Senior Project Proposal - A.P.S (Automatic Parking Garage System)

Isaac Kime, Subin Kim, Sean Koo, and Sam Wimmer

***Abstract*—Parking is a universal problem when dealing with large influxes of people in a limited space due to a lack of knowledge of what spots are open, how many are left, how to get to these spots, and various other issues. One way to address this need for more knowledge is to create a system to inform potential parking garage users of the number of spots available, which spots are available, and provide an expedited intake and outtake experience. By using sensors, microcontrollers, circuitry, and computer software, our team hopes to provide a demo of a system that can address these problems.**

I. Introduction and Motivation

Driving around aimlessly in a parking garage looking for the needle in the haystack is something that frequently occurs in high-traffic areas with low parking space availability. This in turn leads to a loss in customers for both the parking garages and the surrounding establishments. The main source of frustration is a lack of information provided by the parking lot. For instance, if you already knew an entire lot was full, it would save the user time. It would also benefit the surrounding business due to customers having a more satisfactory parking experience. Due to the initial expense of building a parking garage, cost is something that must be taken into account to make this project feasible. It is because of this we aim to use a low-tech detection system, requiring only an IR sensor and a connection to some sort of microcontroller. The goal of the system is to have each parking spot have a sensor that outputs a signal that then gets sent to a microcontroller, which will send the data into a server back end, but also a circuit containing an LED to show a user driving around the parking garage what spots are open and what spots are taking. The server back end can also display data to the entrance and as well as provide other data on various other services such as an app. Our team believes this is the best approach to solve this common issue due to the low cost of light sensors compared to other detection ways, like pressure plates, in addition to the fact that it is very scalable. It also does not require much change to already constructed parking garages due to the low invasiveness of small LED sensors, assuming there are already power lines connected throughout the garage. The microcontroller can either use a physical network cable or wifi depending on the resources of the parking garage.

II. Related Work

One example of related work that is the most similar to our project is the publication "Monitoring Parking Space Availability via Zigbee Technology." In this article, Hee Chien Yee and Yusnita Rahayu go into depth on their technique to monitor vacant spaces in a parking lot. The main modules of the explained design are a microcontroller, a Zigbee module, a sensor module, and an LCD module [1].

A PIC 18F4550 microcontroller is used to control the system and do the logic and a Zigbee is used to transport that information using radio waves to the master module. Infrared sensors are used in each parking stall to determine if the spot is vacant. Once this information is recognized the LCD will display whether the parking lot is empty or full and the Zigbee will send individual stalls status to the GUI laptop display. The GUI on the laptop shows the different stalls and their occupation status.

There are a large number of similarities between our project and theirs, but a couple of differences between them are that we are going to combine the microcontroller and Zigbee by using a Rasberry PI, and we will show the count of parking spaces available instead of just a boolean value of open or closed, we will show LED lights at every individual stall to show whether that stall has a car, and we will take all of this information and store it in a database along with license plate recognition data.

III. Body

A. Project Tasks and Specific Task Interfaces

There are a few simple project tasks that can be completed individually, that when put together create the functioning parking system. The first one is completing a detection circuit. The inputs for this are the LED signal from the IR sensors which should output two things, a digital signal to the Raspberry Pi and an LED output of either red or green. If one LED is off, the other one should be on, indicating whether the parking spot is taken or not.

Another task is integrating the Raspberry Pi with the output of the LED detection circuit. The Pi should take that digital signal and be able to make SQL requests to an outside database using wifi to update the current taken and free parking spots. This task specifically involves writing code that interacts with the Raspberry PI GPIO pins and then based on those values making an SQL request.

The Database task will simply be creating an SQL database that can take in this Raspberry PI parking spot data and store it. The database should also be able to hold license plate numbers and a status flag, either being it is entering/currently in the parking garage and whether or not the specific license plate has been paid. It should also store a time of entry.

Another task is the machine learning algorithm. This program should be able to use a webcam to detect a license plate, and when it detects it, it should send an SQL request updating

the license plate number, status, time of entry, and payment status. Another is the Project housing, which should be able to hold the PCB design for the electronic detection circuit, the Raspberry Pi, and the IR sensors.

Our final task would be to design some sort of gate system or LED to indicate whether a car can enter or exit. This would be another program housed on the Raspberry Pi that will query the SQL database and if a car has recently entered or is trying to exit, it will send a signal out on the GPIO of the Raspberry Pi .

B. Testing and Integration Strategy

There are four main components to the automated parking system that will all be integrated to function correctly. It will consist of an LED circuit that will have an infrared sensor from the bottom to the top of the parking stall to determine if a car is currently in the spot. We will use inverters to turn on a red or green light according to whether the spot is available or taken. This circuit will send the digital signal to the Rasberry PI which is the next component in our system.

The Rasberry PI will be used for the computation and communication between all of the other components in our system. It will take the signal from the LED circuit and send this via WIFI to a database to store which parking spots are open and which are full. We will also have a camera system to detect a license plate upon entrance and exit of the parking garage using machine learning to read the plate and use the Rasberry PI to send this information to the database to be recorded. The license plate component will allow for tracking whether a car has paid and how long they have been in the garage. This will all be possible with the database component in our system. Our database will have two main tables that will track the parking spot status and license plate information.

1) LED Circuit: The LED circuit will have a green and red LED for parking status, an infrared sensor, and other logic gates to determine and show if a parking spot is taken. The green light will indicate an open space while the red indicates that it is taken. The sensor will give a digital signal which we can also use to send to the Rasberry PI. Testing this component will be relatively easy because our LEDs will indicate if we are doing this correctly.

2) Raspberry PI: A Rasberry PI is an SBC (Single-board Computer) that has access to many peripherals such as WIFI. We will use the microcontroller on the Rasberry PI to control our data using Python. We will then be able to communicate with the database and store the desired information from the LED sensor and license plate recognition in the desired tables. Testing this section will be important because our team is new to using a Rasberry PI so we will do so in phases and slowly increment our code to finish the project to ensure that we save time debugging.

3) License Plate detection: This component will consist of two cameras that will be at the entrance and exit of the parking garage. These cameras combined with the Rasberry PI will use machine learning to read a license plate and store/use database information according to whether that car is leaving or entering and if they have paid. We will test this component by holding license plates in front of the cameras and verifying the output of what we are computing to test that it is working as expected.

4) Database: The database will consist of information on the parking status as well as the license plate information. Upon arrival, we will store the time that the car enters along with its plate number. Then this information will be stored until they leave it will add the time that they left and how much they need to pay as well as if they paid. For parking status, we will have a table with all of the parking lots and alter the table every time a parking spot changes availability. We can test our database by writing test queries and then also testing it with our Python scripts to verify that we can add and alter information in the database.

C. Group Management and Communication Plan

1) Github.com

Our code files are going to be committed to the GitHub repository along with the Gantt chart, which is implemented with embedded Gantt chart function in Readme.md in GitHub.

2) Google Drive

For the documentation, our team is going to use Google Drive because it is one of the most useful documentation tools that is real time based. We are going to store our documentation data along with the datasheets of each part.

3) Overleaf

Since it is required to write the proposal in Latex format, we have decided to use Overleaf to write the proposal. Overleaf is a free website that group of people can edit the document in real time. It also clouds the document so that anyone in the team can access and edit in various devices.

4) Discord

We decided to use Discord for team's remote communication method because it supports both PC and mobile applications and it also support video chat with screen share as well. Discord is also very useful with the feature that user can refer to the previous chats and can create different sub channels for different type of information. Currently, our team has created a main chat channel and a sub channel just for all of the useful links.

5) Weekly meeting

Our team has decided to meet every Tuesday at 2pm for our weekly meeting. We discuss our progress in tasks, and help each other with some difficulties. We also have discussed that even though other teammate is in charge of a specific task, we still want to teach and learn from each other so that each individual can learn how to build the entire project. Therefore, the weekly meeting will be all about teaching and learning from each other.

D. Schedule and Milestones

In Figures 1, 2, and 3, our milestones and due dates are listed.

Milestones – Design		
Milestones/Tasks	**Person in charge**	**Due date (2024)**
Finish BOM and order parts	Team	Feb 15
PCB design of detection circuit	Isaac, Subin	Mar 15
Simulate detection circuit with Raspberry Pi	Isaac, Subin	Apr 1
Research APIs and tools for deep learning for car plate recognition	Sean	Apr 1
Research on database and how to connect Raspberry Pi to the database	Sam	Apr 1

Fig. 1. Design Milestones w/ Dates

Milestones – Implementation		
Milestones/Tasks	**Person in charge**	**Due date (2024)**
Physical implementation of detection circuit with Raspberry Pi	Isaac, Subin	Apr 23
Basic implementation of database	Sam	Apr 23
Basic implementation of car plate recognition	Sean	Apr 23
Building miniature demonstration model with car parking detection circuit	Isaac, Subin	Sep 1
Finish and simulation of database and car plate recognition	Sean, Sam	Sep 1
Simulation of connection between database, car plate recognition program, and detection circuit.	Team	Oct 1
Debug and risk handle buffer period in entire system	Team	Nov 1
Finish implementation of the system and prepare for demonstration.	Team	Dec 1
Create online parking lot status app (optional)	Team	Dec 1

Fig. 2. Implementation Milestones w/ Dates

Milestones – Documentation		
Milestones/Tasks	**Person in charge**	**Due date**
Proposal first draft	Team	Mar 1
Proposal second draft		Apr 1
Proposal final draft		Apr 20
Proposal presentation		Apr 20

Fig. 3. Documentation Milestones w/ Dates

E. Risk Assessment

1) High Risk
   a) Machine learning
      i) Risk Description:
      The implementation of license plate recognition should leverage Machine Learning (ML) techniques. However, our team lacks prior knowledge and experience in ML, which poses a challenge in integrating ML methodologies into our project effectively.
      ii) Mitigation Plan:
      It is imperative to study Machine Learning's principles and applications, as well as understand the methodologies for integrating with database connectivity to align with our project's requirements.
2) Medium Risk
   a) Sourcing Raspberry Pi
      i) Risk Description:
      The proper configuration of the Raspberry Pi is essential for the optimal functioning of this system. However, our team has not previously used the Raspberry Pi.
      ii) Mitigation Plan:
      Given the Raspberry Pi's reliance on Python, acquiring proficiency in Python is essential for effectively sourcing and programming the Raspberry Pi. Additionally, it is advisable to consult and analyze other similar projects utilizing Raspberry Pi to gain insights and enhance our implementation strategy.
   b) Interfacing Raspberry Pi with database and circuit
      i) Risk Description:
      It is necessary to establish data communication between the Raspberry Pi, the circuitry, and the database for the system to function effectively. Nonetheless, our team's understanding of Raspberry Pi's technical details and application methodologies is currently insufficient to implement it successfully in our project.
      ii) Mitigation Plan:
      The mastery of Raspberry Pi operations is crucial for our project, particularly for facilitating data transmission between the Raspberry Pi, the circuit, and the database. Therefore, it is beneficial to refer to other similar projects utilizing Raspberry Pi to gain insights and strategy.
   c) APP
      i) Risk Description:
      We have resolved to develop an application that integrates with this automated system. Achieving this requires comprehensive knowledge of interfacing the application with both the Raspberry Pi and the database. Furthermore, given

the interaction between software systems and hardware components, it is also imperative to understand the methodologies for connecting software systems with hardware parts effectively.

ii) Mitigation Plan:
Utilize a no-code platform to facilitate the straightforward creation of an application without the necessity for programming.

3) Low Risk

a) Circuit Design

i) Risk Description:
The design of the circuit necessitates careful consideration of component selection and the identification of an efficient circuit layout to ensure optimal performance.

ii) Mitigation Plan:
It is beneficial to refer to similar circuit designs and what components they used in designing our circuit design.

b) PCB design

i) Risk Description:
We should develop a streamlined and efficient Printed Circuit Board (PCB) layout to ensure the system's optimal functionality and compactness.

ii) Mitigation Plan:
Experiment with multiple designs and select the most efficient one for implementation.

c) Database

i) Risk Description:
It is necessary to capture and store data concerning vehicles entering the parking area. To accomplish this, understanding the process of acquiring data from the Machine Learning component is essential.

ii) Mitigation Plan:
Investigate the methodologies for data acquisition and explore techniques for retrieving data from Machine Learning. Should this process prove challenging, it would be advisable to refer to other machine learning studies for guidance and alternative approaches.

d) Housing

i) Risk Description:
The implementation of this comprehensive system requires the construction of a physical parking space. Utilizing 3D printing technology is pivotal for crafting the parking space infrastructure.

ii) Mitigation Plan:
Secure early access to the 3D printers at the University of Utah to facilitate the creation of our physical parking space by our plans.

F. Bill of Materials

BOM		
Part	QTY	Part Number
Raspberry Pi	1	SC1112
IR break-beam sensor	12	2167
LED (Red color) 25 Pack	1	299
LED (Green color) 25 Pack	1	298
10kΩ Resistor	12	CF14JT10K0
Micro SDXC Memory Card	1	LSMICRO64GU3D
USB C Pi Power Supply	1	B0CQ2DL2RW
XOR IC	10	SN74HC86N

TABLE I
BILL OF MATERIALS

G. Vendor List

Vendor List	
Part	Vendor
Raspberry Pi	Raspberry Pi
IR break-beam sensor	Adafruit
LED (Red color) 25 Pack	Adafruit
LED (Green color) 25 Pack	Adafruit
10kΩ Resistor	Stackpole Electronics, Inc
Micro SDXC Memory Card	Amazon Basics
USB C Pi Power Supply	GeeekPi
XOR IC	Texas Instruments

TABLE II
LIST OF VENDORS FOR EACH PART

IV. CONCLUSION

The first milestone that we are going to accomplish is writing this proposal for the project. Then we need to design the detection circuit and then design the PCB. After that, a lot of the milestones can be completed in parallel such as interfacing the Rasberry PI to the database, the LED circuit, and the license plate detection. We can then create the model garage and put all of the components together to test the whole system. The two largest risks we are taking on as a team are the usage of machine learning and a Raspberry PI because nobody in our team has experience with either of these two technologies. We plan to start early with these sections to ensure that the majority of the risk is mitigated and controlled by the end of the semester.

The benefit of this project lies behind the struggle that can be faced in parking garages. With this project, we will be able to make a hands-free experience of pulling into, parking, and exiting these parking garages without the hassle of searching for extended amounts of time for a parking spot. An individual will be able to easily determine where spots are empty and remove the need for keeping track of your parking slip when trying to exit and pay for the time spent in the parking lot.

REFERENCES

[1] H. Yee and Y. Rahayu, "Monitoring parking space availability via zigbee technology," *International Journal of Future Computer and Communication*, vol. 3, pp. 377–380, 12 2014.