**Requirements Document**

**Project Scope**

Take-Note is a webapp that is meant as a to-do list type application, it will allow users to add tasks to a list, edit them when needed and mark them as complete when done. It will also provide a calendar functionality that will allow users to record tasks to specific dates. A chat function will also be implemented to allow groups to talk and post links. Take-Note will store data locally on the user’s computer so when the browser tab is closed and re-opened the data will be retrieved.

**User needs**

The intended audience are students and professionals that need time management tools. The tool would need to provide a mix of features that allow the users to record and manage schedules and activities.

**Specific Requirements**

Functional Requirements:

* Be able to create a task
* Can mark the task as done
* Be able to remove a task
* Can see a list of completed tasks
* Chat area for group work
* Calendar to view team schedules

Non-Functional Requirements:

* Comprehensive testing
  + Unit testing
  + Integration testing
  + Usability
* Code review
* Version control with CI pipeline

Functional requirements – Strong Analysis

System design complete – appropriate diagrams (UML, Wireframes)

Implementation of all requirements, WebSocket’s and a database are used, and both works. Very few apparent bugs

Comprehensive code testing with both unit and integration tests, a thorough usability study has been performed and feedback has been taken on board. Other types of testing also used.

Version control is used well. CI Pipeline is well organized and has been demonstrated to work.

4 Days 20 hours = 92 Hours

5 hours for initiation Done

* Project proposal
* Git Repo
* Backlog

10 hours for Design

* Design document
* Requirements document
* Wireframes
* UML Diagrams

30 hours for Creation

* Database
* API
* Front-end
* Unit and integration tests
* Pipeline
* Heroku deployment

10 hours for testing

* Getting unit and integration tests properly done
* Usability testing

15 hours for polishing

* Fixing bugs and problems with front end
* Fixing bugs in back end of the app

10 hours for report and video

* Report written
* Video recorded