

Software Analysis & Design

Course Project - Guideline & Materials



Project

Objectives:

- Apply **Software Analysis & Design principles**
- Practice teamwork & collaboration
- Experience full system development cycle
- Develop critical thinking & problem-solving skills

General Guidelines:

- Form teams of **5 members**
- Select project via online form (**first-come, first-served**)
- Once chosen, project is **locked for your team**
- Full responsibility for:
 - a. Requirements
 - b. Analysis & Design
 - c. Implementation (prototype)
 - d. Documentation & Presentation

Deliverables

Working Prototype (web or mobile)

Documentation:

- Requirements Specification
- Use Case Diagrams
- System Architecture & Design Models
- Database Schema

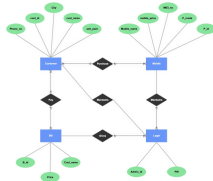
Materials & Tools

Analysis & Design:

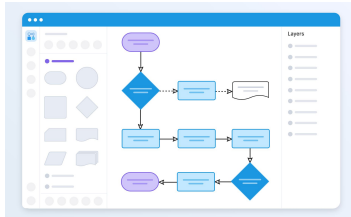
- **UML** (Use Case, Class Diagrams, Sequence Diagrams)



- **ERD** (Entity Relationship Diagrams)
-

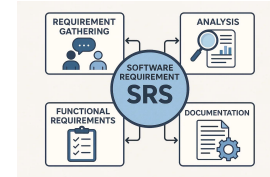


- **Flowcharts & Wireframes**



Documentation

- **SRS** (Software Requirements Specification) template



- **Git/GitHub** for version control:

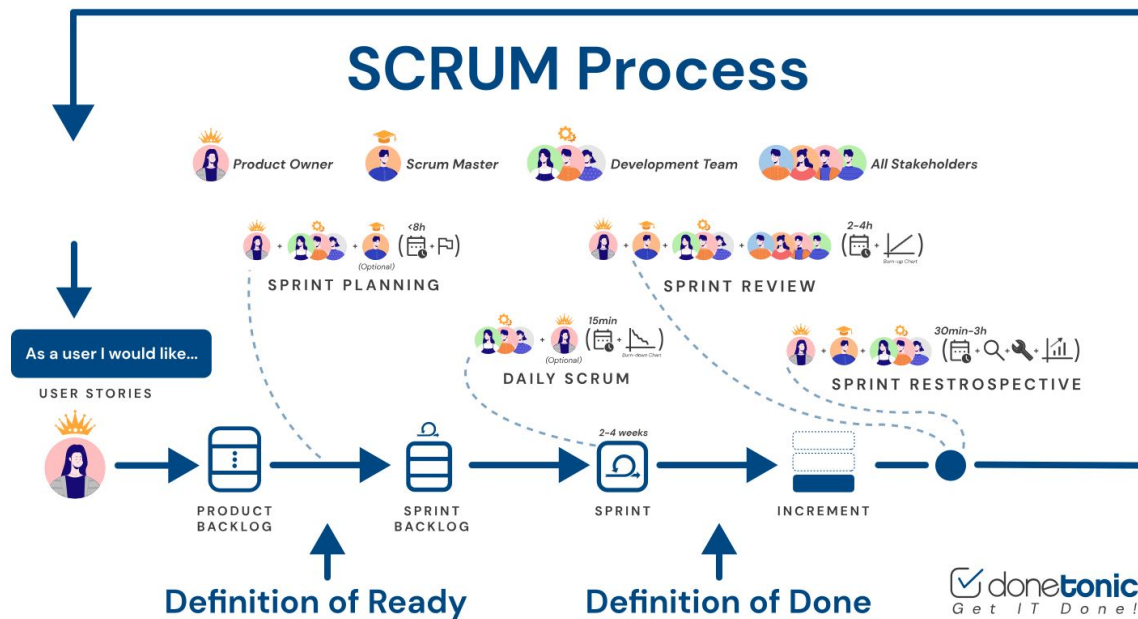


- **Postman** for API testing:



Methodology

Recommended Methodology:



How to apply Agile in your project

- **Sprint Planning:** Break project into 3–4 sprints (2–3 weeks each).
- **Backlog Creation:** List features → prioritize them.
- **Daily/Weekly Meetings:** Short updates on progress & blockers.
- **Sprint Review:** End each sprint with a demo.
- **Retrospective:** What went well? What can improve?

Agile & Software Tools

- **Jira** → Project tracking, sprint planning (industry standard)
- **Trello** → Simple Kanban boards (free, beginner friendly)
- **ClickUp / Asana** → Task management with timelines
- **Notion** → Documentation + task boards in one place
- **Slack / Google Chat** → Team communication
- **Figma** ⇒ For design and wireframing
- dbdiagram.io, **Lucidchart** → for diagrams
- **Replit** → online coding and collaboration environment



Evaluation Criteria

