

NeoCharge

December 05, 2019



The Juicerz

Josh Boe | Casey Daly | Lauren Hibbs
Hannah Kwan | Pranathi Guntupalli

Problem

- NeoCharge is a smart splitter designed to increase the availability of electric vehicle charging at home
- NeoCharge Mobile Application:
 - Allow for a customizable charging experience
 - Gain insights into your charging process



Goals

Cross-Platform Application

Charge Scheduling via Weekly Calendar

Remote Control of Charging

Push Notifications

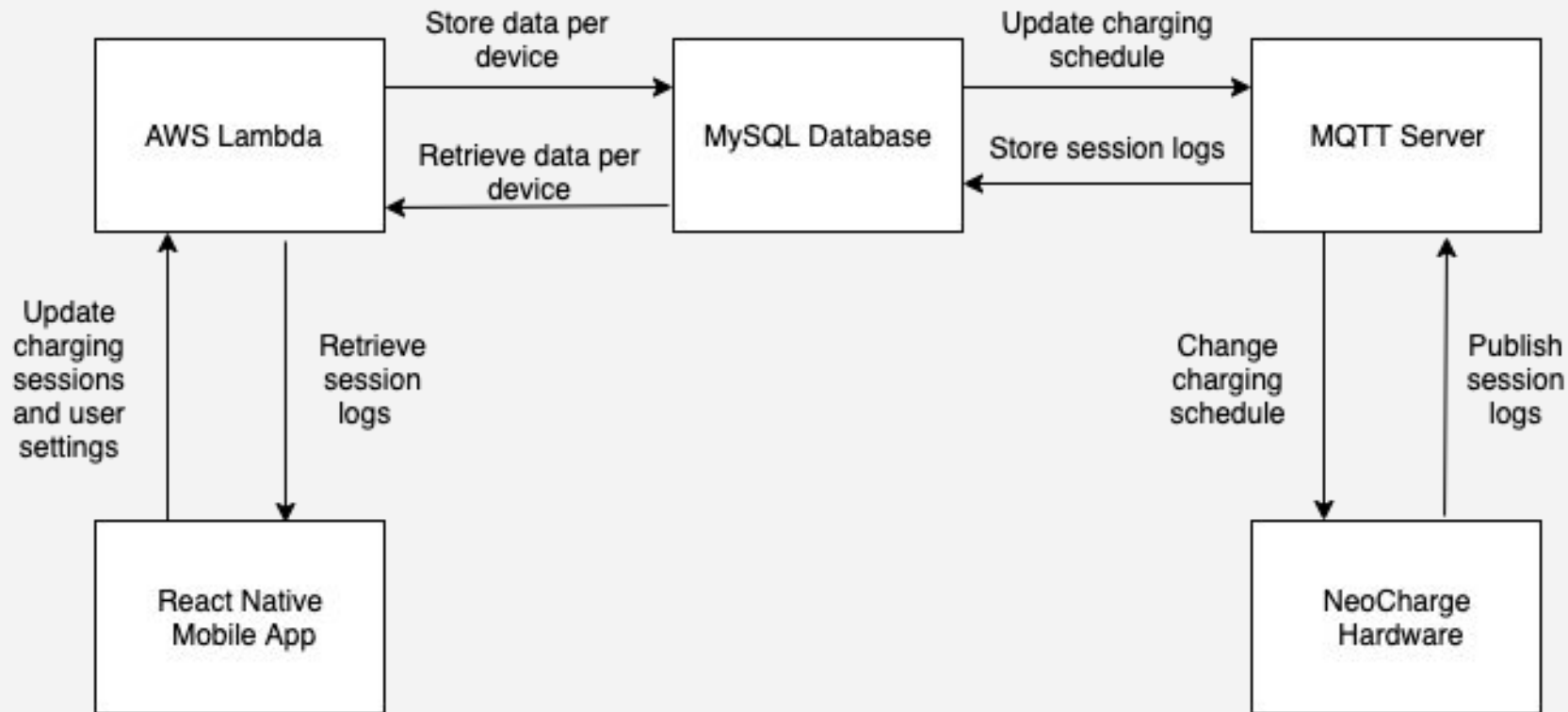
Charge History

Green Usage History - WattTime

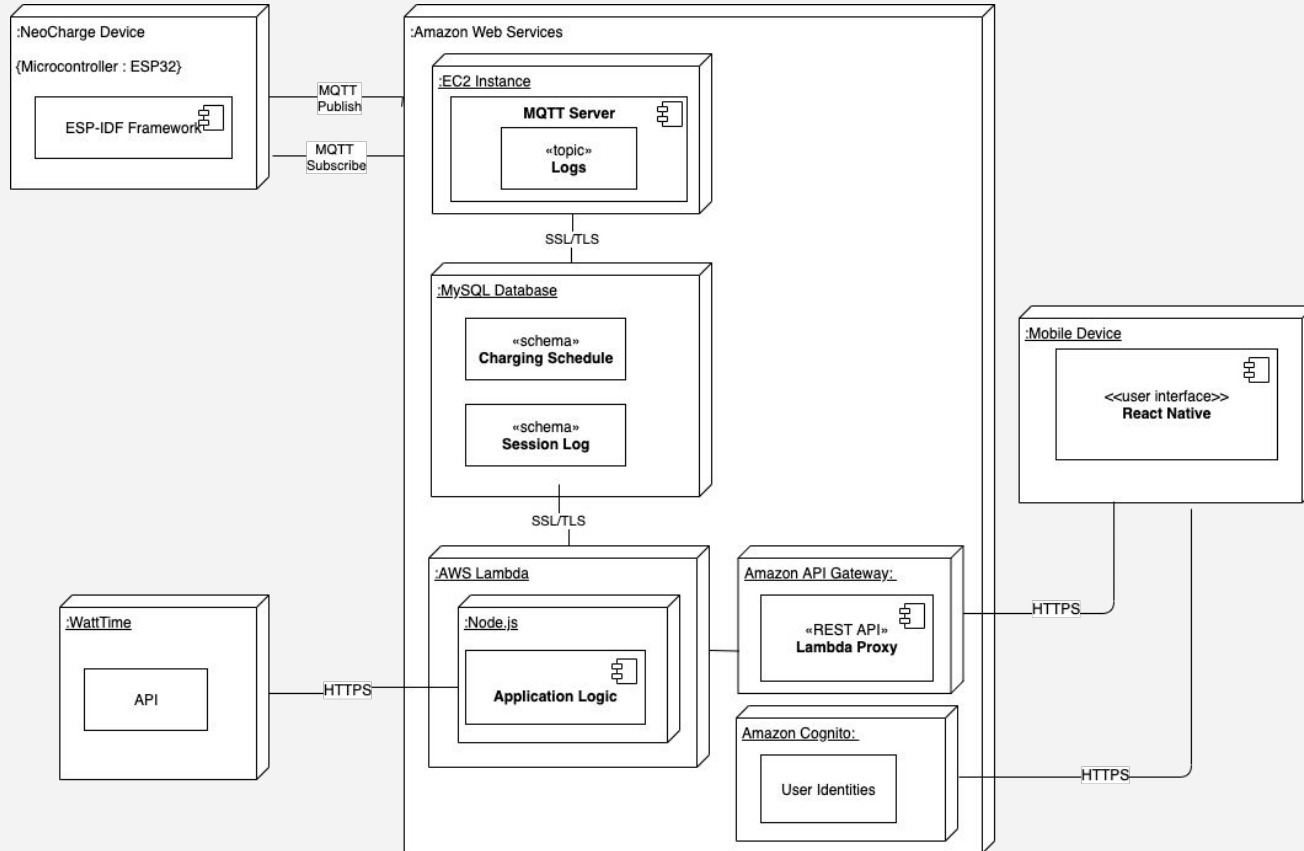
Horizontal Prototype Walkthrough

Software Architecture

Data Flow

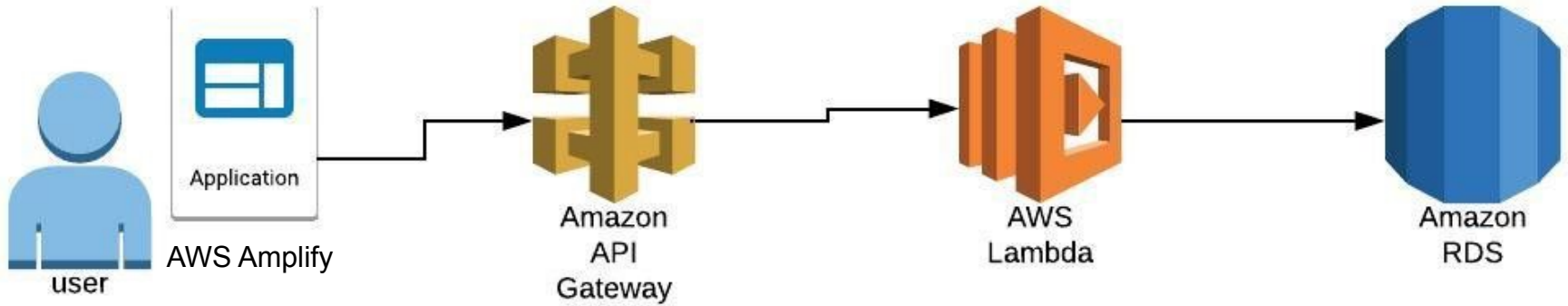


Deployment



Vertical Prototype Walkthrough

Amazon Services





Services ▾

Resource Groups ▾



TheJuicerz ▾

Oregon ▾

Support ▾

AWS Lambda ×

- Dashboard
- Applications
- Functions
- Layers

Lambda > Functions

Functions (2)



Actions ▾

Create function

Filter by tags and attributes or search by keyword



1



	Function name ▾	Description	Runtime ▾	Code size ▾	Last modified ▾
<input type="radio"/>	addusers-test		Python 3.8	111.0 kB	14 days ago
<input type="radio"/>	postToDb-test		Python 3.8	111.0 kB	1 hour ago



Services ▾

Resource Groups ▾



TheJuicerz ▾

Oregon ▾

Support ▾

postToDb-test

Throttle

Qualifiers ▾

Actions ▾

queryDbWParams ▾

Test

Save

File Edit Find View Go Tools Window

Environment

- postToDb-test
 - pymysql
 - PyMySQL-0.9.3.dist-info
 - postToDb-test.py
 - rds_config.py

postToDb-test.py

```

1 import sys
2 import logging
3 import rds_config
4 import pymysql
5 import json
6 import datetime
7 #rds settings
8 rds_host = rds_config.rds_host
9 name = rds_config.rds_username
10 password = rds_config.rds_password
11 db_name = rds_config.rds_db_name
12
13 logger = logging.getLogger()
14 logger.setLevel(logging.INFO)
15
16 try:
17     connection = pymysql.connect(rds_host, user=name, passwd=password, db=db_name, connect_timeout=5, cursorclass=pymysql.cursors.DictCursor)
18 except pymysql.MySQLError as e:
19     logger.error("ERROR: Unexpected error: Could not connect to MySQL instance.")
20     logger.error(e)
21     sys.exit()
22
23 logger.info("SUCCESS: Connection to RDS MySQL instance succeeded")
24 def handler(event, context):
25
26     infos = ""
27
28     deviceId = event["queryStringParameters"]["deviceId"]
29
30     query = 'select * from NeoCharge.ChargeLogs where deviceId = \'' + deviceId + '\'';
31
32     with connection.cursor() as cur:
33         cur.execute(query)
34         rows = cur.fetchall()
35         for row in rows:
36             infos += str(row)

```

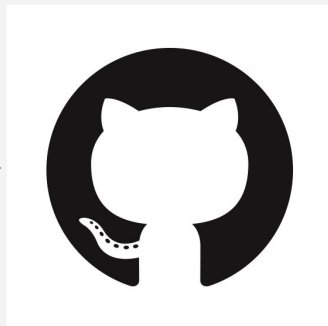
1:1 Python Spaces: 4

Testing

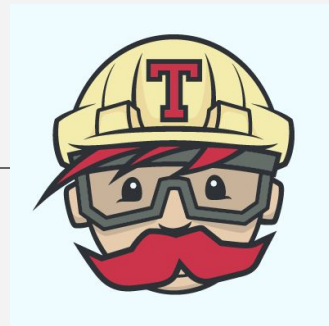
```
import pytest

@pytest.mark.parametrize('test_input, expected', [
    (3, 'Fizz'),
    (6, 'Fizz'),
    (5, 'Buzz'),
    (10, 'Buzz'),
    (15, 'FizzBuzz'),
    (30, 'FizzBuzz'),
    (2, 2),
    (7, 7),
])
def test_fizzbuzz(fizzbuzz, test_input, expected):
    assert fizzbuzz.say(test_input) == expected
```

**Python Testing
Framework**



**Push changes to
master repository
in GitHub**



**Automatically
trigger build jobs
in Travis CI**

Cost Proposal

Travis CI

Free for first 100 trial builds

Use throughout development phase (6 months)

Expected to exceed limit

Total: \$315

AWS Services

Cognito, Lambda, RDS

All within free tier

Do not expect to exceed limit

Total: Free

WattTime API

Stretch Goal

Real time emissions and grid data

Access costs money, available upon request

Total: Unavailable

Reflection

What Went Well?

Team Participation and Flexibility

It runs on all our systems!

Despite being assigned specific tasks,
the team took time to understand the
full stack

What Can Be Improved?

Steep learning curve with React Native
and AWS Lambda

Set-Up Struggle

The plan: utilize winter break to get a
head start on React Native



NEOCHARGE

The *new* way to charge