



**CS7080 – Cloud Computing and the Internet of Things – 2018**

**System Development Report (2018)**

**Project: Smart Parcel Care**

**GitHub Repository URL: <https://github.com/Khalid145/SmartParcelCare>**

**Name: Khalid Ahmed Mohamed**

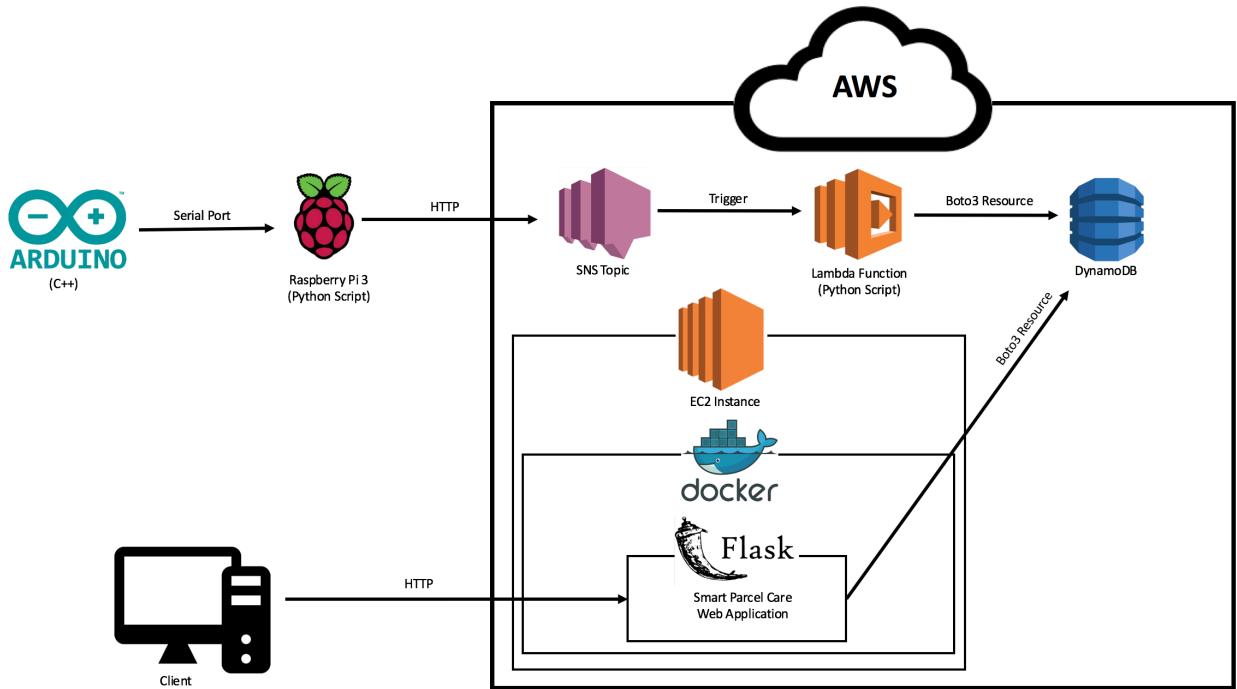
**ID: 13036087**

## **Table of Contents**

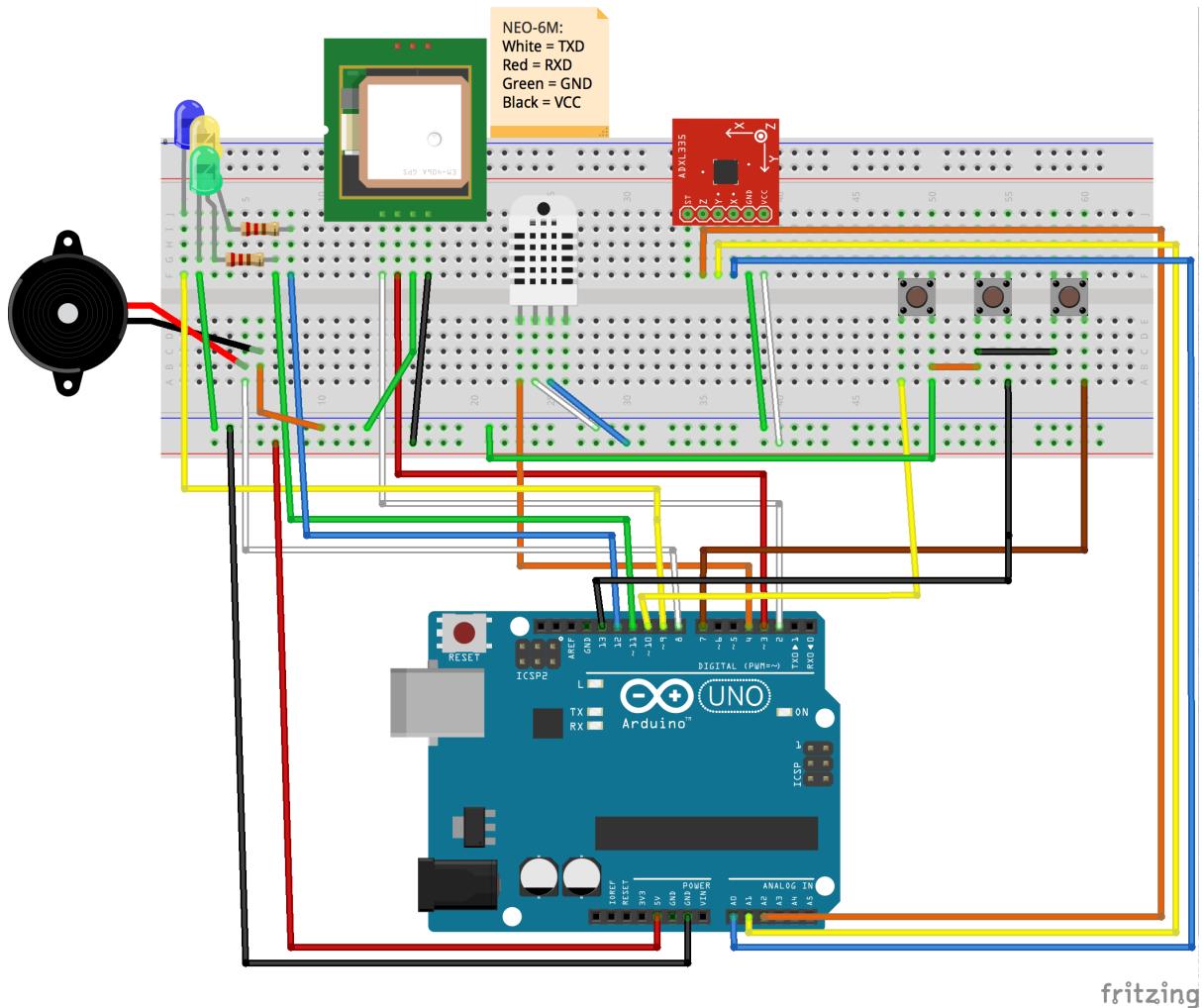
<b>1.</b>	<b>Application Architecture .....</b>	<b>3</b>
<b>1.1</b>	<b>Software.....</b>	<b>3</b>
<b>1.2</b>	<b>Hardware.....</b>	<b>4</b>
<b>2.</b>	<b>LSEP Issues .....</b>	<b>5</b>
<b>2.1</b>	<b>Legal Issue .....</b>	<b>5</b>
<b>2.2</b>	<b>Social Issue.....</b>	<b>5</b>
<b>2.3</b>	<b>Ethical Issue.....</b>	<b>5</b>
<b>2.4</b>	<b>Professional Issue.....</b>	<b>5</b>
<b>3.</b>	<b>Deployment.....</b>	<b>6</b>
<b>3.1</b>	<b>Create and Start EC2 Instance.....</b>	<b>6</b>
<b>3.2</b>	<b>SSH into the EC2 Instance.....</b>	<b>6</b>
<b>3.3</b>	<b>Add AWS Credentials as Environment Variables.....</b>	<b>7</b>
<b>3.4</b>	<b>Execute the Run Command for the Dockerfile in Docker .....</b>	<b>7</b>
<b>3.5</b>	<b>Check if Docker Container is Running.....</b>	<b>7</b>
<b>3.6</b>	<b>Check if Web Application is Visible in the Browser .....</b>	<b>8</b>
<b>4.</b>	<b>Walkthrough .....</b>	<b>9</b>
<b>4.1</b>	<b>Home Page .....</b>	<b>9</b>
<b>4.2</b>	<b>Parcel Information Page.....</b>	<b>10</b>
<b>5.</b>	<b>Project Evaluation .....</b>	<b>11</b>

# 1. Application Architecture

## 1.1 Software



## 1.2 Hardware



## 2. LSEP Issues

### 2.1 Legal Issue

Customers' parcels can be tracked in real-time due to an IoT device being attached with it throughout the whole delivery process. This IoT device will be able to track the exact location of the parcel. The biggest legal issue for this project is data protection. The challenge is how long is the data kept on the system once delivered, when to stop transmitting data so that irrelevant data is not collected and whether the data is needed for analytical processes.

### 2.2 Social Issue

As all customer parcels will be tracked in real-time, there could be some customers who won't be happy with the thought of having their parcels' location openly available on the internet before they receive it or following a parcel to the recipient's home. Therefore, an opt-out option will be provided to the customer to confirm whether they would like an IoT device sent with their parcel.

### 2.3 Ethical Issue

There are many ethical issues that could appear from this project. One issue is to make sure that this device is available to everyone on many factors especially financially. As the IoT device would not come free with their parcel, there will be a fee, this fee should not discriminate against low earning customers where a high fee for the IoT might be financially difficult compared to a high earning customers. Another ethical issue is to respect the privacy of customers and honour their confidentiality.

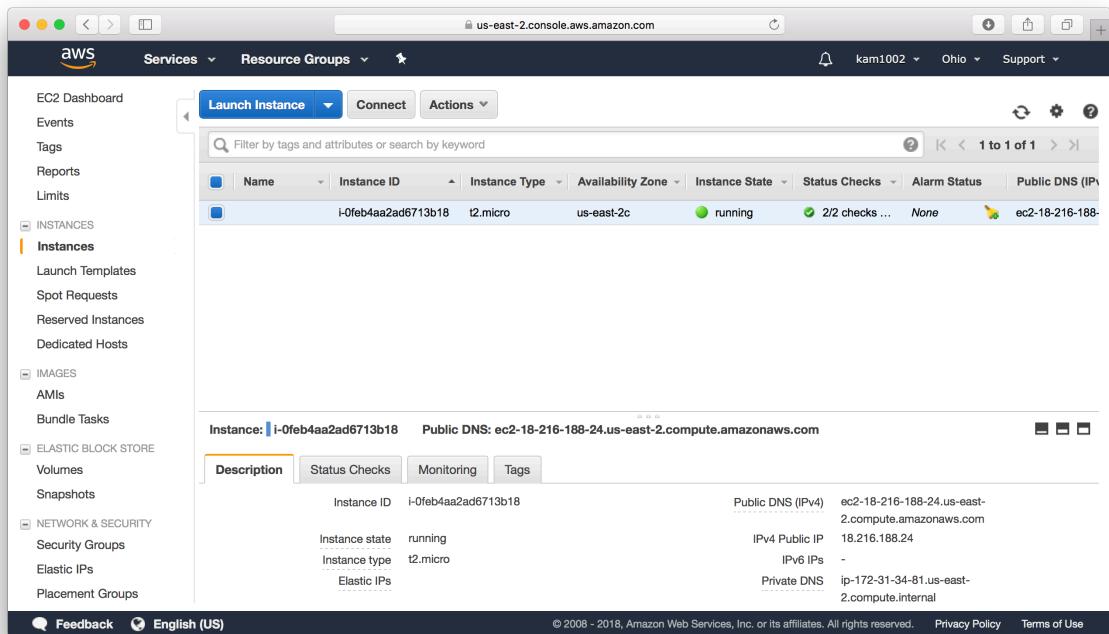
### 2.4 Professional Issue

As IoT is a relatively new field, conventional developers and architects will need to train and understand the new concept, in terms of developing the architecture of the application, the protocols required and the platform needed. Also, another professional issue is to ensure that professionals adhere to confidentiality and obedience to the updated laws updated data protection.

### 3. Deployment

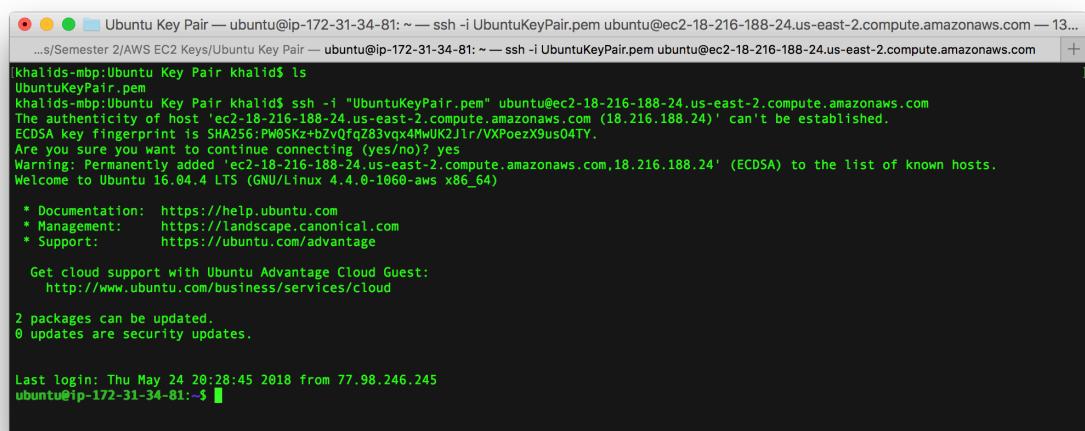
This application will be run and deploy on AWS in an EC2 instance. As there are many modern methods of deploying a web application, I have decided to use Docker. Docker allows you to run a web application with very little to no configuration, on any system provided that firstly Docker is install onto the host OS and that a Dockerfile is obtainable.

#### 3.1 Create and Start EC2 Instance



The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with options like EC2 Dashboard, Instances (selected), Launch Templates, Spot Requests, Reserved Instances, Dedicated Hosts, Images, AMIs, and Elastic Block Store. The main area shows a table of instances. One instance is listed: i-0feb4aa2ad6713b18, t2.micro, us-east-2c, running, 2/2 checks, None, ec2-18-216-188-24.us-east-2.compute.amazonaws.com. Below the table, there's a detailed view for the selected instance, showing fields like Instance ID, Public DNS (IPv4), Instance state, Instance type, and Private DNS.

#### 3.2 SSH into the EC2 Instance



```
[khalid@mbp:Ubuntu Key Pair khalid]$ ls
UbuntuKeyPair.pem
khalid@mbp:Ubuntu Key Pair khalid$ ssh -i "UbuntuKeyPair.pem" ubuntu@ec2-18-216-188-24.us-east-2.compute.amazonaws.com
The authenticity of host 'ec2-18-216-188-24.us-east-2.compute.amazonaws.com (18.216.188.24)' can't be established.
ECDSA key fingerprint is SHA256:PW0SKzbVQfz83vqx4MwUk2J1r/VXPoexX9us04TY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-18-216-188-24.us-east-2.compute.amazonaws.com,18.216.188.24' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-1060-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 Get cloud support with Ubuntu Advantage Cloud Guest:
 http://www.ubuntu.com/business/services/cloud

2 packages can be updated.
0 updates are security updates.

Last login: Thu May 24 20:28:45 2018 from 77.98.246.245
ubuntu@ip-172-31-34-81:~$
```

### 3.3 Add AWS Credentials as Environment Variables

```
Last login: Thu May 24 20:28:45 2018 from 77.98.246.245
[ubuntu@ip-172-31-34-81:~]$ docker ps
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get http://<http://>%2Fvar%2Frun%2Fdocker.sock: dial unix /var/run/docker.sock: connect: permission denied
[ubuntu@ip-172-31-34-81:~]$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
REPOSITORY          TAG                 IMAGE ID          CREATED            STATUS              SIZE
dockeredwebauciton_glassfish   latest              465c84f2d9af   4 days ago        Up 4 days         774MB
<none>              <none>             4517757a1b03   4 days ago        Up 4 days         774MB
mariadb              latest              447a28508139   5 days ago        Up 5 days         481MB
glassfish             latest              7004a1452d00   20 months ago    Up 20 months      773MB
corbinu/docker-phpmyadmin   latest              5c663962b799   2 years ago       Up 2 years        473MB
[ubuntu@ip-172-31-34-81:~]$ export AWS_ACCESS_KEY_ID=
[ubuntu@ip-172-31-34-81:~]$ export AWS_SECRET_ACCESS_KEY=
[ubuntu@ip-172-31-34-81:~]$ export AWS_DEFAULT_REGION=us-west-2
[ubuntu@ip-172-31-34-81:~]$
```

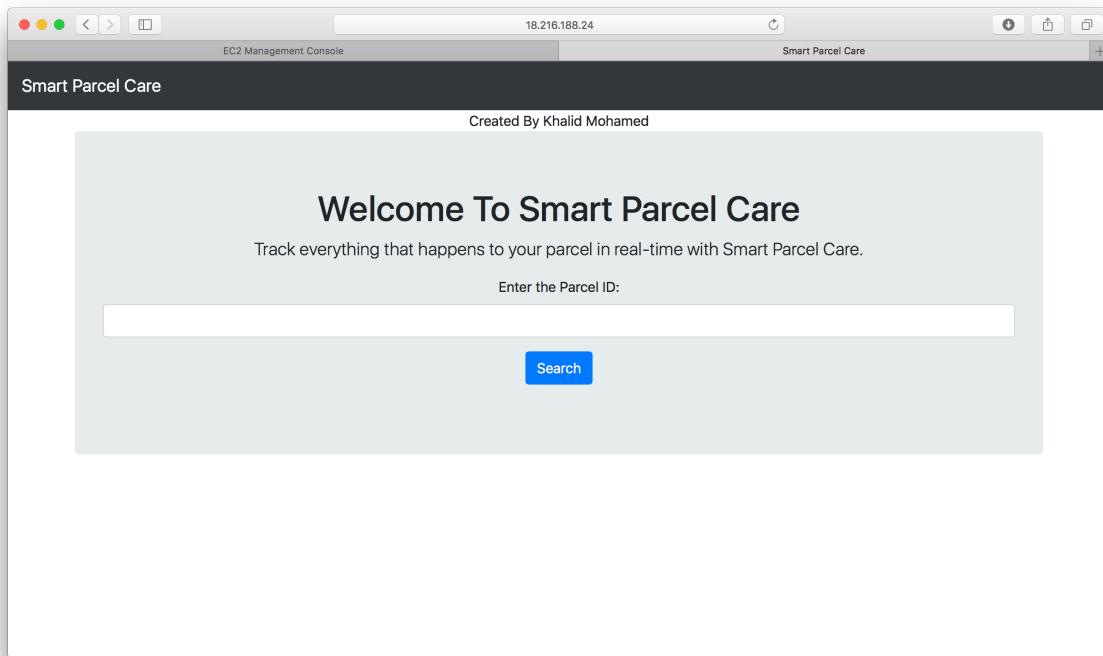
### 3.4 Execute the Run Command for the Dockerfile in Docker

```
[ubuntu@ip-172-31-34-81:~]$ export AWS_ACCESS_KEY_ID=AKIAIZYGKEGVYHCC375Q
[ubuntu@ip-172-31-34-81:~]$ export AWS_SECRET_ACCESS_KEY=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
[ubuntu@ip-172-31-34-81:~]$ export AWS_DEFAULT_REGION=us-west-2
[ubuntu@ip-172-31-34-81:~]$ sudo docker run -d -e APP_CONFIG=application.config.example -e AWS_ACCESS_KEY_ID=$AWS_ACCESS_KEY_ID -e AWS_SECRET_ACCESS_KEY=$AWS_SECRET_ACCESS_KEY -e AWS_DEFAULT_REGION=$AWS_DEFAULT_REGION -p 80:5000 khalid145/smartparcelcare:v4
Unable to find image 'khalid145/smartparcelcare:v4' locally
v4: Pulling from khalid145/smartparcelcare
c2c80a08aa8c: Pull complete
Gace04d7a42: Pull complete
f03114bcfb25: Pull complete
99df43987b12: Pull complete
9c646cd4d155: Pull complete
130c491b1715: Pull complete
993d38e59e92: Pull complete
0cd25af899e: Pull complete
809b523d0c2f: Pull complete
a060081d6215: Pull complete
0972c65e1bdd: Pull complete
9flaafr879c1c: Pull complete
78a20037f309: Pull complete
6664af71502d: Pull complete
a276d61713c4: Pull complete
cc205b89bb49: Pull complete
c1080607e614: Pull complete
Digest: sha256:400663a55eeef0f9b37ba7c1e082b396b0fcc141741b26d18df5ed22b09aff2a8
Status: Downloaded newer image for khalid145/smartparcelcare:v4
0a57ac39b13d98ac4ef3974e51f9e2184b2fa4752a21fac76dc86525b8c22ec
[ubuntu@ip-172-31-34-81:~]$
```

### 3.5 Check if Docker Container is Running

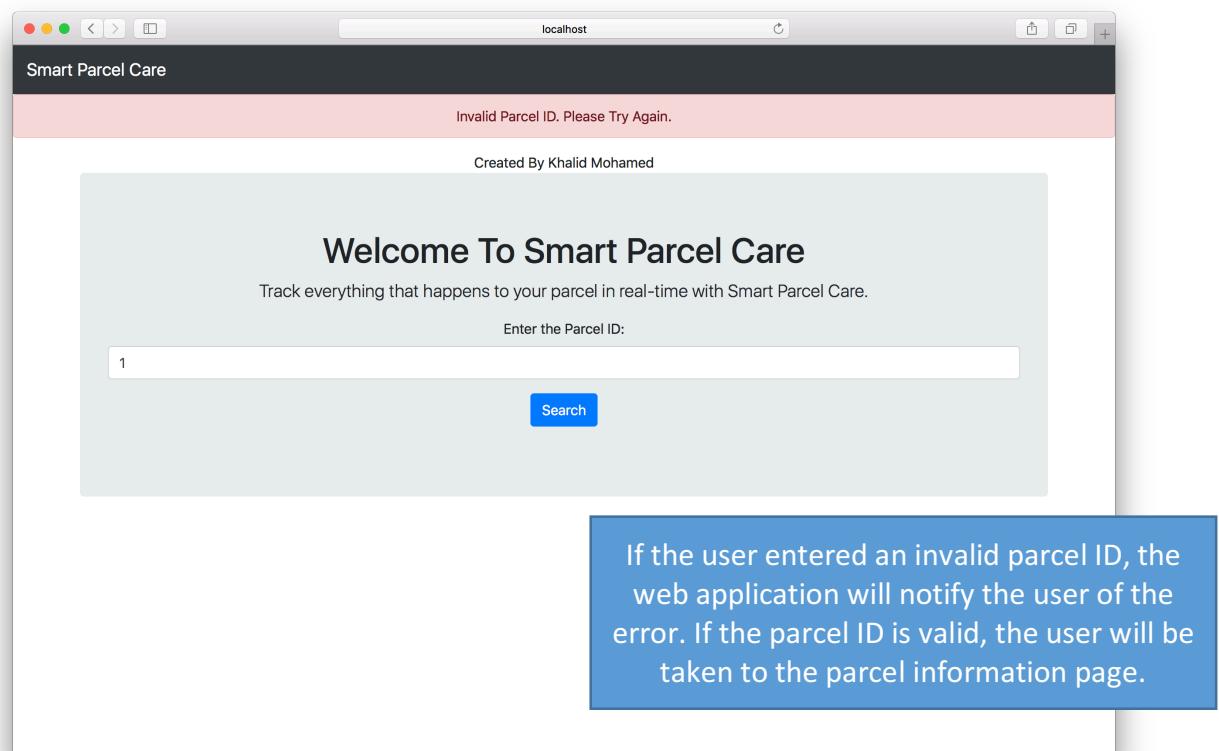
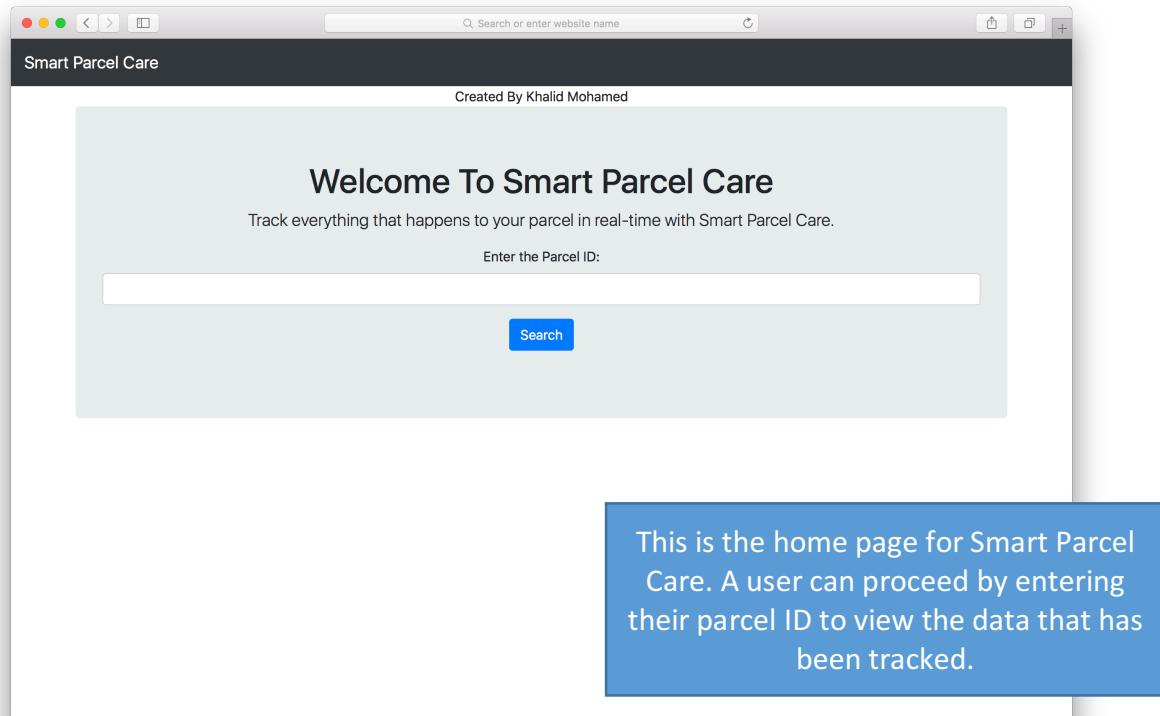
```
[ubuntu@ip-172-31-34-81:~]$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS               NAMES
0a57ac39b13d         khalid145/smartparcelcare:v4   "/bin/sh -c 'python .."   About a minute ago   Up About a minute   0.0.0.0:80->5000/tcp   agitated_bohr
[ubuntu@ip-172-31-34-81:~]$
```

### 3.6 Check if Web Application is Visible in the Browser



## 4. Walkthrough

### 4.1 Home Page



## 4.2 Parcel Information Page

This is the parcel information page. This page will contain information about the location of the parcels, temperature and humidity and whether the parcel has experienced any force to it.

Device ID	Date   Time	Data Point Number	Temperature	Humidity	Latitude	Longitude	Impact Measurement	Orientation Measurement
device3	2018-04-13   13:35:04	1	24°C	43%	Unknown	Unknown	-	-
device3	2018-04-13   13:35:09	2	24°C	44%	51.5580173	-0.1047221	-	-

This is the table for the parcel information page. If the parcel experiences any force or being oriented incorrectly, it will be marked as red in the table.

Device ID	Date   Time	Data Point Number	Temperature	Humidity	Latitude	Longitude	Impact Measurement	Orientation Measurement
device3	2018-04-13   13:35:04	1	24°C	43%	Unknown	Unknown	-	-
device3	2018-04-13   13:35:09	2	24°C	44%	51.5580173	-0.1047221	-	-
device3	2018-04-13   13:35:14	3	23°C	43%	Unknown	Unknown	-	-
device3	2018-04-13   13:35:18	4	23°C	43%	Unknown	Unknown	-	432
device3	2018-04-13   13:35:24	5	23°C	43%	Unknown	Unknown	-	-
device3	2018-04-13   13:35:29	6	23°C	42%	Unknown	Unknown	352	-
device3	2018-04-13   13:35:34	7	21°C	44%	Unknown	Unknown	-	-
device3	2018-04-13   13:35:39	8	23°C	43%	Unknown	Unknown	-	-

## 5. Project Evaluation

Overall, I believe that the project was a success because I was able to deliver all of the functionality that I specified that I will at the start of the project and users of the system have been impressed by its quality. However, even though all of the functional requirements were met, the process of achieving these functionalities differed from what was first stated at the start of the project due to technical issues.

One main technical issue that I had to face and unsuccessfully could not implement was the transportation of data from the raspberry pi to AWS via text messages. The component that I brought to achieve this was the GSM SIM900A. This component allowed me to insert a sim card into the sim card slot and send SMS programmatically. However, I faced issues right from the start. Once I wired the GSM SIM900A to the breadboard, it was having trouble recognising the sim card based on the led blinking pattern on the component. After some research, I found out that this component was developed for the Asian market and therefore needed the firmware flashed. I was able to flash the component and after this, the component recognised the sim card. This was identified by the led blinking pattern on the component.

However, I was still facing issues with the GSM SIM900A. The component will struggle to obtain a carrier and would restart completely every few seconds. There was an assumption that the component wasn't receiving enough power and current, however, once this was requirement was met, the problem was still persisting. After careful analysis with specialised software, it showed that the component could be damaged. Due to the time schedule of the project, it was not worth persisting to solve the issue or wait for another component to be delivered. Therefore, I decided to use mobile data from mobile phone as a hotspot to connect the Smart Parcel Care into the Internet. Even though, this solution was not what was asked for, it has helped with the proof of concept.

If I was able to further develop this project, I would first and foremost make the Smart Parcel Care more user-friendly, in terms of managing the devices without the need for prior knowledge of programming. Currently, to change the device id or temperature type i.e. Celsius/Fahrenheit, the user is required to understand C++ and have access to the backend code for each device. However, if I had more time, I would develop a web application interface so that the user can configure the device for the browser.