Explain here in one or two sentences what is the goal of your product/assistant.

More details about the product should be found here below, including the product vision, market research and domain analysis.

Vision

For product teams who struggle with unclear or incomplete requirements, OptiReq is an AI-driven requirements analysis tool that verifies and improves requirement quality. Unlike manual reviews, our product provides instant, actionable feedback to ensure requirements are clear, complete, and feasible.

Research

Survey of similar projects and analysis of their pros and cons when compared to the product to be developed.

Domain Analysis

Include high-level class diagram with key domain concepts. Complement this diagram with other high-level diagrams has appropriate (activity, sequence, etc.).

Architecture and design

Describe the architecture and design of the tool. Use component/deployment diagrams. If needed, resort to package diagrams to organize them into more manageable parts.

Be clear about what is the current architecture/design and what is the one you envision in the future, in case they are different. Identify main risks and justify the most important choices to show the soundness of the architecture and design that you have implemented or plan to implement.

Technologies

Identify the main technologies, languages and frameworks used.

Clearly identify which ones were restrictions imposed by the client and which were your own choices. Justify your choices and explain in your own words the motivation for the restrictions of your client.

Explain the prototype or base implementation that you have implemented in Sprint 0, and how that has informed the rest of the development.

Development guide
Explain what a new developer to the project should know in order to develop the system, including who to build, run and test it in a development environment.

Document any APIs, formats and protocols needed for development (but don't forget that public APIs should also be accessible from the "How to use" above).

Describe coding conventions and other guidelines adopted by the team(s).

Security concerns
Identify potential security vulnerabilities classes and explain what the team has done to mitigate them.

Quality assurance
Describe which tools are used for quality assurance and link to relevant resources. Namely, provide access to reports for coverage and mutation analysis, static analysis, and other tools that may be used for QA.

How to use

Explain how to use your tool from an user standpoint. This can include short videos, screenshots, or API documentation, depending on what makes sense for your particular software and target users. If needed, link to external resources or additional markdown files with further details (please add them to this wiki).

How to contribute

Explain what a new developer should know in order to develop the tool, including how to build, run and test it in a development

environment.

Defer technical details to the technical documentation below, which should include information and decisions on architectural, design and technical aspects of the tool.

Contributions

Link to the factsheets of each team and of each team-member. For example:

 * [Team 1](factsheets/team1.md)
   * [John Doe](factsheets/john_doe.md) (PO)
   * [Jane Doe](factsheets/jane_doe.md)
   * [Richard Doe](factsheets/richard_doe.md) (SM)
 * [Team 2](factsheets/team2.md)
   * ...