COFFE SHOP WAIT TIME MONITORING USING SMARTPHONE

1. Team “Wait n Watch”

- Manasvi Karanam
- Rohit Zawar

2. Project Idea

The long and unpredictable queues at places like coffee shops, stores, etc. cause a large amount of inconvenience in our day to day lives. Many scientific studies indicate that customers have shown a lot of dissatisfaction over such waiting times. So, this project aims to develop a system which can predict and report current/future waiting time of the queue using smartphone. This will help the users make more informed choices. In our project we consider the coffee shop present at Indiana University campus.

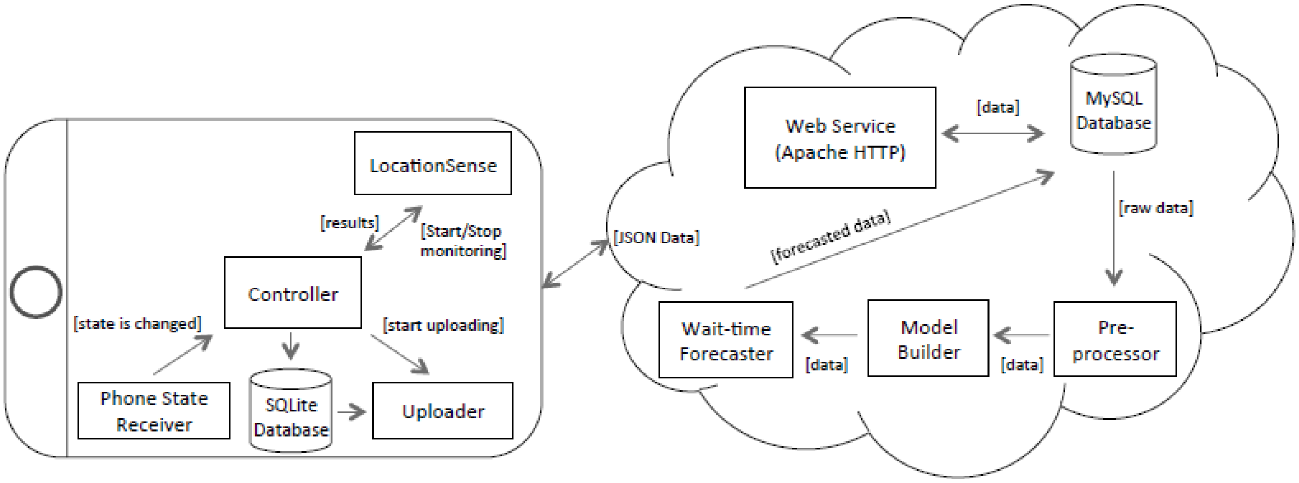
3. Approach

Our application consists two components:

- **Client-side component on the smartphone.**
- **Server-side component**.

The client-side component provides automatic wait-time detection and uploads the data onto the server. The server-side on the other hand, uses the collected data to provide accurate wait-time estimation.

The overall architecture of our system:



3.1.1 Wait-time detection

In order to detect the wait-time automatically, the client side makes use of two main sub-components along with the controller – Location sensor and Data uploader.

- Location Sensor

  We use GPS in user's smartphone to accurately predict if the user has entered the Coffee shop. We also make use of Phone-state-receiver subsystem of android i.e. *Proximity Alert Receiver s*ervice to monitor the wait-time. Proximity Alert Receiver periodically checks the location of device and fires alerts for entering and exiting for a specified geo-fence region. So once it triggers entering, we start monitoring. Once triggered exit we stop monitoring and store the time in database.

- Data Uploader

  After completing wait-time detection, the system tries to upload the collected data to the server. The application immediately tries to upload the data. However, due to certain connection problems, if the application isn't able to upload the data immediately, then it is the responsibility of the data uploader sub-component to transmit the pending data at later point of time.

3.1.2 Wait-time estimation

Once the client-side component uploads the wait-time detection data, the server-side uses this data to build estimation model to predict the waiting time. This task is accomplished mainly by four sub-components at the server side – Web service, Pre-processor, Model Builder and Wait-time forecaster.

- Model Builder

  To solve the problem of wait-time estimation we choose Nearest Neighbor Estimation(NNE). We consider week of the year, day of the week and interval of the day as three parameters to construct the estimation model. We later plan to use regression-based optimization technique to improve NNE. We use Mean Square Error (MSE) to measure the accuracy of the system.

- Wait-time Forecaster

  The current query is fed to the wait-time forecaster to predict the wait-time.

4. Key Challenges

- Learning Android

  Being inexperienced with android development, One of the key challenges is to learn android from scratch. Way to tackle the problem is to learn android from various online resources such as Lynda tutorials, Vogella tutorials, etc. and in short span of time.

- Hosting web server

  For implementing server side for fetching the results, the entire web server need to be deployed on the specific domain. The key challenge is to implement and host the web service with database on purchased domain. Way to tackle the problem is to use PHP as server-side technology for implementation and looking into some free domain to host the service.

- Building model

  For Implementation of Model, Key challenge is to estimate the line wait time for any arriving query using collected data. Our approach to solve the problem is a nearest neighbor estimation (NNE) based on constrained nearest-neighbor search in a multi-dimensional space. Also we would like to propose second approach where we improve NNE by building time-series model on collected data using the previous history of wait-times. Basically, both the approaches will be evaluated using the Mean absolute error metric and the improvements will be implemented accordingly.

- Collecting data

  Collecting data is one of the biggest challenge. Way to tackle is to distribute the application to the users who frequently visits one particular coffee shops or team would manually go and collect data on frequent basis to gather more and more data at different times.

5. TimeLine

Date	Manasvi	Rohit
Present to February end	Learning Android and Location sense module	Learning android, SQLlite and uploader module
March to Mid-March	Web server and hosting	Web server and hosting
Mid-March to March end	Data filtering and implementing model	Data collection and implementing model.
April to April end	Implementing model and testing	Evaluations and optimizing