PROJECT PLAN FOR TUNE TECH MUSIC PLAYER

Submitted by

Afzal Mukhtar
SRN- PES2201800675
Section A
Worked on part 2,3

Hritika Rahul Mehta
SRN-PES2201800024
Section A
Worked on part 1,4

Meghana I.
SRN- PES2201800028
Section A
Worked on part 5,6

1. Lifecycle to be followed

Agile software methodology will be followed in our project. Agile methodology is based on collaborative decision making between requirements and solutions teams, and a cyclical, iterative progression of producing working software. It is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Every iteration involves cross functional teams working simultaneously on various areas like −

- Planning
- Requirements Analysis
- Design
- Coding
- Unit Testing and
- Acceptance Testing.

In our project we focus on developing a music player and with the agile method we can develop a plan specific to our project. It also helps to have frequent customer interaction to design a software to satisfy all their needs. Given the small time frame for implementation we can focus on developing a working software using an agile model.

2. Tools to be used throughout the lifecycle

Planning Tool

- Spreadsheet – Excel and Google Sheet
- Notes App
- Google Calendar

Design Tool

- PowerPoint
- CorelDraw
- Canva

Version Control

- GitHub
- Google Drive

Development Tool

- XCode
- VS Code
- Sublime Text
- CodePen
- BBEdit

Bug Tracking

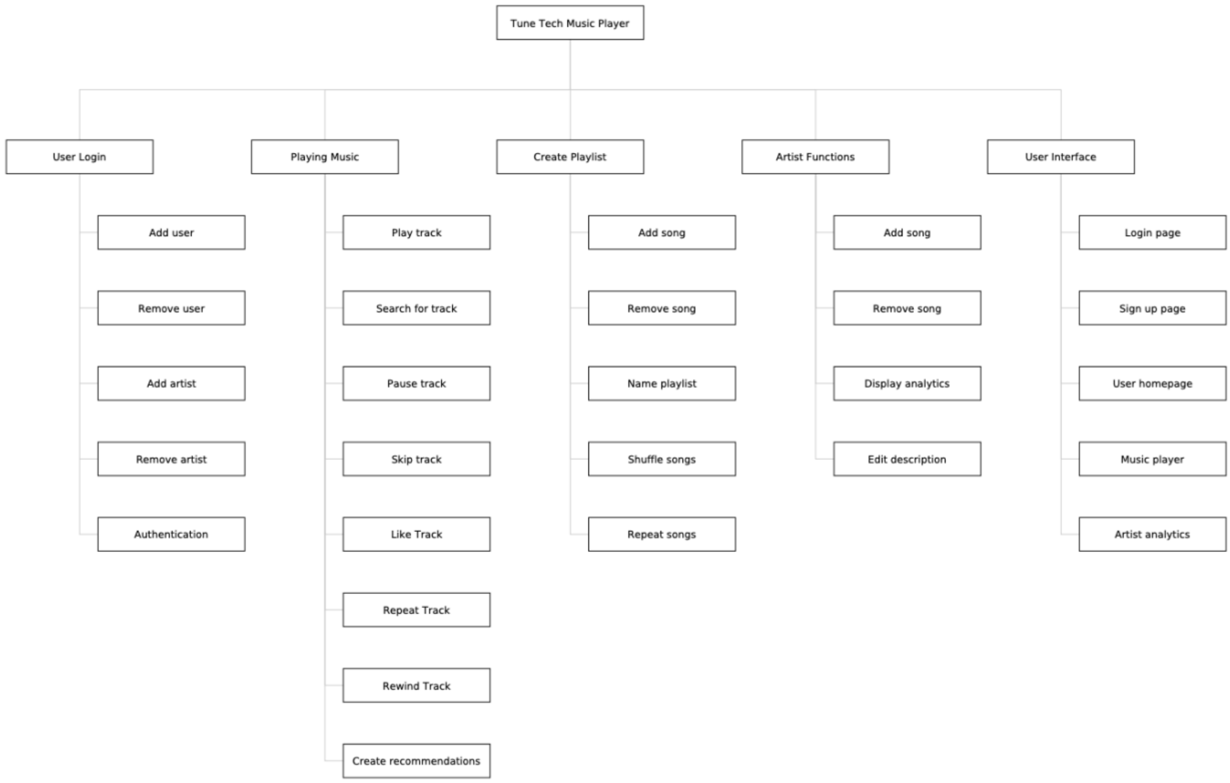
- VSCode (in-built)
- XCode (in-built)
- PythonTutor

3. Deliverables

Item	Reuse / Build / Document	Justification
Final Documentation	Document	Final documentation provided after the end of project
Use Cases	Document	Information on what the project's intended use is, so as to prevent any misuse
Directions for Setup	Document	Setup directions will be provided so as to ensure a bug free usage of the application
Privacy Requirement	Document	Information on the data being collected to ensure user's privacy
Function Details	Document	A detailed information on the functional aspects of the project to help other developers understand the code and debug any issues on hand-over
General Bug Fix	Document	Fix for generally occurring bugs will be provided, so that users and other developers can overcome the most commonly occurring bug easily.
Code Structure	Build	The project is built from scratch for focusing on the specific goals.

Front-end	Build	A front-end design will be built from scratch using some code templates for references for a good User Interface
Front-end UI Design Template	Reuse	The front-end UI design template will be reused from our previous project (Web Technology - 1), so as to provide a template for faster completion of the front-end design.

4. WBS for project functionalities



5. Cost Estimation

What is a man-month?

In Software Engineering, a man-month or a person-month is the effort put in by one person in one month.

Model Used: Basic COCOMO

COnstructive COst MOdel (COCOMO) is an algorithmic cost estimation model.

Team Type: Organic
A relatively small team develops a software in a known environment. The people involved generally have a lot of experience with similar projects in their organisation. Projects of this type will seldom be very large projects.

Effort in man-months:
$E = b * (KLOC)^{c}$, where

- E —>Effort required in man-months.
- b —>2.4
- c —>1.05

Lines of Code for Each Component of WBS (Work Breakdown Structure)
User Log In: 500 lines of code
This is the smallest component.

- Add User: 100 lines of code
- Remove User: 100 lines of code
- Add Artist: 100 lines of code
- Remove Artists: 100 lines of code
- Authenticate: 100 lines of code

Play Music: 1,000 lines of code
This is the largest component.

- Play Track: 100 lines of code
- Search for Track: 100 lines of code
- Skip Track: 100 lines of code
- Like Track: 100 lines of code
- Create Recommendation: 400 lines of code
- Pause Track: 100 lines of code
- Repeat Track: 100 lines of code

Create Playlist: 575 lines of code

- Add Song: 100 lines of code
- Remove Song: 100 lines of code
- Name Playlist: 100 lines of code
- Shuffle Songs: 150 lines of code
- Repeat Songs: 125 lines of code

Artists: 600 lines of code

- Add Song: 50 lines of code
- Remove Song: 50 lines of code
- Display Analytics: 400 lines of code
- Edit Description: 100 lines of code

User Interface: 550 lines of code

- Login Page: 100 lines of code
- Sign Up: 100 lines of code
- User Homepage: 100 lines of code
- Music Player: 100 lines of code
- Artists Analytics: 150 lines of code

Computing KLOC and Effort Required for Each Component of WBS

User Log In

- Lines of code = 500
- KLOC = 0.500
- E = 1.159 PM

Play Music

- Lines of code = 1,000
- KLOC = 1.000
- E = 2.400 PM

Create Playlist

- Lines of code = 575
- KLOC = 0.575
- E = 1.404 PM

Artists

- Lines of code = 600
- KLOC = 0.600
- E = 1.404 PM

User Interface

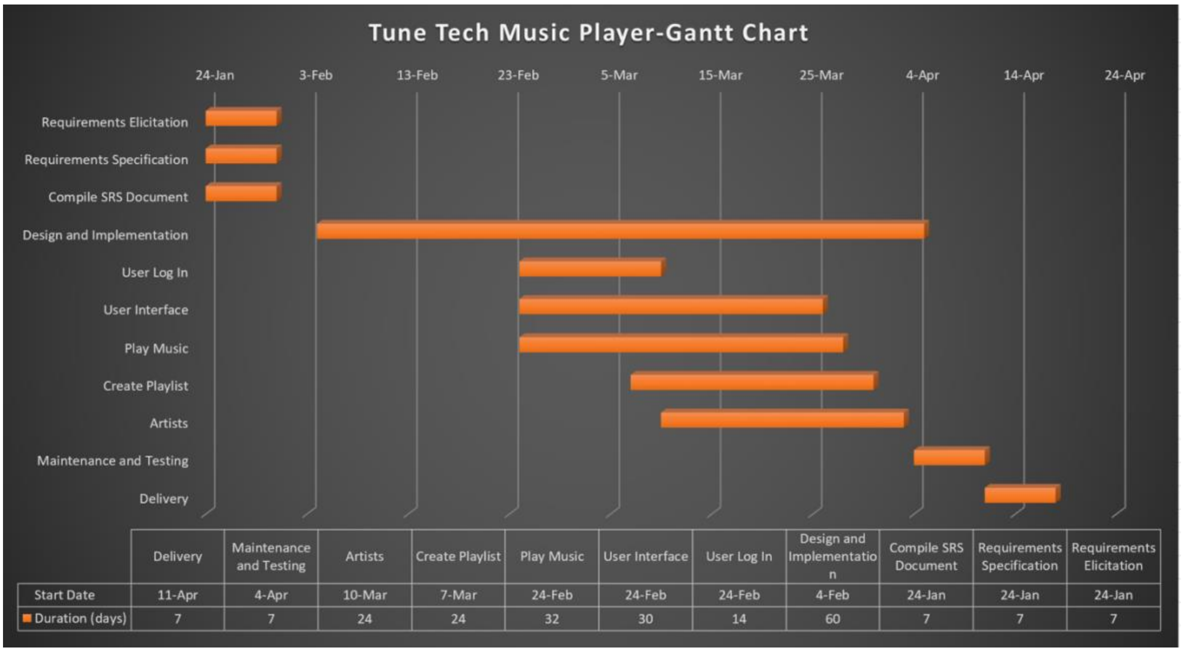
- Lines of code = 550
- KLOC = 0.550
- E = 1.281 PM

Total Effort Required

E = 7.648 PM
Hence, the three members of our team should take = 7.648/3 = **around 2.5 months.**

6. Gantt chart for scheduling

The Gantt chart below is a visual representation of our project schedules and timeline. Project activities and their estimated duration can clearly be inferred from this Gantt chart.



	Delivery	Maintenance and Testing	Artists	Create Playlist	Play Music	User Interface	User Log In	Design and Implementatio n	Compile SRS Document	Requirements Specification	Requirements Elicitation
Start Date	11-Apr	4-Apr	10-Mar	7-Mar	24-Feb	24-Feb	24-Feb	4-Feb	24-Jan	24-Jan	24-Jan
■ Duration (days)	7	7	24	24	32	30	14	60	7	7	7