Project planification

Project

Kodi is a free and open source media center application. It allows users to play and view most videos, music, podcasts, and other digital media files from local and network storage media and the internet. This document specify the project planification.

Deadline

Project submission date	31 janv. 2026

Team

Name	Contact
Rosa Pier-Alexandre	pier-alexandre.rosa@epitech.eu
L'huillier Léo	leo.l-huillier@epitech.eu

Key resources and documents

- Github repository
- Github project

Planning

	Octobre	Novembre	Decembre	Janvier
Preparation	🟥			
Project Setup & Foundations	🟥	🟪		
Core Media Engine		🟪		
User Interface		🟪	🟩	
Database & Library Management			🟩	
Advanced Features			🟩	
Plugin & Extension System			🟩	🟨

Detailed Planning

Team Organization and Time Management

As a two-person team, each member will be involved in all aspects of the project. However, for certain tasks, it is expected that one of the two members will take a leading role depending on their individual strengths or interests.

The estimated time for each task includes the development of tests and documentation, as well as the resolution of potential bugs. We also take into account the fact that we have chosen to implement the project in Rust, a language we are currently learning. This naturally introduces an initial adaptation and learning period that may affect our productivity at the beginning of the project.

Our limited experience with Rust also makes it challenging to accurately estimate the workload for certain features, particularly regarding plugin management, extension systems, and external API integration. We plan to reassess and refine these estimations once we have gained more familiarity with the language and obtained concrete progress data from the early stages of development

If the development progresses faster than anticipated, the document will be updated at the end of each project phase to reflect the new planning and progress.

The estimated time does not include discussion or reflection periods related to design decisions, brainstorming sessions, or general project coordination.

Tasks	Required time (h)	Team members
Preparation 19/10 – 25/10		
Benchmark	~30	PAR LL
Setup Github repository	N/A	LL
Setup Github Project	~3h	LL
Write the project specifications document	~3h	LL
Project Setup & Foundations 26/10 – 07/11		
Write benchmark document	~4	PAR
Properly initialize the project	~3	LL
Add linting and formatting tools	~3	LL
Setup CI/CD	~14	LL
Design modular structure and architecture	~60	PAR LL
Core Media Engine 08/11 – 21/11		
Add Basic media scanning	~10	PAR
Media playback	~10	LL
Extract metadata	~10	PAR
Media management	~10	LL
Keep media entries into a small database	~10	PAR
Get Rust audio/video libraries	~10	LL
Add controls management	~10	PAR

Add media state management	~10	LL
User Interface 22/11 – 05/12		
Build minimal window with media library list	~16	PAR
Add media controls and progress bar	~16	LL
Display metadata and player state	~16	PAR
Manage multiple media state	~16	LL
Connect GUI to media core backend	~16	PAR LL
Database & Library Management 06/12 – 19/12		
Implement local library storage	~20	PAR LL
Add indexing & search system	~20	LL
Allow tagging and categorization	~20	LL
Manage media data	~20	PAR LL
Advanced Features 20/12 – 02/01		
Add queue system	~2	PAR
Add repeat/shuffle	~2	LL
Manage playlists	~2	PAR
Implement user configuration	~10	LL LL
Extensions et plugins 20/12 – 31/01		

Manage plugin, extension and online media API	~230	PAR LL

Last update 26 oct. 2025