

## In - Class activity 8: Group 1

### Group Members:

- Lindzey
- Kaden
- Chris
- Stuart
- Kat
- Ryan
- Carissa
- Sam
- Kaleb

### 10/22/24 Session 1 - Requirement gathering, design, and initial implementation

#### Sprint 1.1 - Scrum Master: Kaden, Report manager: Kaleb

- Scrum Meeting: Set up the project board and plan the initial coding tasks (**Kaden**)
- Create GitHub repository and set up the project structure:
  - src/, tests/, docs/, README.md. (**Chris**)
  - **Project Board**: Columns for Backlog, In Progress, Done (**Stuart**)
- Implement basic features such as a simple command-line interface (CLI).
- **Deliverable**: GitHub repository with initial code and project board.
- **Milestone**: Project setup (Version 0.1.1).

#### Sprint 1.2

- Scrum Meeting: Discuss user needs and gather functional and nonfunctional requirements.
- Create and finalize the Software Requirements Specification (SRS) document.
- Document 10+ user stories. (**Included Below, Ryan, Kat, Carissa, Lindzey**)
- Scrum Master: Leads the discussion and keeps the team focused.
- Report Manager: Documents the SRS and uploads it to GitHub.
- Deliverable: SRS document with user stories on GitHub.
- Milestone: Complete SRS (Version 0.1.2).

#### Sprint 2 - Scrum Master: **Chris**, Report manager: **Carissa**

- Scrum meeting: Plan the system design, focusing on user stories from sprint 1.
- UML Diagrams:
  - Use Case Diagram: **Kat**
  - Class Diagram: **Ryan**
  - Sequence diagram: **Sam**
- Design Pattern: Factory
- Scrum Master: Ensured each member is assigned a diagram task.
- Report Manager: Documents the diagrams and ensures they are uploaded to GitHub.
- Deliverable: Diagrams are uploaded to GitHub.
- Milestone: Complete the systems design. (Version 0.2)

Sprint 3A - Scrum Master: **Sam** Report Manager: **Lindzey**

- Scrum Meeting: Plan and assign coding tasks for adding and deleting tasks
- Implement features:
  - Adding tasks with title, description, and due date
  - Deleting tasks from the list
- Scrum Master: Monitors progress and resolves blockers
- Report Manager: Updates project board and ensures tasks are tracked
- Deliverable: Adding and deleting tasks implemented
- Milestone: Core feature implementation Part 1 (Version 0.3.1).

Sprint 3B - Scrum Master: **Sam** Report Manager: **Lindzey**

- Scrum Meeting: Plan and assign coding tasks for editing and completing tasks
- Implement features:
  - Editing task details (title, description, due date)
  - Marking tasks as complete
  - Filtering tasks (complete/pending)
  - Write test cases for features created in sprints 3A and 3B
- Scrum Master: Ensures progress is being made and adjusts tasks as needed
- Report Manager: Updates project board and ensures code is committed to GitHub
- Deliverable: Editing, completing, and filtering tasks implemented
- Milestone: Core feature implementation Part 2 (Version 0.3.2).

Sprint 4: Final Review: Scrum Master: Reporting Manager:

- Scrum Meeting: Plan final documentation (basic) tasks.
- Finalize the README.md with project overview, installation guide, and usage instructions.
- Scrum Master: Leads the final review and ensures completeness of the project.
- Report Manager: Ensures all documents and code are properly organized on GitHub.
- Deliverable: Finalized documentation.
- Milestone: Project complete (Version 1.0)

## Software Requirements Specification (SRS)

Project name: Task Manager Application

Date: 10/22/2024

Version: 0.2

By: Group 1

Revision History			
Version	Author	Version description	Date completed
0.1.2	Kaleb	Requirements	10/22/2024
0.2	Carissa	Design	10/24/2024
0.3.1	Lindzey	Core features 1	10/29/2024
0.3.2	Lindzey	Core features 2	10/29/2024
1.0		Complete Project	

Review history			
Approving party	Version approved	Signature	Date

Approval history			
Reviewer	Version reviewed	Signature	Date

## User Stories

1. As a user, I should be able to create a task with a title, description, and date.
2. As a user, I should be able to edit tasks' title, description, and date.
3. As a user, I want to be able to delete outdated tasks.
4. As a user, I'd like to see tasks in order of due date.
5. As a user, I'd like to be able to register using my email address and name.
6. As a user, I should be able to add a priority level to each task.
7. As a user, I would like to be able to mark tasks as complete
8. As a user, I'd like to be able to filter my tasks by view all, completed and pending.
9. As a user, I'd like to be able to share my task on social media.
10. As a user, I'd like to be able to login using my registration information.

## Functional Requirements

1. Display a list of tasks created by the user.
2. Whenever a user creates a task, it should automatically marked as incomplete
3. When a task is completed, it should still be saved to be viewed later.
4. Send Notifications when a task's due date is upcoming.
5. Have an option for a task to be recurring.

## Non-Functional Requirements

1. Users can have an email and password to login and see their tasks.
2. The user's data is encrypted

3. Users can have up to 1000 tasks
4. Each task can have subtasks.