ParkWizard: Street Parking Android App

Siddharth Shah, Kunal Baweja, Dhruv Shekhawat, Anand Naik

Abstract

A good day begins with a good parking. Finding a parking spot rather than ending up with a ticket is one of the daily struggles faced by millions of drivers worldwide, however the problem is not lack of parking, rather most of the times drivers are simply not aware of parking spots.
Various apps have been developed in the past but with an intention to provide paid garage parking. In this project, we introduce *ParkWizard*, a user incentive based android app that crowdsources parking data to provide on-demand search facility for hassle free parking.

Introduction

ParkWizard is an Android app that helps users find available street parking locations. It works on a score based system where users earn points by reporting parking locations and updating availability status. The users further spend their earned score back into the app to search for available parking spots when needed. This score-based system works well towards maintaining a smooth information flow between users(drivers) who want to find available parking spots and those(informers) who have information about them. At any point, a user can update as well as search for parking locations on the app.

User Incentive & Protocol Design

Parkwizard does not contain any parking location data of its own. It aims to incentivize users to provide the app with latest and most correct information about parking locations in their knowledge in return of earning reward points on the app. The scoring scheme is as follows:

- A user is rewarded 10 points for reporting a new parking location. However, a user is not allowed to report a parking within 50m of an already existing parking to avoid duplicates.
- A user is rewarded 2 points for every update they provide about an existing parking spot. To limit misuse of the update feature, we limit a user to make a maximum of 5 updates per day. These updates can be made for parking spots within 100m of the user device location.
- For updating false information such as suspiciously high number of parking spots or fake information, a user is penalized 2 points.

Algorithm

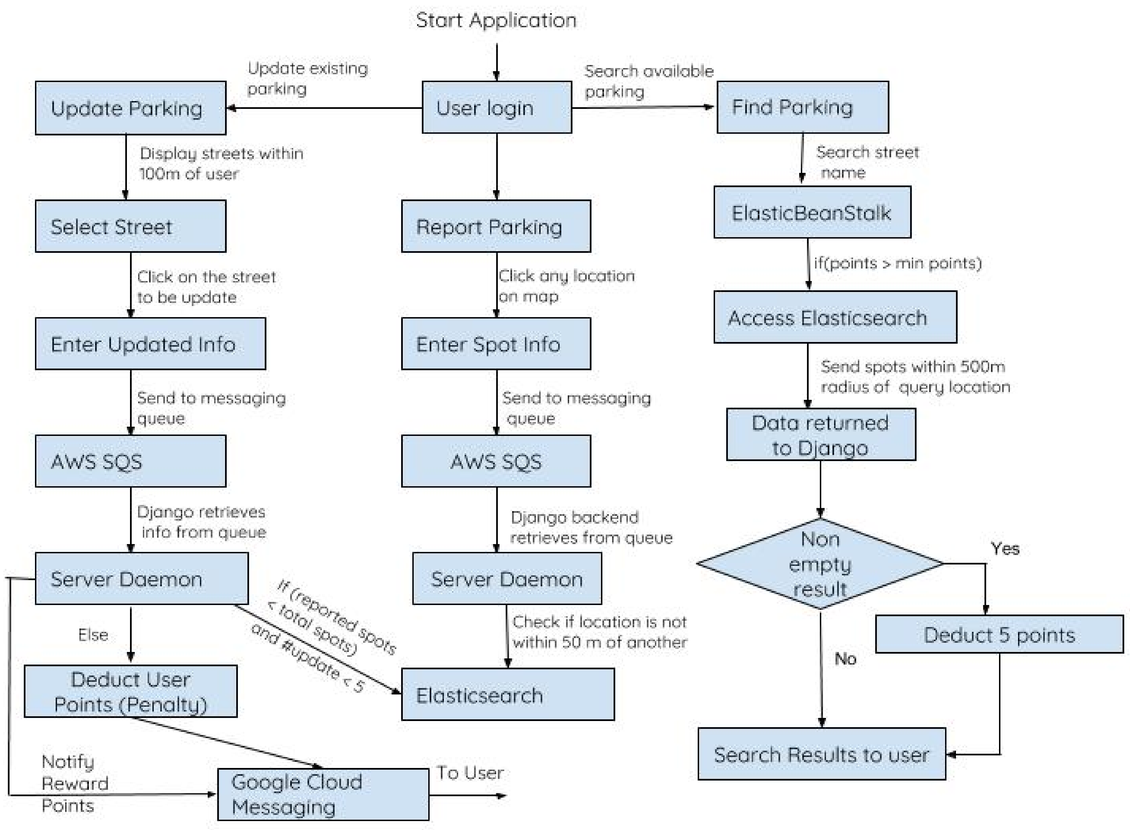

Figure 1: App State Transitions

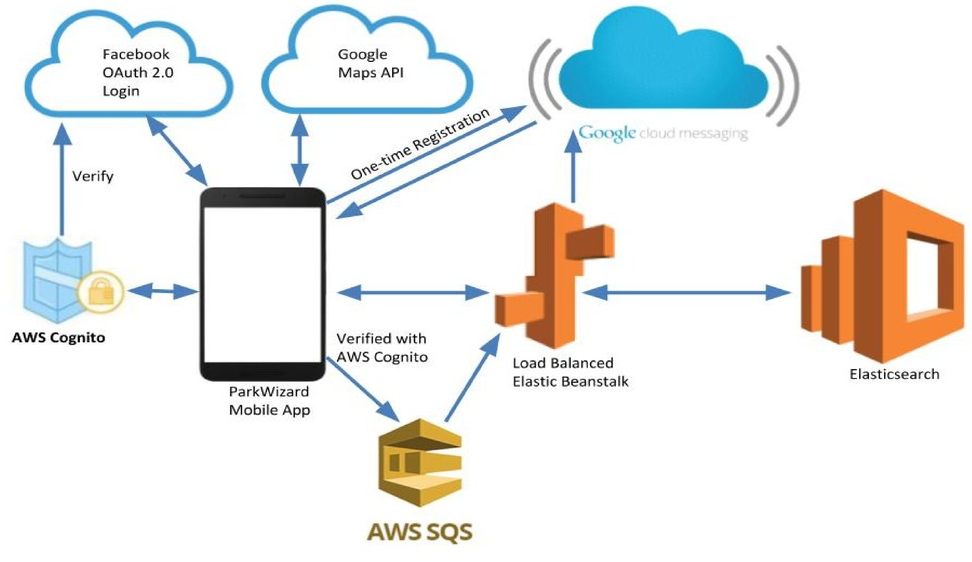

Figure 2: Architecture

- A user has to spend 5 points to search for parking locations they wish to find near their destination. By default the parking locations are shown within a 500m radius of destination.
- To encourage users to keep the parking location data most up to date, the app rewards them back 2 points if they choose to use a parking spot from the search results and navigates them to the location via Google Maps.

System Architecture & Working

Users sign up using Facebook OAuth 2.0 authentication. The app uses *AWS Cognito service* to authenticate the user for accessing the underlying AWS services associated with *ParkWizard*. The server side business logic is implemented using Django 1.10.4, deployed on *AWS ElasticBeanstalk* auto-scaling environment. This is backed by an auto-scaling deployment of *AWS ElasticSearch* where the user and parking location data is indexed for fast real time searching. *Parkwizard* app uses *Google Maps API* to display the geographical map as the primary user interface, where the user can see parking location search results, select parking locations to update available parking spots or report new parking locations from the menu option.
The user requests for searching parking locations and signing up require fast realtime processing and hence serviced directly by the servers deployed in *AWS ElasticBeanstalk* auto-scaling environment. On the other hand, updates to parking locations and new locations need to be processed in order with some delay tolerance, for which we use the *AWS SQS* service to queue up the requests. A continually running server daemon process, consumes the request messages from the SQS queue, updates the parking data on ElasticSearch and notifies users about score updates via push notifications sent through *Google Cloud Messaging*.

Conclusion

ParkWizard's mission is to solve the tedious problem of daily parking as the cities grow more and more complex. The user incentive based approach for crowd sourcing latest information about parking location also ensures cheap, reliable and efficient data collection of parking locations which in turn results in most relevant search results and user activity.