



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



TFG del Grado en Ingeniería  
Informática  
AI-Music



Presentado por José Ángel López Estrada  
en Universidad de Burgos — 25 de junio  
de 2023

Tutor: César Ignacio García Osorio, Alicia  
Olivares Gil



---

# Índice general

---

<b>Índice general</b>	<b>i</b>
<b>Índice de figuras</b>	<b>iii</b>
<b>Índice de tablas</b>	<b>iv</b>
<b>Apéndice A Plan de Proyecto Software</b>	<b>1</b>
A.1. Introducción . . . . .	1
A.2. Planificación temporal . . . . .	1
A.3. Estudio de viabilidad . . . . .	2
<b>Apéndice B Especificación de Requisitos</b>	<b>3</b>
B.1. Introducción . . . . .	3
B.2. Objetivos generales . . . . .	3
B.3. Catalogo de requisitos . . . . .	3
B.4. Especificación de requisitos . . . . .	5
<b>Apéndice C Especificación de diseño</b>	<b>9</b>
C.1. Introducción . . . . .	9
C.2. Diseño de datos . . . . .	9
C.3. Diseño procedimental . . . . .	9
C.4. Diseño arquitectónico . . . . .	9
<b>Apéndice D Documentación técnica de programación</b>	<b>11</b>
D.1. Introducción . . . . .	11
D.2. Estructura de directorios . . . . .	11
D.3. Manual del programador . . . . .	11

D.4. Compilación, instalación y ejecución del proyecto . . . . .	12
D.5. Pruebas del sistema . . . . .	12
<b>Apéndice E Documentación de usuario</b>	<b>13</b>
E.1. Introducción . . . . .	13
E.2. Requisitos de usuarios . . . . .	13
E.3. Instalación . . . . .	13
E.4. Manual del usuario . . . . .	13

---

## Índice de figuras

---

---

# Índice de tablas

---

B.1. CU-1 Carga de la canción . . . . .	6
B.2. CU-2 Extracción de características de audio . . . . .	7
B.3. CU-3 Ejecución del modelo de machine learning . . . . .	8

## *Apéndice A*

---

# Plan de Proyecto Software

---

### **A.1. Introducción**

Este plan de proyecto de software se refiere al desarrollo de una aplicación que clasifica automáticamente los géneros musicales basándose en características de audio. Este proyecto ha sido concebido como un trabajo de fin de grado y ha implicado la implementación de técnicas de aprendizaje automático y procesamiento de señales de audio digitales para lograr el objetivo. La aplicación busca proporcionar una solución efectiva para clasificar pistas musicales de forma automática.

En este apéndice se describirá la planificación sobre la que se ha desarrollado el proyecto.

### **A.2. Planificación temporal**

El desarrollo del proyecto se organizó siguiendo la metodología ágil de SCRUM, con iteraciones semanales o «sprints». Cada sprint implicó una serie de actividades que culminaron con una entrega incremental del proyecto.

**Sprints totales: 14**

**Sprint 1 (08/03/2023 - 15/03/2023)**

**Sprint 2 (15/03/2023 - 22/03/2023)**

**Sprint 3 (22/03/2023 - 05/04/2023)**

**Sprint 4 (05/04/2023 - 12/04/2023)**

**Sprint 5 (12/04/2023 - 19/04/2023)**

**Sprint 6 (19/04/2023 - 26/04/2023)**

**Sprint 7 (26/04/2023 - 10/05/2023)**

**Sprint 8 (10/05/2023 - 17/05/2023)**

**Sprint 9 (17/05/2023 - 24/05/2023)**

**Sprint 10 (24/05/2023 - 02/06/2023)**

**Sprint 11 (02/06/2023 - 09/06/2023)**

**Sprint 12 (09/06/2023 - 16/06/2023)**

**Sprint 13 (16/06/2023 - 23/06/2023)**

**Sprint 14 (23/06/2023 - 30/06/2023)**

### **A.3. Estudio de viabilidad**

**Viabilidad económica**

**Viabilidad legal**



## *Apéndice B*

---

# Especificación de Requisitos

---

## B.1. Introducción

## B.2. Objetivos generales

- Desarrollar un sistema de reconocimiento de estilos musicales utilizando inteligencia artificial.
- Diseñar e implementar una aplicación web que permita usar el modelo de una forma sencilla.
- Obtener conclusiones y conocimiento a partir de los datos.

## B.3. Catalogo de requisitos

### Requisitos funcionales

Un requisito funcional es una especificación que describe lo que un sistema debe hacer o cómo debe comportarse. Los requisitos funcionales pueden incluir detalles como cálculos, manipulación de datos, interacción con el usuario, etc. A continuación se listan los requisitos funcionales extraídos para este proyecto:

- **RF-1 Predicción de estilo musical:** la aplicación debe ser capaz de predecir el estilo musical de una canción.
  - **RF-1.1 Extracción de características de audio:** la aplicación debe ser capaz de extraer características relevantes de la canción que se utilizarán para predecir el estilo musical.

- **RF-1.2 Implementar un modelo de machine learning para realizar la predicción:** la aplicación debe implementar un modelo de machine learning que se entrenará con las características de un conjunto de datos musical.
- **RF-2 Presentación de la información:** la aplicación debe mostrar información relevante sobre la predicción realizada.
  - **RF-2.1 Detalles de la predicción:** la aplicación debe ser capaz de mostrar datos sobre los detalles de la predicción como la confianza (probabilidad) o posibles géneros musicales alternativos.
  - **RF-2.2 Información de la canción:** la aplicación debe mostrar información relevante sobre la pista de audio o canción utilizada como la duración, formato o frecuencia de muestreo.
- **RF-3 Implementación de una interfaz de usuario:** la aplicación debe contener una interfaz de usuario para facilitar su uso.
  - **RF-3.1 Funcionalidad de carga de archivos:** la interfaz de usuario debe incluir una función de carga de archivos que permita seleccionar y cargar una canción para realizar la predicción.
  - **RF-3.2 Representación de los resultados:** la interfaz de usuario debe incluir sección donde se presenten de forma clara los resultados de la predicción.

## Requisitos no funcionales

Los requisitos no funcionales se centran en las características del sistema que no están directamente relacionadas con su comportamiento, por ejemplo rendimiento o seguridad. A continuación se listan los requisitos no funcionales extraídos para este proyecto:

- **RNF-1 Rendimiento:** la aplicación debe ser capaz de procesar y analizar una canción y devolver los resultados dentro de un tiempo razonable, idealmente unos pocos segundos.
- **RNF-2 Seguridad:** la aplicación debe cumplir con protocolos de seguridad como cifrado de comunicaciones o la eliminación de la canción después de su procesamiento.
- **RNF-3 Compatibilidad:** la aplicación debe ser compatible con diversos navegadores y sistemas operativos.
- **RNF-4 Usabilidad:** la aplicación debe ser fácil de usar para el usuario. El usuario debería ser capaz de realizar el proceso de subida de archivos y comprobar la predicción y resultados de una forma sencilla.

- **RNF-5 Portabilidad:** la aplicación debe ser compatible con diversos dispositivos como PCs, tablets o dispositivos móviles.

## B.4. Especificación de requisitos

Los casos de uso son descripciones detalladas del funcionamiento de una parte del sistema desde la perspectiva del usuario.

En esta sección se mostrará el diagrama general de casos de uso y se explicará en detalle cada uno de ellos.

### Diagrama general de casos de uso

Insertar diagrama general de casos de uso.

### Casos de uso

CU-1	Carga de la canción
<b>Versión</b>	1.0
<b>Autor</b>	José Ángel López
<b>Requisitos asociados</b>	RF-1.1, RF-3.1
<b>Descripción</b>	El usuario utiliza la función de carga de archivo para seleccionar y cargar una canción en la aplicación.
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. La aplicación se encuentra en funcionamiento.</li> <li>2. El usuario tiene acceso a los archivos de música que desea cargar.</li> </ol>
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona la opción para cargar una canción pulsando el botón <i>Upload File</i>.</li> <li>2. El usuario busca y selecciona la canción que desea cargar desde su almacenamiento local.</li> <li>3. La canción se carga en la aplicación.</li> </ol>
<b>Postcondiciones</b>	<ol style="list-style-type: none"> <li>1. La canción se encuentra cargada en la aplicación.</li> <li>2. La aplicación está lista para extraer características y realizar el proceso de predicción de la canción cargada.</li> </ol>
<b>Excepciones</b>	<ol style="list-style-type: none"> <li>1. El archivo cargado no es un fichero de audio.</li> <li>2. La canción tiene un formato incompatible.</li> </ol>
<b>Importancia</b>	Alta

Tabla B.1: CU-1 Carga de la canción

CU-2	Extracción de características de audio.
<b>Versión</b>	1.0
<b>Autor</b>	José Ángel López
<b>Requisitos asociados</b>	RF-1.1
<b>Descripción</b>	La aplicación extrae las características relevantes del audio para su posterior análisis.
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. La canción se ha cargado correctamente en la aplicación.</li> </ol>
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. La aplicación inicia el proceso de extracción de características del archivo de audio cargado.</li> </ol>
<b>Postcondiciones</b>	<ol style="list-style-type: none"> <li>1. Las características relevantes de la canción se han extraído correctamente.</li> <li>2. La aplicación está lista para utilizar las características extraídas para realizar la predicción del estilo musical.</li> </ol>
<b>Excepciones</b>	<ol style="list-style-type: none"> <li>1. Ocurrió un error durante la extracción de las características del audio.</li> </ol>
<b>Importancia</b>	Alta

Tabla B.2: CU-2 Extracción de características de audio

CU-3	Ejecución del modelo de machine learning
<b>Versión</b>	1.0
<b>Autor</b>	José Ángel López
<b>Requisitos asociados</b>	RF-1.2
<b>Descripción</b>	La aplicación ejecuta el modelo de machine learning para realizar la predicción del estilo musical según las características extraídas.
<b>Precondiciones</b>	<ol style="list-style-type: none"> <li>1. Las características de audio de la canción se han extraído correctamente.</li> <li>2. Las características de audio de la canción son suficientes para realizar el proceso de predicción.</li> </ol>
<b>Acciones</b>	<ol style="list-style-type: none"> <li>1. El modelo procesa las características extraídas de la canción.</li> <li>2. Se obtiene el resultado de la predicción, el cual consiste en un vector de probabilidades entre diferentes estilos musicales.</li> </ol>
<b>Postcondiciones</b>	<ol style="list-style-type: none"> <li>1. La aplicación ha generado una predicción sobre el estilo musical de la canción.</li> <li>2. La aplicación está lista para mostrar los detalles de la predicción al usuario.</li> </ol>
<b>Excepciones</b>	<ol style="list-style-type: none"> <li>1. El modelo no puede procesar las características extraídas.</li> </ol>
<b>Importancia</b>	Alta

Tabla B.3: CU-3 Ejecución del modelo de machine learning

## *Apéndice C*

---

# **Especificación de diseño**

---

### **C.1. Introducción**

Cómo se han resuelto e implementado los requisitos anteriormente mencionados. Definición de datos, estructuras, directorios, clases, etc.

### **C.2. Diseño de datos**

Tipos de datos con los que se van a trabajar.

INSERTAR DIAGRAMA DE CLASES, RELACIONES, E/R, ETC.

### **C.3. Diseño procedimental**

DISEÑO PROCEDIMENTAL SIGNIFICA CÓMO SE HAN DISEÑADO LOS PROCEDIMIENTOS DE LA APLICACIÓN (CÓMO LOS MÉTODOS, FUNCIONES, ETC.) MOSTRAR CADA PROCEDIMIENTO EN PSEUDOCODIGO O LENGUAJE NATURAL. diagramas de interacción, secuencia, etc.

### **C.4. Diseño arquitectónico**

MODELO ARQUITECTONICO PREVISTO, PATRONES DE DISEÑO. Por ejemplo decir si hemos usado patrón vista controlador, patrón MVM, etc. herencias, genericidad, clases, etc.

EXPLICAR LA FORMA GENERAL DE LA ARQUITECTURA DE LA APLICACIÓN (CON DIAGRAMAS, FLECHAS, ETC.)

INSERTAR NUEVA SECCIÓN CON DISEÑO DE INTERFACES SI PROCEDE.



## *Apéndice D*

---

# **Documentación técnica de programación**

---

## **D.1. Introducción**

En esta sección se describe la documentación técnica de programación, incluyendo la instalación del entorno de desarrollo, estructura de la aplicación como la estructura de carpetas, clases, etc. Compilación, configuración e servicios de integración utilizados y TESTSSS.

## **D.2. Estructura de directorios**

Insertar estructura de directorios y carpetas y subcarpetas de la aplicación. Explicar en detalle que hace cada archivo, carpeta, etc.

## **D.3. Manual del programador**

Manual creado con el objetivo de que los futuros programadores que trabajen en esta aplicación sepan como funciona todo, como montar el entorno de desarrollo, como obtener el código fuente, etc. INCLUIRCAPTURAS CON HERRAMIENTAS Y ENTORNO DE TRABAJO.

## **D.4. Compilación, instalación y ejecución del proyecto**

compilación y ejecución del proyecto en una máquina propia del posible desarrollador futuro. Incluir todo lo necesario como capturas de pantalla e indicaciones donde se vea todo.

## **D.5. Pruebas del sistema**

Docuemntar tests realizados, captura de pantalla con los tests en verde, captura del buen funcionamiento de los componentes, etc.

## *Apéndice E*

---

# Documentación de usuario

---

### **E.1. Introducción**

Manual del usuario, como utilizar la aplicación, requisitos del sistema para el funcionamiento local, etc.

### **E.2. Requisitos de usuarios**

Requisitos que debe tener el sistema del usuario para utilizar el programa, o requisitos del navegador en el caso de que la aplicación este subida a internet, etc.

### **E.3. Instalación**

Como instalar el programa en el pc o como acceder a la página web etc.

### **E.4. Manual del usuario**

Descripción de como usar diferentes funcionalidades de la aplicación.

Como subir canciones. Como analizar canciones. Ver características. Cambiar algoritmo ETC.