I Dream of Parking

Architecture and Design Overview

Mina Anis, Vincent Cottone, William Lin, and Dan Mackey

Spring 2019

Table of Contents

1. Purpose

1.1 Statement of Goals

The purpose of this document is to provide an architectural overview of the “I Dream of Parking” application. The intended goal of the “I Dream of Parking” application is to remove the user’s hassle of trying to find a parking space in a parking lot by finding the best available parking space for them. This program’s intended audience are the people who commute on the daily, and as such are forced to deal with the constant turmoil of finding parking. Unfortunately it isn’t even as simple as just “finding” a spot anymore, as modern day parking lots also hold many other coin tosses, such as is the spot clean, did the people in neighboring spots park in such a way that allow others to safely park their cars, and even simply how far the spot is from their destination. All these things are much less prevalent in a garage that implements the “I Dream of Parking” application.

1.2 Functional Description

The “I Dream of Parking” program is a software application that will simplify the process of finding available parking in a parking lot. It’s main use will be to calculate the best remaining parking space in a parking lot through the use of an algorithm that prioritizes user convenience above all, placing incoming users into the most convenient open spot so not only do the users save time parking, but they save time walking as well. It does this by being taught where the

"exit" of the parking garage is and then assigns users spots based on the closest spot opening on the edge of the first rows, moving inwards in order to relieve potential traffic congestion. The system would be setup with two computer screens placed at the entrance of the parking lot, one on either side for the entrance and exit and both of these screens will be paired with their own

respective boom barrier. See below for example.

Image Source: https://www.justdial.com/photos/autoparker-india-teynampet-chennai-automatic-gate-dealers-479spzi-pc-24594000-sco-20tgxkdr

An example of a "run" of the program would be when a user pulls up to the entrance, they are greeted by the screen and see all the spots that are unoccupied in that respective parking lot. They would then have the option to request a parking spot from the program through a button integrated into the UI of the program. Once pressed, that button will tell the program to run through an algorithm based on a weighted system that prioritizes the closest spot to the entrance, and then it will tell the user which spot to go to by turning that specific spot yellow(with open spots being green and occupied spots being red) on the screen, as well as telling them in writing where to park. Shortly after assigning the user a spot the machine will print out a card with a magstripe. This card comes in later when the user wants to leave, by allowing them to swipe

their card thus freeing their spot in the system for future occupants, and to open the exit boom barrier.

2. Architectural Goals and Constraints

2.1 Architectural Goals

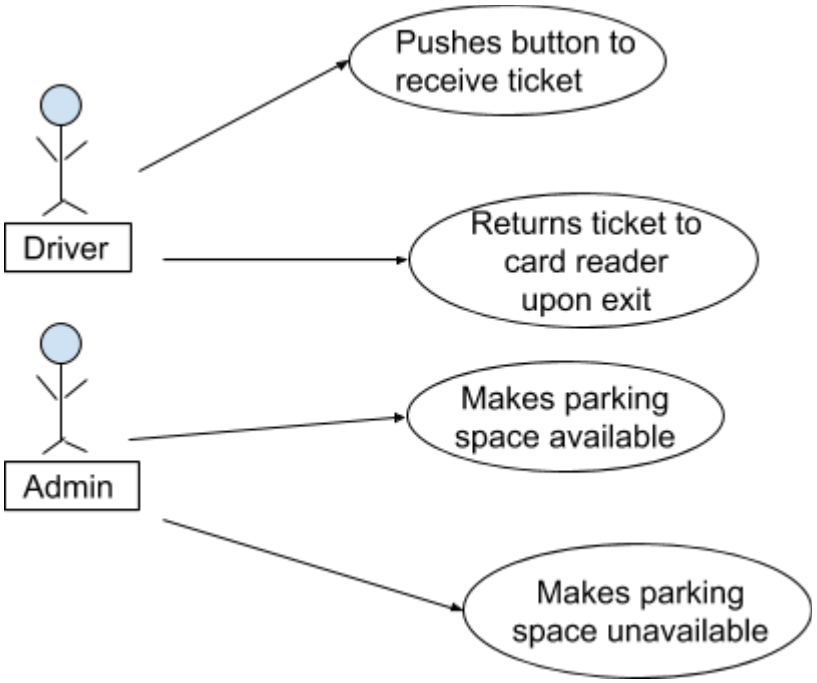
1. Organize the parking process so that users entering a parking lot will quickly and easily be assigned a parking space.
2. Allow for businesses that operate parking lots to avoid lost profits from users being unable to find a parking space.
3. Assure that new features and/or updates can be implemented as the system evolves.

2.2 Constraints

1. The system will provide the user with a simplistic interface.
2. Uptime, as well as response time will be well within the 99th percentile.
3. System will be easily maintainable.
4. External administrator will have privileges to edit spot settings.

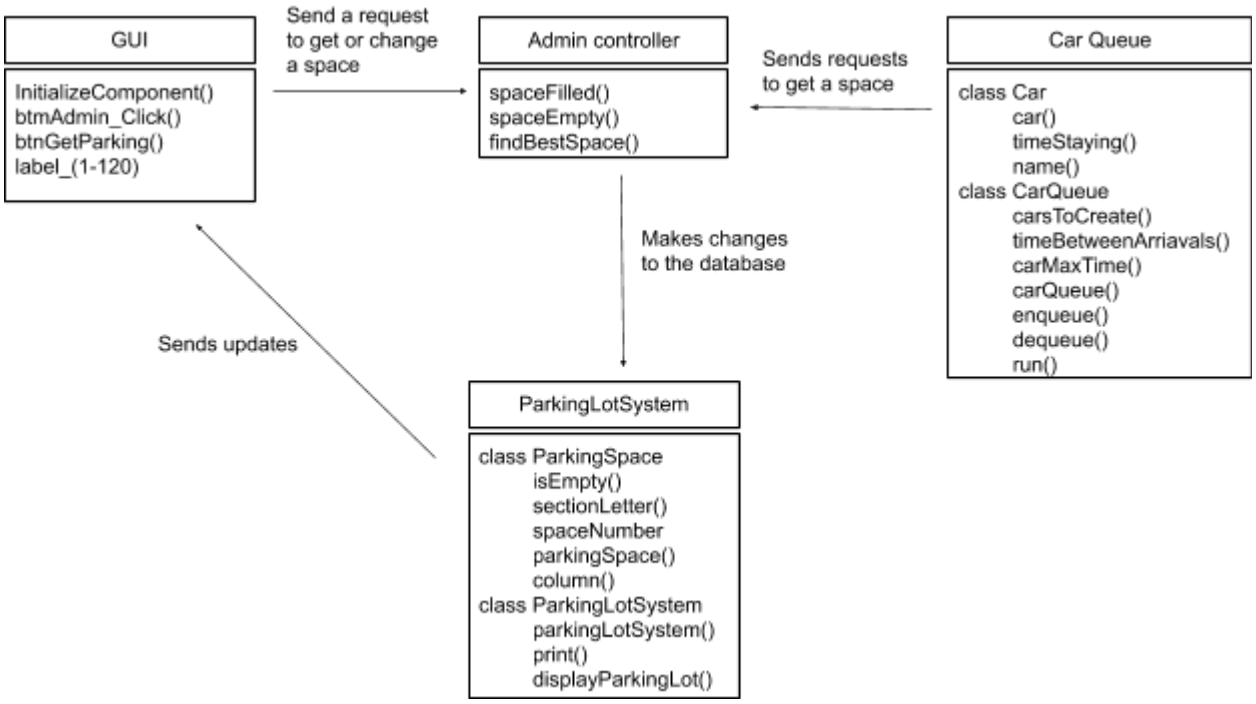
3. Architectural Representation

3.1 Use-Case View

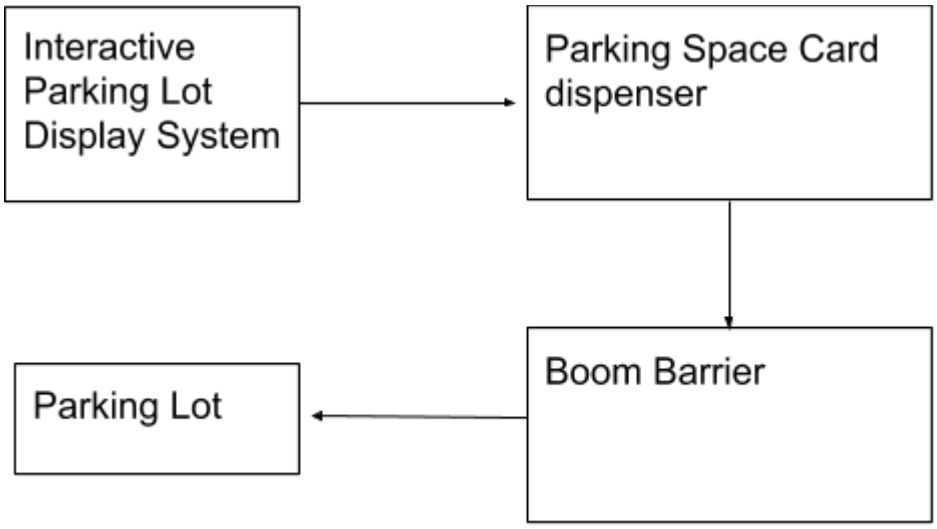


The user has two actions. The driver will drive up to the ticket system. They hit the button on the GUI to receive their assigned spot. Then park, and upon exit they insert their ticket into the end system. The end system updates the parking lot array to make the spot available. The admin has control over the system. They can edit the parking distribution to make a spot available if no car is present. They can also make a parking spot unavailable if there is a car present or obstructing the spot.

3.2 Logical View



3.3 Process View



3.4 Deployment View