

Table of Contents

- 1. Introduction**
 - 1.1. Project Overview
 - 1.2. Project Deliverables
 - 1.3. Evolution
- 2. Project Organization**
 - 2.1. Organizational Structure
 - 2.2. Project Responsibilities
- 3. Managerial Process**
 - 3.1. Management Objectives and Priorities
 - 3.2. Assumptions, Dependencies, and Constraints
 - 3.3. Risk Management
 - 3.4. Monitoring and Controlling Mechanisms
- 4. Technical Process**
 - 4.1. Methods, Tools, and Techniques
 - 4.2. Version

1. Introduction

1.1 Project Overview

Laze is an application containing a live and dynamic GPS map overview of Wilfrid Laurier University. It allows users to pin points of interest, such as study spaces, restaurant lines, parking spots, etc - all of which are visible to all other users of the application. These pins enables users to view indicated information such as the length of a line or the availability of a parking spot nearby. Users have the ability to support the validity of an active pin or flag to decrease its time before expiry.

1.2 Project Deliverables

The following table outlines a list of tasks and their respective deadlines.

no.	Deliverable	Description	Adjusted Deadline (2018)
1	Gathering Requirements and Analysis	Customer will state their expectations including who will use the product, how it will be used, and any custom specifications such as app design, look and feel, logo, app functionality, etc	Fri., Sep. 28, 2018
2	Requirements and Analysis SQA	Team analyzed the requirements to ensure that it can be included into the software and app without causing breaks or issues with system functionality. The SQA Team determined testability, ambiguity, validity and verification.	Mon., Oct. 1, 2018
3	GUI Prototype	Team designed an essential user interface such as different screens (i.e. log-in page, study spots page, etc). This will initially be done on paper and then shifted toward screen designs. (Also includes integration of UML diagrams and transitioning into concrete UI prototypes such as HTML pages)	Fri., Oct. 19, 2018
4	GUI Prototype Validation	SQA Team ensured that the prototype best reflects user requirements and tested it to ensure no breaks or leaks in designs (i.e. background color, font, # of items/page, etc)	Mon., Oct. 22, 2018
5	SQA for requirements and analysis document	Allowing 2 days for quality assurance of all documents thus far; updating content and grammar checks	Wed., Oct. 24, 2018
6	Design	Includes design review & validation, interaction diagrams, model class diagrams, as well as a	Fri., Oct. 26, 2018

		detailed client/user design	
7	Development /Coding	This is a 2 week cycle where the customer requirements are divided into units (i.e. front-end, back-end, server, GUI, etc) and code is written.	Dev Start: Nov 5 First Beta: Nov 18 Second Beta: Dec 3
8	Testing	Features the SQA team testing the code provided by the devs after the first beta and making according bug/feature fixes. This is an ongoing and continuous process.	Starts: Dec 10 Ends: Dec 14
9	Contingency	This fixes any problems found in second beta and includes User Acceptance Testing (UAT).	December 10 - 15
10	Gold Candidate Build	The project has been completed at this point and has all previously mentioned features working.	December 15
11	Project Deliverable / Deployment	The project is complete and all documentation should be properly updated, and the project shall be presented for consumers.	December 16
12	Maintenance	N/A in terms of Laze. However, this is normally when production fixes or newer version with updated features are released.	

1.3 Evolution

Within the main group consisting of 26 members, subgroups each with their own corresponding objectives have been formed. Weekly updates on Slack and Github go over current status of deliverables and the progresses of each subgroup.

2. Project Organization

2.1 Organizational Structure

The project deliverables have been worked in collaboration of members from their respective groups, for instance, members from the front-end and docs team. This is because our organizational structure is based on the project responsibilities. The teams in which we work consist of different components of this project such as: Front-end, Back-end, SQA, Server, Graphic Design and Documentation.

2.2 Project Responsibilities

Group Members	Deliverable
Jay, Anthony, Ashaab, Binoli, Marek, Ralph, Sakshi, Mansi, Judy	Requirements & Analysis
Jay, Kumiko, Clayton, Tomiwa, Daniel	GUI Prototype
Luke, Mansi, Jay, Brayan, Ashaab	SPMP
Tomiwa, Marko, Clayton, Kumiko, Jay, Luke, Mansi, Ralph	Design
Kumiko, Clayton, Ralph, Tomiwa, Shawn, Alejandro, Brayan, Erwin, Tayania, Charmi, Desai, Binoli, Katie, Max, Ashaab, Judy, Daniel, Mansi	Implementation & Development
All members will participate in SQA/Testing	Software Quality Assurance/Testing

3. Managerial Process

3.1 Management Objectives and Priorities

The main objective of this project is to develop an application that enables users to view an interactive map of the Wilfrid Laurier University Campus with live updating pins containing useful information in many areas of interest. At the end of the term (December, 2018) our team goal to accomplish all project deliverables and have a working version of our application available on web browser has been met. Having consistency in our documentation and updating objectives and timelines has been at the top of our list of priorities ensuring proper allocation of time and accountability, as there is no budget for this project.

3.2 Monitoring and Controlling Mechanisms

Following lectures on Monday and Wednesday, updates were communicated through group channels on Slack. Meetings to go over deliverables took place on an agreed-upon time and location. This allowed for effective group work requiring multiple members to be present simultaneously - i.e. implementation and testing. All accountability and time tracking has been recorded on a Google Doc and submitted in HTML on a minimum weekly basis. Testing and SQA has been completed by non authors promptly following the completion of a deliverable. This ensured the team is working diligently to correct any errors before they cause any larger setbacks.

4. Technical Process

4.1 Methods, Tools and Techniques

Backend: Django

Frontend tools and libraries: Google Maps Javascript API, Bootstrap, JQuery

Languages: Python, Javascript

Webpage Building: HTML, CSS

Repositories: GitHub

GUI Prototype Tool: Figma

Project Management: GitHub, Google Docs, Slack, Facebook

4.2 Versions

Version 1.0 - 15/9/2018

Document outline created, content added throughout the week

- Luke, Mansi, Jay, Brayan, Ashaab, Judy

Version 1.2 - 10/12/2018

Dates edited, overall structure reviewed

- Mansi, Sakshi

Version 1.3 - 15/12/2018

Final Copy

- Ashaab, Mansi