



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería
Informática**

**Transformaciones invariantes a
la iluminación para mejora de
la segmentación en imágenes.
Aplicación a piezas metálicas.**



Presentado por Jonás Martínez Sanllorente
en Universidad de Burgos — 17 de junio
de 2024

Tutor: Dr. Carlos López Nozal y Dr. Pedro
Latorre Carmona

Índice general

Índice general	i
Índice de figuras	iii
Índice de tablas	iv
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	2
A.3. Estudio de viabilidad	10
Apéndice B Especificación de Requisitos	13
B.1. Introducción	13
B.2. Objetivos generales	13
B.3. Catálogo de requisitos	13
B.4. Especificación de requisitos	13
Apéndice C Especificación de diseño	15
C.1. Introducción	15
C.2. Diseño de datos	15
C.3. Diseño procedimental	15
C.4. Diseño arquitectónico	15
Apéndice D Documentación técnica de programación	17
D.1. Introducción	17
D.2. Estructura de directorios	17
D.3. Manual del programador	17

D.4. Compilación, instalación y ejecución del proyecto	17
D.5. Pruebas del sistema	17
Apéndice E Documentación de usuario	19
E.1. Introducción	19
E.2. Requisitos de usuarios	19
E.3. Instalación	19
E.4. Manual del usuario	19
Apéndice F Anexo de sostenibilización curricular	21
F.1. Introducción	21
Bibliografía	23

Índice de figuras

A.1. Burndown del sprint 4	5
A.2. Burndown del sprint 5	6
A.3. Burndown del sprint 6	7
A.4. Burndown del sprint 7	8

Índice de tablas

A.1. Desglose de tiempo dedicado al proyecto.	9
A.2. Costes de hardware.	10
A.3. Costes de software.	10
A.4. Costes de personal.	11
A.5. Costes totales.	11
B.1. CU-1 Nombre del caso de uso.	14

Apéndice A

Plan de Proyecto Software

A.1. Introducción

La fase de planificación es esencial en cualquier proyecto. Durante esta etapa se calculan el trabajo, el tiempo y el dinero necesarios para completar el proyecto. Para ello, se analiza detalladamente cada una de las partes que lo componen lo que permite conocer los recursos necesarios para su realización. En este anexo se describe todo este proceso.

La fase de planificación se puede dividir en dos partes:

- **La planificación temporal:** se crea un calendario o cronograma. Aquí se calcula el tiempo necesario para completar cada una de las partes del proyecto. Se debe establecer una fecha de inicio fija y una fecha de finalización estimada, considerando la importancia de cada tarea y los requisitos necesarios para comenzar a trabajar en ellas.
- **El estudio de viabilidad:** se enfoca en la factibilidad del proyecto y se divide en dos apartados.
 - **La viabilidad económica:** en esta parte se estiman los costos y los beneficios que podría traer la realización del proyecto.
 - **La viabilidad legal:** aquí es donde se tiene en cuenta el contexto del proyecto y toca aquella ley que le afecta. Se deben analizar todas aquellas que afecten al proyecto.

A.2. Planificación temporal

En líneas generales, se ha seguido la metodología Scrum, aunque de manera adaptada a mi situación como único desarrollador, he mantenido ciertos principios clave.

- Se aplicó una estrategia de desarrollo incremental mediante iteraciones (sprints) y revisiones.
- La duración media de los sprints ha sido de dos semanas.
- Aunque no he tenido reuniones diarias, se han realizado revisiones periódicas al final de cada sprint para evaluar el progreso y planificar el siguiente.
- Esto ha permitido mantener la esencia de la mejora continua y la entrega incremental de valor.

He de mencionar que no es hasta la reunión de control del 26 de marzo, que empiezo a reflejar el progreso en GitHub añadiendo todas las issues de reuniones anteriores indicando S1, S2 y S3 en el propio nombre de la issue para indicar de qué semana habían sido ya que en GitHub aparecen como que todas fueron añadidas entre el 26 y el 28 de marzo. Zube lo comienzo a utilizar para organizar mejor los sprints desde el sprint 4 debido a que Carlos me recomienda utilizar dicha herramienta para tener un mejor control de estos.

Sprint 0 (04/03/2024 - 11/03/2024)

La reunión que dio inicio a este sprint marcó el comienzo del proyecto de fin de carrera. Inicialmente, Pedro Latorre me había explicado en qué consistía la idea inicial del proyecto en una reunión previa. Esta idea la profundizamos un poco más, además de indicarme unas pautas a seguir a la hora de buscar el estado del arte en esta área y plantear unos objetivos a cumplir durante este primer sprint.

Los objetivos de la reunión del primer sprint fueron: crear un repositorio en GitHub para el proyecto, probar dos métodos de procesamiento de imágenes, específicamente el Análisis de Componentes Principales (PCA) y la Teoría Retinex, y guardar la imagen resultante permitiendo acceder al valor de cada píxel. También se acordó investigar y aplicar métodos de

agrupamiento de datos, como K-Means, en la imagen original y en la imagen procesada para comparar los resultados.

Además, se decidió eliminar el cuarto canal en las imágenes que lo tengan y evaluar críticamente los resultados obtenidos. Se enfatizó que no se esperen resultados inmediatos, sino que se debe evaluar y descartar posibles caminos de manera progresiva.

En este momento del proyecto simplemente creaba una serie de tareas a completar antes de la próxima semana por lo que no tengo como tal un registro temporal en Zube de cuando se completo cada tarea. Mas tarde estas tareas las pase el 26/03/2024 a GitHub por recomendación de mi tutor ya que así podía reflejar el trabajo realizado hasta el momento. Estas issues tienen en el nombre [S1].

Sprint 1 (11/03/2024 - 25/03/2024)

Este sprint tuvo como objetivo: mostrar la representación del histograma de los valores de los píxeles de cada canal en la imagen resultante, determinar los valores mínimos y máximos de los píxeles, y visualizar la matriz resultante utilizando imshow. Se discutió que, aunque imshow(M, []) normaliza la imagen para visualización, imwrite guarda la imagen sin normalizar. Se acordó guardar la imagen utilizando el método saveas para conservar el rango dinámico y almacenar la matriz en un archivo .mat con el resultado de la transformación. También se destacó la importancia de guardar una imagen que se pueda visualizar fácilmente y familiarizarse con los resultados de cada método aplicado.

Se decidió añadir los métodos de agrupamiento FUZZY C-Means y GMM, y probar otros métodos de agrupamiento que tengan en cuenta el contexto y no solo el valor del píxel. Además, se acordó buscar información sobre Geoffrey Machlachlan para profundizar en el tema.

Al igual que en el sprint 0, simplemente creaba una serie de tareas a completar antes de la próxima semana por lo que no tengo como tal un registro temporal en Zube y en GitHub fueron añadidas el 26/03/2024. Estas issues tienen en el nombre [S2].

Sprint 2 (25/03/2024 - 04/04/2024)

Este sprint tuvo como objetivo: pasar los algoritmos de agrupamiento de python a MATLAB ya que los tiempos de ejecución en Python el cual era el lenguaje que estaba utilizando hasta ese momento eran muy elevados. Se

decidió crear una comparación en la misma ventana dependiendo del número de centros para así poder observar las diferencias de una manera más cómoda y rápida. Además, se acordó agregar 4-5 métodos más de transformación invariante.

También se decidió generar la imagen ground truth y evaluar el resultado mediante una máscara binaria, calculando la tasa de acierto con la fórmula $(n^{\circ} \text{ de píxeles que coinciden} / n^{\circ} \text{ total de píxeles}) * 100$, y mostrar en la ventana el valor de esta tasa.

Al igual que en el sprint 0 y sprint 1, simplemente creaba una serie de tareas a completar antes de la próxima semana por lo que no tengo como tal un registro temporal en Zube y en GitHub fueron añadidas el 26/03/2024. Estas issues tienen en el nombre [S3].

Sprint 3 (04/04/2024 - 18/04/2024)

En la reunión de preparación de este sprint es cuando Carlos López se unió como cotutor al proyecto al pedírselo Pedro Latorre, este en un futuro pasaría a ser el tutor del proyecto de fin de grado. Este sprint tuvo como objetivo: familiarizarse con Overleaf y comenzar a rellenar los tres primeros apartados del documento. Además, se acordó crear una cuenta en Zube para organizar las tareas en milestones y reorganizar los directorios del proyecto. La nueva estructura incluirá un directorio "doc" con la plantilla de Ubuvirtual y un directorio para el programa MATLAB, el cual contendrá todos aquellos métodos que venía utilizando separados en distintos scripts para que de esta manera se mejora la modularidad, reutilización y mantenibilidad del código.

Otro objetivo fue la integración de MATLAB con una aplicación web, iniciando un pequeño proyecto en GitHub para gestionar esta integración. Otro objetivo fue el desarrollo de una pequeña aplicación de escritorio de MATLAB la cual servirá para probar distintos algoritmos sobre diferentes imágenes. De esta manera además de demostrar numéricamente que hay una mejoría notable tras la aplicación de métodos invariantes sobre piezas metálicas, permitirá de una forma sencilla y visual el probar distintos métodos sobre imágenes propias.

En el anterior sprint no conseguí encontrar un método de agrupamiento con contexto espacial que suponga una mejora notable en los resultados por lo que esta issue no se cerrará y formara parte también de este cuarto sprint.

Al ser este el primer sprint en el que incorpore Zube para gestionar las issues abiertas, aun no terminaba de manejar bien con la herramienta

por lo que aunque me ayudo con la gestión, no cree el sprint en el por lo que no tengo gráfico burndown de este sprint.

Sprint 4 (18/04/2024 - 02/05/2024)

Es en este sprint cuando al fin comienzo a reflejar como tal los distintos sprints en Zube. Gracias a esto, a partir de este momento me ha sido más cómodo trabajar con las issues de cada sprint. Este sprint tuvo como objetivo: mejorar las funcionalidades de la aplicación y realizar cambios menores en el texto mostrado al usuario dentro de la aplicación para evitar confusiones y corregir incoherencias. Se acordó cambiar el orden en el que se muestran los resultados de la aplicación. También se discutió la importancia de eliminar las imágenes mostradas de ejecuciones anteriores.

Añadir un selector para guardar los archivos y su correspondiente mensaje de confirmación indicando que los archivos se han guardado correctamente. Además, de buscar una solución para el problema relacionado con el sistema de directorios donde se guardan los archivos.

Finalmente, se planteó la creación de otra aplicación para poder probar con imágenes propias. Al tener una imagen ground truth, se podría obtener un porcentaje de acierto de la ejecución, lo que ayudaría a evaluar mejor los resultados obtenidos.

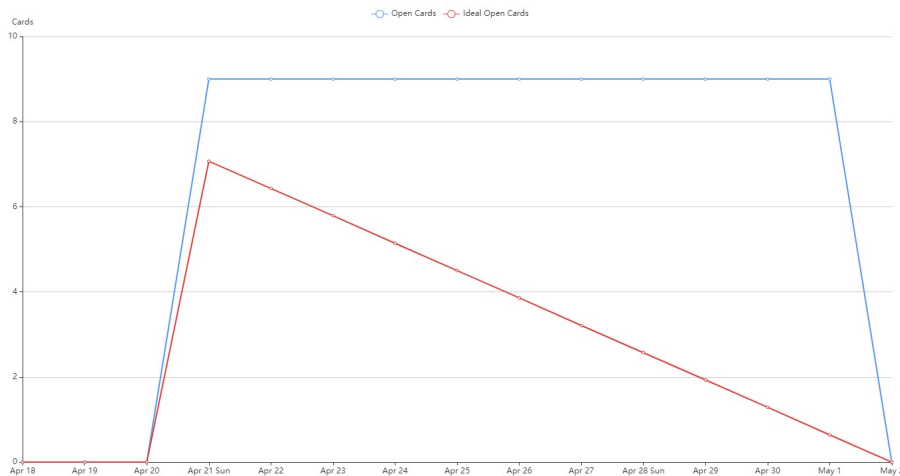


Figura A.1: Burndown del sprint 4

Como se puede ver en el gráfico burndown [A.1](#), las issues de este sprint no es hasta el ultimo día que las marco como completadas ya que en ese

momento no era consciente que las tenía que cerrar en vez de dejarlas en revisión y ya cerrarlas tras la reunión.

Sprint 5 (02/05/2024 - 23/05/2024)

Este sprint tuvo como objetivo: mejorar la experiencia del usuario y optimizar la funcionalidad de la aplicación. Implementar notificaciones periódicas sobre el estado de la ejecución, indicando que podría tomar mucho tiempo además de añadir la funcionalidad de borrar resultados anteriores. También se plantearon pequeños cambios en la interfaz, como eliminar el "de los porcentajes y añadir un botón para guardar imágenes.

Distintas modificaciones de la estructura de la interfaz, añadiendo menús de ayuda, una lista de nombres de imágenes en caché, un menú de bienvenida y ayuda contextual. Una actualización de los nombres de los archivos de resultados, añadiendo la fecha de ejecución al inicio. Además, de corregir errores en la selección de imágenes desde rutas diferentes a /data y una mejora de la gestión de errores.

Se llegó a la conclusión de agregar una caché de imágenes para almacenar temporalmente los resultados, cargando ejecuciones anteriores con los mismos parámetros para reducir el coste computacional. Al cerrar la aplicación, la caché se borraría. Finalmente, se planteó la corrección de algunos errores detectados en los algoritmos, una mejora de la documentación y el README del proyecto.

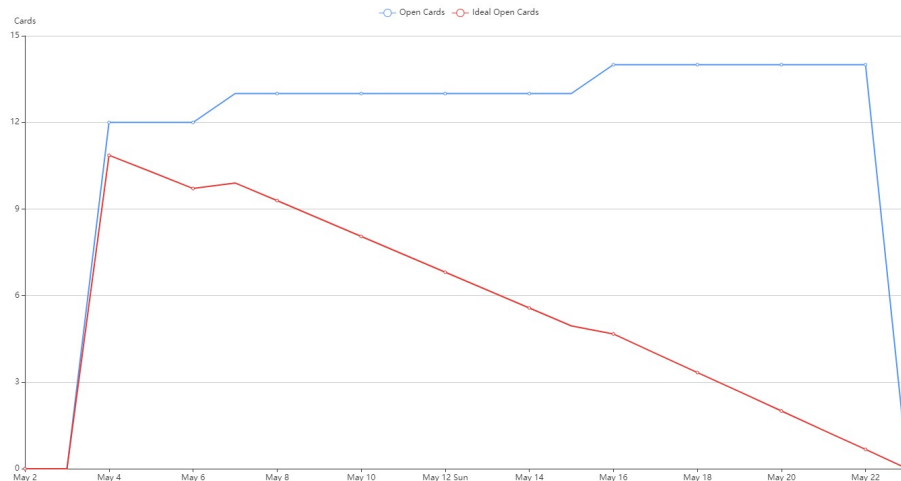


Figura A.2: Burndown del sprint 5

Como se puede ver en el gráfico burndown A.2, las issues de este sprint no es hasta el ultimo día que las marco como completadas ya que en ese momento no era consciente que las tenia que cerrar en vez de dejarlas en revisión y ya cerrarlas tras la reunión.

Sprint 6 (23/05/2024 - 30/05/2024)

Este sprint tuvo como objetivo: mejorar la funcionalidad y la usabilidad de la aplicación. Se planea realizar diversas mejoras en la documentación en HTML y crear un README en la carpeta de imágenes. Además, se modificará la visualización de los nombres de las imágenes dentro de la aplicación y se añadirán tooltips para una mejor comprensión del usuario.

Se establecerán condiciones numéricas en los campos de número de centros y se añadirá un límite de iteraciones máximo en los métodos de agrupamiento. También se trabajará en la caché para hacerla persistente, permitiendo borrar el historial y mejorando su escalabilidad.

En cuanto a la evaluación de la calidad, se incorporará una medida de comparación utilizando la imagen ground truth y se mejorará la nomenclatura de los algoritmos invariantes, reemplazando los nombres vacíos con “no aplicado”. Se abordarán problemas técnicos como la minimización de la app al cargar una imagen, la eliminación de la imagen ground truth al introducir una nueva que no tenga una asociada, y se revisarán soluciones para evitar que la aplicación se detenga por errores. Además, se tratará de introducir concurrencia para manejar tareas como el cancelar la ejecución.

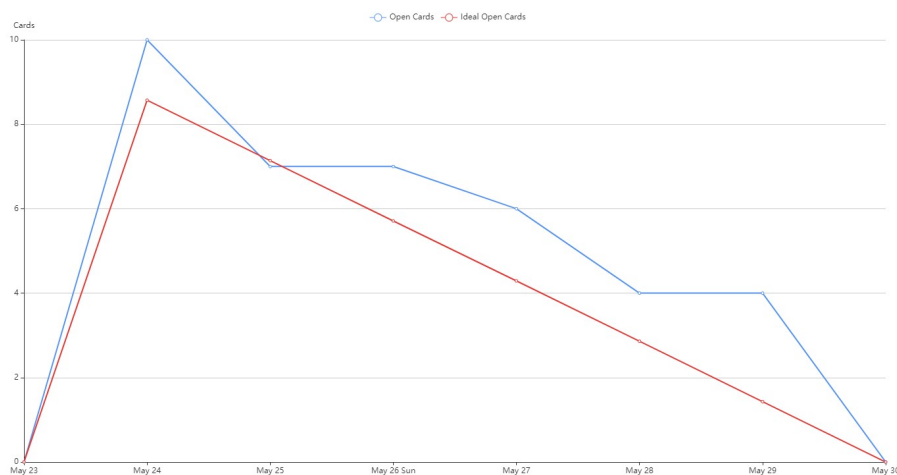


Figura A.3: Burndown del sprint 6

Como se puede ver en el gráfico burndown [A.3](#), las issues de este sprint si que se fueron cerrando correctamente a medida que se fueron completando.

Sprint 7 (31/05/2024 - 10/06/2024)

Este sprint tuvo como objetivo: preparar la primera release oficial de la aplicación. Se realizarán mejoras en la gestión de errores para asegurar una experiencia de usuario más robusta. Además, se tratará aplicar concurrencia para permitir la cancelación de la ejecución, la cual es una tarea que no se logró completar en el sprint anterior.

Se deshabilitarán los botones durante la ejecución de procesos para evitar acciones no deseadas y se revisará el código, actualizando la documentación interna para mantener la coherencia. También se implementará soporte bilingüe, añadiendo opciones en español e inglés para hacer la aplicación accesible a un público más amplio.

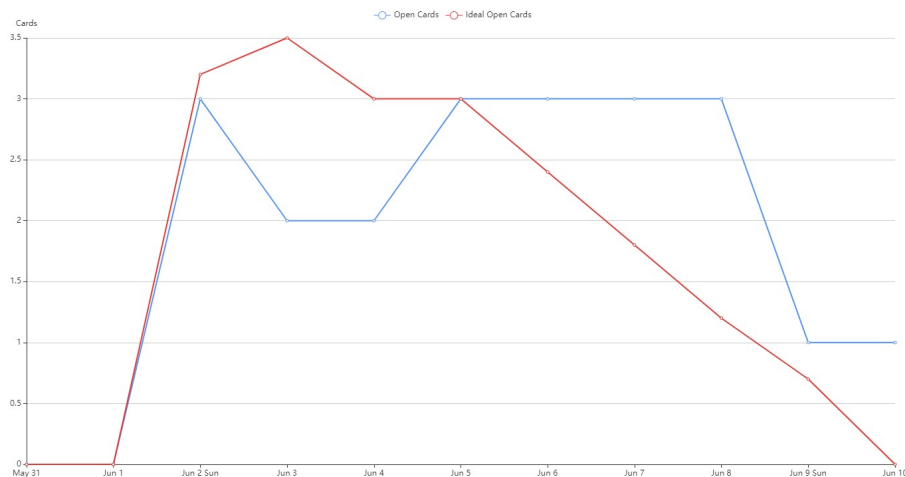


Figura A.4: Burndown del sprint 7

Como se puede ver en el gráfico burndown [A.4](#), las issues de este sprint si que se fueron cerrando correctamente a medida que se fueron completando a excepción de una que por falta de tiempo quedo a completar en el siguiente sprint.

Sprint 8 (11/06/2024 - 27/06/2024)

Este sprint tuvo como objetivo: implementar varias mejoras y preparar la aplicación para una release oficial. Se mejorarán los tooltips, especialmente

el de centros, explicando que es una agrupación de colores. Además, se añadirá la posibilidad de imprimir y visualizar la imagen seleccionada en la tabla de la caché, y se modificará el texto del botón "Abrir directorio" por "Ver imágenes". También se habilitará la opción de ordenar la caché por filas y se actualizará la imagen de bienvenida para hacerla más atractiva y profesional.

Para asegurar la compatibilidad, se adaptará la aplicación para Linux y Mac, y se realizarán pruebas unitarias para garantizar la calidad del software. Se creará un README personalizado que detalle la instalación y ejecución de la aplicación, y se reorganizarán los directorios, eliminando los que ya no se utilizan. Además, se subirá la release de la semana anterior, que incluye mejoras en el tratamiento de errores y la adaptación al inglés.

En cuanto a la documentación, se incluirán citas en la introducción de la memoria y se revisarán las correcciones en la misma. También se redactará el plan de proyecto software, la especificación de requisitos y la especificación de diseño en los anexos lo cual será la principal tarea de este sprint.

Finalmente, se realizará la release con todos los cambios implementados en este sprint, consolidando las mejoras de cara a la versión final.

Resumen del tiempo empleado

A continuación se muestra un resumen del tiempo dedicado a la realización del proyecto.

Categoría	Tiempo (h)
Sprint 0	20
Sprint 1	32
Sprint 2	20
Sprint 3	26
Sprint 4	49
Sprint 5	54
Sprint 6	28
Sprint 7	42
Sprint 8	25+
Total	296

Tabla A.1: Desglose de tiempo dedicado al proyecto.

A.3. Estudio de viabilidad

Viabilidad económica

En este apartado se analizarán los costes y beneficios que se estiman de haber sido desarrollado el proyecto en un entorno empresarial. He de mencionar que dichos datos son estimados.

Costes

Costes materiales:

En este apartado se desglosarán todos aquellos costes materiales, tanto hardware como software necesarios para el correcto desarrollo del proyecto. En ambos casos se considera que se van a amortizar a 5 años y se han utilizado durante los cuatro meses de desarrollo del proyecto.

El desglose de todos los dispositivos hardware necesarios para la realización del proyecto se encuentra a continuación en la figura A.2.

Concepto	Coste	Coste amortizado
Ordenador portátil	900,00€	60,00€
Total	900,00€	60,00€

Tabla A.2: Costes de hardware.

En cuanto a las licencias software no gratuitas se tiene el siguiente desglose en la figura A.3.

Concepto	Coste	Coste amortizado
Windows 10 Pro	259,00€	17,27€
MATLAB Academic (Anual)	262,00€	87,33€
Total	521,00€	104,60€

Tabla A.3: Costes de software.

Hay que destacar que la licencia de MATLAB utilizada ha sido la Academic que la propia Universidad de Burgos ofrece a los alumnos, la cual se paga anualmente.

Costes de personal:

El proyecto lo ha llevado a cabo un ingeniero informático recién graduado a tiempo completo durante cuatro meses. Se considera que el salario bruto

anual es de 24.000,00€ [2], el IRPF sobre el salario bruto es del 13,55 % y la Seguridad Social es del 6,35 % respectivamente como se puede ver en la figura A.4.

Concepto	Coste
Salario anual en bruto	24.000,00€
IRPF	3.251,04€
Seguridad Social	1.524,00€
Salario anual neto	19.224,96€
Total 4 meses	8.000,00€

Tabla A.4: Costes de personal.

Costes totales:

El sumatorio de todos los distintos tipos de costes se encuentra en la figura A.5.

Concepto	Coste
Hardware	60,00€
Software	104,60€
Personal	8.000,00€
Total	8.164,60€

Tabla A.5: Costes totales.

Beneficios

La aplicación desarrollada se distribuirá de forma libre y sin restricciones, permitiendo su uso sin limitaciones adicionales. La única condición necesaria para su funcionamiento será disponer de MATLAB. En una fase posterior, se evaluará la posibilidad de incorporar características adicionales que mejoren la experiencia del usuario, manteniendo siempre el enfoque en la accesibilidad y la libertad de uso.

Viabilidad legal

Apartado dedicado al marco legal que envuelve el proceso de desarrollo del proyecto, su distribución, modificación y utilización. No es necesario atender al uso de licencias de terceros, pues todo el proyecto ha sido desarrollado con la licencia MATLAB Academic proporcionada por la Universidad de

Burgos para desarrolladores individuales o estudiantes y bajo una finalidad no comercial. En concreto, la licencia utilizada ha sido:

- **MATLAB Academic:** Al tratarse de un desarrollador individual, el uso de la licencia MATLAB Academic está justificado. Esta licencia es distribuida de forma gratuita por la Universidad de Burgos a sus alumnos.

El proyecto actual ha sido asignado bajo una licencia Creative Commons [1]. Esto implica que cualquier usuario es libre de compartir, redistribuir y adaptar la aplicación, siempre y cuando se cumplan las siguientes condiciones:

- **Atribución:** Se debe dar crédito apropiado al autor original, proporcionando un enlace a la licencia e indicando si se han realizado cambios.
- **NoComercial:** No se puede utilizar el material para fines comerciales.
- **CompartirIgual:** Si se remezcla, transforma, o crea a partir del material, se debe distribuir el material modificado bajo la misma licencia que el original.

Esta licencia asegura que la aplicación mantenga su accesibilidad y flexibilidad, fomentando la colaboración y el desarrollo comunitario mientras protege los derechos del autor original.

Apéndice B

Especificación de Requisitos

B.1. Introducción

Una muestra de cómo podría ser una tabla de casos de uso:

B.2. Objetivos generales

B.3. Catálogo de requisitos

B.4. Especificación de requisitos

CU-1	Ejemplo de caso de uso
Versión	1.0
Autor	Alumno
Requisitos asociados	RF-xx, RF-xx
Descripción	La descripción del CU
Precondición	Precondiciones (podría haber más de una)
Acciones	<ol style="list-style-type: none"> 1. Pasos del CU 2. Pasos del CU (añadir tantos como sean necesarios)
Postcondición	Postcondiciones (podría haber más de una)
Excepciones	Excepciones
Importancia	Alta o Media o Baja...

Tabla B.1: CU-1 Nombre del caso de uso.

Apéndice C

Especificación de diseño

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

- D.1. Introducción
- D.2. Estructura de directorios
- D.3. Manual del programador
- D.4. Compilación, instalación y ejecución del proyecto
- D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario

Apéndice F

Anexo de sostenibilización curricular

F.1. Introducción

Este anexo incluirá una reflexión personal del alumnado sobre los aspectos de la sostenibilidad que se abordan en el trabajo. Se pueden incluir tantas subsecciones como sean necesarias con la intención de explicar las competencias de sostenibilidad adquiridas durante el alumnado y aplicadas al Trabajo de Fin de Grado.

Más información en el documento de la CRUE https://www.crue.org/wp-content/uploads/2020/02/Directrices_Sostenibilidad_Crue2012.pdf.

Este anexo tendrá una extensión comprendida entre 600 y 800 palabras.

Bibliografía

- [1] CC BY-NC-SA 4.0 Deed | Atribución-NoComercial-CompartirIgual 4.0 Internacional | Creative Commons — creativecommons.org. <https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>. [Accessed 17-06-2024].
- [2] Glassdoor. Sueldos para el puesto de Ingeniero Informático en España. https://www.glassdoor.es/Sueldos/ingeniero-inform%C3%A1tico-sueldo-SRCH_K00,21.htm. [Accessed 16-06-2024].