



Project Plan

Date Submitted: 10/27/2021

Team Hobby:

Colin Creasman

Daniel Bribiesca

Team Lead: Jacob Delgado

Long Nguyen

Rifat Hasan

Table of Contents

Introduction.....	3
Project Plan.....	3
Project Goals.....	3
Due Dates.....	4
Risk Mitigation.....	5
Deadline Risks.....	5
Operational Risks.....	6
Technical Risks.....	7
External Risks.....	8
Other Risks.....	9
Resources.....	10
Development Team.....	10
Budget.....	10
Sprint Plan.....	11

Introduction

Purpose of Project Plan

- This document serves as a guideline to implement our project from start to finish, laying out a specific plan to identify and breakdown tasks with intended completion dates. While the project roadmap - a later section within this document - is a general timeline meant for our client, the project plan mainly provides details and information for our development team. The project plan will be used to keep track of and maintain a consistent workflow in order to reach our goals and meet specific deadlines.

Project Goals

- The goal of this project is to follow agile methodology using scrum to create a web based application that caters to hobbyists in a variety of fields that are stated in our product proposal. Ideally, we will:
 - Meet all deadlines
 - If a deadline is close to not being met, a resolution will be created to address the issue
 - Maintain communication with the client and development team members throughout the project
 - Follow scrum and provide documentation
 - Find and correct any errors prior to deployment
 - Create a functioning web application
- We hope to provide a quality application that may fill a need in the market that is both unique and useful.

Due Dates

10/4/21

- BRD
- Technical Specifications
- HL Design
- Site Level Map

10/27/21

- Project Plan
- Test Plan
- Network Diagram

12/8/21

- Low-Level (LL) Design
- UM
- Logging

1/17/22

- Research and education on programming languages and technologies being used for the project
 - Will need to know this before beginning the coding segment of our project

1/24/22 - 4/19/22

- Coding/Implementing Features
- Testing

4/20/22 - 5/6/22

- Testing
- Deployment - **4/25/22**
- Demo

Risk Mitigation

- This section is to address the potential risks involved throughout project research and development. Any future risks not listed in this section will be addressed and/or added as they are encountered. Each risk described will have a **risk level to quantify the impact on development (1 = High, 2 = Medium, 3 = Low)** along with details and how we plan on handling each specific risk.

Deadline Risk

- **Risk: Not meeting deadlines for tasks and/or deliverables**
- **Risk Level: 1**
 - Deadlines can be the only opportunity for the groups' grade.
 - The minimum amount of work needs to be finished in order to receive a passing grade.
- **What else does it affect?**
 - Sprints
 - Missing a deadline as a team delays other sprints by increasing workload for the next sprint.
 - If enough individual tasks are missed by the group members then it can completely ruin the next sprint and push back all of the scheduled work even further.
 - Internal Project deliverables
 - If core components of our project(Log in/out, Web server, Server database) are not completed by the time the internal due date comes, then all other tasks for group members will be delayed until it is completed.
- **Mitigation Threshold**
 - This risk should be acted upon and addressed immediately; in addition, it should be avoided at all possible means. Missing any deadline places us behind the project schedule in all instances.
- **What will be done to Mitigate this risk?**
 - Task checkups during daily meetings.
 - Our group meets on all business days out of the week.
 - During these meetings we check up on how each member is progressing with their work and if they

- need support from another team member that is ahead of schedule.
- Informing all members about vacation plans.
 - Some members have plans from before the semester.
 - As a group we create a shorter internal deadline for that member so that they can take their vacation worry free.

Operational Risks

- **Risk: Poor (or lack of) communication within development team**
- **Risk Level: 1**
 - Lack of communication leads to delays and possibly incomplete tasks for deadlines. This can lead to stress and tension within the group.
- **What else does it affect?**
 - Sprints
 - If the group doesn't properly communicate what needs to be done during sprints, then it's possible that not everything will be completed by the time the sprint ends. This causes other sprints to delay.
 - Internal Project deliverables
 - An example would be the group not knowing how to work with a new technology. The group could be assigned to research and learn the technology by a certain date. If anyone in the group didn't fully understand the research task, then deliverables can be delayed because they would have to take more time to learn.
- **Mitigation Threshold**
 - Excessive lack of communication can lead to a risk in meeting a deadline, incorrect assumptions, or incongruencies with our designs. This risk should be acted on before reaching this level.
- **What will be done to Mitigate this risk?**
 - Twice a week during the daily meetings, outlines for internal deliverables are being made by the team lead. This is to keep each member on track and focused with what they need to accomplish each week and by when.

- **Risk: Lack of communication with client regarding specific deliverables (and not receiving feedback)**
- **Risk Level: 2**
 - Inadequate communication with the client can lead to inaccurate, incomplete, or incorrect deliverables. Receiving feedback regarding each document can greatly increase the quality and overall points earned
- **What else does it affect?**
 - Sprints
 - If a deliverable is not being submitted early enough for feedback, it may delay progression, as we would essentially be working to complete the document blindly.
 - Cohesiveness:
 - Not receiving feedback for each deliverable can lead to unclear or incomplete documents, which will affect the overall understanding and clarity of our project as a whole
- **Mitigation Threshold**
 - Failing to communicate with the client can lead to issues in design and lost time. If this occurs often enough, it could impact the project schedule.
- **What will be done to Mitigate this risk?**
 - Sprint planning and structured task breakdowns for each deliverable will allow us to make enough progress to submit the documents for feedback at different points in the sprint.

Technical Risks

- **Risk: Inadequate knowledge of technology used**
- **Risk Level: 2**
- **What else does it affect?**
 - Sprints
 - If the group isn't knowledgeable enough or at all with the technology we chose, then it can cause severe delays with the amount of time they need in order to learn it.
 - Internal Project deliverables
 - Internal deliverables can be delayed or incomplete if the group members do not know what they are doing.
- **Mitigation Threshold**

- Issues with coding design due to this risk can lead to errors and incorrect/improper implementation. Though this may not impact the project schedule directly, it can lead to lost time and/or design flaws.
- **What will be done to Mitigate this risk?**
 - Our group has planned dates so that each member knows how to use the technology they need. An example would be React, our group is already planning to attend a workshop offered by the school in order to learn it.

- **Risk: Software changes during development**
- **Risk Level: 2**
- **What else does it affect?**
 - Sprints
 - A software change that breaks working code could heavily delay the current and future sprints. Because the problem is usually unknown and the scale of the issue is also unknown; taking the time to properly debug will take a considerable amount of time.
 - Internal Project deliverables
 - Internal deliverables could also be heavily affected. If a feature that was working before that update breaks, then the team member now needs to figure out how to fix it and make it work with the latest software.
- **Mitigation Threshold**
 - Any issues due to this type of risk can immediately impact the project schedule and should be avoided if at all possible.
- **What will be done to Mitigate this risk?**
 - Our group is focused on one thing, version history. We use github integration in order to monitor version history and will use it as reference in order to debug our code should issues arise after an update to software.

External Risks

- **Risk: Unforeseen events affecting team member(s) progress**
- **Risk Level: 1**

- **What else does it affect?**
 - Sprints
 - Because team member story points are so important, if an event that reduces their story points happens. It could lead to small or large delays or the current and future sprints.
 - Internal Project deliverables
 - Each member has a limited amount of story points per sprint. If the story points are reduced or the member was planning on finishing a large portion of their deliverable on that day, then it could cause delays internally.
- **Mitigation Threshold**
 - This type of risk might not be avoidable and should be addressed as they occur. Mitigation should prevent team from falling behind schedule
- **What will be done to Mitigate this risk?**
 - Our group follows the outline and communicates daily regardless of a meeting taking place that day or not. We have set closer internal deadlines for portions of the deliverables in order to stay on track.

- **Risk: Team member drops (or is unable to complete) course**
- **Risk Level: 1**
- **What else does it affect?**
 - Sprints
 - The amount of work each person needs to be done has to be reassessed based on the remaining team members' story points. This could cause small to large delays of the current or future sprints.
 - Internal Project deliverables
 - A member could have tasked themselves with a certain deliverable needed for a deadline.
- **Mitigation Threshold**
 - If this happens, regardless of timing, it could put the project at risk of falling behind.
- **What will be done to Mitigate this risk?**
 - Our group members are asked once a month if they still intend on taking this course. We continue to communicate with each other and make sure we are all on the same page.

Other Risks

- **Risks not stated within project plan**
- **Risk Level: 1 - 3**
 - Any risks not listed within this project plan will be assessed as soon as possible to minimize impact and to create a course of action. Throughout development, there are numerous risks that may or may not occur; as a development team, we will try to predict those that we can and intend on acting immediately to resolve those we cannot.
 - When a new risk has the possibility of occurring (or is occurring), the team member(s) will communicate it to the rest of the team as soon as possible so we can meet to discuss and enact the correction.

Resources

Development Team

- All Members of the development team will be working on research, deliverables, coding, and all necessary tasks to complete the project
- Projected capacity over 7 days without vacations, holidays, or other factors that could lessen a team member's capacity
 - **Colin Creasman - Meeting Notes**
 - Capacity: 12hrs/week
 - **Daniel Bribiesca - Utility**
 - Capacity: 12hrs/week
 - **Jacob Delgado - Team Lead**
 - Capacity: 17hrs/week
 - **Long Nguyen - Scrum Master**
 - Capacity: 17hrs/week

- **Rifat Hasan - Utility**
 - Capacity: 12hrs/week
- **Group Weekly Total:** 70 hrs

Budget

Name	Cost	Quantity	Total Cost
Amazon AWS Server Free Month	\$0	1	\$0
Amazon AWS Server (Billed Monthly)	\$3	7	\$21
Amazon AWS Education Credit	-\$100	1	-\$100
Software Licensing	0	0	0
External Libraries	0	0	0
Total			-\$79

Sprint Plan

Sprint 2: 10/12/21 - 10/26/21

Team Capacity: 140 hrs

- Test Plan
- Project Plan
- Network Diagram

Sprint 3: 11/1/21 - 11/15/21

Team Capacity: 145 hrs

- DAR for RDS
- Core Components

- Project Plan - **Rework**
- Site Map - **Rework**
- BRD - **Rework**

Sprint 4: 11/16/21 - 11/23/21

Team Capacity: 130 hrs

- Low-Level (LL) Design
- UM
- Logging
- Archiving
- HL Design - **Rework**

Sprint 5: 11/30/21 - 12/14/21

Team Capacity: 152 hrs

- **DAR for Testing Library**
- UM
- Logging
- Archiving
- Testing

****Note Regarding the UI:** The UI and specific UI components will be built throughout our development as we implement each feature. The related UI and UI components are expected to be complete by the end of the sprint they are being implemented**

Sprint 6: 1/20/22 - 2/3/22

Team Capacity: 150 hrs

- Implementing
 - Environment: 1/20/22-1/24/22
 - Server
 - Design
 - Setup
 - Deploy
 - Front End/Back End
 - Design
 - Setup
 - Deploy

- Questionnaire: 1/24/22 - 1/27/22
 - Skills
 - Skill Levels
 - Tool List
- Project Categories: 1/27/22 - 1/30/22
- Login: 1/30/22 - 2/1/22
- Testing
 - Questionnaire: 2/2/22
 - Account Creation: 2/2/22
 - Login: 2/2/22

Sprint 7: 2/4/22 - 2/18/22

Team Capacity: 140 hrs

- Implementing
 - Pulling Resources
 - Post/Delete Project or Draft
 - Save Project as a Draft
 - Logging
 - Data Store
- Testing
 - Post Project
 - Collect Projects from External Source
 - Display Collected Projects
 - Upload a File to Project Post
 - Remove File from Project Post
 - Post a Saved Project
 - Save a Project Draft
 - Delete a Project or Project Draft
 - Logging
 - Data Store (Ex: Projects, Ratings, Comments)

Sprint 8: 2/19/22 - 3/5/22

Team Capacity: 150 hrs

- Implementing
 - Search/Discovery Page
 - Notifications
 - Notification page
- Testing
 - Search

- Notification Page
- Notification Alerts
- Apply Search Filter
- Apply Discovery Filter
- Public UI Controls
- Archiving (What are we storing?)
- Edit a Posted Project or Saved Draft

Sprint 9: 3/6/22 - 3/20/22

Team Capacity: 130 hrs

- Implementing
 - Recommendation Section
 - Tool Recommendations
 - Project Recommendations via notification page
- Testing
 - Receive a Tool Recommendation
 - Receive a Project Recommendation through notification
 - Receive a Project Recommendation

Sprint 10: 3/21/22 - 4/4/22

Team Capacity: 125 hrs

- Implementing
 - Filters
 - Comments
- Testing
 - Filter Comments
 - Uploading a File to a Comment
 - Posting a Comment

Sprint 11: 4/5/22 - 4/19/22

Team Capacity: 150 hrs

- Implementing
 - Ratings
 - Upgraded Tool List with Search Functionality
- Testing
 - Private UI Controls
 - Edit a Comment Rating
 - Rate a Project (or edit current rating)
 - Rating a Comment

- Test Upgraded Tool List

Sprint 12: 4/20/22 - 5/1/22

Team Capacity: 120 hrs

- Testing
- Deployment

Sprint 13: 5/2/22 - 5/6/22

Team Capacity: 50 hrs

- Demo