PLeaze

BCS430W: Senior Project (Writing Intensive)

07/28/21

By: Will Leuenroth, Adarsh Uday, Nick Bosenbeck, Nick Forgione

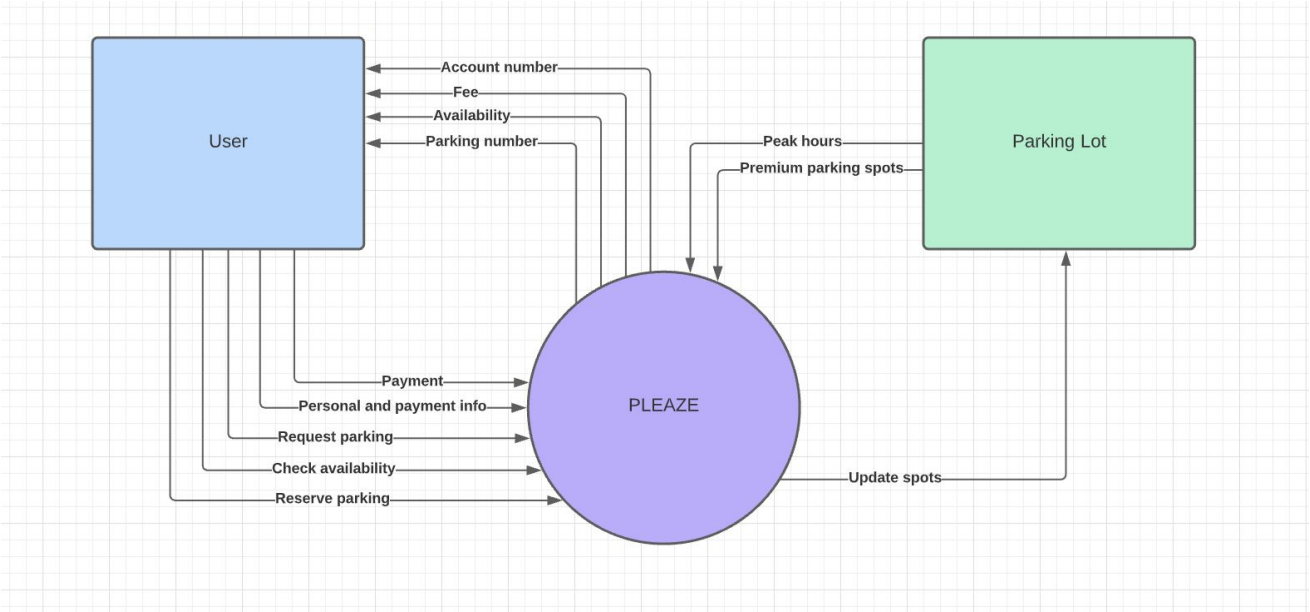
The Problem

Drivers often find themselves in situations where they are unable to reliably find a parking space. This is due to the fact that there is a high volume of people parking in a parking lot at one time, which can make it look like there are no spots available. We want to make sure people are able to find parking efficiently. To make parking more convenient, our parking app will direct the user to available spots by assigning them a parking number.

Product Story

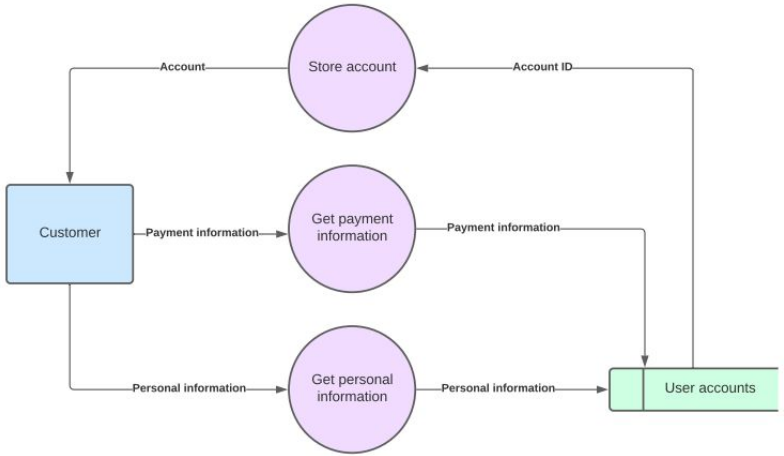
Many drivers suffer from the same stress of dealing with horrendous parking lots in which all spots are taken. This then leads to drivers aggressively piloting their vehicle across each parking aisle until they either have to give in and park in the next lot over or succumb to parking in the very back. This can stack on more issues like the potential of having to be drenched in the rain if you are leaving your destination at the worst possible time, or leg strains from the grueling walk across the lot. Many people seek answers to fix these issues, but to no avail, they have not found much of a solution. One answer that drivers can only dream of is a product that guarantees a parking spot ahead of leaving their homes, as they would not have to leave for work even earlier than they already do to ensure they do not arrive late. This product would theoretically provide enough utility to justify a paid fee in advance, but no less at the whim of the driver's utmost content.

Context Diagram



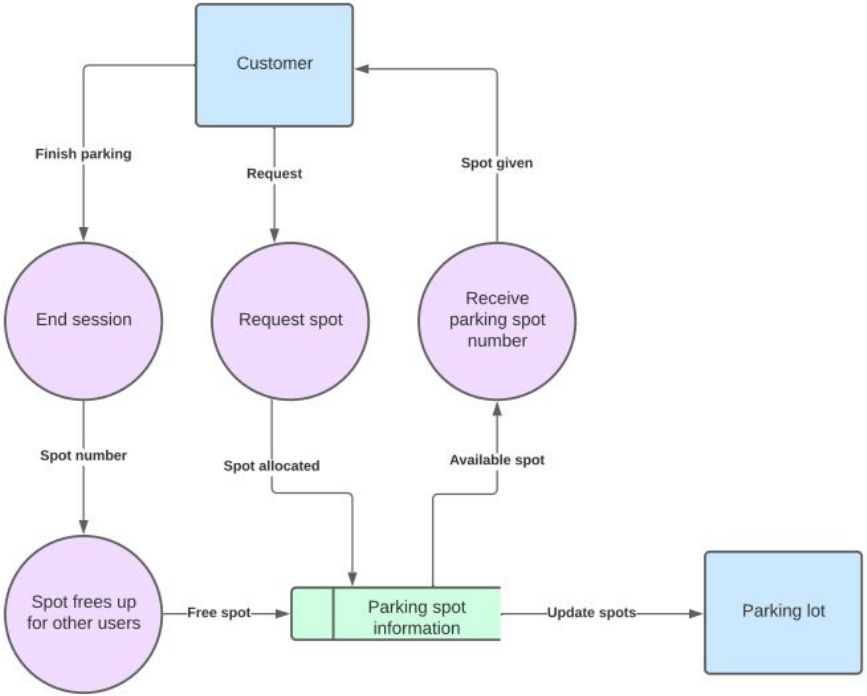
Users can input their personal and payment information to create an account and receive an account ID. They can request parking to receive a spot number to park in. Availability can be checked and is based on people parking and leaving. Users can reserve parking in advance. Our Parking lot will have designated premium parking spots that can charge a higher rate.

Create an Account DFD



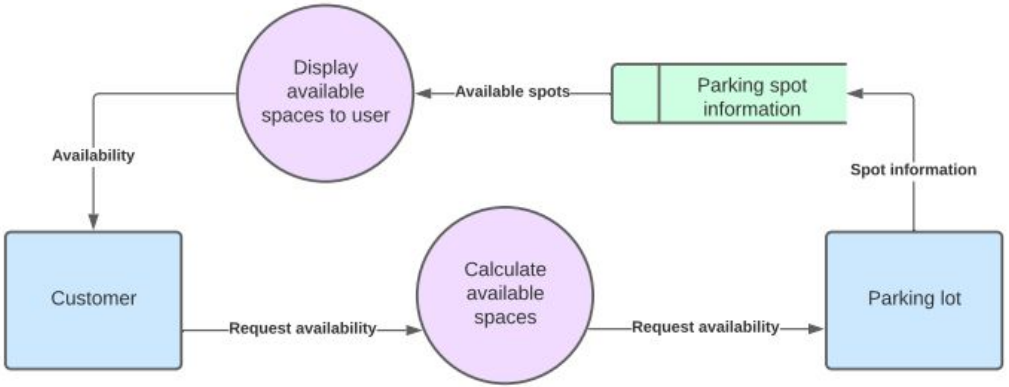
When creating an account, users input their personal and payment information. They are then given an account and unique account ID. Their account is added to our database.

Parking DFD



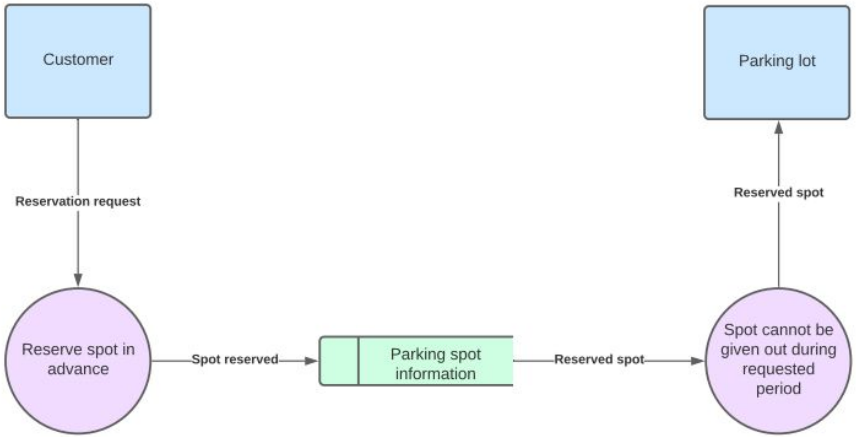
Customers can request a spot to park in and they are given an available spot. That spot is then taken up in our database. When they are done parking, the spot frees up for other users.

Availability DFD



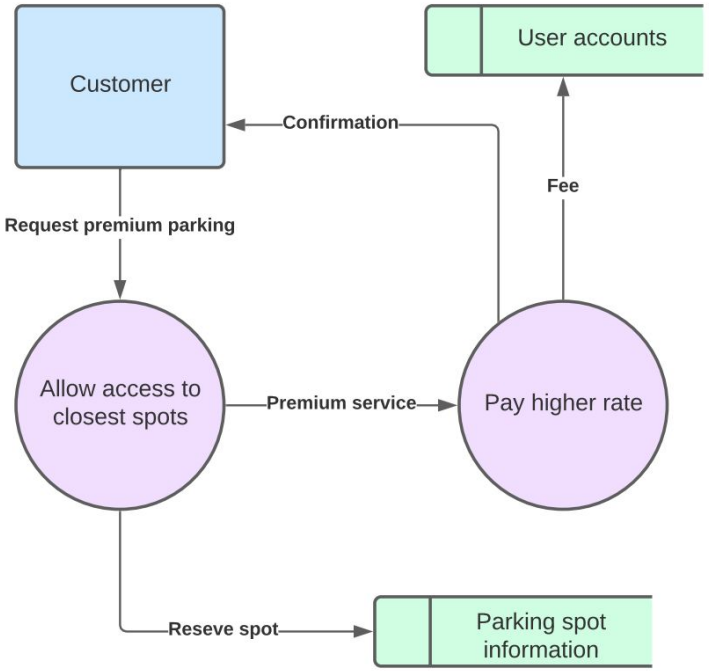
Users can request availability before they go to park. We get that information based on people parking and leaving

Reservation DFD



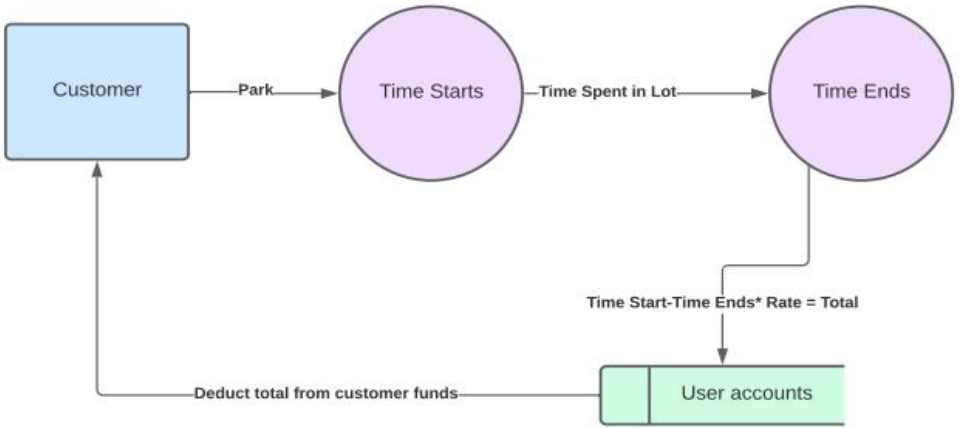
To reserve spots, users request a spot in advance, and they are given a spot that has not been given out to other people who reserve or regular users. When the user arrives, their spot will be available for that user and that user only.

Premium Parking DFD



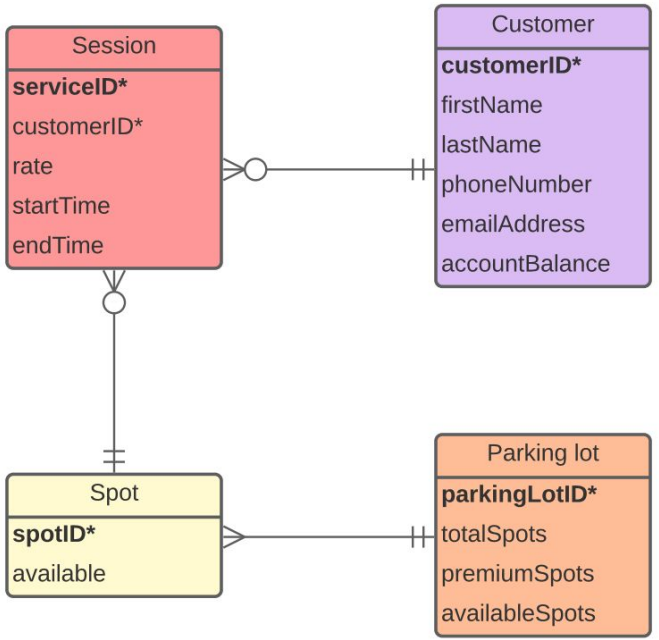
Some spots will have a premium option that are very close to their destination. Users can pay a slightly higher rate to access these spots.

Payment DFD



For our payment DFD, the user parks. Their timer begins. When they unpark, the time spent in the lot is multiplied by the current rate. That amount is than deducted from the users funds.

Entity Relationship Diagram



In our ERD our entities are car, customer, service, parking lot, and spot. IDs will all be int values. Names, phone number, address, and email address are all strings. Premium and available in the spot entity are boolean values, representing whether that specific spot is a premium parking spot and whether it is available to be given out or not.

Test Plans/User Stories

Project Team Name:	Wanna Be Done	**Tested Date:**	7/7/2021		
Testing Team Name:	Wanna Be Done				
	Given	**When**	**Then**	**Pass/Fail**	**Test Notes**
1	A user downloads the app	They open it for the first time	They create an account	Pass	
2	A user enters an incorrect email, password, etc.	They try to login or create an account	An error message will pop up to let them know they need to fix a credential	Pass	
3	A user has an account	They are logged out	They sign in	Pass	
4	A user is logged into their account	They first sign in	They are brought to the home screen	Pass	
5	A user is on the home screen	They need to add credits/run out of credits	They click the profile icon and click add credits; they then enter the amount they'd like to add	Pass	
6	A user has funds in the account	They are ready to park	They make sure they are on the home screen and they click our park button; Once clicked, a spot will be given to them	Pass	
7	A user wants to edit their profile	They need to change a certain credential	They click our edit profile button	Pass	
8	A user wants to reserve a spot	They need to get a spot ahead of time	They click our reservation button and input the necessary information	Fail	In Process
9	A user wants to check our availabilty	They want to make sure a spot is available	They click our availability button and check the spots available	Pass	
10	A user has a parking spot	They want to check out and end their time	They unpark their car and sign out from the spot on our application	Pass	
11	A user is logged into their account	They are ready to sign out	They click the sign out button that brings them back to the log in/create screen	Pass	
12	A user on the reservation page	They want to check the differing rates	They click the home button; and view the rates section *In Progress*	Fail	In Process
13	A user wants to park for 23 minutes	They don't want to spend on time they wont use	They can pay for the exact minute so they dont waste money or time	Pass	
14	A user checks the availabilty	They need to get somewhere with a time constraint	They can make sure enough spots are available; if it is cutting it close, the user can decide of reserving a spot or taking a chance in a spot being available	Pass	
15	A user has a reservation	They arrive and need to park	They access the spot given to them and their time begins(*In Progress*)	Fail	In Process

For our initial In-class testing, our application failed three out of the fifteen stories or plans we created. The numbers failed were number 8, number 12, and number 15. If you notice, all the three that were incomplete related to reservation. This is because during this time, we had not completed all the functions.

Final Testing Results

Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	
Pass	

However, if you take a look at our results compared to our test plans, you will see we have completed each of the tasks we set out to complete. If you take a look at our presentation, you will see our results in live time.

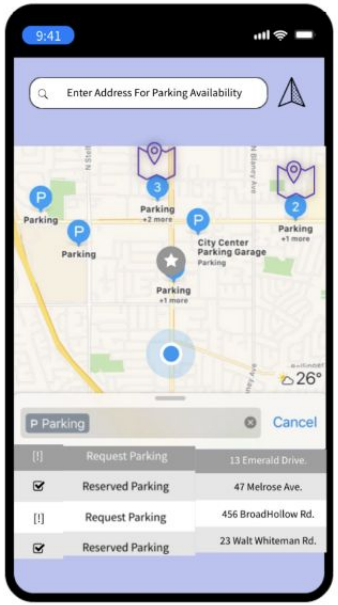
Stretch Stories

1. As a user of this application, I want to be able to access different parking lots that allow me to use this service.
2. As an electric vehicle driver, I want to receive discounts on certain spots, that charge my car as I pay for the time.

Fantasy Story

1. As a user of this application, I want to be able to view the complete history of my previous parking

Original User Interface



Our original user interfaces really set the scene for what we wanted to do. Our application looks much simpler than this, however, it performs the same way we wanted this mockup to. The first screen shows how we intended our user to login or sign up. The second screen shows our intention for the availability screen as well as the reservation screen. Our current model has those two features on differing screens, but they do go hand in hand. Truthfully, we think our group has come a long way and we could not be more excited and prouder of the work we accomplished.