ejercicios de Verdadero o Falso	17
5.1. Fuente escolar.	25
5.2. Diseño 1 Álvaro Gómez Sittima iteración competitiva.	29
5.3. Diseño 2 Álvaro Gómez Sittima iteración competitiva	30
5.4. Diseño Dunia Namour Doughani iteración competitiva	31
5.5. Diseño 1 Alberto Alejandro Rivas Fernández iteración competitiva.	32
5.6. Diseño 2 Alberto Alejandro Rivas Fernández iteración competitiva.	33
5.7. Diseño 3 Alberto Alejandro Rivas Fernández iteración competitiva.	33
5.8. Diseño 3 Johan Sebastian Salvatierra Gutierrez iteración competitiva.	34
5.9. Diseño 3 Johan Sebastian Salvatierra Gutierrez iteración competitiva.	34
5.10. Diseño final aplicación.	35

Índice de tablas

Capítulo 1

Introducción

En este capítulo se explica la introducción del Trabajo de Fin de Grado que va ser presentado en este documento. Primero en la Sección 1.1 se explicará la motivación que ha dado lugar al trabajo. A continuación, en la Sección 1.2 los objetivos que se pretende alcanzar. Por último, la estructura del proyecto final se detalla en la Sección 1.3.

1.1. Motivación

La educación escolar tiene como objetivo promover el desarrollo de ciertas habilidades, y el aprendizaje de varios contenidos necesarios para que los estudiantes se conviertan en miembros activos de la sociedad. Para ello, la escuela debe dar respuestas educativas que eviten la discriminación y promuevan la igualdad de oportunidades.

En el currículo escolar, todos los alumnos tienen necesidades educativas comunes. Sin embargo, no todos los estudiantes se enfrentan con las mismas capacidades de aprendizaje, sino que cada alumno tiene necesidades individuales. La mayoría de estas se abordan a través de acciones simples: dar a los alumnos más tiempo para aprender determinados contenidos, diseñar actividades complementarias, etc. Sin embargo, también existen necesidades individuales que no se pueden atacar por estos medios, lo que precisa una serie de medidas didácticas especiales, diferentes de las normalmente requeridas para la mayoría de los estudiantes. Dichas necesidades se pueden satisfacer con las adaptaciones curriculares. Existen dos tipos de adaptaciones curriculares:

- Adaptación no significativa: Adaptaciones en la metodología, las actividades, los tiempos, las técnicas e instrumentos de evaluación. No modifican los contenidos del currículo.
- Adaptación significativa: Ajustes significativos en el currículo, es decir

se eliminan apartados del currículum oficial.

Las adaptaciones curriculares no significativas para los ACNEE (Alumnos con Necesidades Educativas Especiales) deben ser realizadas por los profesores, sin embargo, no se les facilita una herramienta para ello a pesar de ser un trabajo muy costoso, ya que requiere la adaptación personalizada de materiales, pruebas de evaluación, etc. Por esa razón nació AdaptaMaterialEscolar 1.0, una herramienta para facilitar la adaptación curricular no significativa y ahorrar tiempo a los docentes.

1.2. Objetivos

La finalidad de este TFG es proporcionar una herramienta al profesorado que permite la adaptación curricular no significativa de los contenidos de las asignaturas de forma intuitiva, rápida y simple, con el objetivo de hacer materiales, que se trabajen en el aula, adaptados a las distintas necesidades de los alumnos.

Partimos de la aplicación AdaptaMaterialEscolar 1.0 que permite la creación de diferentes tipos de ejercicios (por ejemplo, sopas de letras, rellenar espacios en blanco, etc). Para poder ayudar al profesorado, estudiaremos las aplicaciones similares y las funcionalidades, rediseñaremos tanto la interfaz de la aplicación como la arquitectura, implementaremos las funcionalidades que les podrían ser útiles y por último se evaluará la aplicación.

Además, seguiremos un Diseño Centrado en el Usuario (DCU) con el fin de ajustar la aplicación a las necesidades de los usuarios finales, el profesorado.

En relación a los objetivos académicos, aspiramos a emplear los conocimientos adquiridos durante el Grado de Ingeniería del Software en un proyecto real.

1.3. Estructura del proyecto

La memoria se encuentra organizada en 8 capítulos, a continuación se relaliza un pequeño resumen de cada uno:

- **Capítulo uno:** Se presenta la introducción incluyendo la motivación, los objetivos y la sección de estructura del proyecto.
- **Capítulo dos:** Se presenta la introducción en inglés.
- **Capítulo tres:** Se presenta el estado del arte, en el que se define qué es la adaptación curricular y los tipos que hay, además se incluye las herramientas existentes y una mención a la apliación actual de AdaptaMaterialEscolar1.0.

- **Capítulo cuatro:** Se presenta la metodología usada juanto a sus re-gla, políticas y el tablero. También se encuentra explicado el plan de pruebas.
- **Capítulo cinco:** En este capítulo se explica todo lo relacionado con la segunda versión de AdaptaMaterialEscolar.
- **Capítulo seis:** En este capítulo se presenta las conclusiones y el tra-bajo futuro a realizar.
- **Capítulo siete:** Se presenta las conclusiones y el trabajo futuro a realizar en inglés.
- **Capítulo ocho:** Se presenta el trabajo individual realizado por cada integrante del grupo.

Chapter **2**

Introduction

Capítulo 3

Estado del arte

En esta sección nos centraremos en explicar el ámbito en el que se enmarca nuestro proyecto. Inicialmente en la Sección 3.1 definiremos el concepto de adaptación curricular, explicaremos en detalle los posibles tipos y añadiremos ejemplos de adaptaciones según las diferentes necesidades que pueden tener los estudiantes. En la Sección 3.2 se expondrán las herramientas existentes para la adaptación curricular, en la cual se encuentra explicada AdaptaMaterialEscolar 1.0.

3.1. Adaptación curricular

Para poder comprender la adaptación curricular inicialmente se debe definir la palabra currículum. El (BOE, 2014, p. 5, Artículo 2) define el currículum como *regulación de los elementos que determinan los procesos de enseñanza y aprendizaje para cada una de las enseñanzas y etapas educativas*.

Es decir, el currículum especifica los contenidos, materiales y recursos que deben presentar y administrar las instituciones educativas de todos los niveles.

En nuestro sistema educativo, aceptar la diversidad del alumnado y la individualidad de cada uno de ellos, constituye la base del quehacer de los docentes. Los profesores deben modificar el currículum y el programa de aula con el fin de mejorar el desarrollo del aprendizaje. Para poder realizar dicha actividad, el profesorado deberá detectar, evaluar y valorar al alumno y a los elementos curriculares y del entorno. Tras llevar a cabo lo anterior, el profesorado debe realizar las adaptaciones curriculares necesarias para lograr una mejora en el desarrollo personal y social del alumno. (Luque Parra y Romero Pérez (2002))

El (BOE, 2010, p. 7, Artículo 7) define la adaptación curricular como *la medida de modificación de los elementos del currículo a fin de dar respuesta a las necesidades del alumnado. En todo caso, la adaptación tendrá como referente los objetivos y las competencias básicas del currículo que corresponda.*

Es decir, la adaptación curricular es cualquier adaptación personal para estudiantes, cuyas necesidades no se encuentran cubiertas por el currículo ordinario y, por tanto, no pueden acceder a él de la misma manera que sus compañeros.

Para poder aplicar la adaptación curricular a un alumno se debe precisar el tipo de adaptación en función de sus necesidades, de manera que pueda alcanzar los objetivos propuestos. Para ello, definimos a continuación los tipos de adaptaciones que más uso tienen, enfocándonos tanto en las adaptaciones curriculares de acceso al currículum como en las adaptaciones curriculares individualizadas.

La información plasmada desde la Sección 3.1.1 en adelante, exceptuando las definiciones de cada discapacidad, se ha obtenido del libro (Gento Palacios, Sánchez Sáinz y Lakhdar, 2010, Diseño y ejecución de planes, proyectos y adaptaciones curriculares para el tratamiento educativo de la diversidad).

3.1.1. Adaptaciones de acceso al currículum

Las adaptaciones de acceso permiten al alumno acceder a los diferentes componentes del currículum. No implica una adaptación del currículum sino una adaptación en los recursos materiales, espaciales o de comunicación para que los alumnos con necesidades educativas especiales puedan desarrollar el currículum. Dicha adaptación puede tomar a su vez diferentes tipos: espacial, material o de comunicación.

3.1.1.1. Acceso espacial

Hace referencia a las adaptaciones en relación con el espacio ya sea el aula o el colegio. Destacan las siguientes:

- **Adaptación en la sonorización del aula:** Las aulas deben tener un cierto nivel de volumen adecuado, sin que haya ruidos continuos, sin eco, etc. Dichas clases son especialmente adecuadas para estudiantes que tienen impedimentos auditivos o visuales o que requieren, por su propia condición especial, un entorno con pocos estímulos auditivos que les distraiga.

- **Adaptación en la iluminación del aula:** Las aulas bien iluminadas requieren que estas no tengan sombras, deben poseer ventanales que suministren luz natural o en su lugar luz artificial que proporcione una iluminación óptima. Estos requisitos son necesarios para los estudiantes con discapacidades sensoriales, como por ejemplo las personas presentan visión monocular.
- **Adaptación en el espacio físico:** Es todo aquello relacionado con la ausencia de barreras arquitectónicas: braille en las puertas, ascensores, pasamanos, remate de las escaleras rugoso, etc. En esta sección también se encuentran los aspectos relacionados con la ubicación del aula (sin escaleras, lugares poco ruidosos) y con la disposición del estudiante dentro del aula (al lado de un enchufe, del profesor, de la puerta, etc.)

3.1.1.2. Acceso material

Hace referencia a la adaptación de materiales de uso frecuente o cuando se introducen nuevos instrumentos que faciliten el desarrollo del aprendizaje:

- **Materiales adaptados:** Se refieren a materiales que se usan habitualmente, y que se adaptan para un uso apropiado por parte de los alumnos con necesidades especiales. Ejemplos de ello sería plastificar un libro o hacer dibujos con relieve.
- **Materiales específicos:** Los materiales específicos deben superar las dificultades relacionadas con los elementos del aula de los alumnos. Por ejemplo, el mobiliario, las sillas y mesas deben de estar adaptadas, los comunicadores electrónicos con salidas de voz o escrita, etc.

3.1.1.3. Acceso de comunicación

Algunos alumnos son incapaces de comprender o expresarse por medio del lenguaje oral, o su nivel no es suficiente para poder comunicarse correctamente. Dichos alumnos requieren estudiar y usar códigos de comunicación suplementarios al lenguaje oral, o alguna alternativa al mismo. Aprender dichos códigos de comunicación facilitarán el acceso a los elementos curriculares ordinarios y les concederá una herramienta imprescindible tanto para el desarrollo de algunas competencias y aprendizajes de diferentes contenidos, como para relacionarse y comunicarse con el resto de personas. De esta manera, podemos destacar los siguientes sistemas que mejoran el acceso a la comunicación:

- **Sistemas alternativos a la comunicación:** En estos sistemas no se requiere del lenguaje oral para su utilización. Ejemplo de estos procedimiento son los símbolos pictográficos para la comunicación, la lengua de signos española, etc.

- **Sistemas complementarios a la comunicación:** No sustituyen al lenguaje oral, se encargan de reducir los problemas de comunicación y apoyar el acceso al lenguaje oral. Destacan la palabra complementada (sistema complementario de lectura labial), la comunicación bimodal (es un sistema de comunicación que utiliza el habla y los signos al mismo tiempo).

3.1.2. Adaptaciones curriculares individualizadas

Como se ha mencionado anteriormente, la adaptación curricular es un conjunto de refinamientos y cambios en los elementos del currículum para adaptar la respuesta educativa a las necesidades formativas de los estudiantes especiales.

Las adaptaciones curriculares individualizadas se dividen en dos tipos, las adaptaciones curriculares significativas y las adaptaciones curriculares no significativas.

3.1.2.1. Adaptaciones curriculares significativas

Según el (BOE, 2010, p. 7, Artículo 8), *una adaptación curricular será significativa cuando la modificación de los elementos del currículo afecten al grado de consecución de los objetivos, los contenidos y los aprendizajes imprescindibles que determinan las competencias básicas en la etapa, ciclo, grado, curso o nivel correspondiente.*

Es decir, la adaptación curricular significativa son los ajustes que se realizan en el contenido del currículum. Para su elaboración e implementación se debe seguir los siguientes enfoques:

- Inclusión. Se introducen elementos curriculares no presentes en el currículo. Puede incorporar objetivos, contenidos, criterios de evaluación, etc; conforme a las necesidades del alumno.
- Reformulación. Esta adaptación conlleva la amplia modificación de los elementos del currículo.
- Temporalización fuera de ciclo. Los alumnos con ritmo de aprendizaje lento con respecto a sus compañeros tendrán la oportunidad de conseguir los objetivos en otro ciclo posterior, posponiendo a otras etapas algunos elementos curriculares.
- Eliminación. Este tipo de adaptación es la más significativa. Inicialmente se deben eliminar contenidos, a continuación, criterios de evaluación y objetivos y finalmente se propondrá quitar material.

3.1.2.2. Adaptaciones curriculares no significativas

Son adaptaciones que no modifican sustancialmente el contenido del currículo oficial, es decir, se adapta la metodología, las actividades, los tiempos, las técnicas e instrumentos de evaluación. Para su elaboración se deben seguir los siguientes elementos:

- Evaluación: Se ajusta la manera de evaluar a las necesidades del alumno. Ejemplo de ello es cuando un alumno con escayola no puede realizar un examen escrito por lo que se le adapta la forma de evaluar realizando un examen oral.
- Metodología: Aquí se hace mención a cómo se enseña, es decir, a la forma de transmitir el aprendizaje. El desarrollo de la enseñanza-aprendizaje ha de ser activo, partiendo desde las necesidades del alumno. Además, ha de ser creativo, cooperativo y buscar una opción distinta al método tradicional de trabajo.
- Priorización de objetivos o contenidos: Dentro de la planificación se podría dar más valor a unos contenidos que a otros.
- Temporalización de contenidos u objetivos: Permitir más tiempo para alcanzar algunos de los contenidos pero respetando el ciclo.

3.1.3. Ejemplificación de algunas adaptaciones curriculares asociadas a diferentes necesidades

En esta sección se enumerarán algunos ejemplos de adaptaciones curriculares de acceso al currículum y de adaptaciones curriculares individualizadas asociadas a distintas necesidades según la clasificación de (Gento Palacios, Sánchez Sáinz y Lakhdar, 2010, Diseño y ejecución de planes, proyectos y adaptaciones curriculares para el tratamiento educativo de la diversidad)

- Discapacidad motora: Es un grupo de alteraciones que se producen como consecuencia de diversas anomalías en los sistemas que forman el movimiento (EducaMadrid). Este tipo de discapacidad requiere adaptaciones de acceso tales como rampas, pasamanos, suelos antideslizantes, etc.

En relación a las adaptaciones significativas ataúnen sobre todo al área de Educación Física, Música o Plástica ya que en estas asignaturas se precisa del manejo de instrumentos. Por ejemplo, en el área de Educación Física se realizarán ejercicios en los cuales el alumno utilizará los músculos que presenten una mayor funcionalidad, con el fin de mejorar la capacidad de respuesta.

Con respecto a las adaptaciones no significativas se debe adaptar la forma de evaluar ya que se debe tener en cuenta su movilidad. Un ejemplo de ello sería realizar los exámenes de forma oral.

- Discapacidad auditiva: Es la pérdida parcial o completa de la audición, supone la sustitución del lenguaje oral por otras vías como la visual (EducaMadrid).

Con respecto a las adaptaciones de acceso, el alumno se debe encontrar en una zona del aula en la que no haya muchas sombras ya que la adquisición de conocimientos se realiza por vía visual. Por otro lado, los alumnos con una pérdida parcial de la audición necesitan de un ambiente poco ruidoso.

En relación a las adaptaciones curriculares significativas, los profesores deberán trabajar de forma conjunta con especialistas en audición y lenguaje para que el alumno logre alcanzar los objetivos conectados con el lenguaje oral.

En cuanto a las adaptaciones curriculares no significativas, hay que tener en cuenta la manera de evaluar (a través de lengua de signos, evaluaciones objetivas o de respuesta cortas), además de la forma de hablar al alumno, ya que esta debe ser de un modo claro, sin gesticular excesivamente, etc.

- Discapacidad intelectual: La discapacidad intelectual es una condición que se caracteriza por limitaciones significativas tanto en el funcionamiento intelectual como en el comportamiento adaptativo (CDC). En este tipo de discapacidad no es muy relevante la adaptación de acceso, pero podemos destacar el posicionamiento del alumno en el aula, de manera que se encuentre en una zona donde no tenga muchas distracciones.

Las adaptaciones curriculares significativas se aplicarán en función de su nivel de competencia curricular.

En relación a las adaptaciones curriculares no significativas se centrarán en la metodología, como por ejemplo programar actividades que permitan la experimentación (manipulación).

- Espectro autista: El espectro del autismo se refiere a una variedad de trastornos del neurodesarrollo que se caracterizan por déficits sociales y de comunicación, y conductas restringidas y repetitivas (CDC).

En relación a las adaptaciones de acceso al espacio se precisa no realizar grandes cambios en la disposición del mobiliario. También destacamos las adaptaciones de comunicación, ya que las personas autistas se caracterizan por la ausencia de comunicación según el nivel. Para ayudar

a romper la barrera de la comunicación lo que realizan es asociar palabras con gestos e impulsar un refuerzo positivo.

En cuanto a las adaptaciones curriculares significativas, se debe introducir o priorizar el contenido en lo que respecta a la comunicación o rediseñar los objetivos o elementos que no puedan alcanzar.

Las adaptaciones curriculares no significativas se centran en la metodología. Las actividades deben de ser consistentes, con una estructura y organización claras.

- Altas capacidades intelectuales: En este caso no se requiere de adaptaciones de acceso ya que este tipo de alumnos no tienen dificultades para acceder al currículum.

Con respecto a las adaptaciones curriculares significativas lo que deberían realizar los profesores es ampliar el currículum añadiendo objetivos y contenidos.

Las adaptaciones curriculares no significativas hacen hincapié en la metodología, por ejemplo, haciendo actividades de ampliación.

3.2. Herramientas existentes para adaptaciones curriculares

3.2.1. SymWriter

SymWriter¹ es un procesador de texto que traduce las palabras a pictogramas a medida que se escriben, lo que facilita el acceso a vocabulario nuevo o complejo. También puede ayudar a profesores a crear textos basados en símbolos para potenciar la comprensión visual, entre otras cosas. Además de la traducción automática de palabras a pictogramas, ofrece una función de apoyo sonoro, un corrector ortográfico con soporte para símbolos y permite añadir imágenes. Esta herramienta ofrece una versión de prueba de 21 días, pero al acabarse será necesario adquirir la versión completa para poder seguir usandola. En la figura 3.1 se muestra la interfaz y un ejemplo de uso de SymWriter.

3.2.2. EducaPlay

EducaPlay² es una web que permite crear actividades interactivas o juegos didácticos. Entre las actividades que se pueden realizar en EducaPlay hay sopas de letras, pruebas, juegos de memoria, mapas interactivos, ejercicios de relacionar columnas, etc. EducaPlay está pensada, principalmente,

¹<https://www.widgit.com/es/support/index.htm>

²<https://es.educaplay.com/?lang=es>

para que los alumnos realicen las actividades desde un ordenador o dispositivo móvil, pero permite imprimir las actividades en papel. En la figura 3.2 se muestra una actividad de EducaPlay y en la figura 3.3 se muestra el formato de impresión de esta actividad.

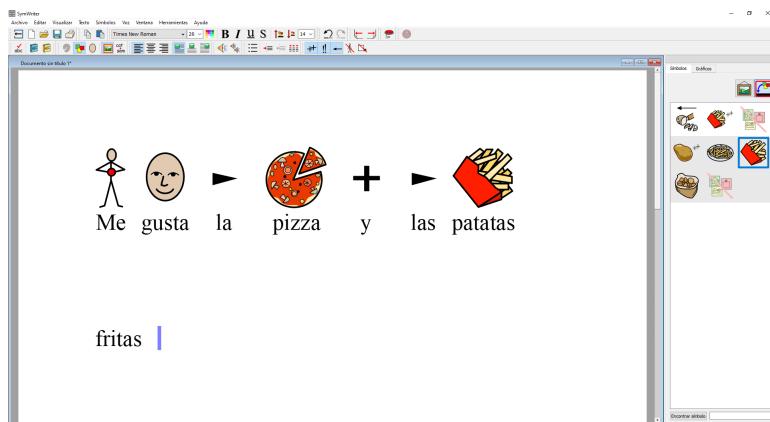


Figura 3.1: Interfaz de SymWriter

Figura 3.2: Actividad de EducaPlay

3.2.3. AdaptaMaterialEscolar 1.0

La finalidad de AdaptaMaterialEscolar es proporcionar una herramienta para el profesorado con el fin de realizar adaptaciones no significativas a los contenidos de las asignaturas de forma intuitiva, rápida y simple. Actualmente permite realizar las siguientes funciones:

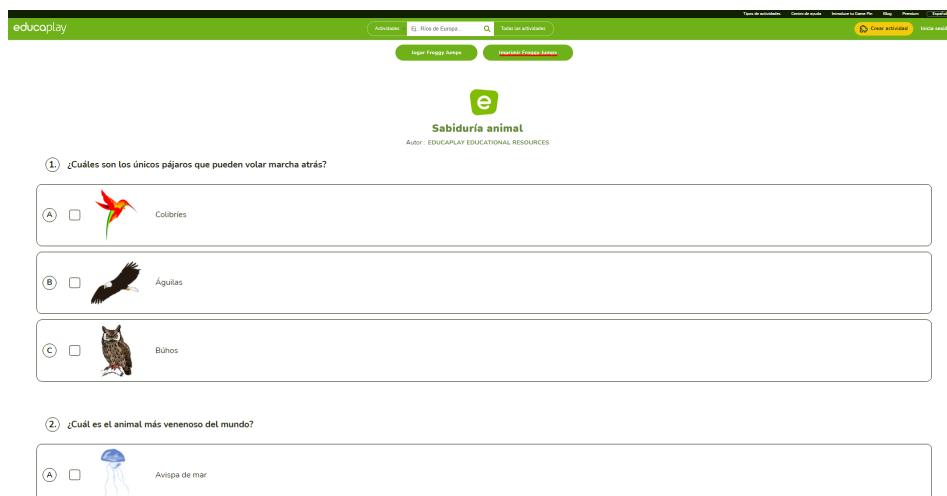


Figura 3.3: Impresión de actividad de EducaPlay

- Subir un documento fuente PDF, a partir del cual se puede realizar las adaptaciones, ejercicios, etc. (Figura 3.4)
- Buscar pictogramas: Dado una palabra devuelve un pictograma. (Figura 3.5)
- Generar ejercicios de completar huecos: Dado un texto se pueden seleccionar las palabras que deben ser sustituidas por espacios en blanco. (Figura 3.6)
- Proporcionar ejercicios de definiciones: Recibe una serie de conceptos a definir y devuelve un ejercicio en base a ellos. (Figura 3.7)
- Crear ejercicios de desarrollo: Permite crear un enunciado y añadir un cierto número de líneas para la respuesta. (Figura 3.8)
- Generar sopas de letras: El usuario introduce las palabras que desea que se pongan en la sopa de letras y el tamaño de la matriz. (Figura 3.9)
- Crear ejercicios de verdadero o falso: Genera una lista ordenada o no ordenada con las frases introducidas permitiendo definir cuales son verdaderas o falsas. (Figura 3.10)

La aplicación se creó siguiendo un Diseño Centrado en el Usuario, para que se adapte a las necesidades reales de los usuarios finales (los docentes). La captura de requisitos se realizó hablando directamente con el usuario final para poder conocer sus necesidades reales. Para esto se hicieron varias reuniones con 2 profesoras del Aula TEA del IES Maestro Juan de Ávila de Ciudad Real.



Figura 3.4: Subir PDF

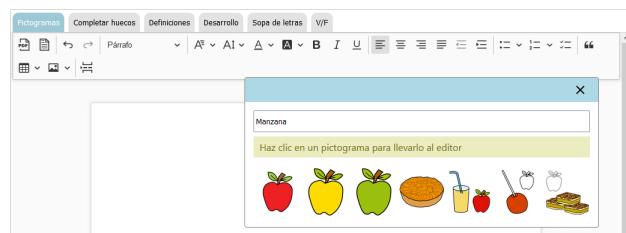


Figura 3.5: Busquedad de pictogramas

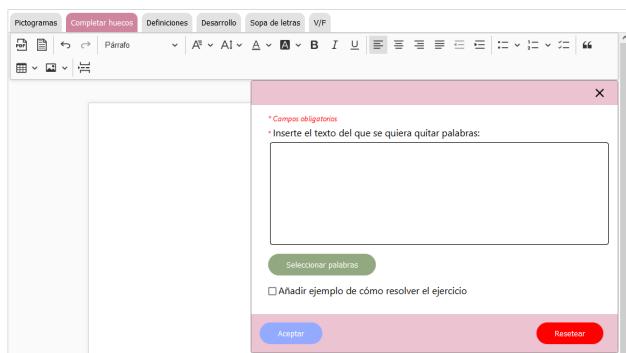


Figura 3.6: Ejercicios de completar huecos

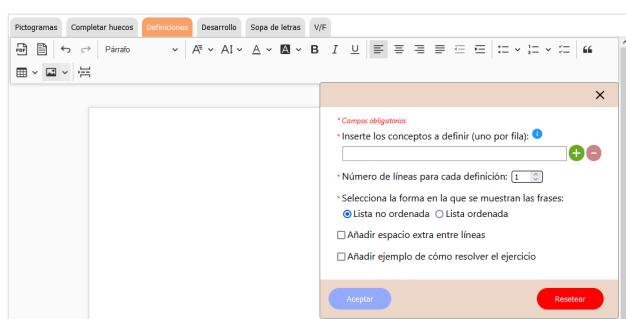


Figura 3.7: Ejercicios de definiciones

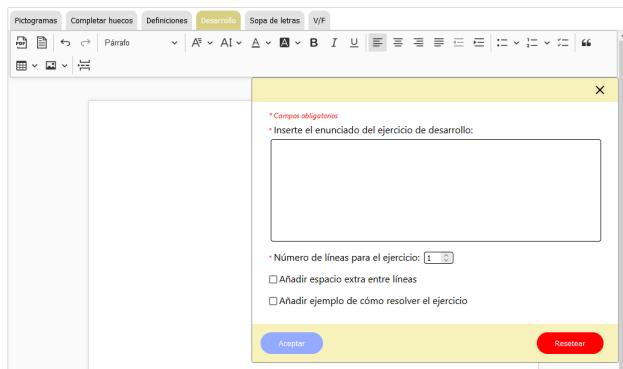


Figura 3.8: Ejercicios de desarrollo

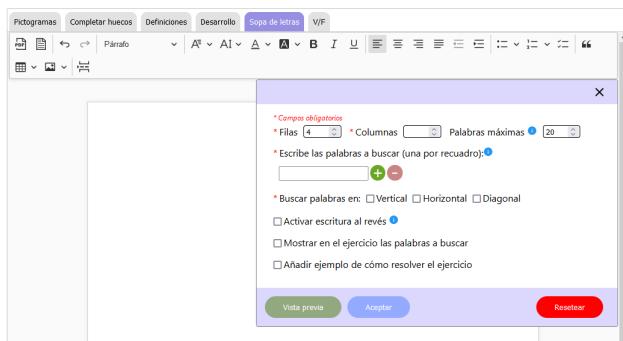


Figura 3.9: Sopas de letras

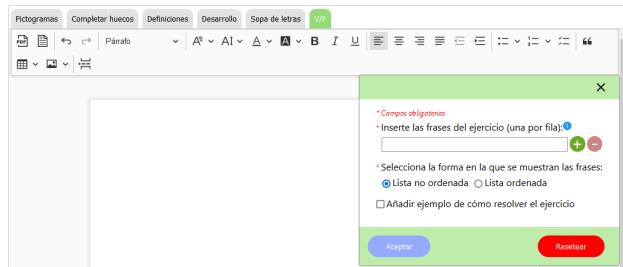


Figura 3.10: Ejercicios de Verdadero o Falso

Para el diseño de la aplicación se realizó una primera iteración de diseño competitivo y el resultado se presentó a los usuarios finales para obtener su feedback.

Para desarrollar AdaptaMaterialEscolar 1.0 se utilizó React. Para simplificar la gestión del estado de la aplicación, se utilizó Redux. Para el editor de texto de la aplicación se utilizó CKEditor. Por último, para el buscador de pictogramas, se empleó la API de ARASAAC. Esta API permite hacer una petición con un término de búsqueda y devuelve una serie de pictogramas

relacionados.

Finalmente la aplicación fue evaluada por los usuarios finales, con el objetivo de descubrir si la aplicación es, en cuanto a la adaptación curricular no significativa, realmente útil para los profesores y si les ayudaba a resolver sus problemas. Para esto se creó un examen de Ciencias Naturales adaptado usando AdaptaMaterialEscolar 1.0. Luego, se replicó este examen con los profesores para mostrarles cómo se usaría la herramienta en situaciones reales.

Después de esta demostración, se les hizo una encuesta a los profesores para que pudieran dar su opinión y feedback. Este cuestionario tenía preguntas sobre usabilidad, diseño, funcionalidades y utilidad real de la aplicación. Además, se pasó a los evaluadores el cuestionario SUS (System Usability Scale), que sirve para medir qué tan buena es la usabilidad de un sistema. Esta evaluación de la aplicación se realizó con 6 profesores. Como resultado de la evaluación se llegó a la conclusión de que la aplicación sí resuelve problemas reales que tienen los profesores y que se debería seguir desarrollando. Se observó que los pictogramas a veces eran muy pequeños y debían poder aumentarse de tamaño. A partir del feedback de los usuarios finales que evaluaron la aplicación, se obtuvo una nueva lista de requisitos a implementar para mejorar la aplicación:

- Traducir pictogramas a lenguaje natural.
- Introducir cuadrícula para ejercicios de matemática.
- Poder añadir doble pauta, en vez de renglones de una línea, para determinar el tamaño de letra del alumno.
- Añadir herramienta para recortar imágenes.
- Añadir encabezado con el nombre del centro educativo, asignatura y nombre del alumno.
- Permitir añadir espacio para dibujar en los ejercicios.
- Permitir ejercicios de cálculo con fórmulas con huecos que puedan ser llenados por el alumno.
- Enumerar ejercicios automáticamente.
- Añadir una fuente de texto parecida a la que suelen aprender la mayoría de los alumnos cuando empiezan a escribir.

Metodología

En este capítulo se explicará la metodología de desarrollo utilizada en la Sección 4.1 y se describirá el plan de pruebas en la Sección 4.2.

4.1. Metodología de desarrollo

La metodología es un marco de trabajo que permite estructurar, organizar y controlar el proceso del desarrollo software. Para el desarrollo de este trabajo hemos decidido aplicar la metodología Kanban. Esta metodología se rige por unas reglas básicas como: gestionar el flujo; hacer las políticas explícitas; visualizar el trabajo y las fases del ciclo de producción o flujo de trabajo y determinar a la vez que respetar el límite de trabajo en curso.

4.1.1. Tablero Kanban

Para el tablero Kanban hemos decidido crear cinco columnas: *To Do*, *Doing*, *Testing*, *Validate* y *Done*. Previamente hemos distinguido dos tipos de tareas que habrá en el proyecto, las tareas relacionadas con la memoria y las de implementación. Cada tipo de tarea se probará o revisará de manera diferente. Las tareas continuarán a través del flujo siguiendo las siguientes definiciones de las columnas:

- ***To Do***: Listado de todas las tareas sin empezar.
- ***Doing***: Tareas que se encuentran en desarrollo, ya sea la implementación de código o la redacción de la memoria.
- ***Testing***: Una vez desarrollada la tarea, se probará que cumpla con los requisitos.
- ***Validate***: La tarea de memoria será comprobada por las tutoras.

- **Done:** Las tutoras han dado el visto bueno a la tarea de memoria o cuando un integrante del grupo haya finalizado su tarea de código.

Para las tareas de documentación, cuando se situen en la columna de *Testing* la realizarán todos integrantes del equipo siguiendo los siguientes pasos:

1. Cuando haya una tarea de memoria en la columna de *Testing*, esta dispondrá de una lista con *checkboxes* con los nombres de los integrantes.
2. Cuando un miembro del equipo haya terminado de revisar la tarea debe marcarlo en el *checkbox* referido a él.
3. Cuando todos los miembros del equipo hayan revisado la tarea, el último revisor se encargará de mover la tarea a la columna de *Validate*.

Por otra parte, si se encuentra un error en una funcionalidad, ya sea durante la fase de *Testing* o tras haberse dado por terminada, se creará una nueva tarea de tipo *bug*.

4.1.2. Políticas explícitas

A continuación se presentan las políticas explícitas que hemos ido estableciendo a lo largo del proyecto:

- Límites del trabajo en curso (WIP): cuatro tareas en curso por persona en todo el tablero sin tener en cuenta la columna *Validate*.
- Definición de *Done*:
 - Tareas de memoria: Cuando hayan sido validadas por las tutoras.
 - Tareas de implementación: Cuando hayan pasado todo el plan de pruebas.
- Cuando un integrante del grupo haya terminado su tarea él será el encargado de moverla a la columna correspondiente.
- Cualquier integrante del grupo puede poner una tarea en el tablero tras consultarla con el resto.
- Hemos llegado al acuerdo de realizar reuniones todos los domingos a las 12:00, para poner en común el trabajo realizado por cada miembro.

4.1.3. Clases de servicios

En Kanban para priorizar las tareas del tablero en ocasiones se emplean las clases de servicio. Estas son una serie de categorías que nos son útiles para clasificar cada una de las tareas de nuestro sistema, las cuales nos permiten

identificar rápidamente el nivel de prioridad que tiene la tarea sin hacer un análisis o estimación muy extensa del mismo. Además, la categoría asociada a una tarea determinará como se moverá la tarea en el tablero.

En nuestro caso las clases de servicio que empleamos son las siguientes:

- **Expedite:** Tareas que necesitan ser gestionadas de manera acelerada o urgente.
- **Fixed Delivery Date:** Tareas con fecha fija que debemos cumplir.
- **Standard:** Tareas que ya ha hecho antes el equipo y que no tienen una fecha fija.
- **Intangible:** Tareas que son nuevas, se desconoce el tiempo que se le va dedicar y el riesgo que suponen.

Para su aplicación tomaremos unas medidas en base a su prioridad. Las clases *Expedite* son las más prioritarias por lo que serán las primeras en ser realizadas. Las *Fixed Delivery Date* si en su debida fecha, la cual estará indicada en su descripción, no están implementadas se convierten en *Expedite*. Las *standard* son un poco menos prioritaria que las anteriores al contar con el coste y el esfuerzo que suponen pero presentan un cierto grado de incertidumbre al no tener una fecha fija. Por último, las *Intangibles* su prioridad varía al presentar un alto grado de incertidumbre inicialmente se desconoce el riesgo que suponen pero pueden convertirse en *Standard* o en *Expedite*.

4.2. Plan de Pruebas

Las pruebas de implementación consistirán en dos tipos de pruebas: las pruebas unitarias y las pruebas de integración. Las pruebas de una tarea de implementación concreta las realizará algún miembro del equipo que no haya participado en el desarrollo de esta y las hará cuando la tarea se encuentre en la columna de *Testing*. La ventaja de que las pruebas las realice un miembro que no se haya visto involucrado en el desarrollo de la tarea es que puede sacar más casos de prueba que aquellos miembros que han implementado la tarea y conocen el código.

Para ambos tipos de pruebas, unitarias y de integración, se utilizará la herramienta Jest (ver sección de herramientas).

4.2.1. Pruebas Unitarias

Las pruebas unitarias son pruebas, mayoritariamente automatizadas, que verifican la funcionalidad de una unidad software (componente, clase o método), de forma aislada. Esta verificación no se debe ver afectada por otros casos de prueba que se hayan ejecutado anteriormente o por dependencias

con otros módulos de la aplicación. Para evitar que las dependencias de un módulo, por ejemplo el acceso a una base de datos o una petición a una API externa, afecten al resultado de la prueba, se pueden usar *stubs* para simular esas dependencias.

La razón por la que vamos a realizar pruebas unitarias es porque permiten detectar errores en fases tempranas y evitar que los errores se propaguen a fases posteriores, aumentando la calidad del software. Además, facilitan los cambios, ya que se puede comprobar rápidamente que estos no han afectado al funcionamiento esperado de la aplicación.

Para aprovechar al máximo las ventajas que ofrecen las pruebas unitarias hay que asegurarse de que cumplen las siguientes características:

- Deben ser automáticas y repetibles, es decir, se deben poder ejecutar tantas veces como uno quiera sin necesidad de intervención manual durante las pruebas.
- Deben ser rápidas, ya que si el ejecutar las pruebas unitarias es un proceso lento, no se va a llevar a cabo tantas veces como sea necesario.
- Cada caso de prueba debe estar aislado completamente del resto de casos o pruebas de otros módulos. Si no se aísla correctamente, se pueden producir resultados no consistentes, complicando la detección de errores.
- Las pruebas deben ser relevantes para el futuro y deben tratar de cubrir la totalidad del código.

4.2.2. Pruebas de integración

Las pruebas de integración se utilizan para comprobar que las conexiones o interfaces entre los distintos módulos, ya probados individualmente (pruebas unitarias), funcionan correctamente.

En este proyecto las pruebas de integración se realizarán de forma incremental, es decir, los módulos se integran uno a uno, a medida que estos se van desarrollando y probando. En esta estrategia, la integración se lleva a cabo de arriba abajo, siguiendo el flujo natural de la aplicación.

Se seguirá una estrategia *Top-Down* para realizar las pruebas. Utilizaremos esta estrategia, en vez de *Bottom-Up*, porque nos queremos centrar en errores de diseño y tener una aplicación funcional probada lo antes posible. Puede darse el caso de que una prueba de integración no se pueda realizar debido a que alguno de los módulos a integrar está en desarrollo o todavía se está probando individualmente. En este caso, se utilizarán *stubs* para simular la dependencia entre módulos.

4.2.3. Integración continua

Para complementar las pruebas unitarias y de integración, en este proyecto se implementará integración continua. La integración continua es la práctica de desarrollo software mediante la cual los miembros del equipo combinan su trabajo frecuentemente en un repositorio compartido. Cada integración se verifica mediante una serie de fases por las que va pasando el software y que se automatizan. Esta serie de fases se conoce como *pipeline*.

El *pipeline* que se va a implementar en este proyecto constará de 3 fases principales: *Build*, *Test* y *Release*. Cuando un miembro integre una nueva versión del proyecto en el repositorio compartido, pasará primero por la fase *Build*, durante la cual se instalarán las dependencias necesarias en una maquina virtual y se preparará todo lo necesario para pasar a la fase *Test*. En la fase *Test*, se ejecutarán todas las pruebas especificadas en el plan de pruebas a la versión que se está intentando subir al repositorio. En caso de que el resultado de las pruebas sea correcto, se pasará a la fase *Release*. En caso de que se produzca un error en alguna de las pruebas se abortará la ejecución de cualquier prueba restante, no se pasará a la fase *Release* y por lo tanto no se subirán los cambios realizados en la versión que se estaba compartiendo. En la fase *Release* se subirán los cambios automáticamente al repositorio.

Las razones por las que creemos necesaria la integración continua en este proyecto son las siguientes:

- La integración continua mejora la productividad de desarrollo, ya que libera a los desarrolladores de tener que ejecutar las pruebas manualmente y tener que esperar a que se arreglen errores en el repositorio. Si al integrar el trabajo se produce un error en la verificación se notificará a los desarrolladores y el cambio que ha producido ese error no se subirá al repositorio compartido.
- Mejora la detección de errores debido a la ejecución de pruebas de forma automática y frecuente, lo que permite a los desarrolladores descubrir los errores y arreglarlos antes de que se conviertan en problemas graves.

En este proyecto se utilizará la herramienta de GitHub Actions (ver sección de herramientas) para implementar la integración continua.

AdaptaMaterialEscolar 2.0

En este capítulo explicaremos la obtención de requisitos y su priorización en la Sección 5.1. También se describirá la iteración competitiva para el diseño de la aplicación en la Sección 5.2.

5.1. Requisitos

Lo primero que hicimos fue analizar la memoria de AdaptaMaterialEscolar 1.0 extrayendo las funcionalidades que faltaban por implementar y los resultados de la evaluación que se realizó. Tras este análisis surgieron una serie de cambios y nuevas funcionalidades. Agrupamos dichas funcionalidades en tres grupos: formato, ejercicios y auxiliar. Quedando las funcionalidades agrupadas de la siguiente manera:

Formato:

- Añadir encabezado al texto: el usuario elegirá un encabezado y se le añadirá al procesador de texto.
- Añadir un tipo de fuente escolar: incluir en los tipos de fuentes la escolar. Dicha fuente se refleja en la imagen 5.1.



ABCDEFGHIJKLMNÑOPQRSTUVWXYZ
abcdefghijklmnñopqrstuvwxyz
áéíóú

Figura 5.1: Fuente escolar.

- Añadir una leyenda de colores con la categoría de cada tipo: crear leyenda de colores que permita asociar colores con términos de un texto.
- Añadir leyenda de colores para el tema de cada asignatura: crear una leyenda según el color del borde de la hoja asignada a cada asignatura.
- Añadir ejercicios de matemáticas con cuadrícula para escribir los números: crear una hoja de cuadrículas para los ejercicios de matemáticas.
- Añadir la alternativa de añadir doble pauta: en vez de renglones de una única línea, se podrá crear una hoja con doble pauta, para determinar el tamaño de la letra del alumno.
- Estandarizar formato para títulos e índices del temario: escoger o definir un formato para el documento editable.
- Enumerar ejercicios de forma automática: establecer un orden numérico para los ejercicios.

Ejercicios:

- Ejercicios de relacionar contenido mediante flechas: generar un ejercicio para relacionar conceptos.
- Añadir ejercicios de cálculo con huecos a llenar por el alumno: espacios en blanco para que el alumno pueda llenarlos con el contenido adecuado.
- Añadir ejercicios con espacio para dibujar: amplio hueco en blanco con el fin de que el alumno pueda dibujar.
- Ejercicios de completar los espacios en blanco en tablas y esquemas: dado una tabla o esquema se establecen espacios en blanco para que el alumno los rellene con el contenido adecuado.

Auxiliar:

- Generar un resumen a partir de un texto: se crea un resumen a partir de un texto.
- Exportar el documento a formato Word para hacer modificaciones: exportar un documento a Word.
- Añadir un pictotraductor como funcionalidad: dado una frase genera sus respectivos pictogramas.
- Añadir imágenes buscando una palabra: a partir de una palabra se busca su respectiva imagen en las bases de datos de imágenes libres.

- Sustituir una palabra por una imagen: una palabra se reemplazará por una imagen.
- Crear una herramienta de recorte de imágenes para el texto original: añadir una herramienta que recorte imágenes del texto original.
- Crear tablas que organicen el temario y/o las actividades, seleccionando contenido: crear una tabla seleccionando el contenido.
- Crear esquemas que faciliten la visualización: añadir un esquema para visualizar fácilmente los contenidos.

Tras haber analizado en detalle las funcionalidades anteriores hemos encontrado que varias funciones ya están realizadas y otras no se van a implementar por falta de información. A continuación, especificamos las funciones realizadas, las funciones que no aportan información suficiente y las que realizaremos.

Funciones realizadas:

- Añadir encabezado al texto: el usuario elegirá un encabezado y se le añadirá al procesador de texto.
- Enumerar ejercicios de forma automática: establecer un orden numérico para los ejercicios.

Funciones sin información suficiente:

- Añadir imágenes buscando una palabra: a partir de una palabra se busca su respectiva imagen en las bases de datos de imágenes libres.
- Sustituir una palabra por una imagen: una palabra se reemplazará por una imagen.
- Crear una herramienta de recorte de imágenes para el texto original: añadir una herramienta que recorte imágenes del texto original.
- Crear tablas que organicen el temario y/o las actividades, seleccionando contenido: crear una tabla seleccionando el contenido.
- Crear esquemas que faciliten la visualización: añadir un esquema para visualizar fácilmente los contenidos.
- Ejercicios de completar los espacios en blanco en tablas y esquemas: dado una tabla o esquema se establecen espacios en blanco para que el alumno los rellene con el contenido adecuado.

Por lo tanto, las fuciones a implementar son las que se muestran a continuación:

- Generar un resumen a partir de un texto: se crea un resumen a partir de un texto.
- Exportar el documento a formato Word para hacer modificaciones: exportar un documento a Word.
- Añadir un pictotraductor como funcionalidad: dado una frase genera sus respectivos pictogramas.
- Ejercicios de relacionar contenido mediante flechas: generar un ejercicio para relacionar conceptos.
- Añadir un tipo de fuente escolar: incluir en los tipos de fuentes la escolar. Dicha fuente se refleja en la imagen 5.1.
- Añadir una leyenda de colores con la categoría de cada tipo: crear leyenda de colores que permita asociar colores con términos de un texto.
- Añadir ejercicios de cálculo con huecos a llenar por el alumno: espacios en blanco para que el alumno pueda llenarlos con el contenido adecuado.
- Añadir ejercicios con espacio para dibujar: amplio hueco en blanco con el fin de que el alumno pueda dibujar.
- Añadir leyenda de colores para el tema de cada asignatura: crear una leyenda según el color del borde de la hoja asignada a cada asignatura.
- Añadir ejercicios de matemáticas con cuadrícula para escribir los números: crear una hoja de cuadrículas para los ejercicios de matemáticas.
- Añadir la alternativa de añadir doble pauta: en vez de renglones de una única línea, se podrá crear una hoja con doble pauta, para determinar el tamaño de la letra del alumno.
- Estandarizar formato para títulos e índices del temario: escoger o definir un formato para el documento editable.

5.2. Diseño de la aplicación

Para el diseño de la aplicación web hemos realizado una iteración competitiva. Cada integrante del grupo ha proporcionado un diseño de las funcionalidades. El diseño de Álvaro Gómez Sittima se muestra en las figuras 5.2, 5.3, el de Dunia Namour Doughani se reflejan en la figura 5.4, el diseño de Alberto Alejandro Rivas Fernández se muestra en las figuras 5.5, 5.6, 5.7 y por el último el diseño de Johan Sebastian Salvatierra Gutierrez se muestra

en las figuras 5.8, refIteracionCompetitivaJ2. Una vez que cada integrante ha explicado su diseño, hemos cogido lo mejor de cada uno quedando la aplicación de la siguiente manera 5.10.

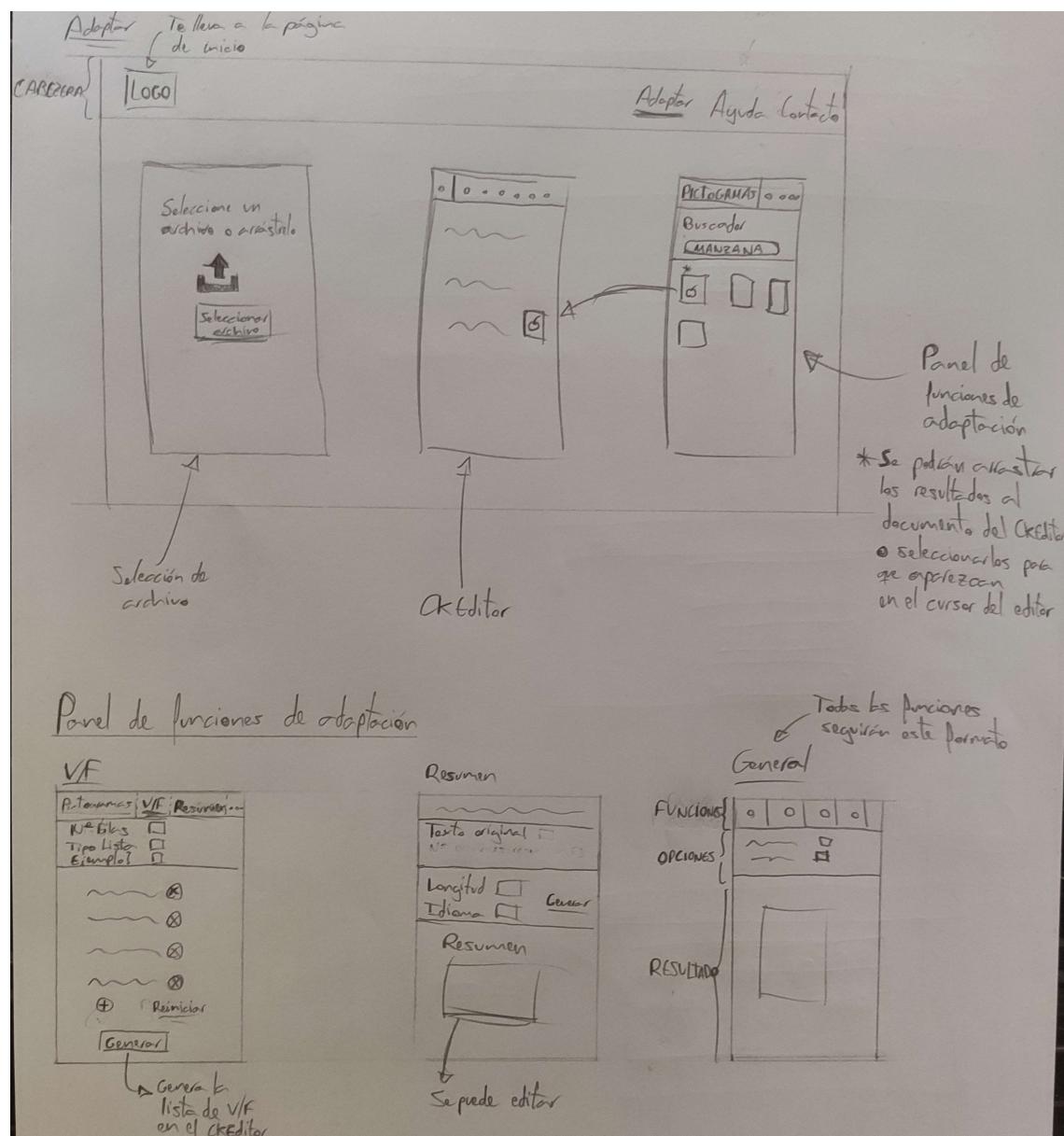


Figura 5.2: Diseño 1 Álvaro Gómez Sittima iteración competitiva.

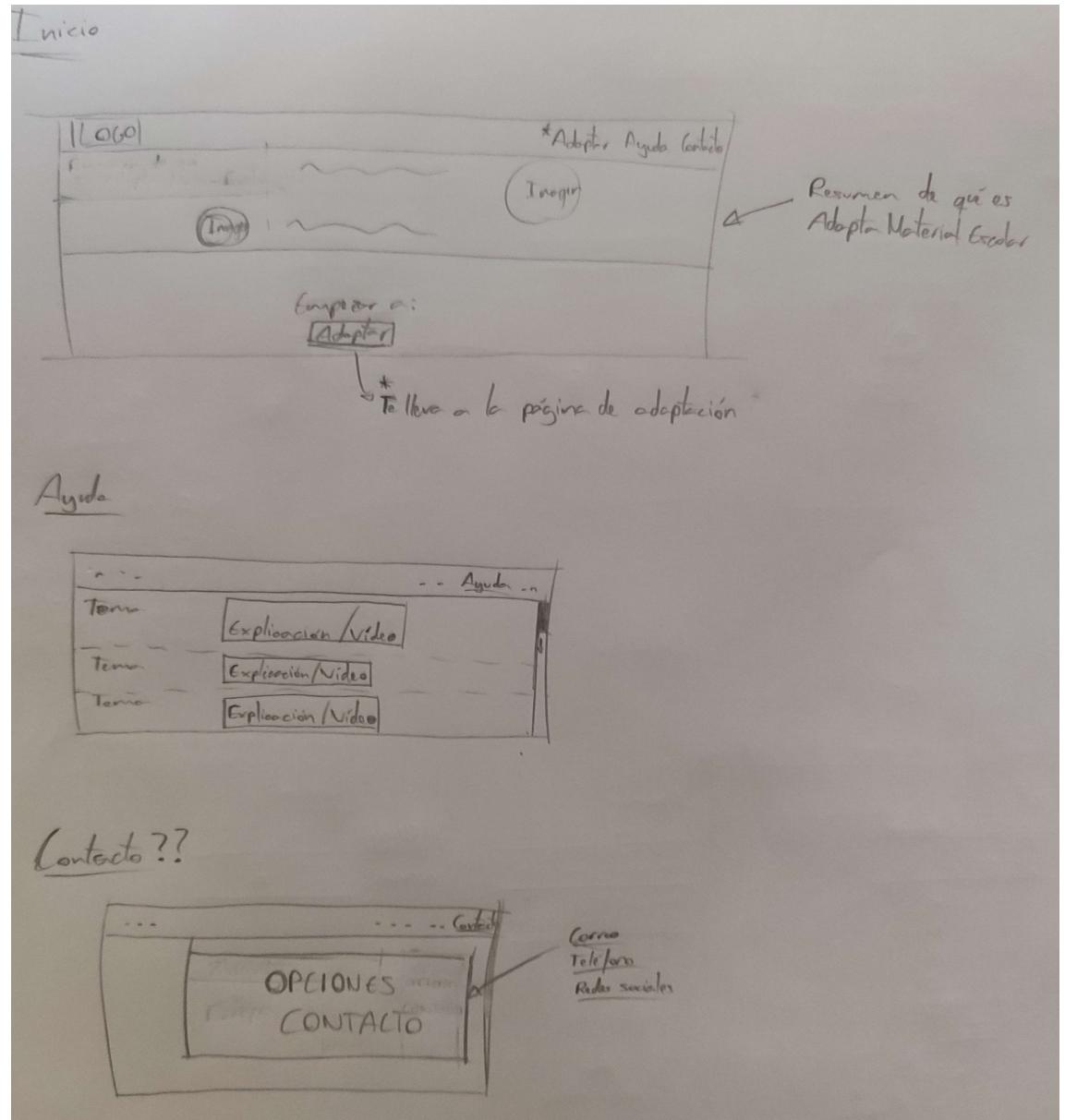


Figura 5.3: Diseño 2 Álvaro Gómez Sittima iteración competitiva .

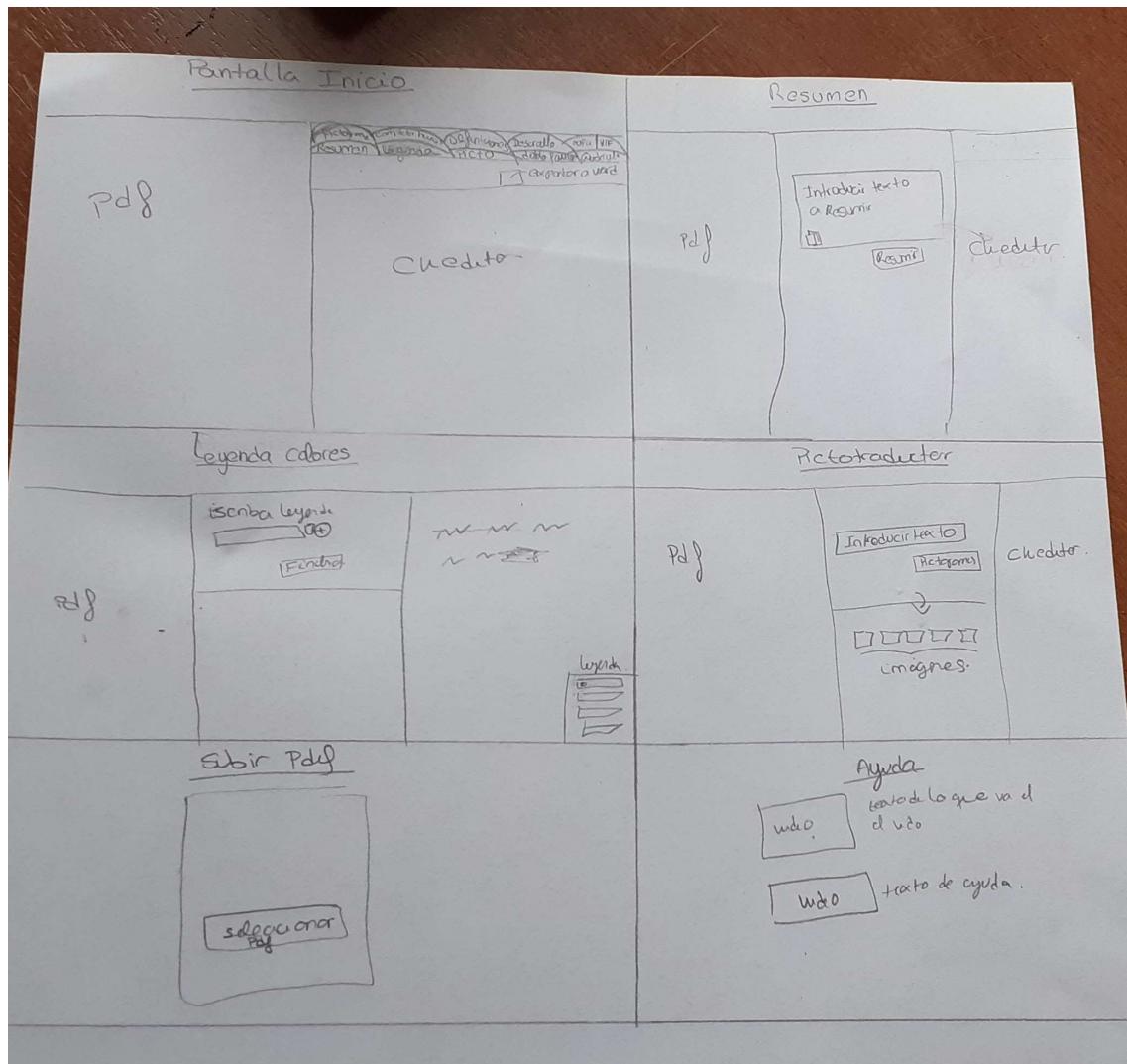


Figura 5.4: Diseño Dunia Namour Doughani iteración competitiva .

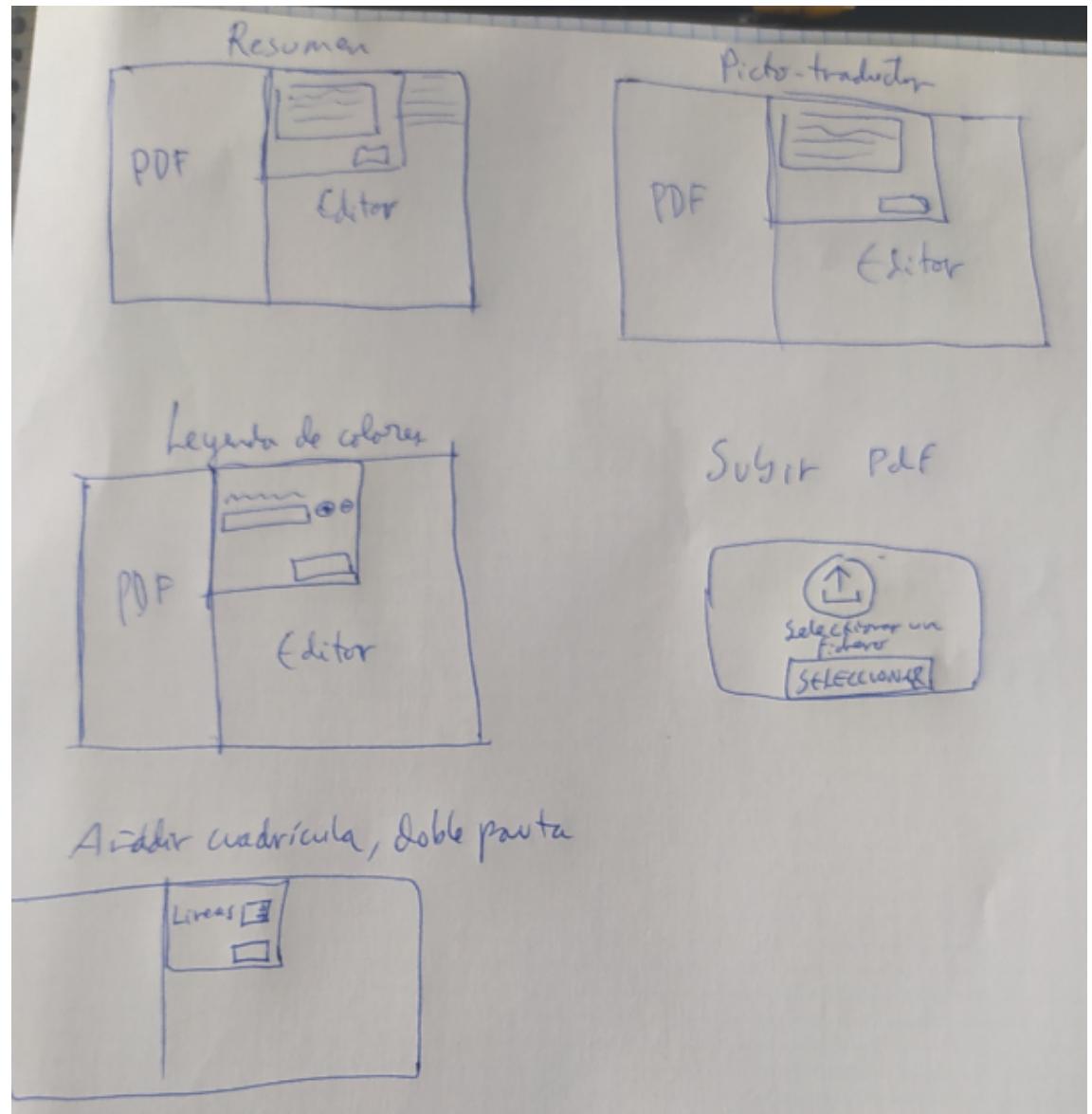


Figura 5.5: Diseño 1 Alberto Alejandro Rivas Fernández iteración competitiva.

*** Campos obligatorios**

Inserte el texto del que se quiera quitar palabras:

G

Seleccionar palabras

Añadir ejemplo de cómo resolver el ejercicio

Aceptar Resetear

Inserte el texto del que se quiera quitar palabras:

Seleccionar palabras

Añadir ejemplo de cómo resolver el ejercicio

Resetear Aceptar

Figura 5.6: Diseño 2 Alberto Alejandro Rivas Fernández iteración competitiva.

*** Campos obligatorios**

Inserte las frases del ejercicio (una por fila):
 + **-**

Selecciona la forma en la que se muestran las frases:

Lista no ordenada Lista ordenada

Añadir ejemplo de cómo resolver el ejercicio

Aceptar Resetear

Inserte las frases del ejercicio (una por fila):
 + **-**

Selecciona la forma en la que se muestran las frases:

Lista no ordenada Lista ordenada

Añadir ejemplo de cómo resolver el ejercicio

Resetear Aceptar

Figura 5.7: Diseño 3 Alberto Alejandro Rivas Fernández iteración competitiva.

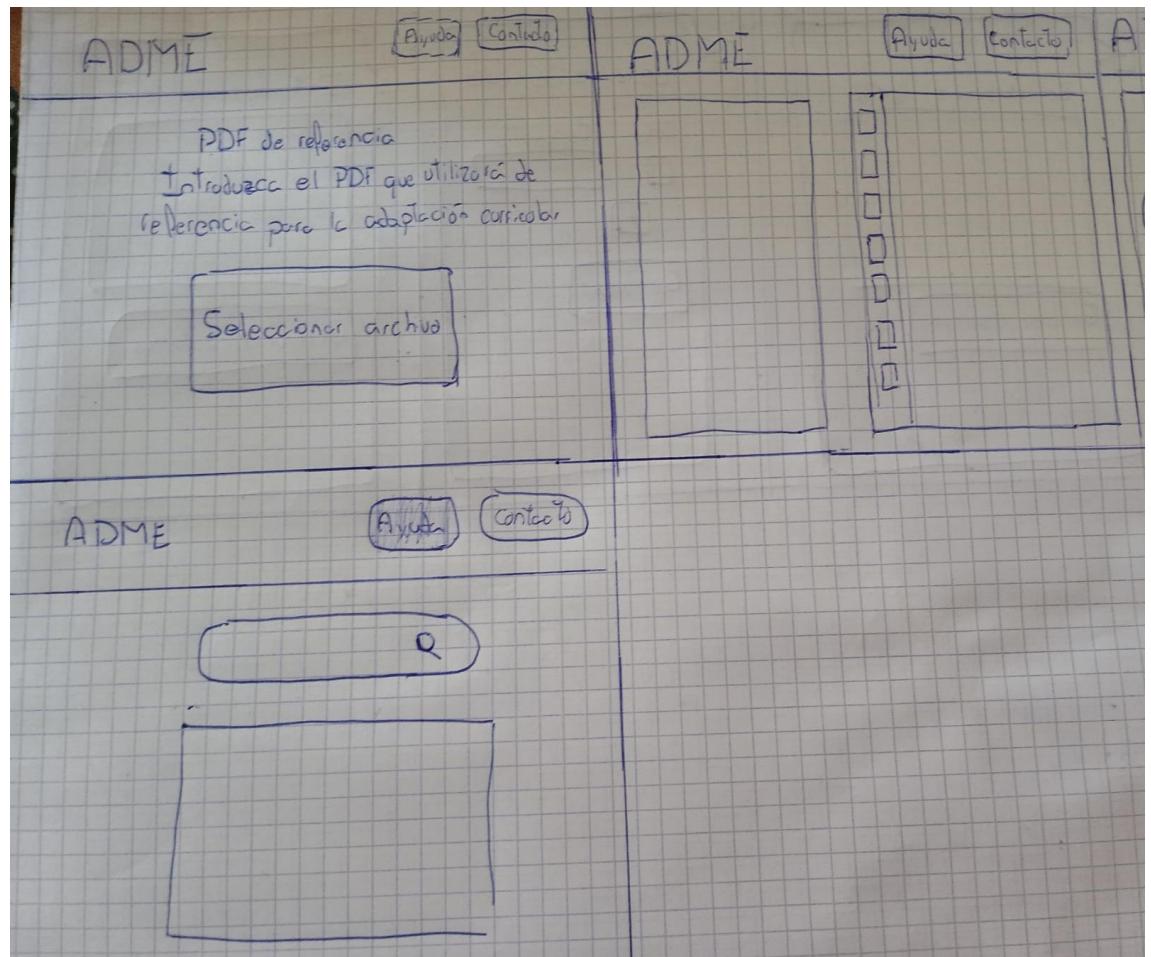


Figura 5.8: Diseño 3 Johan Sebastian Salvatierra Gutierrez iteración competitiva.

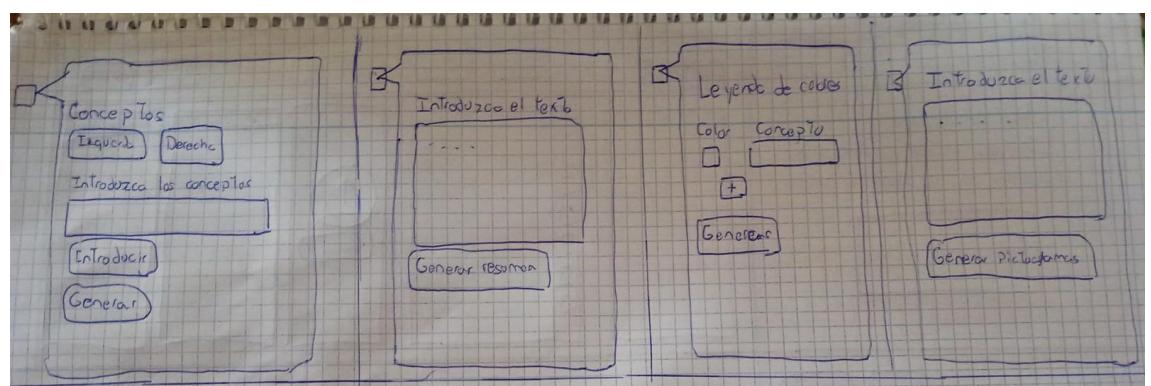


Figura 5.9: Diseño 3 Johan Sebastian Salvatierra Gutierrez iteración competitiva.

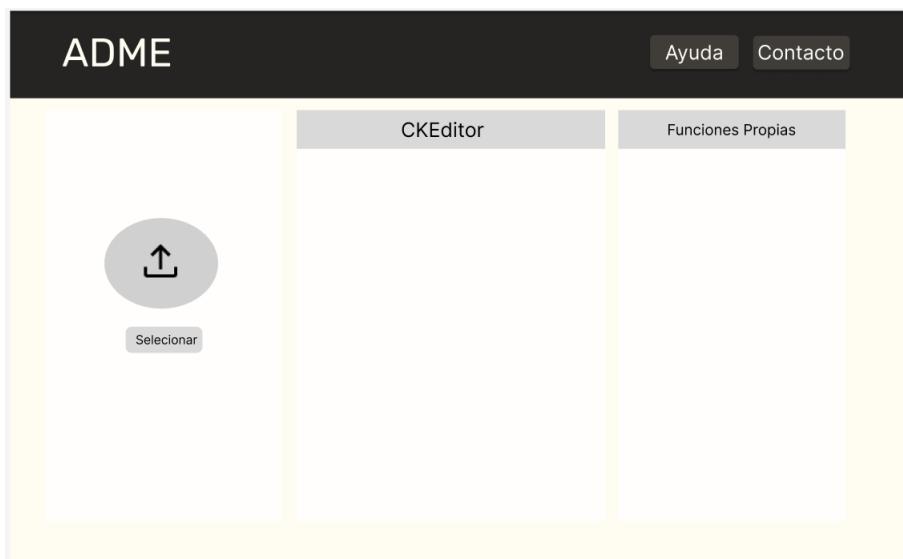


Figura 5.10: Diseño final aplicación.

Capítulo **6**

Conclusiones y Trabajo Futuro

Conclusiones del trabajo y líneas de trabajo futuro.

Chapter 7

Conclusions and Future Work

Conclusions and future lines of work.

Trabajo Individual

En este capítulo se habla del trabajo que ha realizado cada miembro del equipo en el proyecto.

8.1. Álvaro Gómez Sittima

Con respecto a la memoria he realizado la Sección 3.2 en la cual se comparan algunas aplicaciones similares a AdaptaMaterialEscolar. También, he redactado la Sección 4.2 donde se explica el plan de pruebas que ha seguido este proyecto. Además, junto a mis compañeros, he realizado los objetivos, el diseño y la metodología empleada. También he realizado una investigación para la lista de priorización.

8.2. Dunia Namour Doughani

Con respecto a la memoria he realizado en Capítulo tres la Sección 3.1 en la cual se explica la adaptación curricular y sus tipos. También he buscado información sobre las funcionalidades para poder realizar una adecuada priorización. Además, he redactado la estructura del proyecto. Junto a mis compañeros he realizado los objetivos, la parte relacionada con la metodología y el diseño de la aplicación.

8.3. Alberto Alejandro Rivas Fernandez

En relación a la memoria he realizado en el Capítulo tres la Sección 3.2.3 en la cual hablaba sobre la primera versión de AdaptaMaterialEscolar. También junto a mis compañeros hemos explicado la metodología y priorizado las funcionalidades, para ello investigué acerca de ellas, sobre todo la funcionalidad de pasar de pdf a word. Asimismo realicé un diseño de la página principal

de la aplicación para compararlo con los diseños de mis compañeros. Además, le escribí un email al servicio al cliente de CKEditor para preguntarles cómo obtener una licencia que podamos usar para nuestro proyecto. Luego creé una cuenta gratis en CKEditor para poder obtener la licencia y también probé comentar un trozo de código para hacer que el CKEditor funcionase al ejecutar la aplicación.

8.4. Johan Sebastian Salvatierra Gutierrez

Con respecto a la memoria he realizado el Capítulo uno Sección 1.1 en la cual se explica la motivación de nuestro TFG y el Capítulo 4 Sección 4.1.3 donde se aclara las clases de servicio que empleamos para el tablero Kanban. Además junto a mis compañeros he realizado los objetivos, el diseño y la metodología empleada. También he realizado una investigación para la lista de priorización.

Bibliografía

AMAZON. Integración continua. <https://aws.amazon.com/es/devops/continuous-integration/>, n.d.

BOE. Concepto de adaptacion curricular y tipos. <https://boe.es/buscar/pdf/2010/BOE-A-2010-5493-consolidado.pdf>, 2010.

BOE. Definición de currículum. <https://www.boe.es/buscar/pdf/2014/BOE-A-2014-2222-consolidado.pdf>, 2014.

CDC. Definición de discapacidad intelectual. <https://www.cdc.gov/ncbdd/spanish/developmentaldisabilities/hoja-informativa-sobre-discapacidad-intelectual.html>, n.da.

CDC. Definición de espectro autista. <https://www.cdc.gov/ncbddd/spanish/autism/signs.html>, n.db.

EDUCAMADRID. Definición de discapacidad auditiva. <https://www.educa2.madrid.org/web/centro.eoep.torrejondeardoz/discapacidad-auditiva>, n.da.

EDUCAMADRID. Definición de discapacidad motora. <https://www.educa2.madrid.org/web/eoep.discapacidadmotora.madrid>, n.db.

GENTO PALACIOS, S., SÁNCHEZ SÁINZ, M. y LAKHDAR, S. M. *Diseño y ejecución de planes, proyectos y adaptaciones curriculares para el tratamiento educativo de la diversidad*. Universidad Nacional de Educación a Distancia, 2010.

LUQUE PARRA, D. J. y ROMERO PÉREZ, J. F. Trastornos del desarrollo y la adaptación curricular. 2002.

REDHAT. Pipeline. <https://www.redhat.com/en/topics/devops/what-is-pipeline>, 2022.

SOFTWARE TESTING HELP. Pruebas de integración incrementales. <https://www.softwaretestinghelp.com/incremental-testing/>, 2022.

