Milestone One

SW Engineering CSC648/848 Spring 2019
Team 04
02/24/2020

SFStateEats

Team Members

Name	Email	Role
Rachit Joshi	rjoshi@mail.sfsu.edu	Team Lead
Samhita Brigeda	pandu.barigeda@gmail.com	Front End Lead
Pedro Souto	Pedro.Souto.SFSU@gmail.com	Back End Lead
John Pham	JohnPhamDeveloper@hotmail.com	Github Master
Vincent Tran	vtran6@mail.sfsu.edu	Database Lead
Khang Tran	ktran26@mail.sfsu.edu	Frontend Developer

Milestone 1 Version History

Version Name	Date
Milestone 1 Version 2	03/10/2020
Milestone 1 Version 1	02/24/2020

Table of Contents

Section 1: Executive Summary

Currently, San Francisco State does not have an organized and easy method to view food locations on campus. There are many applications that are used to find great restaurants and popular food spots. For example, *Yelp* and *Google Restaurants* are greatly used outside of campus and for personal use. On our campus, there is a lack of information on smaller restaurants and food locations. Some food locations like cafes, restaurants, and food courts are too small to show up on some of the previously mentioned services. **SFStateEats** will be a service that allows its users to rate, view, and discover all food places on San Francisco State University Campus. It will also allow restaurant owners to add their restaurants to the platform.

Fact is, trends catch on. New and hot apps spread between users, and what better place to spread a new service than a close knit school campus? Our service will increase in users via a network effect, as more and more students begin to experience SFStateEats. The reach that this web application can have is enormous, with San Francisco State University being just the tip of the iceberg.

From day one, our application will be free to use. In order to survive in this competitive market, we shall implement ads throughout the application. These ads will have a much higher cost per impression (CPM) due to the fact that they are highly targeted, since the demographics is exclusively San Francisco State University students. Thus companies whose targeted demographic is college students have a great incentive to purchase ads in our application. These ads will allow us to monetize our products, without increasing the barrier to entry for new customers.

We plan to launch our web application in the beginning of summer 2020, where every student of San Francisco State University will be able to join us. When this happens, all features will go live.

*"Our mission at SFStateEats is to make sure **every** student has access to **all** information of **every** restaurant on campus."*

Section 2: Main Use Cases

1. Bob is a Computer Science major student at San Francisco State University, attending his first semester after transferring from a local Community College. He just visited Caffe Rosso on campus for the first time and had an amazing breakfast sandwich. He wished that he knew about this place earlier, and decides to find someplace to leave a review so that others can learn about Caffe Rosso too. Bob learns about SFSTateEats, and opens the website to give the restaurant a 5 star review. The SFStateEats application first asks Bob to login before he can rate a restaurant, but Bob does not have an account. Bob attempts to register for a regular user account, but the username he's trying to register with is already taken. Bob is given a prompt to try a different username instead. Once Bob has finished the registering process, he is given a message stating his registration is successful and is automatically logged in. He then proceeds to write a good review for Cafe Rosso.

Use case 1:   System
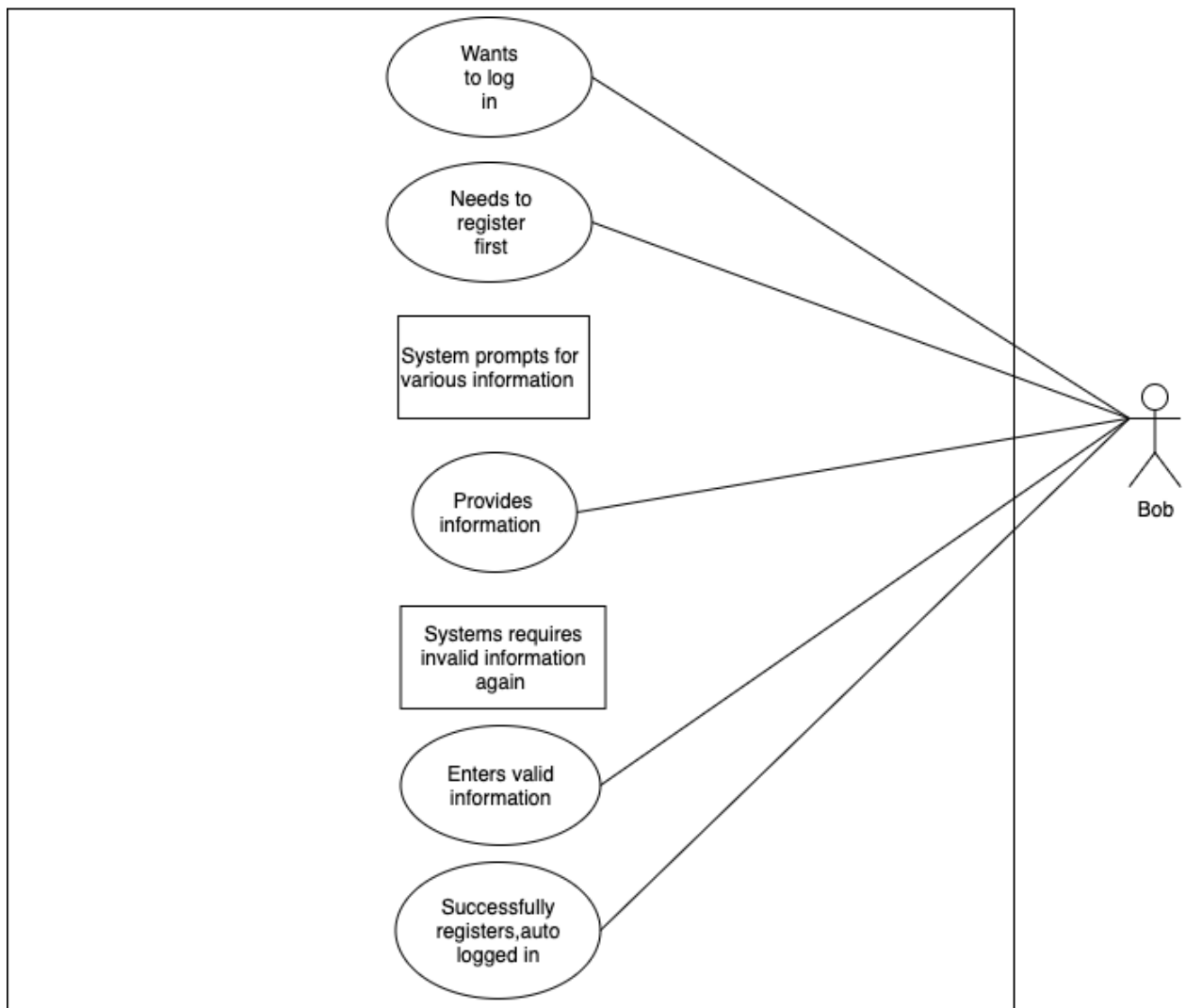
2. Alice is a third year transfer from a Community College in Los Angeles, and this is her first time living alone. She has received financial aid and taken out student loans, but is still struggling to pay all her bills as San Francisco is very expensive. Due to this fact, she has been focusing on spending as little money as possible. She is on a budget and wants to find a cheap place to eat on campus before her next class. She opens the SFStateEats application and searches for every food place on campus. Since she wants the cheapest options, she sorts the list by price, then looks for a location with good reviews. She sees that Nizario's Pizza is located in the basement of the Cesar Chavez Student Center building. She also sees that this food place has 4.5 / 5 stars, and decides to try it out.
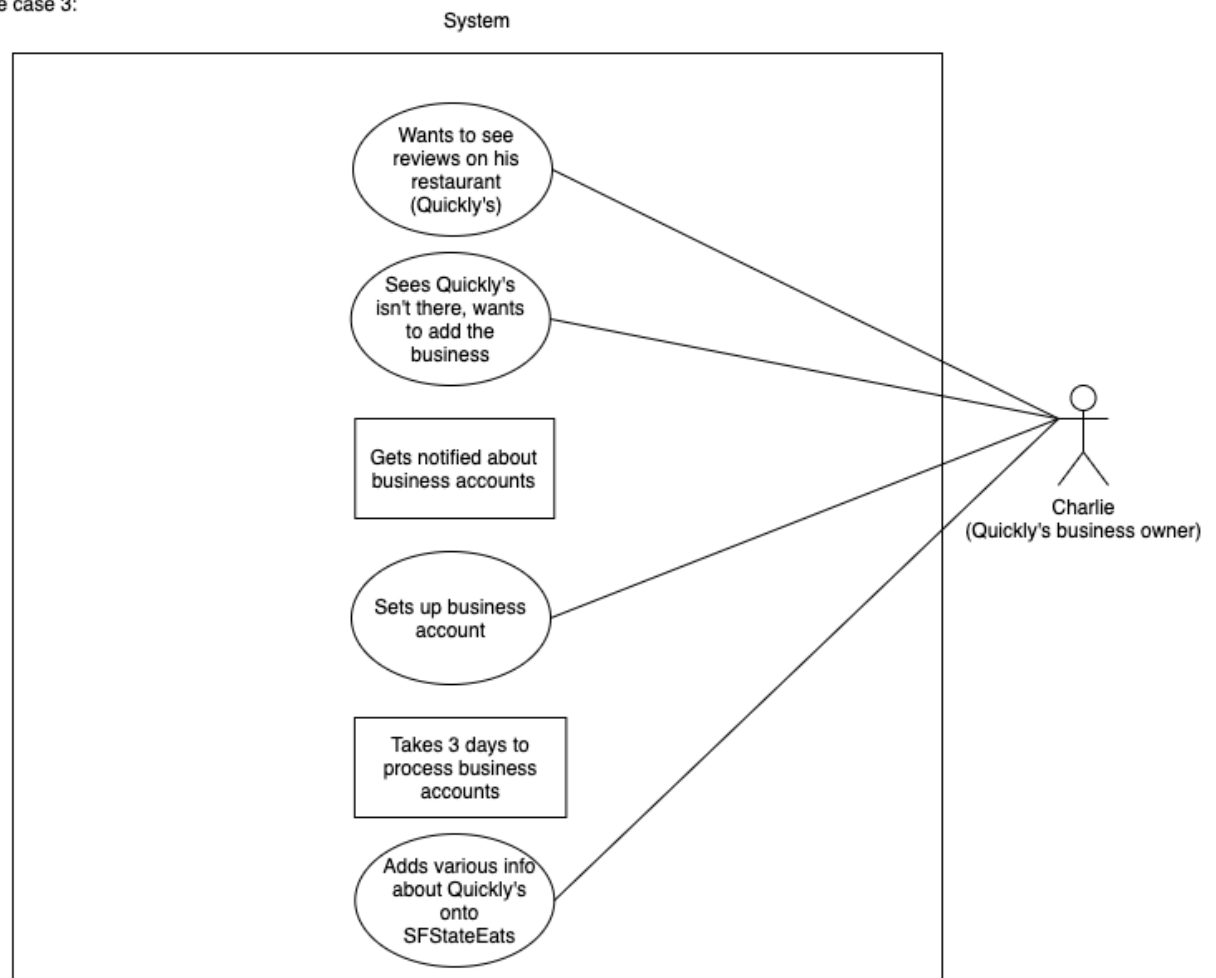

Use case 2:
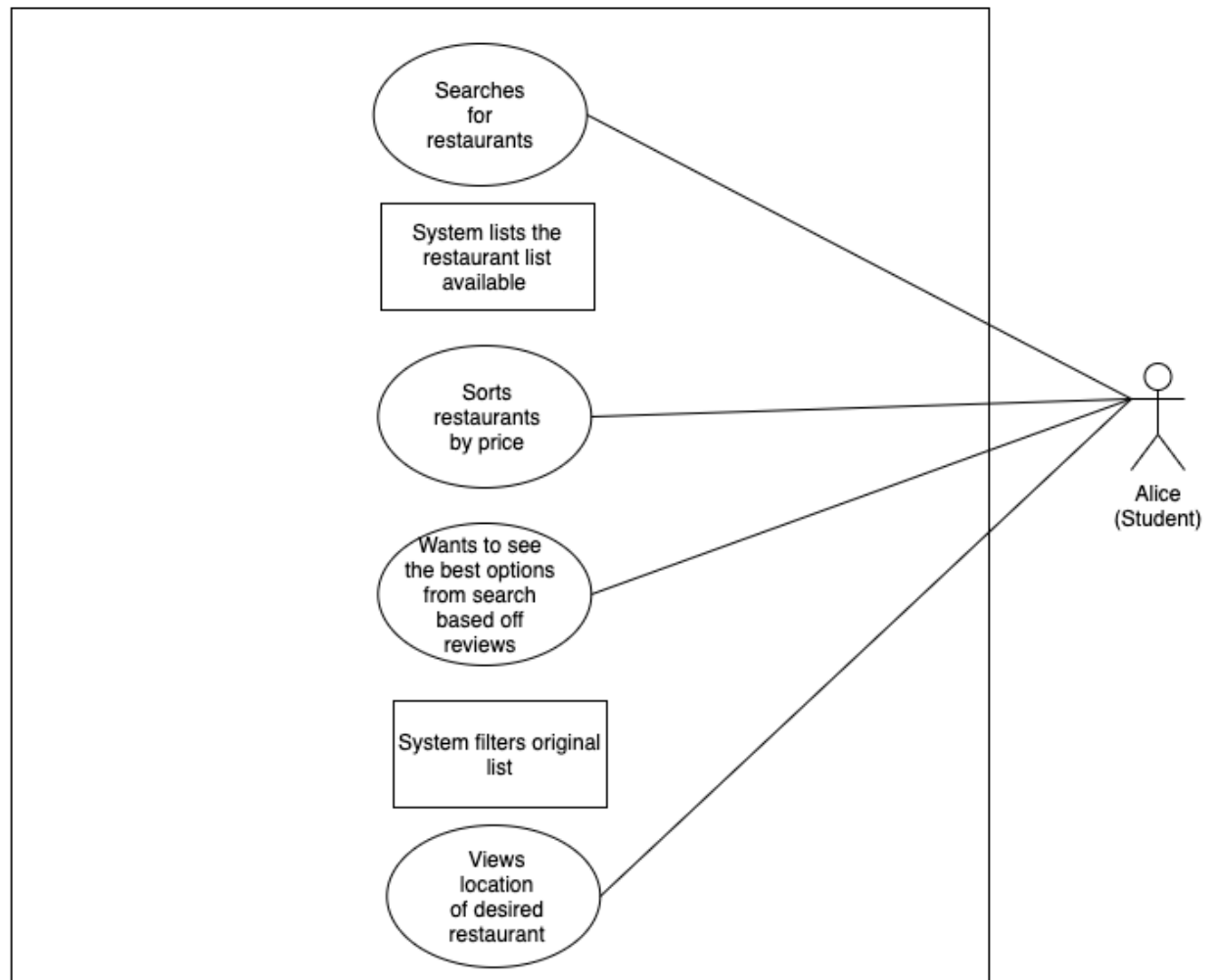
3. Charlie is not a student at San Francisco State University, but he is there almost every day because he is the owner of a food place called Quickly's on campus. He has owned this small restaurant for 8 years, and knows many of the customers personally. A few of them that he knows particularly well said they wanted to leave a great review on this new application, SFStateEats. Charlie is curious about the application. He looks up Quickly's on SFStateEats, and realizes this restaurant is not listed on the web app. He tries to add his restaurant onto the platform, but is notified that he needs a business account in order to add a new place to the platform. He starts the process to create his business account, and after 3 days, it is verified. With his business account, he adds Quickly's to the list of campus restaurants, along with the menu, hours, and prices.

Use case 3:

System

Wants to see reviews on his restaurant (Quickly's)

Sees Quickly's isn't there, wants to add the business

Gets notified about business accounts

Sets up business account

Takes 3 days to process business accounts

Adds various info about Quickly's onto SFStateEats
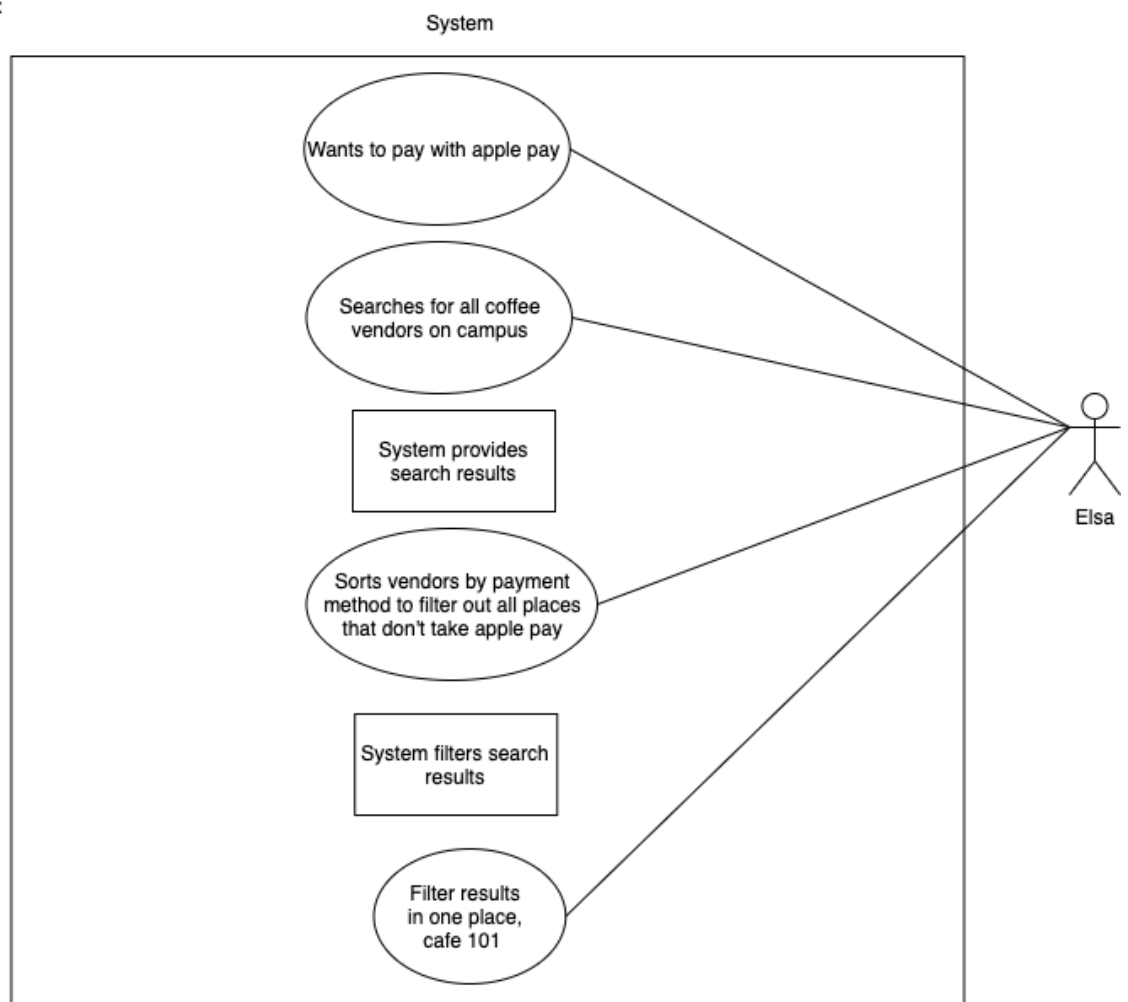
Charlie
(Quickly's business owner)

4. Dan is a new graduate student at San Francisco State University and loves trying out new food. He attended San Jose State University for 4 years prior, attaining his Bachelor's Degree in physics. While going to school at San Jose State University, he learned all the local food places, but now that he is a new student on a new campus, he feels lost on where to go. After asking some classmates, he was informed that most people on campus use SFStateEats to scout out new places, so he decides to visit the website. He is looking for a good burger for lunch today and wants to see all the burgers on campus. He opens the list of all restaurants, and pulls up the menu for each of them. He then filters out all the menus that do not have a burger as an item on it, and is left with a handful of menus that do. He compares the prices and description for all the burgers, and decides to go with the burger on the third menu. He is able to see that it belongs to a place called Burger Express, located on the first floor of the Ceasar Chavez building.
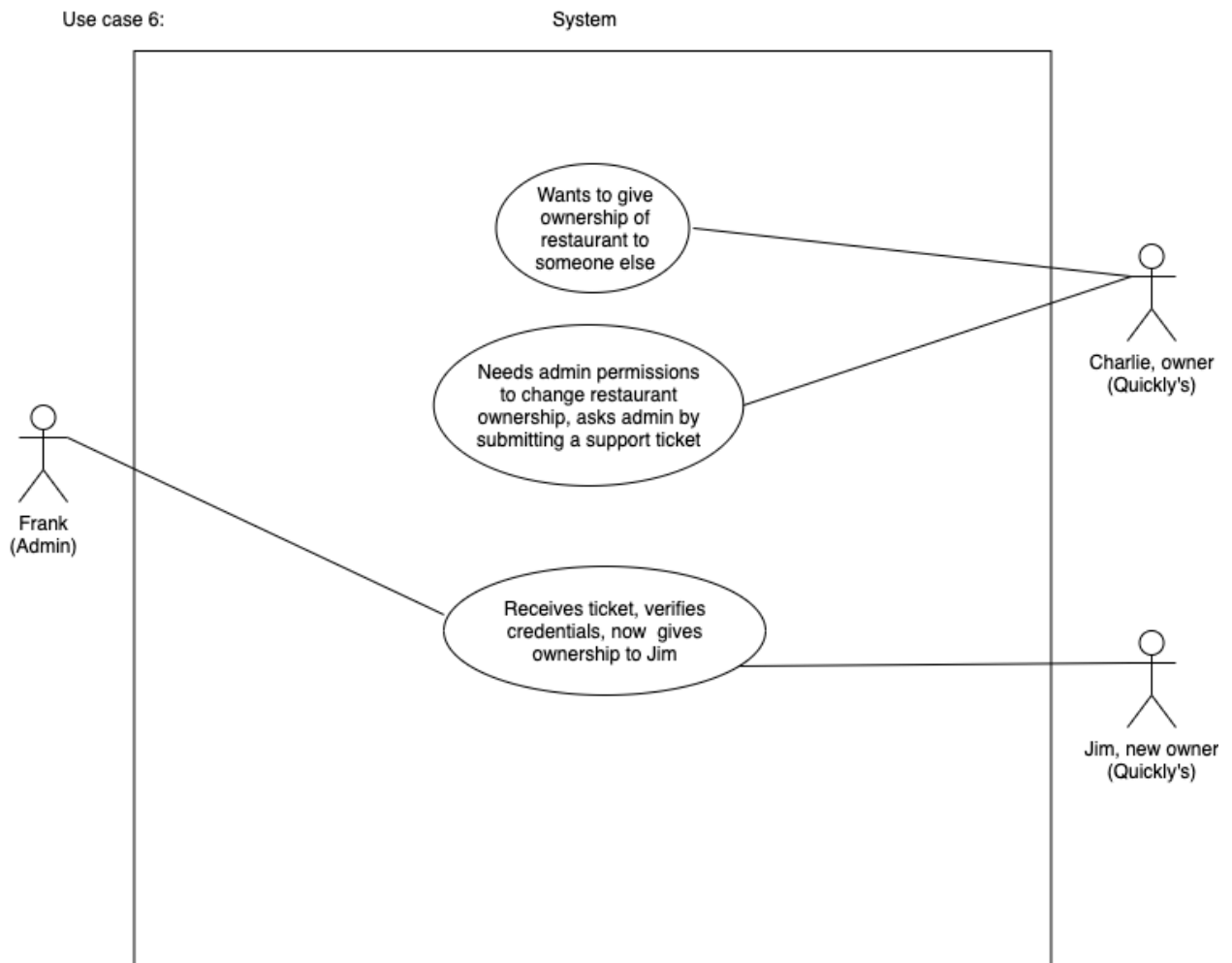
Use case 4:
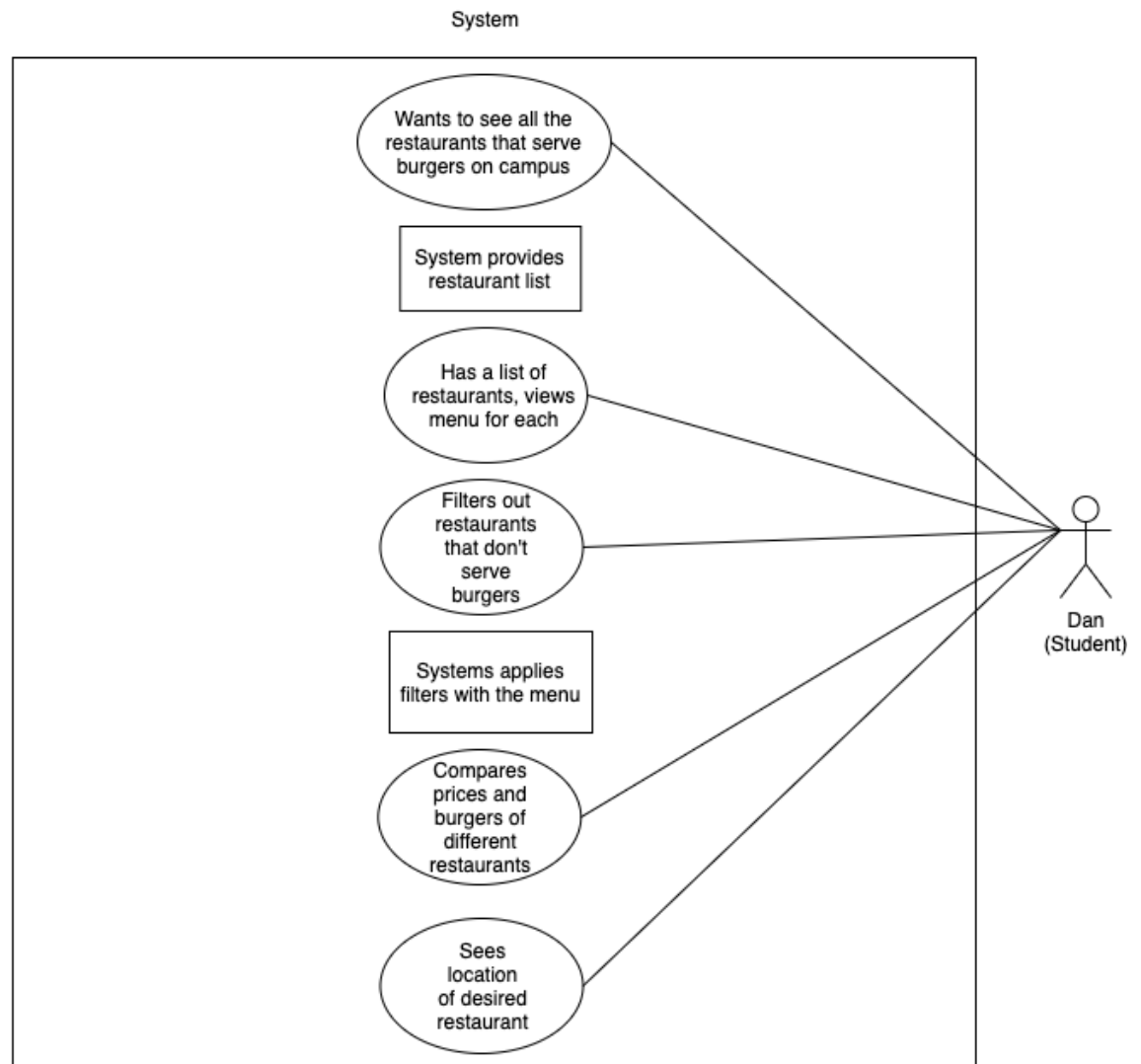
5. Elsa is a third year student attending San Francisco State University pursuing a degree in Sociology. Every semester she does great in her classes, and tries to offer as much support to her peers as possible. This semester, she has Tuesday and Thursday free, and leads a study session for Sociology 304 every Tuesday at 9:00 AM. In her rush to get out of the door in the morning, she forgot her wallet at home, and only has her iPhone with her. She needs a coffee to get her day started, but does not have any cash or credit cards with her. She opens SFStateEats, and pulls up all the vendors that have coffee. She then sorts it by payment method to filter out all the places that do not accept ApplePay. Luckly, one place did, so she decides to go to Cafe 101.

Use Case 5:

System

Wants to pay with apple pay

Searches for all coffee vendors on campus

System provides search results

Sorts vendors by payment method to filter out all places that don't take apple pay

System filters search results

Filter results in one place, cafe 101

Elsa

6. Frank is a senior at San Francisco State University, and has run many clubs in the past. This year he is the leader of 'Google Codes', a club for Computer Science majors sponsored by Google. Due to his experience working with people and his proven track record, the creators of SFStateEats offered him a position as an admin for the website, which Frank accepted. Frank has been an admin on the SFStateEats app for 3 weeks, and has been excelling so far. When users encounter a problem on the app, they can open a support ticket to receive help. Charlie, who is the owner of Quickly's is selling his restaurant to move to another state. He needs to change the owner on the Quickly's account to someone else, so he opens a support ticket to have an admin do it. Frank receives a support request, and after verifying all the proper paperwork, Frank manually changes the owner of Quickly's to the new owner, Jim.

Use case 6:   System

Wants to give ownership of restaurant to someone else

Needs admin permissions to change restaurant ownership, asks admin by submitting a support ticket

Charlie, owner (Quickly's)

Frank (Admin)

Receives ticket, verifies credentials, now gives ownership to Jim

Jim, new owner (Quickly's)

7. Sawara has been a student at San Francisco State University for 2 years. When out riding her bike, she fell and broke her leg, and now has to use crutches for the next 2 months. Normally when she has time between her classes, she walks around exploring food places on campus. Now that her leg is broken she has limited mobility, but still wants to explore and try out new food places. She learns about an app called SFStateEats that lists all the restaurants on campus, along with a host of useful information, and thinks this might be the perfect solution to her problem. She downloads the app and registers for a new account. After registering, she logs into the app and starts exploring the profile pages of all the restaurants that she is interested in. Then she starts seeing the locations on the map to see which is closest to her. Based on her search and interests, she choses to get an item at Cafe Russo.

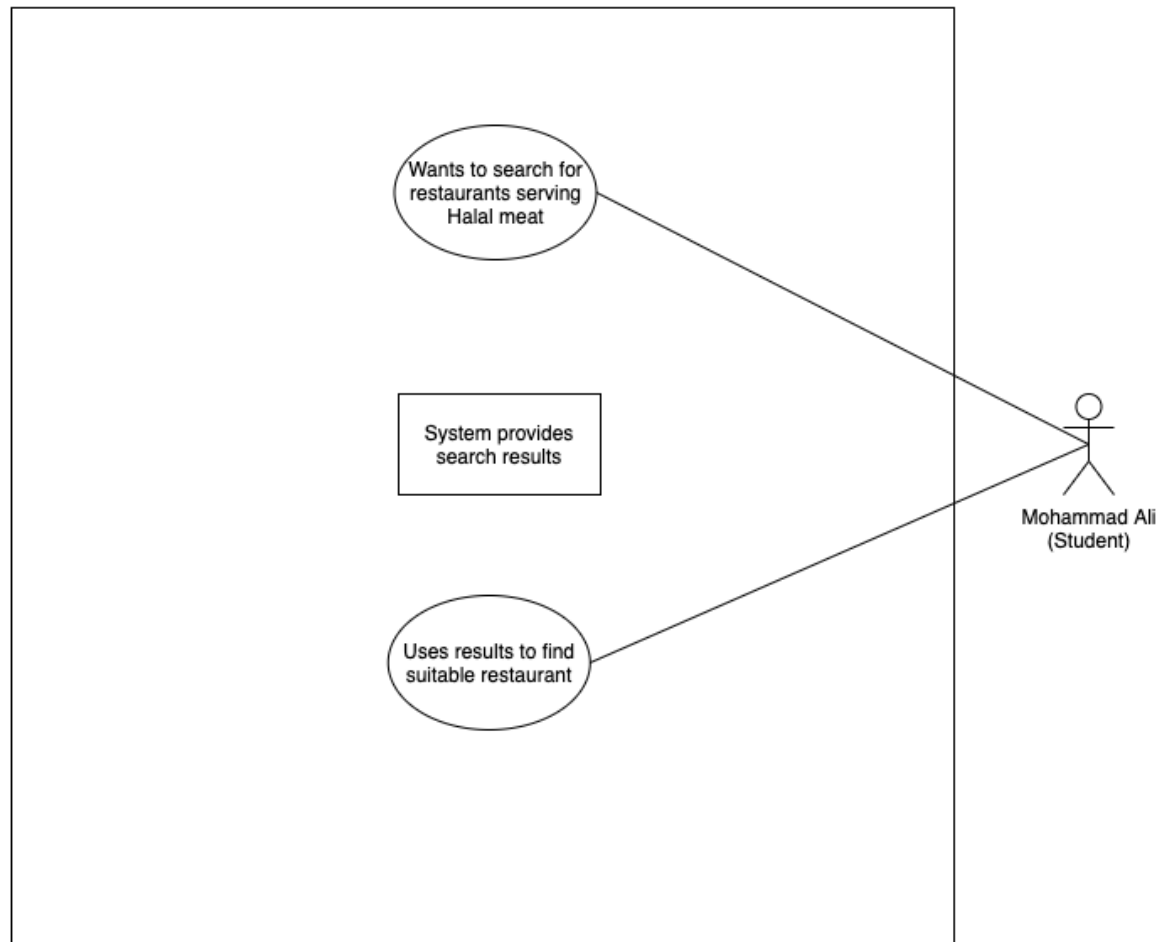Use case 7:   System
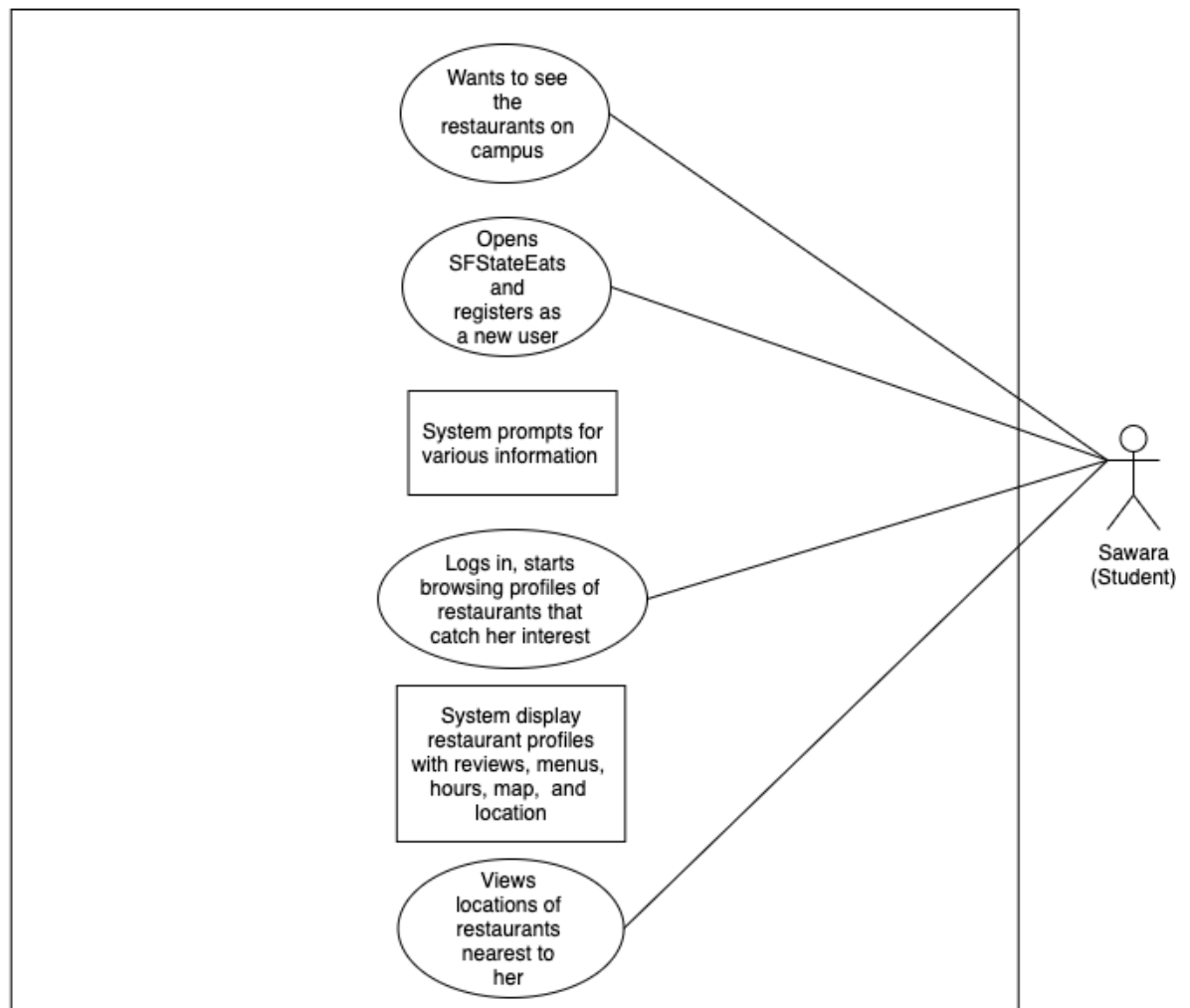
8. Mohammad Ali is an international student at San Francisco State University. Back in his home country, United Arab Emirates, it is easy to find Halal food since many adhere to Islamic Law. After transferring to schools, he has struggled to find Halal on campus, and looks for a new way to find it. After doing some Googling, he comes across a website called SFStateEats, which is exactly what he was looking for. Mohammad logs into SFStateEats and uses halal meat in the search options. After skimming through the list of restaurants serving halal meat, he sees a place called Crave that serves sub sandwiches, and decides to try it out. After eating there, he was glad it was Halal, but thought the quality was not amazing. He leaves a 3 star rating along with his review explaining his thoughts.

Use case 8:   System

Wants to search for restaurants serving Halal meat

System provides search results

Uses results to find suitable restaurant

Mohammad Ali (Student)

Section 3: List of Main Data Items and Entities

Entity or Data Item	Description
System Administrator	Account for those users who are employed or trusted by the application owners. These accounts have access to moderator tools that no other account has access to. This may include the ability to remove reviews or change business information.
Unregistered User	A user who visits the website to use its features, but does not create an account.
Registered User	A user who uses the website and has created an account. This allows them to access features that an unregistered user may not have.
Registered Business Owner	A user who owns a business that is listed on SFStateEats. This account has access to features relating to business pages that other users do not.
Restaurant	An entity that contains all the information about a restaurant. This will be the page that has reviews, menus, hours, map, and location.
Review	A data item relating to restaurants. Reviews can be left by registered users and can be seen by any user. Reviews may include a section of text, photos, and ratings.
Menu	A data item relating to restaurants. A menu lists all of the items that the restaurant servers. This may include prices along with other information.
Map/Location/Directions	A data item relating to restaurants. A location will be information regarding where the restaurant can be found.

Section 4: Initial List of Functional Requirements

User

1. New users shall be able to an account.
2. Users shall be able to login into their account.
3. Users shall be able to rate restaurants on campus.
4. Users shall be able to rate food places on campus.
5. Users shall be able to write and view reviews.
6. Users shall be able to reply to reviews.
7. Users should be able to post pictures in their reviews.
8. Users shall be able to see the menu for each restaurant.
9. Users shall be able to see the price for each item on the menu.
10. Users shall be able to see the location of the restaurant on the profile page.
11. Users shall be able to see all the business that are involved in the farmers market.
12. Users shall be able to see the map locations of each restaurant on campus.
13. Users shall be able to add photos of each item on the website.
14. Users shall be able to check-in to restaurants.
15. Users shall be able to see busy times for restaurants.
16. Users shall be able to see hours of business for each restaurant.
17. Users shall be able to apply filters to their search.
18. Users shall be able to flag a review as inappropriate for the application.
19. Users shall be able to sort reviews by date.
20. Users shall be able to sort reviews by popularity.
21. Users shall be able to delete a review they have written.
22. Users shall be able to edit a review they have written.
23. Users shall be able to see all reviews they have written.
24. Users shall be able to see all reviews another user has written.
25. Users shall be able to reply to other user's reviews.
26. Users shall be able to share restaurant information and profiles to each other.
27. Users will be able to connect their facebook and instagram to their accounts.
28. Users will be able to login with their sfsu id that is linked to ilearn.
29. Users will be able to see a visual representation of the ratings.
30. Users who are visually impaired will be able to use an automated system (such as siri) to navigate through the app.
31. Users will be able to see a map of all the restaurants and cafes on the college campus.

Business

32. New business owners shall be able to register a business account.
33. Business owners shall be able to login into their account.
34. Business owners shall be able to add their restaurant to the website.

35. Business owners shall be able to create a menu for their restaurant.
36. Business owners shall be able to upload photos of their restaurant.
37. Business owners shall be able to tag their restaurant into categories.
38. Business owners shall be able to write descriptions of their restaurant on the profile page.
39. Business owners shall be able to upload an approved health score.
40. Business owners shall be able to change prices on the menu.
41. Business owners shall be able to change hours of the restaurant.
42. Business owners shall be able to petition for a change of owner.
43. Business owners shall be able to add tags to the restaurant.
44. Business owners shall be able to flag a review as inappropriate for the website.
45. Business owners shall be able to remove their restaurant from the website.
46. Business owners shall be able to see a chart of how their restaurant is doing based on the ratings.

Restaurant

47. The restaurant profile page shall display photos.
48. The restaurant profile page shall display ratings.
49. The restaurant profile page shall display tags.
50. The restaurant profile page shall display a description.
51. The restaurant profile page shall display location.
52. The restaurant profile page shall display a menu.
53. The restaurant profile page shall display hours.
54. The restaurant profile page shall display reviews.
55. The restaurant profile page shall display the owner.
56. The restaurant profile page shall display phone number.
57. The restaurant profile page shall display the option to order online.
58. The restaurant profile page shall display an option to pick up the order.
59. The restaurant profile page will show any nutritional facts of their restaurant
60. The restaurant profile page shall show the customers if they serve halal, vegan, vegetarian, and non-vegetarian meals.
61. The restaurant profile page shall display the various payment methods the restaurant accepts.
62. The restaurant profile page shall display the popularity/traffic of the restaurant.

System Administrator

63. System Administrators shall have the privilege to ban any users from the website for misuse.
64. System Administrators shall have the privilege to delete restaurants from the platform.
65. System Administrators shall have the privilege to add restaurants to the platform.
66. System Administrators shall have the privilege to change restaurant information.
67. System Administrators shall have the privilege to verify new business accounts.

68. System Administrators shall have the privilege to delete reviews from the platform.
69. System Administrators shall be able to cancel orders placed at the customer's request.
70. System Administrators shall be able to refund the customer if the order is not what they wanted.

System

71. The system shall show trending restaurants on the main page.
72. The system shall show recent reviews on the main page.
73. The system shall show new restaurants on the main page.

Section 5: List of Non-Functional Requirements

Security

1. Login shall be required to leave reviews.
2. Login shall be required to create a business listing.
3. A gateway service will redirect traffic to other services to prevent the discovery of the end URL.
4. Email shall not be from a temporary email host provider.
5. Password shall contain at least 1 number, 1 special character, 1 uppercase letter, and 1 lowercase letter.
6. Users shall be required to change password every 12 months.
7. Change of password shall require email verification.
8. Users shall have 5 login attempts before account lockout.
9. Account unlock shall require email verification before unlocking.

Audit

10. New business listing shall be approved by the system administrator.
11. Change in business owner shall be approved by the system administrator.
12. Flagged reviews shall be reviewed by system administrator.
13. Inappropriate reviews shall be removed by the system administrator.

Performance

14. Each service should be hosted in its own server to prevent overwhelming one server.
15. The web application should restart if it is abruptly shut down to prevent downtime.
16. System shall respond visually within 5 seconds.
17. Animations shall be kept to a minimum to accommodate for slow performing devices.

Data Integrity

18. Images shall only be in the format of jpg, jpeg, and png.
19. Images shall be saved on the server.
20. Images uploaded shall be at most 2mb.
21. Images shall be saved in the original size.
22. Images shall be resized and displayed via CSS formatting, thus avoiding displaying and resizing after.
23. Reviews shall not consist of special characters or emojis.
24. Display name shall not consist of special characters or emojis.
25. Email shall not consist of emojis.
26. Password shall not consist of emojis.

27. Databases shall be backed up every 24 hours.
28. Databases shall be able to be backed up on command by a system administrator.

Compatibility

29. The site shall be compatible for all mobile screen sizes.
30. The site shall be compatible for all monitor screen sizes.
31. The site shall be able to scale to high resolution (2K, 4K).
32. The site shall be compatible for Safari on version 12.1.2.
33. The site shall be compatible for Firefox on version 73.0.
34. The site shall be compatible for Chrome on version 80.0.3987.106.

Conformance with Coding Standards

35. The whole production cycle of the site shall be finished in at least 5 days before the delivery date.
36. Development console logging shall be disabled for production.
37. The DOM tree shall have meaningful semantic for every element.
38. Components shall be created to allow for reusable code.

Look and Feel Standards

39. The site shall have an interface that is intuitive to navigate at first glance.
40. The site shall use clean fonts and colors.
41. The site shall meet modern design standards.

Internalization / Localization Requirements

42. The site shall be internationalized in the English language

Web Site Policies

43. The site shall not allow illegal content.
44. The site shall not allow harassment of other users.
45. The site shall not allow nudity.
46. The site shall not allow business owners to review their own site.
47. The site shall not allow self-promotion or malicious links.
48. Reviews and replies are subject to the website's community guidelines.

Section 6: Competitive Analysis

Feature/Company	Yelp	OpenTable	Google Reviews
Strengths	Sleek and intuitive UI	Unintrusive UI	More accurate restaurant information
Weaknesses	Restaurant information not always up to date	Commenting Section not fully fledged out	UI is not easy to navigate
Customer Support	Responsive customer support	Responsive customer support	Does not offer customer support
Featured Businesses	All businesses	Food businesses only	All businesses
Brand Recognition	Highly popular. Mildly accessible.	Less popular. Mildly accessible.	Mildly popular. Highly accessible
Onboard Experience	Fast and seamless browsing.	Small review base, but organized UI.	Large review base, but unorganized reviews.

Feature	Yelp	OpenTable	Google Reviews	SFStateEats
Reviews	+	+	+	++[1]
Reservation	+	+	-	-
Online Order	+	-	-	-
Campus Event Announcements	-	-	-	+[2]
Location and Hours	+	+	+	+

1: While other competitors also have reviews, none of them have threads within those reviews. SFStateEats shall have the ability to leave comments and start a thread on existing reviews. All reviews a restaurant has will show up in a list that users can browse. If a user thinks a review is good, bad, missing something, etc…, he is able to reply to this review. This will start a self-contained thread, that anyone can comment to add to. As more users comment, the thread gets bigger, but these chains of comments will never crowd the original reviews tab of the restaurant. This is a benefit because people can discuss the topics mentioned in a specific review, as opposed to just the restaurant as a whole. It allows for more user engagement, without cluttering the experience for other users.

2: Yelp, OpenTable, and Google Reviews all lack the ability to add food related events that are occuring on campus. One example of such an event is when a conference occurs, they often give out free food. SFStateEats shall allow organizers of these events to announce their events. This allows users to be on the lookout for food events simply by using the application. When a new event is added by a host, all users of SFStateEats who are subscribed and meet the criteria will be notified of this event. This is also great for hosts, because they are able to attract more people to their events.

Section 7: High Level System Architecture and Technologies Used

Server Host: Amazon EC2 server vCPUs 1 RAM 1GiB

Operating System: Ubuntu 18.04

Database: PostgreSQL v10.12

Web Server: EC2 t2.micro

Server-Side Language: JavaScript

Web Framework: ReactJS

IDE: IntelliJ, WebStorm, VS Code

HTML: 5

CSS: 3

React: 0.61

Node: 13.8.0

Supported browsers: Chrome(ver 80.0.3987.106), Safari(ver 12.1.2) and Firefox(ver 73.0) .

Section 8: Team

Name	Email	Role
Rachit Joshi	rjoshi@mail.sfsu.edu	Team Lead
Samhita Brigeda	pandu.barigeda@gmail.com	Front End Lead
Pedro Souto	Pedro.Souto.SFSU@gmail.com	Back End Lead
John Pham	JohnPhamDeveloper@hotmail.com	Github Master
Vincent Tran	vtran6@mail.sfsu.edu	Database Lead
Khang Tran	ktran26@mail.sfsu.edu	Frontend Developer

Section 9: Checklist

Task	Status
Team found a time slot to meet outside of the class	DONE
Github master chosen	DONE
Team decided and agreed together on using the listed SW tools and deployment server	DONE
Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing	ON TRACK
Team lead ensured that all team members read the final M1 and agree/ understand it before submission	DONE
Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)	DONE