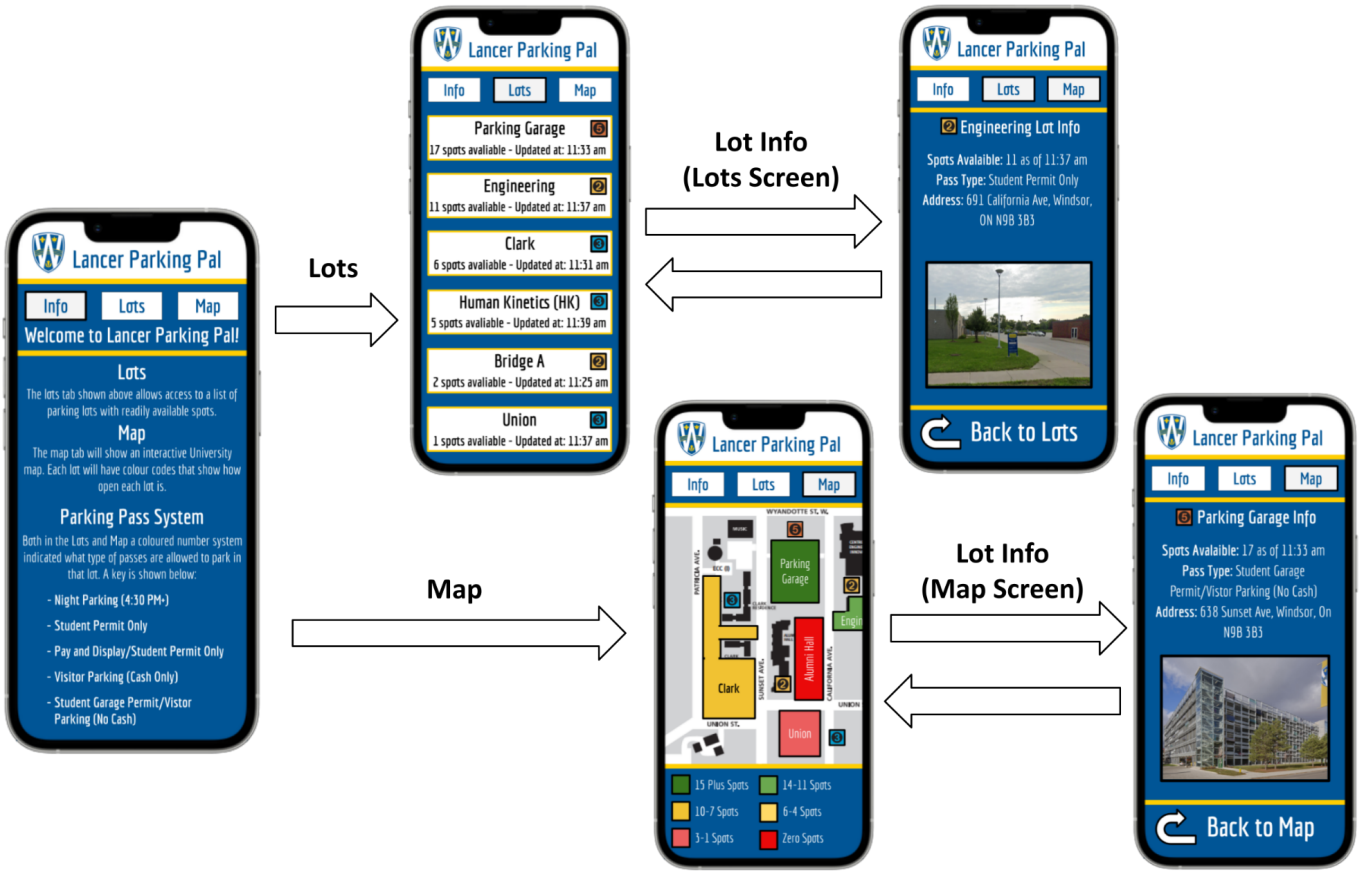
Phase #2 - COMP3110 - Due: November 13th, 2022
Luke Mullins, Lucas Sarweh, Drew St.Amour, and Daniel Chiarcos

Part A) Layout



Part B)

Requirements/Use Cases

- User can view a “help”/”info” page
- User can view a map of the locations of all parking lots
- User can view a list of the parking lots
- User can view the parking spot availability of each lot

Top 3 Use Cases

1. User can view a “help”/”info” page
   This is the starting page that the user will see when they open the app. This page explains what the “lot” and “map” options do along with how the parking pass system works. This page will be used the most for all users, whether it be to understand how our software works or just used as a reference when needed.

2. User can view a list of the parking lots
   This is a navigation tool to help the user see the parking lots existing in the area. If they are not familiar with the area it will be especially helpful to have all the parking lots listed in one place. If they click on a certain parking lot, they will be able to view more details such as where it is located.

3. User can view the parking spot availability of each lot
   This is the page that shows a more indepth look at each parking lot. When the user selects a lot, the name of the lot, number of available spots, pass type, and address of the lot are all displayed to the user. This page allows the user to view all possible information about the lot and can be helpful to new users who want to park in a different lot.

Part C)

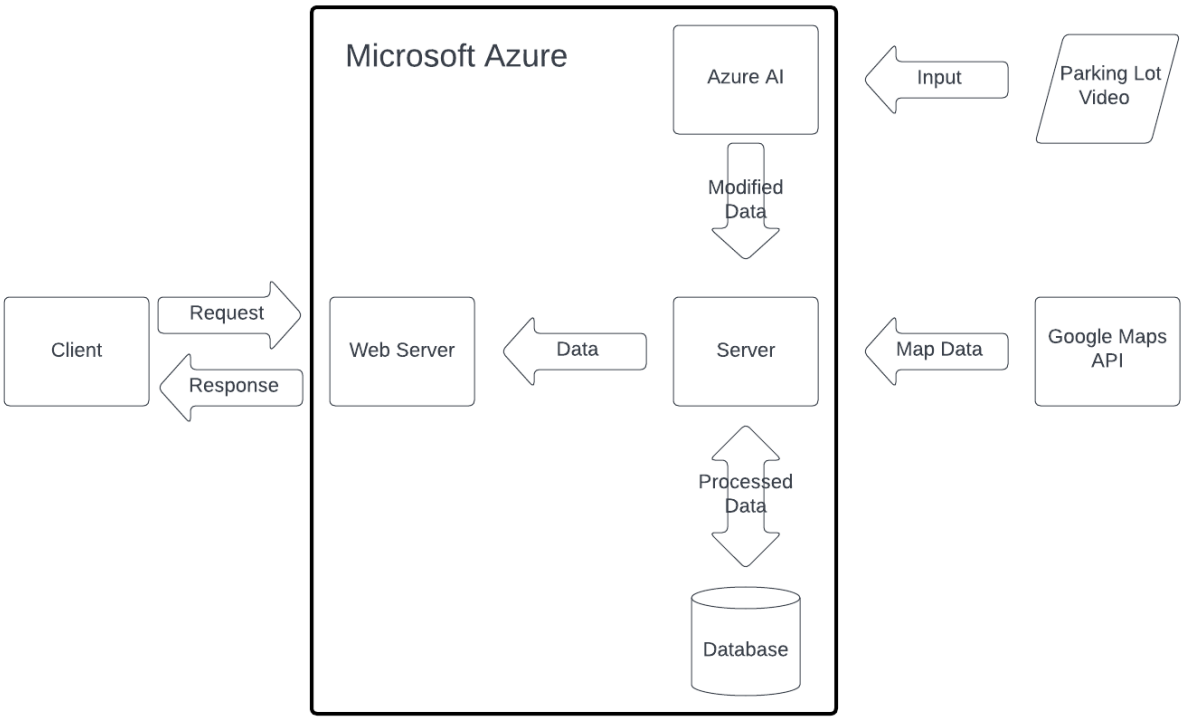
1. To allow users to view information regarding our app, we will create a help page to inform users how to navigate, utilise, and read different information generated from our app such as parking lot maps, parking passes needed, and spot availability. This page will be what the users see when they load into the app. We will make this the first page so the users can learn about the app before even starting to use it. Although information about the app is available on the front page, we will also implement small windows of information throughout the app to assist users wherever they may be. An example of this would be in the map page where we will provide a legend to help users read the map to check for spot availability.

2. As the app is intended to be used with multiple parking lots, it is important that we display the different lots available to the user. By creating a page for parking lots, we can implement a list of different parking lots that can display general information about each one, such as spot availability, name and location. Depending on the number of parking lots, the list will grow. If there are enough parking lots, the user can scroll through the list to look for different parking spots. By seeing the different parking lots available, the users can use this information to determine which parking lot is best for them.

3. There will be two ways to view spot availability in each parking lot. The first way to view availability is through the parking lot list. Clicking on a parking lot here will open an information page showing different data on the parking lot such as availability as well as the parking pass type and location. The second way to view availability is by locating the parking lot on the map. When a user clicks on the parking lot on the map, the information page from the previous option will show. This allows users to navigate a map to get a better understanding of the nearby area and having the ability to select the parking lot on the map means users don't have to remember the parking lot name, go back to the parking lot list, and find that parking lot there. The way this information will be gathered is through a video camera at the parking lots. This video will be fed to an AI who will

interpret the data by counting the occupied spaces. The map of the area will be fetched by google maps API.

Architecture:



We decided to go with a client-server architecture as we need the server to provide data to the user whenever the user requests it. With this architecture the server can gather the information from other sources like the Google Maps API and parking lot footage to then analyse and give back to the user. Information will also be stored in a database for long term statistics for things like projected spots available and times that tend to be busier during different seasons.

Part D)

One issue we had was trying to figure out how to correctly display all the information our app generates. One example was for the parking lot overlay in the map. We had to decide what the overlay represented. We thought of different ideas like having the overlay represent the type of parking pass needed for the spot as or what department the parking lot is intended for. After discussing the different possibilities as well as the implications each option creates, we decided the most useful information to display here is the spot availability. Since the other options were not as useful as spot availability, we decided to move this information to separate pages so the users can still see this data without crowding the map.

Another issue we came across was determining how to implement our mapping system. Originally, we had an idea to get a map of the local area that has all the parking lots on it but there were a couple of problems with this approach. First, we would need to update the map whenever something changes like roads closing or new buildings being built. Another issue was getting this static map to work with our idea of giving users directions to the parking lots. Not only would it be hard to implement something to determine a route from the user to the parking lot. To fix this, we decided to use the Google Maps API so users can move around the map and see more of the surrounding area. The API can also help determine a route for users to take to get to the parking lot. By using the API, not only will it help make our app work better but it will also help reduce the workload we have to get the app working as intended.