Software Requirements Specification

for

Express Parking

Version 1.0 approved

Prepared by
Allen Kaplan - 215494925
Akin Adewale - 215238231
Matthew Walker - 214779300
Chun-Wah Chung - 214901011
Toan Truong - 213943246
Daniel Kozlovsky - 214874879

BasedCode [Team 15]

14NOV2020

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Akin Adewale	10NOV2020	Added Section 2.1 - 2.2; Use Case diagram creation	v0.1
Allen Kaplan	10NOV2020	Added sections 2.4-2.7	v0.2
Matthew Walker	11NOV2020	Added sections 5-6; Filled glossary sections	v0.3
Toan Truong	13NOV2020	Added sections 1.1-1.5, revised wording in 1.3.	v0.4
Daniel Kozlovsky	13NOV2020	Added section 4	v0.5
Chun-Wah Chung	13NOV2020	Added section 3	v0.6
TEAM 15	14NOV2020	Full Document Review	v1.0

1. Introduction

1.1 Purpose

The primary goal of this document is to provide a complete list of requirements needed for an online platform to process payment for select spaces where single parking meters are currently used. With over 17,500 parking spaces across Toronto and 7,500 of them being single parking meters, Toronto Parking Authority (TPA) will be utilizing this platform to replace the single parking meters.

1.2 Document Conventions

The requirements provided in this document are not presented in a particular order. Each requirement has their own description, priority, and justification listed. Any dependencies between requirements are stated as part of the description. Priorities can be broken down into three categories:

- L= Low - Optional requirement
- M = Medium - Non-essential requirement but adds valuable features
- H = High - Essential requirement, must be implemented

1.3 Intended Audience and Reading Suggestions

The primary audience of this document includes, but is not limited to, software engineers, project managers, and engineer managers. A secondary audience may include parking enforcement officers (PEO) and system administrators (sysadmin) who seek out more information regarding the online platform's functionalities. It may also be of interest to individuals working on projects with similar requirements.

1.4 Product Scope

The online platform will be replacing the existing payment system for 7,500 single parking meters located across Toronto. Accounts used by the platform can be broken down into three categories: Users, Operators and Sysadmins. Users can select the parking space number and from their internet-enabled mobile device to initiate payment for time needed. They will also be notified of relevant events pertaining to the parking payments, parking duration, expiration and system information. Operators include the PEOs who will regulate the day to day operations. Sysadmins will be handling the administrative tasks. By moving the payment options from offline to online, it will minimize the number of locations that dispatched personnel will need to collect money from.

1.5 References

[1] L. S. Branch, "Consolidated federal laws of canada, Personal Information Protection and Electronic Documents Act," *Personal Information Protection and Electronic Documents Act*, 29-Oct-2020. [Online]. Available: https://laws-lois.justice.gc.ca/ENG/ACTS/P-8.6/index.html. [Accessed: 13-Nov-2020].

[2] "Standards," *W3C*. [Online]. Available: https://www.w3.org/standards/. [Accessed: 14-Nov-2020].

[3] "HTTP," *MDN Web Docs*. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/HTTP. [Accessed: 13-Nov-2020].

2. Overall Description

2.1 Product Perspective

Express parking is an online parking system for people in Toronto to process payment for single parking meter spaces. The system aims to minimize the number of locations where TPA representatives are required to collect payment while providing PEOs with the ability to check the status of payments on the system. Furthermore, Express Parking will allow the PEOs to dynamically modify the parking spaces (creating and deleting) and their parameters such as cost rate and hours of operation.

2.2 Product Functions

The System functionality is split into two subdivisions: user functionality and operators functionality

Express parking users are able to:

1. *View available parking spaces in supported lots*
2. *Pay for an available space for a specified vehicle and period of time*
3. *Make online payment for available spaces in parking lots where parking meters are currently in use.*
4. *Update reservation*
    a. *Add more time to an existing reservation*

Express parking operators are able to:

1. *Manage several parking locations*
2. *View available parking spaces within each location*
3. *View payment status of specific parking space*

2.3 User Classes and Characteristics

Standard Parking User

- May be a resident or visitor to the city.
- Should be able to locate parking space IDs on a map or by physical signs.
- Uses the platform to pay for parking spaces.
- Must be able to see the cost before paying.
- Should be alerted when their parking is expired.
- Pays for parking spaces by credit card.

System Operator - Parking Enforcement Officers

City of Toronto Employees (TPS, TPA)

- Requires accounts to manage parking spaces.
    - Must be able to see if a space is paid for to issue fines.
    - Must be able to create, delete or edit parking spaces within the system.
        - Edit → Time of payment, payment fee (per hour, flat fee)

System Administrator

- Highest level of platform privileges.
- Must be able to create and delete operator accounts.

2.4 Operating Environment

The user system should be accessible on a mobile device through the download of an application on the appstore of a web application. Further, the application should be able to function with poor internet connection (per 2.5, using asynchronous requests such as HTTP/1.x or MQTT).

2.5 Design and Implementation Constraints

- The platform must be able to work on desktop and mobile devices.
- The platform must conform to privacy and security requirements as specified in PIPEDA [1].
- The server must be able to handle high user throughput.
- The system should support language localization.
    - The system must support English (official language of Ontario).
- The server and user should be able to communicate using asynchronous requests for HTTP and Pub/Sub.

2.6 User Documentation

- Users should be able to have instructions located within the user application.
- Users should be able to request help through a customer support email or phone number.
    - Toronto Police and Standard Users should have different support endpoints.
- The platform should alert users to error messages as they occur.

2.7 Assumptions and Dependencies

- Users should be able to find the unique ID of a parking space by a physical sign or by searching a map on the platform.
- Users have a credit card to make payments.
- Only one person can pay for a parking space at a time.
- If a user pays for a space before they arrive and finds the space is already full, it is a user error to pay for a space before parking there.
- It is assumed that PEOs will not abuse the operator privileges.

3. External Interface Requirements

3.1 User Interfaces

3.1.1 All Users

All users will begin at a login screen

3.1.2 SysAdmin

This role will have a page for:

- Managing accounts
- Sending system notifications.

3.1.3 Operator

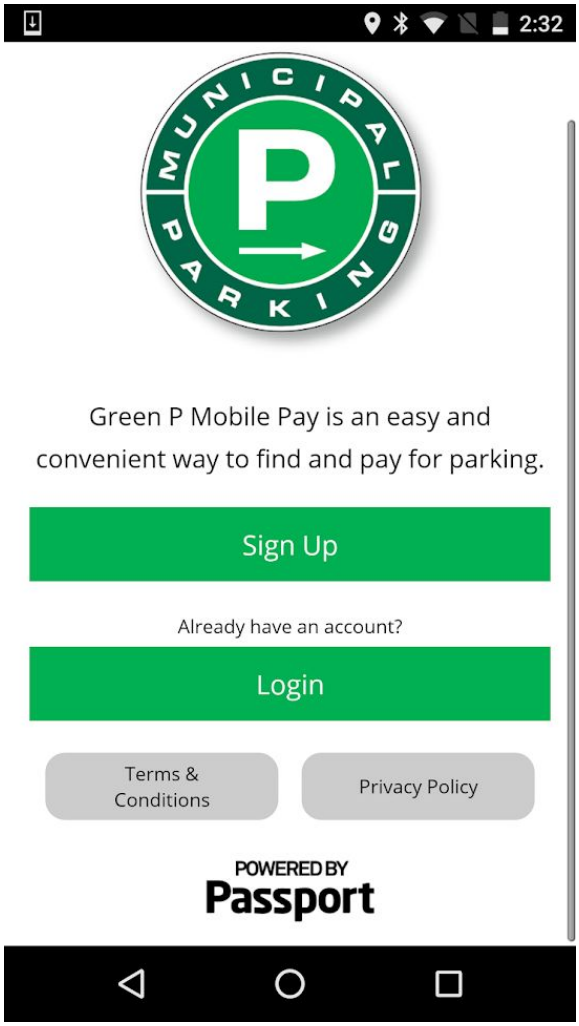
This role will have pages for:

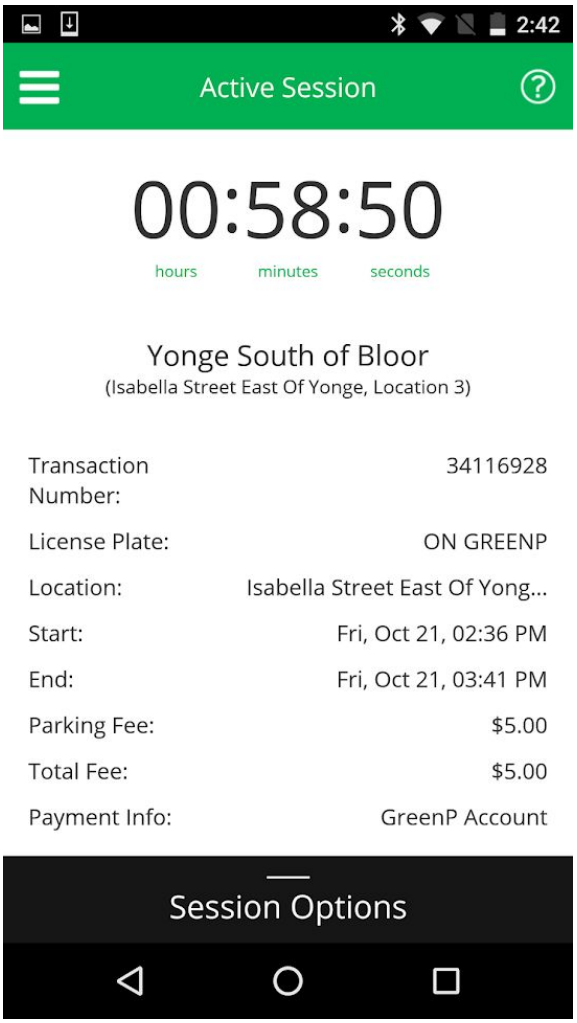
- creating, deleting and updating parking spaces.
- Viewing parking space status.

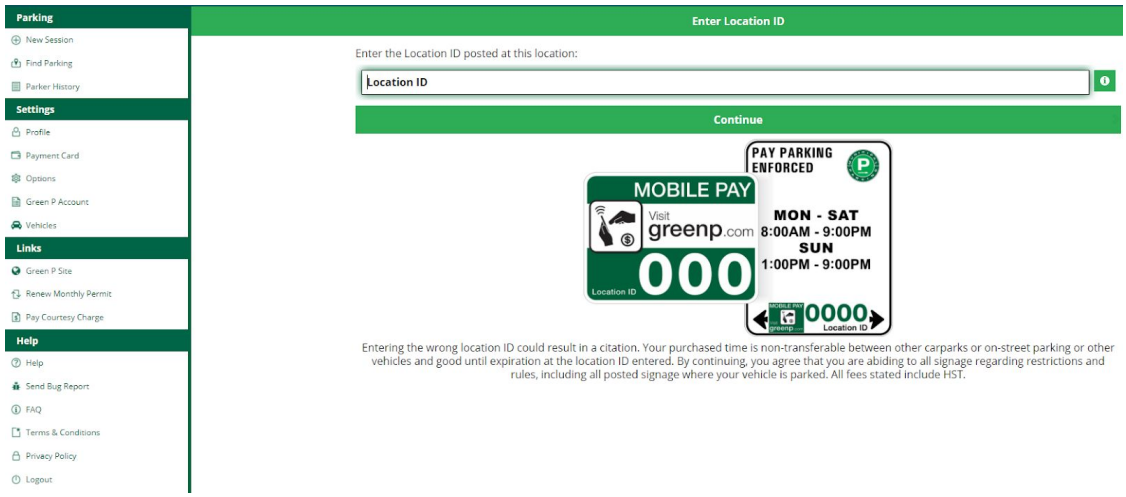
3.1.4 User

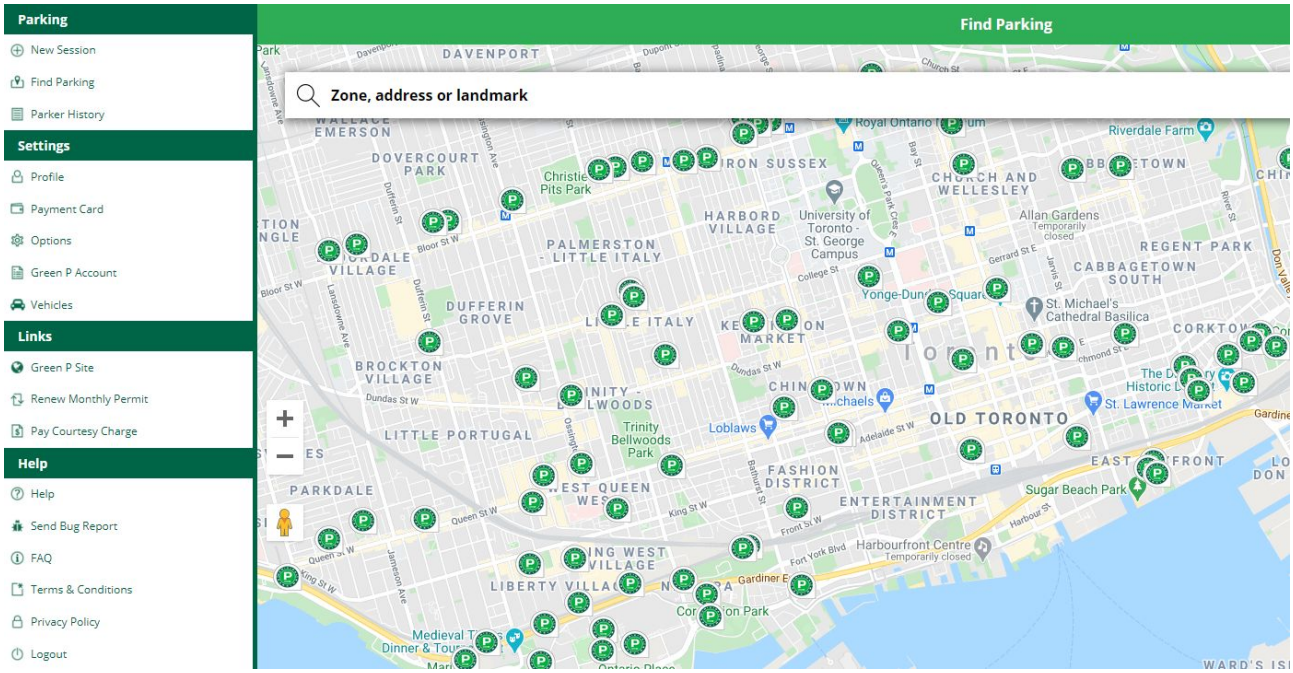
This role will have a pages for:

- Viewing parking spaces on a map with space IDs
- Pay for a parking space.
- View active bookings.
- Update account details.









3.2 Hardware Interfaces

Servers
Runs the frontend and backend of the application.

Database
Persistence layer to store relevant information.

Client Device
Devices that will access the system (e.g. mobile, desktop).

3.3 Software Interfaces

3.3.1 Frontend

Runnable in browser, conforming to w3 standards [2].

3.3.2 Backend

The backend logic of the platform should handle client requests from the frontend and make necessary changes to the databases.

3.4 Communications Interfaces

3.4.1 HTTP

RESTful services will use HTTP/1.x [3].

3.4.2 Publish/Subscription

Users and operators must be able to receive messages from sysadmins through a Publish/Subscription interface.

4. System Features

This section outlines all the functional requirements for the system, organized by feature. Each feature has it's relevant requirements listed along with their priority. There is no particular order to the features or their requirements. Justifications for their necessity in the system are given following each requirement.

4.1 User Paying For Parking

4.1.1 Description and Priority

This feature encompasses the user-side interaction with the system, including the main functionality. This feature is high priority.

4.1.2 Stimulus/Response Sequences

- User selects a parking space from a map
- User selects a parking time interval for the space
- The system prompts the user for payment information, and vehicle identifier, then the user provides it
- The system processes the payment

4.1.3 Functional Requirements

4.1.3-R1 The user is able to find a parking space by a unique parking space identifier.
Priority: H

Parkings spaces should be identifiable and non-repeating to avoid confusion.

4.1.3-R2 The user must have access to the following information about any desired parking space in the system, which may have been imposed by the system administrator
Priority: H

Information provided to the user is due to the self-serve model. User's must make an informed decision of where to park. It is still their responsibility to park according to regulations.

4.1.3-R2.1 The type of space
Priority: H

This allows the user to reference to their vehicle

4.1.3-R2.2 Parking time limits and exclusion periods
Priority: H

User can reference their parking time

4.1.3-R2.3 Parking price rate and limits
Priority: H

User can reference their budget

4.1.3-R2.4 Space number, address or any unique identifier
Priority: H

User knows where the space is and whether or not that is where they parked.

4.1.3-R3 The user is able to pay for a parking space
Priority: H

The city wants to collect payment

4.1.3-R3.1 The user is able to pay for any single meter parking space
Priority: H

A user can technically park anywhere and all parking spaces that are free are available. It is up to the user to follow parking regulations.

4.1.3-R3.2 The user is only allowed to pay for a certain amount of parking spaces at one time. This limit is imposed by the system administrator.
Priority: M

This is to prevent potential system exploitation and monopolization.

4.1.3-R3.3 The user is able to select a time period for which they plan to pay for the parking space.
Priority: H

A user can decide how long they want to park so they can save money. This allows the user to only pay for the time they need.

4.1.3-R3.4 The user must provide the licence plate for which the parking space payment is for.
Priority: H

A PEO must be able to see if the specific vehicle parked has paid for the parking.

4.1.3-R3.5 The user must use a valid credit or debit card to provide payment.
Priority: M

Payment cards should not cause issues in the payment system due to invalid payment information. Credit and debit cards must be from a globally or Canadian approved vendor.

4.1.3-R3.6 The user will only need to enter the necessary payment information in order to process the payment.
Priority: M

The user needs to provide payment info in order for it to be processed.

4.1.3-R3.7 The system will confirm the final payment amount with the user before processing the payment.
Priority: M

To avoid unwelcome surprises to the customer.

4.1.3-R3.7.1 The system will only charge the user the amount displayed during confirmation.
Priority: H

To avoid customer dissatisfaction and problems for the city.

4.1.3-R3.8 The system will notify the user when payment has been processed and approved.
Priority: M

The user should be aware if they are alright to leave their vehicle in the space, without getting a fine.

4.1.3-R3.9 Should payment approval fail, the system will allow the user to retry payment with the same or different payment information of their choosing.
Priority: H

The user has the right to park on city property and should not be constrained by payment errors.

4.1.3-R4 Once a parking space has been paid for, the system must save the link between the unique vehicle identifier, the parking space identifier and the duration of the booking.
Priority: H

The PEO must be able to access the link information for a time after the payment.

4.1.3-R5 The user will receive notifications through the system if their booking is soon to expire.
Priority: M

The user should have the opportunity to extend their booking if their ticket is soon to expire.

4.1.3-R6 The user will receive a notification through the system when their booking expires.
Priority: M

The user will need to be aware that their parking space booking has expired.

4.1.3-R7 The user will be able to read notifications through the system interface.
Priority: L

The user must be able to read notifications in order to valid their purpose.

4.1.3-R8 Upon receiving a ticket expiry notification, the user has the option to extend the time period, provided they pay for it.
Priority: L

This gives the user a chance to avoid a fine if they are approaching the parking limit and are far from their car, or plan to park longer.

4.2 Operator View Parking Space

4.2.1 Description and Priority

Feature to show parking spaces and related status and information. This is a high priority feature.

4.2.2 Stimulus/Response Sequences

- A PEO is checking parking spaces
- The PEO find the parking space from the map
- The PEO checks if the car in the spot has paid for parking
- They will issue fines based on information provided by the app

4.2.3 Functional Requirements

4.2.3-R1 A user or operator (PEO) can view a map of parking spots
Priority: H

A map is a quick and easy way of finding parking spaces.

4.2.3-R2 The operator is able to find a parking space by the unique parking space identifier.
Priority: H

This helps avoid confusion and for the PEO to find the space they are looking to check.

4.2.3-R3 The operator must have access to the following information about any desired parking space in the system, which may have been imposed by the operator.
Priority: H

This information is required for the PEO to make an informed decision about issuing fines.

4.2.3-R3.1 The type of space.
Priority: H

To check if a wrong vehicle type is parked in a space.

4.2.3-R3.2 Parking time limits and exclusion periods.
Priority: H

To check if a vehicle is parked outside the time limit.

4.2.3-R3.3 Parking price rate and limits.
Priority: M

Circumstantial.

4.2.3-R3.4 Space number, address or any unique identifier.
Priority: H

To avoid confusion and link a space to it's information.

4.2.3-R4 In addition, the operator must be able to see the following details about the space which reflect the status based on users interaction.
Priority: H

This is parking information pertinent to a user which informs the PEO's decisions about issuing fines.

4.2.3-R4.1 The operator should be able to see if a user has paid for the space and their linked licence plate.
Priority: H

The PEO must be able to see if a space has been paid for the specific vehicle.

4.2.3-R4.2 If the space was paid for, the operator should be able to see when the transaction took place and when the payment is set to expire.
Priority: M

In case the PEO wants to come back soon to check if the vehicle has gone, or to notice patterns or irregularities.

4.2.3-R5 The operator is able to see a full summary of both unpaid (available) and paid for parking spaces in the system.
Priority: M

For their reference and analytics to possibly influence operation decisions.

4.3 Manage Parking Space

4.3.1 Description and Priority

This feature encompasses the setup and management of all the parking spaces. This is to make the application more dynamic as there could be real-life changes to space due to construction, closure, etc. This feature is high priority

4.3.2 Stimulus/Response Sequences

- A PEO is patrolling and notices that a parking space is blocked by construction
- The PEO removes the parking space from the system to prevent people potentially parking there or paying for it

4.3.3 Functional Requirements

4.3.3-R1 The operator is able to add parking spaces to the system
Priority: H

In case new spaces are built or sanctioned by the city.

4.3.3-R2 The operator is able to remove parking spaces from the system
Priority: H

In case spaces are destroyed, blocked or rendered unusable by the city.

4.3.3-R3 The operator is able to update the following information on single parking spaces, individually or by bulk edit.
Priority: M

In case city regulations change.

4.3.3-R3.1 The type of space based on the following types
Priority: H

spaces may be resized or redesignated

4.3.3-R3.1.1 Standard parking
Priority: H

All vehicle allowed

4.3.3-R3.1.2 Motorcycle only
Priority: H

Smaller space

4.3.3-R3.1.3 Electric vehicle only
Priority: H

Regulatory

4.3.3-R3.1.4 Carpool only
Priority: H

Regulatory

4.3.3-R3.1.5 Accessible parking only
Priority: H

Regulatory

4.3.3-R3.2 Parking time periods.
Priority: H

Some spots may only require payment for certain times of the day.

4.3.3-R3.3 Parking price type, price rate and maximum rate per day.
Priority: H

Parking spaces may be for hourly or daily with a specified rate and maximum rate a user can be charged per day.

4.3.3-R3.4 Space number, address or any unique identifier
Priority: H

To fix mistakes with duplicates or for circumstantial operational purposes.

4.4 Sysadmin Manage Operators

4.4.1 Description and Priority

Administrative functions to tweak global system settings, manage operator accounts, manage user accounts and send global notifications. This feature is high priority.

4.4.2 Stimulus/Response Sequences

- The sysadmin interface will provide a variety of functions.
- The sysadmin can select to create, update or delete operator accounts.
- The sysadmin provides the payload for the account and it is creating, updating or deleting

4.4.3 Functional Requirements

4.4.3-R1 A master sysadmin account is able to create and delete other sysadmin accounts other than its own account
Priority: M

This is the top-level account and should be able to create other accounts.

4.4.3-R2 The sysadmin is able to create operator accounts
Priority: H

Sysadmins set up the system for use.

4.4.3-R2.1 When creating a operator account, the username and password is mandatory to proceed
Priority: H

An account must have credentials.

4.4.3-R3 The sysadmin is able to delete operator accounts
Priority: M

Sysadmins maintain the system for use.

4.4.3-R4 The sysadmin is able to update the information of a operator accounts
Priority: M

Sysadmins maintain the system for use.

4.4.3-R4.1 Name associated with the account
Priority: M

Names of people change

4.4.3-R4.2 Account username
Priority: M

usernames of people will change as a result of the previous

4.4.3-R4.3 Account password
Priority: M

Passwords should be periodically changed, especially if they are compromised or forgotten.

4.5 Receive Notifications

4.5.1 Description and Priority

Notifications can be sent out to accounts for information. This is a low priority feature.

4.5.2 Stimulus/Response Sequences

- A sysadmin sends out a notification that the system will be down for maintenance soon
- Each account holder receives this notification
- After reading this notification, an account holder deletes it

4.5.3 Functional Requirements

4.5.3-R1 A user can open a notification to read at any time after it is received.
Priority: M

It is up to the user to decide when they have a chance to read a notification. This ensures their actions in the system are not interrupted.

4.5.3-R2 Users can delete notifications once they've been opened.
Priority: L

This allows a user to track notifications that have been read/unread.

4.6 User Create Account

4.6.1 Description and Priority

Users need to have an account to use the application and should be able to create it themselves. This is a high priority feature

4.6.2 Stimulus/Response Sequences

- A user invokes the sign in feature
- They input their email, desired username and password

- The system save the info and the user can now log on anytime

4.6.3 Functional Requirements

4.6.3-R1 The user is able to create a user account with an email and password.
Priority: H

User accounts track user and tie a payment to a vehicle

4.6.3-R1.1 The user must specify an email and password.
Priority: H

The user must have login credentials

4.6.3-R1.2 Users accounts must have a unique email.
Priority: H

Account duplicates will cause errors when paying for parking spaces.

4.7 User Update Account

4.7.1 Description and Priority

Users may need to update any account information such as email or password and can do so themselves. This is a medium priority feature.

4.7.2 Stimulus/Response Sequences

- A user invokes the “update info” feature
- They input their new email, or desired password
- The system save the info and the user can now log on with the new credentials anytime

4.8 Functional Requirements

4.7.3-R1 The user is able to update a user account
Priority: M

User should be able to modify their information

4.7.3-R1.1 The user specifies the updated email or password
Priority: M

Users should be able to change their details.

4.9 Account Sign In

4.9.1 Description and Priority

This is the primary screen for all account holders. By authenticating with credentials, a user’s permissions and role are determined by the type of account. This is a High priority feature.

4.9.2 Stimulus/Response Sequences

- An account holder wants to sign in (user, PEO or sysadmin)

- They input their username and password
- The system authenticates them and they now have access to functions within their role

4.9.3 Functional Requirements

4.8.3-R1 A user, operator or sysadmin can sign into the system provided they have a valid account.

Priority: H

Credentials must be valid and authorized for security.

4.8.3-R2 A user, operator or sysadmin cannot access any features beyond signing in, while they are not authenticated with the system.

Priority: H

Prevents unauthorized users from accessing the system.

4.8.3-R3 A user, operator or sysadmin can only attempt to a number of maximum logins before they are locked out

Priority: H

Prevents brute-force attacks and DDoS attacks.

4.8.3-R4 All login attempts should be logged.

Priority: H

Allows for system analytics and increased security.

4.8.4-R1 The system is installed an initial master sysadmin account that cannot be deleted with full permissions and a default username and password

Priority: M

In order to provision sysadmin accounts.

4.10 Sysadmin Send Notification

4.10.1 Description and Priority

Sysadmins must be able to send notifications to users and operators for situations such as user booking expiry, closures, downtime, or data breaches.

4.10.2 Stimulus/Response Sequence

- An account holder wants to sign in (user, PEO or sysadmin)
- They input their username and password
- The system authenticates them and they now have access to functions within their role

4.10.3 Functional Requirements

4.9.3-R1 A sysadmin can send notifications to any combination of users or operators.

Priority: L

Sysadmins are aware of maintenance schedules, data breaches or other relevant information and are thus the ones to send the notifications. They can choose which accounts receive a notification, from a single user or operator, a single role type such as users or operators, or all user accounts

5. Nonfunctional Requirements

5.1 Performance Requirements

The following requirements specify the performance of the system under various circumstances. Justifications for their necessity in the system are given following each requirement.

5.1-R1 The total system planned downtime should not exceed 5 minutes.
Priority: M

This requirement ensures that there is not a high volume of users who need to pay for parking spaces who are unable to do so when the system is down. The time should be chosen such that it would be during a low-traffic period.

5.1-R2 The recovery time after unplanned system failure should not exceed 5 minutes.
Priority: M

The system should be able to recover if there is unexpected failure.

5.1-R3 After a user interacts with the application, the display should be updated within 1 second.
Priority: M

This is to ensure that the application is easy to use and satisfying. It prevents users from becoming frustrated with the application and potentially not paying for their parking space.

5.1-R4 The payment validation process should not take more than 15 seconds.
Priority: M

Some payment services will fail to process should validation take longer than a predetermined period of time. Therefore, it is important that in order to reduce the quantity of failed payments the process should not take longer than specified.

5.1-R5 All servers must handle high request throughput of 100 requests per second.
Priority: L

As the system will be managing at least 7,500 parking spaces, all servers must be able to handle high request throughput to ensure that the performance of Express Parking does not degrade.

5.1-R6 When loading a new area, the map should not take longer than 10 seconds to display parking spaces
Priority: M

The system should minimize barriers for users to find parking spaces and their associated IDs.

5.2 Security and Safety Requirements

The following requirements specify the security and the safety of the operation of the system. In order to provide a secure platform for the system, the following requirements must be met.

5.3-R1 All data passing through the system must be encrypted using a standard encryption algorithm (MD5, SHA, etc.)
Priority: L

Express Parking must keep users payment information secure. To achieve this, all data which passes through the application will be encrypted using standard encryption algorithms

5.3-R2 The collection and transfer of personal data must conform to PIPEDA data regulations.
Priority: L

Through conforming to PIPEDA, the safe collection and transfer of personal data will be ensured

5.3-R3 Session tokens must be implemented to protect against XSS and CSRF.
Priority: L

Session tokens will be used to ensure there are no cross-site scripting attacks (XSS) or cross site request forgeries (CSRF)

5.3-R4 The connections of the server must be secured using TLS with an SSL certificate.
Priority: L

Using transport layer security (TLS) to secure the connections of servers with secure sockets layer (SSL) will provide additional protection against data breaches and connection hijacking

5.3-R5 All users of the system must login using a form of unique identification.
Priority: L

All users must be uniquely identified using a username and associated password combination that can authenticate and authorize access to the system through providing a session token

5.3-R6 Each user role (user, operator, sysadmin) will have a common, secure entry point for the system.
Priority: L

Necessary to provide users with a point to enter their credentials and access the system

5.3-R7 All login attempts must be done in a secure forum
Priority: L

To prevent user credentials from being stolen, the forum in which they enter their credentials needs to be secure.

5.3-R8 All login attempts must be monitored and logged.
Priority: M

To prevent brute-force attacks, all attempts to access the system will be monitored and logged

5.3-R9 Protection against malicious login attempts must be implemented
Priority: M

A "maximum attempt" value will be implemented which if reached will require a "lost-password" recovery account system in order to continue the attempted log-in for the user

5.3 Software Quality Attributes

The following requirements specify the quality attributes of the system. Specifically, it outlines metrics for availability, flexibility, interoperability, maintainability, portability, reliability, and usability of the system.

5.3.1 Availability

5.3.1-R1 The Express Parking system should be available 99.5% of the time.
Priority: M

As people will need to pay for parking at all times of the day, the system needs to be available consistently

5.3.1-R2 The online platform should accommodate all modern (minimum 5 year old) mobile devices.
Priority: L

As most users will be using a mobile device within 5 years of age, for any device older the successful operation of the system will not be guaranteed

5.3.2 Flexibility

5.3.2-R1 Changes in the concurrent throughput of the system should not cause system failure or a significant degradation of performance.
Priority: M

To ensure that the system can be properly scaled to accommodate more parking spaces, the system should maintain a certain degree of performance

5.3.3 Interoperability

5.3.3-R1 The system should use a single form of data format and communication protocol (XML, JSON, etc.) for all relevant data transactions.
Priority: M

To ensure there are no conflicts regarding data and that data transactions are lossless, the system should use a single data format.

5.3.4 Maintainability

5.3.4-R1 A standard glossary must be agreed upon and adhered to for the classes, variables, and packages of the implemented system.
Priority: S

To ensure there are no conflicts regarding the implementation of the system, the implementation should use a standard glossary for classes, variables and packages.

5.3.5 Portability

5.3.5-R1 The system must be compatible with most mobile device hardware
Priority: M

As most users will be using a mobile device within 5 years of age, for any device older the successful operation of the system will not be guaranteed with respect to device hardware

5.3.5-R2 The system must be compatible with the Android OS and Apple's iOS
Priority: L

As most users will be using a mobile device which operates on Android OS or Apple's iOS, the successful operation of the system will not be guaranteed with respect to the operating system.

5.3.6 Reliability

5.3.6-R1 System errors should be handled so as to not cause system failure leading to unplanned downtime.
Priority: M

In order to ensure constant uptime of the system, errors need to be handled efficiently and effectively

5.3.7 Usability

5.3.7-R1 The system will use Eastern Time as the default timezone setting.
Priority: L

As the system will be primarily developed for the Toronto region, Eastern Time will best represent the majority of users

5.3.7-R2 The system must conform to a User Interface (UI) design guideline for each role.
Priority: M

To ensure consistency between design and implementation a UI design guideline should be developed prior to the implementation of the UI

5.3.7-R3 Each UI must reflect the role and it's associated authority level of the user.
Priority: M

To ensure that each user is sure of their permissions, the UI must accurately reflect the role and level of authority of the currently authenticated user

5.3.7-R4 Documentation defining the use of the system must be made available for each role of the system.
Priority: L

Ensures that new users of the system have an easy way to understand the operation of the system for their assigned role

5.3.7-R5 If the user tries to exit the application before payment has been completed, the system must ask the user for confirmation.
Priority: L

Ensures that users do not accidentally believe they have paid for a space that they have not due to an unsuccessful payment validation.

5.3.7-R6 Default settings for parking duration will be automatically filled in for the user.
Priority: L

Promotes the ease of use of the system

5.3.7-R7 The user must be able to find parking spaces and their IDs using a map.
Priority: M

Promotes the easy of use of the system

5.4 Business Rules

The *Express Parking* application has three main business roles. These roles are defined as the primary actors for the system: the user, operators and sysadmin. Each of these roles will have access to a separate view which contains certain functions that are specific to each role. Specifically, users are able to pay for parking spaces, operators have access to manage parking locations, and sysadmins can add or remove operators and change global settings. The separation of these roles enables the system to have a common secure entry point (login) for each role and improves the efficiency of each subsystem.

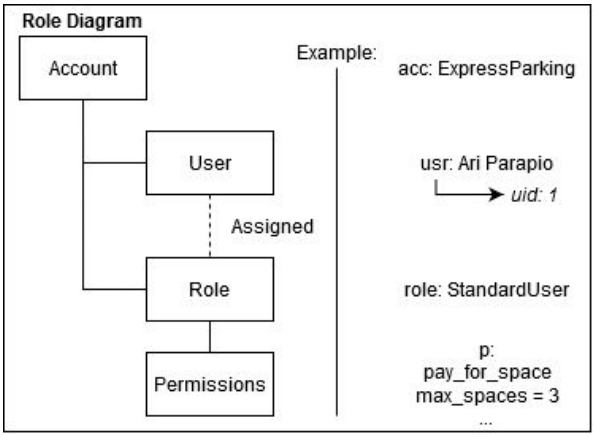


Figure: Example Role Diagram

5.4-R1 Each account of the system will be assigned a role
Priority: L

5.4-R2 Each account must be assigned strictly one role
Priority: M

5.4-R3 Possible roles to be assigned are {User, Operator, Sysadmin }
Priority: L

5.4-R4 Each role will have separate functions as per the requirements.
Priority: L

6. Other Requirements

6.1 Database Requirements

In order for the system to maintain users it needs to have access to a database where the users credentials and roles (authorization level) are stored. Additionally, there needs to be a record of each parking space and it's status.

6.1-R1 The database must be sharded to spread the load and increase the performance of the system
Priority: L

6.1-R2 The database should use a predetermined data dictionary to reduce the likelihood of conflicts
Priority: L

6.1-R3 The database must have daily backups.
Priority: M

6.1-R4 The system must be able to read and write to the database.
Priority: L

Appendix A: Glossary

Acronym	Description
CSRF	Cross-Site Request Forgery
MTBF	Mean time between failures
PEO	Parking Enforcement Officers
PIPEDA	Personal Information Protection and Electronic Documents Act
SSL	Secure Sockets Layer

TLS	Transport Layer Security
TPA	Toronto Police Authority
TPS	Toronto Police Services
UI	User Interface
XSS	Cross-site scripting

Term	**Description**
Admins	Upper management of TPS
Client	Song Wang
Enforcers	TPS enforcement agents making use of the application to manage spaces
Sysadmin	System Administrator
Users	Everyday individuals making use of the parking system to park their vehicles
Accounts	An account can login to the system with given roles