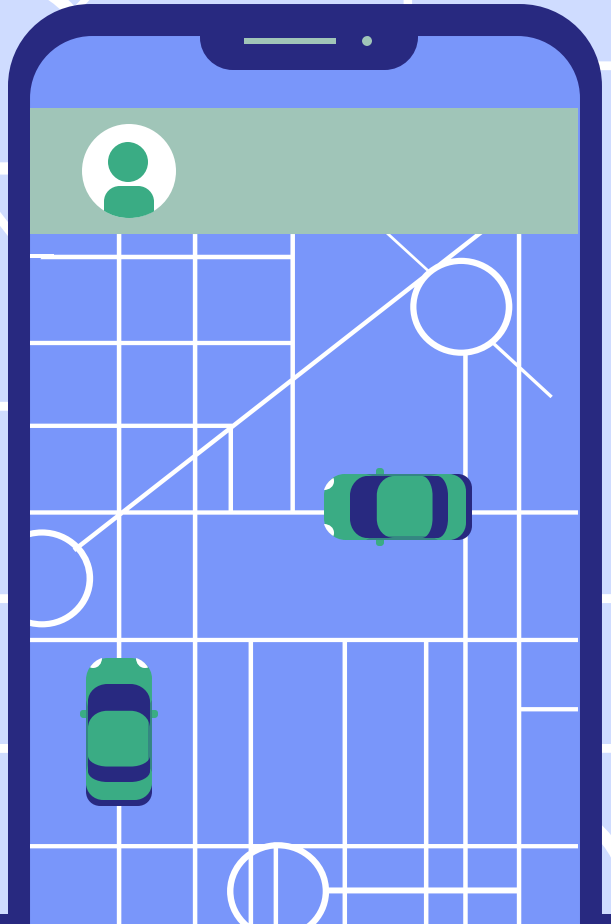
SmartPark

IoT-Driven Automatic Parking Solution

SD 4920: Senior Design II - Team sddec24-17



About Us

Team Members :
William Clemmons, SE
Zachary Sears, CPRE
Brian Witherspoon, EE
Kennedey Reiling, EE
Mubassir Serneabat Sudipto, CYBE
Ethan Haberer, EE

Client/Advisor : Md Maruf Ahamed


Parking can be problematic!

Project Plan

Problem Statement

Streamline parking experience

Create a detection-based system to monitor parking spots for availability and valid payments

Develop an app for students, teachers, etc. to view and reserve available parking

Eliminate issues such as staff-only parking, full lots, and time-consuming searches

User Needs

- Availability
- Quick Parking
- Easy Payment
- Parking Validation

Requirements

Functional

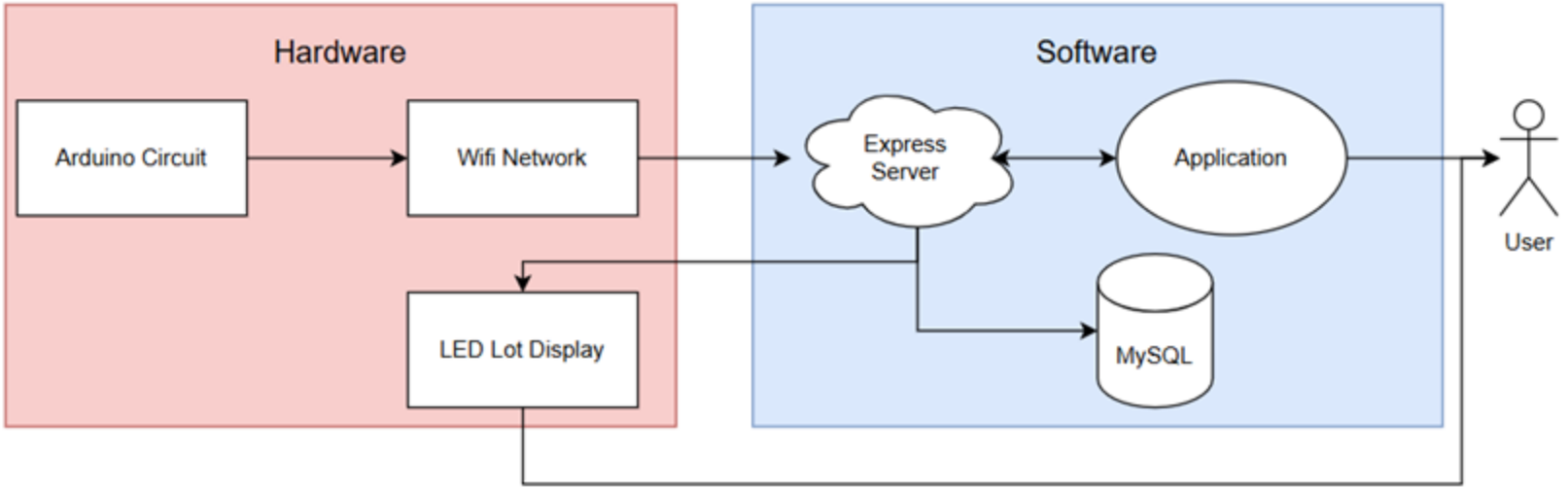
- Hardware
  - Sensors update in real-time
  - A way to communicate to the customers without the application
- Application
  - Users can reserve spots
  - User is directed to their parking space
  - Payment feature

Non-Functional

- Hardware
  - Low-maintenance
- Application
  - Secure payments
  - Availability
  - Low-latency

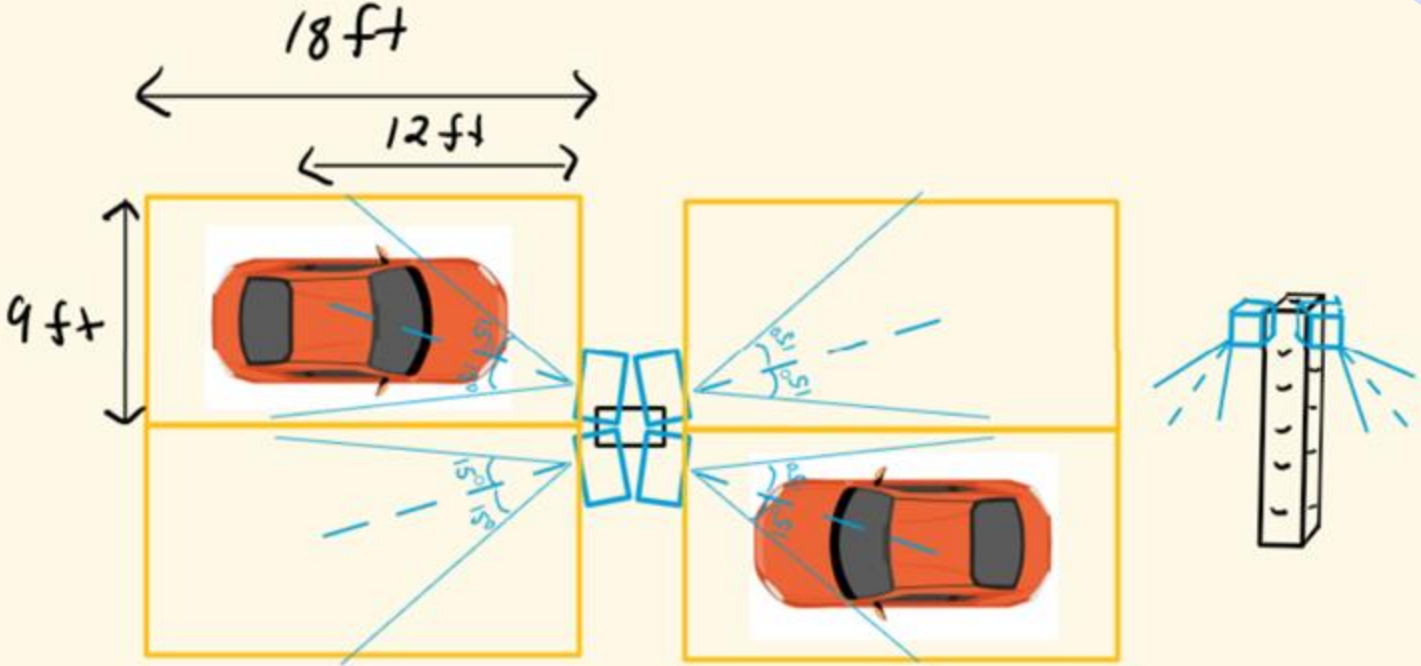
Our Solution

Overall Design

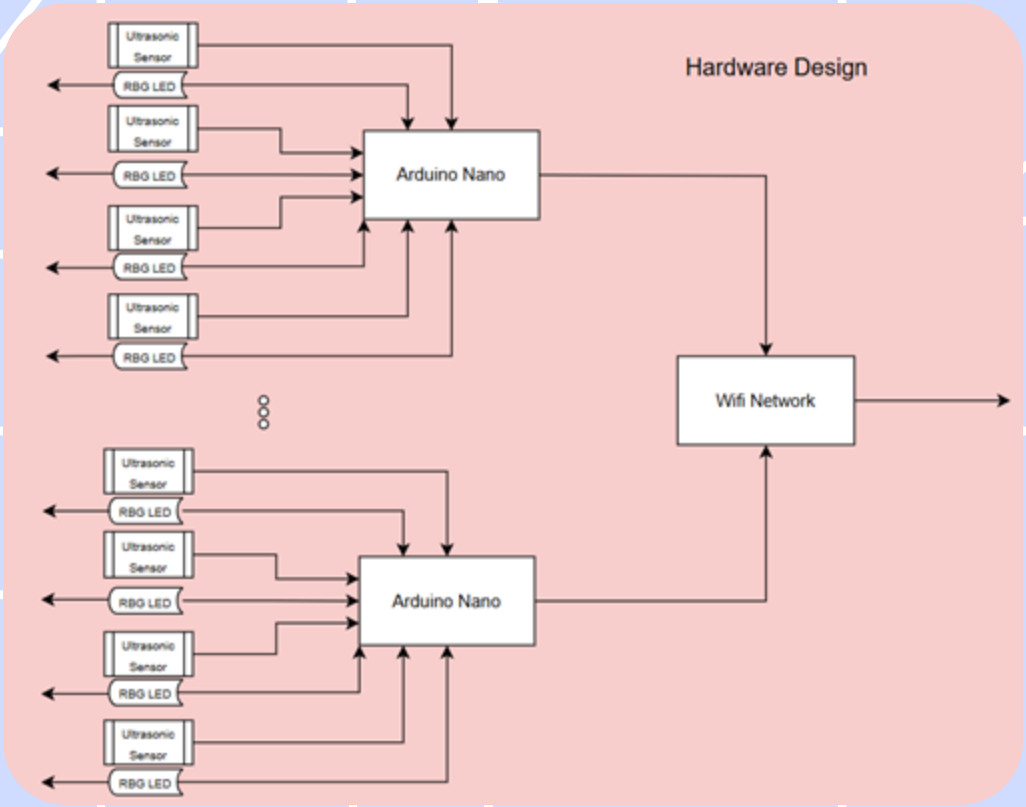


Hardware Design

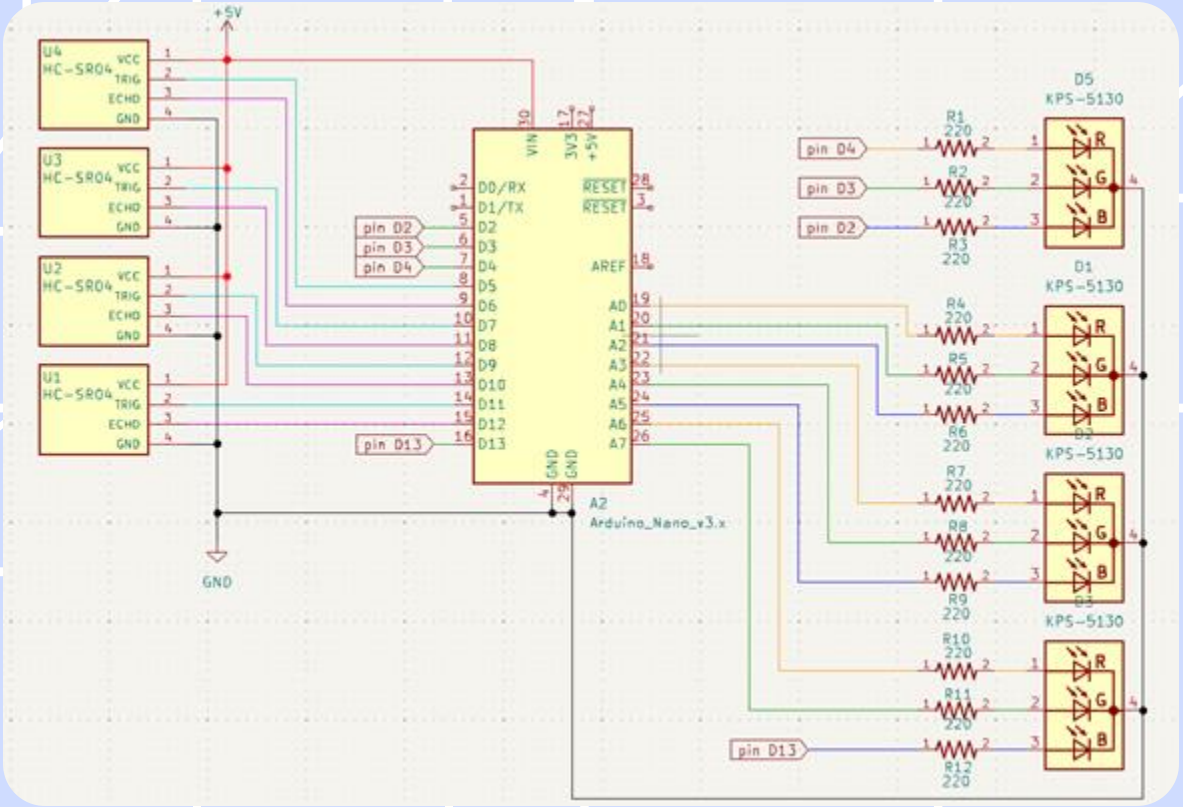
Conceptual Sketch of Parking Lot



Flow chart of Hardware design



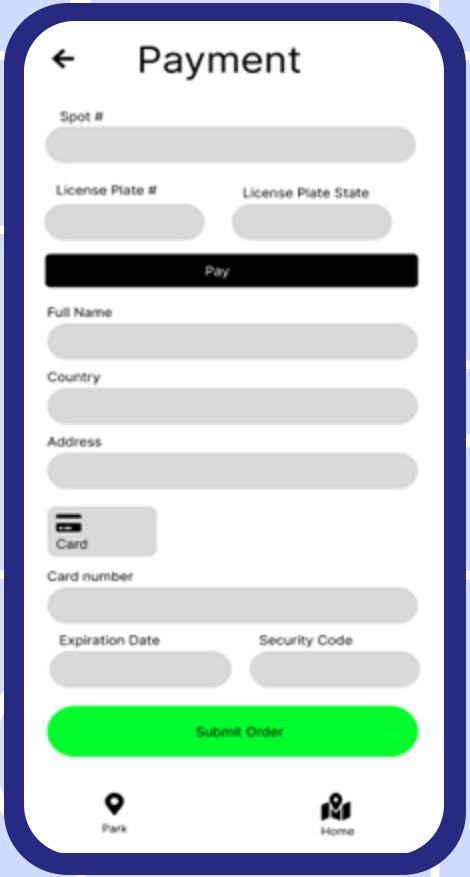
Simulated Hardware Design

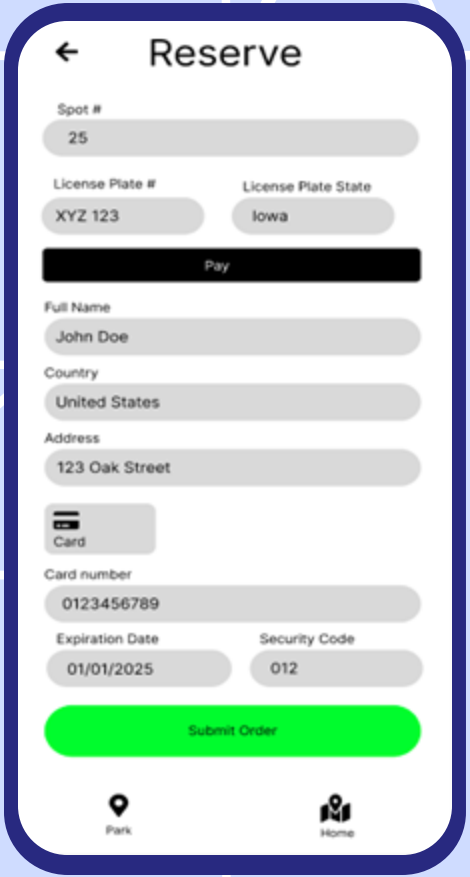


Software Design

Conceptual Sketch App Interface

Conceptual Sketch App Interface (Continued)





Conceptual Sketch App Interface

Backend Requests

Request Name	Inputs	Outputs	Description
Update Spot	Int spot_id Bool is_occupied	Int status	Updates the server if a spot is occupied and updates the LED on the sensor
Get Locations	N/A	Location[ ] locations	returns a list of all the location in our database
Get Available Spots	Int location_id	Int totalSpots Int availableSpots	Returns the count of spots and available spots of a give location
Get One Open	Int location_id	Spot openSpot	Returns one spot that is open to be reserved
Post Reserve	In to_reserve_id	status success	1) Check if the spot is still available 2) Process payment 3) Create reservation in database

Progress

Milestone	Projected Date
4 sensors per Nano	4/19/24
Nano Boards with WiFi connection	9/6/24
Got the system to run off of batteries	10/11/24
Communicate with server	10/18/24
LED lighting up according to sensor data	11/1/24
Home Page Prototype	10/1/24
Payment Page Prototype	10/27/24
Implemented Front-End	11/15/24
Implemented Backend	11/15/24
End-to-End Testing	11/22/24

Pending Issues & Concerns

Phone Use

Our project will require phone use while driving.

Connectivity

Wi-Fi connectivity in parking lots can be unreliable.

Reliability

Hardware and software must always be running.

Integration

Merging hardware and software has been difficult.