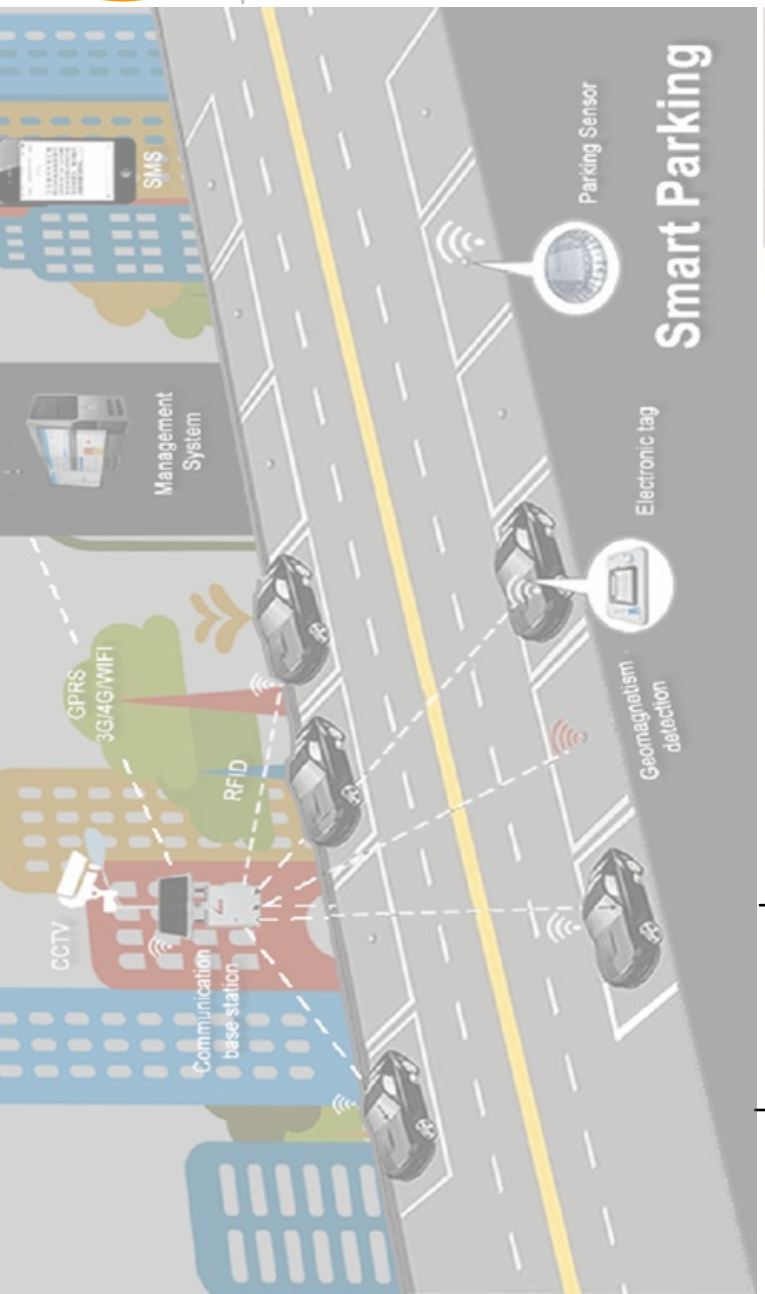
IoT Smart Parking.



Brijesh Prajapati

AWS Solutions Architect/DevOps
Orlando, FL
Mobile: (407) 405 7979
Email: it.brijesh@gmail.com

Brijesh is an enthusiastic software professional with 12+ years of hands-on experience in all phases of Software Development Lifecycle. He has worked as a developer, module leader, production support lead and Technical/Framework Architect & DevOps Engineer.

He has in-depth knowledge of DevOps tools & ecosystem and has helped implement various automated CI/CD pipelines. In addition to that he has extensive knowledge on Java/J2EE with hands-on working knowledge in Spring, Hibernate, Web services & related technologies and has designed, architected and implemented Web application frameworks.

He has in-depth understanding and working knowledge of AWS and has been actively building competency in IoT/IIoT, DevOps and Blockchain.

Industry Experience

- Industry 4.0/IIoT
- Energy, Gas & Utilities
- Public Sector – Tax & Employment Security

Education / Certifications

- B.E (Information Technology)
- AWS Certified Developer
- AWS Certified Solutions Architect

Relevant Experience

- Worked as AWS Solutions Architect for the design, review & re-architect of Industry 4.0 Smart factory solutions utilizing AWS Cloud web services and also migration of existing legacy applications on AWS.
- Hands on experience on setting up CICD pipeline for AWS Cloud Serverless Microservices including S3, DynamoDB, RDS, API Gateway, Websockets, IoT core, Events & Rules engine, Step function, Kinesis and related services utilizing AWS Code commit, build & pipeline for Blue/Green & Canary Deployment.
- Hands on experience on building the code using maven and setting up CICD Pipeline utilizing Jenkins, GitHub/BitBucket, Nexus/Jfrog/ECR Repositories for automated artifact store, Selenium/Sonar cube for automated testing.
- Hands on experience on building Docker containers for Java, NodeJS, Python & Spring microservices and deploying the same for orchestration into Kubernetes cluster for Blue/Green and Canary deployment.
- Hands on experience of deploying applications on AWS Cloud including EC2, S3/CloudFront & Container services such as ECS & EKS for Blue/Green & Canary Deployment including Auto Scaling & Load Balancing.
- Hands on experience on setting up AWS Cloud infrastructure using Ansible & CloudFormation including VPC, EC2, S3, RDS, SNS, ELK and related services.

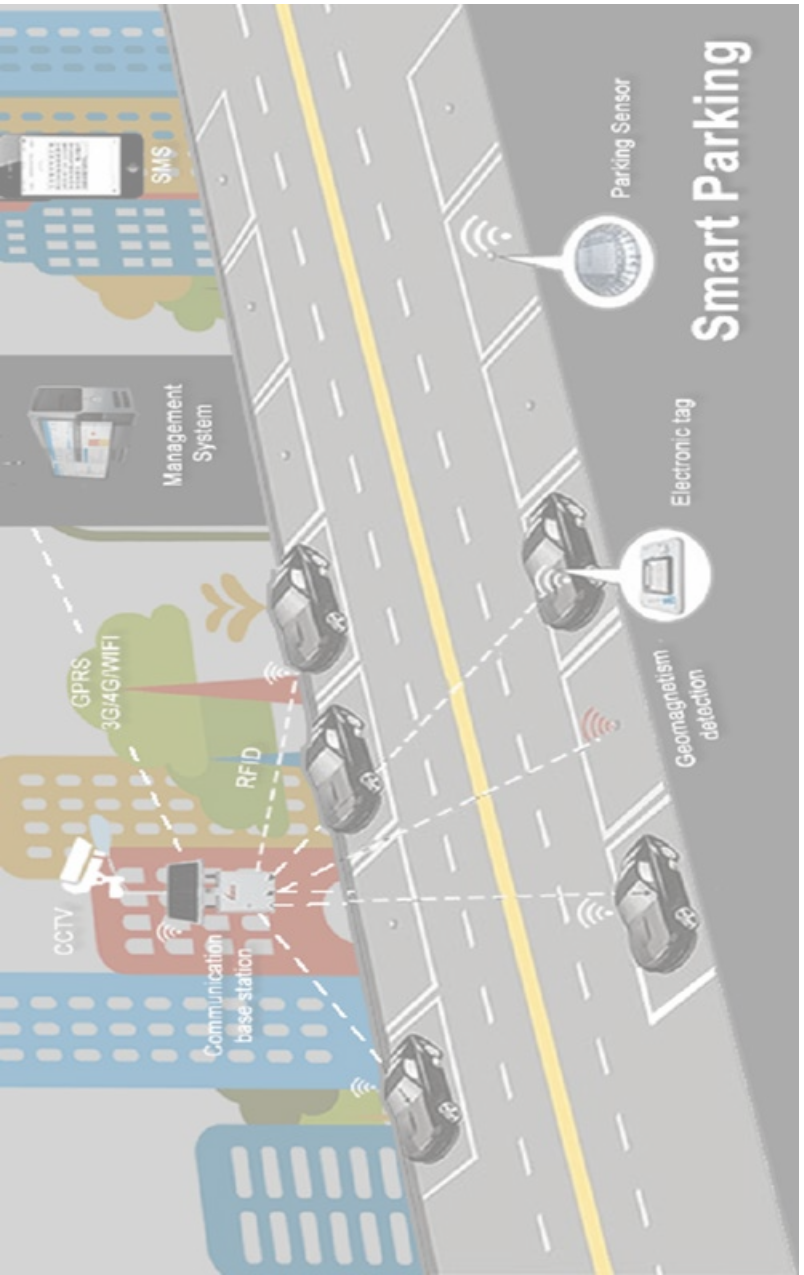
Tools / Technologies

- AWS Cloud – S3, DynamoDB, Lambda, API Gateway, RDS, EC2, DynamoDB, Kinesis – streams, firehose, analytics, IoT Core, Rule Engine, Greengrass, IoT Events, Step functions
- CloudFormation, AWS SAM, Ansible, Jenkins, AWS Code Commit, Build, Deploy & Pipeline, Nexus. Jfrog. GitHub/BitBucket, Confluence
- Java, J2EE, JSP, Servlet, Spring framework, Hibernate, Spring boot & Cloud with Config, Eureka Registry, Zuul, Security with Oauth & JWT.

Demo Topics

- ✓ Overview/Problem Description
- ✓ Software/Hardware/Cloud/Technology Specifications
- ✓ AWS Architecture – Advanced & Simplified (Part of Capstone project)
- ✓ Sequence/Data Flow diagram/Deployment architecture
- ✓ AWS Infrastructure Resources created
  - EC2, RDS, Lambda, DynamoDB
  - IoT Core, Greengrass, Rules Engine, IoT Events/State Machine
  - APIs, CloudFront
- ✓ DevOps CICD Pipelines
  - EKS Jenkins Pod, CloudFormation, Terraform
  - AWS CodePipelines
- ✓ Live Demo with Python Device Simulation scripts
  - Sensor Device Registration
  - Parking Occupied
  - IoT Event/State Machine model for Battery Status

IoT Smart Parking.



Problem Description:

The city of Round about has a downtown parking problem. Motorists find it hard to find parking even though there are many metered parking street spots in and around the downtown area.

The problem is worse during the peak demand hours of weekday mornings and weekend evenings.

The free parking spots are not always easy to locate.

This leads to a number of motorists roaming around looking for parking which makes the problem worse by creating additional traffic congestion.

Impact:

Help local Government to achieve goal of Smart City/Smart Nation

Company can build reusable solution and develop 3rd party APIs for custom app development

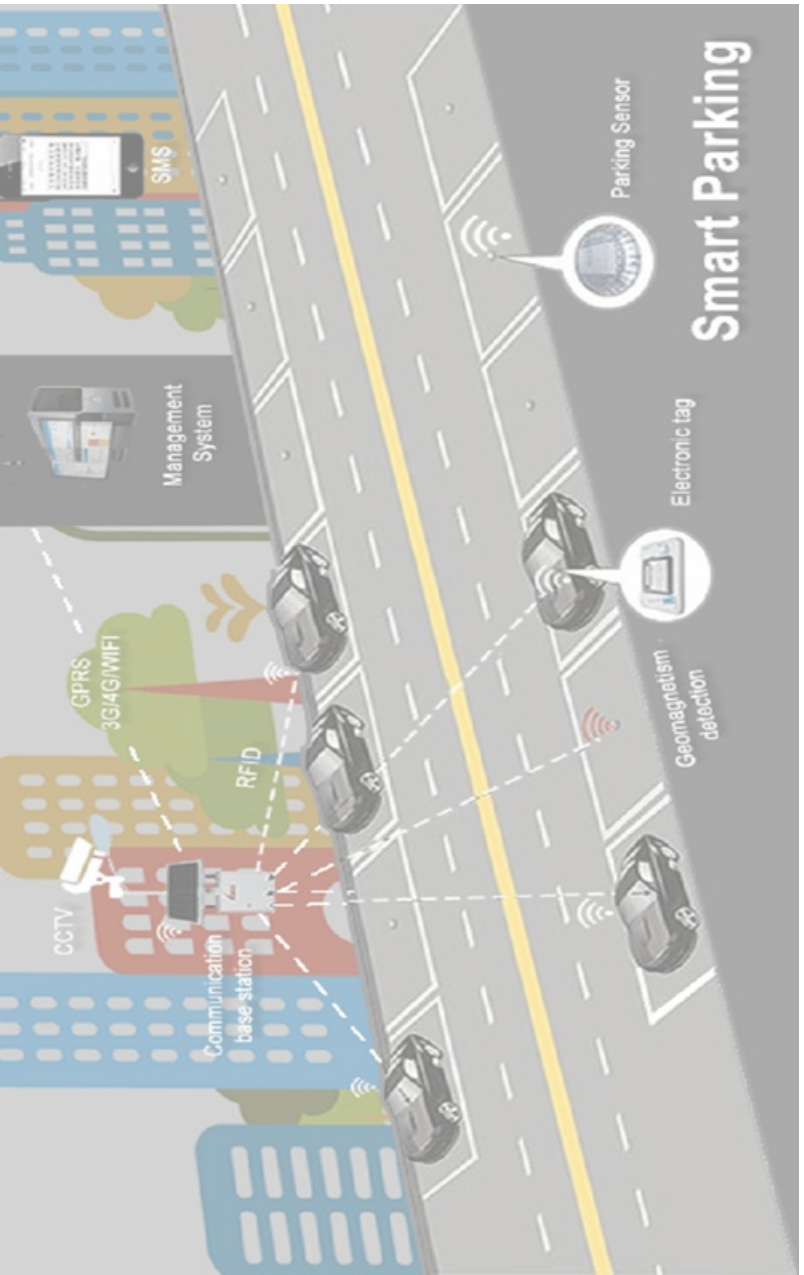
Alliance with AWS for Solution Marketing

Alliance with Sensor Device Hardware companies, Shipping Companies

Supply Chain Network? Digital Twin? …

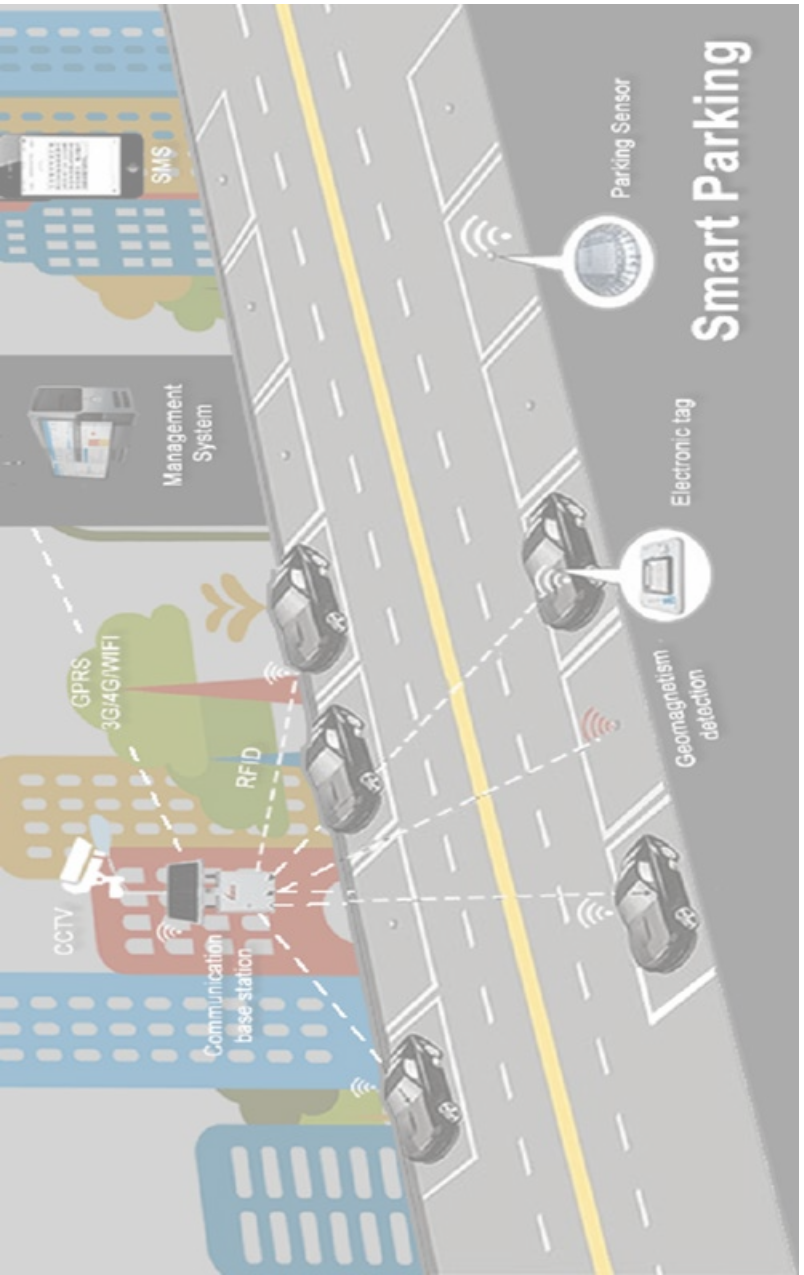
IoT Smart Parking.

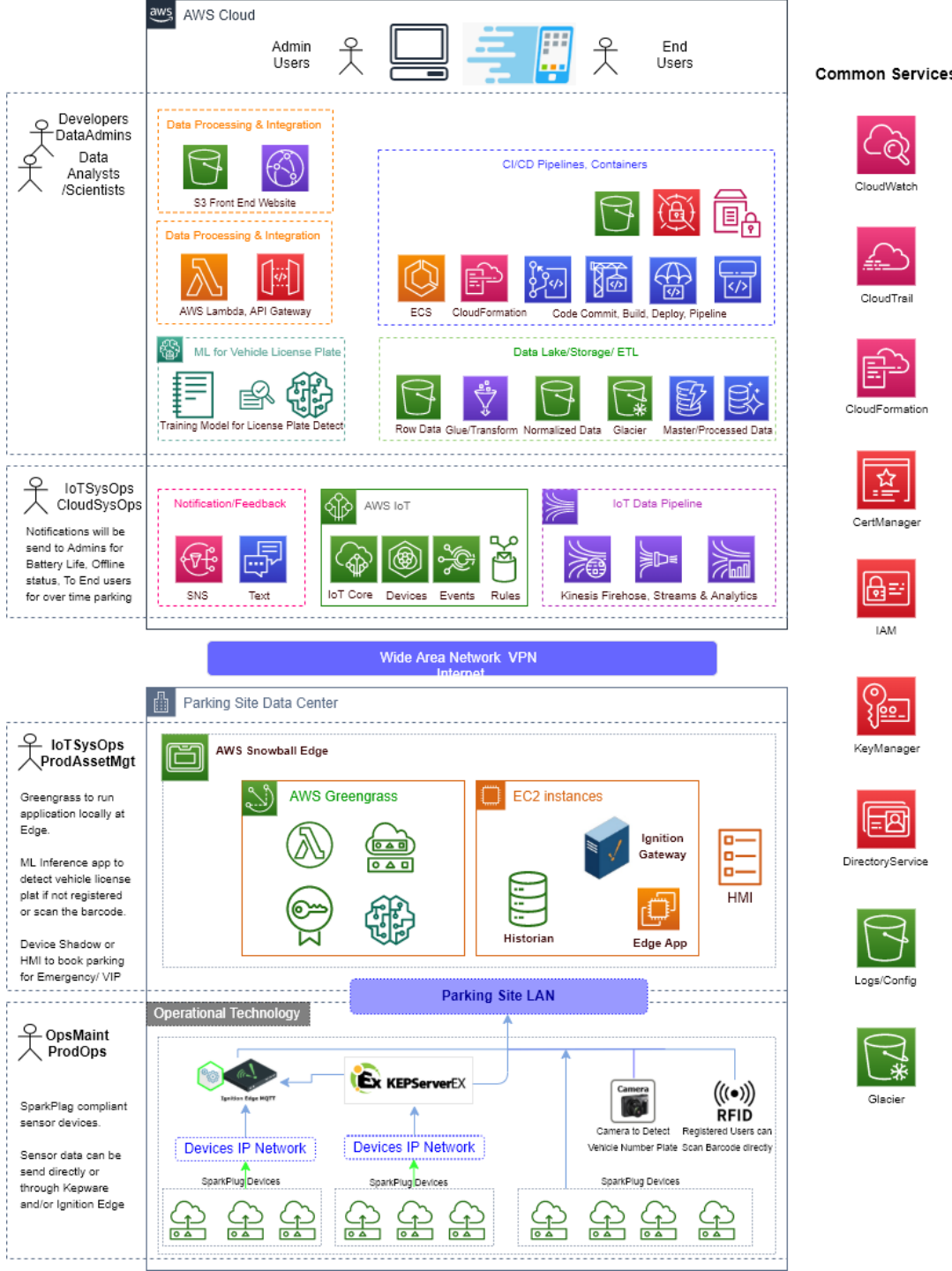
Software/Hardware/Cloud/Technology specification



- ✓ Hardware/Sensor technology to identify if a parking spot is occupied.
  - Microcontrollers
  - Ultrasonic/IR Distance Sensors (SparkPlug compliant devices)
  - Camera for Car object/Vehicle Number Plate detection
  - RFID Scanner for parking booking
- ✓ Connectivity technology to transmit current state of the parking spot.
  - LPWAN
- ✓ A scalable Sensor data processing/ingestion in AWS cloud to maintain parking data.
  - AWS Greengrass, IoT Core, Rule Engine, AWS Lambda
  - Ignition Edge & Gateway
  - ThingWorx Kepware server
  - Python scripts for Device Simulation
- ✓ Parking data is made available to 3rd party app developers to build interesting parking applications.
  - RDS, DynamoDB, AWS Lambda, IoT Events, API Gateway, S3, CloudFront
  - AWS Amplify NodeJS/Angular application
- ✓ Infrastructure & Code setup
  - Single Click deployment of infrastructure & code using CICD pipelines
  - EKS based Jenkins server, Terraform, CloudFormation, AWS CodePipeline
- ✓ Platform security & data protection.
  - MQTT, SSL, X.509, IAM Roles, Cognito/JWT Oauth
  - EC2 & RDS on Private Subnets only
  - AWS SSM for login to EC2/RDS

IoT Smart Parking.



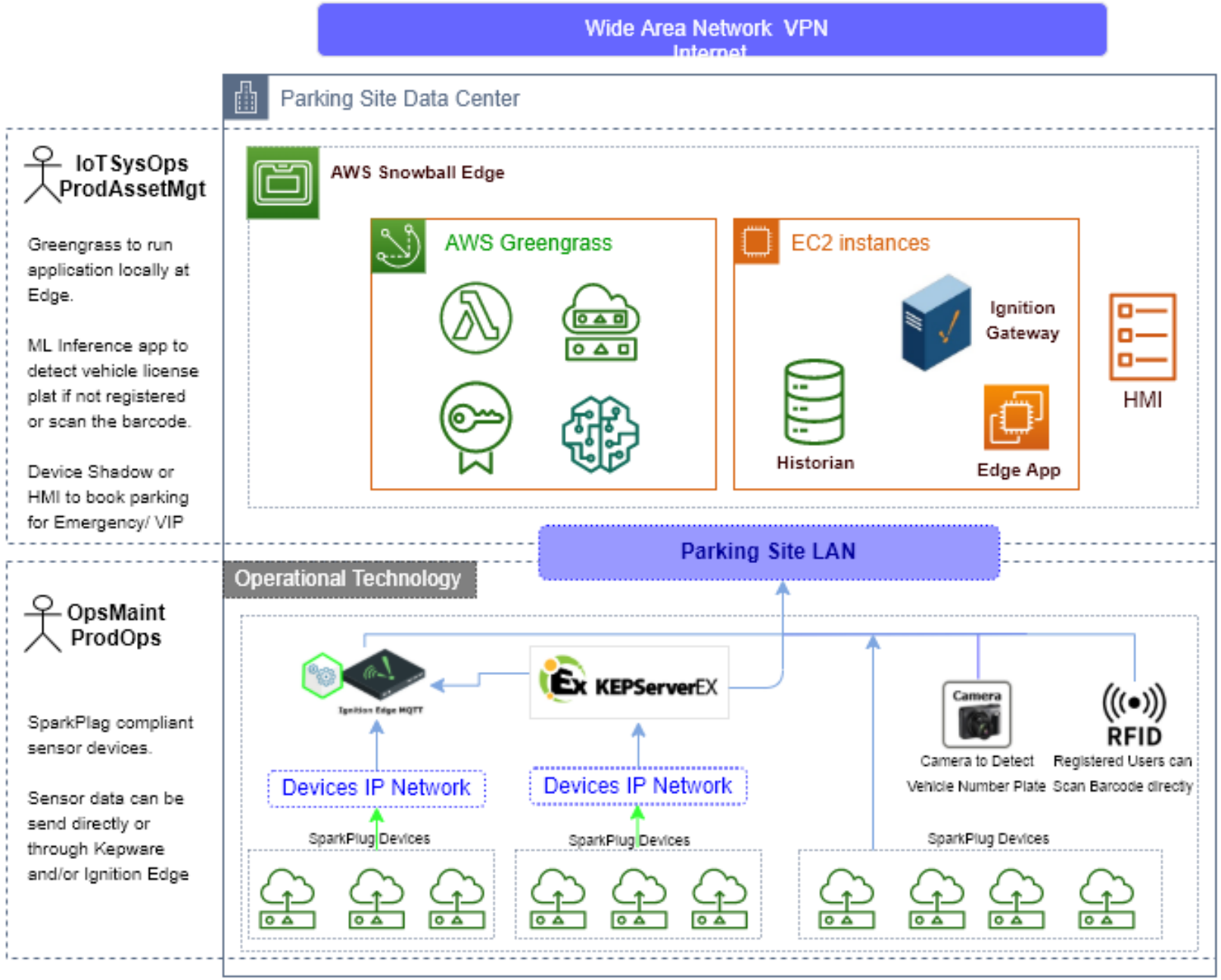


Common Services

- CloudWatch
- CloudTrail
- CloudFormation
- CertManager
- IAM
- KeyManager
- DirectoryService
- Logs/Config
- Glacier

IoT Smart Parking.

AWS IoT Advance Architecture




KeyManager

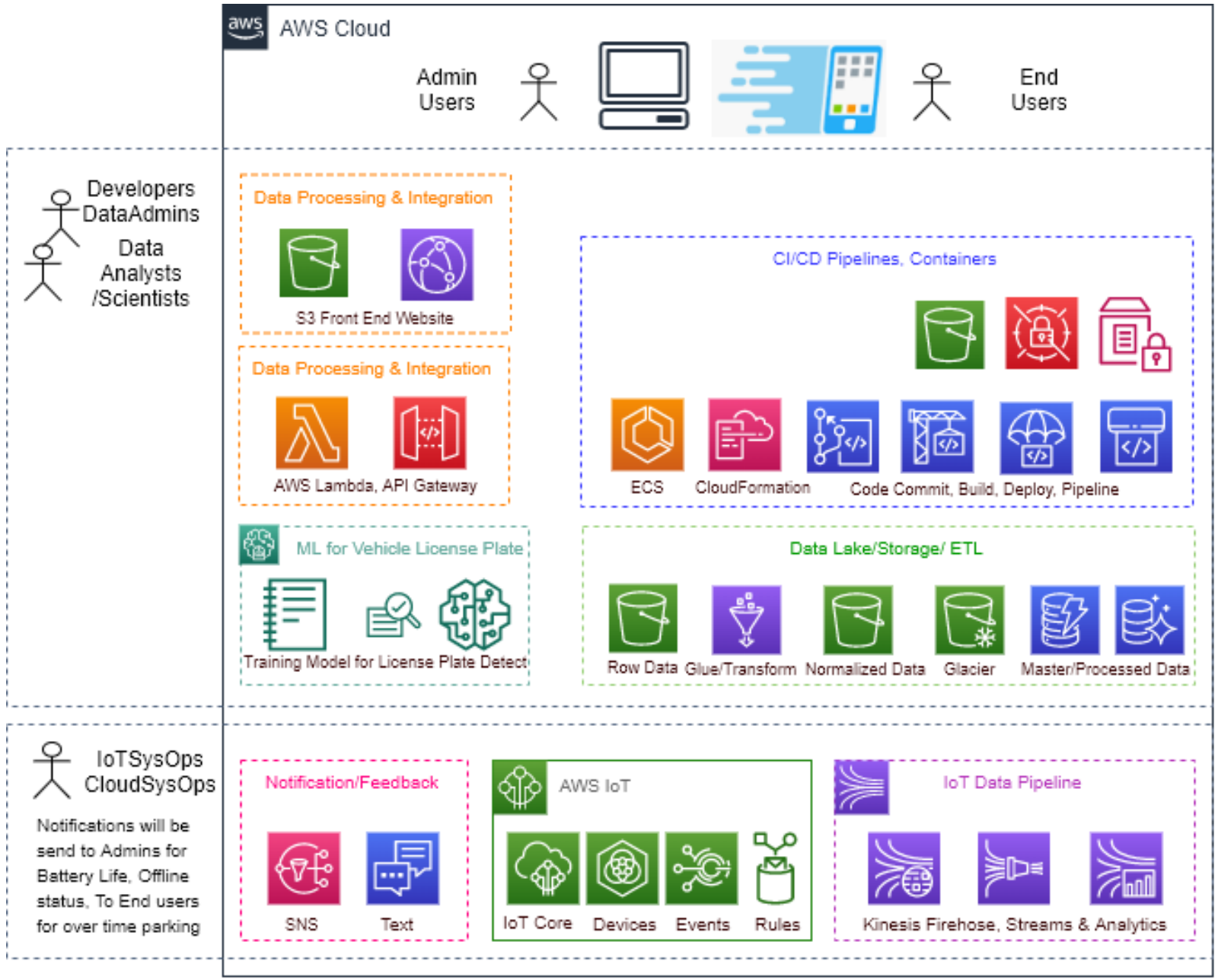

DirectoryService


Logs/Config


Glacier

IoT Smart Parking.

AWS IoT Advance Architecture



Common Services

CloudWatch

CloudTrail

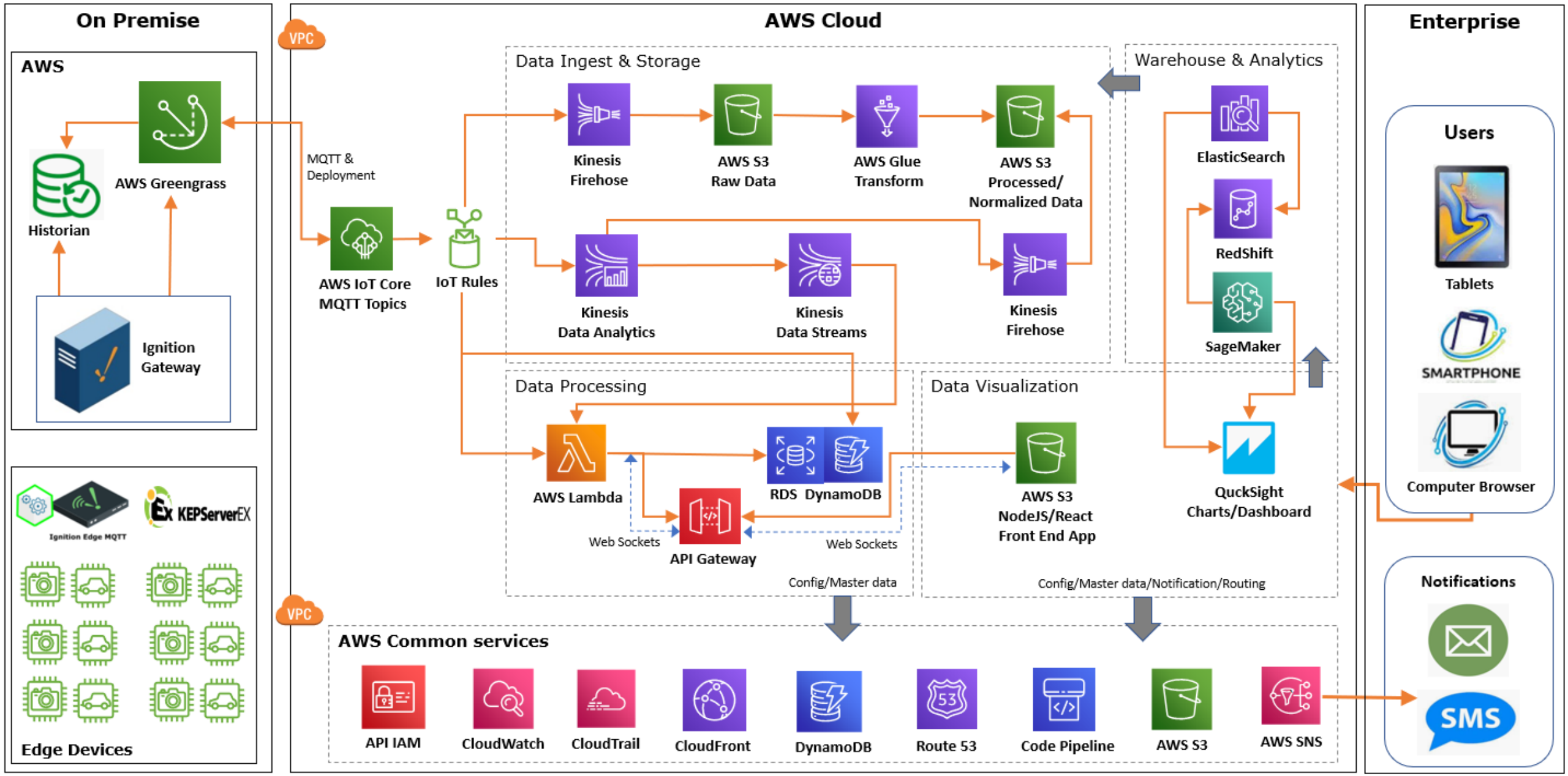
CloudFormation

CertManager

IAM

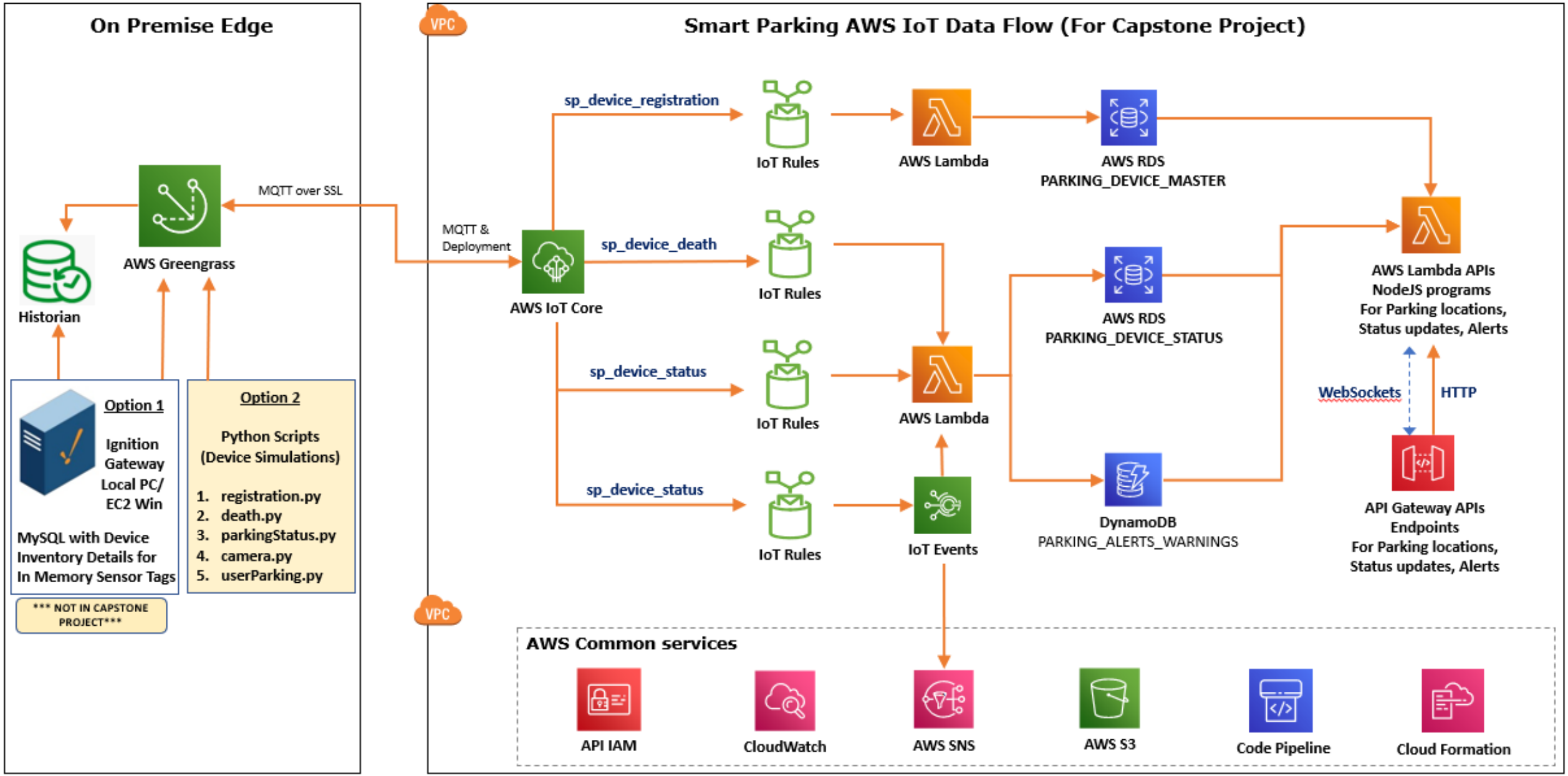
IoT Smart Parking.

AWS IoT Advanced Architecture



IoT Smart Parking.

AWS IoT Simple Architecture for Capstone Project



IoT Smart Parking.

Few Scenarios of sending Device Status Update Data:

- Sending sensor device battery life status at regular frequency.
  - Sensor Data: BatteryLife, serial_number (device unique id)
  - Greengrass code can be updated to control data send to AWS Cloud. (Status threshold lower limit = 40, upper = 90)
- Sending parking occupancy status (IsOccupied) on event change.
  - Sensor Data: IsOccupied, serial_number
- Sending user action on parking. Registered user will scan and provide parking hours.
  - Sensor Data: user_code, parking_hours, serial_number
- Edge Camera Inference will be used to detect object to double check parking is available. (every 15 minutes)
  - (Helpful if device sensor not working due to some reason, OR Someone trying to cheat device)
  - Sensor Data: IsOccupied, serial_number
- Edge Camera Inference will be used to detect Vehicle License Plate Number if user did not check-in parking.
  - Sensor Data: user_code (license plate number), parking_hours (default 1 hour), serial_number

IoT Smart Parking.

AWS IoT Events can be used to log various Parking/Device Events:

- Parking Device Distance Sensor Low Battery Power/Life
- Past Parking Hours
- Parking Edge/Device Maintenances
- Parking without user scan or registration
- Parked Vehicle License Plate Number not identified
- Parked Vehicle License Plate Number mismatch with user registration
- Parking Device Camera not working
- Parking Device Distance Sensor not working

Note: Have implemented first event only as a part of this IoT sample assignment.

Future Scope:

- Need to implement this IoT application based on the proposed Advance Architecture.
- Inform user for over time parking via Text/Call. Add support to extend the parking hours via Text/Call.
- Call Tow Away company for overtime parking violations to Tow vehicle and inform user for the same.
- Detect the nearest parking via user App through GPS.
- Voice support to find nearest parking site while driving. User has to go actual location for confirmed parking. (Online booking will not be supported)
- Handling Emergency & VIP Parking requests. (Device Shadow or similar technique)
- Creating batch programs to calculate dynamic parking rates.
  - Users parking vehicle without registration will have more rates as system has to detect the vehicle license plate number and have to call respective 3rd party services to identify user
  - Users providing proper parking hours and following the parking timings should get discounted rates.
- Machine Learning Vision IoT Inferencing & Training modules for Vehicle Object detection & License Plate Number identification.
- Support of Blockchain to log user parking events & cryptocurrency payments.

IoT Smart Parking.

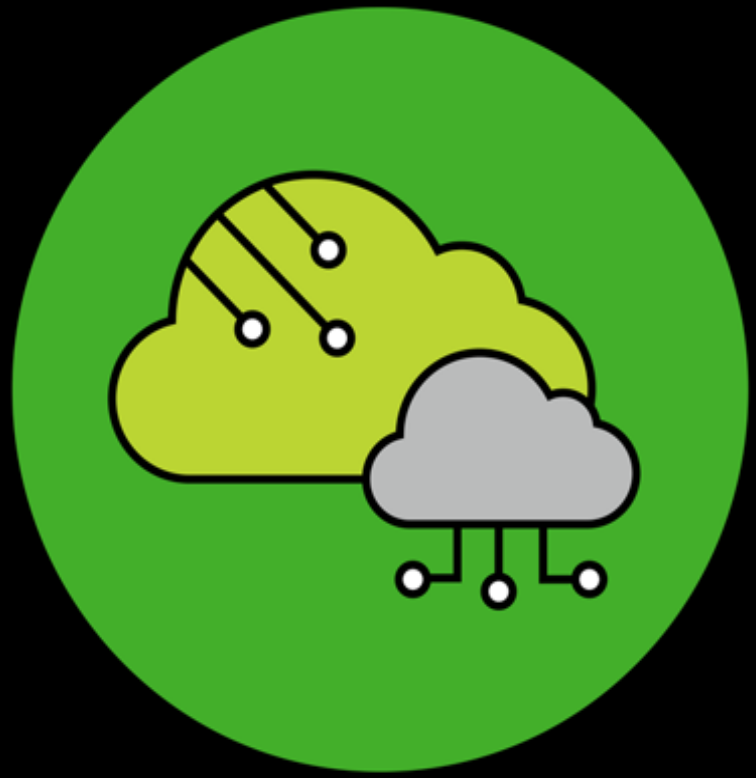
Lessons learn :

- Code/Infrastructure Automation as I had to move between accounts
- IoT Event State Machine
- AWS Amplify framework for Web app development & Authentication
- Connecting to Private EC2 & RDS using SSM & SSH tunneling

Next Action Item :

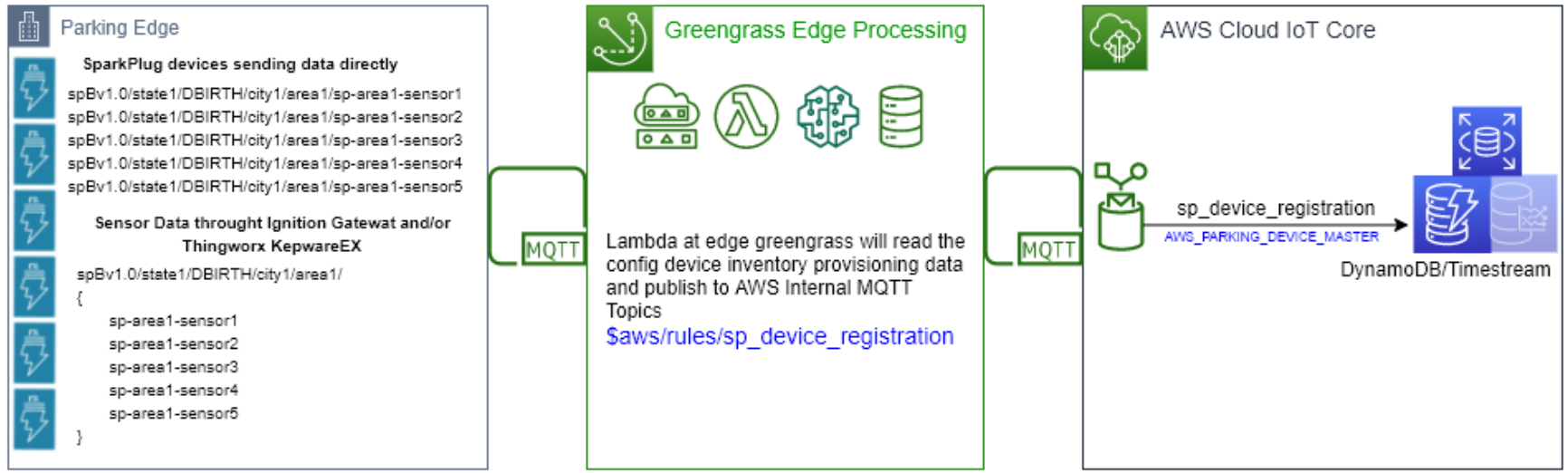
- **Connect to IT.BRIJESH@GMAIL.COM for Simulation Demo** 

Thank You!



IoT Smart Parking.

Data Ingest: Device Registration / DBIRTH



Sparkplug Device Sensor Sample Data OPC Format

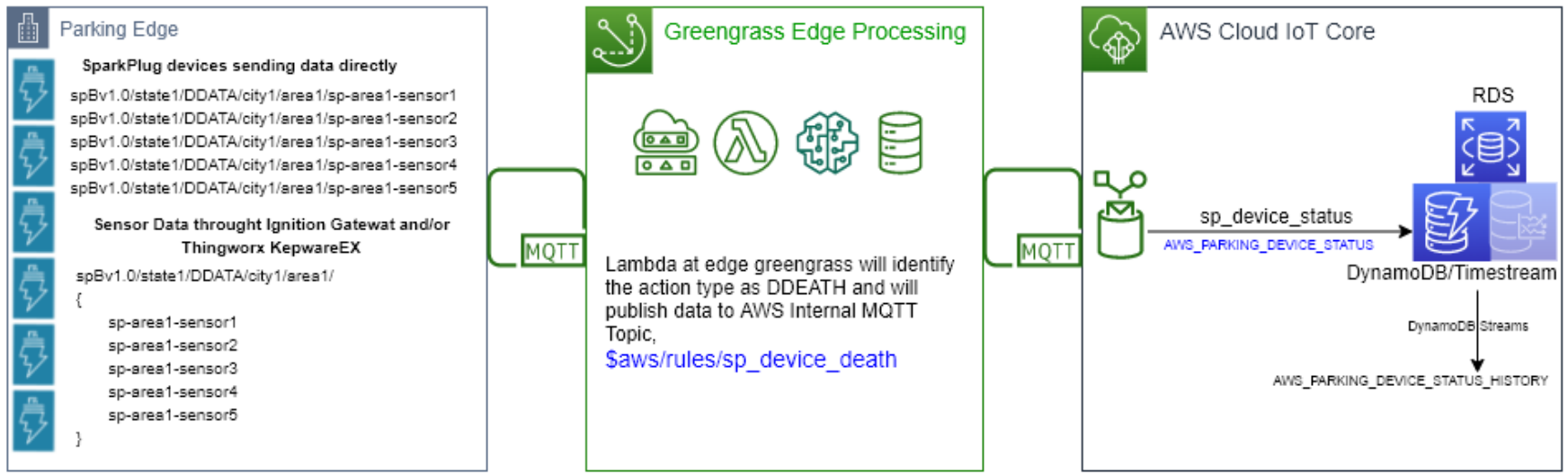

```
{
    invokedFunctionArn": "arn:aws:lambda:us-east-1:441518958191:
                          function:gg-stack-gglambda-O0Z98BR7P84O:13",
    "awsRequestId": "c78fc6a2-189e-4e7e-6d52-284cc0df3425",
    "functionName": "gg-stack-gglambda-O0Z98BR7P84O",
    "functionVersion": "13",
    "clientContext": {
    "client": {},
    "Custom": {
    "subject": "spBv1.0/state1/DBIRTH/city1/area1"
}
}{
"metrics": [
{
    "name": "sp-area1-sensor1/BatteryLife","value": "90%"
},
{
    "name": "sp-area1-sensor1/IsOccupied","value": 0
}
],
    "timestamp": 1572811716.680315
}
```


Transformed Simple JSON data


```
{
    "serial_number": "sp-area1-sensor1",
    "asset": "smart_parking",
    "asset_type": "sensor",
    "iot_birth_rule_cloud": "$aws/rules/sp_device_registration",
    "iot_data_rule_cloud": "$aws/rules/sp_device_status",
    "iot_birth_rule_edge": "spBv1.0/state1/DBIRTH/city1/area1/sp-area1-sensor1",
    "iot_data_rule_edge": "spBv1.0/state1/DDATA/city1/area1/sp-area1-sensor1",
    "iot_death_rule_edge": "spBv1.0/state1/DDEATH/city1/area1/sp-area1-sensor1",
    "iot_death_rule_cloud": "$aws/rules/sp_device_death",
    "guid": "8adee008c0734419abc4f4a8e3f86938",
    "location": "[\"-75.5712\", \"-130.5355\"]",
    "address": "2516 White Magnolia Way, Sanford, FL 32771",
    "meter": "1",
    "timestamp": 1572811716.680315,
    "BatteryLife": "90%",
    "IsOccupied": 0
}
```

IoT Smart Parking.

Data Ingest: Device Offline / DDEATH



Sparkplug Device Sensor Sample Data OPC Format

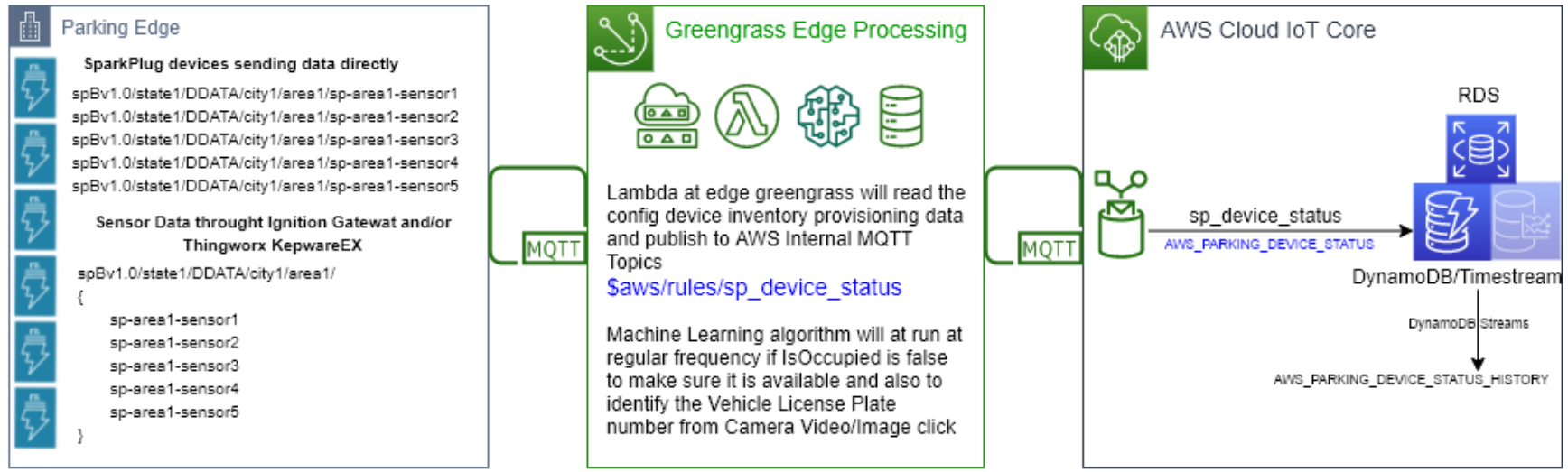

```
{
    invokedFunctionArn": "arn:aws:lambda:us-east-1:441518958191:
                          function:gg-stack-gglambda-O0Z98BR7P84O:13",
    "awsRequestId": "c78fc6a2-189e-4e7e-6d52-284cc0df3425",
    "functionName": "gg-stack-gglambda-O0Z98BR7P84O",
    "functionVersion": "13",
    "clientContext": {
    "client": {},
    "Custom": {
    "subject": "spBv1.0/state1/DDEATH/city1/area1"
}
}{
"metrics": [
{
    "name": "sp-area1-sensor1/BatteryLife","value": "90%"
},
{
    "name": "sp-area1-sensor1/IsOccupied","value": 0
}
],
    "timestamp": 1572811716.680315
}
```


Transformed Simple JSON data


```
{
    "serial_number": "sp-area1-sensor1",
    "guid": "8adee008c0734419abc4f4a8e3f86938",
    "status": "offline",
    "reason": "***Device Maintanance***",
    "timestamp": 34343435.64446
}
```

IoT Smart Parking.

Data Ingest: Device Registration / DDATA



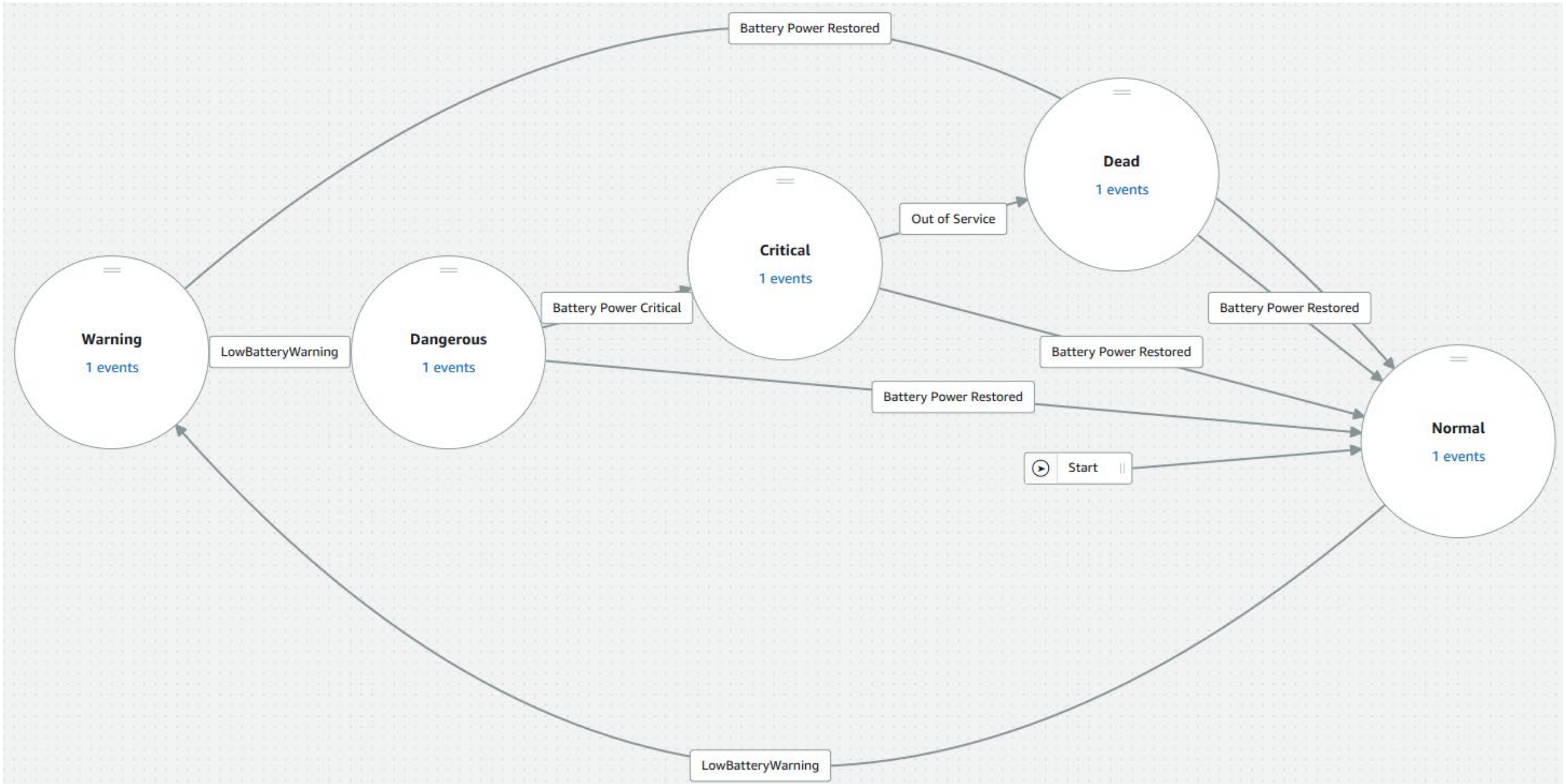
Sparkplug Device Sensor Sample Data OPC Format


```
{
    invokedFunctionArn": "arn:aws:lambda:us-east-1:441518958191:
                          function:gg-stack-gglambda-O0Z98BR7P84O:13",
    "awsRequestId": "c78fc6a2-189e-4e7e-6d52-284cc0df3425",
    "functionName": "gg-stack-gglambda-O0Z98BR7P84O",
    "functionVersion": "13",
    "clientContext": {
    "client": {},
    "Custom": {
    "subject": "spBv1.0/state1/DDATA/city1/area1"
}
}{
"metrics": [
{
    "name": "sp-area1-sensor1/BatteryLife","value": "70%"
},
{
    "name": "sp-area1-sensor1/IsOccupied","value": 1
}
],
    "timestamp": 1572811716.680315
}
```


Transformed Simple JSON data


```
{
    "serial_number": "sp-area1-sensor1",
    "guid": "8adee008c0734419abc4f4a8e3f86938",
    "status": "online",
     "IsOccupied": 1,
     "BatteryLife": "70%",
    "timestamp": 1572811716.680315
}
```

AWS IoT Events Device Battery Status Detector Model



AWS_PARKING_ALERTS_WARNINGS (Testing Data)

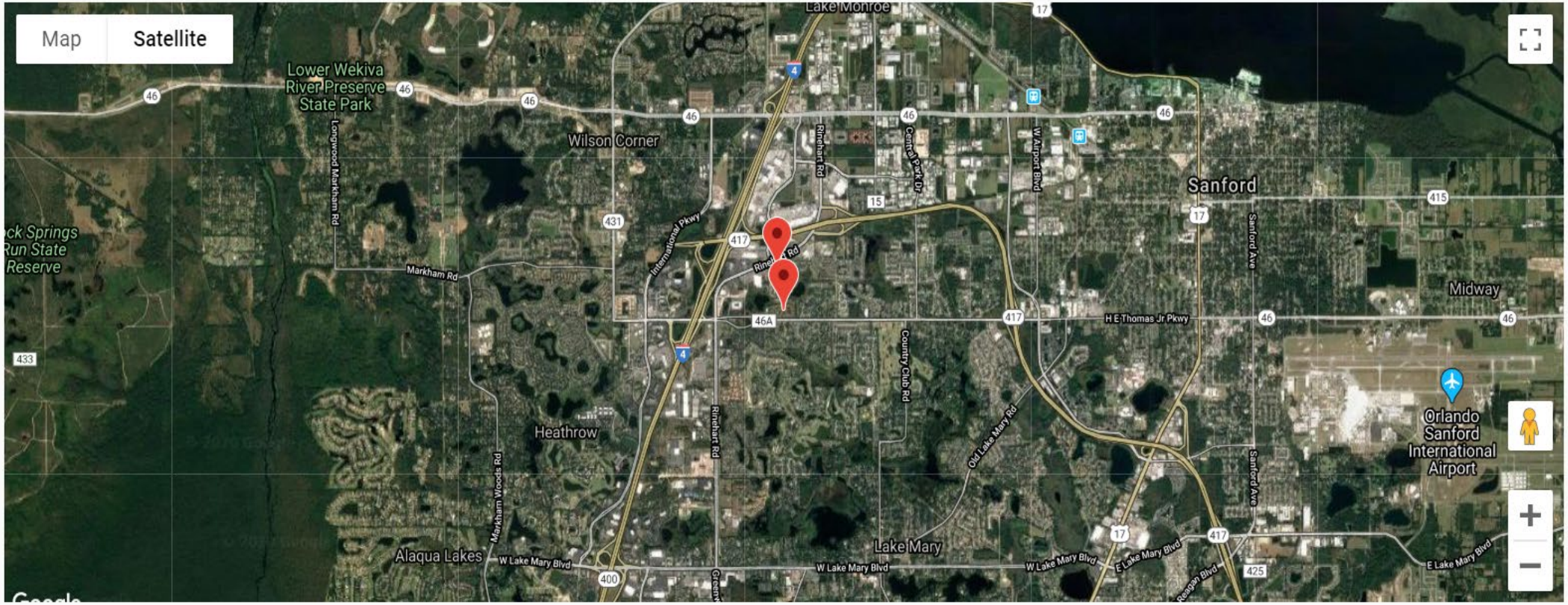
serial_number	timestamp	alert_id	alert_name	data	event	state
sp-area1-sensor1	1574096091820	1573919799958	BatteryStatus	BatteryLife : 0	Out of Service	Dead
sp-area1-sensor1	1574096031463	1573919799958	BatteryStatus	BatteryLife : 15	Low BatteryLife Critical	Critical
sp-area1-sensor1	1574095932172	1573919799958	BatteryStatus	BatteryLife : 25	Low BatteryLife Dangerous	Dangerous
sp-area1-sensor1	1574095858007	1573919799958	BatteryStatus	BatteryLife : 35	Low BatteryLife Warning	Warning

IoT Smart Parking.

Smart Parking

- Home
- Parking Details
- Device/Parking Alerts
- Register Parking Location

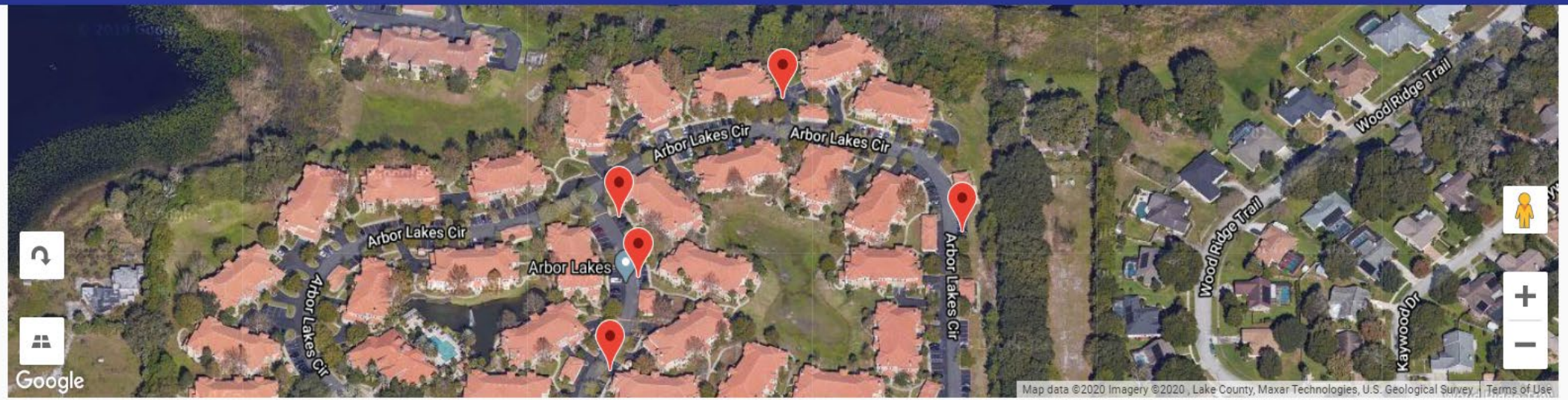
Sr. No	Parking Name	Address	Parking Details
1	Arbor Lakes Parking	100 Arbor Lakes Circle, Sanford FL 32771	GO
2	Lakes Edge Parking	1000 Plantation Lakes Cir, Sanford FL 32771	GO



IoT Smart Parking.

Smart Parking

- Home
- Parking Details
- Device/Parking Alerts
- Register Parking Location



Get Parking Details

Sr. No	Address	Is Available?	Start Time	End Time	Meter No	GPS Location	Battery Life	User Code
1	1325 Arbor Lakes Circle, Sanford FL 32771	No	12:53 AM GMT-4	1:53 AM GMT-4	1	Map	0	CGUILD9
2	2516 White Magnolia Way, Sanford FL 32771	Yes			2	Map	100	AWS094
3	1726 Arbor Lakes Circle, Sanford FL 32771	Yes			3	Map	100	
4	2612 White Magnolia Way, Sanford FL 32771	Yes			4	Map	100	
5	2823 White Magnolia Way, Sanford FL 32771	Yes			5	Map	100	

IoT Smart Parking.

Smart Parking

- Home
- Parking Details
- Device/Parking Alerts
- Register Parking Location

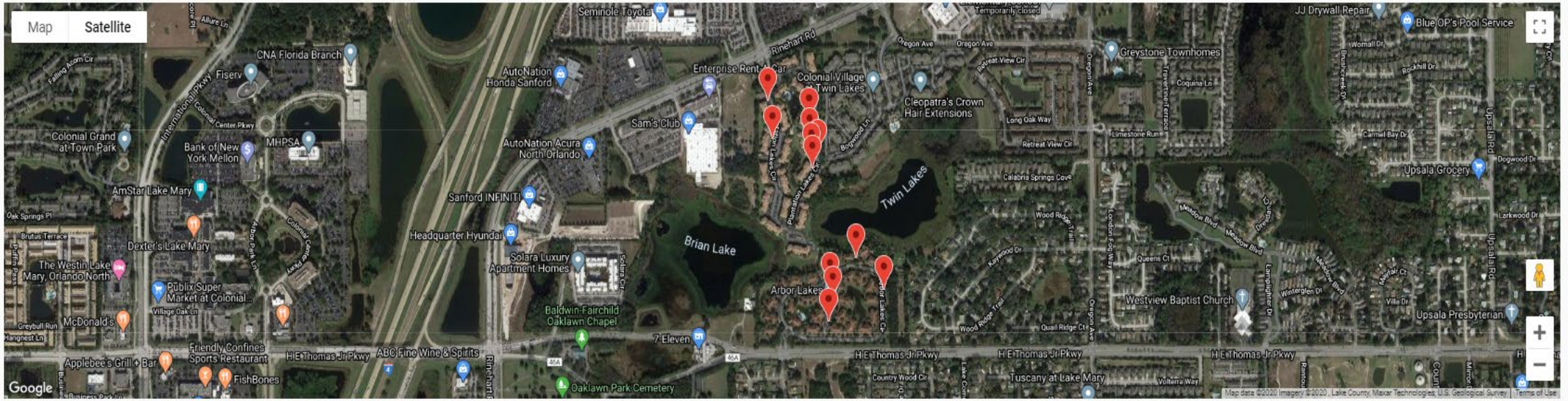
Get Device/Parking Alerts

Sr. No	Alert Type	Serial Number	Event Name	Status	Time	Address	zipcode
1	BatteryStatusChangeModel	sp-area1-sensor1	Out of Service	Dead	1:11 AM GMT-4	1325 Arbor Lakes Circle, Sanford, FL	32771
2	BatteryStatusChangeModel	sp-area1-sensor1	Low BatteryLife Warning	Warning	1:09 AM GMT-4	1325 Arbor Lakes Circle, Sanford, FL	32771
3	BatteryStatusChangeModel	sp-area1-sensor1	Low BatteryLife Dangerous	Dangerous	1:10 AM GMT-4	1325 Arbor Lakes Circle, Sanford, FL	32771
4	BatteryStatusChangeModel	sp-area1-sensor1	Battery Power Restored	Normal	1:09 AM GMT-4	1325 Arbor Lakes Circle, Sanford, FL	32771
5	BatteryStatusChangeModel	sp-area1-sensor1	Low BatteryLife Critical	Critical	1:10 AM GMT-4	1325 Arbor Lakes Circle, Sanford, FL	32771

IoT Smart Parking.

Smart Parking

- Home
- Parking Details
- Device/Parking Alerts
- Register Parking Location



Get Parking Details

Sr. No	Address	Is Available?	Start Time	End Time	Meter No	GPS Location	Battery Life	User Code
1	1325 Arbor Lakes Circle, Sanford FL 32771	No	12:53 AM GMT-4	1:53 AM GMT-4	1	Map	0	CGUILD9
2	2516 White Magnolia Way, Sanford FL 32771	Yes			2	Map	100	AWS094
3	1726 Arbor Lakes Circle, Sanford FL 32771	Yes			3	Map	100	
4	2612 White Magnolia Way, Sanford FL 32771	Yes			4	Map	100	
5	2823 White Magnolia Way, Sanford FL 32771	Yes			5	Map	100	
6	12208 Plantation Lakes Cir, Sanford FL 32771	Yes			1	Map	100	
7	12200 Plantation Lakes Cir, Sanford FL 32771	Yes			2	Map	100	
8	12247 Plantation Lakes Cir, Sanford FL 32771	Yes			3	Map	100	

IoT Smart Parking.