1. Requirements Analysis and Enhancement

What are OptiReq's main objectives in the AI4SD project?

Which kind of requirements - functional, non-functional, user stories - will OptiReq primarily focus on?

Should the AI only make recommendations for enhancements, or should it also be capable of rewriting requirements for better clarity?

How should OptiReq determine whether the requirements are ambiguous? What patterns or language features should it pay attention to?

Should OptiReq be able to logically reorder requirements, and if so, what guidelines should it follow for this reorganization?

What kinds of mission requirements should OptiReq automatically add, and should these be tailored to the type of project?

How should OptiReq handle incomplete requirements—should it prompt the user for more information or try to fill in the gaps automatically?

How detailed should the suggested improvements be? Should they just highlight issues, or should they provide detailed explanations?

What do we consider a "good" requirement in this context? Are there specific standards or criteria that we need to follow?

2. Translation and Multilingual Capabilities

Should OptiReq support requirements written in multiple languages, or focus on just one?

Which languages should OptiReq be able to translate requirements into?

How should the translation feature work—should it be real-time, or handled separately?

When translating, should the requirements maintain the original context and tone, or prioritize technical accuracy?

3. User Interaction and Experience

How will users interact with OptiReq? Through text input, voice commands, or integration within existing development tools?

Should the interface guide users through the analysis step-by-step, or just answer their specific queries?

Will OptiReq support multi-turn conversations, allowing users to dive deeper into requirements and suggestions over time?

Should users be able to give feedback on the AI's suggestions, and if so, how will that feedback help improve OptiReq's performance?

How can we guarantee that OptiReq adjusts its tone and language to fit different user profiles and project contexts?

Should OptiReq adapt its approach based on the user’s expertise, whether they're a seasoned analyst or a beginner?

4. Integration and System Architecture

How should OptiReq integrate with other AI tools within the AI4SD system?

What APIs or communication protocols should be used to ensure smooth interaction between OptiReq and other AI assistants?

What types of data (user inputs, analysis results, etc.) will OptiReq need to share with other tools?

How can we make sure that OptiReq's features complement, rather than duplicate, the functionality of other AI assistants?

5. Security and Privacy

What steps need to be taken to ensure the confidentiality of the requirement data analyzed by OptiReq?

Are there specific privacy or security regulations, like GDPR, that OptiReq must comply with?

What measures will be put in place to ensure high reliability and keep error rates as low as possible for OptiReq?

6. Customization and Flexibility

How customizable should OptiReq be for different projects and teams? Should users have the ability to tweak its analysis parameters or standards?

What kind of knowledge base should OptiReq use to enhance its suggestions? Should it include best practices, domain-specific guidelines, or both?

Should OptiReq perform real-time searches for additional context, or rely on a pre-indexed dataset?

How can we effectively combine information retrieved from a RAG (retrieval-augmented generation) system with the AI's suggestions to ensure both accuracy and relevance?

7. Evaluation and Success Metrics

How will we measure the success of OptiReq? Should we focus on reducing ambiguities, improving requirement quality, or measuring user satisfaction?

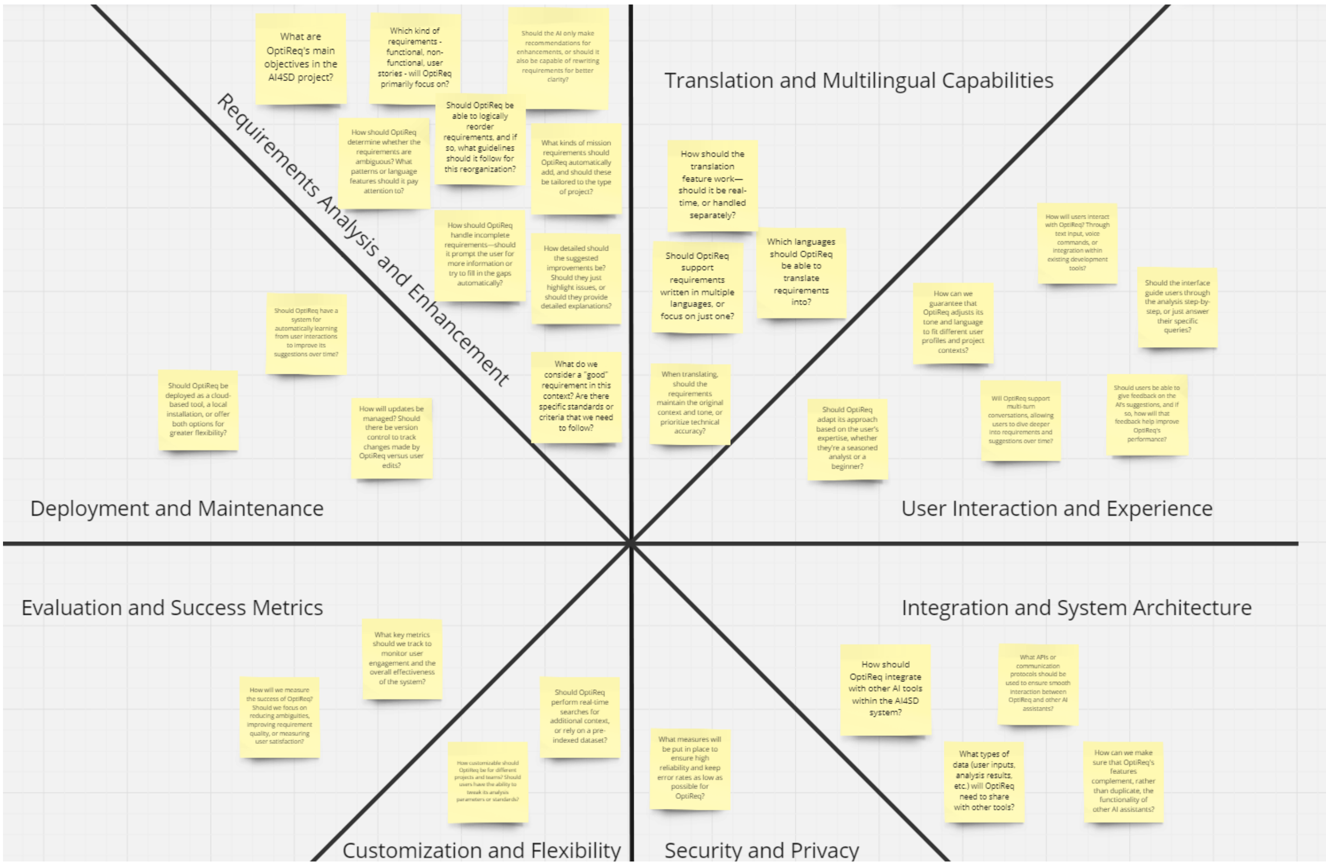
What key metrics should we track to monitor user engagement and the overall effectiveness of the system?

8. Deployment and Maintenance

Should OptiReq be deployed as a cloud-based tool, a local installation, or offer both options for greater flexibility?

How will updates be managed? Should there be version control to track changes made by OptiReq versus user edits?

Should OptiReq have a system for automatically learning from user interactions to improve its suggestions over time?


Requirements Analysis and Enhancement
Translation and Multilingual Capabilities
User Interaction and Experience
Integration and System Architecture
Security and Privacy
Customization and Flexibility
Evaluation and Success Metrics
Deployment and Maintenance

What are OptiReq's main objectives in the AI4SD project?

Which kind of requirements - functional, non-functional, user stories - will OptiReq primarily focus on?

Should the AI only make recommendations for enhancements, or should it also be capable of rewriting requirements for better clarity?

How should OptiReq determine whether the requirements are ambiguous? What patterns or language features should it pay attention to?

Should OptiReq be able to logically reorder requirements, and if so, what guidelines should it follow for this reorganization?

What kinds of mission requirements should OptiReq automatically add, and should these be tailored to the type of project?

How should OptiReq handle incomplete requirements—should it prompt the user for more information or try to fill in the gaps automatically?

How detailed should the suggested improvements be? Should they just highlight issues, or should they provide detailed explanations?

What do we consider a "good" requirement in this context? Are there specific standards or criteria that we need to follow?

How should the translation feature work—should it be real-time, or handled separately?

Should OptiReq support requirements written in multiple languages, or focus on just one?

Which languages should OptiReq be able to translate requirements into?

When translating, should the requirements maintain the original context and tone, or prioritize technical accuracy?

How will users interact with OptiReq? Through text input, voice commands, or integration within existing development tools?

Should the interface guide users through the analysis step-by-step, or just answer their specific queries?

How can we guarantee that OptiReq adjusts its tone and language to fit different user profiles and project contexts?

Should OptiReq adapt its approach based on the user's expertise, whether they're a seasoned analyst or a beginner?

Will OptiReq support multi-turn conversations, allowing users to dive deeper into requirements and suggestions over time?

Should users be able to give feedback on the AI's suggestions, and if so, how will that feedback help improve OptiReq's performance?

How should OptiReq integrate with other AI tools within the AI4SD system?

What APIs or communication protocols should be used to ensure smooth interaction between OptiReq and other AI assistants?

What types of data (user inputs, analysis results, etc.) will OptiReq need to share with other tools?

How can we make sure that OptiReq's features complement, rather than duplicate, the functionality of other AI assistants?

What measures will be put in place to ensure high reliability and keep error rates as low as possible for OptiReq?

Should OptiReq perform real-time searches for additional context, or rely on a pre-indexed dataset?

How customizable should OptiReq be for different projects and teams? Should users have the ability to tweak its analysis parameters or standards?

What key metrics should we track to monitor user engagement and the overall effectiveness of the system?

How will we measure the success of OptiReq? Should we focus on reducing ambiguities, improving requirement quality, or measuring user satisfaction?

Should OptiReq have a system for automatically learning from user interactions to improve its suggestions over time?

Should OptiReq be deployed as a cloud-based tool, a local installation, or offer both options for greater flexibility?

How will updates be managed? Should there be version control to track changes made by OptiReq versus user edits?