System Implementation Methodologies

- Incremental Development
- Research and Development: Invest and Dare
- Waterfall Development: Step by Step
- Rapid Prototyping: A quick solution for the best to come
- Parallel Development: Break the development and work concurrently for the solution
- Iteration Development: Develop the problem in general, and keep on solving by improving the solution.
- Partial Development: Develop only a part
- Spiral Development: Develop after deeply undemanding the problem
- Agile Development: Develop as you go and adjust the solving method to the circumstances and the challenges
- V Development: Requirements Validation based Development
- **Software development lifecycle**

Software Requirements Management

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

Non-Functional Requirements

Requirements, which are not related to functional aspect of software, fall into this category. They are implicit or expected characteristics of software, which users make assumption of.

Non-functional requirements include

- Security
- Logging
- Storage
- Configuration
- Performance

- Cost
- Interoperability
- Flexibility
- Disaster
- recovery
- Accessibility

Functional Requirements

Requirements, which are related to functional aspect of software fall into this category.They define functions and functionality within and from the software system.

EXAMPLES:

- Search option given to user to search from various invoices.
- User should be able to mail any report to management.
- Users can be divided into groups and groups can be given separate rights.
- Should comply business rules and administrative functions.
- Software is developed keeping downward compatibility intact.

User Interface requirements

UI is an important part of any software or hardware or hybrid system. A software is widely accepted if it is:

- Easy to operate
- Quick in response
- Effectively handling operational errors
- Providing simple yet consistent user interface

User interface requirements are briefly mentioned below:

- Content presentation
- Easy Navigation
- Simple interface
- Responsive
- Consistent UI elements
- Feedback mechanism
- Default settings
- Purposeful layout
- Strategical use of color and texture.
- Provide help information

- User centric approach
- Group based view settings

Must have VS. Should have VS. Could have

Other considerations

- **Partnerships needed?**
- **Future development plan**
  - future features and ambitions
- **Current development plan**
  - How the project will be develop with the requested funds?

Design Document

> A technical design document that will cover the project implementation plan with emphasis on the technology development methodology, the application's technical design, the user interface design, and the operations/ functionality process flow. The design document shall cover the operations and functionality flow in a creative and simplified ways that can user friendliness and usability.

Work breakdown structure

Task	Note
Project implementation plan	Choice of above mentioned
Technology development methodology	Choice of above mentioned
Application's technical design	No idea what he exactly means, guessing its about technical features or how those features are implemented design-wise, like the "enter code" feature..
UI design	Description of choices made seen in prototype
Process flow	Flowchart
Operations & functionality	Must, should, could have features of above mentioned

Working Prototype

done

Presentation

- Brief introduction into what the app is (in one sentence)
  - for whom (target audience)
    - user personas
- Relevant / important choices made in the prototype.
- Functionality outlined (must, should, could)
- UI explained
- Typical user journey
- Innovations applied
- Business stuff like revenue streams, pricing strategies (freemium, subscription model, premium version, ads), operating costs etc.

Gamification: once a voter's original song has been elected to be played next, the source user will receive a few "power points" as a reward for having suggested that song. Daily challenge is a trivia like quiz among friends, where the domain is the common music genre.