

# Guia de Repositorio

## Versionado y ramificación

Para el desarrollo de software necesario en el trabajo obligatorio, utilizaremos la política de Trunk Based Development (TBD).

TBD es un modelo de desarrollo de software en el que los desarrolladores realizan cambios directamente en la rama principal del repositorio, llamada *trunk*. En lugar de trabajar en ramas separadas durante largos períodos de tiempo, los cambios se integran rápidamente en el *trunk* para que todo el equipo tenga acceso a los cambios y puedan detectar cualquier problema a tiempo.

A modo general dispondremos de 2 ramas activas: (a) *Main* y (b) *Development*. En este caso, se propone utilizar la rama *Develop* como la rama de trabajo principal, en la cual se realizan los cambios y se integran rápidamente en lugar de trabajar en ramas separadas. Cuando el código esté lo suficientemente estable, se fusionará con la rama *Main* para ser desplegado en el entorno de producción.

Este modelo de desarrollo tiene varias ventajas, como la reducción del tiempo de entrega, la minimización de conflictos en la integración de código y la promoción de la colaboración y el trabajo en equipo. Además, el monitoreo constante de la estabilidad del software permite la detección temprana de errores y su resolución antes de que se conviertan en problemas mayores en el entorno de producción.

## Estructura del repositorio

La estructura del repositorio consta de las siguientes carpetas y subcarpetas:

- a. Recursos: es una carpeta que contiene las siguientes subcarpetas:
  - Base de Datos: posee los archivos de base de datos recibidas inicialmente y en ella se guardará cualquier modificación que sea necesaria realizar;
  - Consigna: contiene la letra del obligatorio base del trabajo recibido, consideramos necesaria;
  - Documentación: contiene la documentación recibida como base del trabajo obligatorio de partida;
- b. Código: contiene todo el código fuente de la aplicación, tanto del *backend* y *frontend* así como las librerías necesarias para la importación y exportación de conciertos;
- c. Entregas: carpeta sobre la cual se generarán las distintas subcarpetas asociadas a las diferentes entregas solicitadas; y,
- d. Proyecto ISA2-Gestion, ver *Tableros*

## Tableros

En lo que respecta al proyecto ISA2-Gestion, inicialmente contiene un tablero kanban que es utilizado para la gestión de todas las tareas identificadas con el comienzo del trabajo propuesto. La intención de utilizar un tablero diferenciado para estas tareas es la de obtener una visión más limpia de las actividades para conseguir que los productos finales estén alineados con lo solicitado en esta primera entrega, es decir, utilizar un tablero específico para la generación y gestión de la documentación solicitada en las entregas.

Posteriormente, utilizaremos otro tablero –o evolucionaremos el actual– para representar las actividades asociadas a la segunda entrega. Por este motivo, esta decisión será parte de la planificación del trabajo correspondiente a la segunda entrega y se considera fuera del alcance actual.

El tablero Kanban utilizado para gestionar la documentación necesaria para las entregas comenzó siendo un tablero Kanban ágil con las columnas de *To Do*, *In Progress* y *Done*. Posteriormente, se agregaron las columnas de *Revisión* antes de la columna *Done* (esto permite que otra persona revise la documentación antes de que se marque como completada), *Análisis de Código* y *Testing* (en ese orden y ambas antes de *Review*).