

**Software++ (Group 15)**

# **Parking Lot++**

**Project Charter**

**Version 1.0**

**October 7th, 2021**

## Document History

Version	When	Who	What
0.1	01/10/2021	Dan Melamed	Created initial Project Charter template
0.2	04/10/2021	Nilay Sondagar	Updated headings and general formatting
0.3	04/10/2021	Nilay Sondagar	Added <i>Project Schedule</i> , <i>Deliverables</i> , and <i>Project Approval</i>
0.4	05/10/2021	Vyom Shah	Updated the <i>Context</i>
0.5	05/10/2021	Dan Melamed	Added <i>Scope</i> and <i>Objectives</i>
0.6	06/10/2021	Franklin Yang	Added <i>Risk</i> and <i>Stakeholders</i>
0.7	06/10/2021	Abdullahi Bashir	Added <i>Needs</i> and <i>Team Organization &amp; Roles</i>
0.8	06/10/2021	Dan Melamed	Made revisions to <i>Risk</i> and <i>Stakeholders</i>
0.9	06/10/2021	Logan Raffkind	Added <i>Milestones</i> and <i>Work Breakdown Structure</i>
0.10	06/10/2021	Vyom Shah	Added <i>Executive Summary</i>
1.0	06/10/2021	Everyone	Final Review

## Table of Contents

<b>1.0 Project Overview</b>	<b>3</b>
1.1 Executive Summary	3
1.2 Context	3
1.3 Need	3
1.4 Scope	4
1.5 Stakeholders	4
1.6 Objectives	5
1.7 Glossary	5
<b>2.0 Project Approach</b>	<b>6</b>
2.1 Team Organization & Roles	6
2.2 Work Breakdown Structure	7
2.3 Milestones	7
2.4 Project Schedule	8
2.5 Deliverables	8
2.6 Risks	9
<b>3.0 Project Approval</b>	<b>10</b>
3.1 Stakeholders Sign-off	10
3.2 Team Members Sign-off	10

# 1.0 Project Overview

## 1.1 Executive Summary

The goal for Software++ (The Analysts) is to design and develop a website (The System) for The University of Faketoria (The Client) that allows end users to find parking lots with available parking spots in real-time. The Client's current system only provides static maps of their parking lots and lacks any way to determine parking lot availability. The System will help end users find parking lots for their vehicles, display a map of all parking lot locations, show the number of available parking spots, and send notifications to end users when their preferred parking lot has availability. This System is expected to complete the requirements gathering phase by November 28, 2021, with development starting shortly after that. The System's full rollout is expected to commence in late 2022, and will be officially referred to as *Parking Lot++*.

## 1.2 Context

The Client provides multiple parking lots on their campus that allows students, staff, faculty members, and visitors to park their vehicles. This allows for easy access to The Client's campus, buildings, and amenities. The Client uses ticketing booths in each parking lot to allow drivers to pay for a parking spot. The Client has multiple options for parking such as *General Parking*, *Long Term Parking*, *Short Term Parking*, and *Reserved Parking*. The Client provides maps physically around campus and virtually on their website. These maps show all parking lot locations on campus, with no information about the availability of parking spots.

## 1.3 Need

The Client's current parking lot setup includes payment booths and maps that indicate the location of each parking lot. However, these maps simply display the location of each lot, and no additional information such as the availability of parking spots. This leads to difficulties in finding parking spots, heavy traffic on campus roads, loss of time and driver frustration. The Client would like to find a solution that can solve this problem, creating a better parking experience for all of their visitors. Each parking lot has been retrofitted with sensors to detect the entering and exiting of vehicles. This data is then aggregated and emitted via an internal API in preparation for a solution.

## 1.4 Scope

As there is an unlimited budget allocated to this project, the main project boundaries pertain to time. In order to create a minimum viable product within the given timelines, the following functionality has been deemed in and out of scope:

### **In scope**

- The System will be designed as a web application.
- The System will store the minimum user information it needs to function correctly.
- The System will display the available capacity of each parking lot.
- The System will display the locations of each parking lot on a map.
- The System will contain a simple notification system for alerting end users of parking lot availability.

### **Out of scope**

- The System will not show the specific location of each available parking spot in a parking lot.
- The System will not implement any functionality required for reserving parking spots.
- The System will not take monthly or yearly parking passes into account.
- The System is not required to implement a payment method for the purchase of parking, although this may be optionally added if time permits.

## 1.5 Stakeholders

The Senior Systems Analyst at the University of Faketoria and the university's students are the main focus of this project. The Senior Systems Analyst will oversee and maintain The System, and the students will be considered the end users.

### **People directly affected by the project:**

1. *Project Sponsor:* The project sponsor decides whether the project moves forward based on how it's implemented, and by evaluating the current system in place.
2. *The Security Department of Faketoria:* The university's security department will manage and update the system. If The System is not easy to use or has other problems, they may request that a new system be built.
3. *Students and Staff of Faketoria:* The students and staff are the major group influencing the success of the project. If the system helps them save time when searching for an available parking spot, they will support the continuation of the project.

## People indirectly affected by the project:

1. *Environmentalists*: Reducing the amount of cars circling campus roads searching for available parking spots will have a net positive impact on the environment.
2. *Pedestrians*: The System aims to reduce congestion and improve efficiency, which creates a safer environment for pedestrians attempting to cross campus roads.

## 1.6 Objectives

A basic implementation of The System must, at the very least, meet these basic objectives:

1. The System must always show the number of spots available for each individual parking lot at The Client's campus.
2. The System must be able to update parking availability for each parking lot in real-time.
3. The System will show the end users a map of The Client's campus, detailing the location of each parking lot.
4. The System will use the hardware sensors placed at the entrances and exits of each parking lot in order to determine the number of vehicles in each parking lot.
5. When the parking is below 95% capacity, The System will mark that parking lot as "Available" to indicate to end users that there are available parking spots in that parking lot.
6. When a parking lot is at or above 95% capacity, The System will mark that parking lot as "Full" to warn end users that finding a spot at that lot may be difficult.
7. The System must allow end users to save a parking lot as their preferred parking lot.
8. The System will have a notification feature that alerts end users when their desired parking lot becomes available.
9. The System will communicate with The Client via an existing API to mark all parking lots as "Full" during holidays and closures.

## 1.7 Glossary

<b>Available / Availability</b>	Indicates whether a parking spot is not currently occupied by a vehicle, or indicates that a parking lot has space for more vehicles to park in it.
---------------------------------	---

<b>End User</b>	Any entity who uses the finished Parking Lot++ web application to find available parking spots. Most end users will consist of students at the University of Faketoria.
<b>Parking Lot</b>	A collection of parking spots that is located at the University of Faketoria. There are multiple parking lots found across their campus, each holding multiple parking spots.
<b>Parking Spot</b>	A spot in a parking lot that can hold a single vehicle. There are multiple parking spots in a parking lot.
<b>Real-time</b>	Refers to interactions and responses that must be completed under 1 second, measured from the start of processing.
<b>The Analysts</b>	Refers to the analyst organization (Software++).
<b>The Client</b>	Refers to the client organization (The University of Faketoria).
<b>The System</b>	The Parking Lot++ web application that will be utilized by end users to find available parking spots in parking lots found at the University of Faketoria.

## 2.0 Project Approach

### 2.1 Team Organization & Roles

Each member of The Analysts, along with their role and description, are listed below for communication purposes.

Member	Role	Role Description
Nilay Sondagar	Director++ of Technical Compliance	Supervises the technological infrastructure that Software++ administers for their clients, as well as manages their projects on a high level.
Abdullahi Bashir	Systems Analyst	Works with clients to provide feedback on their project and analytical deliverables. In addition, he assesses and describes requirements for proposed systems through an iterative prototype design and evaluation process in collaboration with the customer.

Dan Melamed	Technical Assistant & Systems Analyst	Assists the project team with planning, organizing, resolving disputes, requirements gathering, and other technical support needs. In addition, he determines the needs of clients for system development.
Vyom Shah	Analyst	Gathers information to develop the website that the client requested, and also customizes, optimizes, and analyzes the project's requirements.
Franklin Yang	Analyst & Communications Manager	Assists the client in resolving their problems. He also provides professional feedback on the proposed product, and manages the project's operations. In addition, he handles communication between the client and Software++.
Logan Raffkind	Unpaid Intern	Helps any team members that need an extra hand.

## 2.2 Work Breakdown Structure

A list of high-level tasks that will be carried out during this project are as follows:

- An agreement between The Client and The Analysts regarding the scope of The System will be created and signed by both parties.
- Concrete functional and nonfunctional requirements will be determined.
- A set of valid and reproducible use cases will be created.
- The interdependencies of The System will be outlined using domain models.
- A final set of UI mockups will be created using The Client's branding guidelines.

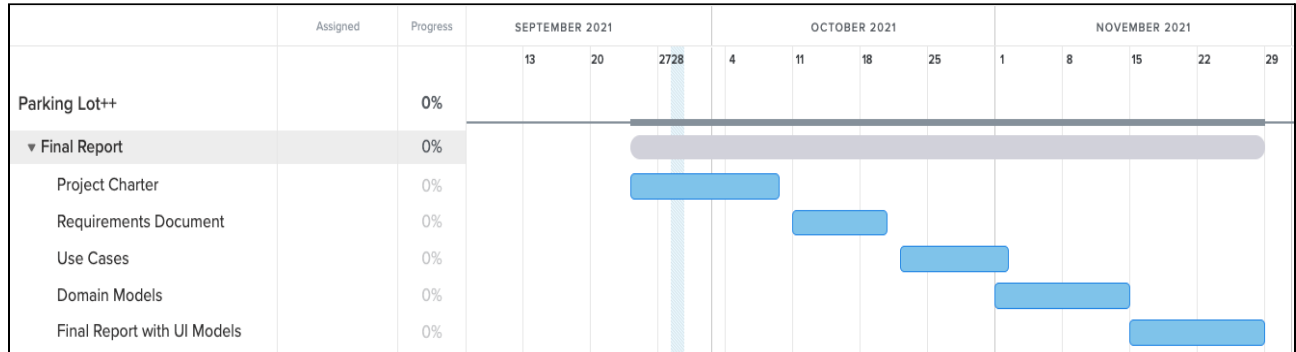
## 2.3 Milestones

The milestones for this project are as follows:

<b>Milestone 1:</b> Project Charter	October 7, 2021
<b>Milestone 2:</b> Requirements Report	October 19, 2021
<b>Milestone 3:</b> Requirements Report Review	October 21, 2021
<b>Milestone 4:</b> Use Cases	October 26, 2021
<b>Milestone 5:</b> Client Meeting 2	October 28, 2021
<b>Milestone 6:</b> Requirements and Use Cases Report	October 31, 2021
<b>Milestone 7:</b> Domain Models	November 2, 2021
<b>Milestone 8:</b> Requirements, Use Cases, Domain Models Report	November 2, 2021
<b>Milestone 9:</b> Client Meeting 3	November 15, 2021
<b>Milestone 10:</b> UI Models	November 23, 2021
<b>Milestone 11:</b> Final Report	November 28, 2021

## 2.4 Project Schedule

The deliverables for the requirements gathering phase of this project are listed in the gantt chart below. A more comprehensive chart will be appended to this document once development effort and timelines have been finalized.



## 2.5 Deliverables

There are several documents and models that are required by the client before the end of the requirements gathering phase. These include the following:

- 1. Project Charter** *October 7th, 2021*  
The document that outlines the scope, resources allocated, general timeline, and high-level objectives for the Parking Lot++ application. This will be a living document that helps ensure that analyst and client organizations have a shared understanding of what function the Parking Lot++ application will serve.
- 2. Requirements Document** *October 19th, 2021*  
This document will outline the functional and nonfunctional requirements that the Parking Lot++ application will need to meet in order to be considered an MVP. These requirements will also aid in the design of use cases, domain models, and UI models, as well as establish structure for the application testing phase.
- 3. Use Cases** *October 31st, 2021*  
The *Requirements Document* will be extended to contain use cases that outline how end users of the Parking Lot++ application will interact with the system. These use cases will also aid in the application testing phase, and will describe basic flows for subsequent UI models.
- 4. Domain Models** *November 14th, 2021*  
The *Requirements Document* will again be extended to include domain models, which define a structural and knowledge-based organization of application



requirements. It will define key concepts and relationships among the entities of the application.

## 5. Final Report with UI Models

November 28th, 2021

This *Final Report* will be the final extension of the *Requirements Document*, and will include a finalized set of UI models that show all relevant application interfaces. These will be fully colorized and branded to match the client organization.

## 2.6 Risks

A list of risks categorized as *High / Medium / Low* probability and *High / Medium / Low* impact are listed in the table below. This list also includes ways to mitigate each risk.

No.	Risk Description	Probability (H/M/L)	Effect (H/M/L)	Planned Mitigation
1	Too many people use the website at the same time to check for available parking spots, causing a denial of service.	M	H	The System will run on a cloud server managed by a reputable company to reduce downtime, as well as implement resource scaling.
2	A hardware sensor fails to correctly count vehicles, resulting in invalid data input.	H	L	The API that aggregates data from the hardware sensors (maintained by The Client) can alert an administrator if the number of available spots exceeds the max capacity, or falls below 0.
3	End users do not understand how to use the website.	M	M	Create a beginner's guide or walkthrough tutorial.
4	A data breach or leak results in user data being exposed.	L	L	A minimal amount of user identifiable information (user email and password only) will be stored directly by The Software. All passwords will be hashed and encrypted when stored, mitigating the impact of a password leak.

## 3.0 Project Approval

### 3.1 Stakeholders Sign-off

By signing in the specified locations in *Section 3.1*, the client organization confirms that they have read through the entirety of the Project Charter and have agreed to the terms, scope, and timeline outlined in the aforementioned document.

*Skylar Buck (Senior Systems Analyst)*

---

Signature

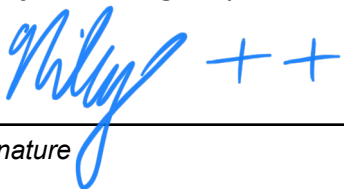
---

Date

### 3.2 Team Members Sign-off

By signing in the specified locations in *Section 3.2*, the analyst organization confirms that they have read through the entirety of the Project Charter and will adhere to the terms, scope, and timeline outlined in the aforementioned document.

*Nilay Sondagar (Director++ of Technical Compliance)*

  
Signature

*Oct. 7, 2021*  
Date