Project 2 Plan

CIS 422/522 Software Methodologies

Group 5 - 5-29-2020

Irfan Filipovic (IF), Shane Folden (SF), Man Him Fung (MF), Mason Jones (MJ), Siqi Wang (SW)

This document will define how the group is managed, the process in which the system will be built, and what risks are possible during. Each header will define what is contained below and be followed by questions which further describe the content. Risks defined in the last section do not encompass all possible scenarios, but establish a basis for more likely ones.

Project Plan

A management plan. How is your team organized? How is the work divided among team members? How does your team make decisions? How will your team meet and how will it communicate?

To develop our system, our group is split into two main teams, the website/database team and the mobile application team. The website/database team consists of Mason and Man Him, and Irfan, Siqi, and Shane are the mobile application team.

Each member of each team has specific tasks to complete. Tasks are assigned according to previous knowledge, accessible technology (such as Xcode), and desire to implement a certain feature. Although tasks are assigned to specific members, they can be transferred to other members or developed collaboratively. When a group member runs out of tasks, they see which task needs the most attention and begin to assist with it. Certain members are more likely to complete work early and expect to assist with other members' tasks.

All major functional decisions are first discussed in their respective teams. After a discussion has been had and the team arrives at a consensus, they consult the other team as to their opinions. If the other team has no objections the consensus is used as the group's final decision.

As a group we created a team on the application Microsoft Teams, which supports file sharing and group calls. Every Tuesday and Thursday a group call is started between 3:00 p.m and 4:00 pm. An additional meeting happens every weekend but does not have a concrete time.

In these calls, project progress is discussed and files are reviewed. On request by any group member, an additional meeting may be set to allow further discussion or assistance.

Work breakdown schedule (with > 10 milestones) and project schedule (who will do what).

(* may change during work process)

Task	Assignment
Ping location on start	IF
Update application user state visuals	SF,IF
Implement setting location	SF
Add application time at location	IF
Create Testing View on App	SW
Implement mapbox layer for stores	MJ
Implement average time at location	MF
Implement people at location	MF
Implement altering time interval	SW
Implement at time in line	SF
Implement time waiting for food	SF

<Table 1>: Tasks assigned to which members. Collaboration and changes may occur.

Monitoring and reporting: How individual and project progress will be monitored to keep track of who did what and when did they do it?

Progress will be tracked utilizing a Gantt chart, each task will be displayed with date assigned, due date, date completed, and the members working on the task.

The data used to populate the Gantt chart is in an excel file which will be updated with more information, such as percent complete.

A build plan. What is the sequence of steps you will take to build the system? When will each "build" of the system take place?

TASK NAME	START DATE	DAY OF MONTH*	END DATE	DURATION* (WORK DAYS)	DAYS COMPLETE*	DAYS REMAINING*	TEAM MEMBER	PERCENT COMPLETE
Project 2								
3-Page Proposal	5/11	11	5/11	1	1	0	Group work	100%
Initial SRS - V1	5/11	11	5/15	4	4	0	Group work	100%
Initial SDS - V1	5/11	11	5/15	4	4	0	Group work	100%
Initial project plan - V1	5/11	11	5/15	4	4	0	Group work	100%
Generate Full Project Timeline	5/11	11	5/15	4	4	0	Group work	100%
Ping location on start	5/14	14	5/17	3	3	0	IF	100%
Update application state visuals	5/14	14	5/23	9	1	8	SF, IF	100%
Implement setting location	5/14	14	5/23	9	1	8	SF	100%
Add application time at location	5/14	14	5/17	3	3	0	IF	100%
Create Testing View on App	5/14	14	5/20	6	1	5	SW	100%
Implement mapbox layer for stores	5/14	14	5/25	11	1	10	MJ	100%
SRS - V2	5/14	14	5/25	11	1	10	Group work	100%
SDS - V2	5/14	14	5/25	11	1	10	Group work	100%
Project Plan - V2	5/14	14	5/25	11	1	10	Group work	100%
Implement average time at location	5/17	17	5/21	4	0	4	MF	100%
Implement people at location	5/17	17	5/23	6	0	6	MF	100%
Implement altering time interval	5/17	18	5/23	5	0	5	SW	100%
Implement at time in line	5/17	18	5/23	5	0	5	SF	100%
Implement time waiting for food	5/17	18	5/23	5	0	5	SF	100%
Implement changing database address	5/17	18	5/23	5	0	5	SW	100%
SRS - V3	5/30	30	6/1	6	0	6	Group work	100%
SDS - V3	5/30	30	6/1	6	0	6	Group work	100%
Project Plan - V3	5/30	30	6/1	6	0	6	Group work	100%
User Documentation	5/30	30	6/1	6	0	6	Group work	100%
Programmer's Documentation	5/30	30	6/1	6	0	6	Group work	100%
Final Submit	5/31	41	6/1	1	0	1	Group work	100%

<Figure 1>: Chart describing the build plan.

A rationale for the build plan. Why have you broken the system into these parts, and why have you chosen these particular steps to build the system? What are your risks and your risk reduction strategies?

Each component will be integrated after development, following the Gantt chart deadlines, in order to provide the maximum time for testing the component. The order in which components are developed is described below.

The system has been broken into specific functions that must be implemented; these are defined as requirements in our software requirements specification. Each feature must be implemented to produce a cohesive system that assists users.

The first task to be started is updating app state visuals as user experience is one of the most important metrics of customer satisfaction and requires much testing. After completing this, the next task will be improving upon the already established tracking system as well as adding a new in-app feature: developer testing within the app.

The developer testing interface will allow for the structure to develop changing time interval and database functionalities. Once the improvements to tracking have been completed,

the system will be reviewed. This will allow for a user friendly application that can be tested and receive new features as they are developed.

Possible Risks:

- Group Event
    - **Risk:** A member of the group is not able to complete their assigned task for one of several reasons, such as sickness.
    - **Risk Reduction Strategy:** Each member will communicate their progress and if any issues arise during the project development. If issues arise the group will either assign another member to assist or provide helpful documentation and comments to guide the member.
    - **Risk:** We are not able to find any times where all members of the group are available to meet
    - **Risk Reduction Strategy:** We will have scheduled meetings which each member will be aware of. Any additional meetings will require a member to take meeting notes in order to update any member that isn't present.
- Location Authorization and Tracking
    - **Risk:** The user's location is secured and cannot be accessed by the application.
    - **Risk Reduction Strategy:** The user will be instructed to accept location services, and the user will be notified if location is not being tracked. The user may then reach out and have the development team correct the specific scenario's issue.
- Data Loss
    - **Risk:** The database storing user data loses all information, and we are left with no data to work on or distribute.
    - **Risk Reduction Strategy:** Have a backup of the database that backs up the primary database every 6 hours to ensure no more than a day of data is ever lost.
- Privacy
    - **Risk:** There is a large risk of exposing private data, such as current location and location of frequent locations. Someone may be able to force access to our data which will expose a unique device identifier, and location pings with times.
    - **Risk Reduction Strategy:** We will transmit data using requests on secure connections (HTTPS), and the user will not be distinguishable on the map. Therefore if data is accessed by unauthorized individuals, there is little to no harm that may be conducted.
    - **Risk:** Because the location of all users is visible through the website, it may be possible to determine a specific user's home address

    - **Risk Reduction Strategy:** Users will have the option to indicate when they are at home, which will offset their location by a set amount to obfuscate their home address and make it difficult to determine where a user lives.
- Data set
    - **Risk:** There is a risk that we do not have enough data to provide accurate peak traffic times
    - **Risk Reduction Strategy:** Using smaller areas for testing , and using repeated location data to simulate the people who have stable life schedules for each day, and randomly create the data for the people who do not. We can test how reliable our system is by this.
- API Implementation
    - **Risk:** When a user enters the wait time at a business, the app will use Apple's MapKit API to determine the address of the business they entered. It is possible that this API may not function the way we expect or is difficult to implement under the current time constraints
    - **Risk Reduction Strategy:** We will explore other options that can yield the same result, such as having the user enter the address manually.