**PRTravel**

**Harry Hernández, Abdiel Vega and Geraldo López**

[**harry.hernandez@upr.edu**](mailto:harry.hernandez@upr.edu) **,** [**abdiel.vega2@upr.edu**](mailto:abdiel.vega2@upr.edu) **and** [**geraldo.lopez1@upr.edu**](mailto:geraldo.lopez1@upr.edu)

**Introduction**

Puerto Rico is going through the worst economic crisis in its history. It owes over 70 billion dollars and experts ensure that the administration of the governor will not be able to pay the debt completely. To aid this problem tourism and social networks can contribute by working together. According to Puerto Rico Tourism Company “the tourism industry is one of the economic engines with the most potential for growth in Puerto Rico” (4). On the other hand, we have been witnesses of the great impact that social networks have had in the world. Taking these facts into consideration PRTravel will be a mobile application that will seek to encourage tourists to visit Puerto Rico’s different attractions by sharing their experiences with other users. While the tourists are having a good time they will be also contributing to the island’s economy. This social network will serve as a tool for the tourists because it facilitates the access to information of the diverse attractions in the island. This application has been chosen to be mobile so that tourists could use it while actually visiting the attractions. The technologies that are going to be used in this project will be Ionic Framework and AngularJS for the client side, Java Play for the server side, and for the database system PostgreSQL. In addition, Eclipse and Ionic Lab will be used as the programming IDEs.

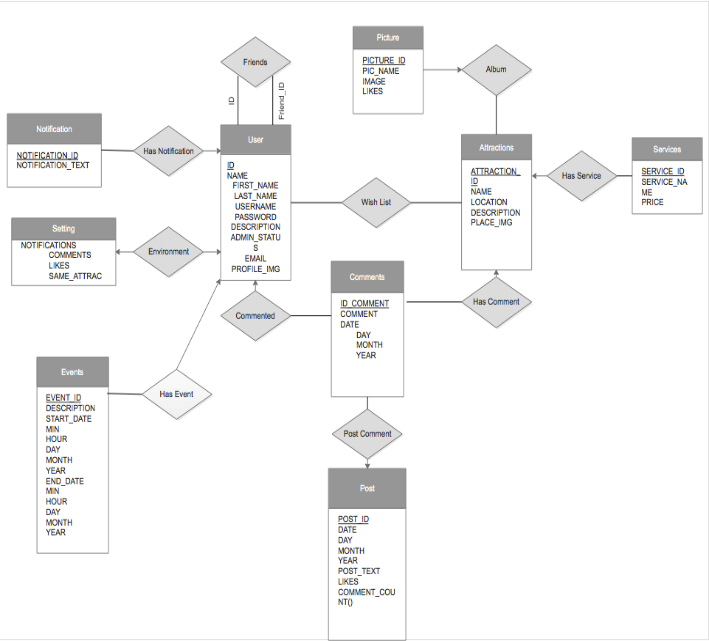
# **Client App Description**

PRTravel will be a social network for visiting different places or attractions around Puerto Rico and sharing it with other users. Users will sign up and create a profile. This application will have five main components: home page, profile page, attraction list, calendar, and notifications. The home page would include a news feed of your friends. The profile page would include the user’s personal information, a wish list of the places the user would like to go, a calendar of the places he is planning to visit, photo albums of the places he has visited, and the settings. The attraction list will be a list of different attractions around the island and the user will have the opportunity to select from this list the places he would like to visit and add it to his wish list. Here will be the services that the user could pay for in advance. For example, activities, tickets, and transportation. The calendar will include the attractions the user and his friends are planning to visit. The notifications will tell the user when somebody added him as a friend or when he has been accepted as a friend. Also, it will notify when a friend commented on his photos or planned trips. This application will be using the AngularJS and Ionic framework for the client side.

# **Server Side Description**

The server side application will be working with the logic, validation and storing of information that the application requires. In the attraction list component, the user will have the opportunity to choose from the attractions that are already predetermined. Also, in this component the users can express their opinion in relation to the attraction. This attraction list component will have a table. For the notifications component the server application will be in charge of notifying to the corresponding user the friend requests, comments or any other activity that occurs. This notification component will have its own table. At the home page the application will have a news feed which will be updated after the user refreshes the page. The information about the news feed will be stored in one table. Every user will have a profile and at that page everyone will have a wish list with the places that he wants to visit. This wish list will be a subset of the list in the attraction list component. After the user visits any attractions he could mark it and then he can upload pictures to an album that will be created automatically by the server application. The name of the album will be the same name of the attraction. Furthermore, the profile page will contain a calendar that will have the places that the user will visit per day. The profile page will have four tables. The first one will include the user's personal information. The user's wish list will be saved in another table and his personal calendar will be saved in another table also. The pictures a user uploads will be saved in another table. The friends calendar component will have all the places that the user's friends will visit. This will let the user know if a friend will visit common places and they can go together. They can also plan to visit a place if they want. This component will have a table with the information about the activities of the user’s friends. Additional tables that are going to be required are for payment services and for user’s friends. In the sign-up process the application will be storing the information of the user in a table and sending an email to confirm identity. To do this the application will use SendGrid’s API. The information about the credit cards will be asked in the sign-up process and validated too depending on what type of card it is. This server application will be using PostgreSQL, pgAdmin, IntelliJ, and Java Play.

# **E-R Diagram**



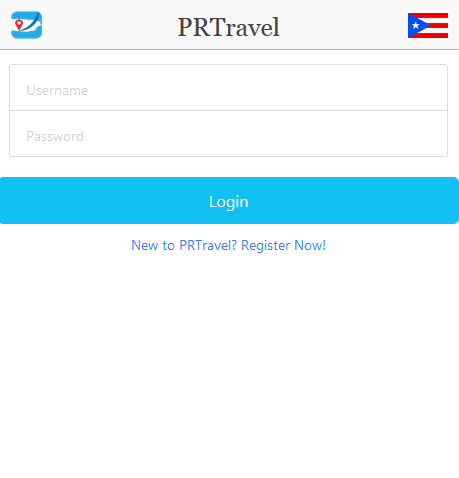
**Explanation**

User has many relationships. The first with friends which has two roles the labels are the id and friend id. It is a many to many relationship. A user is associated with several friends and a friend is a user at the same time. A notification has a many to one relationship with user. Notifications is associated with at most one user via has notification and a user is associated with several (including 0) notifications. A user can have many notifications. User has a one to one relationship with settings. A user is associated with at most one settings option and a setting is associated with only one user. The settings, for example to change email and password, can be done by at most one user. A comment has a many to one relationship with user. A comment is associated with at most one user since all the comments are going to the user but a user is associated with several (including 0) comments. A user can post as many comments as he wants.

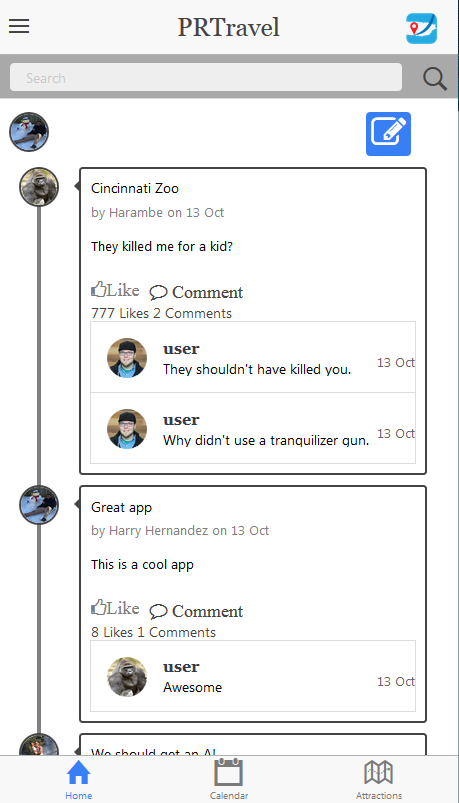
An event has a many to one relationship with user. An event is associated with at most one user but a user can post as many events as he wants in the calendar. User has a many to many relationship with attractions since a user is associated with several attractions and attractions is associated with many users. Comments and posts have a many to one relationship. You can post as many comments as you like via post comment. A comment is associated with at most one post. Comments and attractions have a many to one relationship. An attraction can have as many comments as it wants and a comment is associated with at most one attraction. Service has a many to one relationship with attractions via has service since an attraction can have as many services as it can hold but a service is associated with at most one attraction.

An attraction can has as many pictures as it wants and a picture is associated with at most one attraction. This is then a many to one relationship.

# **Client-Side Screen Shots**



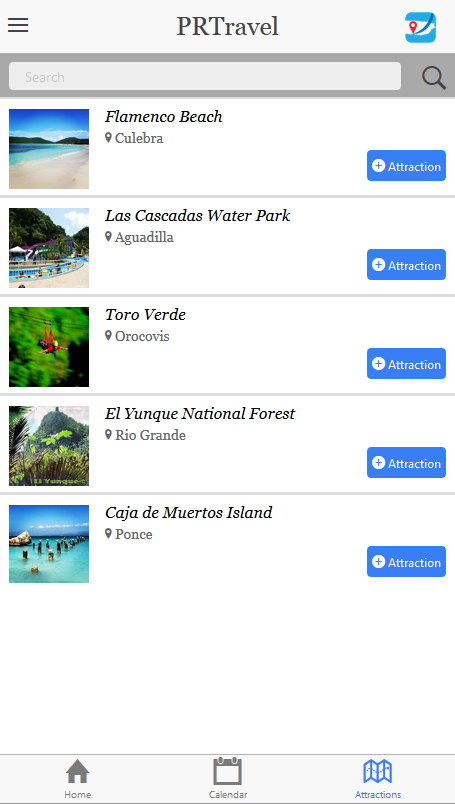
**Figure 1:** Login page.



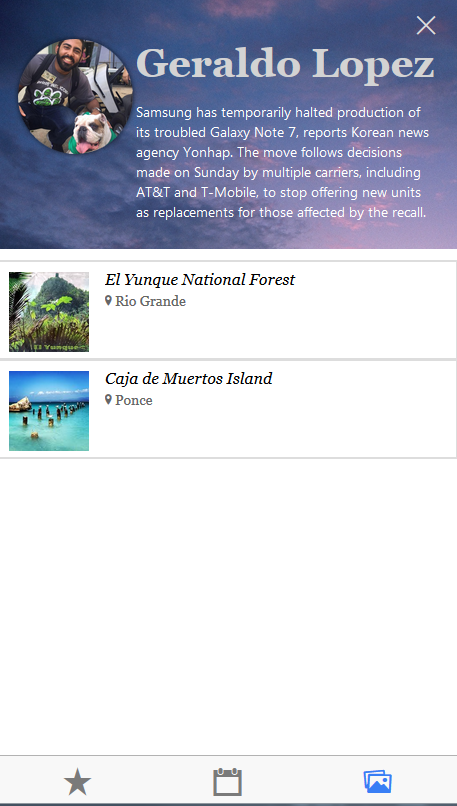
**Figure 2:** Newsfeed tab.



**Figure 3:** Calendar tab.



**Figure 4:** Attractions tab.



**Figure 5:** Profile page.

# **Bibliographies**

Quintero, Laura. "Aseguran Puerto Rico No Puede Pagar Toda Su Deuda." *ElVocero*. N.p., 30 June 2016. Web. 08 Sept. 2016.

"¿Cómo Llegó Puerto Rico Al Mayor Incumplimiento De Deuda De Su Historia?" *BBC Mundo*. N.p., 2 May 2016. Web. 08 Sept. 2016.

"Other Useful Resources for Our Investors." *Puerto Rico Tourism Company*. N.p., 2016. Web. 08 Sept. 2016.