Group C0:
Vayum Arora
Ajay Benno
Peter Germa

Occupancy Monitoring System

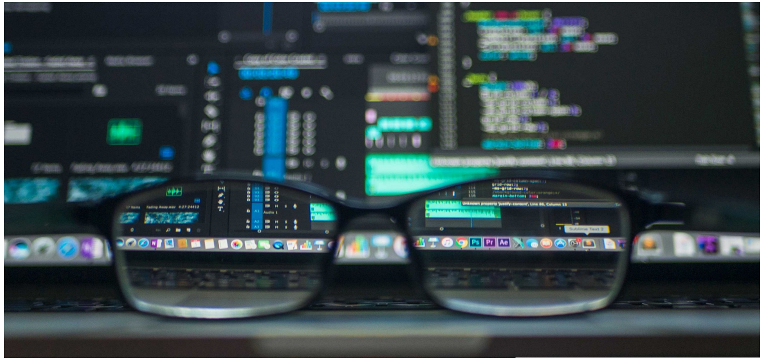
Use Case

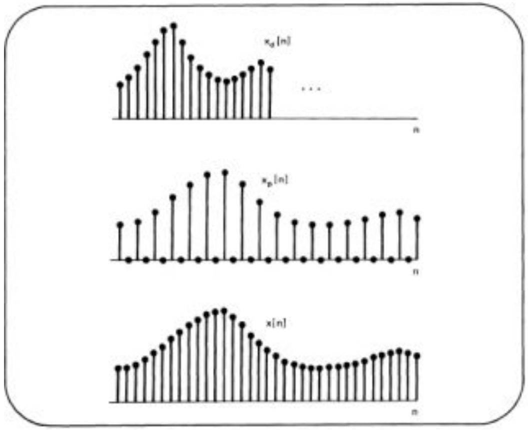
- Track wait times and current table occupancy for fast casual restaurants.
- Currently not that much information on how busy a location is.
- Lets users make a more educated decision on where to eat.

Requirements

1. Capture and store identifying features of a person entering a restaurant
2. Match stored data with with subsequent image taken at the front of the line
3. Calculate average time stood in line, updated in real-time every one second
4. Detect people sitting at a table with 90% accuracy
5. Create a sticky web-application connecting to camera and sensor data, displaying aggregated analytics and predictions with projected wait times

Areas of Study





Signals and sensors

Computer Vision

Software Systems

Solution Approach

Camera

- Integrate Raspberry Pi with cameras, exploring connection to server and pi
- One camera to detect who comes in the restaurant, other camera at the front of the line to match the image to the same person
- Timestamps of images to calculate average wait time
- Computer vision algorithm to match faces, facial features

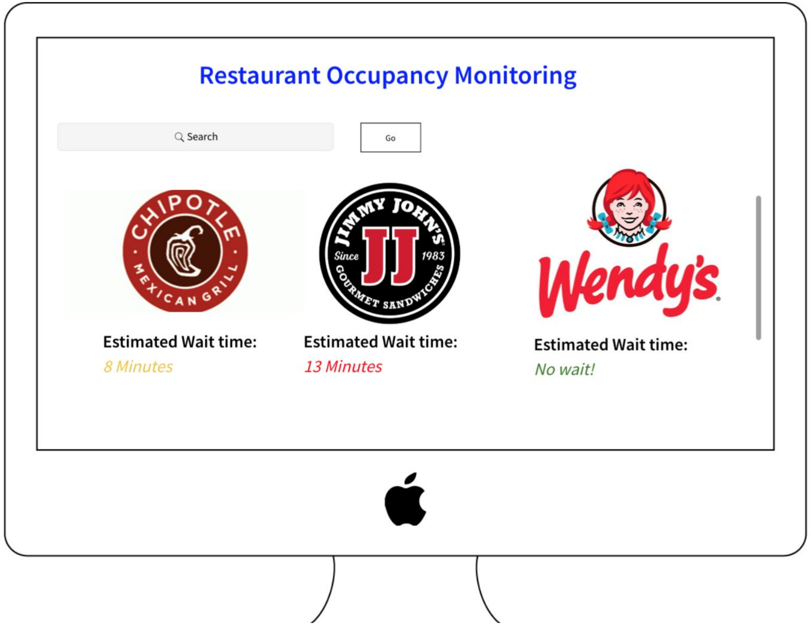
Sensors

- Figure out which sensors to use
- Detect human in two states (occupied/non occupied) with table sensors
- Integrate and send table sensor data to server
- Generate heat maps/analytics tool to represent occupancy of tables

Solution Approach- Continued

Web-Application

- Decide format of data for camera and sensor data
  - Pick frameworks most optimal for project
- Develop backend server
- Connect to data coming in
  - Write a test client to send data, see if it can receive proper formatted data
- Design and create pages



Restaurant Occupancy Monitoring

Estimated Wait time:
8 Minutes

Estimated Wait time:
13 Minutes

Estimated Wait time:
No wait!

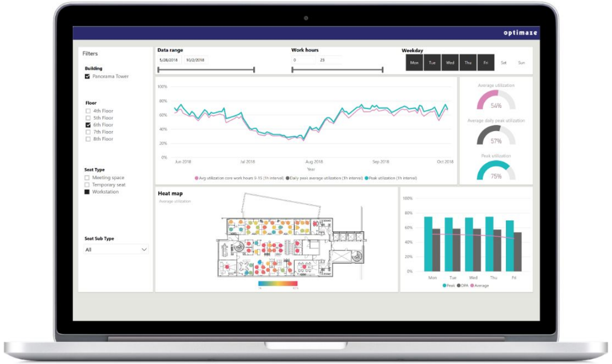
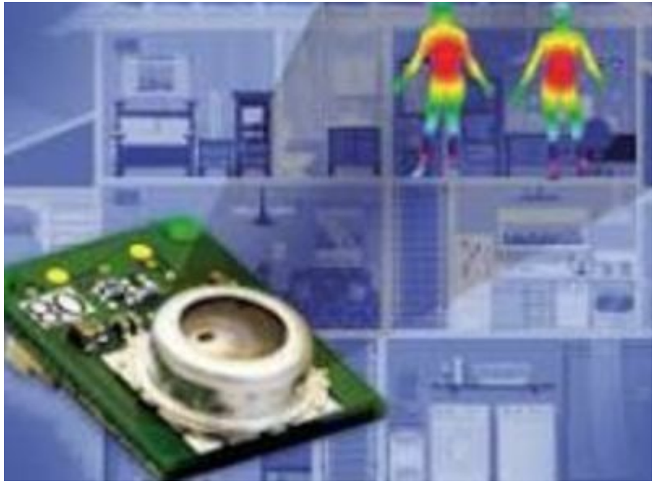


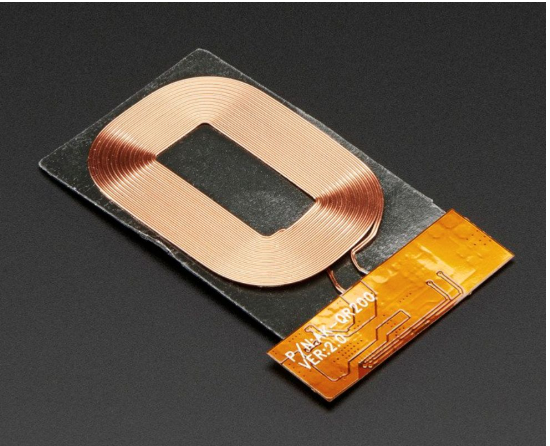
Table Sensing

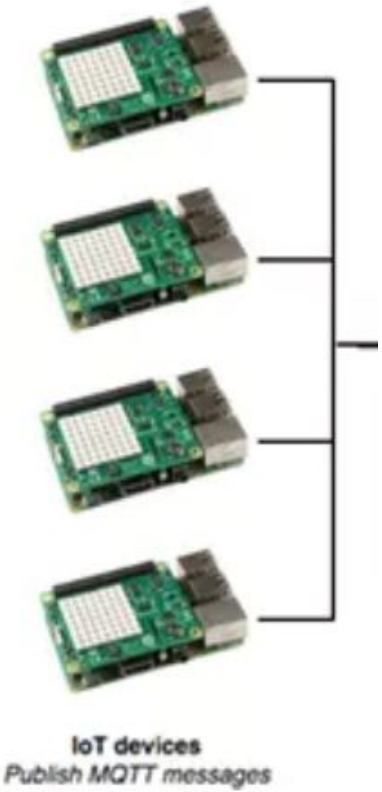
- Table sensing will be handled by nodes of raspberry pis connected to the network
- Identified SENSEHat shield for temperature, humidity, and gyroscope
- Explore using a passive IR sensor to detect movement from under the table
- Explore using a thermal sensor to easily detect stationary humans
- Explore using pressure sensitive sensors under chairs



Processing Sensor Data

- Every 30 seconds the raspberry pi will be polling its sensors to see if they meet our determined threshold and binerize if a person is there
- This data will be sent from each pi to our server using MQTT protocol
- In addition to sensing, we want to create a wireless charging station, providing utility and surefire way to know if someone is at a table
- Design and fabricate a case that houses sensors and wireless charging station





Tasks/Division of Labor

Ajay - Facial Recognition

- Detect Human from camera feed
- Extract Human Features from Camera
- Write matching algorithm
- Raspberry Pi algorithm to take photos and send to Server
- Write Server to receive photos.
- Wait time algorithm.
- Connect with frontend.

Vayum- Frontend/Backend Architecture

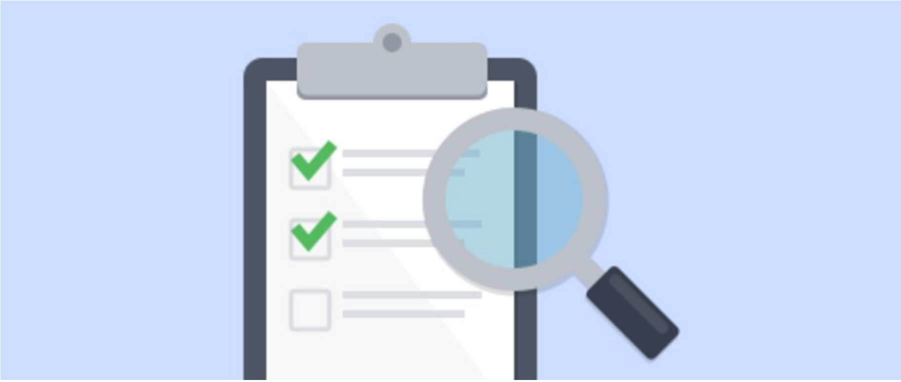
- Design requirements
  - Decide format of data for camera and table
  - Decide database
- Build Layout of Home page
- Build individual restaurant page
- Create fake data to test website
- User testing for layout
- Connect with table sensing and facial recognition

Tasks/Division of Labor-cont

Peter- Table Sensing

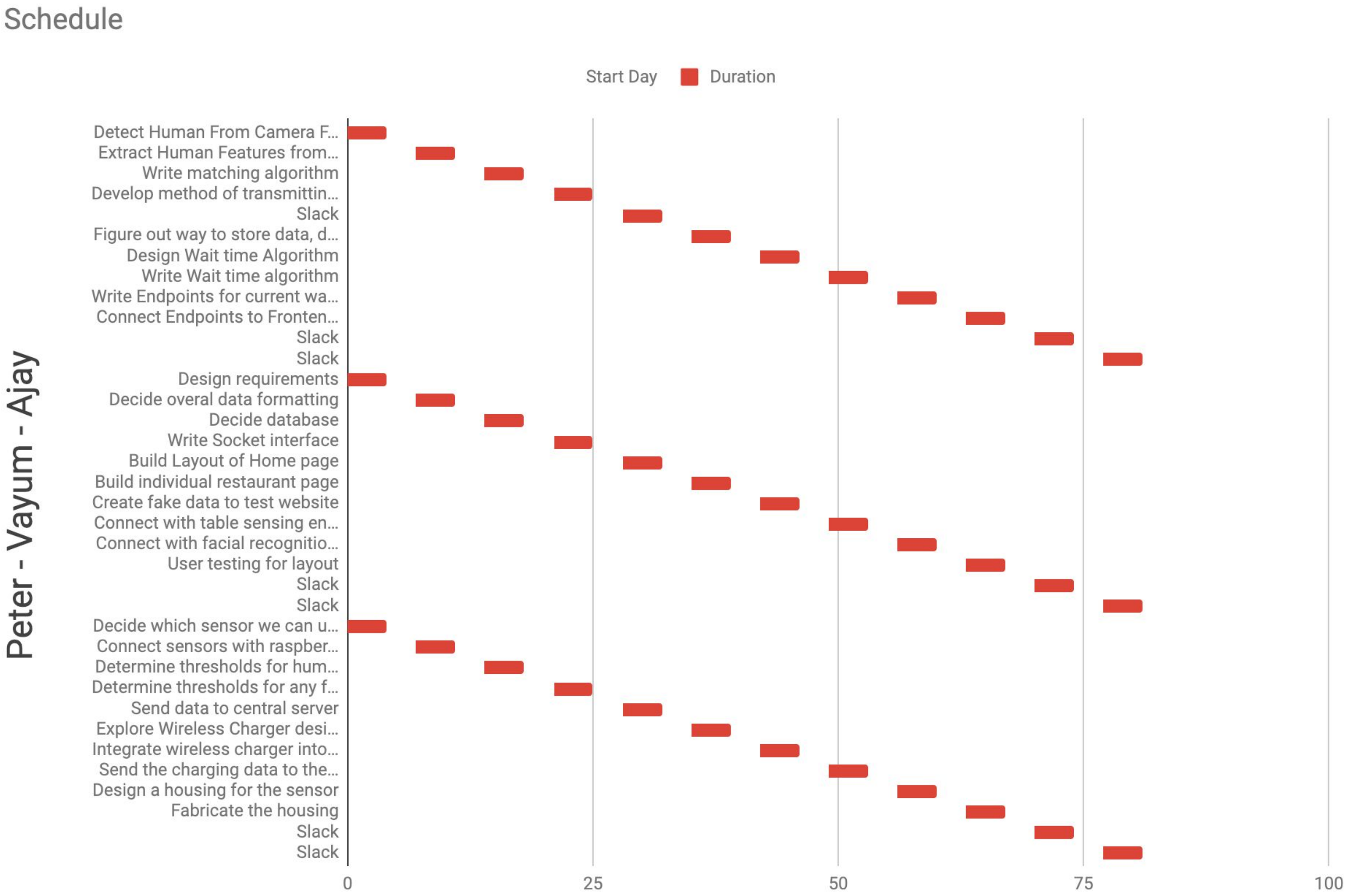
- Decide which sensor we can use
- Connect sensors with raspberry pi
- Determine thresholds for human sensing
- Determine thresholds for any false positives
- Send data to central server

Verification



1. 90% + accuracy detecting human activity at tables
2. 90% margin of detecting wait times for the average person
3. <10% failure rate of identifying a person entering and then ordering
4. Able to display data in such a way that it is helpful to our stakeholders
5. Use past data to predict future behavior (wait times, table availability)

Schedule


Schedule
Start Day
🟥 Duration
Peter - Vayum - Ajay
Detect Human From Camera F...
Extract Human Features from...
Write matching algorithm
Develop method of transmittin...
Slack
Figure out way to store data, d...
Design Wait time Algorithm
Write Wait time algorithm
Write Endpoints for current wa...
Connect Endpoints to Fronten...
Slack
Slack
Design requirements
Decide overal data formatting
Decide database
Write Socket interface
Build Layout of Home page
Build individual restaurant page
Create fake data to test website
Connect with table sensing en...
Connect with facial recognitio...
User testing for layout
Slack
Slack
Decide which sensor we can u...
Connect sensors with raspber...
Determine thresholds for hum...
Determine thresholds for any f...
Send data to central server
Explore Wireless Charger desi...
Integrate wireless charger into...
Send the charging data to the...
Design a housing for the sensor
Fabricate the housing
Slack
Slack
0
25
50
75
100