

Technical documentation

Create a work breakdown structure of provided functionalities. Explain how it will work, what code or software has been used, images and snapshots can work as well. More details can be found in the example below:

Example:

1. Project Structure
 - Describe the high-level structure of the project, including the main components and their relationships.
 - Provide a visual representation of the project's architecture, such as a block diagram or flowchart.
2. Technologies Used
 - List and briefly explain the technologies and frameworks employed in the website project (e.g., programming languages, libraries, databases, etc.).
 - Include version numbers and any specific configurations or dependencies required.
3. Installation and Setup
 - Provide step-by-step instructions that were used to set up the development environment.
 - Include any prerequisites, dependencies, or configurations that were required for successful installation.
 - Specify any environment variables or settings that were configured before or during the development.
4. Development Guidelines
 - Define coding standards and guidelines that were followed by the development team.
 - Specify the file structure and naming conventions for the project.
 - Describe the version control system and branching strategy used.
5. Architecture and Design
 - Discuss the overall architecture of the website, including the frontend, backend, and any additional layers or components.
 - Provide detailed diagrams, such as sequence diagrams or class diagrams, to illustrate the system's design.
 - Explain the data flow and interaction between different modules or components.
6. Features and Functionality

This is the main component of the technical document. It should include a sub-deliverables and activities that were accomplished. Explain each functionalities codes.

- Document each major feature of the website and its functionality.
- Document codes that were used and explain why they were used.
- Include detailed descriptions, input/output examples, and any relevant screenshots or mockups.

7. APIs and Integrations

- If the website interacts with external systems or APIs, describe the integration points.
- Document the API endpoints, request/response formats, authentication mechanisms, and any required API keys or tokens.

8. Deployment and Hosting

- Explain the deployment process and any specific tools or platforms used.
- Provide instructions for deploying the website to different environments (development, staging, production).
- Document any necessary configurations, environment variables, or server requirements.

9. Testing and Quality Assurance

- Describe the testing approach, including unit tests, integration tests, and end-to-end tests.
- Document the testing frameworks and tools used.