Introduction

There are a wide variety of requirement elicitation (RE) techniques that suit the requirements gathering process for the "CLup: Customer Lineup" project. This document will cover; the suitable requirement elicitation techniques to analyze the requirements, and the relevant modeling diagrams to show aspects of the system. Requirements engineering often has no "right way" when designing requirements for a software system, and such requirements may change over time. Requirement elicitation methods include; introspection, background reading, hard data, interviews, surveys, meetings, focus groups, brainstorm/JAD, prototypes, ethnomethodology, participant observation and knowledge elicitation.

Methodology

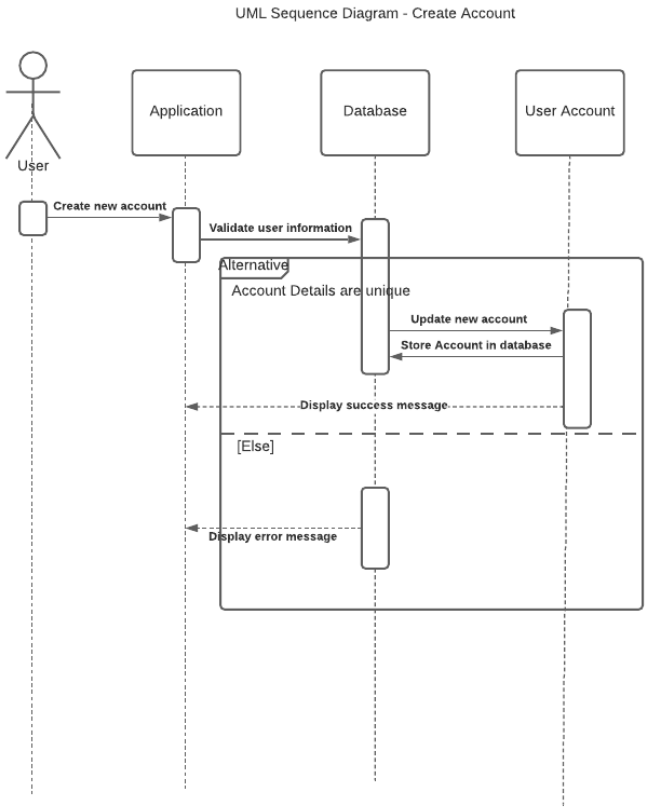
Given that an overview of the requirements have already been defined in the problem statement, a method/approach to further elicit the requirements is the use of a brainstorm. A brainstorm is a creative technique to rapidly visualise ideas of how to solve the problem. Given a problem statement one can come up with new ideas and characterise topics for discussion. For example - who are the potential stakeholders of the product? What is the 'problem' that the project is aiming to solve? Through the use of a brainstorm, key stakeholders can be defined being, the grocery store customers (the users), store employees, managers and executive management, alongside the government body. The problem at hand is that the covid-19 pandemic limits the number of customers in the store, leading to hour long queues outside the store. Brainstorms are generally used in a group setting, but can be done individually as well. Another RE technique is the research of related applications. This is a form of background reading where documentation of existing systems is analysed to see if there is any overlap and to ascertain what similar products exist in the market and what aspects could be incorporated into the current product. A highly relevant example is the appointment booking website 'HotDoc' - https://www.hotdoc.com.au/ which allows users to book appointments for general practitioners around Australia.

Reflection

One challenge encountered was eliciting the requirements for the 'fallback option', where physical tickets/QR codes are given out by hand. Upon review of my analysis, it seems that I have made an oversight and that the system does not fully support this 'offline' feature. For example, how will the person with the physical ticket be notified that there ticket has been called? A potential solution to this is to have a large electronic display outside the store, but this may conflict against the issue that the software is trying to solve. Further analyis is required.

Conclusion

To conclude



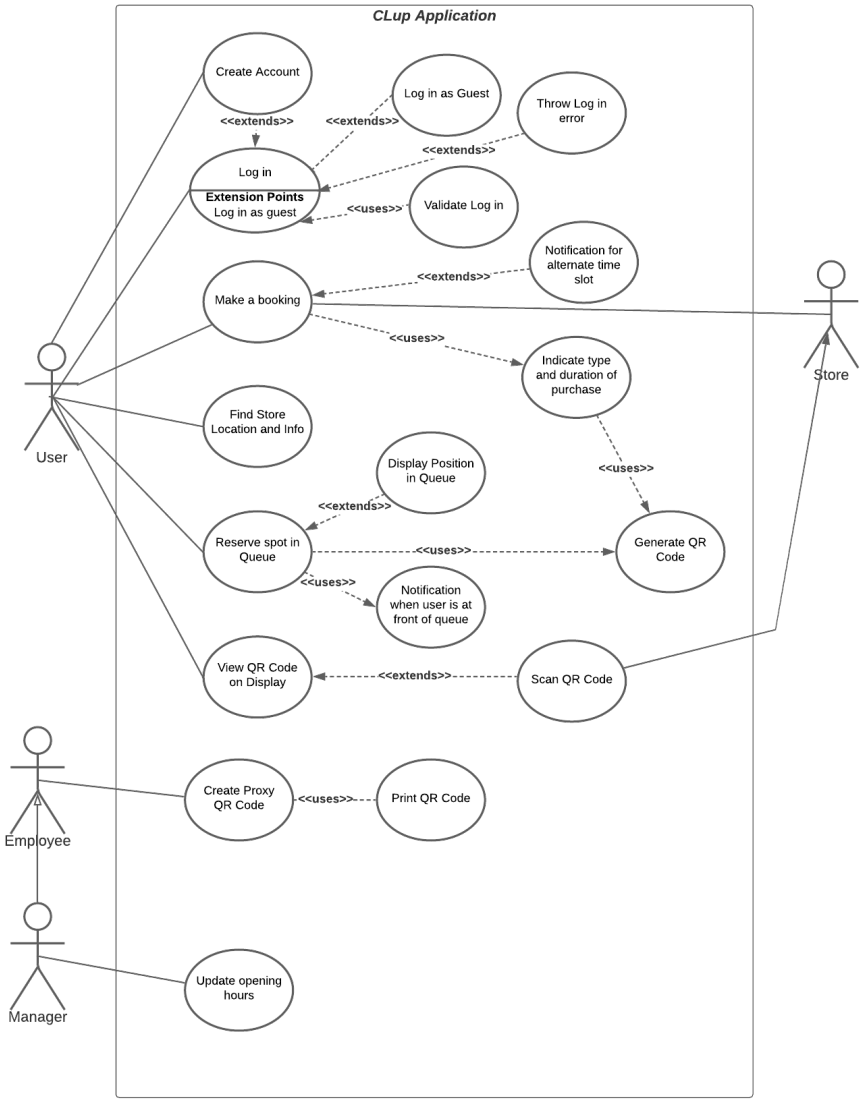
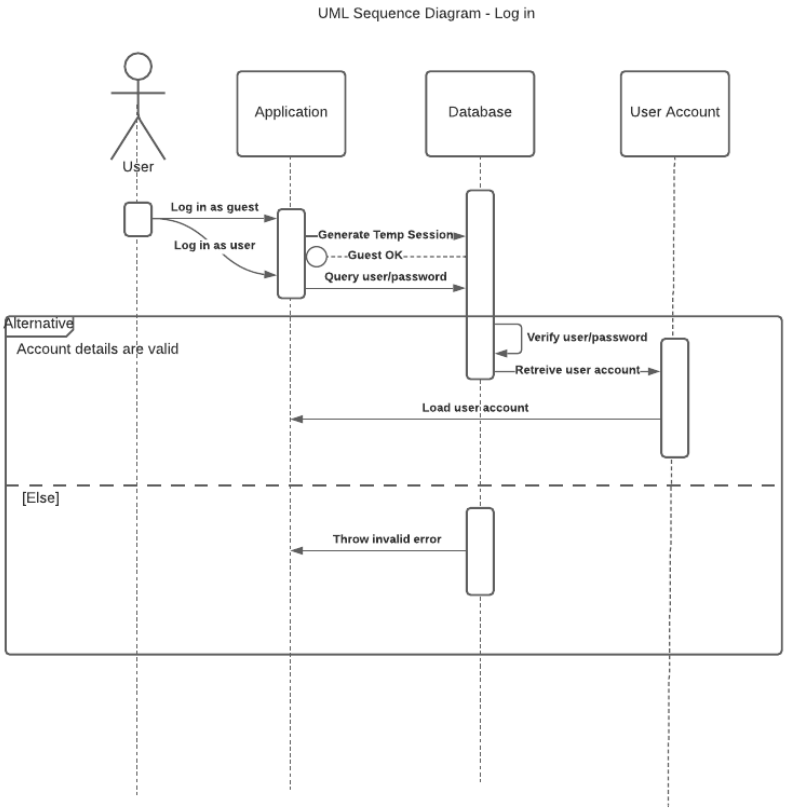
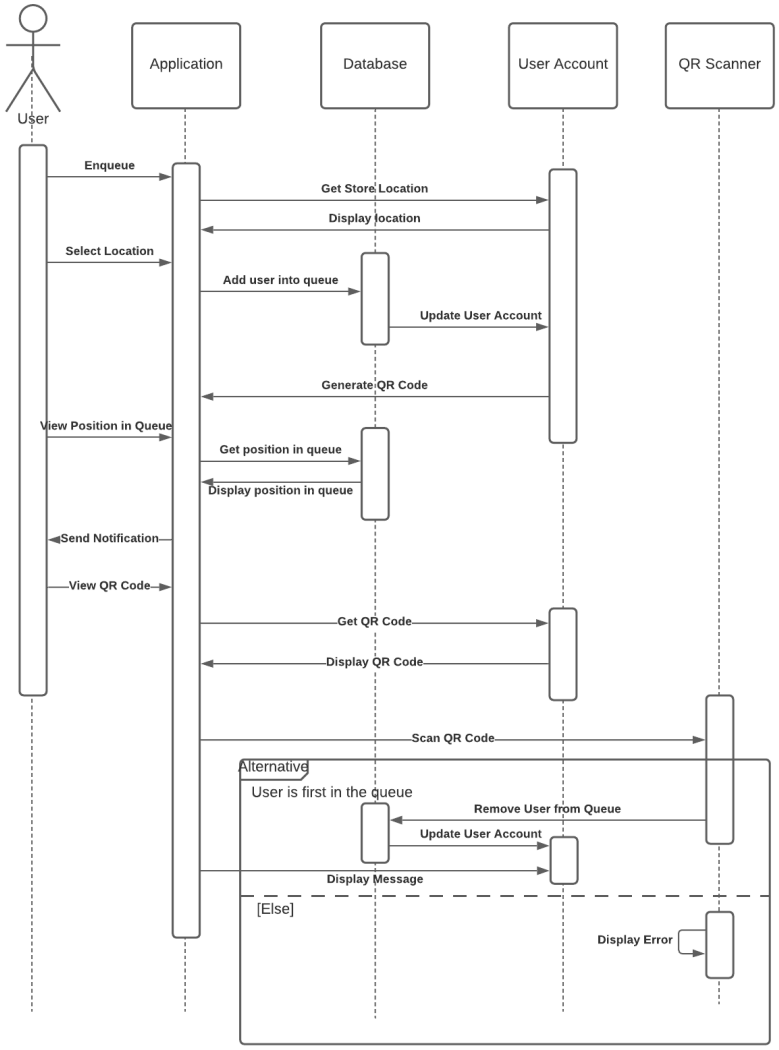


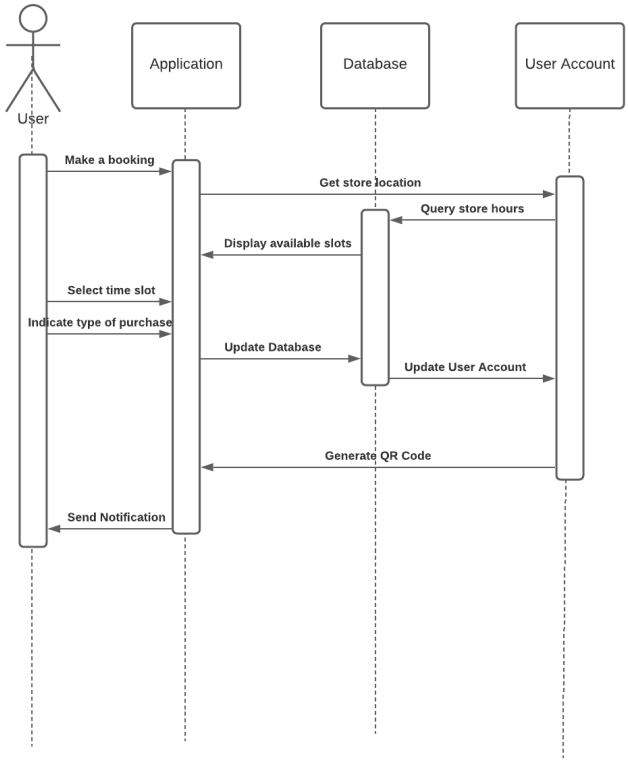
Figure 1: UML Use Case


UML Sequence Diagram - Log in
User
Application
Database
User Account
Log in as guest
Log in as user
Generate Temp Session
Guest OK
Query user/password
Alternative
Account details are valid
Verify user/password
Retreive user account
Load user account
[Else]
Throw invalid error

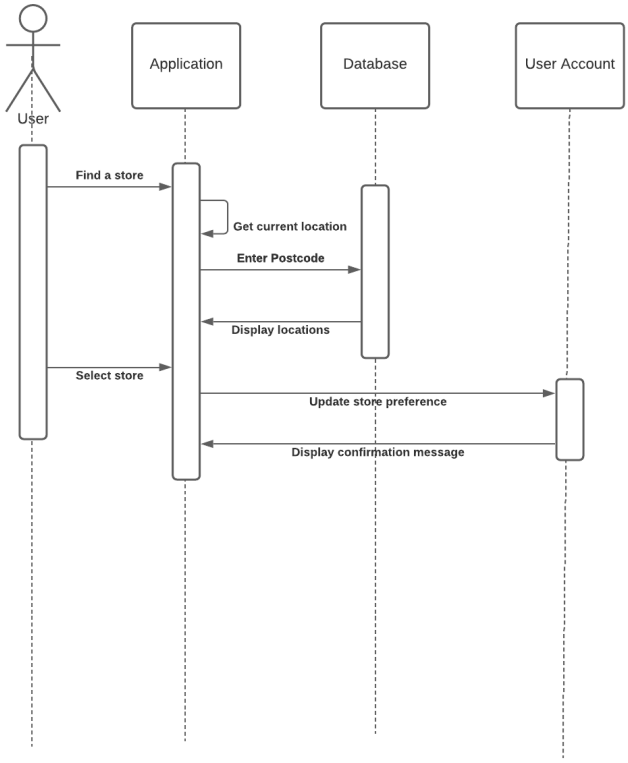
UML Sequence Diagram - Queue System



UML Sequence Diagram - Booking



UML Sequence Diagram - Store Locate



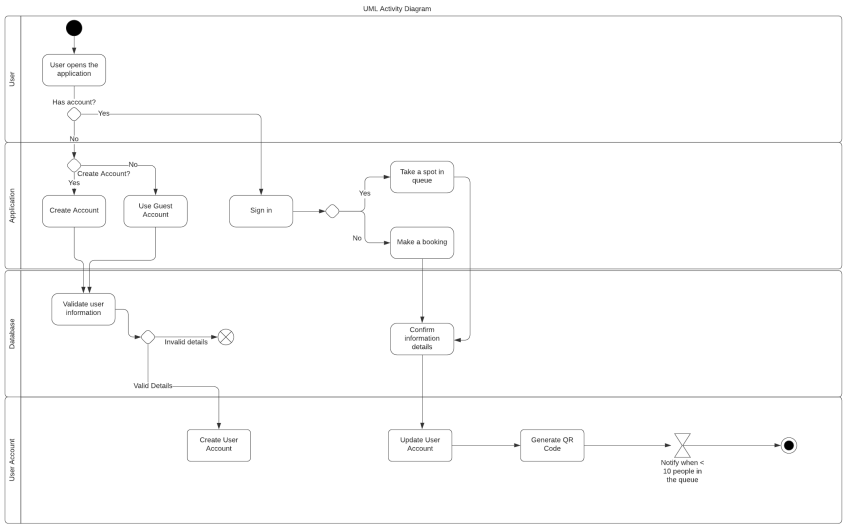


Figure 2: Activity Diagram