



TROT RIGHT IN

Final Technical Report

BAM! Software

Frontend: Evan Kohn. Jim Lomeo. Justin Trantham.

Backend: Jordan Kayse. Jim Lomeo. Jessica Yeh. Story Zanetti.

Introduction

PonyPark was developed to help users find a parking spot. Specifically, PonyPark was developed for members of the Southern Methodist University community. The PonyPark team is comprised of six students at Southern Methodist University. Jordan Kayse, Jim Lomeo, Jessica Yeh, and Story Zanetti constructed the backend software. Evan Kohn, Jim Lomeo, and Justin Trantham developed the frontend software, and Justin Trantham developed the Android application. This document will outline the technology that was used to create the backend and frontend software. It is our hope that this application improves SMU community members' commutes to campus.

Software Features

Visitor

- Visitors can view garage information in list and map form.
- Visitors can view top ten contributing users.
- Visitors can view the average rating of a garage and each of its levels in the last two hours, or the most recent rating if there is no rating for the last two hours.
- Visitors can view a graph of the average availability of a garage for each hour of the day.
- Visitors can create accounts using Facebook or Google.
- Visitors can create generic accounts using their email and a password.

User

- Users can access all functionality that visitors can (expect for creating accounts).
- Users can log in using Facebook or Google.
- Users can log in using their email and a password.
- Users can add their phone numbers.
- Users can rate the availability of garages. Each of these ratings gives the user a point toward being a top contributing user.
- Users can add garages to a list of favorite garages.
- Users can select times during each day of the week to be notified.
- Users can be notified via email about the status of their favorite garages at their notification times.
- Users can request garages to be added.
- Users can view requests, and their acceptance status, for garages they've made.
- Users can view their profile information.
- Users can edit their profile information.

Admin

- Admins can access all functionality that users can.
- Admins can view all requests for new garages that have not been approved or denied.
- Admins can approve or decline requests for new garages.

Use Case Diagram

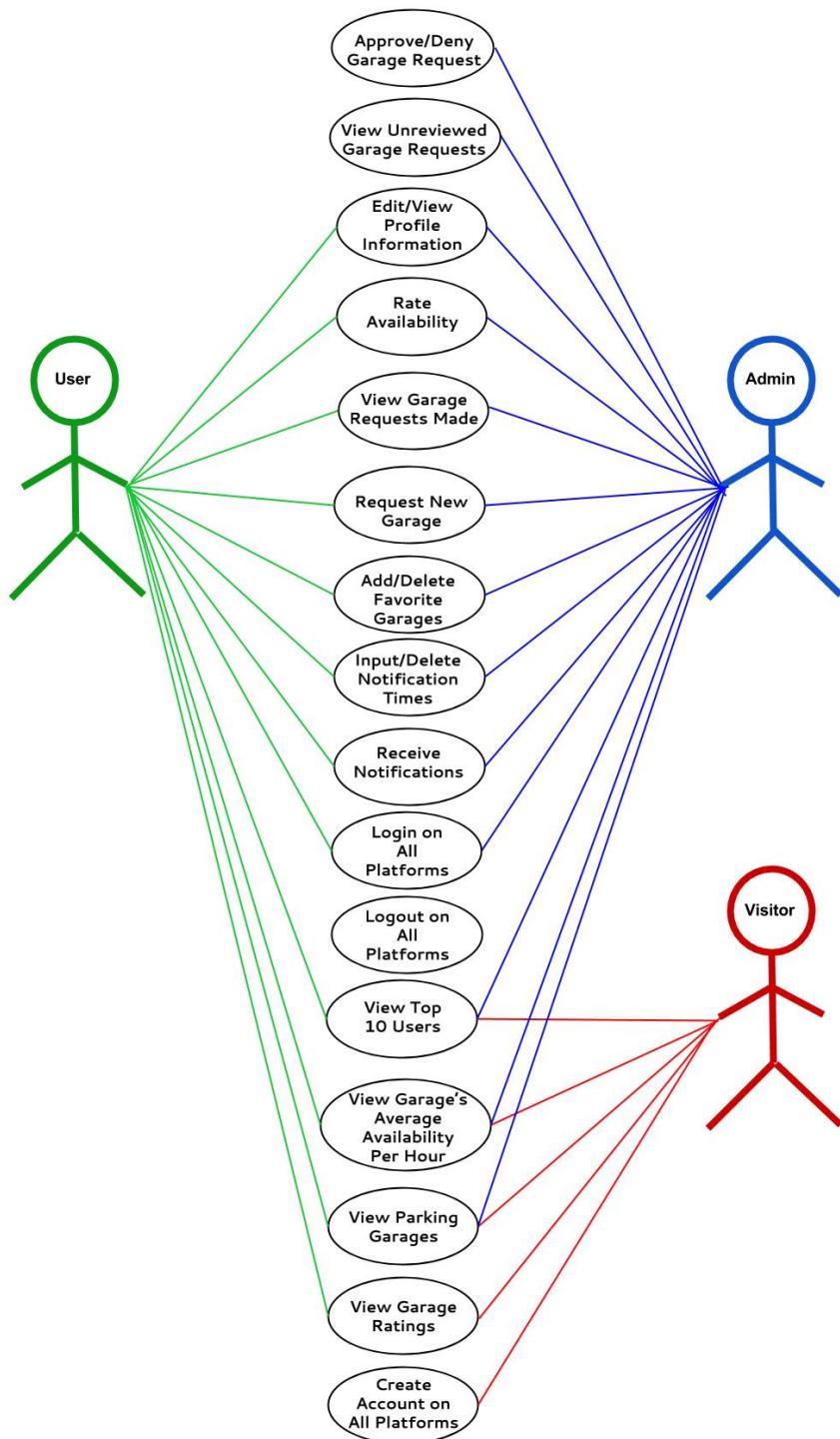


Figure 1: A Use Case diagram of our software.

The above *Figure 1* is a diagram of what each different kind of user is able to do. This is a visual representation of the requirements our software will have to meet to satisfy the needs of all types of users.

Database Model

Legend: Primary key ♀ Foreign Key ♦ Can be Null □ Can't be Null ■

Entity	Attributes
Users	♀ UserID INT ■ FirstName VARCHAR(50) ■ LastName VARCHAR(50) ■ Email VARCHAR(50) □ Password VARCHAR(64) □ PasswordSalt VARCHAR(50) □ PhoneNumber VARCHAR(20) ■ UserType INT ■ ExternalType VARCHAR(10) □ ExternalID VARCHAR(64)

Unique Keys for Users: Email and ExternalType

Table 1.a: The Users table, and its attributes and unique keys.

Entity	Attributes
ParkingLocations	♀ ParkingID INT ■ Name VARCHAR(50) ■ Address VARCHAR(50) □ Cost DOUBLE ■ NumberOfLevels INT

Table 1.b: The ParkingLocations table, and its attributes.

Entity	Attributes
Ratings	♀ RatingID INT ♦ ParkingID INT ♦ UserID INT ■ Timestamp DATETIME ■ Rating INT ■ Level INT

Table 1.c: The Ratings table, and its attributes.

Entity	Attributes
CommuteTimes	◆ CommuteID INT ◆ UserID INT ■ Day INT ■ WarningTime TIME ■ TimeOfNotification DATETIME

Unique Keys for CommuteTimes: UserID, Day, and WarningTime

Table 1.d: The CommuteTimes table, and its attributes and unique keys.

Entity	Attributes
Requests	◆ RequestID INT ◆ UserID INT ■ Name VARCHAR(50) ■ Address VARCHAR(50) □ Cost DOUBLE ■ NumberOfLevels INT □ Comments VARCHAR(250) ■ Status INT

Table 1.e: The Requests table, and its attributes.

Entity	Attributes
TopTen	◆ UserID INT ■ Points INT

Table 1.f: The TopTen table, and its attributes.

Entity	Attributes
FavoriteGarages	◆ FavoriteID INT ◆ UserID INT ◆ ParkingID INT ■ Priority INT

Unique Keys for FavoriteGarages: UserID and ParkingID

Table 1.g: The FavoriteGarages table, and its attributes and unique keys.

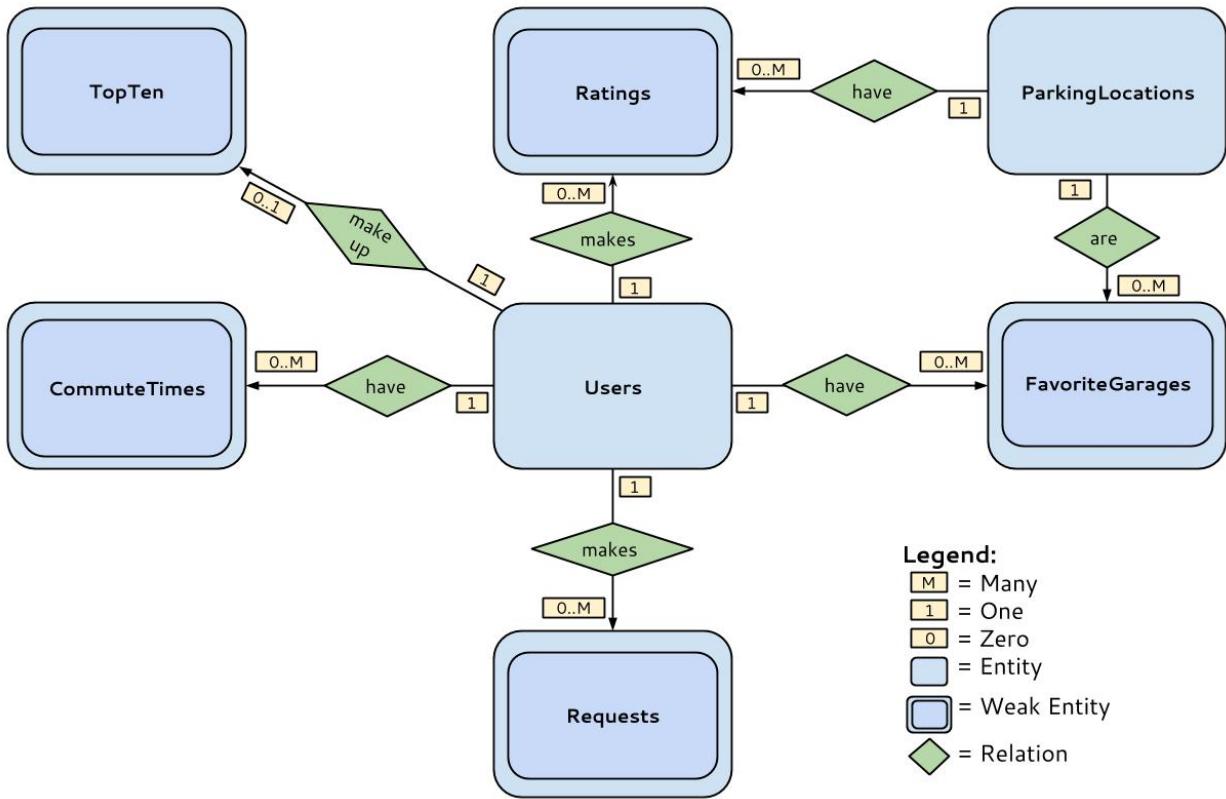


Figure 2: A ER diagram of our database.

Table 1 shows all the different tables with all their attributes, and *Figure 2* shows the relationships between all the tables in the database.

Detailed Software Architectural Diagram

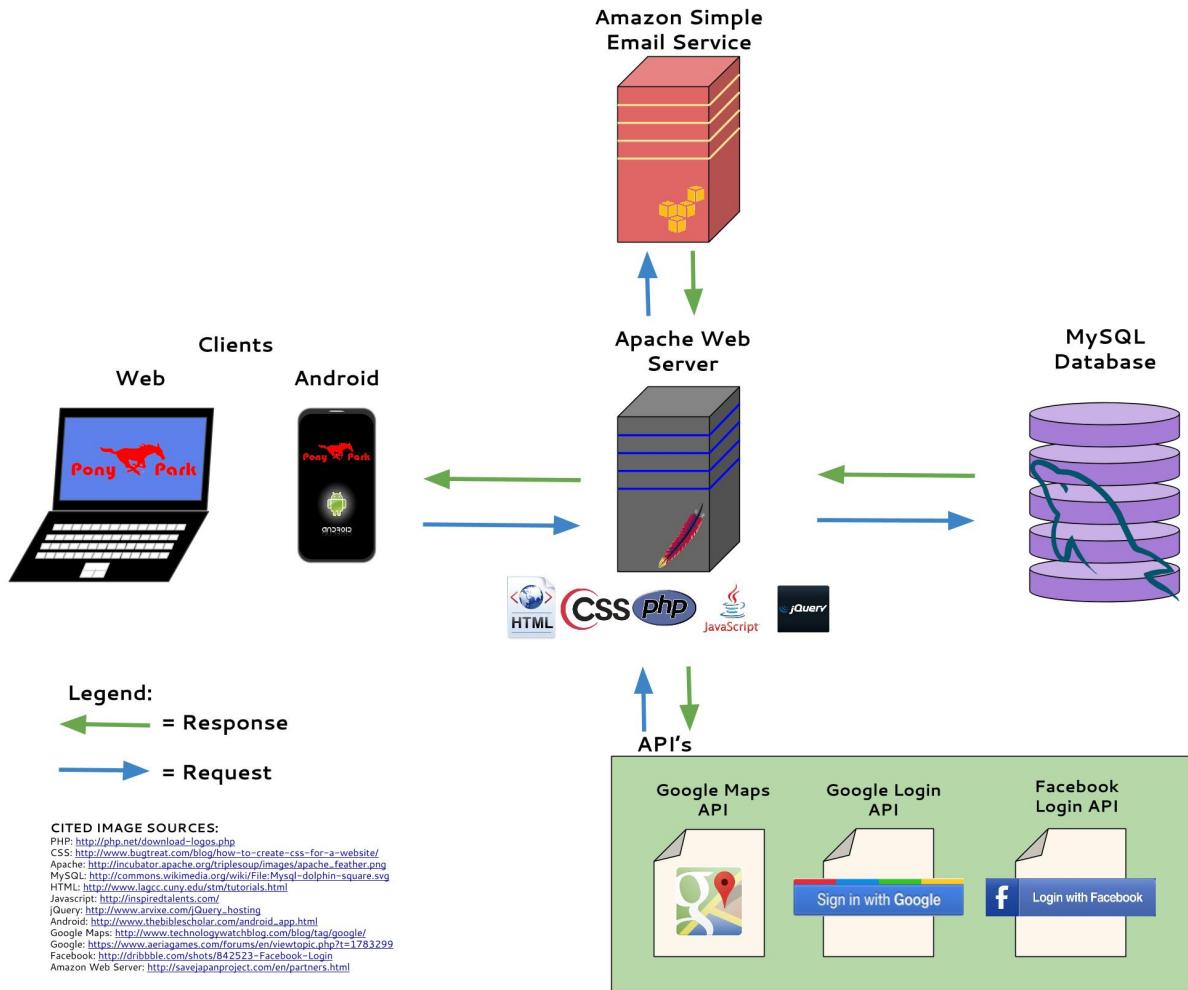


Figure 3: Detailed Software Architecture Diagram.

The Web Client in *Figure 3* represents the website version of our product, accessible at <http://ponypark.floccul.us>. For optimum functionality, the user should use Google Chrome when accessing the website. The Web Client communicates with the Apache Web Server to retrieve the pages of our website. The Android Application in *Figure 3* also communicates with the Apache Web Server, but it does so indirectly, using links to get information via JSON.

On the Apache Web Server in *Figure 3*, we use HTML to create our web pages and CSS to add decorative elements to these pages. We also utilize JavaScript and jQuery to change the content of our pages and to handle asynchronous aspects. Finally, we use PHP to communicate with the MySQL database, primarily in our file `phpapi.php`, where we wrote functions for all the queries we need to execute. The Apache Web Server sends all this information to the Web Client.

As previously stated, the Apache Web Server communicates with the MySQL Database through PHP. The MySQL Database holds all the information for our website (user information, parking garage information, etc.). The MySQL Database receives requests for information from the Apache Web Server and sends back that requested information.

A cron job is used to run a PHP script every 5 minutes to determine if any email notifications need to be sent. If emails need to be sent, the PHP file from the Apache Web Server constructs the contents of the email and then communicates with the Amazon Simple Email Service in *Figure 3* to have the emails sent to the users.

Finally, the Apache Web Server communicates with three APIs. It uses the Google Login API and Facebook Login API to allow users to sign in to our website using their Google and Facebook accounts, respectively. The Apache Web Server communicates with the Google Maps API to create the map viewable on the homepage. It places pins on this map through this API using the stored addresses of each garage.

Web UI

Welcome to PonyPark, Guest! Please [Join](#) or [Sign in](#)

[g+](#) [Sign in](#) [f Sign in](#)

[Trot Right In](#) [Home](#) [Top Users](#) [About](#) [Contact](#)

Parking Availability

Use the map below or the list view to the right to see the parking availability.

Airline Garage
6506 Airline Rd
Average: Full

Binkley Garage
3105 Binkley Avenue
Most Recent (10 hours ago): Some

Law Parking Center
3330 Daniel Avenue
Most Recent (>24 hours ago): Some

Meadows Museum Garage
5900 Bishop Boulevard
Most Recent (8 hours ago): Empty

Moody Garage
3063 SMU Blvd
Most Recent (4 hours ago): Some

PonyPark | Southern Methodist University | Dallas, Texas 2013

[Contact Us](#) | [About](#)

Figure 4.a: Home Screen (Visitor)

When a new user visits the website, the first thing he or she will see is the Home Screen. At the top of the Home Screen is the dashboard, allowing users to Join or Sign In. The dashboard also contains navigation to other parts of the website. This dashboard is constant among all pages of the website. The Home Screen contains both the Map and the List Views of the current garages. The Map will be zoomed in to the SMU campus by default. Red pins are dynamically added to the Map for every garage location. The same data that is displayed on the Map View will be shown in the List View on the right. Clicking on the name of a garage will redirect the user to that particular garage's page. The colored box around each garage signifies the current rating of the garage. Green signifies Empty or Plenty, Yellow signifies Some, and Red signifies Scarce or Full.

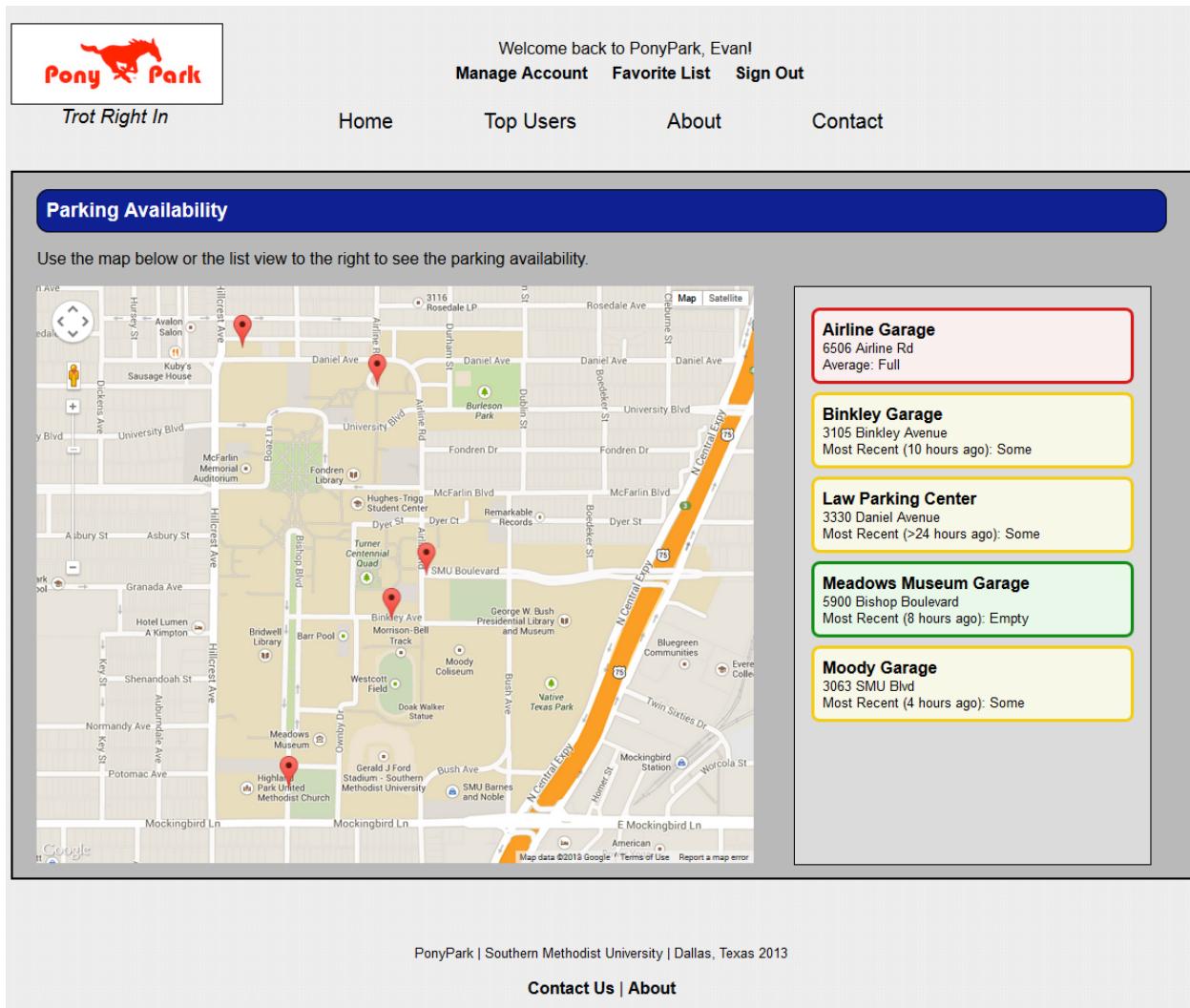


Figure 4.b: Home Screen (User)

After successfully signing in, the Home Screen now appears as shown in *Figure 4.b*. Additional links have been added to the dashboard allowing the user to manage his or her account, see his or her list of favorite garages, or sign out.



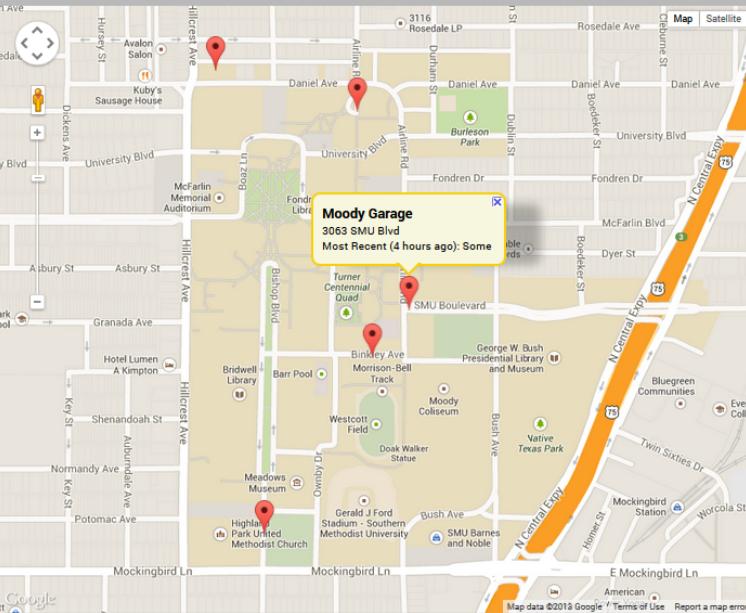
Welcome back to PonyPark, Evan!

[Manage Account](#) [Favorite List](#) [Sign Out](#)

[Trot Right In](#) [Home](#) [Top Users](#) [About](#) [Contact](#)

Parking Availability

Use the map below or the list view to the right to see the parking availability.



The map displays several parking locations with red pins and corresponding info boxes. One pin is highlighted with a yellow border, indicating recent activity.

Garage Name	Address	Last Update
Airline Garage	6506 Airline Rd	Average: Full
Binkley Garage	3105 Binkley Avenue	Most Recent (10 hours ago): Some
Law Parking Center	3330 Daniel Avenue	Most Recent (>24 hours ago): Some
Meadows Museum Garage	5900 Bishop Boulevard	Most Recent (8 hours ago): Empty
Moody Garage	3063 SMU Blvd	Most Recent (4 hours ago): Some

Airline Garage
 6506 Airline Rd
 Average: Full

Binkley Garage
 3105 Binkley Avenue
 Most Recent (10 hours ago): Some

Law Parking Center
 3330 Daniel Avenue
 Most Recent (>24 hours ago): Some

Meadows Museum Garage
 5900 Bishop Boulevard
 Most Recent (8 hours ago): Empty

Moody Garage
 3063 SMU Blvd
 Most Recent (4 hours ago): Some

PonyPark | Southern Methodist University | Dallas, Texas 2013

[Contact Us](#) | [About](#)

Figure 4.c: Home Screen Map Pins

Clicking on a Pin on the Map opens up an info box identical to the one found in the List View. Clicking the name of the garage will also redirect the user to that particular garage's page.



Welcome to PonyPark, Guest! Please [Join](#) or [Sign in](#)

[g+](#) [Sign in](#) [f](#) [Sign in](#)

[Trot Right In](#) [Home](#) [Top Users](#) [About](#) [Contact](#)

Moody Garage

**3063 SMU Blvd
Dallas, Texas 75205**

The current rating of the garage is **SOME**.



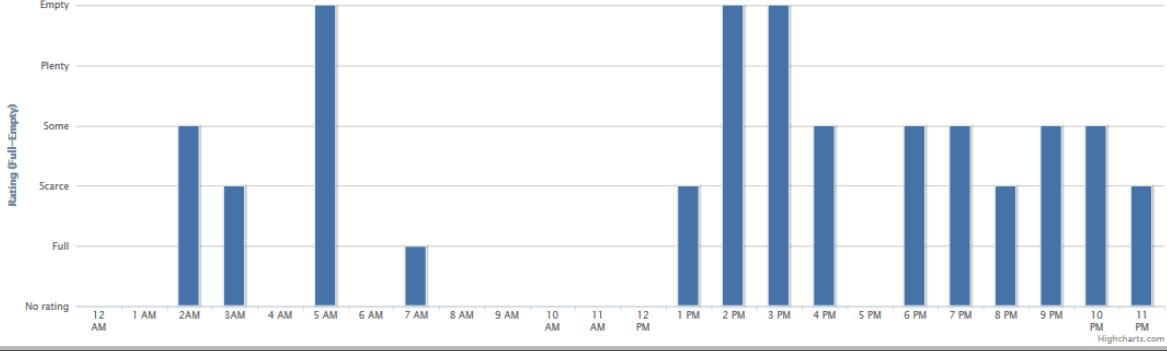
PonyPark Needs You!

Please help contribute to PonyPark. PonyPark relies on users just like you to report the current parking conditions on campus. All you need to do is join or sign in to get started!

Level Information

- Level 1**
Most Recent (>24 hours ago): Scarce
- Level 2**
Be the first to rate this level!
- Level 3**
Most Recent (4 hour(s) ago): Some
- Level 4**
Be the first to rate this level!
- Level 5**
Be the first to rate this level!

Average Rating For Each Hour of the Day



Hour	Rating
12 AM	Empty
1 AM	Plenty
2 AM	Some
3 AM	Scarce
4 AM	No rating
5 AM	Full
6 AM	Scarce
7 AM	Some
8 AM	No rating
9 AM	No rating
10 AM	No rating
11 AM	No rating
12 PM	No rating
1 PM	Scarce
2 PM	Full
3 PM	Full
4 PM	Some
5 PM	No rating
6 PM	Scarce
7 PM	Scarce
8 PM	Scarce
9 PM	Scarce
10 PM	Scarce
11 PM	Scarce

PonyPark | Southern Methodist University | Dallas, Texas 2013
[Contact Us](#) | [About](#)

Figure 4.d: Garage (Visitor)



Welcome back to PonyPark, Evan!
[Manage Account](#) [Favorite List](#) [Sign Out](#)

[Trot Right In](#) [Home](#) [Top Users](#) [About](#) [Contact](#)

Moody Garage

**3063 SMU Blvd
Dallas, Texas 75205**

The current rating of the garage is **SOME**.



Rate the Availability

Which level did you park your car? **1** ▾

How would you rate the availability of parking?

- Full garage
- Scarce parking spots
- Some parking spots
- Plenty parking spots
- Empty garage

[Rate](#)

Level Information

Level 1
Most Recent (>24 hours ago): Scarce

Level 2
 Be the first to rate this level!

Level 3
Most Recent (4 hour(s) ago): Some

Level 4
 Be the first to rate this level!

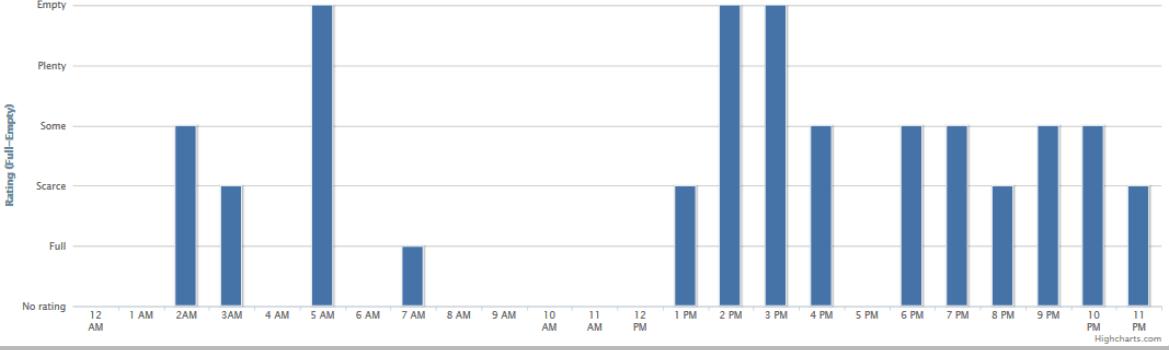
Level 5
 Be the first to rate this level!

Favorite This Garage

If you like this garage, add it to your favorites!

[Add To Favorites](#)

Average Rating For Each Hour of the Day



Rating	12 AM	1 AM	2 AM	3 AM	4 AM	5 AM	6 AM	7 AM	8 AM	9 AM	10 AM	11 AM	12 PM	1 PM	2 PM	3 PM	4 PM	5 PM	6 PM	7 PM	8 PM	9 PM	10 PM	11 PM
Empty																								
Plenty																								
Some																								
Scarce																								
Full																								
No rating																								

Highcharts.com

PonyPark | Southern Methodist University | Dallas, Texas 2013
[Contact Us](#) | [About](#)

Figure 4.e: Garage (User)

13

After a user selects a garage, he or she is brought to that garage's page as seen in *Figure 4.d* (*Figure 4.e* if the user is signed in). At the top, the garage's name and address are displayed, as well as the current overall rating and a picture, if they exist. Under that is the Rating form. As shown in *Figure 4.d*, if the user is not signed in then he or she is unable to rate the availability of the garage. *Figure 4.e* shows the Rating form, allowing the user to select the level of the garage he or she parked in and an availability rating on a scale of Full–Empty. To the right is the Level Information for the garage, which breaks down the rating into ratings for each level or floor of the garage, if there is one available. If there have been ratings within the past 2 hours, then an Average Rating will be shown. Otherwise, only the Most Recent rating is shown. Next, as seen in *Figure 4.e*, if the user is signed in then a button is displayed allowing the user to add this garage to his/her list of Favorites. Finally, at the bottom of the screen is a graph that shows the average rating of the garage for each hour of the day, on a scale of Full–Empty.

The screenshot shows the PonyPark website's sign in and sign up interface. At the top left is the logo 'Pony Park' with a horse icon. Below it is the tagline 'Trot Right In'. To the right is a welcome message: 'Welcome to PonyPark, Guest! Please Join or Sign in'. Below this are two social sign-in buttons: 'g+' and 'f'. The main navigation menu includes 'Home', 'Top Users', 'About', and 'Contact'. The central area contains two forms. The top form is for logging in, with fields for 'Email Address' and 'Password', and a blue 'Sign In' button. The bottom form is for creating a new account, with fields for 'First Name', 'Last Name', 'Email Address', 'Password', and 'Phone Number', followed by a blue 'Sign Up' button. At the very bottom of the page, the footer reads 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and includes links for 'Contact Us' and 'About'.

Figure 4.f: Sign In and Sign Up

Clicking the Join or Sign In links in the dashboard brings the user to the page seen in *Figure 4.f*. Here the user can sign in using their email and password, or sign up for an account. The dashboard also provides buttons allowing the user to sign in with their Google+ or Facebook accounts. When a user signs up for a new account, he or she will automatically be logged in. Asterisks (*) are used to mark required fields, and hovering over a field gives more information about the requirements of the field.

The screenshot shows the PonyPark website's Manage Account page. At the top left is the logo 'Pony Park' with a red horse icon. To the right is a welcome message 'Welcome back to PonyPark, User!' and links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the header are navigation links: 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the title 'Manage Account'. The main content area is titled 'Please choose an action you would like to perform:' and lists several categories with sub-links:

- Profile**
 - [View Your Profile Info](#)
 - [Edit Your Profile Info](#)
- Favorite Garages**
 - [View Favorites](#)
 - [Add/Remove Favorites](#)
- Notification Settings**
 - [View/Edit/Delete Notification Times](#)
 - [Add Notification Time](#)
- Request Garage Additions**
 - [Status of Garage Addition Requests](#)
 - [Request Addition of a Garage to PonyPark](#)

At the bottom of the page, a footer bar includes the text 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Figure 4.g: Manage Account

The Manage Account page allows users to view and make changes to their account as well as to access additional features of the site. The links, as seen in *Figure 4.g*, let the user View or Edit their Profile Info, View or Add/Remove their Favorite Garages, View/Edit/Delete or Add a new Notification Time for their Favorites, and Request a new Garage to be added to PonyPark or see the status of any previously sent Requests.

The screenshot shows the PonyPark website interface. At the top left is the logo 'Pony Park' with a red horse icon. To the right, the text 'Welcome back to PonyPark, Admin!' is displayed, followed by links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below this is a horizontal navigation bar with links 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the text 'Manage Account'. The main content area is titled 'Admin Dashboard' and contains several sections: 'Profile' (with 'View Your Profile Info' and 'Edit Your Profile Info' links), 'Favorite Garages' (with 'View Favorites' and 'Add/Remove Favorites' links), 'Notification Settings' (with 'View/Edit/Delete Notification Times' and 'Add Notification Time' links), and 'Request Garage Additions' (with 'Status of Garage Addition Requests' and 'Request Addition of a Garage to PonyPark' links). At the bottom of the page, a footer bar includes the text 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Welcome back to PonyPark, Admin!

Manage Account Favorite List Sign Out

Trot Right In Home Top Users About Contact

Manage Account

Please choose an action you would like to perform:

Admin Dashboard

Profile

[View Your Profile Info](#)

[Edit Your Profile Info](#)

Favorite Garages

[View Favorites](#)

[Add/Remove Favorites](#)

Notification Settings

[View/Edit/Delete Notification Times](#)

[Add Notification Time](#)

Request Garage Additions

[Status of Garage Addition Requests](#)

[Request Addition of a Garage to PonyPark](#)

PonyPark | Southern Methodist University | Dallas, Texas 2013

[Contact Us](#) | [About](#)

Figure 4.h: Manage Account with Admin Dashboard

If the logged in user is an admin, then in their Manage Account page, he or she will also see a link that goes to their Admin Dashboard, as shown in *Figure 4.h*. Admins have access to all the functionality that a regular user has.

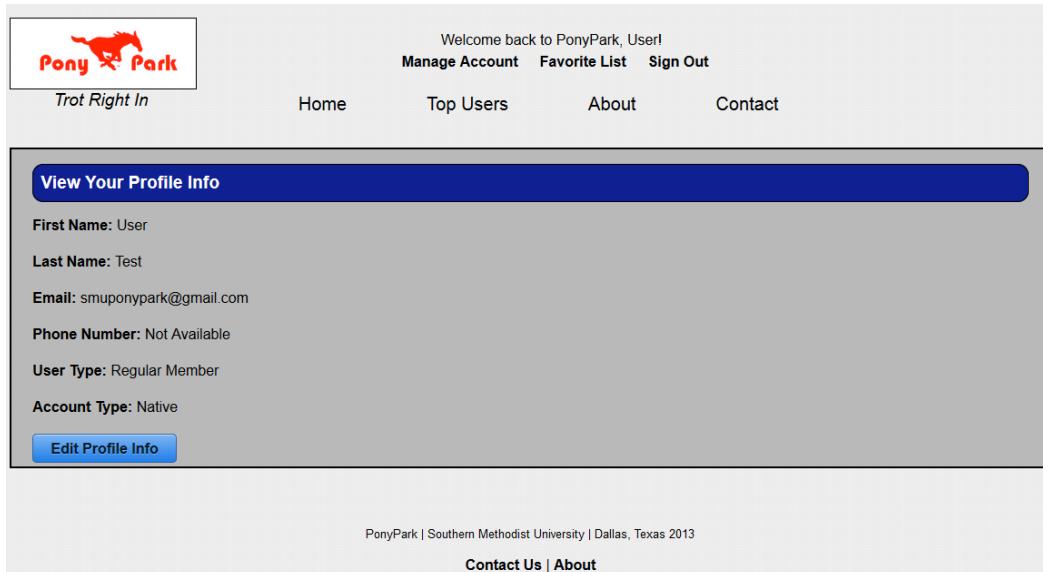


Figure 4.i: View Profile Info

The View Profile Info page in *Figure 4.i* allows the user to see his or her current info including name, email, phone number, user type (Regular Member or Admin), and account type (Native, Facebook, or Google). The button at the bottom redirects the user to the Edit Profile Info page.

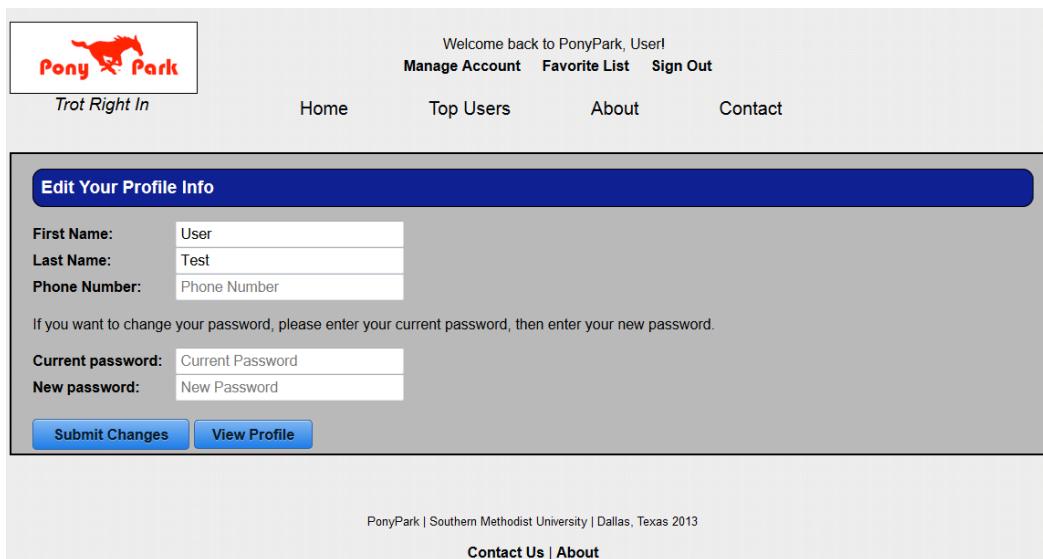


Figure 4.j: Edit Profile Info

The Edit Profile Info page allows the user to edit his or her name, phone number, and password if he or she chooses, although the user can only edit his or her password if he or she is signed in natively. At the bottom are buttons to submit the changes, and to redirect the user back to the View Profile Info page.

The screenshot shows the 'View Favorites' page of the PonyPark application. At the top left is the logo 'Pony Park' with a red horse icon. To the right, the text 'Welcome back to PonyPark, User!' is displayed, followed by links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below this is a horizontal navigation bar with links for 'Home', 'Top Users', 'About', and 'Contact'. A large blue header bar contains the text 'Favorite List'. Underneath it is a blue button labeled 'Edit Favorites'. The main content area displays a message: 'Below are your favorite garages. Click on a garage to get more detailed real time info.' Below this, a single garage entry is shown: 'Airline Garage', 'Address: 6506 Airline Rd', and 'Most Recent (2 hours ago): Full'. At the bottom of the page, there is footer text: 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Figure 4.k: View Favorites

The View Favorites page displays the user's Favorite Garages, with address and current rating. Clicking on the name redirects to that garage's page. At the top is a button that redirects the user to the Edit Favorites page. This page can also be accessed via the Favorite List link in the dashboard.

The screenshot shows a web application interface for managing favorite garages. At the top left is the logo 'Pony Park' with a red horse icon. To its right, the text 'Welcome back to PonyPark, User!' is displayed, followed by links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below this, a horizontal navigation bar includes 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the title 'Edit Favorite Garages'. Below it is a blue button labeled 'View Favorites'. The main content area displays five garage entries, each with a name, address, and an 'Add To Favorites' link:

- Airline Garage**
6506 Airline Rd
[Remove From Favorites](#)
- Binkley Garage**
3105 Binkley Avenue
[Add To Favorites](#)
- Law Parking Center**
3330 Daniel Avenue
[Add To Favorites](#)
- Meadows Museum Garage**
5900 Bishop Boulevard
[Add To Favorites](#)
- Moody Garage**
3063 SMU Blvd
[Add To Favorites](#)

At the bottom of the page, a footer section contains the text 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and a link 'Contact Us Policies'.

Figure 4.1: Edit Favorite Garages

The Edit Favorite Garages page allows the user to remove any of their current favorites from the list, as well as add any garage in the system to their favorites. Clicking on the name of the garage redirects to that garage's page. At the top is a button that redirects the user to the View Favorites page.



Welcome back to PonyPark, User!

[Manage Account](#) [Favorite List](#) [Sign Out](#)

[Trot Right In](#) [Home](#) [Top Users](#) [About](#) [Contact](#)

Add Notification Times

Please select the time you'd like to be notified of parking availability:

11:30

Please select what days of the week your commute begins at this time:

Sunday
 Monday
 Tuesday
 Wednesday
 Thursday
 Friday
 Saturday

[Submit](#) [View Notifications](#)

PonyPark | Southern Methodist University | Dallas, Texas 2013
[Contact Us](#) | [About](#)

Figure 4.m: Add Notification Times

The Add Notification Times page allows the user to set a time and day(s) for email notifications to be sent to them about the current availability of their favorite garages. These notifications are sent using Amazon's Simple Email Service. A button at the bottom redirects the user to the View/Edit/Delete Notification Times page.

The screenshot shows a web application interface for managing notification times. At the top left is the logo 'Pony Park' with a horse icon. To the right, the text 'Welcome back to PonyPark, User!' is displayed, followed by links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the header, there are navigation links: 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the title 'Modify/Delete Notification Times'. Below it is a button labeled 'Add Notifications'. A message instructs the user to use the form below to modify notification times. The main content area is titled 'Notification Time 1' and contains the following information:

Modify your current times and hit update to save the changes. To delete a specific notification time, hit delete.

Your notification time is set to: 00:00:00

Your current notification days are:

Sunday
 Monday
 Tuesday
 Wednesday
 Thursday
 Friday
 Saturday

At the bottom of this section are two buttons: 'Update' and 'Delete This Time'.

At the very bottom of the page, the footer includes the text 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Figure 4.n: View/Edit/Delete Notification Times

The View/Edit/Delete Notification Times page lists each Notification Time that currently exists for the user and allows the user to modify the time, change which days the notification happens on, or delete it altogether.



Trot Right In

Home Top Users About Contact

Add a Garage to the PonyPark System

Please fill out all required fields.

Please ensure your garage is located on the SMU Campus or your request will be denied. Please do not include any zip codes, states, or cities when submitting the address. All fields are required except Comments and Cost.

Name of Garage
 Street Address
 Cost
 Number of levels
 Comments

Submit Request

View Your Garage Requests

PonyPark | Southern Methodist University | Dallas, Texas 2013
[Contact Us](#) | [About](#)

Figure 4.o: Request Garage

The Request Garage page allows users to request garages not currently in the PonyPark system be added. The user must provide the garage's name, address, and number of levels. Both the cost and comments field are optional. The comments field allows the user to give helpful comments about the garage to the PonyPark administrators.

The screenshot shows a user logged into the PonyPark website. The top navigation bar includes links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the navigation, there are links for 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the title 'Request Status'. A blue button labeled 'Make a Garage Request' is visible. The main content area displays four past garage requests:

- Name of Garage:** hjhkhj
Address: jlkjl
Cost: None Specified
Levels: 8
Comments: None Entered
Status: Denied
- Name of Garage:** hjklhi
Address: fhfghj
Cost: None Specified
Levels: 3
Comments: None Entered
Status: Denied
- Name of Garage:** Fasdjk
Address: jkjkj
Cost: None Specified
Levels: 1
Comments: kjkj,kl
Status: Denied
- Name of Garage:** SMU Barnes and Noble Parking Lot
Address: 3060 Mockingbird Lane
Cost: \$0
Levels: 1
Comments: Usually very busy
Status: Approved

Figure 4.p: View Request Status

The View Request Status page allows the user to see the approval status of any of their past requests, whether it be pending, denied, or approved.

The screenshot shows the PonyPark website interface. At the top left is the logo 'Pony Park' with a horse icon. To the right, the text 'Welcome back to PonyPark, Evan!' is displayed, followed by links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the header, there are navigation links: 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. The main content area has a blue header bar with the text 'Top Ten Users'. Below this, a message reads: 'At PonyPark, we value those who help make our service a success. Here are our top ten contributors, updated once a day. Please, aspire to be like these individuals and contribute!'. A table lists the top ten users with their ranks, names, and points:

Rank	Name	Points
1	Jordan K.	70
2	Story Z.	52
3	Jessica Y.	18
4	Story Z.	9
5	Charlie V.	7
6	Justin T.	7
7	Joe S.	6
8	Story Z.	6
9	test m.	6
10	Raz F.	5

At the bottom of the page, there is a footer with the text 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Figure 4.q: Top Users

On the Top Users, page the top ten contributors to the garage rating system are displayed. As seen in *Figure 4.q* each contributor in the top ten is listed with their name and the number of points accumulated. Users gain a point each time they rate a garage. This provides another incentive for users to rate.

The screenshot shows the 'About' section of the PonyPark website. At the top, there is a navigation bar with links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the navigation bar, there are four main menu items: 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the question 'What's PonyPark?'. The main content area contains text explaining the service's purpose: finding parking spots on campus by checking reports from other users. Another blue header bar contains the question 'Can I use PonyPark anywhere?'. The footer of the page includes a copyright notice for 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Figure 4.r: About

Figure 4.r shows the description of our product on the About page, accessed via the dashboard link at the top or in the page footer. It details the purpose and goals of our product.

The screenshot shows the 'Contact' section of the PonyPark website. At the top, there is a navigation bar with links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the navigation bar, there are four main menu items: 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the question 'Contact PonyPark'. The main content area contains text providing the mailing address: 'Want to get ahold of PonyPark? Please send all inquiries to Southern Methodist University Attn: PonyPark 3052 Boaz Lane Dallas, Texas 75205'. The footer of the page includes a copyright notice for 'PonyPark | Southern Methodist University | Dallas, Texas 2013' and links for 'Contact Us' and 'About'.

Figure 4.s: Contact

Figure 4.s shows the Contact page which provides PonyPark's mailing address. The page is accessed via the dashboard link at the top or in the page footer.

The screenshot shows the PonyPark Admin Page. At the top left is the logo 'Pony Park' with a red horse icon. To its right is the welcome message 'Welcome back to PonyPark, Admin!' and links for 'Manage Account', 'Favorite List', and 'Sign Out'. Below the header are navigation links: 'Trot Right In', 'Home', 'Top Users', 'About', and 'Contact'. A blue header bar contains the text 'PonyPark Admin Page'. The main content area displays two requests for new garages:

Name of Garage: asdfghjkl
Address: asdfghj
Cost: None Specified
Levels: 1
Comments: None Entered

Name of Garage: asd
Address: asd
Cost: None Specified
Levels: 2
Comments: None Entered

For each request, there are two blue buttons: 'Approve Request' and 'Deny Request'.

PonyPark | Southern Methodist University | Dallas, Texas 2013
[Contact Us](#) | [About](#)

Figure 4.t: Admin Page

The Admin Page, as seen in *Figure 4.t*, is accessible only by user accounts of type Admin. The Admin Page displays all current requests for new garages, and buttons for the Admin to approve or deny the request.

Android UI

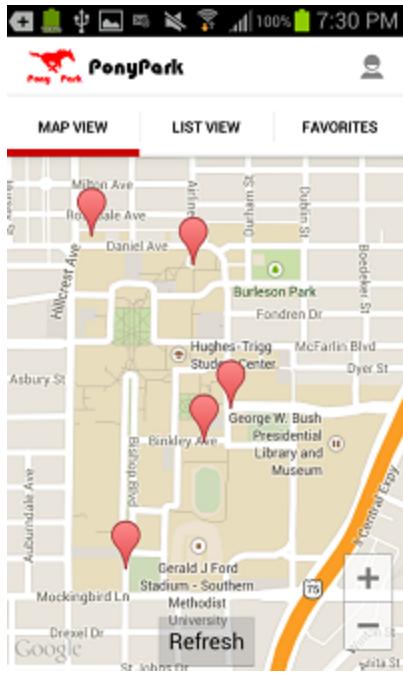


Figure 5.a: Map View

On launch of the PonyPark application, Map View will be the default view shown, zoomed to the SMU campus, as shown in *Figure 5.a*. Red pins will be dynamically added to the map of all the parking garage locations at SMU. The user can also select refresh to gather the latest parking information.

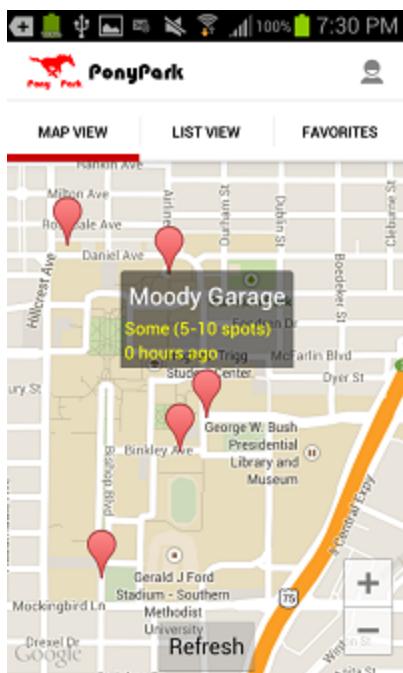
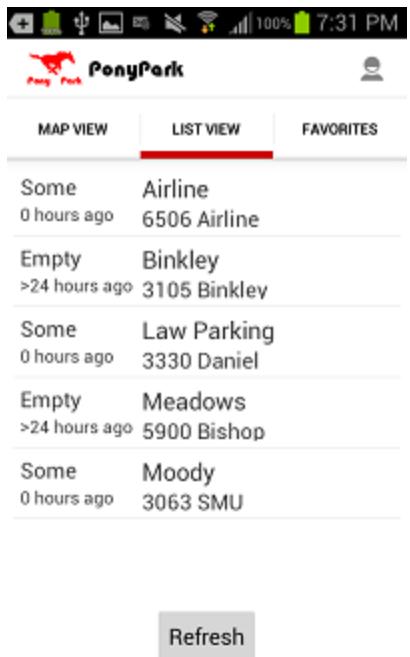


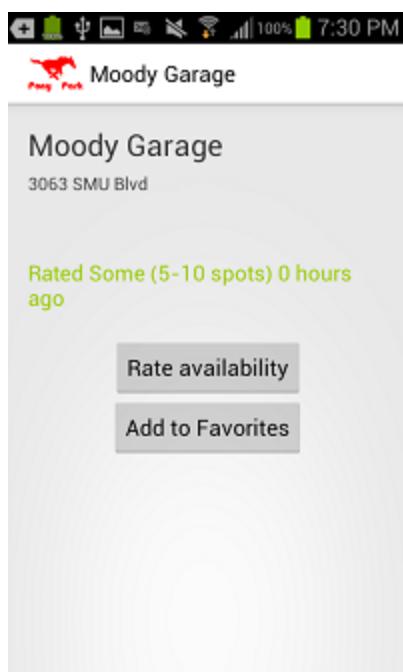
Figure 5.b: Map View Pins

Each pin will be clickable and will display a semi-transparent info box that shows the name and most recent rating of the selected garage, as shown in *Figure 5.b*.



The same data that is displayed on the Map View will be shown in a List View format as shown in *Figure 5.c*. Once clicked, the user will be directed to the same page as if clicked on the Map View.

Figure 5.c: List View



After a user selects a garage, he or she is presented with a screen as listed in either *Figure 5.d* or *Figure 5.e*. The screenshot in *Figure 5.d* presents the user the option to rate or add the garage to their favorites.

If the user already has the garage within their favorites list, then the screenshot in *Figure 5.e* will be shown. The “Add to Favorites” button will be removed and there will be a favorite icon on the upper right hand corner signifying the favorite is already added.

Figure 5.d: When the garage isn't favorited.

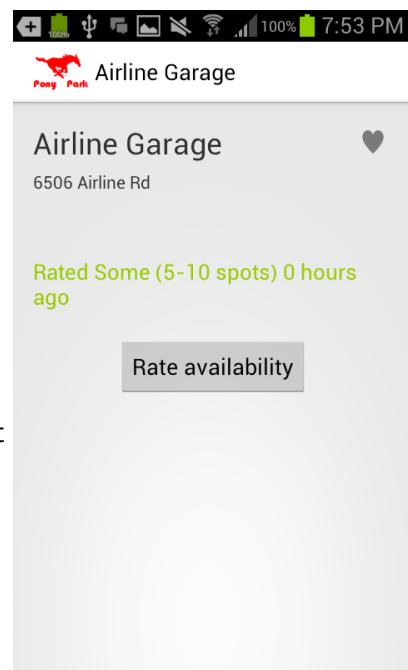
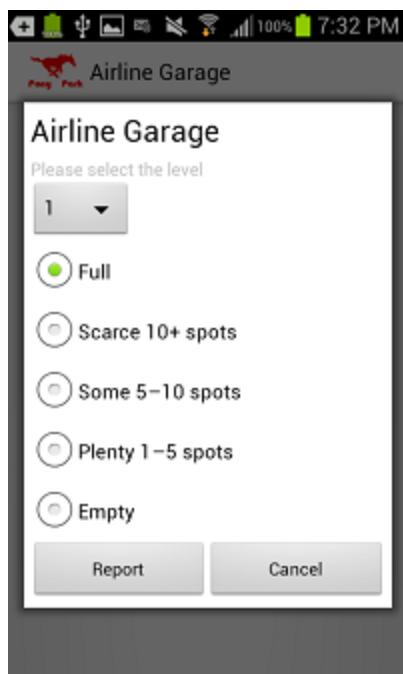


Figure 5.e: When the garage is favorited



After “Rate availability” is pressed then the user will be presented with a dialog as shown in *Figure 5.f* which will allow the user to rate the current parking garage availability.

If the user is not logged in and tries to add to favorites or rate the availability then this warning message in *Figure 5.g* will be shown.

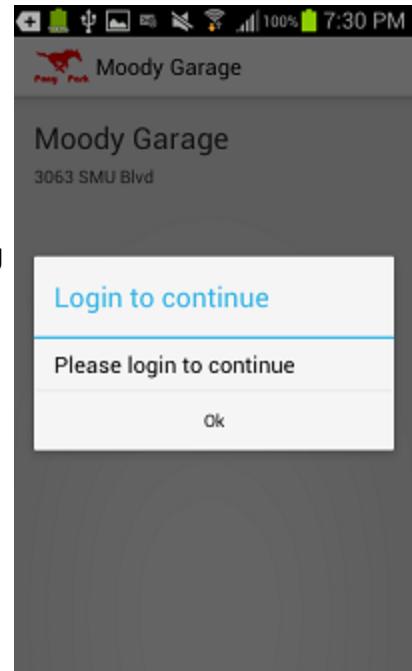
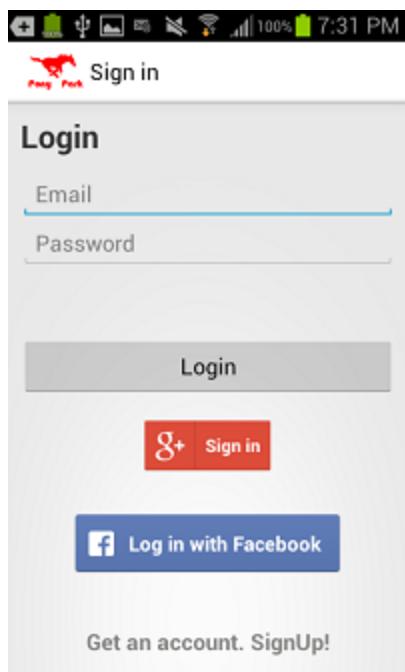


Figure 5.f: Rating

Figure 5.g: Not logged in



The most important and crucial part of this application is getting a large user base. Thus, the signup and login process is made as seamless as possible. So shown in *Figure 5.h* is the login screen where the user has the option to use Google+, Facebook, or just login with their username and password.

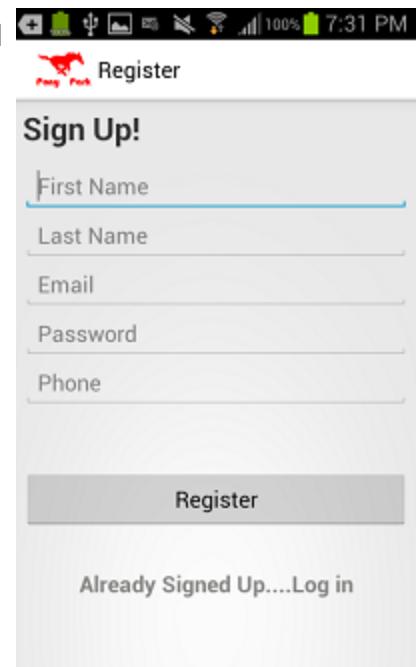


Figure 5.h: Logging In

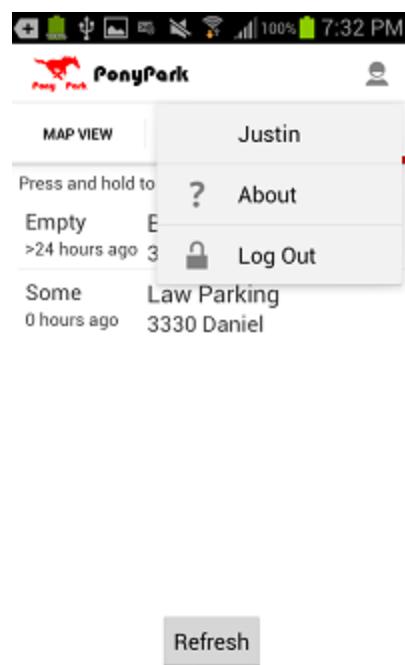
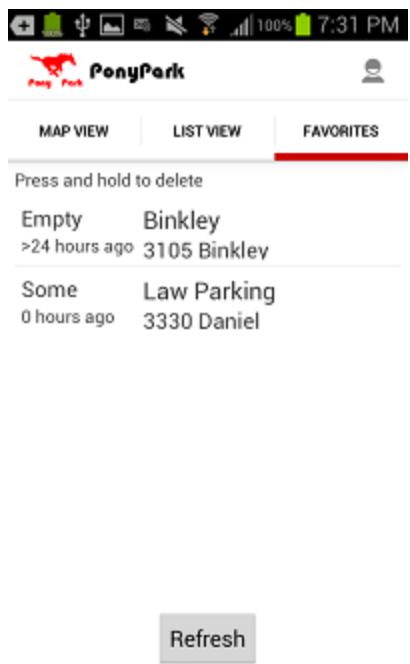


Figure 5.j: Greeting



Finally, presented in *Figure 5.k* is the user's favorites screen which is only shown when the user is logged in. Each favorite can be deleted by long pressing on an item to display a confirmation dialog to delete. As with the Map View and List View, this list can be refreshed to get the latest favorites list from the server.

Figure 5.k: Favorites

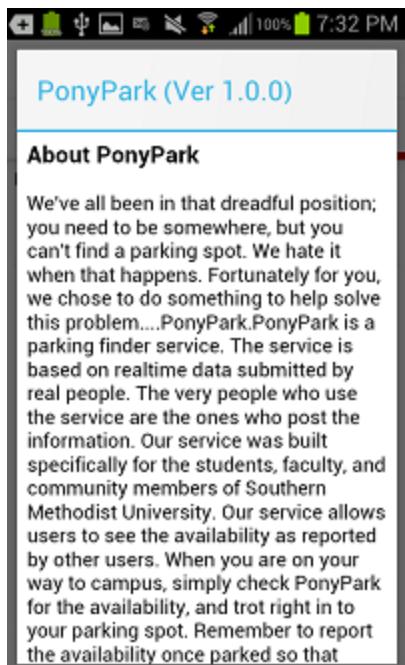


Figure 5.l shows the description of our product, available on the About page of the Android application. It details the purpose and goals of our product.

Figure 5.l: About Box

Testing

Testing is a vital part of software development. PonyPark has spent a significant amount of time testing and seeking others to test our product. While PonyPark never sought people specifically not enrolled in CSE 3330/3345, there have been outside users. The outside users were from a wide range of demographics. These users helped us identify bugs and usability errors.

While outside users were helpful, the most useful test results came from ourselves and the test team. There were numerous occasions where the PonyPark team spent an entire day simply testing and fixing bugs. During these “testing days” bugs would be squashed the moment they were identified. In regards to our test team, we usually found them helpful. It was difficult to take advice seriously from a certain test team member as he appeared to be simply reporting any minuscule item just to earn credit. However, that issue aside, our testing team provided us with very useful findings and information.

While the test team analyzed PonyPark, PonyPark was analyzing their code. It was very difficult to find bugs for this test team as their code always seemed to function appropriately. However, we were able to find some bugs. They seemed very grateful for the bugs that were found.

As we continued to test our code, we learned the importance of setting aside large amounts of time to ensure the service functioned properly under *every* circumstance.

Team Reflection

Working in a team can be helpful and stressful. Overall, working in a team was helpful. As with any situation there were challenges. At the very beginning, there were wide discrepancies on how the software should look and behave. Through various discussions and compromises we were able to decide on a direction for PonyPark. Throughout the course of the project, team members continued to communicate with each other. This was a vital part of PonyPark's success. When a particular group member was unhappy with the outcome of another's task, communication ensued. This communication allowed the problem to be quickly identified, and immediately solved.

Oftentimes, independent learning and research must occur to implement a fully functioning task. This occurred numerous times with PonyPark and the Custom Cupcake assignment. At the beginning of Custom Cupcake, it was unclear how the frontend and middleware would communicate. Through various websites and trial and error, we were able to solve the problem by using JSON.

In order to implement the Google and Facebook signup and login, we had to do an extensive amount of research and trial and error testing. The process itself was confusing because the documentation wasn't always helpful. Additionally, we had great difficulty getting the correct permissions to retrieve the user's email from Google and Facebook.

We also had to research how to set up a mail server and a cron job in order to send email notifications to users. We did some research, and it appeared very complicated to set up a mail server, but, thankfully, we came across the Amazon Simple Email Service. This gave us a simplified and free solution to our problem with a good tutorial to help set everything up.

To actually send the emails, we needed a PHP script to run as continuously as possible and check if notifications need to be sent, so we set up a cron job to execute the script every few minutes, and we had to research this process on our own. It was well documented, so it wasn't too difficult. We also make use of a cron job to run the script that updates the users in the TopTen table.

The ability to solve problems through communication was crucial to our success. We would pass along this statement to next semester's teams. If you are not close friends with your group members, you must communicate. We met every Monday of every week barring holidays. At these meetings we would discuss the tasks for the upcoming week, and address any outstanding issues. If a team cannot communicate, it will fail. We would suggest to all future groups that they take communicating and meeting with their group very seriously.

Hindsight is always 20/20, and this project is no exception. If we had the ability to restart this semester, we would ensure more emphasis is placed on the second iteration. We had no trouble jamming the first iteration, but we underestimated the importance of the second

iteration. We were under the impression that the third iteration would be similar in length to the first and second. However, the third was not a true iteration. This forced us to implement a large amount of features very close to the second iteration due date. The quick implementation was not desirable nor recommended.

Wrap Up

PonyPark has intentions of expanding beyond Southern Methodist University. In a 2.0 release, PonyPark would sense where you are currently located (on Android version) and show you nearby garages and their ratings. PonyPark would be available for garages in Downtown Dallas, and eventually expanding to the entire Metroplex.

In the future, we would also implement a few new features. We would implement a "Forgot Password" functionality to allow users to reset their passwords if they forgot them. For the Priority attribute on the FavoriteGarages table, we would allow users to change the priority, so that the favorite garages they value the most would appear at the top of their favorite garages list and at the top of their notifications email. Finally, we would implement functionality to notify users on their phones.

We believe this software would be useful to more than the tech-savvy student. PonyPark believes every American with a smartphone could benefit from user reported data regarding parking availability. We believe expansion is necessary and inevitable.

Data Dictionary

Users:

This table holds the information for each of the users of the website. We made Email and ExternalType a unique key because we want to ensure that a user can have the same email between native/Google/Facebook accounts, yet we also don't want there to be duplicate emails/accounts.

Attributes	Type	Valid Range	Description
UserID	INT	1 to --	The primary key that uniquely identifies each user.
FirstName	VARCHAR(50)	Any combination of letters	The user's first name.
LastName	VARCHAR(50)	Any combination of letters	The user's last name.
Email	VARCHAR(50)	Formatted like a@a.com	The user's email address. Is a unique key with ExternalType.
Password	VARCHAR(64)	NULL or any combination of letters and digits	The hash of the user's password plus the PasswordSalt. The password should have at least 8 characters before the hashing. Is NULL if the user is a Google/Facebook user.
PasswordSalt	VARCHAR(50)	NULL or any string	A randomly generated string. Used to add security to the password. Is NULL if the user is a Google/Facebook user.
PhoneNumber	VARCHAR(20)	NULL or 10 digits	The user's phone number.
UserType	INT	0 or 1	Specifies if a user is a regular user (0) or an admin (1).
ExternalType	VARCHAR(10)	Native, Google, or Facebook	Specifies if a user is logged on natively or through Google or Facebook. Is a unique key with Email.
ExternalID	VARCHAR(64)	NULL or a Facebook or Google ID	Holds the ID provided by Facebook or Google, or NULL if the user is logged on natively.

ParkingLocations:

This table holds the information for each garage on the website.

Attributes	Type	Valid Range	Description
ParkingID	INT	1 to --	The primary key that uniquely identifies each parking location.
Name	VARCHAR(50)	Any combination of letters	The name of the parking location.
Address	VARCHAR(50)	Any combination of letters and digits	The address of the parking location.
Cost	DOUBLE	NULL or \$0.00 to --	The cost of the parking location. If it is free, the cost is \$0.00. It is NULL if unknown.
NumberOfLevels	INT	1 to --	The number of levels a parking location contains.

Ratings:

This table holds all the ratings (and all associated information) that have been made.

Attributes	Type	Valid Range	Description
RatingID	INT	1 to --	The primary key that uniquely identifies each rating.
ParkingID	INT	1 to --	The foreign key that identifies that parking location the rating was made for.
UserID	INT	1 to --	The foreign key that identifies the user the rating was made by.
Timestamp	DATETIME	YYYY-MM-DD hh:mm:ss	The time and date that the rating was made on.
Rating	INT	1 to 5	The rating on the availability of the garage. 1 is full, 2 is scarce, 3 is some, 4 is plenty, and 5 is empty.
Level	INT	1 to --	The level of the parking garage (identified by ParkingID) that the rating is for.

CommuteTimes:

This table holds all the times at which any user would like to be notified. We made UserID, Day, and WarningTime a unique key because we wanted to ensure that no user could add multiple instances of the same time on the same day of the week as a notification time.

Attributes	Type	Valid Range	Description
CommuteID	INT	1 to --	The primary key that uniquely identifies each notification time.
UserID	INT	1 to --	The foreign key that identifies that user the notification time was made for. Is a unique key with Day and WarningTime.
Day	INT	1 to 7	The day of the week that the notification time is for. 1 is Sunday, 2 is Monday, ..., 7 is Saturday. Is a unique key with UserID and WarningTime.
WarningTime	TIME	hh:mm:ss	The time of day that the notification time is for. Is a unique key with UserID and Day.
TimeOfNotification	DATETIME	YYYY-MM-DD hh:mm:ss	Contains the time and date of the most recent notification. Used to ensure that duplicate emails are not sent to a user.

Requests:

This table holds all requests for new garages that have been made, including those that have already been approved or denied.

Attributes	Type	Valid Range	Description
RequestID	INT	1 to --	The primary key that uniquely identifies each notification time.
UserID	INT	1 to --	The foreign key that identifies that user the request was made by.
Name	VARCHAR(50)	Any combination of letters	The name of the parking location requested to be added.
Address	VARCHAR(50)	Any combination of letters and digits	The address of the parking location requested to be added.
Cost	DOUBLE	NULL or \$0.00 to --	The cost of the parking location requested to be added. If it is free, the cost is \$0.00. Is NULL if unknown.
NumberOfLevels	INT	1 to --	The number of levels of the parking location requested to be added.
Comments	VARCHAR(250)	NULL or any combination of letters	Any comments a user has when requesting a garage.
Status	INT	0, 1, or 2	The status of a request is 0 when a request is unreviewed, 1 when it is denied, and 2 when it is approved.

TopTen:

This table holds the top ten users calculated every 24 hours with a script that executes using cron every night at 9 PM.

Attributes	Type	Valid Range	Description
UserID	INT	1 to --	The primary key that uniquely identifies each user in the top ten users.
Points	INT	1 to --	The number of ratings each user on the list of top ten users has made.

FavoriteGarages:

This table holds the favorite garages for all users. We made UserID and ParkingID a unique key because we wanted to ensure that no user would add multiple instances of the same parking location to his or her favorite garages.

Attributes	Type	Valid Range	Description
FavoriteID	INT	1 to --	The primary key that uniquely identifies each favorite garage.
UserID	INT	1 to --	The foreign key that identifies that user who added the favorite garage. Is a unique with ParkingID.
ParkingID	INT	1 to --	The foreign key that identifies that parking location which the favorite garage is. Is a unique key with UserID.
Priority	INT	1 to --	The priority of a user's favorite garages, automatically incremented when favorite garages are added.

Relationships:

Users to TopTen:

The Users table has a 1 to 0..1 relationship with the TopTen table because each user can be 0 to 1 entries in the TopTen table. TopTen is also a weak entity because it is composed of users and cannot exist without the Users table.

Users to Ratings:

The Users table has a 1 to 0..many relationship with the Ratings table because each user can make zero to infinitely many ratings. The Ratings table is a weak entity but is not reliant on the Users table because when we delete a user's account, we do not delete his or her ratings from the database.

Users to CommuteTimes:

The Users table has a 1 to 0..many relationship with the CommuteTimes table because each user can have zero to infinitely many times he or she wishes to be notified. CommuteTimes is a weak entity because a notification time cannot exist without a user.

Users to Requests:

The Users table has a 1 to 0..many relationship with the Requests table because each user can request zero to infinitely many garages. The Requests table is a weak entity because a request cannot exist without a user to submit it.

Users to FavoriteGarages:

The Users table has a 1 to 0..many relationship with the FavoriteGarages table because each user can have zero to infinitely many garages in his or her list of favorite garages. FavoriteGarages is a weak entity because a parking location cannot be a favorite garage without a user.

ParkingLocations to Ratings:

The ParkingLocations table has a 1 to 0..many relationship with the Ratings table because each parking location can have zero to infinitely many ratings. Ratings is a weak entity because a rating cannot exist without an associated parking location.

ParkingLocations to FavoriteGarages:

The ParkingLocations table has a 1 to 0..many relationship with the FavoriteGarages table because each parking location can appear as zero to infinitely many favorite garages, depending on how many users choose that garage as a favorite. FavoriteGarages is a weak entity because a parking location cannot be a favorite garage if there is no parking location.