**AirPnP Proposal**

***Team Name:***Airpnp

***Team Members:* Yuyang He, Aoyi Li, Zhuoyan Zhu, Luoyin Feng**

**1.Motivation and Objective**

Airpnp is an app focused around providing an easy and convenient way for drivers to park even during high traffic time periods. At certain times, such as around dining hours or when a large event is taking place, traditional city parking often meets capacity, causing drivers to park far away from their destinations. In the case of a major sporting event, fans can spend many tens of minutes walking from their car to the stadium, potentially causing them to miss the start of the game.

Current applications such as SpotHero and ParkingPanda try to solve this by allowing users reserve parking spots at their lots ahead of time, however, these lots also have limited capacity, and thus are susceptible to falling short of demand. As a result, many other drivers are left to find other means of parking. Additionally these lots are located in specific locations, and in the case that the location of the lot is not nearby the event they are attending, the convenience of the app is lost. Finally, with all the lots provided by these apps, each one will have to be maintained resulting in cost increases.

To solve the problems of capacity, limited locations, and maintenance costs, Airpnp takes a more dynamic approach to parking that involves consumer participation in the parking process. Consumers of the app will be able to easily rent out space on their property for other drivers. This opens up two types of app consumers who serve to benefit: the general driving public and the lot providers.

Three themes that this app will encompass include mobility, service providing, and contextual awareness. Because this app is able to be used on one’s mobile phone, the user will be able to find parking no matter where there are. The service being provided is a new marketplace for parking that users will be able to interact with. Finally, the app will be location aware, thus giving the proper context for the app, as parking is only relevant to where the user is going.  
**Use Cases**

To visualize how our application is a solution, imagine that you are venturing to a sporting event in a different city. As you arrive, you noticed that all of the official parking decks and lots for the event are filled to capacity. With the mobile application, all you need to do is look for a nearby parking spot provided by another individual that lives nearby, or owns a personal lot. Upon finding an ideal spot, you would contact the owner, reserve your spot, and complete the payment of the parking spot all within the application.

Another viewpoint of using the application is through a homeowner near an ocean front. As a homeowner, you may have a few spots available to rent to fellow beach goers, but no way to market these spots easily online. Airpnp’s web front will allow them to list their spots online for any travelers to use, and allow spot providers to earn income from their land from renters.  
**Technology**

The app is being developed on the Android platform. A webapp is also in the works. The backend being used is Amazon Web service and the Nexmo API to send text messages. The front end uses Google maps as well geocoding to provide locational data. We currently plan on using Paypal to accept payments.

1. **Related Work**

Current applications such as SpotHero and ParkingPanda try to solve the parking problem by allowing users to reserve parking spots at certain lots ahead of time, however, these lots have a limited capacity, and thus are susceptible to falling short of demand. As a result, many other drivers are left to find other means of parking. Additionally these lots are located in specific locations, and in the case that the lot is not near the event they are attending, the convenience of the app is lost. Finally, with all the lots provided by these apps, each one will have to be maintained by manually adding these specific lots, resulting in cost increases.

1. **Proposed Work**

Our app offers the following value to customer:

Convenience: Fast and easy to use in order to find a place to park your car. You don't have to worry about unavailable public parking spots even in the peak season, and all you need is few clicks from the app.

One-of-a-kind stay experience: You have the right to choose the best place to park your car and deserve the best possible parking experience.

Extra income for hosts: More money in their pockets, why not?

Good way to advertise nearby events: A platform to advertise both events and any parking for those events.

To solve the problems of capacity, limited locations, and maintenance costs, Airpnp takes a more dynamic approach to parking that involves consumer participation in the parking process. Consumers of the app will be able to easily rent out space on their property for other drivers. This opens up two types of app consumers who serve to benefit: the general driving public and the lot providers. Additionally, event organizers will be able to benefit by advertising their events on the app.



Fig.1 Overall Architecture of the AirPnP Application

The implementation of the overall architecture follows the flow as described in Figure 1. The backend is Amazon Web Service. Amazon Relational Database Serviceis in charge of storing user data such as general account information and parking lot location information and Amazon Elastic Compute Cloudis the server that hosts our web application. We have a website and the Android mobile application as front ends for our application. APIs that we utilize within the application are EventBrite for querying nearby events, Google Maps for location services, and Paypal for its payment services.

1. **Plan of Action**

**Schedule**:

|  |  |  |
| --- | --- | --- |
| Week No. | Dates | Scheduled work |
| 1 | Feb 13- Feb 19 | Study: Java, AWS API |
| 2 | Feb 20- Feb 26 | Implement backend application |
| 3 | Feb 27- Mar 6 |
| 4 | Mar 7 – Mar 13 |
| 5 | Mar 14 – Mar 20 | Implement Android application |
| 6 | Mar 21 – Mar 27 |
| 7 | Mar 28 – Apr 3 | Implement Web application |
| 8 | Apr 4 – Apr 10 |
| 9 | Apr 11 – Apr 17 | Integrate and test |

1. **Evaluation and Testing Method**

The first step is to evaluate the whole process of the application. Make sure all of the function can be implement properly.

For security reason, try to do some basic fuzzing work on the web application and do some reversing work to the android application. Doing the penetration test with both black box testing and white box testing.

1. **Bibliography**

* Java
* Android
* nexmo
* Noodlio
* google-maps
* Angular
* Javascript
* Parking Panda - <https://www.parkingpanda.com/>
* SpotHero - https://spothero.com/
* Business Model Canvas - https://canvanizer.com/canvas/wkqNkAE9hXs0l