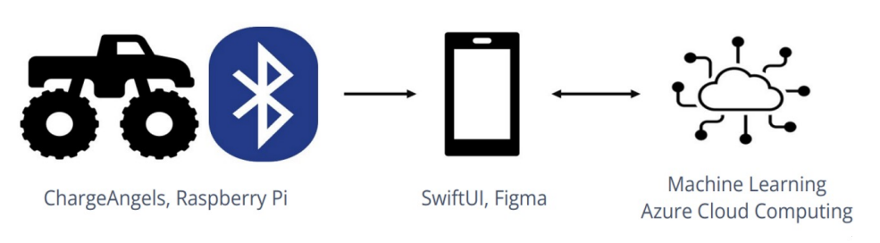
MAKING ELECTRIC VEHICLE CHARGING FUN

STUDENTS: GERIN GEORGE, MOSKA JAMALI, RICO LI, BRENDAN OQUIST, KEVIN SHAO, DIANA VERDUZCO, BRYAN VO

envorso Transforming Productivity

Ford

The VoltVision App - Objectives



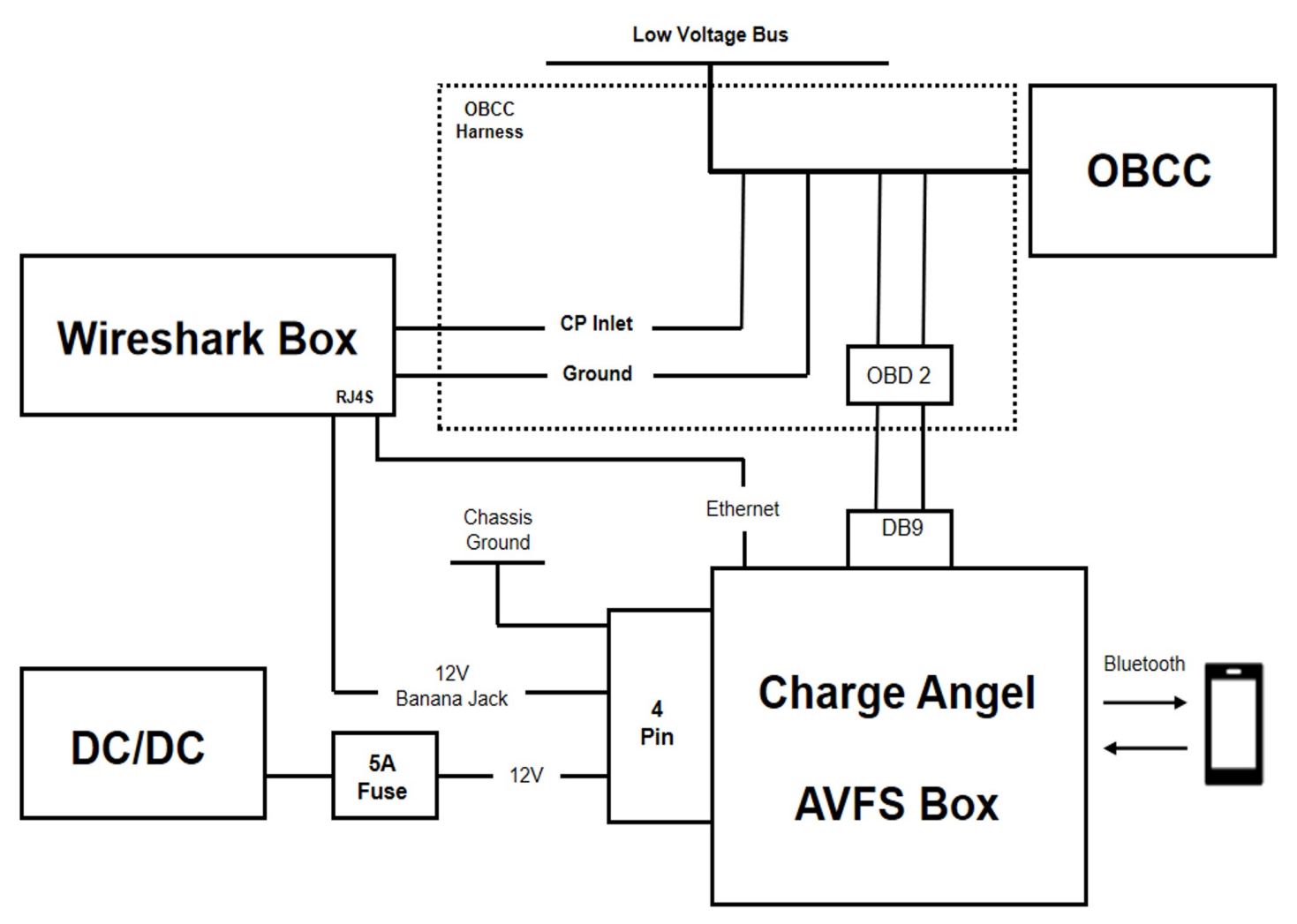
ChargeAngels, Raspberry Pi
SwiftUI, Figma
Machine Learning Azure Cloud Computing

- Design the VoltVision Smartphone App to monitor and analyze data generated by electric vehicle(EV) chargers, enabling accurate assessment of their health conditions and performance.
- Develop features to enhance the convenience of charging an EV for customers and display status of current chargers at a charging station.
- Integrate a mapping feature, providing quick routing options of nearby EV charging stations via Apple Maps.

Hardware - Overview

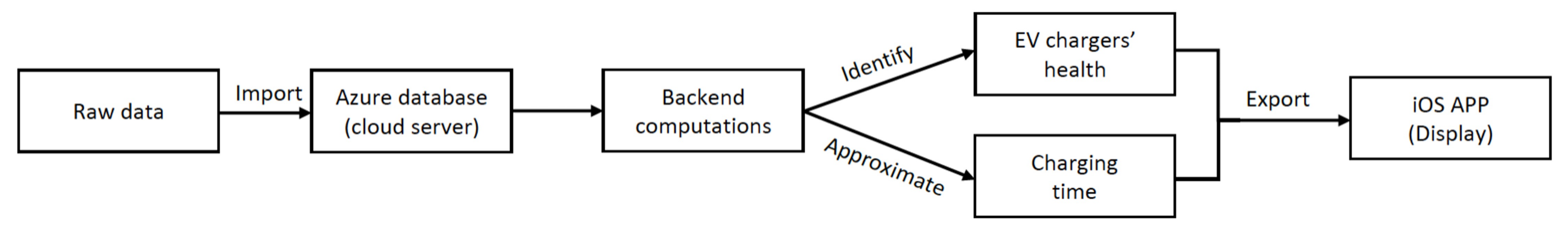
- The Charge Angel AVFS Box is a hardware system developed by Ford to collect the charging data from the car.
- The Raspberry Pi included in the system is being used to store the log data that is generated during charging sessions.
- Charging logs are generated after the end of a charging event and stored in the Raspberry Pi's file system.
- Using the Raspberry Pi's bluetooth connectivity, we send the most recent log to a connected phone via bluetooth, where our VoltVision app is able to interact with it.

Hardware - Data Collection and Bluetooth



- OBD2 is a standard diagnostic system in modern vehicles, capable of collecting various data such as Vehicle Speed and Engine RPM.
- Wireshark Box uses the Wireshark network protocol analyzer tool to capture and analyze network traffic.
- The Raspberry Pi inside the AVFS box gathers the data logs that are generated from charging events inside its folder structure.
- The Raspberry Pi in AVFS triggers data collection upon the "plug in" event. The Wireshark Box produces ethernet packets, recorded by the RPi using tcpdump.
- Bluetooth script using pyBleno and Asyncio [1] python packages is used to create a stable bluetooth connection between the Raspberry Pi and the app.
- Using Bluetooth, we send the relevant information from the log data to the app where it can be sent to our Azure server.
- Raspberry Pi is now also configured to automatically advertise as a BLE device and run our Bluetooth script around 50 seconds after boot.

Software - Overview



The implementation of the software system is composed of four stages,

- Import the raw data to Azure database.
- Perform backend computations on the cloud server.
- Identify EV chargers' health conditions and approximate charging time.
- Export results to iOS App to display.

SOFTWARE - Machine Learning

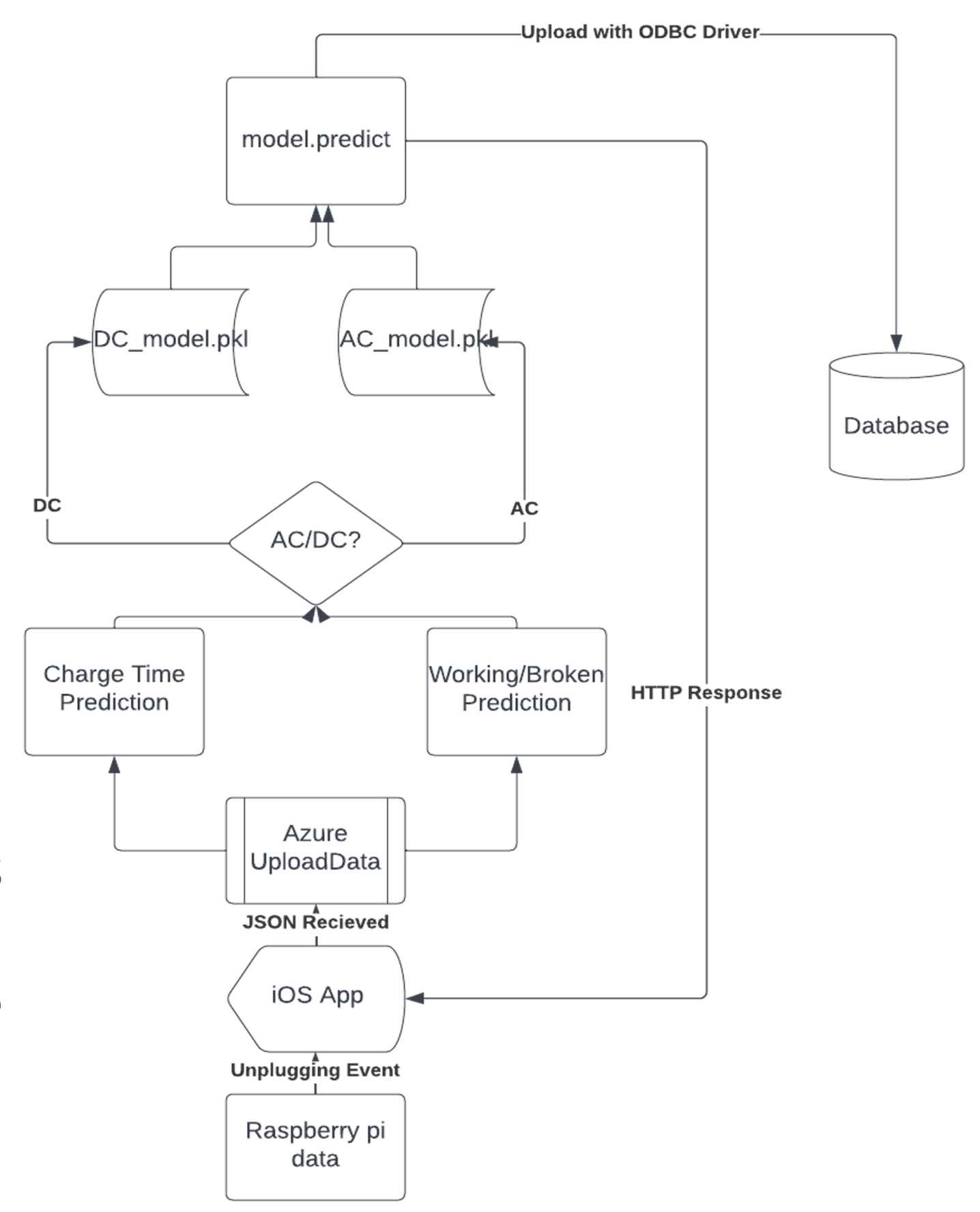
- Two machine learning models were developed. One is to identify the health conditions of the EV chargers and the other one is to approximate the charging time.
- In both models, DC and AC specific models were trained, but 'max_evse_power_kw' and 'max_evse_current_a' are not populated for AC or Level 2 charging. The relatively bad performance of AC models as shown in Table 1 below is caused by the lack of AC data.
- RandomForestClassifier() with the hyperparameter fine-tuning and L1-regularization is used to identify the health conditions of the EV chargers [2]. The same procedure is used to develop the model to approximate the charging time, but mean-squared error(MSE) is used to evaluate the performance as it is a regression problem.

Table 1. Training and Testing Statistics

	Model 1 (health conditions of EV chargers)		Model 2 (charging time approximation)	
	DC	AC	DC	AC
Training Accuracy	0.964	0.826	N/A	N/A
Testing Accuracy	0.955	0.833	N/A	N/A
Training MSE (in minutes)	N/A	N/A	5.523	3.457
Testing MSE (in minutes)	N/A	N/A	9.802	0.667

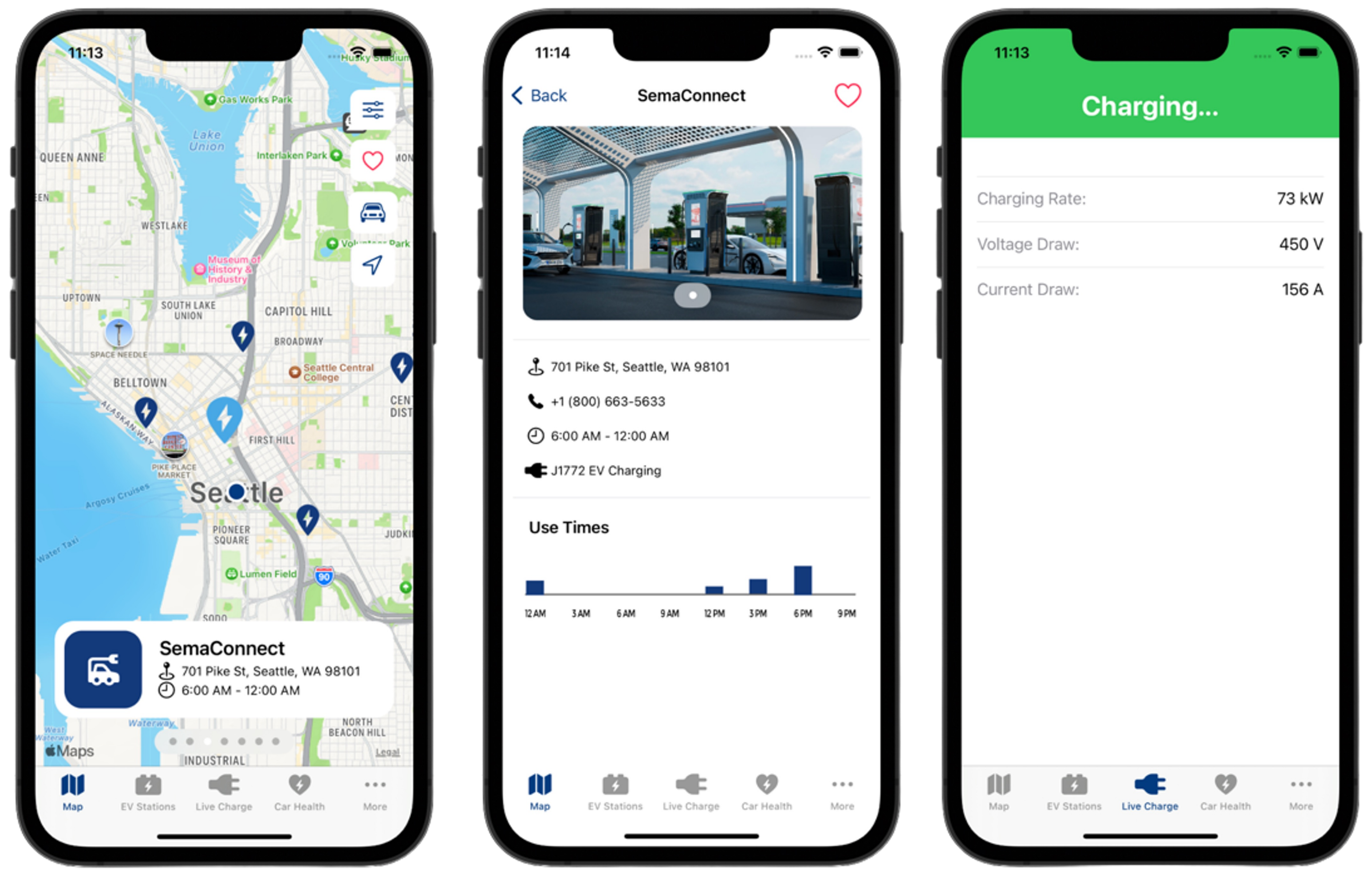
SOFTWARE - Cloud Computing

- Azure Functions is used to run event-triggered code without the need for explicit infrastructure management. The azure.functions library is used to interact with the Azure Functions runtime.
- The pyodbc Python library is used for database interactions, connecting to a SQL Server database and executing SQL queries.
- Machine learning models are incorporated into the code, stored as pkl files. These models are downloaded from Azure blob storage and loaded into the program, where they are used to predict charger status and charging time.
- Data Processing and Prediction: The code processes incoming data, makes predictions using the loaded models, and adds these predictions to the data. The data is then uploaded to the SQL database and returned to the iOS app using an HTTP request.



UI/UX - App Development

- The VoltVision app is implemented using SwiftUI framework that displays BLE and cloud based-data about the charger's health.
- The design provides a user-friendly interface that makes for easy navigation through crucial information.
- As a charger is connected, the app displays live charging updates and details, including charging rate, voltage draw, and current draw.
- For chargers on the map, the user can view additional information about the charging station, such as, hours of operation, address, compatible charger types, and usage times.



Functionalities:

- Navigate map for charging stations
- Search for a charging station
- Real-time updates of status of charging station
- Filter charging stations on map by charger type
- Live charge updates

Future Work, References, and Acknowledgments

- Look for cost-efficient alternatives for Wireshark Box and Raspberry Pi.
- Implement the latest cyber security measures to secure user data.
- Gather more data for AC chargers to improve accuracy.

Faculty: Payman Arabshahi, David Laning
Teaching Assistant: Shruti Misra
Industry Mentors: Jeffrey Hu, Parker Jones, Gavarraju Nanduri, Bob Rapp, Adam Weber

[1] S. Hymel, "Getting started with Asyncio in MicroPython (Raspberry Pi Pico)" [Accessed: 21-May-2023].
[2] S. Ram, "Mastering random forests: A comprehensive guide," Medium, 18-Oct-2020. [Online]. [Accessed: 12-Mar-2023]
[3] "Configure Development Environment for pyodbc Python Development," Microsoft, 17-Apr-2023. [Online]. [Accessed: 21-May-2023].

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISERS: GAVARRAJU NANDURI, PARKER JONES, DAVID LANING, SHRUTI MISRA

SPONSORS: ENVORSO, FORD