EZPark&Ride: Real-time Parking Availability for Park and Rides in Puget Sound

MSIM Capstone 2020 Project Report

Team EveREST

Harry Xie, Mayuresh Mali, Divya Gaurav Tripathi, Yanjie Niu

CONTENTS

EXECUTIVE SUMMARY

Image Source: https://image.cnbcfm.com/

The Puget Sound area is an urbanized and fast-growing region and a major hub of economic vibrance in the United States. The ever-growing economic activity, vast stretch of residential areas, and increasing population put a burden on the transport sector and lead to traffic congestion on the roads. 26% of the commuters* choose drive-alone to commute which adds up to this problem. Besides it adds up to the emission of greenhouse gases and fuel consumption which is detrimental to the environment. Park and Ride facilities are built on major transit routes so that people can park their vehicles and use public transport like buses or trains. However, there are major challenges that inhibit their utilization to full potential.

There is no system in place to provide the real-time availability of parking spaces in Park and Ride facilities to the commuters. Moreover, there is no electronic system in place which can gather data about the utilization of parking spaces. We, Team EveREST, took up this project intending to provide a comprehensive solution in the form of EZPark&Ride which can solve all these problems. We aimed to use cutting-edge technology (such as IoT, cloud computing, machine learning) to provide enhanced user experience for the commuters, analytical and decision-making capabilities to our sponsors, the democratization of data, reduce traffic congestion and carbon footprint; all of which would lead to a better environment and help convert the Puget Sound area into a smart city.

* Data from https://commuteseattle.com/modesplit/

OUR SPONSORS

All the stakeholders in our project (MIC, KCM, ST, SDOT, WSDOT) are committed to sustainable operations that pollute less and consume less fossil fuel. Metro has been a leader in using cleaner fuels, developing hybrid articulated buses, and adopting green operations and maintenance practices. Leveraging new cutting-edge technologies will allow us to develop an information system with real-time parking availability in Park and Ride facilities, which will eventually improve the user experience for transit users and further encourage the use of P&R facilities. We expect our solution to create a positive impact on the community.

Mobility Innovation Center (MIC)

The University of Washington and Challenge Seattle have partnered to create a new Mobility Innovation Center (MIC), to advancing our region's goal to be a global leader in building a smart city transportation system. This multidisciplinary Center brings together the region's leading experts from the business, government, and academic sectors to use technology and innovation to find transportation solutions. The Mobility Innovation Center will tackle specific transportation challenges, using applied research and experimentation. Cross-sector teams will attack regional mobility problems, develop new technologies, apply system-level thinking, and bring innovations to our regional transportation system. MIC is on a mission to build the transportation system of the future for Seattle ("Mobility innovation center | University of Washington", 2020). MIC will act as the primary liaison between our team and the various stakeholders from King County Metro (KCM), Sound Transit (ST), Seattle Department of Transportation (SDOT), and Washington State Department of Transportation (WSDOT).

King County Metro (KCM)

King County Metro (KCM) Transit Department is the public transit authority of King County, Washington, which includes the city of Seattle. It is the eighth-largest transit bus agency in the United States, carrying an average of 395,000 passengers each weekday on 215 routes. 'Metro' as it is known, employs 2,716 full-time and part-time operators and operates 1,540 buses. KCM's mission is to "Provide the best possible public transit services and improve regional mobility and quality of life in King County." ("MetroServiceDevMgrBroch", 2015) and we hope to help KCM achieve this goal with our solution.

Sound Transit (ST)

Central Puget Sound Regional Transit Authority is a public transit agency serving the Seattle metropolitan area in the U.S. state of Washington. It operates light rail service (Link light rail) in Seattle and Tacoma, regional Sounder commuter rail, and Sound Transit Express bus service, as well as managing the regional ORCA fare card system. In 2017, Sound Transit services carried a total of 47 million passengers and averaged 157,000 riders on weekdays ("2017 Q4 service delivery performance report", 2017).

Washington State Department of Transportation

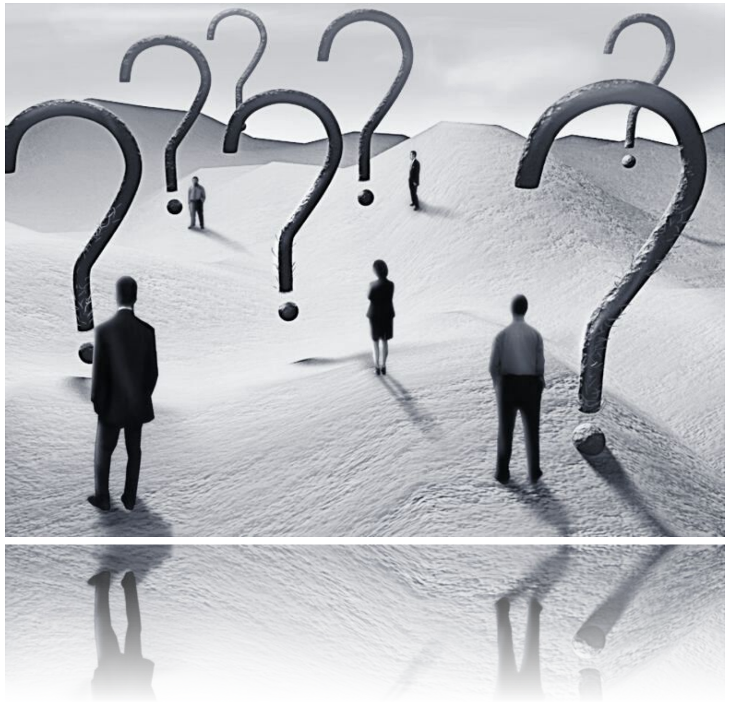
The Washington State Department of Transportation is the steward of a multimodal transportation system and responsible for ensuring that people and goods move safely and efficiently ("About us | WSDOT", 2020).

Seattle Department of Transportation

SDOT is a government agency responsible for maintaining the transportation system including roads, bridges, and public transit in the city of Seattle.

Commute Seattle

Commute Seattle is a Transportation Management Association which is supported by the Downtown Transportation Alliance (DTA) and comprised of the executive leadership of the Downtown Seattle Association, King County Metro, Seattle Department of Transportation, Sound Transit as well as rotating private-sector representation ("Commute Seattle", 2020).



INFORMATION PROBLEM

BACKGROUND

Image Source:
https://s.hdnux.com/photos/61/02/23/12854470/3/

The Park and Ride (P&R) facilities are designed to help commuters avoid stressful drives along congested roads and search expensive parking spaces, especially in the downtown core. It may well reduce traffic congestion by assisting the use of public transport in congested urban areas. As of January 2020, there are 151 P&R facilities located across King County, most are managed by King County Metro (KCM) and Sound Transit (ST), which are the two main public transport operators in the Puget Sound area.

Based on the intercept survey of P&R users conducted by the Alameda County Transportation Commission in 2017, most commuters would like to see real-time information about whether parking is available in P&R facilities and services to attract them to consider switching modes. For now, P&R does not provide real-time parking occupancy information. The lack of real-time data and travel information increases the uncertainty of their use of P&R facilities for commuting, also reduces their access to P&R facilities. For example, if users arrive at the P&R facility during the morning rush hour and find that there is no parking space available, they will need more time to wait for parking spaces to be available or to choose to park elsewhere. This will increase their commute time and disrupt their commute plans, causing the waste of fossil energy while greatly damaging the user experience. Without real-time information, users are less inclined to use Park and Rides in their travel plans through common mapping services such as Google Maps.

As the use of park-and-ride lots grows over time, it will become important to ensure there is sufficient capacity for transit riders.

P&R facility managers and operators also need to know whether parking facilities are fully utilized and who is using them. This information will give them more flexibility in selling parking permits and allocating resources. Having this information can also help them prevent illegal misuse of parking facilities.

There are 135 Park and Rides in the Puget Sound area, which aims to encourage people to take public transit and lessen traffic congestion. Although it's being used more and more, the increasing utilization brings another challenge. Many Park and Ride users complained once they arrived at a certain Park and Ride, after driving around, they found it was full. So they had to change their commute plan to drive directly to their destinations, look for other parking lots, or so forth. Sometimes, they might waste a lot of time and energy looking for available Park and Rides.

Thus, we want to provide people more information about the parking availability data before they arrive at Park and Rides.

Most of the data about parking availability we had from before were collected by manual counting on the site by staff. The frequency is every month. Thus, it is not real-time parking availability data. Also, there is some static data from the API provided by King County Metro agency, which is about the names, addresses, zip codes, total numbers of parking spaces, ADA numbers, and so forth. Our solution is based on the data we have and also collects new data provided by sensors from the new data source, to build the new data pipeline to make the real-time parking availability data access to different users.

INFORMATION GAP

An Empty Park and Ride

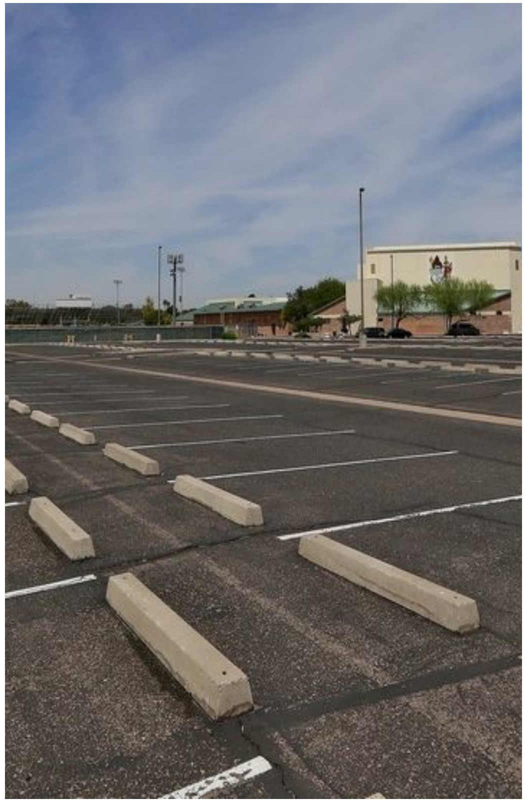


Image Source: https://www.citylab.com/transportation/2018/11/parking-lots-near-me-shopping-plazas-vacant-spaces/576646/

There are over 350 Park and Rides with nearly 60,000 parking spaces in Washington State built near public transit stops. They are built to encourage the use of more public transportation. Although they are intended to serve the community; they are not utilized to the fullest.

In the current scenario, Park and Ride do not collect data about the utilization of its facilities. Although some surveys have been conducted they have been done manually and only on a few occasions. There is no system in place to gather the data so that it can be stored or utilized. It also does not provide real-time parking availability information to the commuters.

Park and Ride still do not offer any mainstream application which is integrated with other trip-planning applications.

Moreover, Park and Ride users have a lot of uncertainty about being able to find a parking space available once they reach the Park and ride facilities. According to the survey results, most of the commuters reach Park and Ride facilities quite early in the morning, and most of the parking spaces get filled up till 8 am. A person who works in late-night shifts and starts his commute in the afternoon might not even bother to check the Park and Ride.

As most of the Park and Ride facilities occupy huge geographical areas, even after a person reaches there, they might have to spend several minutes to manually inspect the area to find a parking stall, or even to fathom out if a parking stall is available at all.

Moreover, there is no means to find out about the parking spaces available on the streets or other locations nearby to a Park & Ride facility. The commuters do not get informed of any backup options. In our survey, most of the commuters responded that they drive to the destination in their vehicle if they find their Park and Ride location is full. Because of these gaps in information, planning every day's commute is not so easy for the commuters.

All of these cause the Park and Ride to fill at unpredictable times and discourage commuters from utilizing it. Our product caters to these problems.

Park and Ride facility managers and operators also need to know whether parking facilities are fully utilized and who is using them. They can find this through the analysis of historical data. This information will give them more flexibility in selling parking permits and allocating resources. Having this information can also help them prevent illegal misuse of parking facilities.

Image Source: https://keydifferences.com/difference-between-data-and-information.html

AROUND THE WORLD

As part of our research, we reviewed existing studies and implementations of real-time parking space available within the USA and also at a global level. We found there were a few field studies that demonstrate technology for providing information in real-time on vehicle parking availability ("SmartPark: Real-Time Parking Availability | FMCSA", 2014). LA Metro is pilot testing a similar project on its L Line (Gold) Rail ("Real-Time Parking Availability", 2020). We also found that the City of Santa Barbara has implemented a similar solution but the underlying tools and techniques used are still unknown ("Santa Barbara - Real-Time Parking Availability", 2020). The installations at the parking lots in Santa Barbara appear to have a basic car counting using sensors at the ingress and egress points as most of these lots seem to have only a single entry/exit point. To take a cue from a global perspective, Singapore has implemented a FASTER system as part of its smart nation initiative (Blandin, Wynter, Poonawala, Laguna & Dura, 2020). This system analyses data collected from various sources to help authorities and operators visualize commuting patterns to improve transport planning. Our solution is aiming to move towards such a digital transformation.

Image Source: https://scandasia.com/research-2019-reasons-why-singapore-is-the-best-country-for-scandinavian-people/

Image Source: https://www.youtube.com/watch?v=IFBGerF--IU

Image Source:
https://www.secureidnews.com/news-item/singapore-digital-id-key-new-national-smart-nations-plan/

OUR APPROACH

PROJECT PLANNING

Image Source: https://www.seattletimes.com/

Project Planning Phase - January 2020 to March 2020

In the initial planning phase, our team spent a considerable amount of time conducting a literature review of the existing solutions and research on parking availability and developed an understanding of the problem space.

- Current Situation Analysis: We conducted research and analyzed how regional Park & Ride facilities of Seattle are owned, managed, and equipped. We used that information to monitor and model a parking availability information system.

- Smart Park & Ride Service Design and Development: Based on data collection, analysis, and modeling, we devised data-driven recommendations, i.e., a real-time solution for communicating parking availability to the traveling public at a pilot location.

Project Execution Phase - April 2020 to May 2020

In phase 2, which was the execution phase, we identified the technology (cloud, IoT, machine learning), hardware and software assets and started working on user research and development of the front-end and back-end systems. The user research process included designing the survey and interview questions, distribution of the survey, and finally the data analysis and visualization of the survey results. We also developed the data pipelines on the cloud platform and integrated it with the hardware assets such as Bosch parking sensors and MultiTech gateways.

USER RESEARCH DESIGN

Our user research design explores various types of users like daily commuters going for work, people working in unusual shifts, parents dropping their children to school, carpoolers, and other commuters from various income groups.

Our solution aims to provide diverse groups with equitable access to Park and Ride facilities.

Coronavirus brought challenges to our user research, but we value users. So instead of canceling user research, we came up with creative ideas to conduct remote UX research.

We spent a lot of time iteratively improving survey design together with our sponsors because we care about the equity impact of our survey. Also, we brainstormed different ways to screen and recruit respondents. Thanks to all of our sponsors to help distribute surveys, because of which, we received 686 survey results for analysis within a week only.

Image Source:
https://seattlegood.org/listings/commute-seattle/

The other type of user is the staff and facility managers in the transit agents, like King County Metro and Sound Transit.

We interviewed their experts to understand their needs. Now, they are using Power BI dashboards, or Microsoft Excel to process the utilization data and visualize parking availability data. All the data were manually collected by on-site staff of the management to count monthly or in half-hour intervals in certain pilot lots.

Our solution also provides decision making and analytical capabilities to the staff who can use Power BI and Azure cloud to automatically process and visualize the collected data.

We considered scalability into our product design and implementation.

In the whole process of UX research design, we tried to help diverse users address their information challenges and provide user-centered and scalable actionable solutions.

CLOUD COMPUTING

Image Source: https://www.azure.com

Image Source: https://aws.amazon.com

Image Source: https://cloud.google.com

Our Cloud Solution

Today, cloud computing applications and platforms are rapidly growing across all industries, serving all kinds of businesses and services to improve the user experience and efficiency by achieving digital transformation without making a huge investment in IT infrastructure. Cloud computing enables access to various computing resources over the internet. It has many benefits, including offering accessible storage with data security measures in place, and scalability all at once.

As a pilot project, we are dedicated to finding an efficient way to implement EZPark&Ride that supports all the features as we designed and has the least requirement on technical expertise for our stakeholders. Also, we think the implementation design has to be scalable because this pilot project will ultimately be implemented on more Park and Ride facilities, which means the computing resource demand can go much higher than the needs of sustaining the pilot system. Meanwhile, we want to fully utilize the benefits of using cloud platforms to reduce the operation cost by adopting cloud features such as serverless computing.

Azure

Azure is a cloud computing platform that comes with an online portal that allows you to access and manage computing services and resources provided by Microsoft.

The minimum viable product (MVP) of EZPark&Ride needs to demonstrate real-time data ingestion and processing pipeline for parking availability data from IoT devices installed at Park and Ride facilities. Our goals include:

- ingest and store data in real-time from parking sensors
- analyze parking availability data to understand the utilization of Park and Ride facilities, and make predictions of future utilization
- provide open data to end-users with real-time parking availability data
- store the data after analysis for other downstream processing (e.g., business intelligence) to provide actionable insights from the historical data

Other Cloud Computing Platforms & Hybrid Cloud Solution

We acknowledge that different cloud computing platforms have their advantages, so we want this system design should have the potential to be deployed on different cloud computing platforms, and even to operate cross-platform as a hybrid cloud computing solution.

HARDWARE ASSETS

Image Source: https://www.bosch-connectivity.com/use-cases/everything-under-control-thanks-to-networked-parking-space-sensors/

IoT sensors

We are using IoT LoraWAN sensors provided by Bosch to collect the parking availability data. The Parking Lot Sensor (PLS) detects and reports parking space occupancy, thus enabling active parking lot management features, such as search, navigation, and reservation. The easy retrofit sensor solution for off-street parking is installed in minutes. It utilizes the discretionary LoRaWAN protocol for wireless communication and can, therefore, be integrated easily into any Smart City project ("Parking Lot Sensor | PLS", 2020).

The sensors from Bosch have the following advantages:

- **Zero maintenance and calibration:** No maintenance by personnel on site is necessary. This saves you time and money.
- **Easy installation:** It only takes minutes from sensor installation to receiving the first data. The sensor is glued to different surfaces or screwed in the ground with ease.
- **LoRAWAN compatible:** Long-range - multiple kilometers, Low power - can last years on a battery, Secure - 128 bit end-to-end encrypted, Low bandwidth and Cost requirements.
- **Two independent sensor principles:** magnetometer and radar. If one of the other mechanisms fails the other acts as a fallback.

Features of the PLS 110 sensor

- **Self-learning algorithm for parking space occupancy detection:**
  The wireless sensor detects and reports parking space occupancy, thus enabling active parking lots management features, such as search, navigation, and reservation.
- **High protection class: IP67**
  The elegant but extremely robust IP67 housing can withstand any external influence.
  Long battery life:
  Inside, a Lithium battery ensures a lifetime of up to 5 years.
- ***96% average parking state change detection performance:***
  Proven in field-tests with more than 2.000 sensors and more than 46 different car types in real parking environments.
- **LoRaWAN certified sensor:**
  PLS utilizes the discretionary LoRaWAN protocol for wireless communication and can, therefore, be integrated easily into any Smart City project.

HARDWARE ASSETS

Gateway

Our sponsors need a technical solution that is affordable, feasible, reliable, maintenance-free and can be deployed quickly. Hence we decided to go with IoT sensors over other technologies as it met all the aforementioned requirements. We have tried integrating the sensors from Bosch with the Multitech Conduit gateway through a LORIOT network server that offloads the data payload to Azure IoT Hub.

The Multitech gateway is IP67 rated as well and is built for outdoor installations. The gateway can survive harshest environmental factors including moisture, dust, wind, rain, snow and extreme heat. We found the gateway to be highly robust and most suitable for installations at any kind of open or closed Park and Ride locations. The Conduit® is the industry's most configurable, manageable, and scalable LoRa gateway for industrial IoT applications. The Conduit features Wi-Fi/Bluetooth/Bluetooth Low Energy (BT/BLE), GNSS and two accessory card slots that enable users to plug in MultiTech mCard accessory cards supporting their preferred wired or wireless interface to connect to a wide range of assets locally to the gateway ("MultiTech Conduit IP67 LoRa Gateway | Outdoor LoRa Deployment", n.d.). With a cellular link, there is no need for any invasive cabling apart from the PoE Ethernet cable to supply power to the gateway.

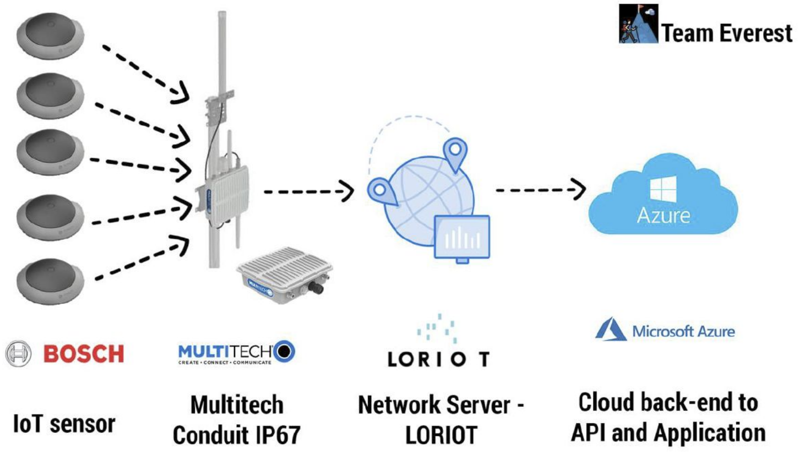
Sensor to Cloud architecture

- Azure IoT Hub is a scalable, multi-tenant cloud platform (IoT PaaS) that includes an IoT device registry, data storage, and security. It also provides a service interface to support IoT application development.

- The Router function is triggered by messages coming from the IoT Hub (connection defined in the EVENTHUB_ROUTER_INPUT environment variable) and routes them to the appropriate decoder.

- Decoder Functions perform the decoding of the sensor raw data payload.

- LORIOT Lifecycle Function-> This function exists as a time-triggered function to ensure regular syncing of new devices. ("loriot/AzureSolutionTemplate", n.d.)



OTHER SOLUTIONS

Image Source: https://www.bosch-connectivity.com/use-cases/everything-under-control-thanks-to-networked-parking-space-sensors/

We did some research on other solutions which can be summarized as follows:

Smart Parking

The common method used to find a parking space is manual as discussed by Geng and Cassandras (2012). If drivers had access to a database containing information about parking spots in real-time, there would be more opportunities for selecting an appropriate route to the desired parking slot.

Long-range Autonomous Valet Parking (LAVP) presents a novel approach, making an autonomous vehicle a temporary park in a car park within an urban area.

Khalid, Cao, Zhang, and Peng (2018) state that the Scheduling Center is the core, as it coordinates the nearest route to a drop-off place and finds the best car park. This approach will benefit users by providing a reservation system in a cost-effective manner. Other approaches (Thomas, etc., 2018) suggest that customers use an Android application to reserve a parking place in advance.

Parking Availability Prediction System

Two different types of predictions are provided: the probability of free space to continue being free in subsequent time intervals, and the short-term parking occupancy prediction in selected regions of an urban road network. The available data come from a wide network of on-street parking sensors in the "smart" city of Santander, Spain.

The sensor network is segmented in four different regions, and then survival and neural network models are developed for each region separately. Findings show that the Weibull parametric models best describe the probability of a parking space to continue to be free in the forthcoming time intervals. Moreover, a simple genetically optimized multilayer perceptron accurately predicts region parking occupancy rates up to 30 minutes in the future by exploiting 1-minute data. Finally, the real-time, Web-based, implementation of the proposed parking prediction availability system is presented.

Other Off-the-shelf Solutions

In our interviews with different people implementing parking solutions, we heard about the state of the art camera-based smart-sensing system that employs Edge Analytics and uses sophisticated software and machine learning algorithms to detect parking spaces.

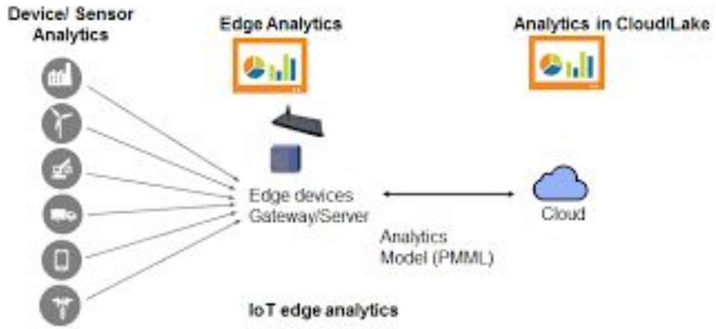


Image Source: http://ashishmahajancrm.blogspot.com/2019/03/introduction-to-edge-analytics.html

RESEARCH ON OTHER SOLUTIONS

Such type of video sensors may come in two flavors, one which gives a basic count of the number of vehicles that enter or exit at the ingress and egress points. Such a system is very basic and maybe useful at multi-level Park and Ride locations such as the North Gate Parking Garage. A few limitations of this kind of system is lower accuracy and granularity. We would not know which parking stall is empty or occupied and just get a high-level overview of the count.

The second system, which involves the use of powerful Smart-sensors with cameras is more multi-purpose in the sense that it could be used for LPR (License Plate recognition) or surveillance apart from just providing parking occupancy information.

Even with a network interruption, it will continue to detect the occupancy of spots because such a system puts processing intelligence right at the device level. This is called edge analytics where the data collection and analysis are done right at the sensor and only statistical or metadata is passed on to the network server. This approach has multiple advantages in terms of privacy as no PII (Personally identifiable information) or video footage is stored in the cloud so chances of third party security risk are minimized. With deep learning and the advancement of ML techniques in the field of video recognition coupled with advanced hardware such as the LiDAR (Light Detection and Ranging) technology, there is no actual need to store raw video footage for analysis.

Other Findings From Interviews

We had been told that these technologies have a huge potential for off the shelf solutions- City Tech(MIC and Microsoft collaboration), Intel Open Vino, Microsoft Aware, Microsoft Synapse software for data manipulation. Please also use IOT Central instead of IoT Hub for an Azure-based solution.

Please check about the Chicago Millennium parking garage which has 4 connected underground infrastructure having 3.5 million parking spaces.
Having said that, such video systems may perform poorly or fail altogether in extreme weather conditions such as snow where it would become difficult to differentiate a car from other snow-covered objects. Also if the camera sensor is blocked or tampered with the resulting information will be inaccurate. Lastly, without a proper line of sight, the system may misclassify a parking spot as either vacant or occupied. So a hybrid solution having IoT sensors and off the shelf video analytics products would work the best for a comprehensive parking solution

PROJECT OUTPUT

USER RESEARCH

We used Tableau to visualize the data we collected through surveys and analyzed them. The data visualization and analysis of each survey question are on pages 37 - 42.

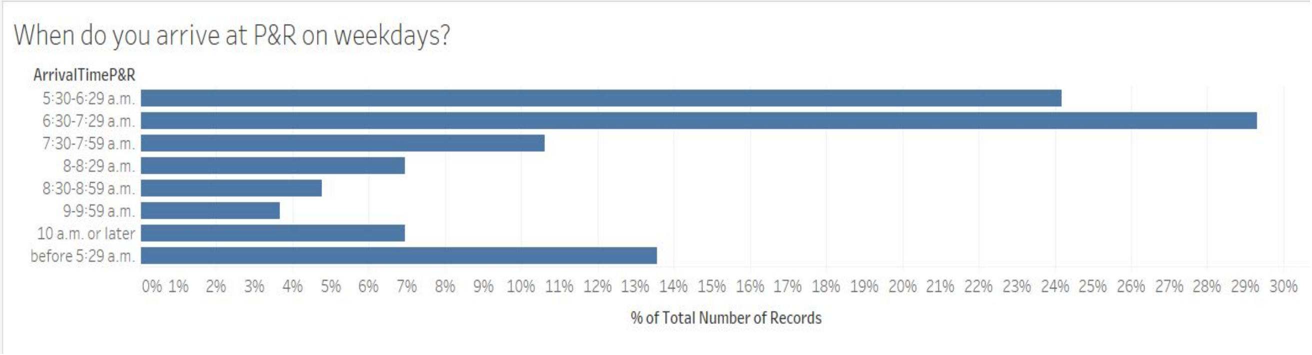
Through data visualization, we find there is a lot of rich information that we could dig into. Among them, some significant insights are listed as follows.

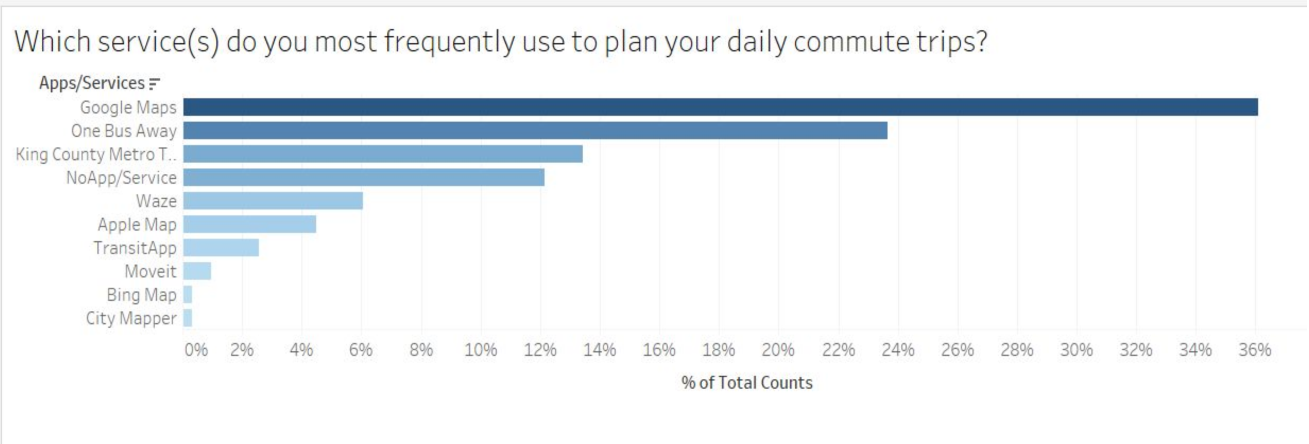
1) We find a lot of respondents arrive at Park & Ride at 6:30 – 7:30 am on weekdays.

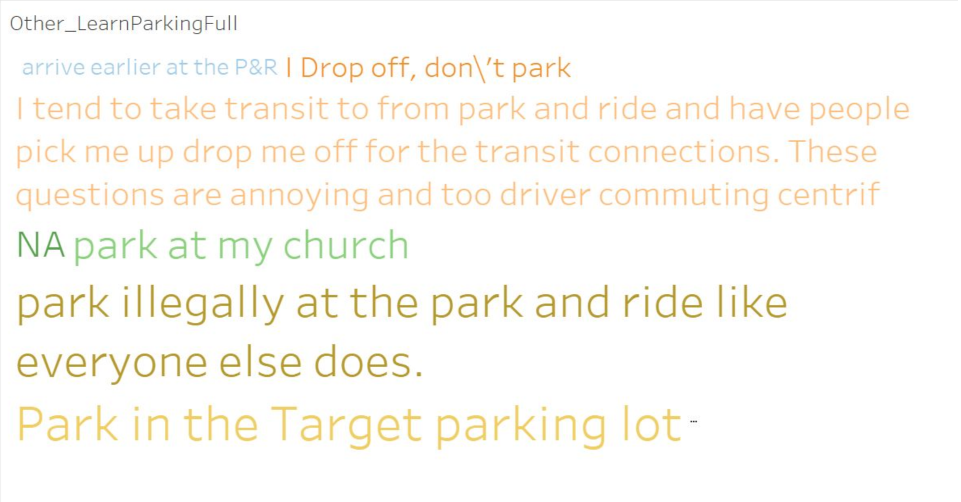
2) Right now, a large number of respondents do not use apps or websites to get information about parking availability.

3) However, if on the days when they were not able to find space at their preferred Park and Ride, they have to drive to the final destination, or park nearby, or go back home to work.

Thus, considering a lot of them declare that they usually do commute plans in advance, we think our solution could meet their information needs to make a commute plan efficiently.






Other_LearnParkingFull

arrive earlier at the P&R | Drop off, don\'t park

I tend to take transit to from park and ride and have people pick me up drop me off for the transit connections. These questions are annoying and too driver commuting centrif

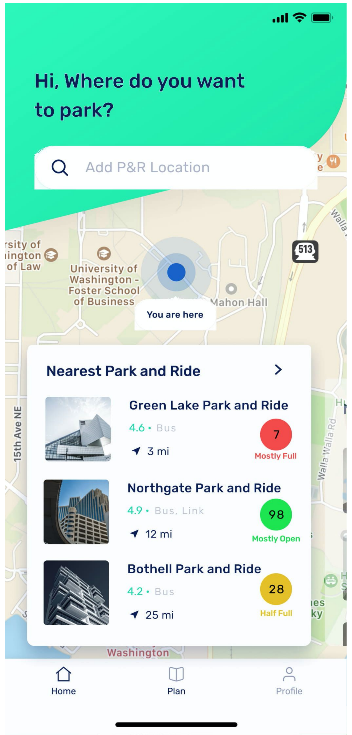
NA park at my church

park illegally at the park and ride like everyone else does.

Park in the Target parking lot

MOBILE APPLICATION

We designed a clickable prototype for EZPark&Ride mobile application. The program is designed to help users find recommended Park and Ride locations by entering their destination and uses real-time parking availability data to help them plan their trip. To understand the user's preferences and needs, we combined the analysis of existing map services and parking software and designed three different types of data display methods for the main interface.



Hi, Where do you want to park?

Add P&R Location

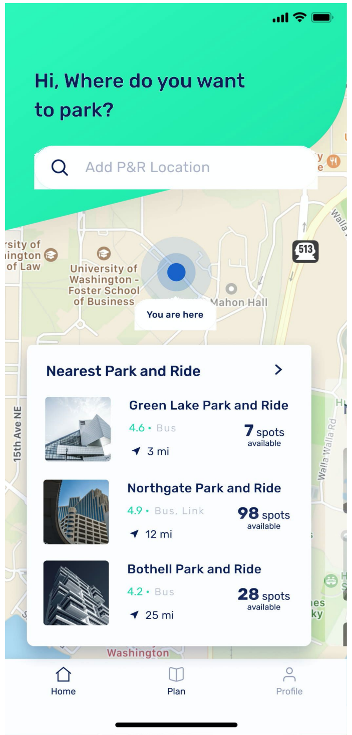
You are here

Nearest Park and Ride >

Green Lake Park and Ride
4.6 • Bus
3 mi
7
Mostly Full

Northgate Park and Ride
4.9 • Bus, Link
12 mi
98
Mostly Open

Bothell Park and Ride
4.2 • Bus
25 mi
28
Half Full



Hi, Where do you want to park?

Add P&R Location

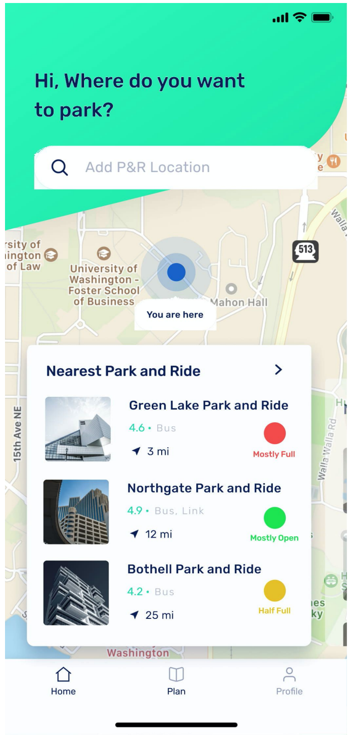
You are here

Nearest Park and Ride >

Green Lake Park and Ride
4.6 • Bus
3 mi
7 spots available

Northgate Park and Ride
4.9 • Bus, Link
12 mi
98 spots available

Bothell Park and Ride
4.2 • Bus
25 mi
28 spots available



Hi, Where do you want to park?

Add P&R Location

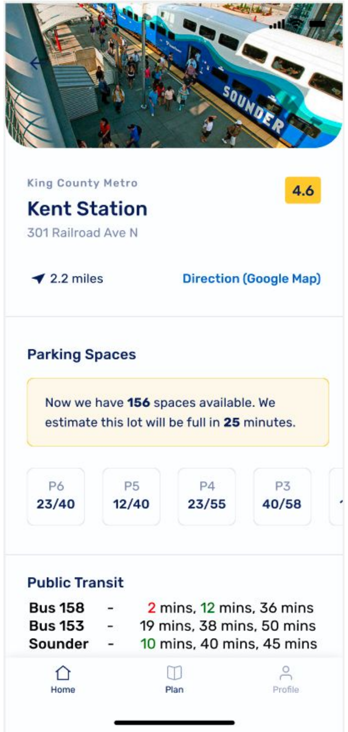
You are here

Nearest Park and Ride >

Green Lake Park and Ride
4.6 • Bus
3 mi
Mostly Full

Northgate Park and Ride
4.9 • Bus, Link
12 mi
Mostly Open

Bothell Park and Ride
4.2 • Bus
25 mi
Half Full



King County Metro

Kent Station

301 Railroad Ave N

4.6

2.2 miles

Direction (Google Map)

Parking Spaces

Now we have **156** spaces available. We estimate this lot will be full in **25** minutes.

P6	P5	P4	P3
23/40	12/40	23/55	40/58

Public Transit

Bus 158	-	2 mins, 12 mins, 36 mins
Bus 153	-	19 mins, 38 mins, 50 mins
Sounder	-	10 mins, 40 mins, 45 mins

In the first design, the parking availability numbers will be shown in combination with a three-color scheme design.

In the second design, the main interface will only provide parking availability numbers, without any supporting visual design.

In the third design, the main interface will not display parking availability numbers. The user will use three color cues to learn the availability of parking.

In addition to the main screen, users can click through to the details screen of each Park and Ride location to get more information. This application allows us to complete our entire vision of the EZPark&Ride design. For example, users can view the availability of parking spaces down to the exact details of each space on the details page. The user can also see predictive data on the page of each Park and Ride location to plan the trip. Besides, the mobile app is integrated with real-time bus information from the GTFS feed, which will allow users to plan the entire commute including the use of the Park and Ride service.

WEB APPLICATION

Web development is based on user experience research.

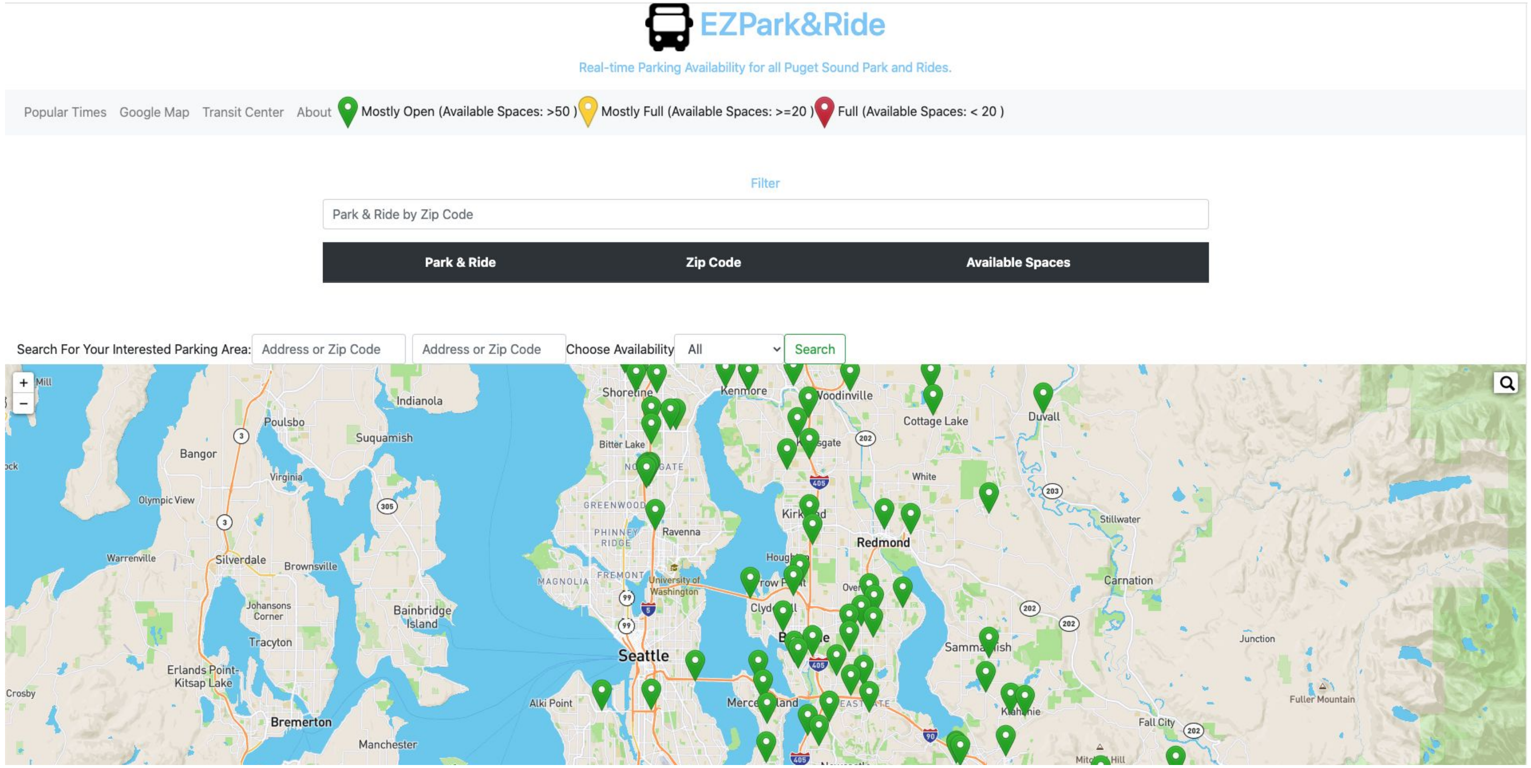
The map visualizes the real-time parking data dynamically.

It groups parking information into three colors.
Specifically, the green icon labels the parking lot with the available spaces more than 50, meaning the lot is mostly open. Red icon labels the parking lot with the available spaces less than 20, meaning the lot is full. Yellow icon labels the parking lot with the available spaces in between, meaning the lot is mostly full.

The web application also has a search feature for users to search for the Park and Rides near the zip codes or addresses they hope to go to.

Besides, through the "popular times" button in the navigation bar, users could access prediction data about certain Park and Ride, which is gained through the Power BI machine learning tool.

The application is designed and developed to make users have an enjoyable experience to check the real-time parking availability information.



EZPark&Ride

Real-time Parking Availability for all Puget Sound Park and Rides.

Mostly Open (Available Spaces: >50) Mostly Full (Available Spaces: >=20) Full (Available Spaces: < 20)

Filter

Park & Ride by Zip Code

Park & Ride	Zip Code	Available Spaces

Search For Your Interested Parking Area: Address or Zip Code | Address or Zip Code | Choose Availability: All

Screenshot of the web application

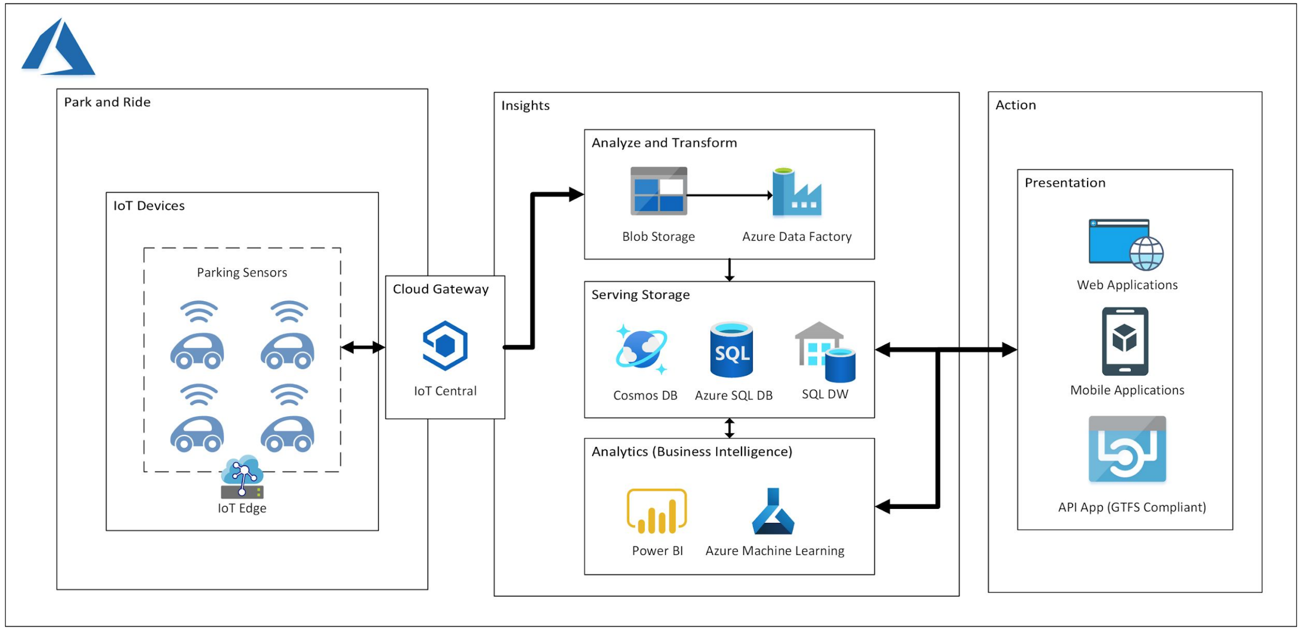
BACKEND

Overview

We have successfully deployed the back-end system we are designing on Azure. Because of the COVID situation, we were not able to deploy the sensors in the planned parking lot, so the real-time parking availability data that the current system used is from a simulated data feed which implemented on mockaroo.com. The simulated data feed provides randomly generated data on parking availabilities of four selected Park and Ride locations - Green Lake Park and Ride, Northgate Park and Ride, Bothell Park and Ride, and Eastgate Park and Ride. As of June 8th, the data pipelines in the back-end system have been successfully run over 100 times.

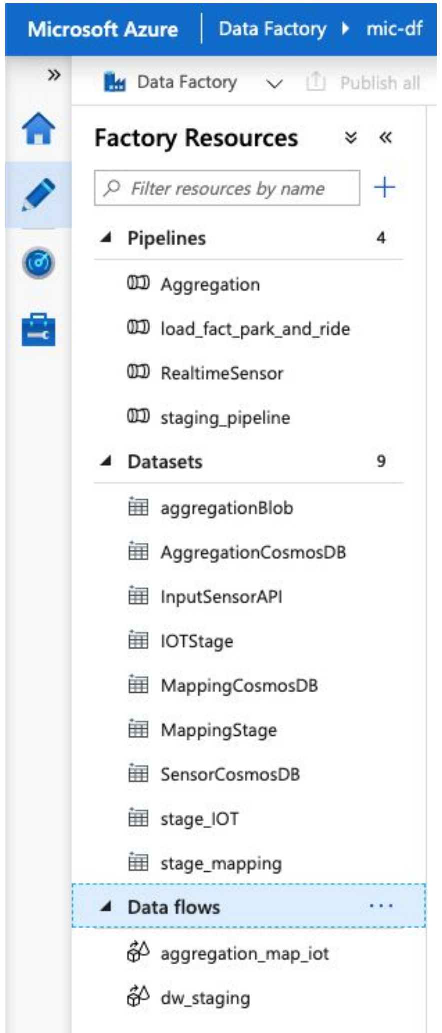
Cloud Architecture (Azure)

In this pilot project, there are three major components of the cloud system in our design. First is the IoT, EZPark&Ride uses Bosch sensor that can communicate with the cloud platform via gateways that have a stable internet connection. We expect hundreds of messages to be generated by the IoT sensors every hour at a small-sized Park and Ride facility. The gateway which is pre-registered with a group of sensors will send all those data to the corresponding cloud endpoint. The cloud endpoint is provided and managed by **Azure IoT Central**, a centralized IoT device management platform that helps to connect all IoT devices to the cloud platform. Azure IoT Central provided solutions that support bidirectional communication between the sensors and the cloud as a part of the ingestion layer. After the configuration, users will be able to see the status of all registered IoT devices and manage them by sending actions to the devices from Azure IoT Central.



The second major component is data processing. This section consists of a Cosmos DB database, a SQL Server Database, a storage account (blob), Azure Data Factory, and Azure Machine Learning. The original data that comes from the IoT Central will be stored in the Blob Storage untouched. This is to ensure the ability to restore in the future stage of data processing. Azure Data Factory is Azure's cloud ETL (Extract, Transformation, Loading) service for scale-out serverless data integration and data transformation. It offers a user-friendly UI that supports code-free authoring and monitoring and management of data pipelines. In the MVP, three pipelines are running in Azure Data Factory. The "RealTimeSensor" pipelines will fetch data from the blob storage to get the real-time sensor data. "Stage_Pipeline" and "Load_Fact" are loading the data from the blob storage to the data warehouse regularly. "Aggregation" uses the sensor data to get the total number of available parking spaces and transforms it into a specific format that can be consumed by the front-end application. The aggregated data will be stored in one Cosmos DB collection named "Total".

DATA PIPELINE



Azure Data Factory

Image Source: https://www.azure.com

Azure Cosmos DB

Image Source: https://www.azure.com

Azure Machine Learning

Image Source: https://www.azure.com

We use Azure Data Factory because it integrates deeply with other Azure databases and storage services such as Cosmos DB, MS SQL Server, and Blob Storage. It supports autoscaling and auto-termination by provisioning/deprovisioning computing clusters to minimize costs automatically. The computing resources will only be provisioned ad hoc, which means the user will only be charged for consumptions of the actively running data flow. Azure Cosmos DB is a multi-model database that is available in multiple regions globally. IoT solutions usually require sophisticated capabilities to collect and process data from IoT devices. Since we believe that a hybrid version of IoT solution could be a more reliable way if our stakeholders want to implement the system on all Park and Ride facilities, which means the system has to have to ability work with multiple data providers using different and sometimes unpredictable data schemas. In this case, NoSQL instead of a relational database is the more suitable choice. A relational database such as MS SQL Server and MySQL requires schemas to be defined before data ingestion, while NoSQL databases such as Cosmos DB and MongoDB don't. NoSQL allows de-normalized documents, regardless of the structure of the data, to be ingested into existing databases quicker and easier.

Also, in our design, every time the state of the parking sensor changes, a unit of event data is sent from the sensor to the cloud computing platform. This means that when a large number of users are entering or leaving a Park and Ride facility in a short period, a large volume of data will be transmitted to the cloud waiting to be processed. We expect that this demand will greatly increase the demand for computing power for data processing during peak commuting hours. NoSQL also has the advantage of scaling when it comes to processing large amounts of IoT data, which can be done simply and without performance bottlenecks. Cosmos DB provides guaranteed speed and business continuity (99.999% availability) at any scale, with the support of fast and flexible app development, and enterprise-level data security.

In the data analytics layer, we use Azure Machine Learning to implement the machine learning model Azure Data Factory will send historical data weekly to the Azure Machine Learning for model training. The model will provide time series forecasting data through its API, and eventually will be used integrated with the data feed to be consumed by end-users.

The third component of the system is Action. It includes a web host for the front-end application and App Service that used to provide open data to any users.

ALTERNATIVE

Image Source: https://www.azure.com

Image Source: https://www.azure.com

Azure Databricks
In the future, as the amount of data processed increases, we recommend replacing Azure Data Factory with Azure Databricks. According to Microsoft: Azure Databricks integrates deeply with Azure databases and stores Azure Synapse, Azure Cosmos DB, Azure Data Lake Storage, and Azure Blob Storage. It also has the following features that work better than Azure Data Factory:

- Autoscaling and auto-termination for Spark clusters to automatically minimize costs.
- Performance optimizations including caching, indexing, and advanced query optimization, which can improve performance by as much as 10-100x over traditional Apache Spark deployments in cloud or on-premises environments.
- Integration with Azure Active Directory enables you to run complete Azure-based solutions using Azure Databricks.
- Role-based access in Azure Databricks enables fine-grained user permissions for notebooks, clusters, jobs, and data.
- Comes with Enterprise-grade SLAs.

Azure Synapse
We consider Azure Synapse as a great alternative to replacing the Data Warehouse (currently implemented on SQL Server). Here are the reasons:

- It works well with other Microsoft services such as Azure Machine Learning, and Power BI.
- Azure Synapse allows anybody to work with data in those disparate places to manage and analyze it from within a single service. It can be used to analyze relational and unstructured data, using standard SQL.
- Azure Synapse Link for Azure Cosmos DB is a cloud-native hybrid transactional and analytical processing (HTAP) capability that enables you to run near real-time analytics over operational data in Azure Cosmos DB. Azure Synapse Link creates a tight seamless integration between Azure Cosmos DB and Azure Synapse Analytics, which means we no longer need a data warehouse and its ETLs. Instead, the Synapse Link will automatically create a fully-isolated column on the Cosmos DB collection to run near real-time business intelligence, and machine learning pipelines.

ALTERNATIVE

Cloud Architecture (AWS)
Using AWS as an example, all the services used in our Azure cloud architecture can find their counterparts of the same type of service on AWS.
For example, the Azure IoT Central can be replaced by AWS IoT. The Blob storage can be replaced by the AWS S3 storage service. The SQL Server (Data Warehouse) can be replaced by AWS DynamoDB.

Early in the project execution phase, we chose to build a MongoDB on AWS's EC2 to house all the sensor data. The reason for this is that a self-configured database service provides more autonomy and is better suited to the operation and management of our initial The need for exploration. Subsequently, while we deployed this system on Microsoft Azure, this AWS on the MongoDB is still up and running and transferring data with other Azure services. The services we use on Microsoft Azure are compatible with the non-Azure services such as AWS DynamoDB, which means a hybrid cloud computing solution is completely viable.

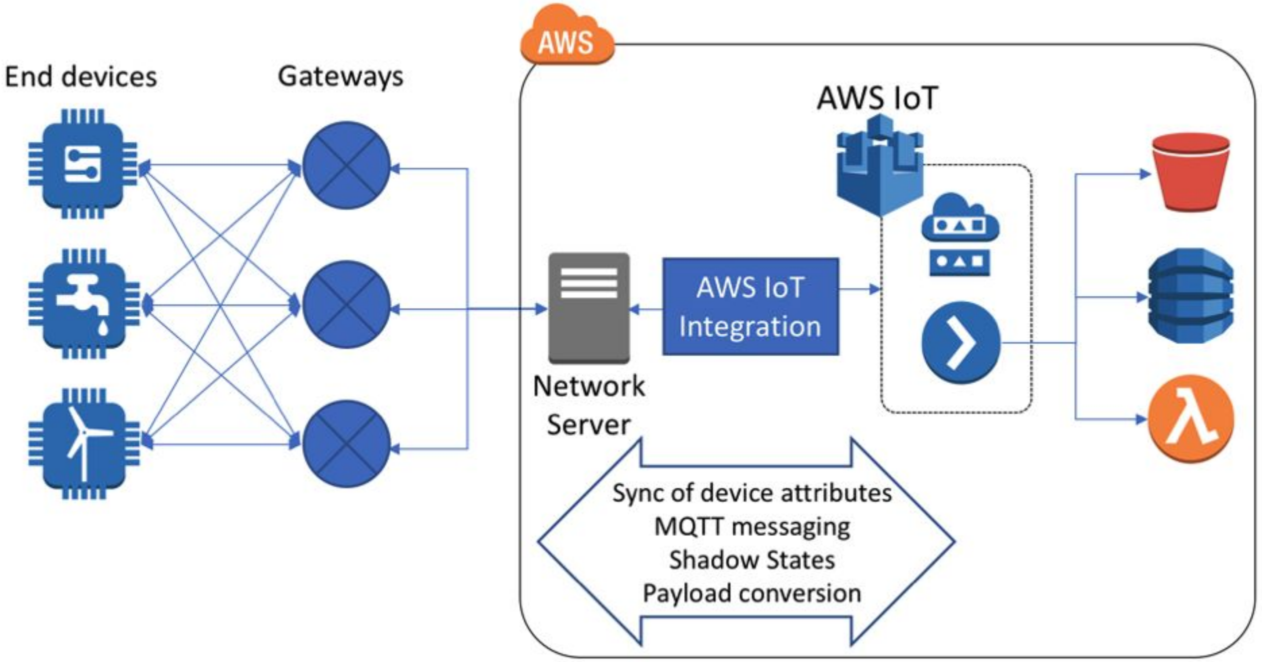
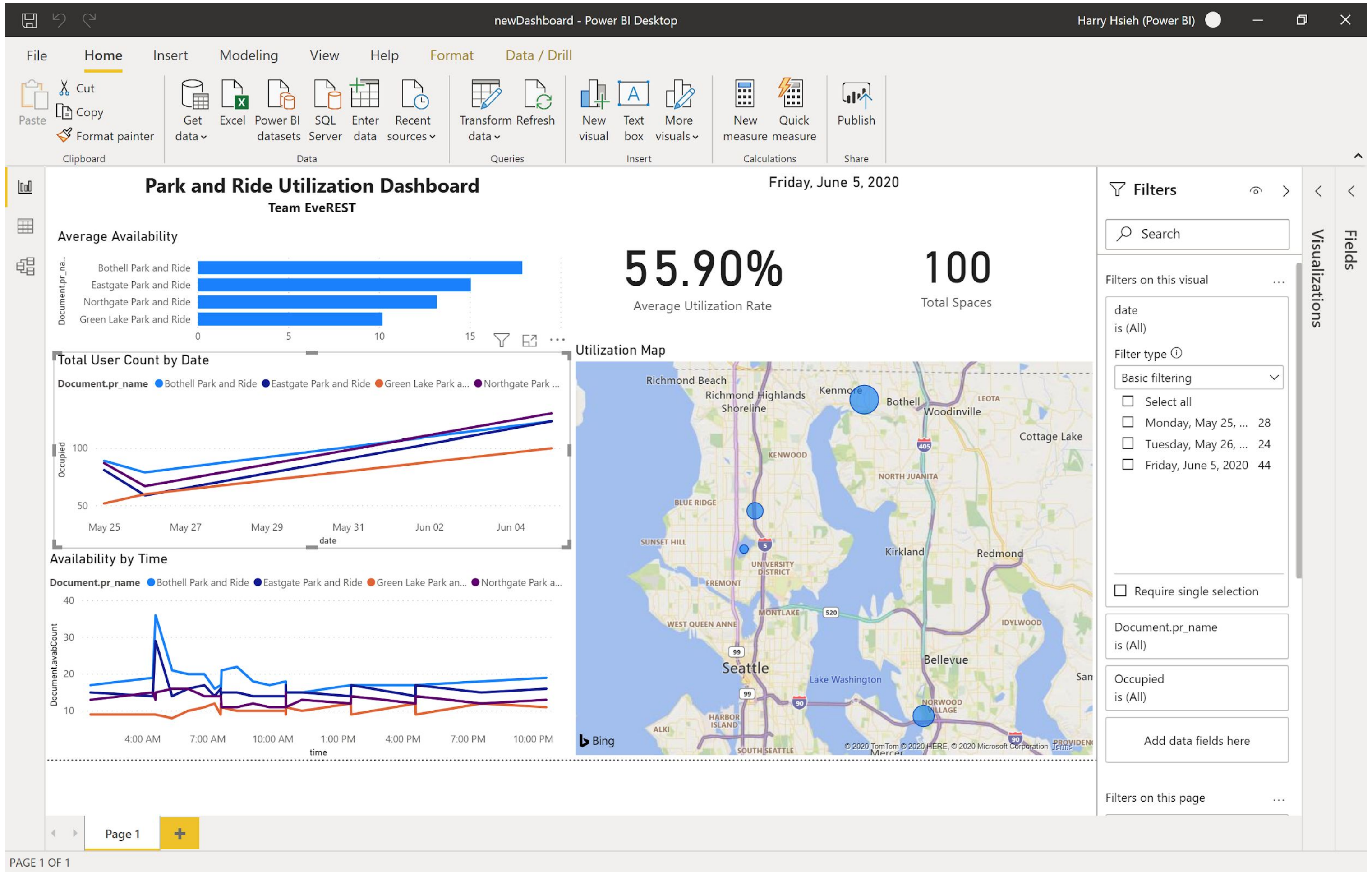


Image Source: https://aws.amazon.com/blogs/iot/connect-your-devices-to-aws-iot-using-lorawan/

BUSINESS INTELLIGENCE

Business intelligence analysis is a significant function of EZPark&Ride's system. Access to usable intelligence and accurate data is an important reason why public transit service providers choose to adopt parking sensor solutions. Our system is designed to enable near real-time business intelligence analysis capabilities by transforming and transmitting data to the data warehouse in real-time. Given that both King County Metro and Sound Transit use Microsoft services, such as Sharepoint and Power BI, to some extent, we chose to implement business intelligence analysis with Power BI as the application. At the same time, because the data is already in a well-transformed format on the back-end, any data analysis applications such as Tableau should be able to process the data for visualization. The user only needs to use SQL Server (data warehouse) as a data source in those applications to start analyzing historical parking data. Here is an example dashboard that we created in Power BI using the mockup data.



Filters on this visual

date
is (All)

Filter type
Basic filtering

- ☐ Select all
- ☐ Monday, May 25, ... 28
- ☐ Tuesday, May 26, ... 24
- ☐ Friday, June 5, 2020 44

☐ Require single selection

Document.pr_name
is (All)

Occupied
is (All)

Add data fields here

Filters on this page

COST ANALYSIS

Image Source: https://www.bosch-connectivity.com/use-cases/everything-under-control-thanks-to-networked-parking-space-sensors/

Example Analysis for

A. North Seattle P&R

- 102 parking stalls
- 1 Bosch parking sensor for each stall
- usually full by 8 a.m. on weekdays

Total

Fixed Asset - 102*$150(Sensors) + 2*$1341(Gateway) + $500(Miscellaneous)= $18,482

Monthly - $1,775 / month

B. Eastgate Park & Ride

- 1614 parking stalls
- 1 Bosch parking sensor for each stall
- usually filled 90% or above by 9 a.m. on weekdays

Total

Fixed Asset - 1614*$150(Sensors) + 20*$1341(Gateway) + $5000(Miscellaneous)= $273,380

Monthly - $5,253 / month

Item	Price (USD)	Note
Bosch Sensor	$150/each	
MultiTech IP67 Gateway	$1341/each	
Miscellaneous Items	$500	
Loriot Server	$175/month	Single Location License
Internet Access	Variable	
Electricity	Variable	
Cloud Computing (Azure)	$1600/month	Estimated for 100 sensors, single location, Pay-As-You-Go plan

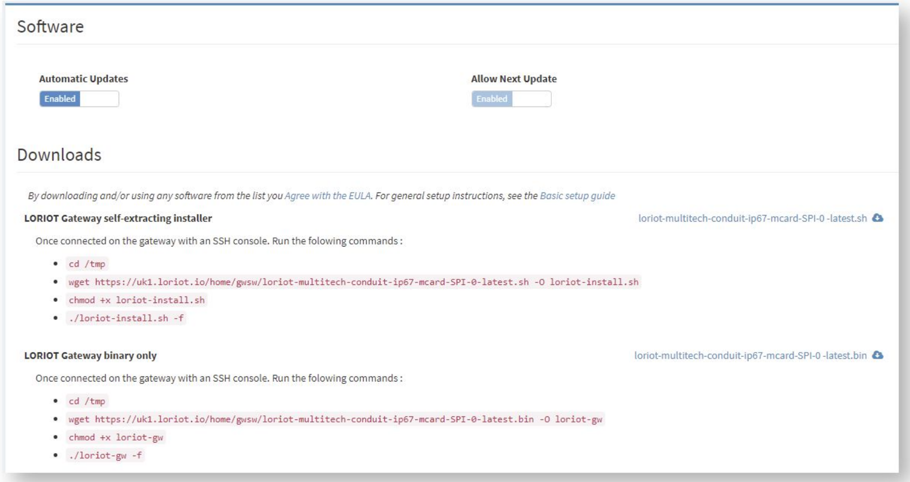
SYSTEM INTEGRATION

Integration of hardware with the system

Initial setup steps for Gateway and sensors:

1. Follow the instructions in the Multitech Gateway user manual. Once we have the Multitech gateway connected via PoE Ethernet cable, power up the gateway and it will go in commissioning mode.
2. Next access the gateway interface via any web-browser by entering the default IP address 192.168.2.1 and log in by setting up a new username and password. The login credentials will be emailed as part of the digital repository and submissions.
3. Go through the initial setup. Few screenshots attached below.
4. Optionally, create an account on Multitech DeviceHQ for provisioning and maintaining gateway and connected devices.

For the data to be analyzed in the cloud you will need a network server to interface between the gateway and cloud endpoints. The Network Server is at the core of an Internet of Things solution. Secure, scalable, carrier-grade network connectivity for your valuable IoT data.



Software

Automatic Updates: Enabled

Allow Next Update: Enabled

Downloads

By downloading and/or using any software from the list you Agree with the EULA. For general setup instructions, see the Basic setup guide

LORIOT Gateway self-extracting installer — loriot-multitech-conduit-ip67-mcard-SPI-0 -latest.sh

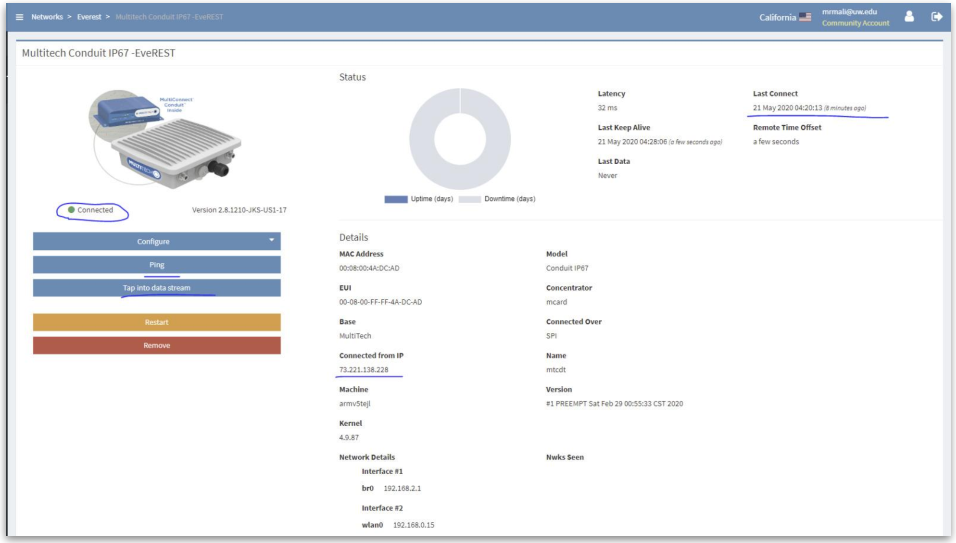
Once connected on the gateway with an SSH console. Run the folowing commands :

```
cd /tmp
wget https://uk1.loriot.io/home/gwsw/loriot-multitech-conduit-ip67-mcard-SPI-0-latest.sh -O loriot-install.sh
chmod +x loriot-install.sh
./loriot-install.sh -f
```

LORIOT Gateway binary only — loriot-multitech-conduit-ip67-mcard-SPI-0 -latest.bin

Once connected on the gateway with an SSH console. Run the folowing commands :

```
cd /tmp
wget https://uk1.loriot.io/home/gwsw/loriot-multitech-conduit-ip67-mcard-SPI-0-latest.bin -O loriot-gw
chmod +x loriot-gw
./loriot-gw -f
```



Networks > Everest > Multitech Conduit IP67 -EveREST — California — mrmali@uw.edu Community Account

Multitech Conduit IP67 -EveREST

Connected — Version 2.8.1210-JKS-US1-17

Status

Latency: 32 ms

Last Connect: 21 May 2020 04:20:13 *(8 minutes ago)*

Last Keep Alive: 21 May 2020 04:28:06 *(a few seconds ago)*

Remote Time Offset: a few seconds

Last Data: Never

Details

MAC Address: 00:08:00:4A:DC:AD

Model: Conduit IP67

EUI: 00-08-00-FF-FF-4A-DC-AD

Concentrator: mcard

Base: MultiTech

Connected Over: SPI

Connected from IP: 73.221.138.228

Name: mtcdt

Machine: armv5tejl

Version: #1 PREEMPT Sat Feb 29 00:55:33 CST 2020

Kernel: 4.9.87

Network Details

Interface #1: br0 192.168.2.1

Interface #2: wlan0 192.168.0.15

Nwks Seen

There are many popular network servers on the market today such as LORIOT (paid), The Things Network (TTN) (paid), ChirpStack (open-source LoRaWAN Network Server), etc. We have explored LORIOT and TTN and they both work fine with the setup we had.

Steps for registering gateway and sensors on the LORIOT network server.

1. We created an account in the US1 server region and used the free tier to test our setup.
2. Next, we registered the gateway using the following steps:

Registering the gateway

1. Log in to the LORIOT Network Server user interface and register the gateway as a MultiTech Conduit IP67.
2. Configure the options to select the appropriate LoRa mCard. Run the LORIOT Gateway Software
3. Select the newly registered gateway and navigate to the "Software" menu.
4. Run the installer commands from the Downloads section.

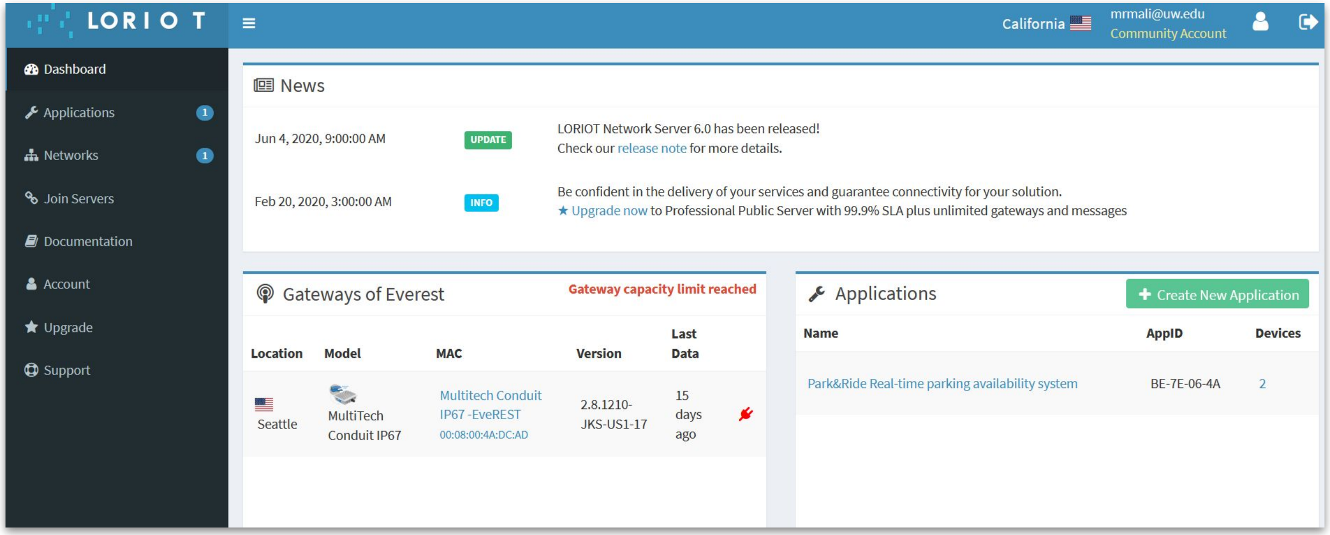
Once installed please check in the LORIOT user interface the gateway is online ("MultiTech Conduit IP67 - LORIOT Network Server - LORIOT documentation", n.d.).

SYSTEM INTEGRATION

Registering the sensor device:
We bought a screwdriver kit and fitted two sensors on the base for testing purposes.
In order to register the device on LORIOT application section the following fields are required ("Enroll a Device - LORIOT Network Server - LORIOT documentation", n.d.):

- Device Title (can be DevUI, Device model, your choice)
- DevEUI - 64-bit end-device identifier, EUI-64 (unique
- AppEUI - 64-bit application identifier, EUI-64 (unique)
- APPKEY - AES128 Root Key (unique)

Again these credentials will be provided as part of the digital submissions.



News

Jun 4, 2020, 9:00:00 AM — UPDATE — LORIOT Network Server 6.0 has been released! Check our release note for more details.

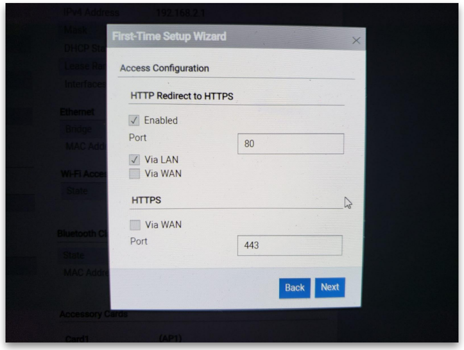
Feb 20, 2020, 3:00:00 AM — INFO — Be confident in the delivery of your services and guarantee connectivity for your solution. ★ Upgrade now to Professional Public Server with 99.9% SLA plus unlimited gateways and messages

Gateways of Everest — Gateway capacity limit reached

Location	Model	MAC	Version	Last Data
Seattle	MultiTech Conduit IP67	Multitech Conduit IP67 -EveREST 00:08:00:4A:DC:AD	2.8.1210-JKS-US1-17	15 days ago

Applications — + Create New Application

Name	AppID	Devices
Park&Ride Real-time parking availability system	BE-7E-06-4A	2



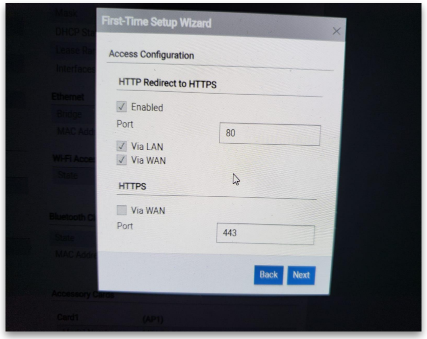
First-Time Setup Wizard

Access Configuration

HTTP Redirect to HTTPS
☑ Enabled
Port 80
☑ Via LAN
□ Via WAN

HTTPS
□ Via WAN
Port 443

Back Next



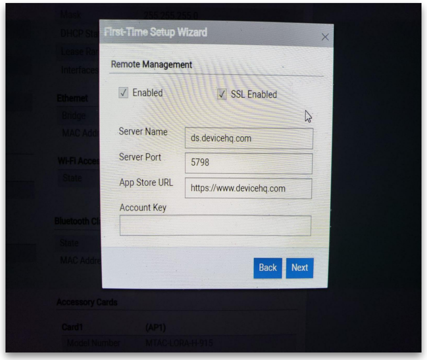
First-Time Setup Wizard

Access Configuration

HTTP Redirect to HTTPS
☑ Enabled
Port 80
☑ Via LAN
☑ Via WAN

HTTPS
□ Via WAN
Port 443

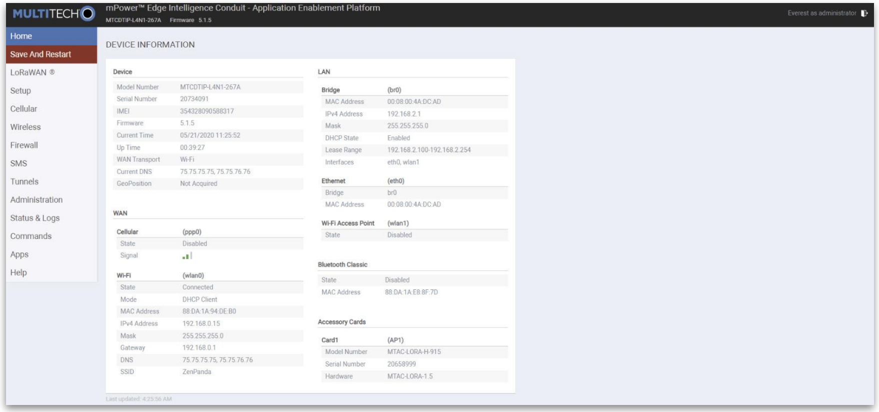
Back Next



First-Time Setup Wizard

Remote Management
☑ Enabled ☑ SSL Enabled
Server Name ds.devicehq.com
Server Port 5798
App Store URL https://www.devicehq.com
Account Key

Back Next



mPower™ Edge Intelligence Conduit - Application Enablement Platform
MTCDTIP-L4N1-267A Firmware 5.1.5

DEVICE INFORMATION

Device

Model Number	MTCDTIP-L4N1-267A
Serial Number	20734091
IMEI	354328090588317
Firmware	5.1.5
Current Time	05/21/2020 11:25:52
Up Time	00:39:27
WAN Transport	Wi-Fi
Current DNS	75.75.75.75, 75.75.76.76
GeoPosition	Not Acquired

WAN

Cellular	(ppp0)
State	Disabled
Signal	
Wi-Fi	(wlan0)
State	Connected
Mode	DHCP Client
MAC Address	88:DA:1A:94:DE:B0
IPv4 Address	192.168.0.15
Mask	255.255.255.0
Gateway	192.168.0.1
DNS	75.75.75.75, 75.75.76.76
SSID	ZenPanda

LAN

Bridge	(br0)
MAC Address	00:08:00:4A:DC:AD
IPv4 Address	192.168.2.1
Mask	255.255.255.0
DHCP State	Enabled
Lease Range	192.168.2.100-192.168.2.254
Interfaces	eth0, wlan1
Ethernet	(eth0)
Bridge	br0
MAC Address	00:08:00:4A:DC:AD
Wi-Fi Access Point	(wlan1)
State	Disabled

Bluetooth Classic

State	Disabled
MAC Address	88:DA:1A:E8:8F:7D

Accessory Cards

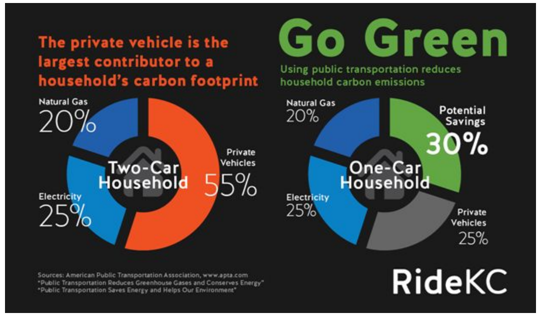
Card1	(AP1)
Model Number	MTAC-LORA-H-915
Serial Number	20658999
Hardware	MTAC-LORA-1.5

Last updated: 4:25:56 AM

IMPACT

IMPACT

Image Source: https://ec.europa.eu/jrc/en/news/air-quality-traffic-measures-could-effectively-reduce-no2-concentrations-40-europe-s-cities



Transportation in major cities is one of the major reasons for greenhouse gas emissions whose dangers are known to everybody and is a big challenge for environmental scientists all over the world. The change in climate patterns are leading to submergence of small islands, droughts, floods, melting of glaciers and the list can go on. All of it is having a very detrimental effect on the environment which would ultimately threaten the very human civilization. Even a single person switching from car to existing public transport can contribute to a significant reduction in CO_2 emission.

Life on Land is the goal 15 of the Sustainable development Goals of the United Nations. This product can revolutionize the use of public transport by making the use of Park and Ride more environmentally friendly for the commuters. The application we are making would be free for everybody and the data stored would be in adherence to the GTFS which defines a common format for public transportation schedules and associated geographic information. Moreover, the data would be represented on the map which can be ingested in other trip-planning applications through an API.

This would lead to the democratization of data and a better user experience for the commuters. We depict the parking availability in simple color-coded schemes like Red, Yellow, Green which is easy to comprehend and the commuters would not need to learn anything.

EZPark&Ride is for every commuter as it works for people with different transit patterns. The commuters will be able to find the data in any map services that they like, such as Google Maps, which would help reduce rush-hour stress by giving the most accurate data free of cost to everybody. With equitable access given to all commuters, EZPark&Ride can help the city promote transit options that not only can reduce traffic congestion, but also use less energy which is beneficial to environmental sustainability.

This would help to convert the Puget Sound area into a smart city where digital transformation would support trip planning and transportation. EZPark&Ride users data to help the city to improve transit services by tailoring public transit schedules to commuter's travel patterns.

It's a solution that can give real-time parking information, backup parking locations have predictive algorithms to suggest when would the parking spaces get filled and public transit routes (bus numbers and train stations) from your parking location. It gets all the necessary intelligence to help decision-makers shape the future of the city's mobility system.

DATA STANDARD

The General Transit Feed Specification (GTFS), defines a common format for public transportation schedules and associated geographic information. GTFS "feeds" let public transit agencies publish their transit data and developers write applications that consume that data in an interoperable way.

Since Park and Ride facilities aim to encourage people to use public transit, to add both real-time parking availability data and public transport information together in the data pipeline is an important step to make the whole system work hand in hand efficiently.
Our product design integrates bus schedules which it ingests from GTFS.

For current public transit data in GTFS format, it includes static timetables, stops locations, and route shape information as well as regional routes, trackwork, and transport routes. When we build our ERD and ETL for our database and Dataware, we consider the GTFS format and try to make the new real-time parking availability data consistent with the current data format.

Meanwhile, we know making GTFS is not just a technical issue, it also involves the collaboration among different partners to make data open to keep data formats consistent and be used by the public efficiently.

It is important to study the policies of certain areas, certain organizations, and certain companies to think about strategies to make data open for further building GTFS data solutions. That is an important aspect to explore in the next step.

Image Source: https://cdn2.hubspot.net/hubfs/2753005/GTFS%20post%20title_600x200@2x.png

PREDICTION FUNCTION

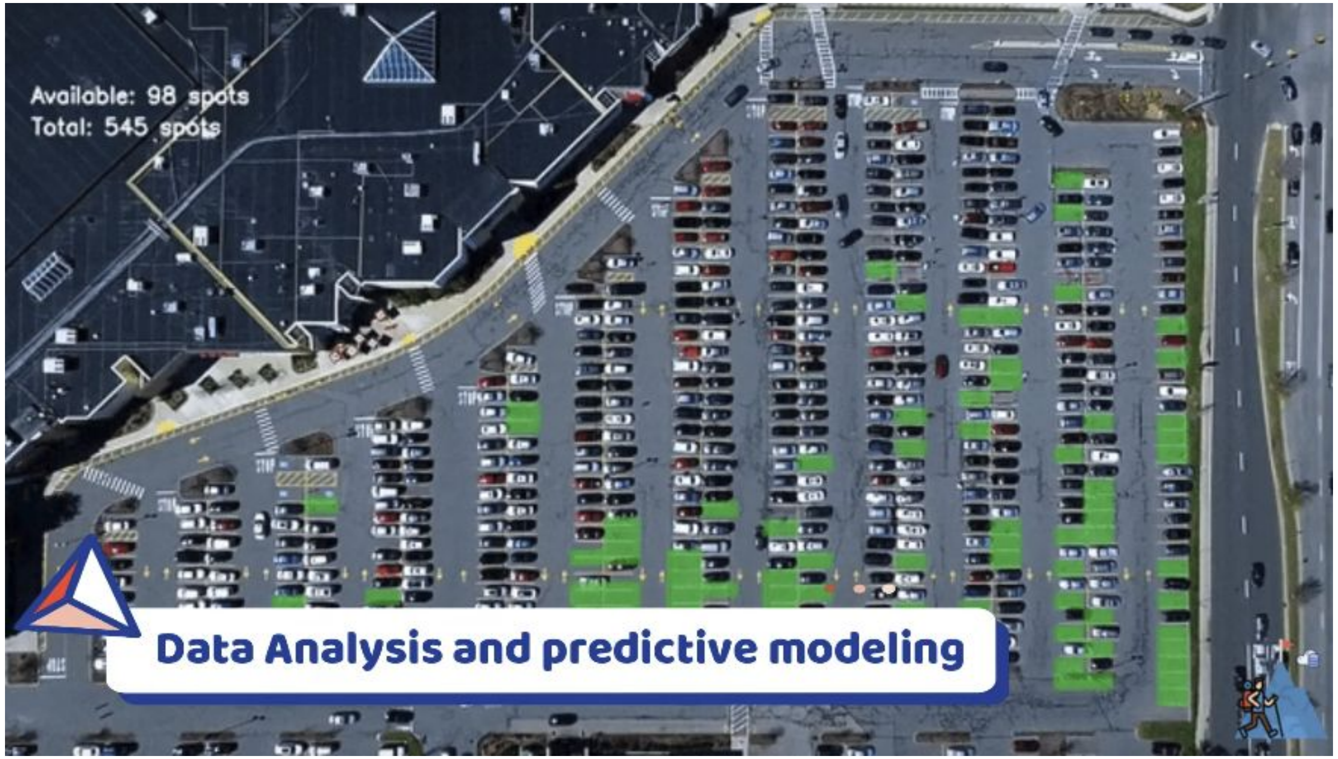
Prediction is something that sets our product apart from all other products. However, it was a priority 3 deliverable in our scope.
Our product should be able to tell about the expected time remaining before the available parking spaces would get filled up in all the Park and Ride locations.

We have applied a machine-learning algorithm to do this. Due to the time constraints and prediction being low in priority of our deliverables, we did not apply any customized model. We used the Azure ML Studio and its inbuilt time series model for prediction.

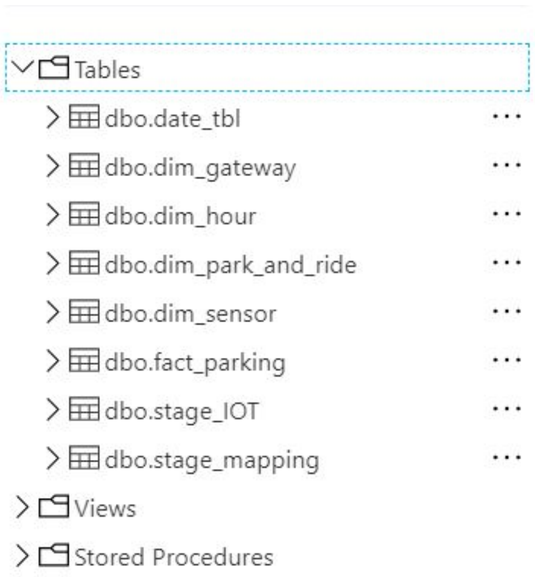
The model is supposed to regress on features like Park&Ride id, the total number of parking spaces, number of occupied spaces, date time and predict how much time is required to fill up the required spaces. It would be a time series model to do regression. Our data were not from actual IoT sensors which were not sending signals because of not being compatible with US carrier frequency bands but rather being compatible with the Japanese carrier frequency bands. Our data was from an API and was not correct due to which the machine learning model was failing. The output variable can take a continuous set of values, so it's a case of regression.

Azure Machine Learning

Image Source:
https://nub8.net/machine-learning-with-azure-machine-learning-services/



SCALABLE DATA WAREHOUSE



Tables
- dbo.date_tbl
- dbo.dim_gateway
- dbo.dim_hour
- dbo.dim_park_and_ride
- dbo.dim_sensor
- dbo.fact_parking
- dbo.stage_IOT
- dbo.stage_mapping

Views

Stored Procedures

The raw data we would collect should be stored in a data warehouse for processing. Moreover, you would require this data to train your predictive models. There should be a data warehouse in a star schema. There should be a fact table that has the parking status at various timestamps; and dimension tables for Park&Ride, IOT_Device, Gateway, Date, Address, etc.

It is parking data, so it might not go to a big data level. The data warehouse may be built-in Azure SQL. We wrote our data pipeline in the Azure data factory to get source data(fake data from API) in the staging tables, and a stored procedure to load data in the fact table. We did not have consistent source data. We inserted some rows manually into our dimension tables.

Please get the source data in staging tables before doing the ETL. It is better to explore how Azure DataFactory can be integrated with Microsoft SSIS. And to make a fully functional data warehouse, please get consistent source data and write ETL pipelines to load the data in the dimension tables. The data load should be incremental and through ETL.

As there would not be dynamic changes in various attributes of the dimensions, the team should go for Slowly Changing Dimension-2 to maintain full history; disk space utilization should not be a problem as the dimensions are also not very big. The data load in the fact table also must be incremental and through a scheduled ETL. You can select the latest date time from the already existing data in the fact table, save it in a variable; and then load incremental data having date timestamp later than what you stored in the variable.

Note -

We interviewed a Microsoft team and they suggested us to consider just having a data lake instead of a data warehouse, as parking data is not very huge.

LIMITATIONS

Image Source: https://media-ecn.s3.amazonaws.com/embedded_image/2015/12/

Real-world Data

Because of the COVID situation, we did not get to deploy the parking sensors and other hardware in this project. That results in a lack of real-world data. In order to implement the system as a minimum variable product, we developed a simulated data feed that we used to substitute the sensor data feed in the system. As we do not have adequate training data to build the time series forecasting model and did not get sufficient time to explore more machine learning algorithms, the prediction module in EZPark&Ride is not functioning.

Hybrid Solution

We did not have enough time to experiment with other smart parking solutions, which includes image processing using cameras.

Data Standard

We only have a basic idea of integrating out data feed with the GTFS framework but have not implemented it. Since the GTFS standard does not include any parking availability data, we realized that there should be a mechanism on the back-end that works to include the parking data as a part of the GTFS feed. We also did not know of any other standard format that commercial parking systems utilize. Such kind of format could help us understand the integration of parking data with mapping services (e.g., Google Map, Waze, etc).

Information Security

Even though the cloud data storage services we use (e.g., Cosmos DB) come with enterprise-level encryption enabled, we did not consider any Identity and Access Management (IAM) measures and other ways of securing data from an information security perspective.

FUTURE SCOPE OF PREDICTION FUNCTION

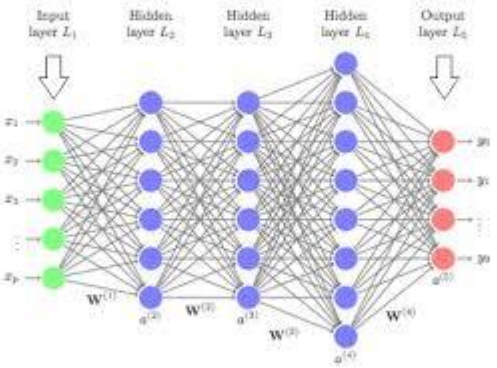


Image Source:
https://orbograph.com/deep-learning-how-will-it-change-healthcare/

The prediction was not the primary deliverable, rather a Priority 3 deliverable for team EveREST as per the defined scope. So, our team had just used the ML Studio inbuilt in Azure.

However, to make a model worthy of live deployment, the next team would have to build their custom model for EZPark&Ride.

There is very limited parking data available from surveys. The team can replicate this data over several days, do some minor variations, add some noise, and generate some dummy data from it. It might not be a good approach, as there would be too much dummy data and very little actual data. Moreover, such data would not capture seasonal variation; but it is still worth trying.

Another approach that the team can take is to find if such data is available for another parking lot whose parking pattern may be correlated to Park & Ride; however, it would depend too much on assumptions. Lastly, the team can train several machine learning models. The goodness of fit for the model can only be calculated after testing the model and finding its Root Mean Squared Error, accuracy, and other metrics.

For a better approach, the team would have to gather a huge amount of data from the sensors and apply time series models to get a better prediction about how much time it would take to fill the available number of parking spaces. Different types of time series would have to be tried and hyperparameter tuning would have to be performed for all the algorithms to select the best model.

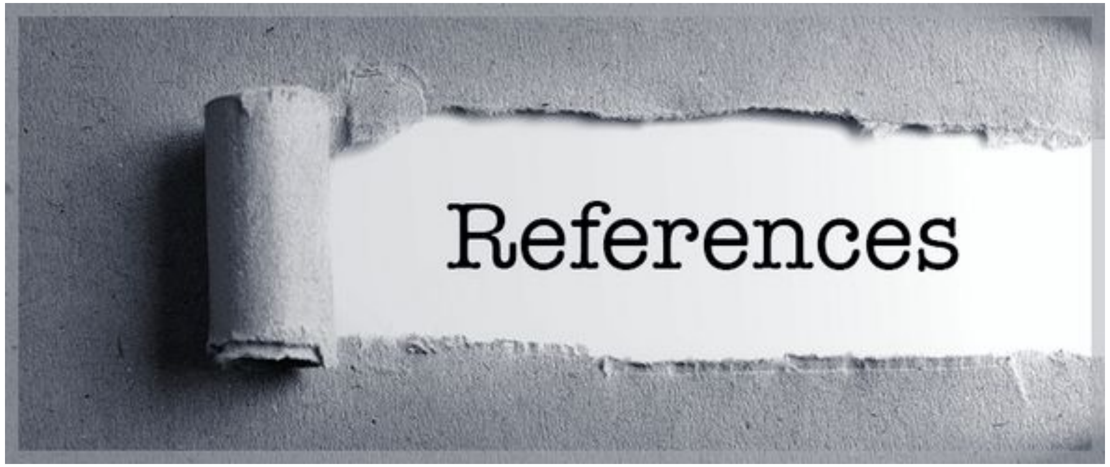
Besides testing some time series models, the Remaining Useful Life (RUL) algorithm should also be tested.

Creating such a prediction engine would be a huge task and not a small side task. It should be considered as a separate mini Capstone project if not a full-fledged project. So, the data collected should have a lower level of granularity and stored either in a data warehouse or a data lake. This would ensure that even if the created model is not working very well; there would be enough data resources available to enhance it in the future.

SUGGESTIONS FOR THE NEXT TEAM

Our job in this project was to study the existing solutions, conduct user research to know what the commuters want and created the prototype of an infrastructure that can provide real time parking availability. Our suggestions for the next team is as follows:

1. Our scope was very broad, we would suggest the next team to narrow down the scope.
2. This project is huge and consists of the following major sub-projects:
   - gather real-time data and store it
   - make it available on a map and provide through an API
   - integrate the application with other parking facilities owned by different entities
   - data warehouse for analytics
   - prediction engine to predict the time required to fill parking spaces
3. Try to gather a consistent source data from fact and dimension tables for data warehouse.
4. Try to find a hybrid solution and experiment with off the shelf solutions for better quality.



REFERENCES & APPENDICES

REFERENCES

1. 2017 Q4 service delivery performance report. (2017). Retrieved June 8, 2020, from: https://www.soundtransit.org/sites/default/files/2017-q4-service-delivery-performance-report.pdf

2. About us | WSDOT. (2020). Retrieved June 8, 2020, from: https://www.wsdot.wa.gov/About

3. Blandin, S., Wynter, L., Poonawala, H., Laguna, S., & Dura, B. (2020). FASTER: Fusion AnalyticS for public Transport Event Response. Retrieved June 8, 2020, from: http://www.ifaamas.org/Proceedings/aamas2019/pdfs/p1404.pdf

4. Commute Seattle. (2020). Retrieved June 8, 2020, from: https://commuteseattle.com/who-we-are/

5. Engdahl, S. (2018, October 1). Connect your devices to AWS IoT using LoRaWAN. Retrieved June 8, 2020, from https://aws.amazon.com/blogs/iot/connect-your-devices-to-aws-iot-using-lorawan/

6. Enroll a Device - LORIOT Network Server - LORIOT documentation. Retrieved June 8, 2020, from: https://docs.loriot.io/display/LNS/Enroll+a+Device

7. Geng, Y.; Cassandras, C.G. (2012). A new “Smart Parking” System Infrastructure and Implementation. Procedia Soc. Behav. Sci.

8. Khalid, M.; Cao, Y.; Zhang, X.; Han, C.; Peng, L.; Aslam, N.; Ahmad, N. (2018). Towards autonomy: Cost-effective scheduling for long-range autonomous valet parking (LAVP). In Proceedings of the 2018 IEEE Wireless Communications and Networking Conference (WCNC), Barcelona,Spain; pp. 1–6.

9. loriot/AzureSolutionTemplate. Retrieved June 8, 2020, from: https://github.com/loriot/AzureSolutionTemplate

10. MetroServiceDevMgrBroch. (2015). Retrieved June 8, 2020, from: https://www.kingcounty.gov/~/media/transportation/kcdot/MetroTransit/human-resources/MetroServiceDevMgrBroch.ashx?la=en&usg=AOvVaw3ely_RqVm-40VTcwOmm2at

REFERENCES

11. Mobility innovation center | University of Washington. (2020). Retrieved June 8, 2020, from: https://mic.comotion.uw.edu/#

12. MultiTech Conduit IP67 - LORIOT Network Server - LORIOT documentation. Retrieved June 8, 2020, from: https://docs.loriot.io/display/LNS/MultiTech+Conduit+IP67

13. MultiTech Conduit IP67 LoRa Gateway | Outdoor LoRa Deployment. Retrieved June 8, 2020, from: https://www.multitech.com/brands/multiconnect-conduit-ip67

14. Parking Lot Sensor | PLS. (2020). Retrieved June 8, 2020, from: https://www.bosch-connectivity.com/products/connected-mobility/parking-lot-sensor/

15. Process real-time vehicle data using IoT - Azure Architecture Center. (2020, March 10). Retrieved June 8, 2020, from https://docs.microsoft.com/en-us/azure/architecture/example-scenario/data/realtime-analytics-vehicle-iot

16. Real Time Analytics on Big Data Architecture - Azure Solution Ideas. (2019, December 16). Retrieved June 8, 2020, from https://docs.microsoft.com/en-us/azure/architecture/solution-ideas/articles/real-time-analytics

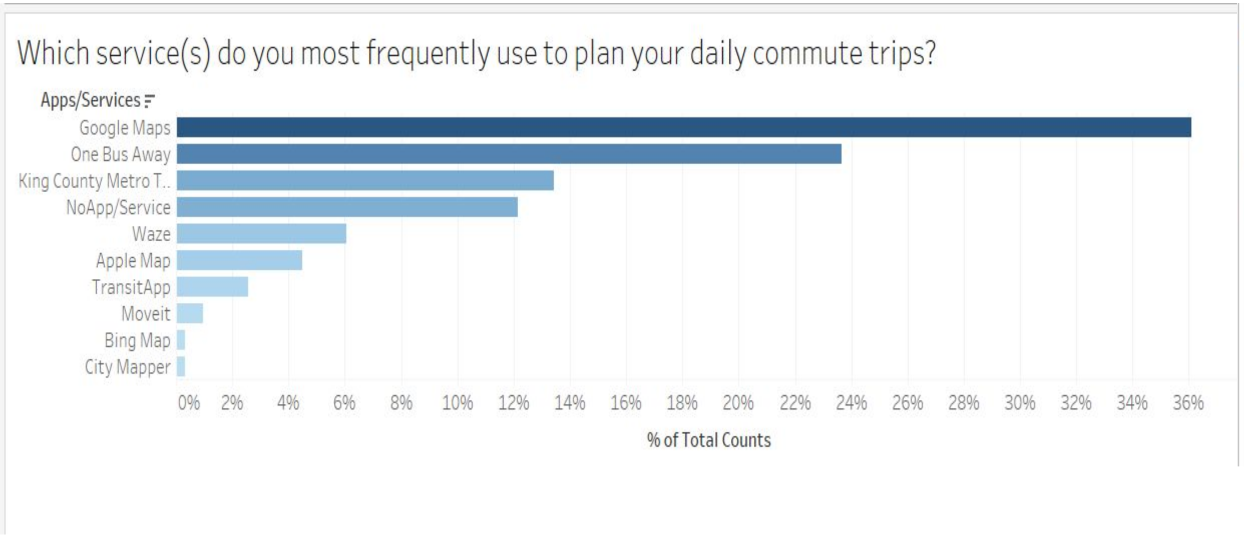
17. Real-Time Parking Availability. (2020). Retrieved June 8, 2020, from: https://www.metro.net/riding/parking/real-time/

18. Santa Barbara - Real-Time Parking Availability. (2020). Retrieved June 8, 2020, from: https://www.santabarbaraca.gov/gov/depts/pw/dtp/pmap.asp?utm_source=PublicWorks&utm_medium=realtimeparking&utm_campaign=QuickLinks

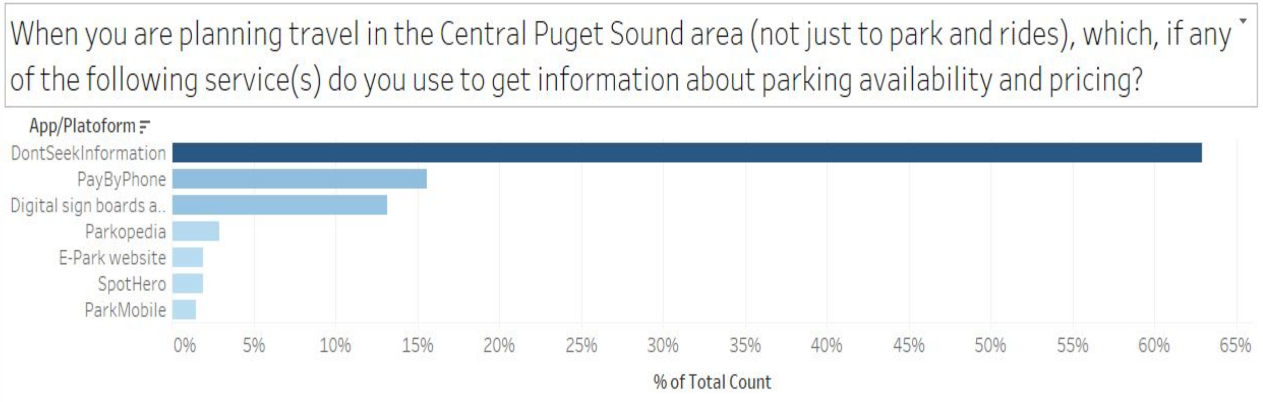
19. SmartPark: Real-Time Parking Availability | FMCSA. (2014). Retrieved June 8, 2020, from: https://www.fmcsa.dot.gov/research-and-analysis/technology/smartpark-real-time-parking-availability

20. Thomas, D.; Kovoor, B.C. (125). A Genetic Algorithm Approach to Autonomous Smart Vehicle Parking system. Procedia Comput. Sci. 125, 68–76.

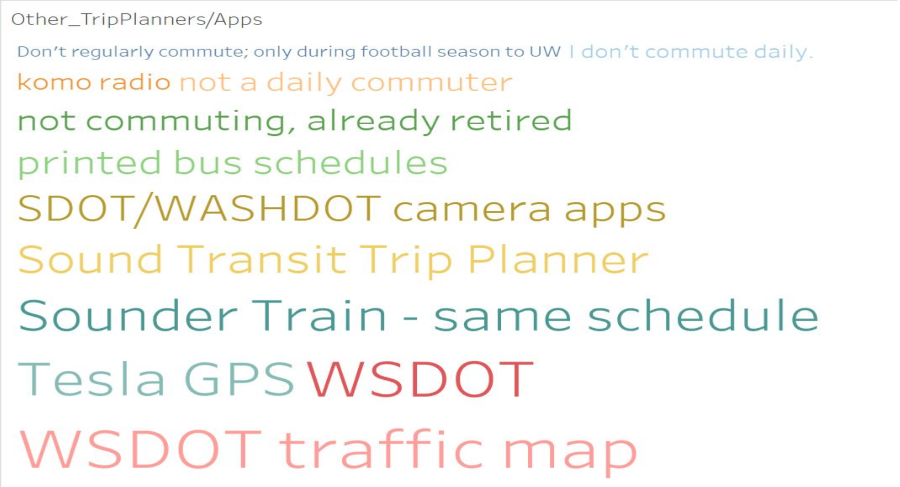
Survey Data Analysis and Visualization



Which service(s) do you most frequently use to plan your daily commute trips?

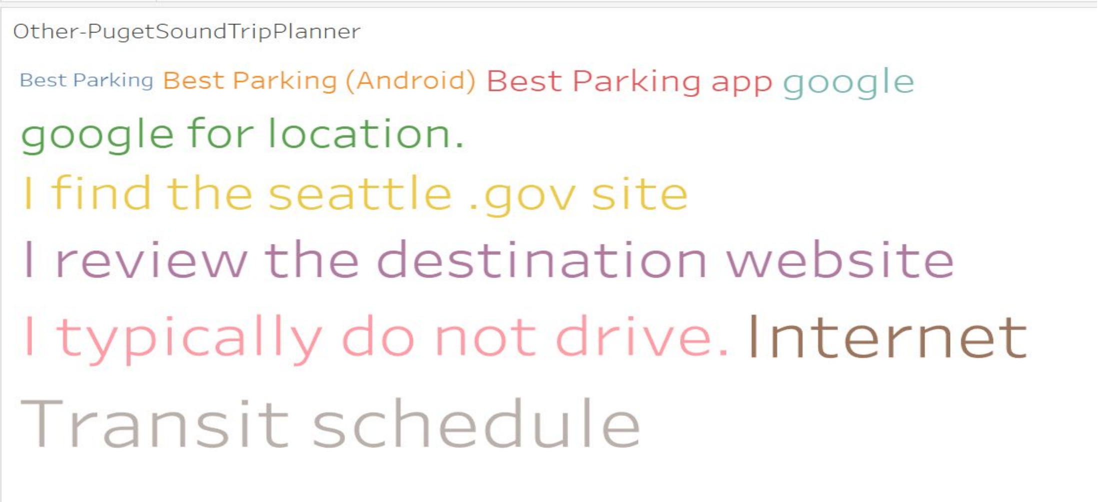


When you are planning travel in the Central Puget Sound area (not just to park and rides), which, if any of the following service(s) do you use to get information about parking availability and pricing?



Other_TripPlanners/Apps

Don't regularly commute; only during football season to UW I don't commute daily.
komo radio not a daily commuter
not commuting, already retired
printed bus schedules
SDOT/WASHDOT camera apps
Sound Transit Trip Planner
Sounder Train - same schedule
Tesla GPS WSDOT
WSDOT traffic map

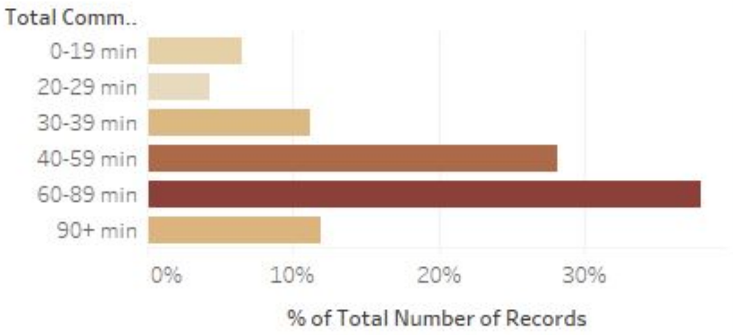


Other-PugetSoundTripPlanner

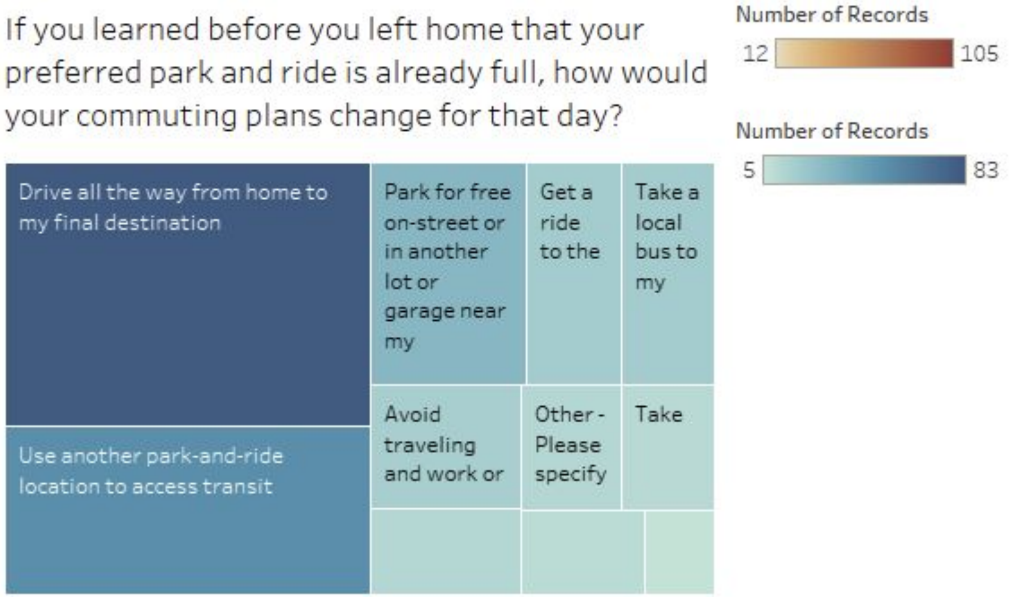
Best Parking Best Parking (Android) Best Parking app google
google for location.
I find the seattle .gov site
I review the destination website
I typically do not drive. Internet
Transit schedule

APPENDICES

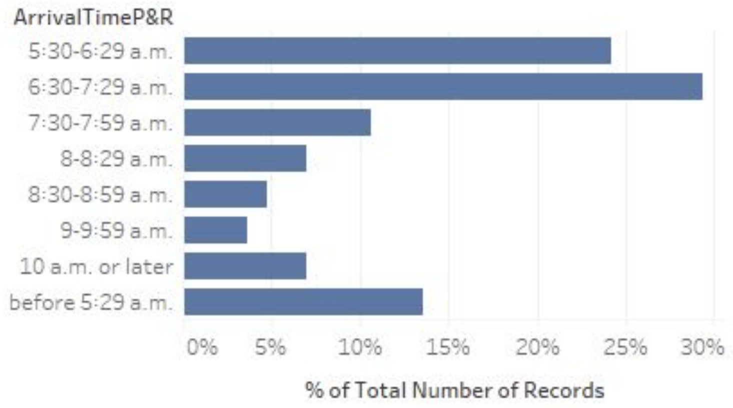
TotalCommuteTime_PercentofResponses



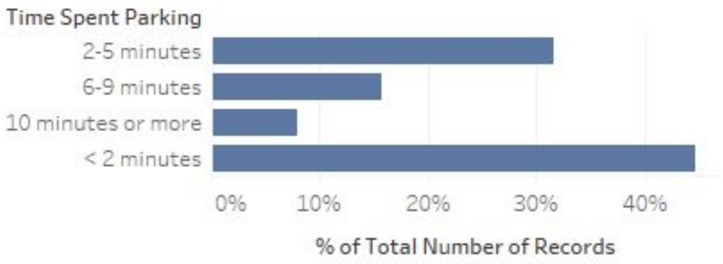
If you learned before you left home that your preferred park and ride is already full, how would your commuting plans change for that day?



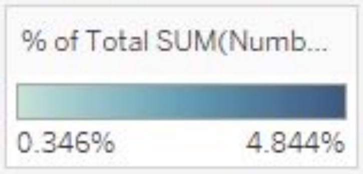
When do you arrive at P&R on weekdays?



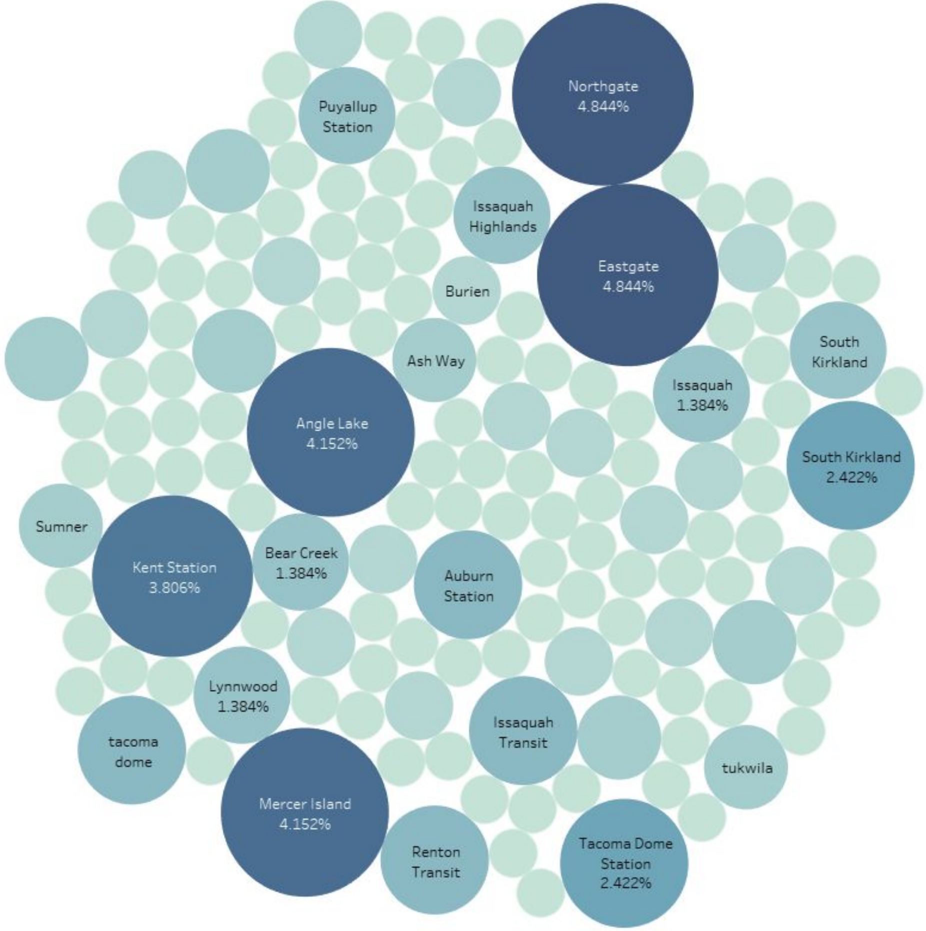
How much time did you typically spend looking for a parking space?



APPENDICES



Which P&R you use most frequently?



Other-Unable_FindParking

dont know Drive in circles like a complete asshole. Give up. Go home. go home & work from home
Go home and take a shared ride service park a few miles away and walk
Park for free at a friend's house near another Light Rail station Park in a spot not designated for parking
Park in the Target parking lot park longer distance away and take a bus to light rail walk

Other_LearnParkingFull

arrive earlier at the P&R I Drop off, don\'t park
I tend to take transit to from park and ride and have people pick me up drop me off for the transit connections. NA
These questions are annoying and too driver commuting centrif
park at my church park illegally at the park and ride like everyone else does. Park in the Target parking lot
park longer distance away and take a bus to light rail Street parking near a second choice rail station
Walk to a different bus stop work from home

Other_TripPlanners/Apps

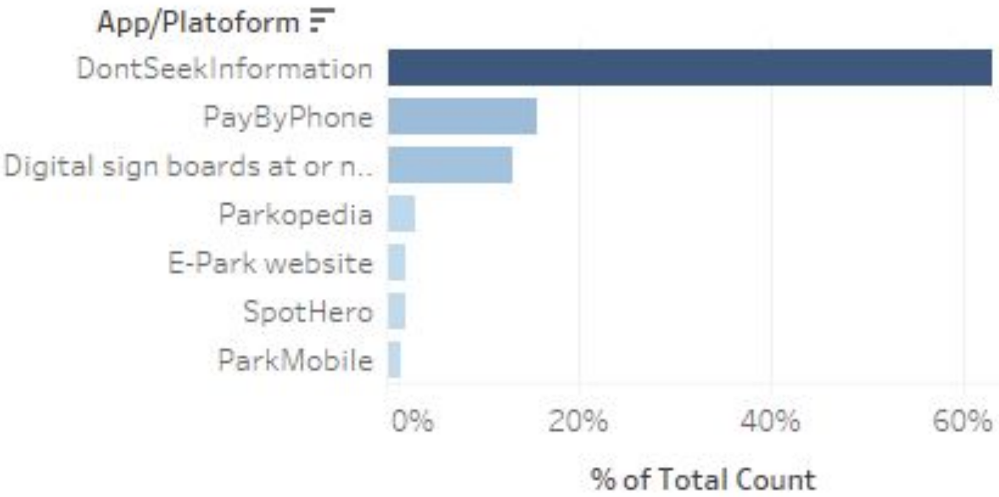
Don't regularly commute; only during football season to UW
I don't commute daily. komo radio not a daily commuter
not commuting, already retired printed bus schedules
SDOT/WASHDOT camera apps Sound Transit Trip Planner
Sounder Train - same schedule Tesla GPS WSDOT
WSDOT traffic map

Other-PugetSoundTripPlanner

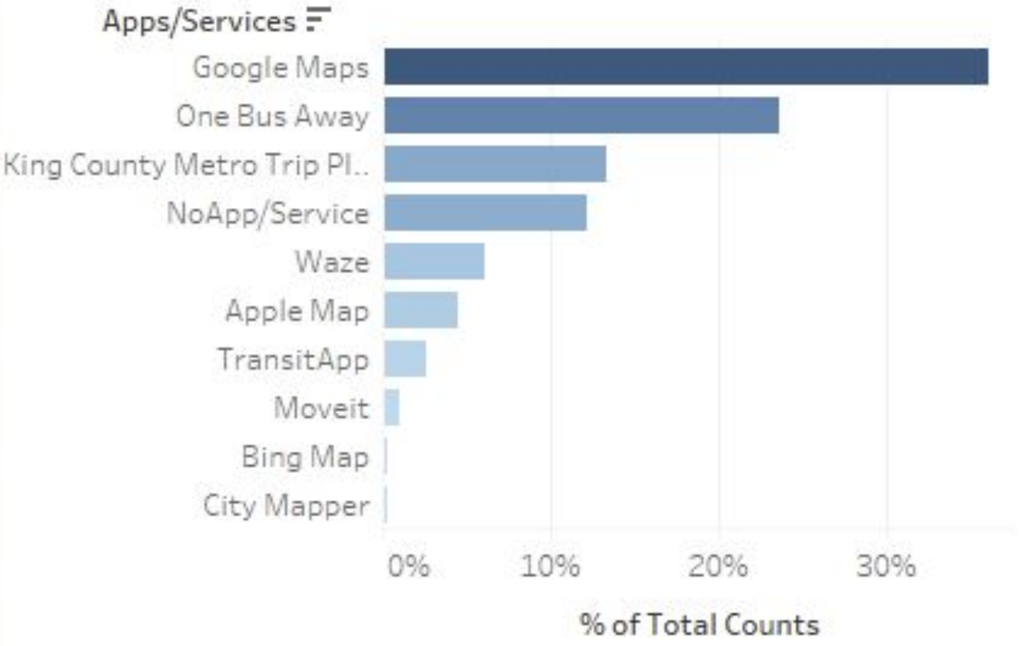
Best Parking Best Parking (Android) Best Parking app google google for location. I find the seattle .gov site
I review the destination website I typically do not drive. Internet Transit schedule

APPENDICES

When you are planning travel in the Central Puget Sound area (not just to park and rides), which, if any of the following service(s) do you use to get information about parking availability and pricing?



Which service(s) do you most frequently use to plan your daily commute trips?



% of Total Count

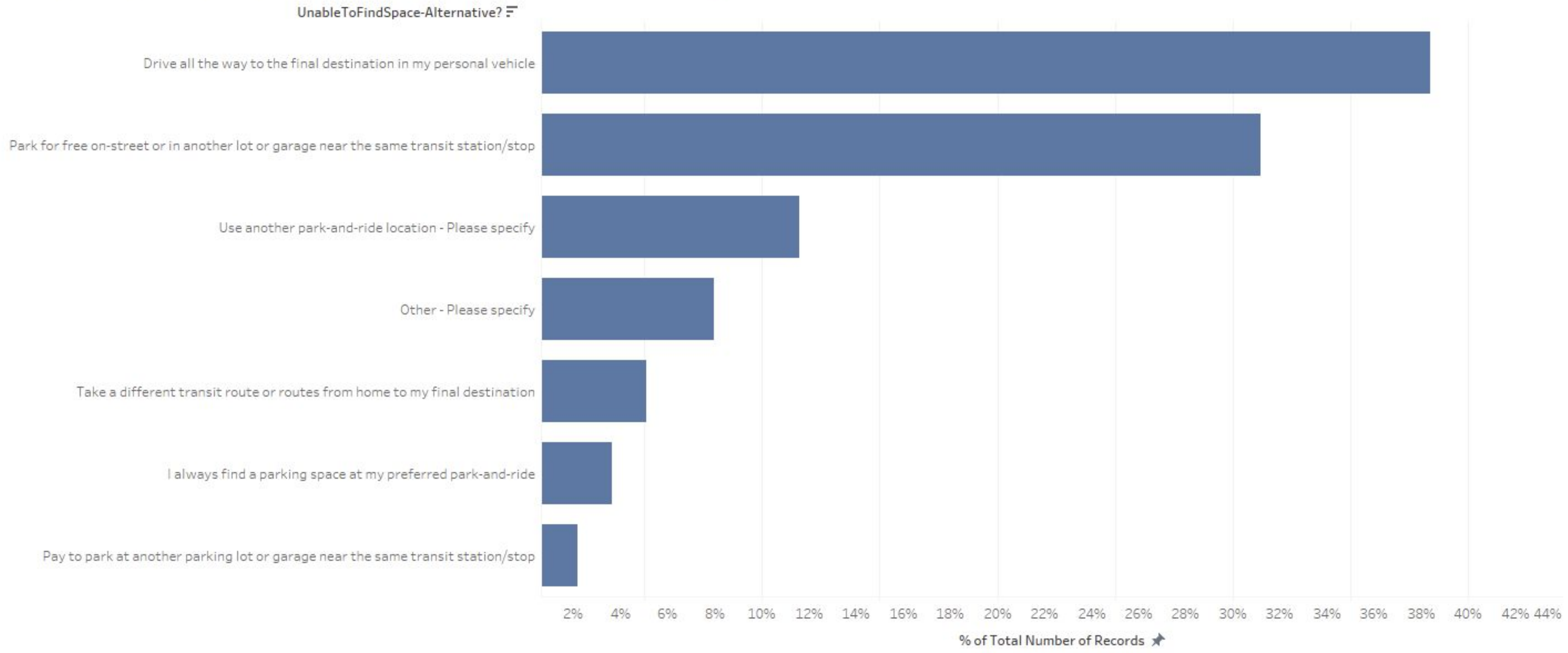
1.46% 62.93%

% of Total Counts

0.32% 36.10%

APPENDICES

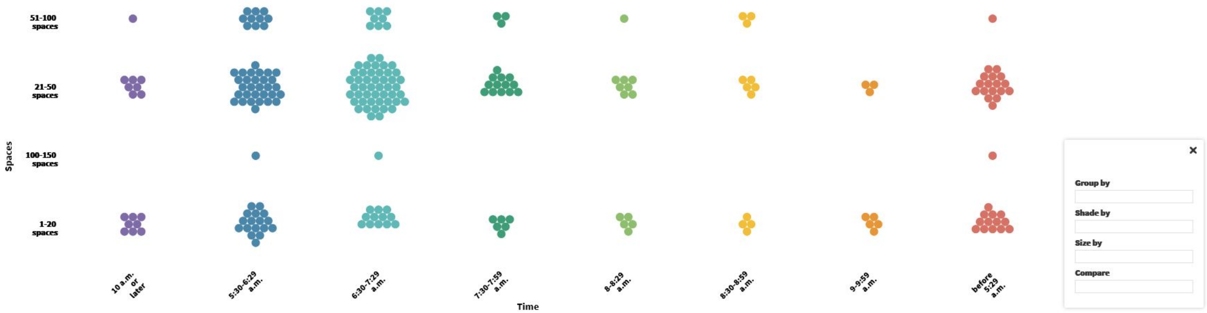
On days when you were not able to find space at your preferred park-and-ride, what did you typically do?



APPENDICES

Parking spaces requested by users arriving at Park and ride facilities to feel confident of parking during different times.

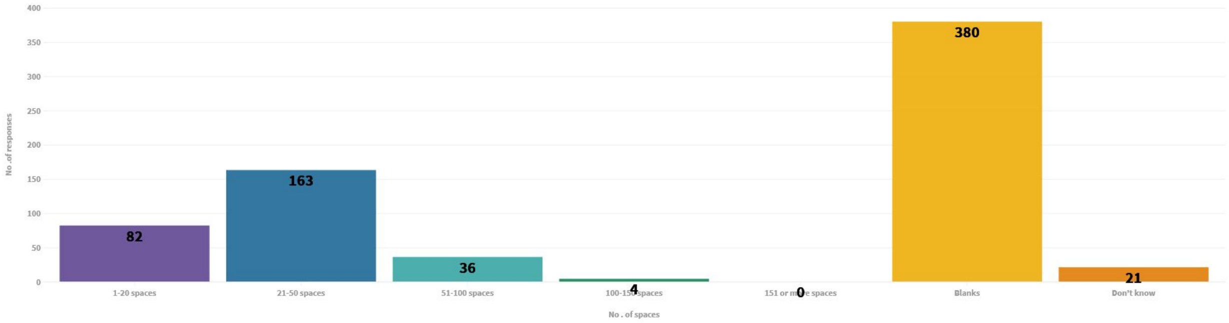
When did you most often arrive at the park and ride on weekdays? Accounting for the time required to reach your preferred park and ride, how many spaces should be open and available when you check the app/service, for you to feel confident that you will be able to find a parking space when you arrive?



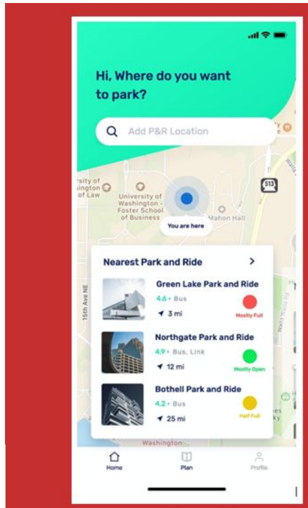
Source: Team EveREST Survey 2020

Parking spaces

How many spaces should be open and available when you check the app/service, for you to feel confident that you will be able to find a parking space when you arrive?

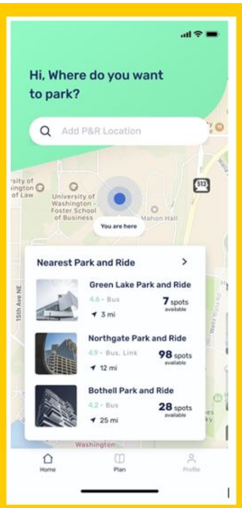


Source: Team EveREST Survey 2020



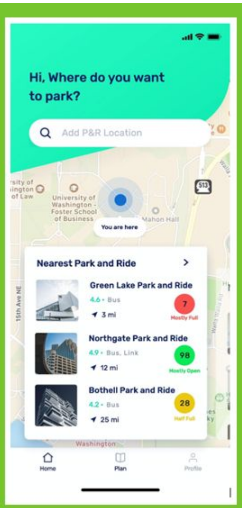
A.Only Color

211 people chose A as their THIRD choice



B.Only count of stalls

162 people chose B as their SECOND choice



C.Count of available stalls shaded by color

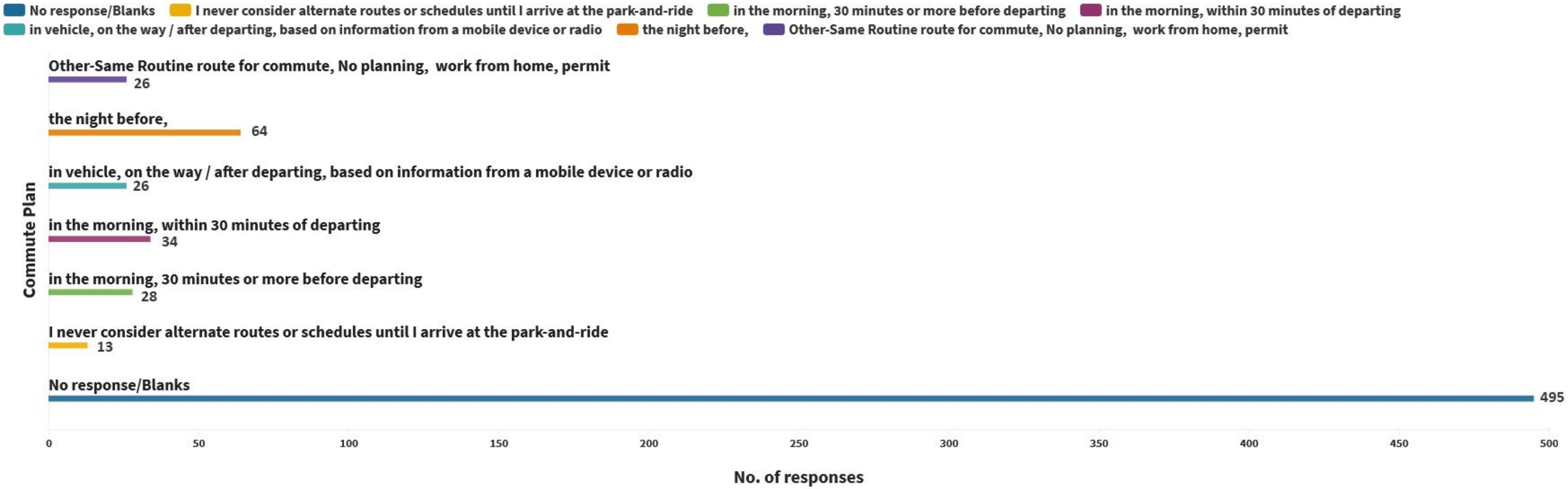
162 people chose C as their FIRST choice

61.6% people rated C,B,A as their top choice order

APPENDICES

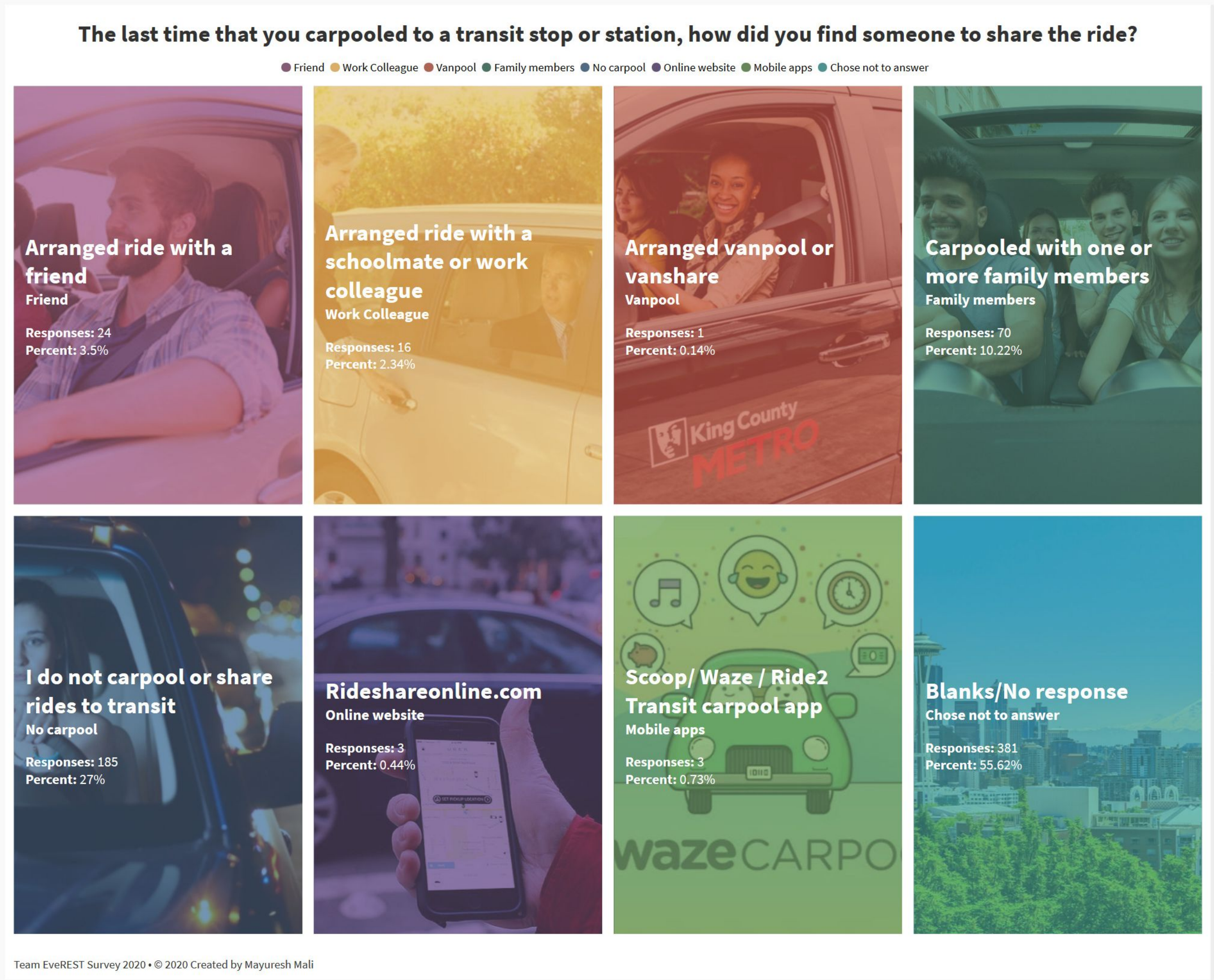
Commute Planning

For a typical weekday, when and how do you usually plan your commute trip (e.g. pick your mode of transportation, time of departure, and preferred park and ride)?



Source: Team EveREST Survey 2020 • ©2020 Created by Mayuresh Mali

APPENDICES



The last time that you carpooled to a transit stop or station, how did you find someone to share the ride?

● Friend ● Work Colleague ● Vanpool ● Family members ● No carpool ● Online website ● Mobile apps ● Chose not to answer

Arranged ride with a friend
Friend

Responses: 24
Percent: 3.5%

Arranged ride with a schoolmate or work colleague
Work Colleague

Responses: 16
Percent: 2.34%

Arranged vanpool or vanshare
Vanpool

Responses: 1
Percent: 0.14%

Carpooled with one or more family members
Family members

Responses: 70
Percent: 10.22%

I do not carpool or share rides to transit
No carpool

Responses: 185
Percent: 27%

Rideshareonline.com
Online website

Responses: 3
Percent: 0.44%

Scoop/ Waze / Ride2 Transit carpool app
Mobile apps

Responses: 3
Percent: 0.73%

Blanks/No response
Chose not to answer

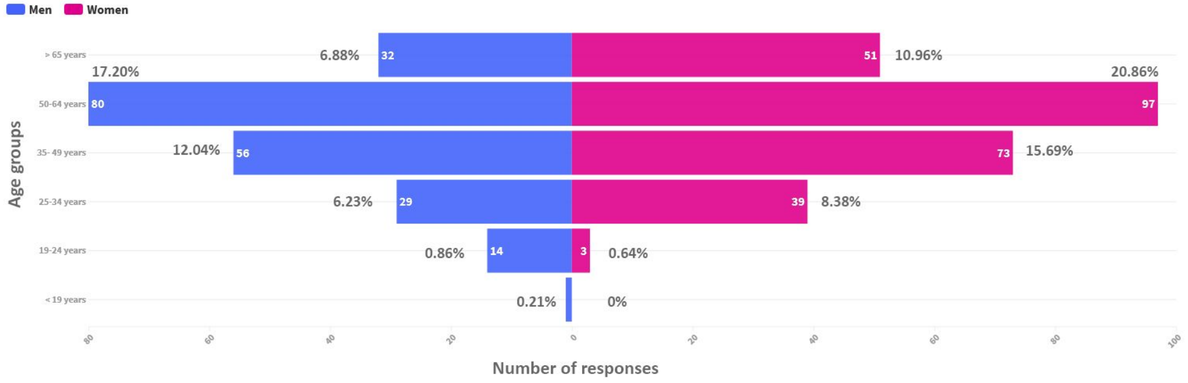
Responses: 381
Percent: 55.62%

Team EveREST Survey 2020 • © 2020 Created by Mayuresh Mali

Distribution of Gender by Age

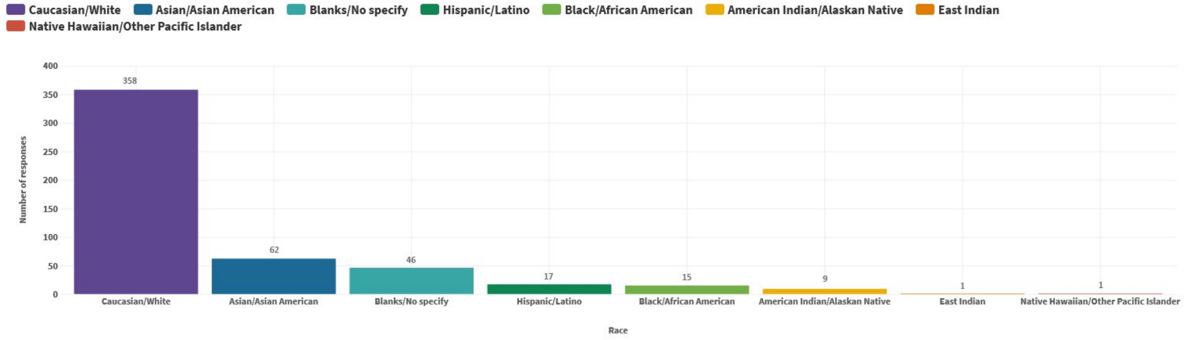
Participation of people using the Park and Ride was high among the age bracket of 35 and up.

Of the 687 participants surveyed, 499 disclosed their gender and age details. 465 identied themselves as either Male or Female with 5 non-binary responses and 29 who preferred not to say.



Race distribution

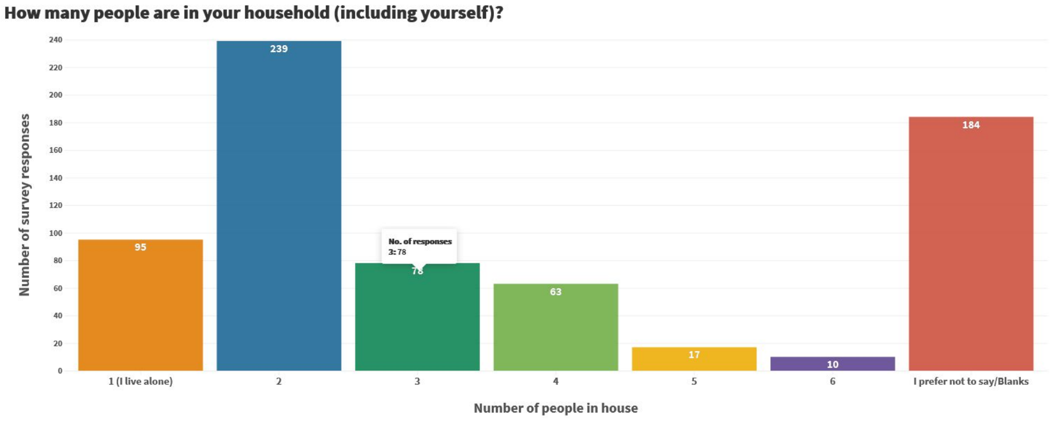
509 respondents chose to identify their race.

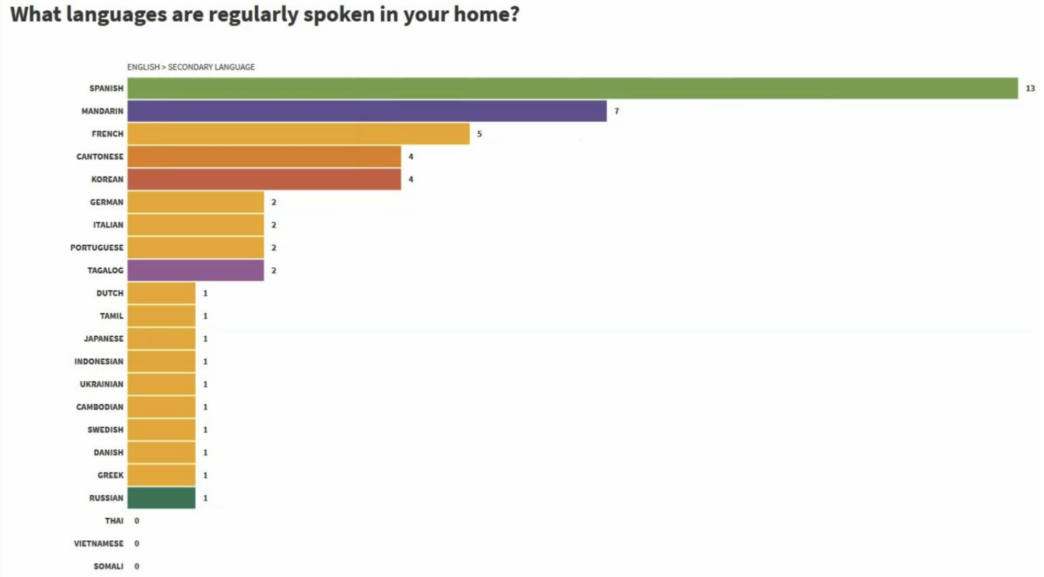


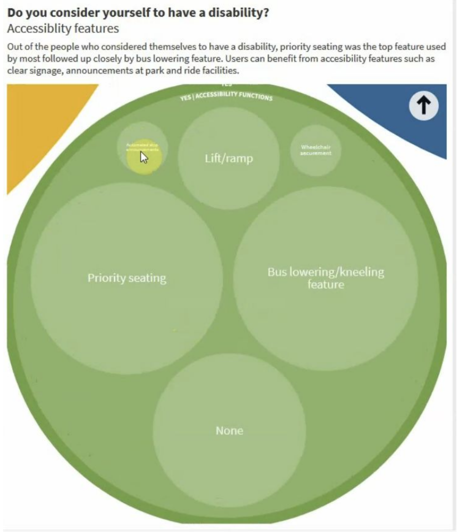
Source: Team EveREST Survey 2020

Team EveREST

APPENDICES

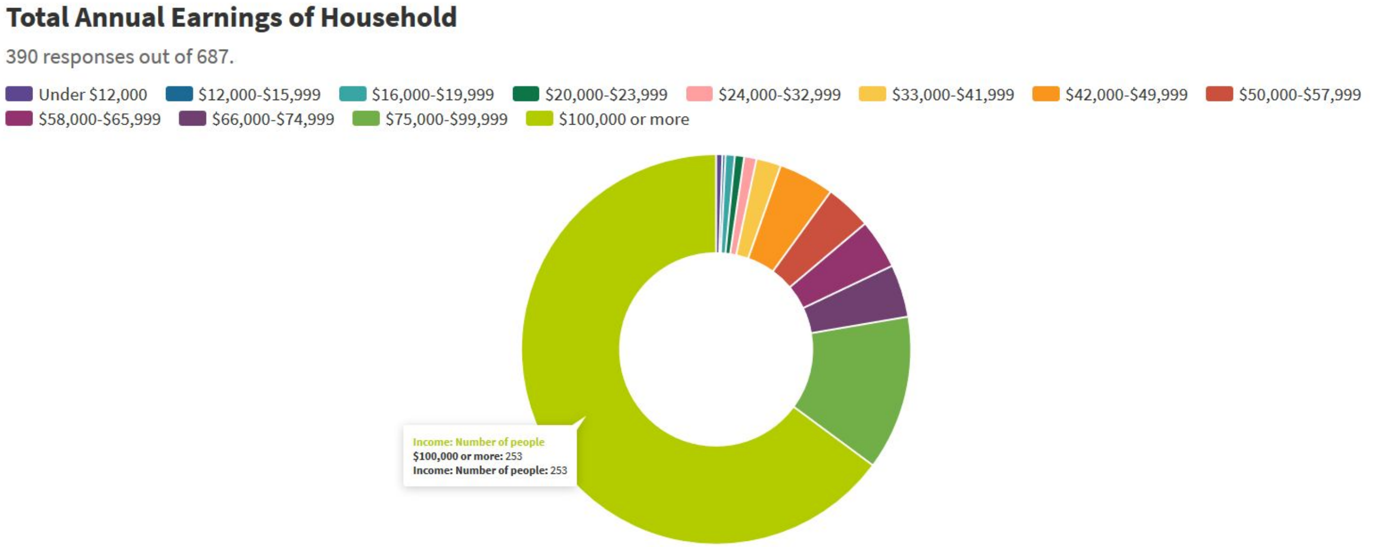

How many people are in your household (including yourself)?


What languages are regularly spoken in your home?


Do you consider yourself to have a disability?

Accessiblity features

Out of the people who considered themselves to have a disability, priority seating was the top feature used by most followed up closely by bus lowering feature. Users can benefit from accesibility features such as clear signage, announcements at park and ride facilities.

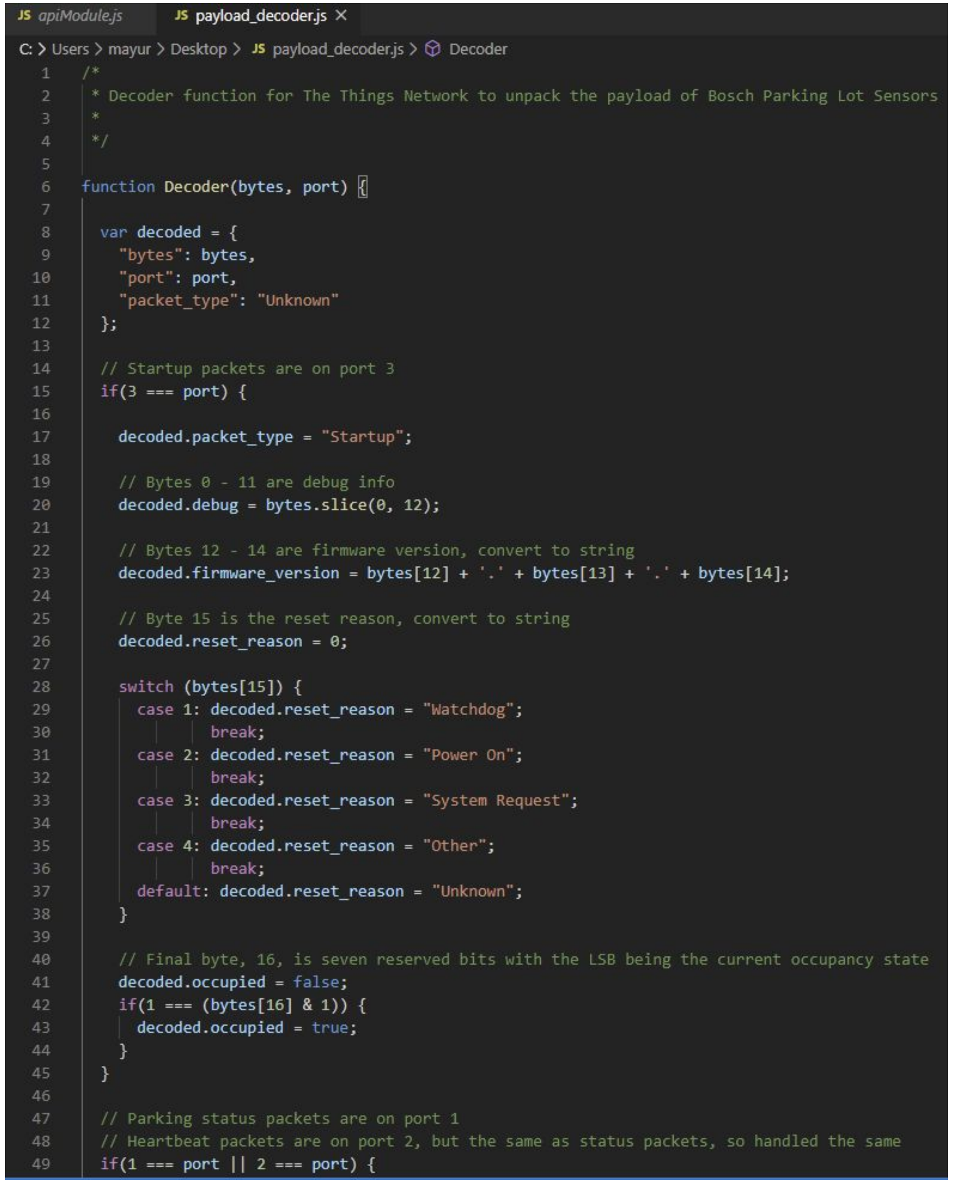

Total Annual Earnings of Household

390 responses out of 687.

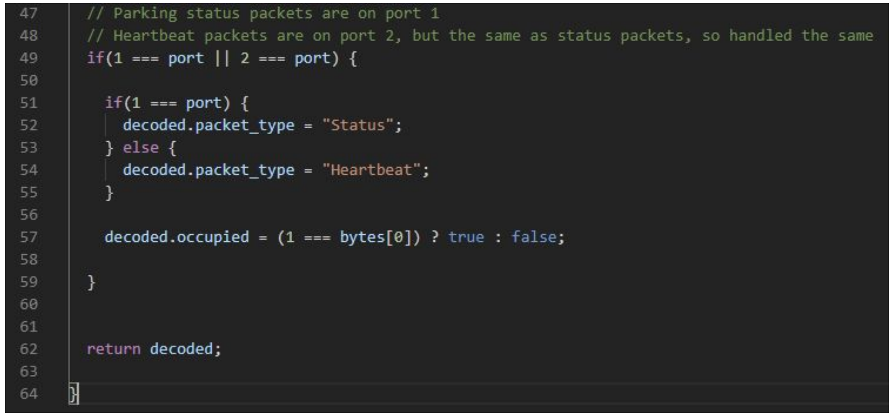
Source: Team EveREST Survey 2020. • 296 survey participants chose not to mention their total annual household income.

APPENDICES

Payload decoder function written in JavaScript



```
/*
 * Decoder function for The Things Network to unpack the payload of Bosch Parking Lot Sensors
 *
 */

function Decoder(bytes, port) {

  var decoded = {
    "bytes": bytes,
    "port": port,
    "packet_type": "Unknown"
  };

  // Startup packets are on port 3
  if(3 === port) {

    decoded.packet_type = "Startup";

    // Bytes 0 - 11 are debug info
    decoded.debug = bytes.slice(0, 12);

    // Bytes 12 - 14 are firmware version, convert to string
    decoded.firmware_version = bytes[12] + '.' + bytes[13] + '.' + bytes[14];

    // Byte 15 is the reset reason, convert to string
    decoded.reset_reason = 0;

    switch (bytes[15]) {
      case 1: decoded.reset_reason = "Watchdog";
              break;
      case 2: decoded.reset_reason = "Power On";
              break;
      case 3: decoded.reset_reason = "System Request";
              break;
      case 4: decoded.reset_reason = "Other";
              break;
      default: decoded.reset_reason = "Unknown";
    }

    // Final byte, 16, is seven reserved bits with the LSB being the current occupancy state
    decoded.occupied = false;
    if(1 === (bytes[16] & 1)) {
      decoded.occupied = true;
    }
  }

  // Parking status packets are on port 1
  // Heartbeat packets are on port 2, but the same as status packets, so handled the same
  if(1 === port || 2 === port) {
```



```
  // Parking status packets are on port 1
  // Heartbeat packets are on port 2, but the same as status packets, so handled the same
  if(1 === port || 2 === port) {
```

```

    if(1 === port) {
      decoded.packet_type = "Status";
    } else {
      decoded.packet_type = "Heartbeat";
    }

    decoded.occupied = (1 === bytes[0]) ? true : false;

  }


  return decoded;

}
```