CSS: Car Sentry System

Qrizelle Crisostomo, Ricardo Nunes Alcobia, Ari Pantoja, and Robert Zarrella

Dept. of Electrical and Computer Engineering, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — This paper describes a detection system that aims to recognize and document license plates using object recognition via computer vision software. The system will be comprised of a combination of different components: a single board computer to perform computer vision operations, a project controller to communicate with the single board computer and various modules, local storage to preserve obtained data, Bluetooth module to transmit data to a mobile application, and power system to supply the appropriate amount of power to the entire unit. For the convenience of the user, the unit will be lightweight and portable so it can be utilized in any vehicle. Overall, the unit is to be used to log license plate data in a case of a hit-and-run or any other vehicle-related accident.**

***Index Terms* — Batteries, Bluetooth, computer vision, converters, microcontroller, mobile applications, object recognition.**

I. Introduction

Driving is a dangerous activity. Operating a piece of moving steel weighing thousands of pounds is a daunting task. Human error further increases the chances of motor vehicle accidents to occur. Accidents include various kinds of collisions involving two or more vehicles. Unfortunately, there are drivers that worsen these situations by fleeing the scene (hit-and-runs). Drivers who experience these kinds of incidents likely lack any form of surveillance equipment.

Why not purchase a dash cam? Baseline dash cams do not provide the quality required to identify license plate information. While higher quality dash cams do allow for better video output, the increased quality leads to the increase in price. Currently, a good dash cam will cost at least a hundred dollars, if not more. The high cost of dash cams currently on the market makes it difficult for the average day-to-day driver to fully equip their vehicle with the necessary safety precautions. Without these devices, drivers are not able to document accidents for insurance and/or legal purposes and are oftentimes at a loss.

The Car Sentry System (CSS) has been developed to provide a low-cost solution for the everyday driver. The Car Sentry System's primary focus is to quickly obtain a vehicle's license plate information and store it for future reference. Data will be stored locally and be transmitted via Bluetooth to a mobile application for ease of access. For added convenience, the unit is constructed to be lightweight and portable. This allows for the user to utilize the unit in any enclosed, 4-wheeled civilian vehicle.

A. Goals and Objectives

While the goal of CSS is to document license plates as a user is driving on the road, there are several objectives the unit must satisfy to enhance safety and provide ease of usability. To solidify our solution with this project, our team has applied advanced engineering techniques and put into perspective the needs of the day-to-day driver. These techniques have been derived from the knowledge gained throughout our undergraduate career in addition to personal research. With our experience driving to and from the Orlando area, we were also able to consider the practicalities of a license plate scanner.

The following list describes the objectives CSS achieves to ensure usability and user safety:

- Scan license plate information for enclosed, 4-wheel civilian vehicles
- Plug-and-Play functionality
- Lightweight & Portable Design
- Crash Survivability
- Prevent Obstruction in Driver View
- Build to IEEE/IEC/UL Standards
- Modular codebase
- Accessible and Convenient to Users (Mobile Application)

B. Specifications

There are a set of specifications that act as the foundation to produce an accurate, viable license plate scanner. Each specification can be classified as either a hardware or software specification. The hardware specifications relate to the needs of the physical components and overall construction. The software specification identifies the services required for the unit to function accordingly.

Shown below are the specifications created for the CSS unit and satisfied post-development:

CSS Specifications		
Battery	*Discharge Life*	3 - 5 days (Avg. 40-60 minutes/day)
Camera	*Resolution*	1080p with accuracy of ≥ 90%
Camera	*Frame Rate*	Process video feed ≥ 20 fps
System	*Dimensions*	Will not exceed 5”x4”x4”
System	*Weight*	< 2lb
Enclosure	*Survivability*	Readable storage after 2-story drop

II. System Overview

The Car Sentry System design can be divided into several subsystems and components. Shown below is a high-level overview of the system components.

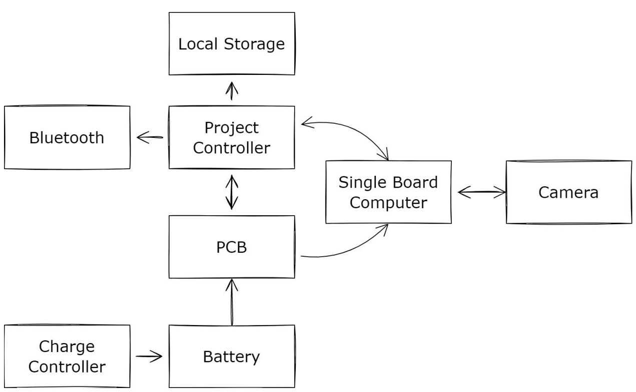

Fig 1. General System Components Diagram

The PCB will be the hub in which all the components will interact with one another. The project controller and modules, the single board computer, and the power system will all be attached to the PCB.

The single board computer will retrieve license plate information from the camera input and the system software. This data will then be transmitted to the project controller where the local storage, or the microUSB, will store files of the license plate data obtained. It will also be sent to the Bluetooth module so the mobile application can retrieve and store that data on the database. In order for the unit to power up, there will be a power supply attached to the PCB. This power supply will grant users different options of providing power to the entire system.

III. Power System

The Car Sentry System is designed to be powered from a wall adapter, vehicle auxiliary power port, or the internal battery. This choice was made to give the end user supreme flexibility in their deployment of the system and allows them to hardwire CSS into their vehicle, or operate it solely from battery power.

The CSS power system is designed to be able to handle a wide range of inputs, since the CSS needs to be powered from either an AC to DC mains adapter, hardwired into the vehicle power, or from the vehicle auxiliary power depending on use case. Two 3.7 volt 10,000 mAh Li-Po batteries are connected in series yielding 7.4 volts, for 74 Watt-Hours of energy which is more than sufficient to achieve the design goal of 5 days of operation at 60 minutes per day. The regulated system power is provided by two voltage rails: 5 volt and 3.3 volt. The rails are cascaded to maximize efficiency, since the primary variable for switching regulator efficiency is the delta between input and output voltage. Additionally, since the entire power system is cascaded with respect to voltage levels and the battery voltage is sufficiently high over the entire discharge range, the system can utilize buck regulators which offer higher efficiency and lower cost, instead of sacrificing efficiency with a buck/boost topology as the primary concern of the power system is maximizing battery life to meet our stated specifications.

A. Charge Controller

The primary component of the power system is the Texas Instruments BQ24703 charge controller. It provides for a wide input range of -0.3 volts up to 30 volts and facilitates seamless source switching to allow system power from wall adapter, car adapter or battery without system interruption. This is particularly important in the design case of the CSS, since a main function of the device is to provide evidence in the case of a car accident, where the power adapter may be jostled out of the socket at the critical moment, leading to loss of data and functionality. The charge controller also provides the ability to concurrently provide system power while charging the battery, which allows the system to retain full functionality during battery charging to provide the end user with the maximum system configuration options. External input power is sensed from a feedback net connected to the power adapter port, and the comparators internal to the BQ24703 sense a change in input power within 120 to 150 nanoseconds, and the system source control MOSFETs are switched accordingly. The battery isolation MOSFET is connected with the source terminal connected to the system power network, such that the body diode of the MOSFET instantaneously begins to conduct current when the system voltage sags to the level of the battery. If the load is demanding enough such that the source control switching is not fast enough to keep the system voltage from sagging when the power adapter is disconnected, the current through the body diode shall provide the necessary current.

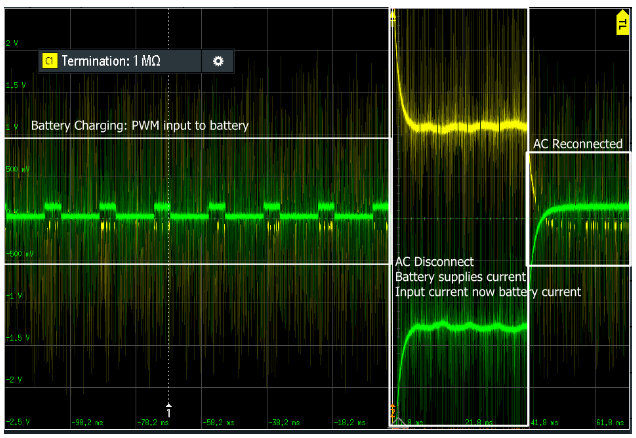

Fig 2. Seamless Source Switching Detail

The primary drawback of the BQ24703 is its large PCB area and susceptibility of external sensing nets to noise.

B. Voltage Supply Rails

The Texas Instruments LM3150 switching voltage regulator is utilized to supply the 5 volt rail because of its high efficiency, low cost, external switching, and robust safety features. This regulator has a nominal efficiency of 96.3 percent in the configuration utilized for the CSS, while the entire bill of materials is only 61 percent the cost of a comparably efficient system. By utilizing external MOSFETs, the system achieves high efficiency, high current switching while allowing much greater heat dissipation, and easily handling the large input voltage range demanded by the CSS operating environment.

The Texas Instruments TLV62568 was chosen because it offered high efficiency with low PCB area and cost, while providing safety features comparable to the LM3150 stated previously. The TLV62568 utilizes internal switching MOSFETs and therefore the device must dissipate any heat generated by the voltage regulation process, but the current demanded by the 3.3 volt rail is only 0.75 amps. Since the CSS is in a high temperature environment, the thermal metrics of the TLV62568 provided by TI were utilized to determine if any additional heat dissipation was required for the 3.3 volt regulator. It was found that the maximum junction temperature was 80.9 degrees Celsius, which is far under the allowed maximum junction temperature of 125 degrees Celsius and thus no additional heat dissipation devices are required.

C. Battery

The MakerFocus 9065115 battery was chosen because it offered a wide range of desired features and was one of the only budget minded options which could be verified to meet the industry standards the team targeted, namely IEEE 1725-2021 and IEC/UL 62133. The features desired fall into two categories: efficiency and safety. Efficiency features are the lithium polymer battery chemistry, which provides extremely low self-discharge of 0.3 percent of capacity per month, and provides an extremely flat discharge voltage curve to provide stable system operation. The protection circuit module provides an extremely low quiescent power consumption. Safety features include Lithium Polymer's inherent resilience to high temperatures, and a circuit protection module which isolates battery contacts under short circuit, overtemperature, and overcurrent conditions. The battery capacity was specified based off of the power modes available on the nVidia Jetson Nano, which throttle the CPU and GPU to keep the maximum power draw under 10 Watts. Under testing, the total system power draw under a sustained 80% combined CPU and GPU load is 8.246 Watts, which the two internal batteries can supply for

$$\frac{74\,Watt*Hour}{8.246\,Watt} = 8.974\,Hours \tag{1}$$

Which is far more than the value of 5 hours specified by the design team, and is bolstered even further by the fact that the computer vision software under full load places the CPU and GPU under a combined load of approximately 60%, and this only occurs while a license plate is present and is significantly less otherwise.

IV. Electronics System

The electronics system consists of two main components, the single board computing module and the project controller. The Jetson Nano was picked as the single board computer for its processing power, price point, resources available for working on the platform, and wide range of supported communication protocols. For the project controller, the MSP430FR6989 was selected due to familiarity working with the platform, its low power consumption, and supported communication protocols.

In this manner, the Jetson Nano will be solely focused on running the computer vision program and will be free from interruptions. As data is obtained, it will store it and send license plate information to the MSP430. The MSP430FR6989 will then handle writing to an SD card located on the PCB and transmitting the data wirelessly to the mobile application using the Nordic Bluetooth module.

A. MSP430FR6989 Microcontroller

The TI MSP430FR6989 was chosen due to availability and previous knowledge from required coursework. The MSP430 takes in license plate data from the Jetson Nano, writes it as a file in the SD card and transmits the plate number via Bluetooth to a mobile app. In addition to an SD card tray, the CSS needed a Bluetooth capable module. We chose the Raytac MDBT42Q-512KV2 for its availability and low cost.

The SD card operates utilizing SPI communication and the Bluetooth module is capable of any communication scheme as it is its own system on chip. To conserve resources on the MSP430 for later added features, the same SPI bus was utilized for both the SD card and Bluetooth module. This freed up EUSCI channels A0, A1, and B1 for additional communications from the MSP430 with the power system.

Originally we wanted to connect the Jetson Nano to the same SPI bus because of convenience and speed, but in practice this proved to be too difficult. The CSS team ended up successfully receiving communications from the Jetson Nano via UART to the MSP430.

B. Micro SD Card

We realized that the MSP430FR5994 had example code in which TI wrote to an SD card on the corresponding TI Launchpad for that chip. This was used as a blueprint to develop writing to an SD card for the MSP430FR6989. In the code setup for the FR6989 model, initialization designations of the necessary pins were the reverse of the FR5994. After thoroughly researching the “Out of the Box" code for the MSP430FR5994 where the microcontroller creates a file on an SD card, the correct function calls were identified and we were ready to start a new project in Code Composer Studio and utilize the functions for the MSP430FR6989.

In our final code, the MSP430 waits for the Jetson to send data via UART. The appropriate data is received from the Nano by only recording each character in the array after the ‘$’ symbol is received. If each character received is an alpha-numerical, it is saved in the character array created in the MSP430 memory. The array is closed out by receiving a ‘$’ again from the Jetson. This was done to eliminate garbage values received throughout the process.

We edited the function “createFileSDCard();” to receive a character array and use it to create the .txt file on the SD card. The function creates a string beginning with the file directory “plates/” and then concatenates the string with the license plate number and file extension “.txt”. Then the Fatfs library is utilized in the same manner as on the FR5994 to create, open, and save the file.

It was originally thought that in order to write at only 8 MHz, an older and much smaller SD card would be needed to use with the Fatfs library. In practice, the Fatfs library connects all the necessary drivers to write to or read from any SD card. There exists conditional statements within the Fatfs library that differentiate between each SD card type and executes the initialization process needed accordingly.

C. Raytac MDBT42Q-512KV2 Bluetooth Module

The Raytac module recycles use of the Nordic Semiconductor function calls. We combined the use of a Nordic Semiconductor SPI Slave example and a Bluetooth send example. The Raytac Bluetooth module is set up as a slave to the MSP430. The existing code receives data via SPI from the MSP430 and then sends it out via Bluetooth to the “Nordic UART Service”.

In this way we can utilize the “nRF Toolbox" app and select the “UART” tab to receive data from the MSP430. This can be used as a preliminary test as to the functionality of the Car Sentry System. This was later configured to be accessed via the PWA mobile app developed specifically for the CSS. The PWA was successfully connected to the CSS and output license plate data.

V. Printed Circuit Board

The large PCB area demanded is in direct opposition to the requirement of the CSS to fit within a 4-inch by 4-inch area, but this problem was solved by utilizing both the top and bottom layer of the PCB to mount components. This design choice also afforded more flexibility in minimizing high current loops in the charging and 5 volt supply nets such that unintentional inductive EMI generation was minimized. The charge controller and both voltage regulators employ high frequency switching while employing millivolt sensing nets which are sensitive to noise coupled from such high frequency nets. By employing a dual sided design, there was sufficient area available to route these high frequency switched nets far from the noise sensitive sensing nets which helps protect system integrity. The Bluetooth module presented a special case, in that it required absolutely no copper pours along the entire edge of the PCB on which the antenna was mounted. Despite minimizing the area of signal current loops, it was impossible to eliminate them completely. This is exacerbated by the larger currents and/or high frequency switched signals of input/output connectors. To mitigate this, the input and output connectors were routed far from switching electronics. While most signals routed on the PCB were low current, and thus utilized small gauge 6-mil traces, the 5 volt rail which supplies the Jetson needed to provide current up to a maximum of 2.5 amps. The copper layers of the PCB are 0.05 mm or 1.968 mil which constrains the trace width required to conduct the required current, since current is dependent on the cross-sectional area of the conductor. Utilizing the IPC-2221 standard to calculate the trace width given, it was determined that 40 mil width would accommodate 150% of the demanded current with a 20 °C temperature rise.

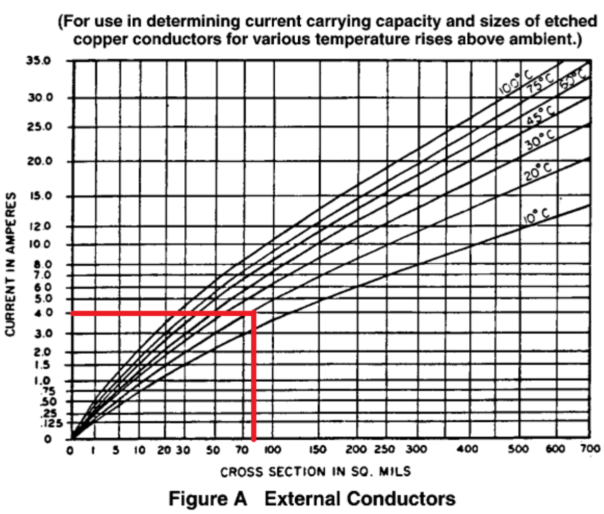

Fig 3. IPC-221A Cross Sectional Area vs. Conductor Current

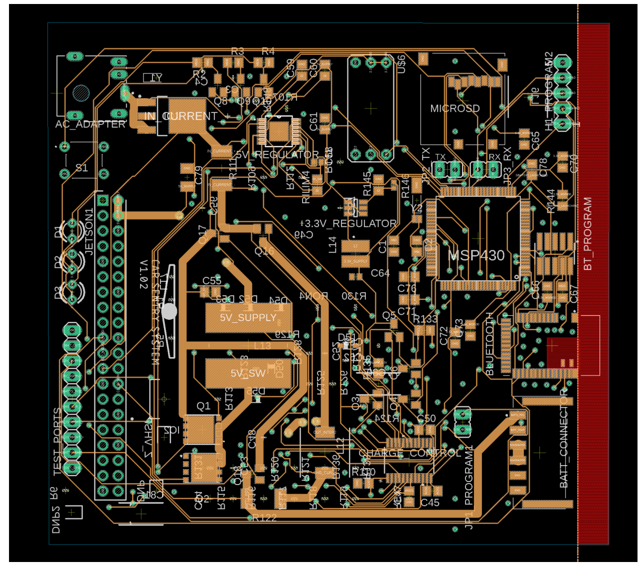

Fig 4. Top Layer of the PCB

The PCB was designed with the intention that SPI would be used for communication between the MSP430 and the nVidia Jetson Nano, but the SPI protocol proved unusable and thus the system was reverted to UART communication. There were no provisions for UART communication on the PCB, but it was not possible to produce a revision in time. Two 24 AWG wires were soldered directly from the programming headers of the MSP430 to the underside of the header for the Jetson, and this was sufficient to enable UART communication between the two.

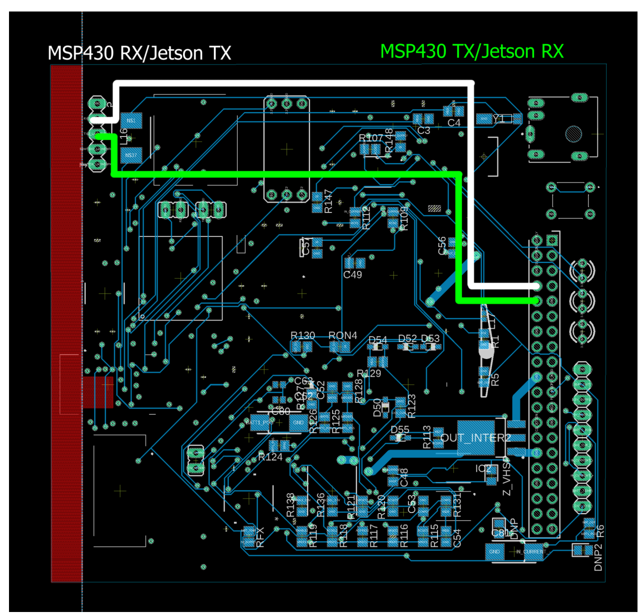

Fig 5. Bottom Layer of the PCB

VI. Enclosure

All CSS components are fitted into a custom enclosure. The enclosure model was designed using Autodesk's Fusion 360 and printed using an Anet A8 3D printer. The design began using the overall size limitations of 5" x 4" x 4". The rake angle of one of our car windshields was measured and that information was used in the angle and placement of the suction cup mechanism on the case.

The suction cup mechanism was incorporated into the overall case. It functions by spinning the switch protruding on the driver's side of the case. When the switch is engaged, the lever spins the bracket holding the suction cup in place to a side with a longer distance between the holding pin and surface. This increases suction between the case and the windshield for greater hold.

Inside the case are racks to hold up the Jetson to prevent too much torque on the GPIO pins. There is also a rack perfectly fitted to hold the PCB in place inside of the case. On the bottom of the inner cabin of the case is an indentation to house the rechargeable batteries.

There is a camera opening at the very front of the case. The camera is held in place by 4 nylon screws and 2mm nuts embedded in the case. The angle of the camera can be adjusted by tightening or loosening the mounting screws. There are also openings cut out of the case for accessing the SD card and power adapter.

Finally, the case door slides in and out of the top of the case for secure yet easy access to the CSS electronics when not affixed to a windshield. Shown below is the final enclosure.

Fig 6. The CSS Enclosure

VII. Software Design

The software component comprises two different programs: computer vision and mobile application. The computer vision is responsible for all the license plate detection while the mobile application takes the captured license plate data and displays it for the user.

A. Computer Vision Functionality

The entire license plate capturing process has been completely automated on the single board computing platform. This provides users with a plug-and-play experience in which all that is required from them is to add an SD card and plug in CSS to a power source. Upon being connected to a power source, the system will go through its boot up process. After ensuring that all essential startup processes have been completed, the license plate detection program will launch. CSS will then capture video footage in real time with a resolution of 1920x1080 at 30 frames per second. Alternatively, it can also capture at a resolution of 1280x720 at 60 frames per second. The first resolution allows for greater accuracy of plates at a distance, while the later uses less system processing power to capture plates at a closer distance.

Fig 7. OpenALPR identifying license plates and extracting information

License plate detection, as shown above in Figure 7, is achieved via OpenALPR. This allows for plate capturing of all states in the U.S, as well as 70 other countries. However, CSS has been optimized for reading U.S plates. This makes it easier to recognize U.S license plates. Each frame captured is parsed to extract license plate information. Only information obtained with a high degree of confidence will be recorded. A still image of the captured frame will also be stored on the single board computer as a secondary backup, while the extracted text will be passed on to the project controller utilizing the UART communication protocol. A log of the entire running process of CSS will also be stored in a debug file on the single board computer for testing or record keeping purposes.

From there the project controller will write the license plate information to an installed SD card that is easily accessible for users. In addition to writing to the SD card, the project controller will also pass the information to a Bluetooth module to transmit the information in real time to a connected Bluetooth device. Which will then store the information in a cloud database as an offsite backup option and for even greater ease of access for users.

B. Computer Vision Technologies

Computer vision for license plate reading is realized on the Nvidia Jetson Nano platform with OpenALPR, an open source library built using OpenCV for the license plate detection algorithms and Tesseract for the optical character recognition. Additionally, OpenALPR provides pretrained models for license plate detection in over 70 countries. This was a major factor in the decision to implement OpenALPR versus a custom trained model, as it proved difficult and time consuming to obtain a large high quality data set to train with.

The IMX219-130 camera module has built in integration for the Nvidia Jetson Nano ecosystem. It also allows for a range of resolutions and frame rates, from 720p at 60 frames per second to 2140p at 21 frames per second. As well as featuring a field of view of 130 degrees. All these factors make it a lucrative option for capturing input.

Python was the programming language of choice for the CSS codebase, due to the vast library of tools available, ease of development, and readability. Libraries such as Py-serial significantly sped up the development process of implementing UART protocol communication on the Jetson Nano platform. OpenALPR also has Python bindings allowing for easy integration.

C. Mobile Application Functionality

The mobile application is intended to be simple and easy to use. Users will be able to register, login, and view data. Figure 8 depicts the use case diagram of the application.

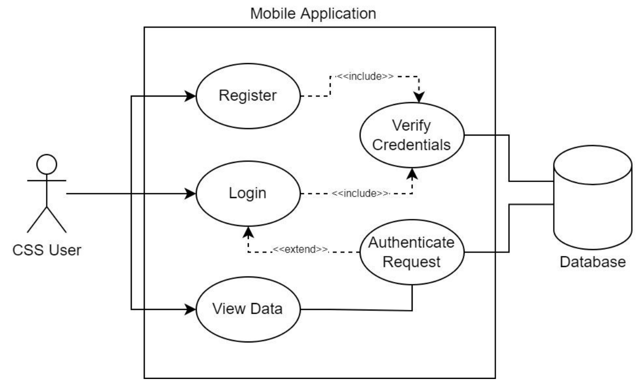

Fig 8. Use Case Diagram for Mobile Application

When users access the application, they will be presented with a basic graphical user interface. The first page they will encounter is the *login* page. There they will be prompted to insert their login credentials. This is to establish data security. If a user does not have an account, they will have the ability to create their own account on the *register* page. When registering, users will be able to create their own username and password. In the case the user forgets their password, they can simply request to reset their password on the *forget password* page.

Upon login, users will be presented with a few features. Users will be able to update their profile in the case that login credentials need to be changed. They will also have the ability to logout with the *log out* button. However, our primary focus is the Bluetooth functionality and the output of license plate data.

On the navigation bar, users can find a “Connect CSS Unit” button. Here, they will detect their CSS unit via Bluetooth. CSS should be the only one to appear as there are not many devices using Nordic UART Bluetooth services. Once the application detects the unit, the user will be able to connect to the device. With the established connection, any detected license plate and a timestamp will be captured by the Jetson Nano, transmitted to the Bluetooth module, and appear on the table of the mobile application in real time.

D. Mobile Application Technologies

The software created is based on the FERN Stack framework. The technologies used for this stack are of the following: Firebase, Express, React, and Node.js. This stack has been selected due to the consistency of the code and the overall ease of use.

Firebase provides authentication and data storage in a No-SQL database. User interaction is done with the frontend, which is constructed using React. The REST API/backend of the application is designed using Node.js and Express. These components are all interconnected as user authentication data is being transmitted between Firebase authentication and React frontend. When a user communicates through the web application, these requests are transmitted across the frontend and the backend's REST API. To complete the loop, the data requests and the data obtained is being shared across the backend and the stored user data of Firebase's realtime database (Firestore).

VII. Conclusion

An extensive amount of research has been completed to ensure that the selected components were compatible with one another while also being able to achieve the listed specifications. The batteries and heat sink accounted for a large portion of the weight and size, making it more difficult to work with the constraints provided during the development process. However, the end result proved fruitful, as a lightweight and compact form factor was achieved, which was essential in ensuring that the CSS does not obstruct the driver's field of view. Furthermore, the complete automation of CSS successfully achieved the goal of creating an easy to use and accessible device for users.

CSS lays the foundation for further ALPR projects as a lot can be expanded upon. Due to the time constraint, a custom license plate recognition model was forgone. Instead, OpenALPR was used for its rapid deployment and wide range of plates it could recognize. CSS now provides the perfect platform for easily detecting and capturing license plates, which can be used to quickly obtain a large data set to train a custom model. The model could also be expanded to include car model, brand, and color detection. Additionally, CSS may be enhanced with a GPS to track location. The mobile application can also be further expanded to include a map of where plates were detected and also to include the images captured.

VIII. The Engineers

Qrizelle Crisostomo will graduate with a Bachelor's degree in Computer Engineering from the University of Central Florida in Spring 2022. She has accepted a position for one of Disney's Professional Engineering Internships where she will work full-time in Attraction Simulation.

Ricardo Nunes Alcobia will graduate with a Bachelor's degree in Computer Engineering: Digital VLSI Circuits Track from the University of Central Florida in Spring 2022.

Ari Pantoja Ari will graduate in May 2022 with a Bachelor's degree in Electrical Engineering. He will then begin working at Enercon Services, Inc as an Electrical Engineer Level IV in the Utilities and Industrial Division.

Robert Zarrella will graduate Cum Laude from the University of Central Florida with a Bachelor of Science in Electrical Engineering. He has accepted a position with ENERCON as an Electrical Engineer Level IV in the Utilities and Industrial Division.

Acknowledgement

The authors wish to acknowledge the assistance and support of Dr. Samuel Richie, Dr. Lei Wei, Dr. Zakhia Abichar, and David Douglas from the University of Central Florida along with the faculty and staff of the Evaluation Committee who have agreed to review our project.

References

[1] Nvidia, "Jetson Nano Module Data Sheet," 12 February 2020. [Online]. Available: https://developer.nvidia.com/embedded/downloads#?search=Data%20Sheet

[2] François Beaufort, "Communicating with Bluetooth devices over JavaScript," 21 July 2015. [Online]. Available: https://web.dev/bluetooth/

[3] Texas Instruments, "MSP430FR698x(1), MSP430FR598x(1) Mixed-Signal Microcontrollers," 29 August 2018. [Online]. Available: https://www.ti.com/product/MSP430FR698

[4] Institute of Printed Circuits, "IPC-221A Generic Standard on Printed Board Design," May 2003. [Online] Available: http://www-eng.lbl.gov/~shuman/NEXT/CURRENT_DESIGN/TP/MATERIALS/IPC-2221A(L).pdf