



Asignación de características musicales para ambientación comercial

Autor:

Ing. Carlos Alberto Rivas Araque

Director:

Esp. Ing. Martín Moreyra (GatekeeperX)

*Esta planificación fue realizada en el curso de Gestión de proyectos
entre el 26 de agosto de 2025 y el 14 de octubre de 2025.*

Índice

1. Descripción técnica-conceptual del proyecto a realizar.	5
2. Identificación y análisis de los interesados	6
3. Propósito del proyecto	7
4. Alcance del proyecto	7
5. Supuestos del proyecto.	8
6. Product Backlog	9
7. Criterios de aceptación de historias de usuario	11
8. Fases de CRISP-DM	13
9. Desglose del trabajo en tareas	14
10. Planificación de Sprints	16
11. Diagrama de Gantt (sprints)	17
12. Normativa y cumplimiento de datos (gobernanza)	18
13. Gestión de riesgos	18
14. Sprint Review	21
15. Sprint Retrospective	22

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
0	Creación del documento	26 de agosto de 2025
1	Se completa hasta el punto 5 inclusive	9 de septiembre de 2025
2	Se completa hasta el punto 9 inclusive	16 de septiembre de 2025
3	Se completa hasta el punto 12 inclusive	23 de septiembre de 2025
4	Se completa hasta el punto 15 inclusive	30 de septiembre de 2025

Acta de constitución del proyecto

Buenos Aires, 26 de agosto de 2025

Por medio de la presente se acuerda con el Ing. Carlos Alberto Rivas Araque que su Trabajo Final de la Carrera de Especialización en Inteligencia Artificial se titulará “Asignación de características musicales para ambientación comercial” y consistirá en la implementación de un sistema de generación de características musicales a partir de canciones en el contexto de una aplicación de ambientación musical. El trabajo tendrá un presupuesto preliminar estimado de 600 horas y un costo estimado de 18750 dólares estadounidenses, con fecha de inicio el 26 de agosto de 2025 y fecha de presentación pública en junio de 2026.

Se adjunta a esta acta la planificación inicial.

Dr. Ing. Ariel Lutenberg
Director posgrado FIUBA

Carolina Arbelaez
Plusyc Live SAS

Esp. Ing. Martín Moreyra
Director del Trabajo Final

1. Descripción técnica-conceptual del proyecto a realizar

El proyecto se hace para la empresa Plusyc Live SAS, dedicada a la creación de experiencias musicales personalizadas para locales comerciales a través de una aplicación móvil. La aplicación utiliza un *pool* de canciones propio, y un *backend* que crea listas de reproducción. Las listas se generan gracias a características musicales, tales como el género, ánimo, energía y tempo de cada canción. Actualmente, la asignación de características musicales depende del acceso a interfaces de programación de aplicaciones (API) de terceros, lo que genera restricciones de disponibilidad y costos. Esto afecta la escalabilidad y autonomía del sistema, y en consecuencia, la calidad de la experiencia ofrecida a los clientes.

Se propone el desarrollo de un sistema de inteligencia artificial (IA) que reciba como entrada canciones y genere como salida sus características musicales. Se entrenarán y evaluarán modelos de aprendizaje supervisado, a partir de técnicas y representaciones numéricas del sonido, fuentes de datos privadas de la empresa, y datos públicos disponibles. Y se podrán inferir las características de nuevas canciones sin depender de proveedores externos.

En el ámbito de la inteligencia artificial aplicada a la música, existen herramientas que permiten extraer información musical a bajo nivel, y herramientas en línea que permiten extraer algunas características específicas. Sin embargo, no se cuenta con una solución que provea resultados completos y resuelva el problema de la disponibilidad.

La propuesta de valor radica en otorgar a Plusyc Live SAS una solución completa, integrada y altamente disponible, en pro de la escalabilidad y la independencia tecnológica. Los comercios tendrán una renovación en la experiencia de sus consumidores. El sistema será además reutilizable en otros proyectos de la empresa.

A continuación, en la figura 1, se presenta el diagrama de bloques de alto nivel de la aplicación. A la derecha del diagrama se observa el sistema basado en IA, que es el foco del proyecto.

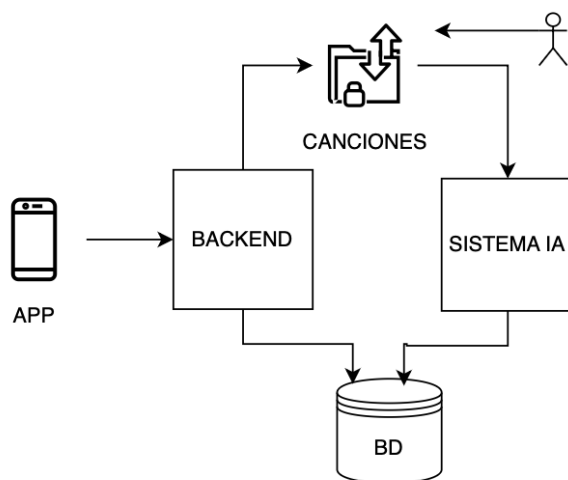


Figura 1. Diagrama en bloques del sistema.

En la figura se puede observar un actor que provee los archivos de las canciones que son las entradas del sistema IA, donde se procesa cada canción para extraer sus características musicales. Las características son las salidas, y son almacenadas en una base de datos que alimenta el *backend* de la aplicación. Ahora, en la figura 2, se muestra un diagrama de bloques más detallado del sistema IA.

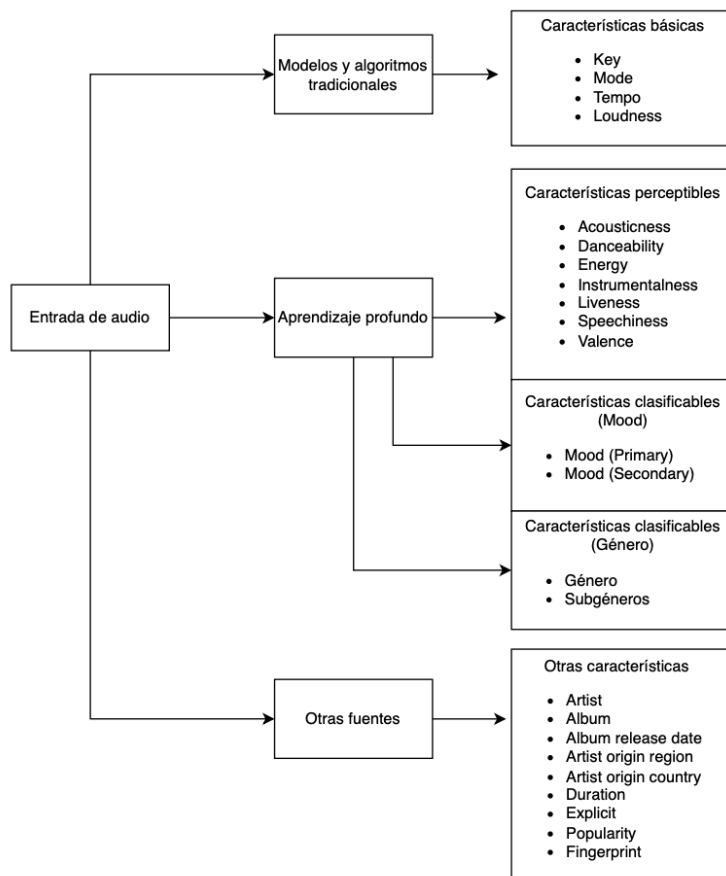


Figura 2. Diagrama de bloques del sistema IA.

En la figura se observa que el sistema IA utiliza tres módulos principales: “Modelos y algoritmos tradicionales”, “Aprendizaje profundo” y “Otras fuentes”. Cada módulo procesa el archivo de audio y se encarga de inferir o consultar un subconjunto específico de características musicales.

2. Identificación y análisis de los interesados

- Cliente: Carolina Arbelaez, es la encargada de aprobar los entregables.
- Impulsor: el Ing. Andrés Saldarriaga, es el responsable de brindar acceso a los datos del cliente, y las herramientas tecnológicas necesarias en producción, como credenciales, y proveedores de infraestructura en la nube.
- Responsable: Ing. Carlos Alberto Rivas Araque, es quien llevará a cabo el desarrollo del proyecto.

- Orientador: Esp. Ing. Martín Moreyra, es experto en solución de problemas de inteligencia artificial y va a ser consultor y ayuda para resolver problemas puntuales en caso de *blockers* técnicos.
- Usuario final: el usuario final son los clientes de Plusyc Live SAS, que utilizan la aplicación para ambientar su comercio con listas personalizadas.

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Carolina Arbelaez	Plusyc Live SAS	C.E.O.
Impulsor	Ing. Andrés Saldarriaga	Plusyc Live SAS	C.T.O.
Responsable	Ing. Carlos Alberto Rivas Araque	FIUBA	Alumno
Orientador	Esp. Ing. Martín Moreyra	GatekeeperX	Director del Trabajo Final
Usuario final	-	-	Locales comerciales

3. Propósito del proyecto

Desarrollar un sistema de inteligencia artificial capaz de caracterizar automáticamente canciones a partir de su archivo de audio, con el fin de reducir la dependencia de servicios externos y garantizar la escalabilidad, autonomía y confiabilidad de la aplicación de Plusyc Live SAS. Con ello, se busca fortalecer la experiencia musical personalizada que la empresa ofrece a los comercios y abrir la posibilidad de aplicar esta tecnología en otros contextos dentro del sector musical.

4. Alcance del proyecto

Se definen los siguientes subconjuntos de características musicales:

- Básicas: {*Key, Mode, Tempo, Loudness*}.
- Perceptibles: {*Danceability, Energy, Speechiness, Acousticness, Instrumentalness, Liveness, Valence*}.
- Clasificables por ánimo (*Mood*): {*Primary Mood, Secondary Mood*}.
- Clasificables por género: {*Genre, Subgenres*}.
- Otras (Aquellas que no requieren de IA para su asignación): {*Artist, Album, Album release date, Artist origin region, Artist origin country, Duration, Explicit, Popularity, Fingerprint*}.

El proyecto incluye la asignación de las características básicas, perceptibles y las clasificables por ánimo y género. Para lograr esto es necesario:

- Revisión del estado del arte sobre *datasets* musicales disponibles para modelos de aprendizaje supervisado (algoritmos tradicionales) y aprendizaje profundo.

- Evaluación de los datos del cliente para verificar volumen, calidad de asignaciones y posibilidad de conformar un *dataset* propio.
- Análisis de audio, representaciones numéricas de señales de audio para depurar los *datasets*.
 - Extracción de representaciones tradicionales para los modelos supervisados.
 - Generación de representaciones intermedias para aprendizaje profundo.
- Depuración de los *datasets* para los modelos de aprendizaje supervisado tradicionales.
- Elección de modelos de aprendizaje supervisado y algoritmos tradicionales para la predicción de las características básicas.
- Evaluación de los *datasets* para modelos de aprendizaje profundo.
- Selección de modelos de aprendizaje profundo y modelos preentrenados para la inferencia de características perceptibles, y clasificables por ánimo y género.
- Implementación de los modelos seleccionados.
- Construcción de un *pipeline* para la asignación de características mediante el uso de los modelos seleccionados como los mejores.
- Almacenamiento de las características de salida en la base de datos.

El presente proyecto no incluye:

- Asignación del subconjunto de otras características que no requieren de IA para su asignación.
- Desarrollo de una interfaz para la recepción de un nuevo archivo de audio.
- Desarrollo de interfaces gráficas de usuario.
- Puesta en producción del sistema en un entorno del cliente.
- Mantenimiento y monitoreo a largo plazo.

5. Supuestos del proyecto

Para el desarrollo del presente proyecto se supone:

- Disponibilidad de recursos de *hardware*: se dispondría de acceso a instancias de procesamiento locales, y en la nube con los suficientes recursos para las fases de entrenamiento y validación de los modelos.
- Disponibilidad de datos para el entrenamiento del modelo de procesamiento de audio: se tendría acceso a un conjunto diverso de datos de audio que cubrirán una amplia gama de géneros y estilos.
- Factibilidad técnica: se asume que las tecnologías actuales de procesamiento de audio son lo suficientemente avanzadas para implementar el proyecto. Y en particular.

- Las características básicas se pueden generar usando modelos de aprendizaje de máquina y algoritmos tradicionales.
- Las perceptibles y las clasificables por ánimo y género se pueden predecir usando un modelos de aprendizaje profundo de múltiples salidas.
- Tiempo: se estima que las 600 horas asignadas para el desarrollo del proyecto serán suficientes para completar todas las etapas, incluyendo la planeación, análisis y diseño, desarrollo, pruebas y potenciales correcciones del sistema.
- Asistencia de los interesados: se contaría con la ayuda del cliente ante *blockers* técnicos, y dudas sobre los requisitos del sistema.

6. Product Backlog

Se definen los siguientes roles:

- Usuario: la persona del lado del cliente con conocimientos en características musicales, para verificar que las salidas son correctas.
- Desarrollador de software con conocimiento en inteligencia artificial: la persona ocupando este rol deberá tener conocimiento suficiente en procesamiento de audio, y entrenamiento de modelos de inteligencia artificial, para la generación de características musicales.
- Cliente: es el encargado de verificar que el proyecto cumple con las especificaciones, y dar la aprobación final del sistema.

Épicas, historias de usuario (HU) y *spikes*:

Se llevarán a cabo sprints de 2 semanas, y cada sprint tendrá una dedicación de aproximadamente 40 horas. Para la estimación del total de Story Points (SP) se usa la serie de Fibonacci. Se suman las ponderaciones de dificultad, complejidad e incertidumbre, y se aproxima al inmediato superior de la serie.

■ Selección de *datasets* y técnicas de representación de audio

- **Spike 1:** como desarrollador de IA, quiero encontrar *datasets* musicales públicos, y técnicas de representación de audio para ser usados en algoritmos y modelos tradicionales.
Prioridad máxima, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **Spike 2:** como desarrollador de IA, quiero determinar cuales *datasets* musicales públicos, y técnicas de representación de audio, se usarán para aprendizaje profundo.
Prioridad máxima, 2 SP (complejidad: 1, dificultad: 1, incertidumbre: 0)
- **Spike 3:** como desarrollador de IA, quiero evaluar los datos del cliente, en términos de volumen y calidad, para determinar si es posible conformar un *dataset* propio.
Prioridad alta, 2 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)

■ **Desarrollo de algoritmos y modelos tradicionales para características básicas**

- **HU 1:** como desarrollador de IA, quiero realizar un análisis exploratorio de datos (EDA) de las características básicas, para entender su distribución.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)
- **HU 2:** como desarrollador de IA, quiero preparar los datos para que sean aptos para el modelado.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)
- **HU 3:** como desarrollador de IA, quiero entrenar modelos tradicionales para predecir cada una de las características básicas.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)
- **HU 4:** como desarrollador de IA, quiero evaluar los resultados del modelo con métricas apropiadas, para validar su calidad, y seleccionar el mejor.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)

■ **Desarrollo de modelos de aprendizaje profundo para características perceptibles**

- **HU 5:** como desarrollador de IA, quiero realizar un EDA de los datos de las características perceptibles, para entender su distribución y correlaciones.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)
- **HU 6:** como desarrollador de IA, quiero preparar los datos de audio para entrenar un modelo.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)
- **HU 7:** como desarrollador de IA, quiero entrenar un modelo multi-salida de aprendizaje profundo para predecir las características perceptibles.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)
- **HU 8:** como desarrollador de IA, quiero evaluar los resultados del modelo con métricas apropiadas para validar su desempeño.
Prioridad alta, 5 SP (complejidad: 2, dificultad: 2, incertidumbre: 1)

■ **Desarrollo de modelos de aprendizaje profundo para características clasificables por ánimo**

- **HU 9:** como desarrollador de IA, quiero realizar un EDA de los datos de las características clasificables por ánimo, para analizar su distribución y categorías.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **HU 10:** como desarrollador de IA, quiero preparar los datos para el modelado de *moods*.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **HU 11:** como desarrollador de IA, quiero entrenar un modelo multi-salida de aprendizaje profundo para predecir *primary* y *secondary mood*.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **HU 12:** como desarrollador de IA, quiero evaluar el modelo con métricas de clasificación multi-clase para validar su desempeño.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)

■ **Desarrollo de modelos de aprendizaje profundo para características clasificables por género**

- **HU 13:** como desarrollador de IA, quiero realizar un EDA de los datos de géneros y subgéneros para entender su distribución y categorías.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **HU 14:** como desarrollador de IA, quiero preparar los datos para modelado de género.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **HU 15:** como desarrollador de IA, quiero entrenar un modelo multi-salida de aprendizaje profundo para predecir el género y los subgéneros.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)
- **HU 16:** como desarrollador de IA, quiero evaluar los resultados del modelo con métricas de clasificación, para validar su desempeño.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)

■ **Implementación del *pipeline* de procesamiento**

- **HU 17 - opcional:** como cliente, quiero una interfaz para la recepción de un nuevo archivo de audio, para alimentar el *pipeline* que genere las características musicales.
Prioridad baja, 8 SP (complejidad: 3, dificultad: 2, incertidumbre: 1)
- **HU 18:** como cliente, quiero un *pipeline* que procese los modelos desarrollados para automatizar la generación de características musicales, y almacenar los resultados en una base de datos.
Prioridad media, 8 SP (complejidad: 3, dificultad: 2, incertidumbre: 1)
- **HU 19 - opcional:** como cliente, quiero desplegar el sistema en el *backend*, para automatizar la generación de características musicales del audio en producción.
Prioridad baja, 13 SP (complejidad: 5, dificultad: 3, incertidumbre: 1)
- **HU 20:** como usuario, quiero tener acceso de lectura a la base de datos para validar la calidad de las características resultantes.
Prioridad alta, 3 SP (complejidad: 1, dificultad: 1, incertidumbre: 1)

7. Criterios de aceptación de historias de usuario

■ **Selección de *datasets* y técnicas de representación de audio**

- **Spike 1**
 - Se genera un documento que incluye los hallazgos de *datasets* públicos, con ventajas y desventajas.
 - Se listan y detallan las representaciones de audio disponibles.
 - Se dispone de criterios de uso de los *datasets* y representaciones de audio para el conjunto de características básicas.
- **Spike 2**
 - Se extiende el documento anterior con otros *datasets* específicos en el marco del aprendizaje profundo.
 - Se agregan las representaciones de audio disponibles en este marco.

- Se añaden los criterios de uso de los *datasets* y representaciones de audio para los conjuntos de características perceptibles y de clasificación.
- **Spike 3**
 - Se incluye un resumen del inventario de datos privados del cliente, audios y características.
 - Se describe la calidad de los datos en términos de la consistencia de las características asignadas.
 - Se concluye si es viable construir y usar un *dataset* propio.
- **Desarrollo de algoritmos y modelos tradicionales para características básicas, perceptibles y clasificables por ánimo y género**
 - **HU 1, HU 5, HU 9 y HU 13**
 - Se presenta un análisis exploratorio de datos con estadísticas descriptivas.
 - Se identifican las distribuciones y correlaciones de los datos según gráficos y visualizaciones.
 - El informe muestra el tratamiento de valores nulos o atípicos.
 - Se describen las categorías del problema de clasificación: Aplica en HU 9, HU 13, y para *Key* y *Mode* en HU 1.
 - **HU 2, HU 6, HU 10 y HU 14**
 - Se realiza la normalización o estandarización de datos numéricos según aplique.
 - Se codifican variables categóricas con técnicas estándar (p. ej., *one-hot encoding*) según aplique.
 - Se documentan los pasos de limpieza de datos y el tratamiento de valores faltantes.
 - **HU 3, HU 7, HU 11 y HU 15**
 - En HU 3 se implementa al menos un modelo tradicional de aprendizaje supervisado, como máquinas de vectores de soporte (SVM), árboles aleatorios (RF), o k vecinos cercanos (kNN).
 - Se ejecutan algoritmos tradicionales (sin IA) para comparar los resultados con los resultados del modelo tradicional. Aplica en el caso de HU 3.
 - En HU 7, HU 11 y HU 15, se implementa al menos un modelo de red neuronal.
 - Se documenta el proceso de selección de hiperparámetros en un *Jupyter notebook*.
 - **HU 4, HU 8, HU 12 y HU 16**
 - Se aplican métricas de desempeño (p. ej., *accuracy*, *F1*, *RMSE* según el caso).
 - Se selecciona el mejor modelo o algoritmo por característica en función de estas métricas.
 - Se entrega reporte comparativo de modelos y algoritmos.
- **Implementación del *pipeline* de procesamiento**
 - **HU 17 - opcional**
 - La interfaz permite subir archivos de audio en formato estándar (.mp3 o .wav).
 - Cada archivo se agrega a la cola de procesamiento.
 - El sistema devuelve confirmación de recepción.

- **HU 18**

- El *pipeline* procesa automáticamente un archivo nuevo.
- Se ejecutan los modelos desarrollados.
- Los resultados se almacenan en la base de datos.
- El sistema almacena estado y detalle asociado al archivo.

- **HU 19 - opcional**

- El sistema está disponible en un entorno de producción del cliente.
- El *backend* permite ejecutar al menos 10 archivos consecutivos sin fallo.
- La documentación de despliegue está disponible para reproducir la instalación.

- **HU 20**

- El usuario accede a la base de datos en modo lectura.
- El usuario puede filtrar los resultados por canción.
- El usuario puede validar y reportar cualquier incidente al cliente y al desarrollador.

8. Fases de CRISP-DM

1. Comprensión del negocio

Objetivo: minimizar dependencias de APIs externas mediante un módulo de IA propio que genere características musicales automáticamente.

Impacto: reducción de costos, autonomía tecnológica y optimización del flujo de ingestión de canciones.

Métricas: cobertura de características musicales, reducción de tiempo de respuesta en el *pipeline* y satisfacción del cliente.

2. Comprensión de los datos

Tipos de datos: archivos de audio y características musicales tabulares.

Fuentes de datos: datos privados del cliente y *datasets* públicos (p. ej., *GTZAN*, *Million Song Dataset*).

Cantidad de datos: aproximadamente 65,000 canciones con características asignadas, más la información pública disponible.

Calidad de los datos: datos revisados por músicos, sujetos a EDA.

3. Preparación de los datos

Transformaciones necesarias: se transforman los audios en representaciones numéricas como *MFCC*. Normalización, segmentación de señales, y construcción de *datasets* balanceados.

Características clave: *chroma features* para características básicas, y espectrogramas para perceptibles y clasificables.

4. Modelado

Tipo de problema: clasificación y regresión para características básicas, regresión para características perceptibles y clasificación multi-clase para características clasificables por ánimo y género.

Arquitecturas posibles:

- Modelos tradicionales para características básicas (SVM, RF, kNN).
- Algoritmos de procesamiento de señales (DSP) para características básicas.
- Modelos de aprendizaje profundo como redes neuronales convolucionales (CNN), redes neuronales recurrentes (RNN), *Transformers* y modelos preentrenados, para las características perceptibles, ánimo y género.

Se ejecuta un ciclo iterativo de experimentación con validación cruzada, comparación de arquitecturas y ajuste de hiperparámetros.

5. Evaluación del modelo

Métricas de evaluación: *F1-score*, *AUC-ROC* para evaluar la capacidad del modelo de clasificación de características. *RMSE* y *MAE* para regresión.

6. Despliegue del modelo (opcional)

No se contempla como parte del alcance.

9. Desglose del trabajo en tareas

HU	Tarea técnica	Estimación	Prioridad
HU1	Generar visualizaciones de las distribuciones para características básicas	5 h	Alta
HU1	Identificar patrones, anomalías y faltantes para las características básicas	6 h	Alta
HU2	Implementar normalización y estandarización de variables numéricas	6 h	Alta
HU2	Codificar variables categóricas (ej., <i>one-hot encoding</i>)	5 h	Media
HU3	Implementar modelos SVM, RF, para predicción de características básicas	7 h	Alta
HU3	Implementar algoritmos DSP para generar características básicas	6 h	Media
HU4	Calcular métricas de desempeño (<i>accuracy</i> , <i>F1</i> , <i>RMSE</i>)	5 h	Alta
HU4	Elaborar reporte comparativo de modelos y algoritmos	4 h	Media

HU	Tarea técnica	Estimación	Prioridad
HU5	Aplicar técnicas de visualización e interpretar los resultados obtenidos para características perceptibles	5 h	Alta
HU5	Identificar patrones generales, anomalías, faltantes y correlaciones para las características perceptibles	6 h	Alta
HU6	Generar espectrogramas y normalizar audios	7 h	Alta
HU6	Implementar <i>data augmentation</i> para ampliar <i>dataset</i>	6 h	Media
HU7	Definir arquitectura de red neuronal multi-salida	8 h	Alta
HU7	Entrenar modelo inicial y guardar pesos	7 h	Alta
HU8	Evaluar desempeño del modelo con métricas apropiadas	5 h	Alta
HU8	Documentar resultados de evaluación y ajustes realizados	4 h	Media
HU9	EDA de distribución de <i>moods</i> (categorías principales/secundarias)	5 h	Alta
HU9	Generar gráficos de dispersión y matrices de correlación	4 h	Media
HU10	Preparar <i>dataset</i> de entrenamiento para clasificación de <i>moods</i>	6 h	Alta
HU10	Codificar etiquetas multi-clase y balancear <i>dataset</i>	5 h	Alta
HU11	Diseñar y entrenar red neuronal para predicción de <i>moods</i>	8 h	Alta
HU11	Ajustar hiperparámetros iniciales (tasa de aprendizaje, épocas)	6 h	Media
HU12	Evaluar el modelo con métricas multi-clase ($F1$)	5 h	Alta
HU12	Elaborar informe comparativo de experimentos realizados	4 h	Media
HU13	EDA de distribución de géneros y subgéneros musicales	5 h	Alta
HU13	Generar reportes visuales de categorías de género	4 h	Media
HU14	Preparar <i>dataset</i> para clasificación de género y subgénero	6 h	Alta
HU14	Aplicar técnicas de balanceo para clases poco representadas	5 h	Alta
HU15	Diseñar y entrenar red neuronal multi-salida para géneros	8 h	Alta
HU15	Comparar desempeño de diferentes arquitecturas (CNN, RNN)	7 h	Media
HU16	Evaluar métricas de desempeño en clasificación multi-clase	5 h	Alta
HU16	Documentar resultados y selección del mejor modelo	4 h	Media
HU18	Diseñar e implementar <i>pipeline</i> que integre modelos desarrollados	8 h	Alta
HU18	Configurar base de datos para almacenamiento automático de resultados	6 h	Alta
HU20	Crear consultas de solo lectura para validación de resultados	5 h	Alta
HU20	Implementar filtros de búsqueda por canción en la base de datos	6 h	Media

10. Planificación de Sprints

Sprint	HU o fase	Tarea	Horas / SP	Responsable	% Avance
Sprint 0	Planificación	Definir alcance, cronograma y <i>backlog</i> inicial	35 h	Alumno	95 %
Sprint 0	Planificación	Reuniones iniciales con tutor/cliente	10 h	Alumno	100 %
Sprint 1	Spike 1 – 3	Selección de <i>datasets</i> y técnicas (EDA preliminar)	45 h	Alumno	66 %
Sprint 2	HU1 – HU2	EDA + preparación de datos (características básicas)	50 h	Alumno	0 %
Sprint 3	HU3 – HU4	Entrenamiento y evaluación de modelos tradicionales	50 h	Alumno	0 %
Sprint 4	HU5 – HU6	EDA y preparación de datos (perceptibles)	50 h	Alumno	0 %
Sprint 5	HU7 – HU8	Modelos profundos y métricas (perceptibles)	50 h	Alumno	0 %
Sprint 6	HU9 – HU10	EDA + preparación de datos (ánimo)	40 h	Alumno	0 %
Sprint 7	HU11 – HU12	Modelos profundos y métricas (ánimo)	40 h	Alumno	0 %
Sprint 8	HU13 – HU14	EDA + preparación de datos (género)	40 h	Alumno	0 %
Sprint 9	HU15 – HU16	Modelos profundos y métricas (género)	40 h	Alumno	0 %
Sprint 10	HU17 – HU20	<i>Pipeline</i> , BD y pruebas de integración	50 h	Alumno	0 %
Sprint 11	Escritura	Redacción memoria y documentación técnica	50 h	Alumno	0 %
Sprint 12	Defensa	Preparación de exposición y ajustes finales	50 h	Alumno	0 %

11. Diagrama de Gantt (sprints)

A continuación se puede observar el diagrama de Gantt. Este muestra gráficamente cómo se distribuirá el desarrollo de las historias de usuario a lo largo de los sprints. Las columnas representan las semanas del proyecto, y las filas, los sprints. Del sprint 1 al 9, los sprints tienen una duración de dos semanas, los demás son un poco más extensos (plan, *pipeline*, escritura y defensa).

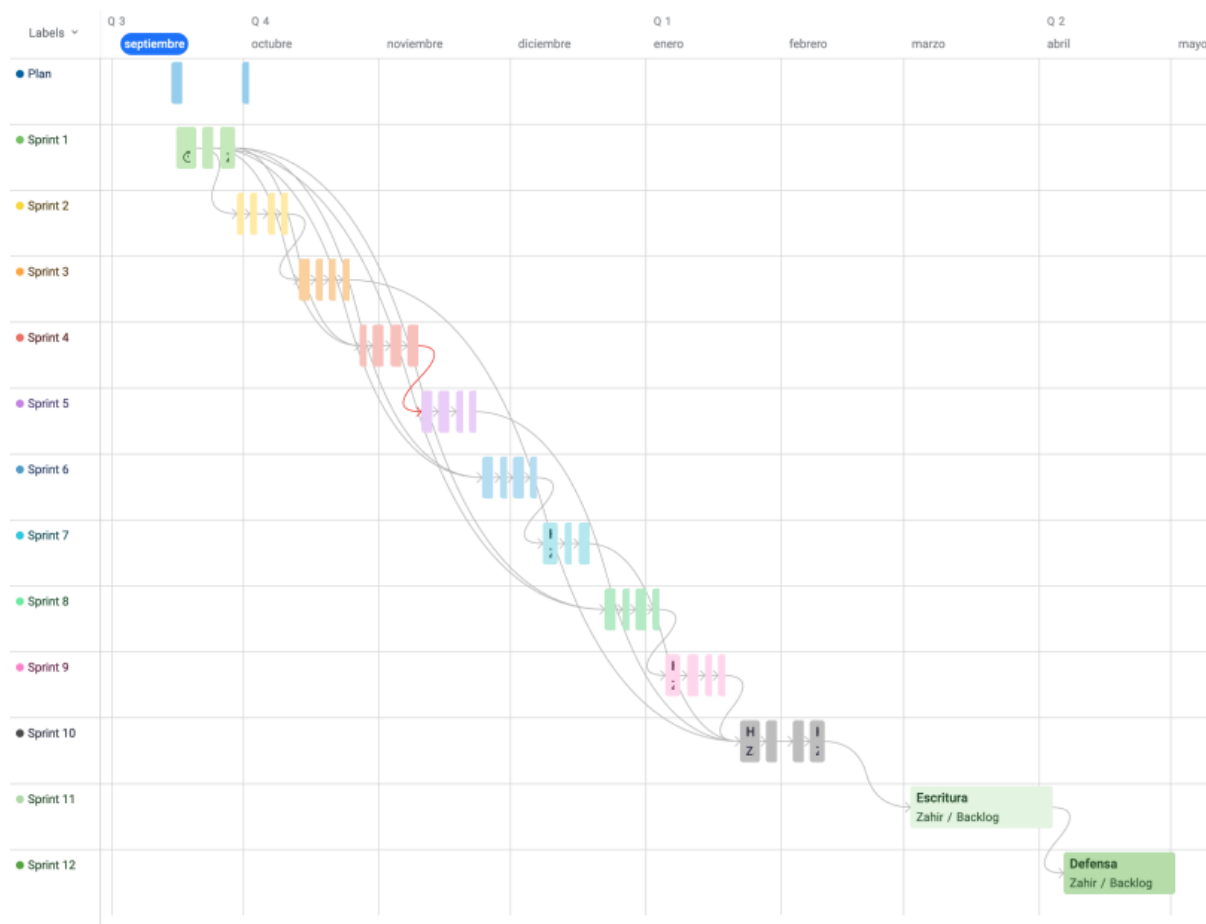


Figura 3. Diagrama de Gantt.

12. Normativa y cumplimiento de datos (gobernanza)

En este proyecto no se analizan ni se almacenan datos personales de individuos, por lo que no resulta aplicable la normativa de protección de datos personales (como el Reglamento General de Protección de Datos –GDPR– en la Unión Europea o la Ley 25.326 en Argentina). En consecuencia, no es necesario recabar consentimiento explícito de usuarios finales, dado que no se involucra información sensible ni identificable.

Los conjuntos de datos empleados provienen de dos tipos de fuentes:

- *Datasets* públicos: disponibles abiertamente para fines de investigación y desarrollo, sin restricciones contractuales adicionales.
- *Dataset* privado del cliente: de propiedad exclusiva del cliente, autorizado para su uso en este proyecto académico. Para efectos de la evaluación del trabajo, se va a disponer de subconjuntos de datos o material de referencia compatible con el director y los jurados, preservando siempre los derechos de propiedad del cliente.

Es importante aclarar que los archivos de audio contenidos en los *datasets*, tanto públicos como privados, son propiedad intelectual de sus respectivos compositores e intérpretes. El proyecto no transfiere ni modifica esos derechos, limitándose al uso autorizado de copias digitales para fines de análisis técnico y académico.

En términos de propiedad y derechos sobre los resultados del proyecto, estos corresponden de forma conjunta al cliente y al alumno, de acuerdo con los acuerdos establecidos.

En conclusión, el proyecto cumple con la normativa vigente en materia de gobernanza de datos, respeta los derechos de propiedad intelectual de terceros y establece claramente las condiciones de uso de los *datasets* empleados, garantizando la viabilidad legal y ética de la investigación.

13. Gestión de riesgos

a) Identificación de los riesgos y estimación de sus consecuencias:

Riesgo 1: datos insuficientes para el entrenamiento de alguno de los modelos.

- Severidad (7): la falta de datos variados impactaría directamente en la capacidad de los modelos para generalizar. Se requiere diversidad de géneros y estados de ánimo.
- Probabilidad de ocurrencia (4): aunque se espera contar con datos suficientes, podría faltar variedad, lo cual afectaría el desempeño del modelo.

Riesgo 2: falta de disponibilidad de recursos de procesamiento.

- Severidad (8): la falta de acceso a *hardware* en la nube podría ser un obstáculo para el proyecto.
- Ocurrencia (2): los proveedores en la nube son confiables; el mayor riesgo es la falta de pago, con probabilidad baja.

Riesgo 3: retrasos en la integración de los diferentes módulos del sistema.

- Severidad (6): la integración tardía puede generar incompatibilidades y aumentar la carga de trabajo en fases finales.
- Ocurrencia (5): la coordinación entre módulos depende de avances paralelos; existe riesgo medio de retrasos.

Riesgo 4: cambios en los requisitos del cliente.

- Severidad (8): cambios significativos en requisitos pueden obligar a rehacer trabajo, afectando tiempo y recursos.
- Ocurrencia (3): es poco común, pero posible, que el cliente ajuste expectativas en el transcurso del desarrollo.

Riesgo 5: problemas de calidad en los datos (ruido, etiquetas erróneas).

- Severidad (7): la baja calidad de los datos de entrenamiento compromete la precisión y confiabilidad del modelo.
- Ocurrencia (6): es probable que los datos recolectados contengan ruido o inconsistencias.

Riesgo 6: costos de procesamiento mayores que los costos de consumo de APIs de terceros.

- Severidad (8): un gasto excesivo en procesamiento propio puede hacer que el proyecto no sea sostenible económicamente.
- Probabilidad de ocurrencia (4): es de probabilidad menor, pero en algunos casos los modelos de aprendizaje profundo pueden requerir recursos intensivos.

b) Tabla de gestión de riesgos: (El RPN se calcula como $RPN=S \times O$)

Riesgo	S	O	RPN	S*	O*	RPN*
Datos insuficientes para entrenamiento	7	4	28	5	3	15
Falta de recursos de procesamiento	8	2	16	-	-	-
Retrasos en integración de módulos	6	5	30	4	4	16
Cambios en requisitos del cliente	8	3	24	-	-	-
Problemas de calidad en los datos	7	6	42	5	4	20
Altos costos de procesamiento	8	4	32	7	3	21

Criterio adoptado:

Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a 25.

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

c) Plan de mitigación de los riesgos que originalmente excedían el RPN máximo establecido:

Riesgo 1: datos insuficientes para el entrenamiento. Mitigación: combinar fuentes de datos públicas y privadas, aplicar técnicas de aumento de datos musicales (*data augmentation*).

- Severidad (5): con datos adicionales y estrategias de aumento, el impacto se reduce.
- Probabilidad de ocurrencia (3): la recolección activa de datos baja la probabilidad.

Riesgo 3: retrasos en la integración de módulos. Mitigación: aplicar integración continua y pruebas unitarias desde etapas tempranas.

- Severidad (4): la detección temprana de fallos reduce el impacto.
- Probabilidad de ocurrencia (4): con reuniones de seguimiento y *continuous integration* / *continuous deployment* (CI/CD) la probabilidad baja.

Riesgo 5: problemas de calidad en los datos. Mitigación: establecer procesos de limpieza y validación de datos antes del entrenamiento.

- Severidad (5): la limpieza reduce el efecto de datos defectuosos en el modelo.
- Probabilidad de ocurrencia (4): el preprocesamiento y validación disminuyen la probabilidad de error.

Riesgo 6: costos de procesamiento mayores que consumo de APIs de terceros. Mitigación: realizar un análisis comparativo de costos antes de comprometerse con infraestructura propia. Establecer alertas de consumo en la nube.

- Severidad (6): usar los costos de procesamiento en entrenamiento y test como criterios para descartar modelos potencialmente costosos.
- Probabilidad de ocurrencia (3): con monitoreo de costos disminuye la probabilidad de sobre costos.

14. Sprint Review

He elegido HU5, HU6, HU7 y HU8, con sus tareas y entregables.

HU seleccionada	Tareas asociadas	Entregable esperado	¿Cómo sabrás que está cumplida?	Observaciones o riesgos
HU5	<ul style="list-style-type: none"> - Aplicar técnicas de visualización - Identificar patrones y correlaciones 	<i>Jupyter notebook</i>	<ul style="list-style-type: none"> - Contiene EDA - Visualización de distribuciones y correlaciones - Identificación de nulos, <i>outliers</i> y anomalías 	Riesgo: los datos usados no reflejen variabilidad real
HU6	<ul style="list-style-type: none"> - Realzar preprocesamiento de datos - Generar espectrogramas 	Datos preprocesados	<ul style="list-style-type: none"> - Suficientes datos normalizados / estandarizados y codificados en espectrogramas 	Riesgo: aumento de tiempos de cómputo y almacenamiento
HU7	<ul style="list-style-type: none"> - Definir arquitectura de red neuronal - Entrenar modelo inicial 	Modelo entrenado con pesos iniciales guardados	El modelo compila, entrena y genera métricas básicas	Riesgo: posible sobreajuste en los primeros entrenamientos
HU8	<ul style="list-style-type: none"> - Evaluar desempeño del modelo - Iterar, documentar resultados y ajustes 	Reporte con métricas y documentación de ajustes	Las métricas alcanzadas son comparables con umbrales definidos	Riesgo: falta de tiempo para realizar iteraciones

15. Sprint Retrospective

Aquí incluyo 3 sprints técnicos (2, 3 y 8) y 1 no técnico (12: Defensa), siguiendo la plantilla de la estrella de mar. **Formato sugerido:**

Sprint tipo y N°	¿Qué hacer más?	¿Qué hacer menos?	¿Qué mantener?	¿Qué empezar a hacer?	¿Qué dejar de hacer?
Sprint tipo y N°	¿Qué hacer más?	¿Qué hacer menos?	¿Qué mantener?	¿Qué empezar a hacer?	¿Qué dejar de hacer?
Sprint técnico - 2	Validaciones continuas con el tutor	Cambios sin registro en el backlog	Costumbre de pruebas rápidas	Documentar cada cambio propuesto	Repetir pruebas manuales innecesarias
Sprint técnico - 3	Revisión del estado del arte de problemas similares	Modificar parámetros sin guardar historial	Uso de componentes reutilizables	Manejo de <i>logs</i>	Ajustes improvisados sin análisis previo
Sprint técnico - 10	Preparar configuraciones en múltiples entornos	Mantenimiento de componentes de sprint previos	Reducción de la deuda técnica	CI/CD	Despliegues manuales de componentes y datos
Sprint no técnico - 12 (Defensa)	Ensayos orales con <i>feedback</i> de terceros	Cambiar diapositivas en último momento	Material visual claro y estructurado	Dividir la presentación en bloques con tiempos definidos	Agregar gráficos muy complejos de explicar