

## 2 Project Plan

### 2.1 PROJECT MANAGEMENT/TRACKING PROCEDURES

We will be using the agile project management style for our project. Agile is very commonly used in software, especially for both web and mobile application projects, which are also two of the deliverables for our project. Our goals and objectives will also likely be small, and incremental, which is another characteristic of agile project management.

We will use Gitlab to track version history and to allow for simultaneous development. Git issue boards will be used to organize and assign tasks. Alongside this, Gitlab milestones and epics will assist in tracking progress, and ensure the development process is moving forward in a timely manner. Group communication, useful resources, and meeting minutes will be organized in the team discord server. Important documents and reports will be stored in a shared Google Drive folder.

### 2.2 TASK DECOMPOSITION

The tasks necessary to implement our solution to Dynamic Spatial Crowdsourcing are outlined as follows:

**Task 1** - Finish studying background knowledge on Spatial Crowdsourcing

- a. Familiarize ourselves with the specifics of the project and problem statement
- b. Background research on similar projects to find best practices

**Task 2** - Complete requirements engineering process and planning.

- a. Identify relevant IEEE standards that must be followed
- b. Recognize users and build use-cases for project

**Task 3** - Complete review of client-provided studies and scope definition of project.

**Task 4** - Document and present original project plan for client approval

**Task 5** - Solidify design ideas and concepts for both UI and structural implementation.

**Task 6** - Finalize selection of application platforms, frameworks, languages, and tools.

**Task 7** - Complete documentation for the final design proposal, testing plans, and development/implementation plan.

**Task 8** - Specifics subject to change with time and client request.

- a. Implementation of the applications
- b. Revisions based on client feedback

**Task 9** - Final application testing should be completed.

- a. Final pre-launch preparations
- b. Presentation of application and appropriate documentation to stakeholders.
- c. Deployment of final product

### 2.3 PROJECT PROPOSED MILESTONES, METRICS, AND EVALUATION CRITERIA

The project will be evaluated with the use of standard agile charts and documentation. The development team and stakeholders may also use our non-functional requirements, as described in

the design document, for evaluation. Some of the specific criteria on which we will be evaluating the project's progress are:

1. User experience elements such as response time across all platforms
2. Design should be extensible
3. Data should be stored safely so that only authorized users may access it
4. Worker location data should be stored accurately in terms of proximity to exact location as best attainable by GPS sensors
5. Clean and best practice code to improve future maintainability
6. Application should function as expected under heavy load
7. Privacy and security of workers should be maintained
8. The UI for both web and mobile should be Intuitive and appealing to the users

These metrics will be organized and evaluated using the following tools:

1. Burndown Chart
2. Gantt Chart
3. Work Breakdown Structure (WBS)
4. Sprint Reports
5. Code reviews by peers
6. Web server load testing
7. JMeter (API testing tool)

Notable project milestones may be as follows:

### **September**

1. Familiarize ourselves with the specifics of the project and problem statement
2. Completed first stage of requirements engineering process

### **October**

1. Application scope and industry finalized
2. Project plan document finalized and accepted by client

### **November**

1. Solidified design ideas for both UI and structural implementation.
2. Selection of application platforms, frameworks, languages, and tools are finalized and documented

### **December**

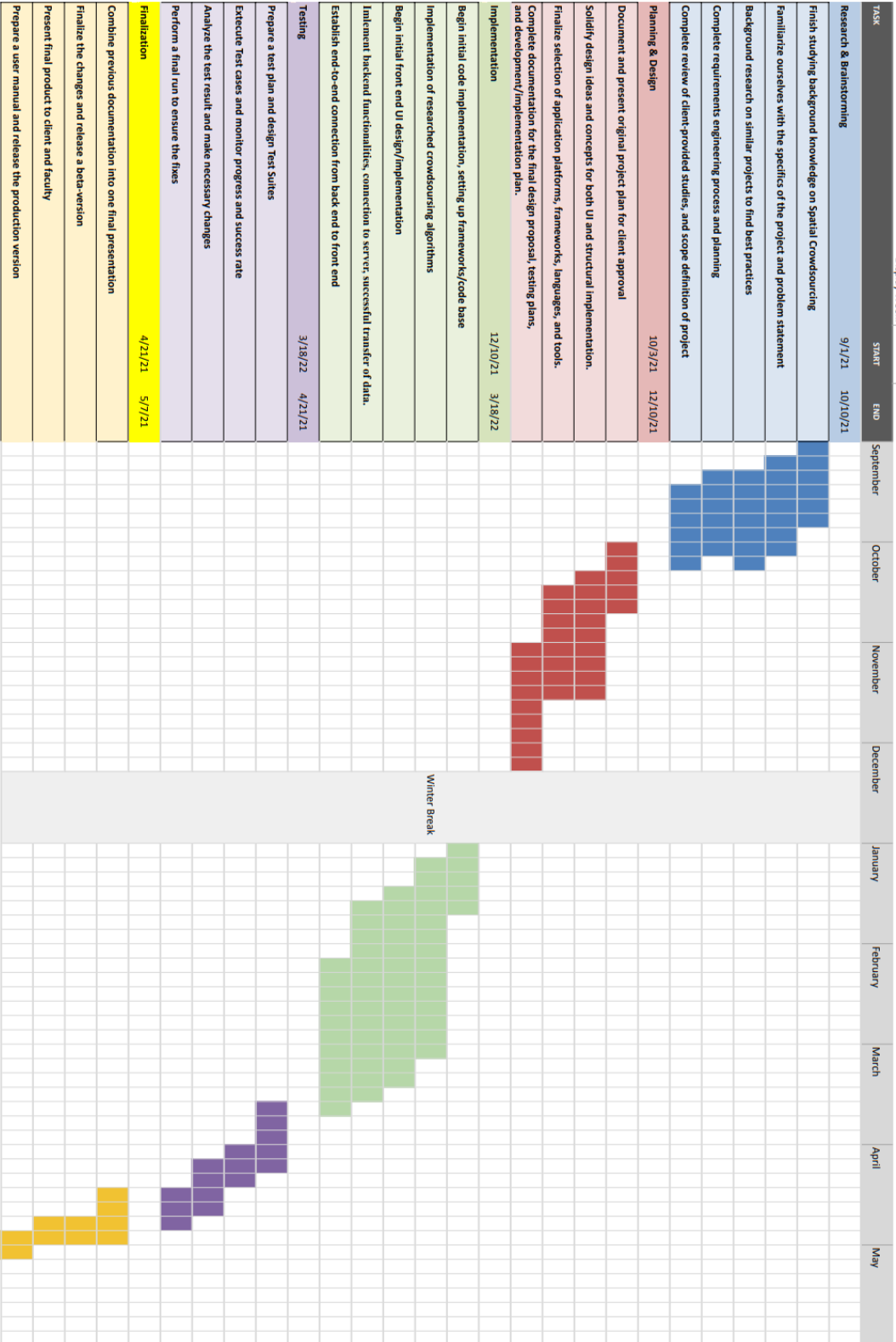
1. Documentation for the final design proposal, testing plans, and development/implementation plan completed

**January - May [*Specific dates to be determined*]:**

1. Worker location data controller optimized such that the data is reflected in real time. This means it shall stay within the standard margin of **1-10 seconds** to update.
2. Data controllers optimized such that all important changes should be disclosed to the relevant users in the same **1-10 second** real time window. An example of these changes would be the addition/removal of a task or driver
3. The worker assignment algorithm completed, and optimized for minimal runtime, such that it fulfills the proposed quality attribute. Clients now receive a timely UI response confirming this process has occurred.
4. The application can now smoothly support **X** users at once. (**X** being subject to capabilities of server provided by client)
5. The application's functionalities have all been optimized and connected smoothly. It now responds within the standard **1 second** time frame. If a request takes longer, some indication is provided to ensure the user something is happening and the app has not stopped.

Project Start: Wed, 9/1/2021

Display Week: 1



2.4 PROJECT TIMELINE/SCHEDULE

## 2.5 RISKS AND RISK MANAGEMENT/MITIGATION

Relative Severity	Relative Probability	Phase	Description	Mitigation	Rank*
8	8	Design	Client base size unknown	Conduct the market size research and expand the existing database accordingly.	8
1	6	Design/ Coding	Requirement not well defined cause wasting time on unnecessary features	Consult the decided use cases. Consultation with faculty. Adopt Agile refinements.	2
7	7	Design/ Coding	Task members don't like doing designated tasks	Negotiate optimal delegation of tasks	7
2	5	Design/ Coding	Task members are falling behind in their assigned task	Regularly check the progress of team members to ensure help on any difficulties faced	3
3	1	Coding	Developer inexperience with the required project technologies	Have dedicated learning time during work week and sharing knowledge between team members	1
4	4	Coding	Change in requirements aren't compatible with existing architecture	Implement a flexible and extensible architecture.	5
6	3	Training	Clients/Workers are unfamiliar with software	Create user-friendly tutorials on the app.	6
5	2	Testing	Tests not covering all algorithmic behaviours.	Writing quality tests. Using visualization tools for covering all program paths.	3

\*Rank = Relative Severity\*Relative Probability

## Risk Based

Firing Event	Probability	Cost in 1000s
Client base size unknown	0.2	1
Requirement not well defined cause wasting time on unnecessary features	0.6	10
Clients/Workers are unfamiliar with software	0.75	3
Developer inexperience with the required project technologies	0.8	7
Tests not covering all algorithmic behaviours.	0.75	5
Task members don't like doing designated tasks	0.4	2
Change in requirements aren't compatible with existing architecture	0.7	6
Task members are falling behind in their assigned task	0.7	8

To get an intuitive idea of what the impact of this risk assessment is, here is an approximate value -

$$a = 1.2 * 9 * 0.0043 + 3.5$$

$$= 3.55$$

$$\approx 4$$

Cost for the top 4 risks is \$31,000, which will be set aside as contingency funds.

But for this Senior Design project, we don't actually need to deal with these risk costs.

## 2.6 PERSONNEL EFFORT REQUIREMENTS

For each task, we are estimating the completion time considering all seven team members spending an average of half an hour a day per person.

Task	Estimated Completion Time (in hours)
<b>Sept. 1</b> - Finish studying background knowledge on Spatial Crowdsourcing	7 workers, 7 days $7 * 3.5 = 24.5$ hours
<b>Sept 10</b> - Familiarize ourselves with the specifics of the project and problem statement	7 workers, 3 days $7 * 1.5 = 10.5$ hours
<b>Sept 15</b> - Background research on similar projects to find best practices	7 workers, 3 days $7 * 1.5 = 10.5$ hours
<b>Sept 25</b> - Complete requirements engineering process and planning.	7 workers, 8 days $7 * 4 = 28$ hours
<b>Oct. 1</b> - Complete review of client-provided studies, and scope definition of project.	7 workers, 7 days $7 * 3.5 = 24.5$ hours
<b>Oct 10</b> - Document and present original project plan for client approval	7 workers, 14 days $7 * 7 = 49$ hours
<b>Nov. 1</b> - Solidify design ideas and concepts for both UI and structural implementation.	7 workers, 10 days $7 * 5 = 35$ hours
<b>Nov. 15</b> - Finalize selection of application platforms, frameworks, languages, and tools.	7 workers, 8 days $7 * 4 = 28$ hours
<b>Dec. 1</b> - Complete documentation for the final design proposal, testing plans, and development/implementation plan.	7 workers, 10 days $7 * 5 = 35$ hours
<b>Jan - May</b> - Specifics subject to change with time and client request. Implementation of the applications	7 workers, 100 days $7 * 50 = 350$ hours

May - Final application testing should be completed.	7 workers, 15 days  7 * 7.5 = 52.5 hours
--	--

## 2.7 OTHER RESOURCE REQUIREMENTS

- Remote web server to host the database and provide logic assignments, with one provided by the school no additional acquisition is required by this.
- Physical devices for testing of both the web client and the android applications, applicable devices are already owned by stakeholders meaning no additional device acquisition should be required.