

GatorTraders

Milestone 2

Team 7

Javier Kirksey, Team Lead | Javier.kirksey21@gmail.com

Evan Terry, Technical Lead | eterry@mail.sfsu.edu

Chohee Kim | ckim4@mail.sfsu.edu

Kendrick Kwok | kkwok@mail.sfsu.edu

Brandon Chiong | bchiong@mail.sfsu.edu

Marc Panlilio | mpanlili@mail.sfsu.edu

Ramsay Wong | ramsayw@mail.sfsu.edu

Date Last Modified: 03/14/2017

1. Use Cases

Use Case: Seller, Students

Steve is a fourth year San Francisco State University student who will be graduating once this current spring semester ends. He has stored up all of his textbooks from all four years but since he will be graduating soon, he figured there is not a need for all these textbooks anymore. During one of his classes, he overheard one of his classmate talk about a website that is similar to websites like Craigslist but only for SFSU students. This website allows SFSU students to browse for whatever items they desire and if something catches their eyes, they can buy it then and there. On the other hand, if a student wants to sell an item(s) the website provides that function as well. The website is only provided for students who are SFSU students so to buy or sell anything, one must need their SFSU ID number to create an account to use all the functions of the website. Steve wants to help others by selling his textbooks for a cheaper price since he experienced firsthand how pricey buying textbooks can get. He goes onto the site and decides to post, but is prompted to register or log in. Steve goes through the account creation and verification process, and is approved. He ends up posting up his textbooks and they were all sold by the time next school semester starts when he has officially become a SFSU alumni.

Use Case: Buyer, Students

Max is an incoming first year student who will be attending San Francisco State University when fall comes. Since Max is a first year student, he decides to live close to campus by living in one of the dorms provided by the University so that he would not have to commute back to his hometown and so that he would not have to be late to his classes and extracurricular activities by using public transportation. Max needed to buy some items because the housing department did not provide enough of the items he has been using at home. His brother, who is now a San Francisco State Alumni, mentioned that he used this website provided only to SFSU students to sell his textbooks to other SFSU students. Max went onto the website and started browsing what it had to offer. Max wanted to narrow down his search so he searched up specific keywords to reduce the number of results of items in the list. He ended up searching for a lamp, a coffee table, a coffee maker machine, a bed comfortable, a few textbooks, and a calculator. Once his search has been completed, he proceeded to press the message button. Since he does not have a verified account, he will be prompted to log in, but because it is his first time he will press the create account

button. From there he will have to verify his SFSU email, and once finished he will be able to buy the items.

Use Case: Buyer, Guest

Sarah heard of this website similar to Craigslist that caters to SFSU students. She is looking to buy a couple textbooks for her classes. She goes onto the website and looks through the item listings. She sees the **item name, price, and descriptions** that people post. Sarah is interested in buying an item, but she is prompted to create an account or login after clicking the Messages button. She sees that there is a warning message that says "You must have an SFSU email to register." Unfortunately she isn't an SFSU student and is not able to create an account.

Use Case: Administrator

Allan is an administrator of the website. As he is browsing the listings placed on the site, he stumbles upon a vulgar listing. Allan decides that it is inappropriate and decides to take down the listing from the website. He first informs the student seller that his listing is inappropriate then removes it from the site.

2. Data Definitions

1. Administrator (Admin)

Overlooks the website and changes the content that is appropriate. Makes note of bugs and brings forth the issue to fix.

2. Guests

Can access posted items, but can't see contact information.

3. Items

Products sold from sellers that are usable for users to browse and purchase.

Item contains pictures and descriptions of what is to be sold by other users.

- a. Item Name
- b. Description
- c. Price
- d. Picture (Optional)

4. Students

Can access all information of posted items and can post items.

- a. Buyer
- b. Seller

5. Blockee

Can't access student in which blocked them or content listed by that student.

3. Functional Requirements

Priority 1

All Guests

1. All guests shall be able to visit, browse, and search the site.
2. All guests shall be able to create an account and login.
3. All guests shall provide email address and password to create an account.
4. Guests transition to students impose email restriction based on SFSU email address.

Students

1. Students shall be able to list their products for sale.
2. Students shall be able to reply to a posted item for sale.
3. Students shall be able to message other students.

Administrator

1. Admin shall be able to visit, browse, and search the site.
2. Admin shall be able to remove posts.
3. Admin shall be able to delete accounts.

Priority 2

All Guests

1. All guest shall be able to flag posts.

Students

2. Students shall be able to save listing to their favorite.
3. Students shall be able to block other students.

Blocked Students

1. Blocked students shall not able to access listings by blockee.

Priority 3

Students

1. Students shall be able rate interaction of other students.
2. Students shall be able to see rating of other students.
3. Students shall be able to see history of sold items.
4. Students shall be able to create a biography for their profile.

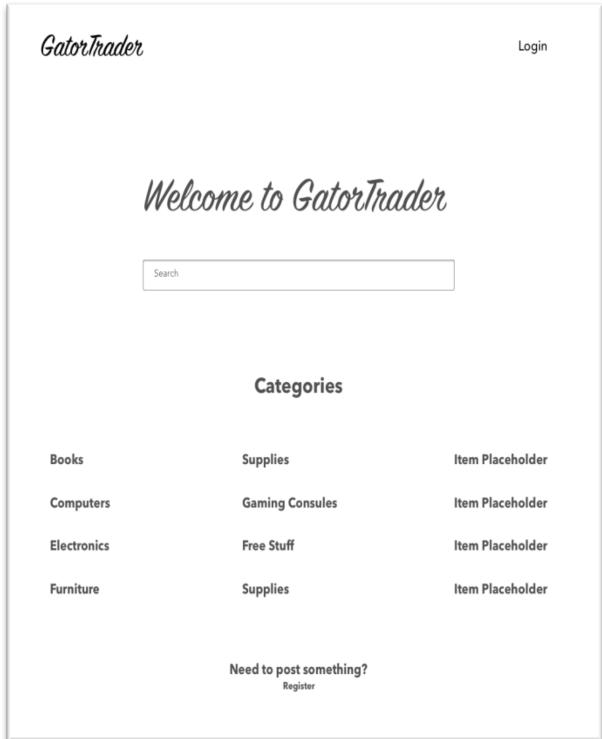
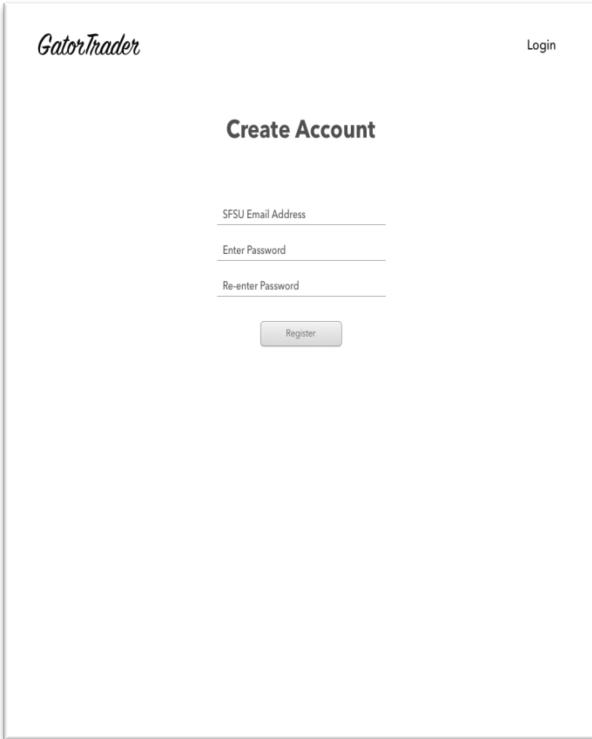
4. Non-functional specs

1. Application shall be developed using class provided LAMP stack
2. Application shall be developed using pre-approved set of SW development and collaborative tools provided in the class. Any other tools or frameworks must be explicitly approved by Anthony Souza on a case by case basis.
3. Application shall be hosted and deployed on Amazon Web Services as specified in the class
4. Application shall be optimized for standard desktop/laptop browsers, and must render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome and Internet Explorer.
5. Application shall have responsive UI code so it can be adequately rendered on mobile devices but no mobile native app is to be developed
6. Data shall be stored in the MySQL database on the class server in the team's account
7. Application shall be served from the team's account
8. No more than 50 concurrent users shall be accessing the application at any time
9. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
10. The language used shall be English.
11. Application shall be very easy to use and intuitive. No prior training shall be required to use the website.
12. Google analytics shall be added
13. Messaging between users shall be done only by class approved methods to avoid issues of security with e-mail services.
14. Pay functionality (how to pay for goods and services) shall not be implemented.
15. Site security: basic best practices shall be applied (as covered in the class)
16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
17. The website shall prominently display the following text on all pages "*SFSU Software Engineering Project, Spring 2017. For Demonstration Only*". (Important so as to not confuse this with a real application).

5. UI Mockups and Storyboards (high level only)

Storyboard 1 - Seller, Student

1.1 - Steve goes into GatorTraders website. He is currently at the Welcome page. Steve wants to post a listing but sees that he needs to register. He presses the Login button on the top right corner of the website, and is taken to the create account page. From the create account page, he is able to put in his credentials to register.

Welcome Page	Create Account Page
 <p>The Welcome Page features the GatorTrader logo in the top left. In the top right, there is a "Login" link. Below the logo, the text "Welcome to GatorTrader" is displayed in a large, italicized font. A search bar with the placeholder "Search" is centered below the welcome message. Underneath the search bar is a section titled "Categories" containing four items: Books, Supplies, Item Placeholder, Computers, Gaming Conssoles, Item Placeholder, Electronics, Free Stuff, Item Placeholder, and Furniture, Supplies, Item Placeholder. At the bottom of the page, there is a message "Need to post something?" followed by a "Register" link.</p>	 <p>The Create Account Page features the GatorTrader logo in the top left. In the top right, there is a "Login" link. The main title "Create Account" is centered above three input fields: "SFSU Email Address", "Enter Password", and "Re-enter Password". Below these fields is a "Register" button.</p>

1.2 - After the registration has been a success, Steve is taken to the Student Welcome page. He clicks the 'Post Listing' button and he is brought to the post a listing page. Steve is trying to sell all his books before the start of the semester, so he puts in Books as the title and the descriptions for those items. He presses 'Add Listing' to create the post.

Student Account Page

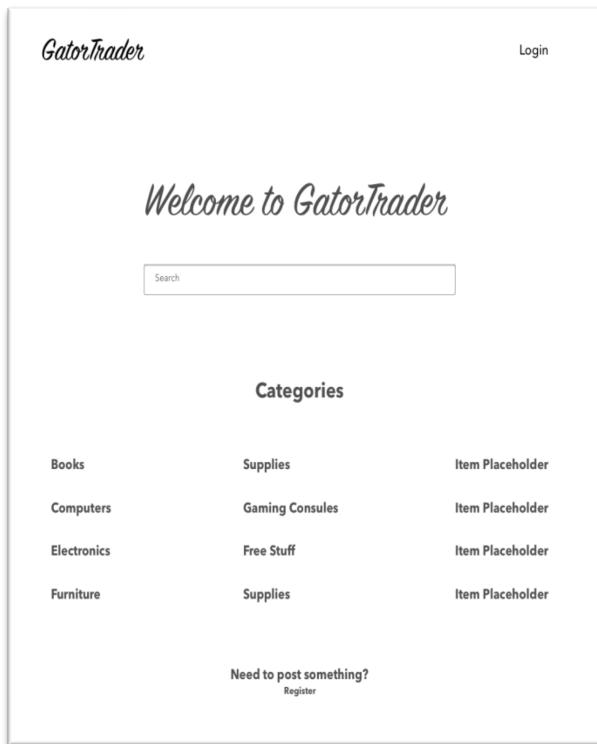
Post Listing Page

The image contains two wireframe diagrams side-by-side. The left wireframe, labeled 'Student Account Page', features a header with 'GatorTrader' and 'Student'. Below the header is a large 'Welcome Student' message. A search bar labeled 'Search' is positioned above two buttons: 'Messages' and 'Post Listing'. The right wireframe, labeled 'Post Listing Page', also has a 'GatorTrader' header and 'Student' status. It displays a 'Post a listing' heading, followed by input fields for 'Title' and 'Price', and a larger 'Text area content' field. At the bottom are two buttons: 'Post Listing' and 'Messages'.

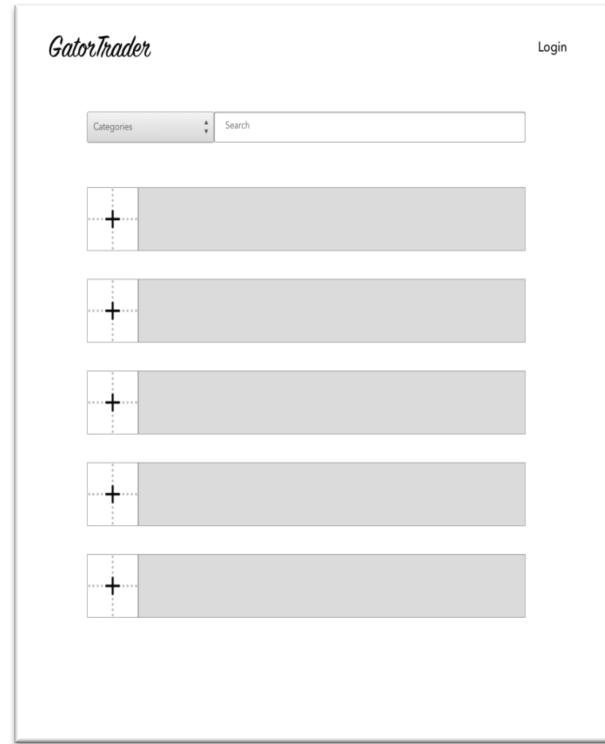
Storyboard 2 - Buyer, Student

2.1- Max needs to buy some home supplies for his new dorm on campus. He goes to Gator Traders and is taken to the Welcome Page. He locates the search bar located in the middle of the Welcome Page, and searches for furniture in the search bar. Max is taken to the Item Results page. He can then browse the listings and see what he wants.

Welcome Page

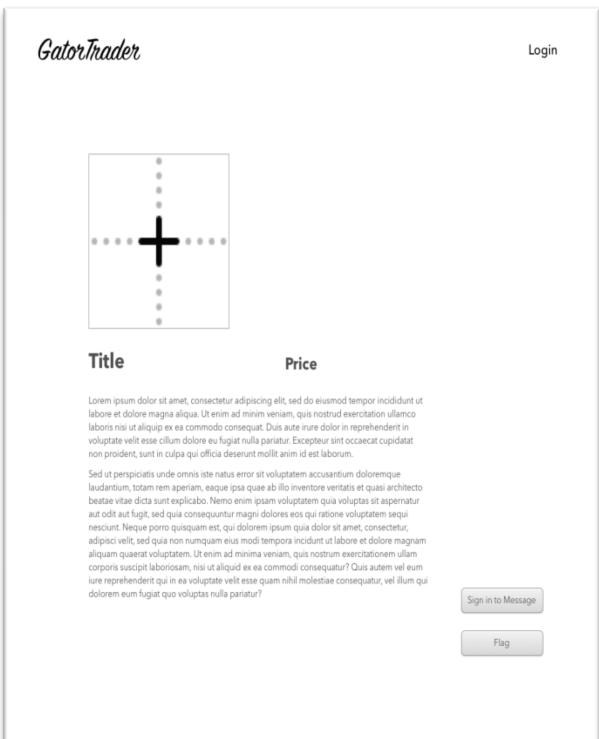


Item Results Page



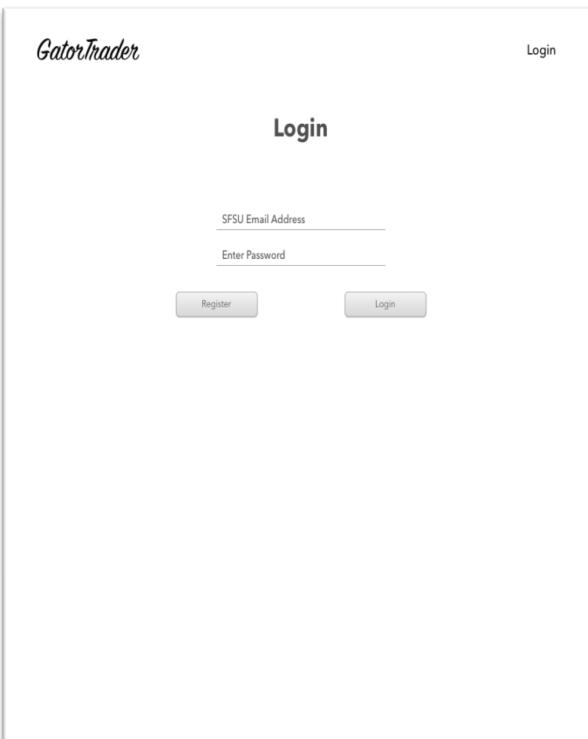
2.2 - He locates the item that he wants. He clicks on the item and Max is brought to the Guest Item Page. Max presses "Pay" on the bottom right of the screen to purchase the item, but GatorTrader website realizes that Max has not signed up. He is then redirected to the Login Page where he has to enter his credentials to use these features.

Guest Item Page



The Guest Item Page features the GatorTrader logo at the top left. A large, stylized dollar sign (\$) icon is centered in the upper half. Below it, there's a table with two columns: "Title" and "Price". Under "Title", there is placeholder text: "Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum." Under "Price", there is also placeholder text: "Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi-nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptate nulla pariatur?" At the bottom right of the page are two buttons: "Sign in to Message" and "Flag".

Login Page



The Login Page features the GatorTrader logo at the top left. The word "Login" is centered above a form field labeled "SFSU Email Address". Below it is another form field labeled "Enter Password". At the bottom of the form are two buttons: "Register" and "Login".

2.3 - Max logs in with his credentials and the login is successful. Max presses the "Message Student" button on the bottom right of the screen to tell the student that he wants to purchase the item. He is redirected to the Message Page where he can see all his messages from his previous purchases. He clicks back on the browser and returns to the Students Item Page. He presses the "Pay" button to make the purchase.

Students Item Page

The page header is "GatorTrader" and the user role is "Student". A large placeholder image for an item is centered. Below it, there are two buttons: "Message Student" and "Flag".

Message Page

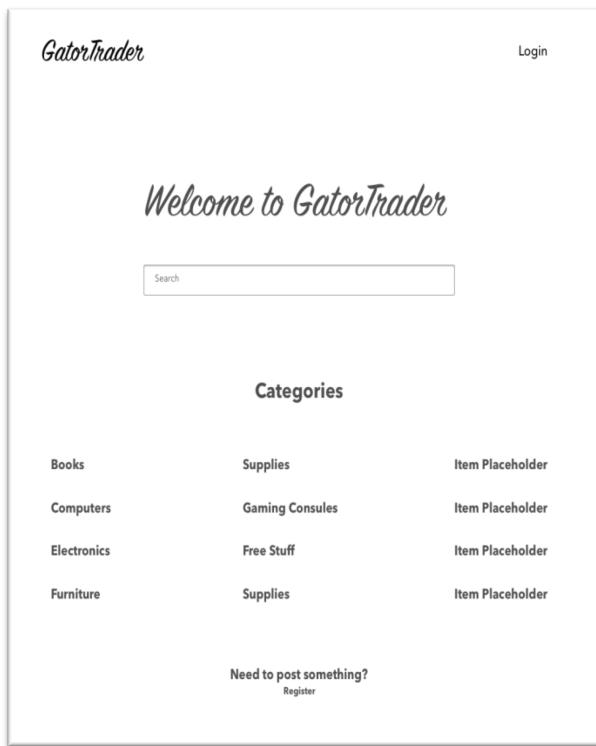
The page header is "GatorTrader" and the section title is "Messages Inbox". A table lists a single message:

Date	Post	Student
03/16	iPhone 6s for sale	jappleseed

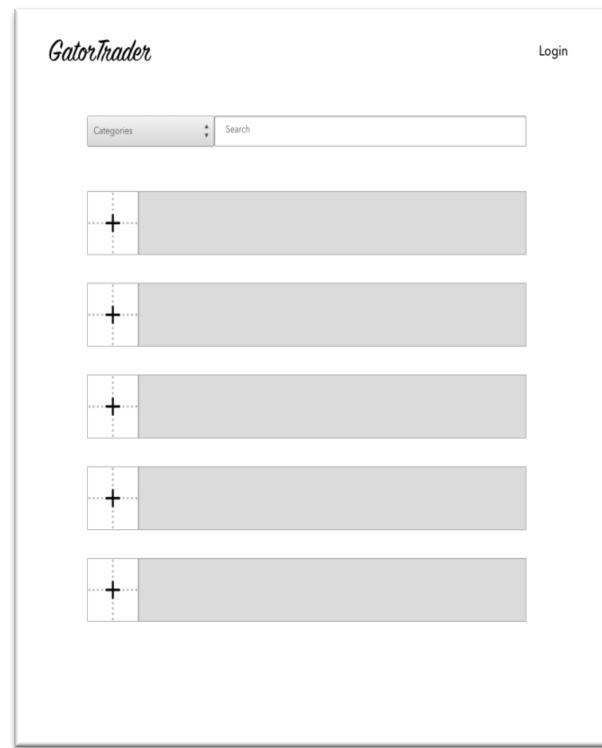
Storyboard 3 - Use Case: Buyer, Guest

3.1 - Sarah is in need of books for the upcoming semester. She goes to GatorTraders, and is directed to the Welcome Page. She types in books in the search bar, and is directed to Item listings page. She is able to see all the items that are under the category of books.

Welcome Page



Item listings



3.2 - Sarah chooses the book that she wants. She is directed to the Students Item Page. She presses the "Pay" button in the bottom right of the screen, and she is redirected to the Login Page. She clicks on the Register button in the middle of the login page.

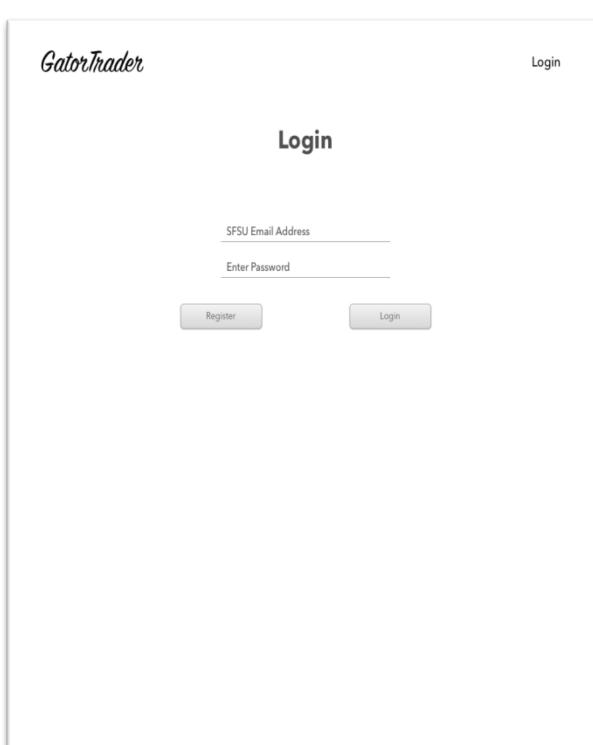
Students Item Page



Title	Price
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.	
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatis videt accusamus est, vel illum qui doloremque sequitur, vel illum qui doloremque sequitur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequitur neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur, vel illum qui iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugit quo voluptas nulla pariatur?	

[Sign in to Message](#) [Flag](#)

Login Page



GatorTrader [Login](#)

Login

SFSU Email Address
 Enter Password

[Register](#) [Login](#)

3.3 - Sarah wants to register for the website, but she sees that she needs an SFSU Email address in order to register. Sarah is just a guest. Sarah cannot sign up for Gator Traders.

Register Page

The screenshot shows a web page titled "GatorTrader" with a "Create Account" form. The form includes fields for "SFSU Email Address", "Enter Password", and "Re-enter Password", followed by a "Register" button. A "Login" link is visible in the top right corner.

GatorTrader

Login

Create Account

SFSU Email Address
Enter Password
Re-enter Password

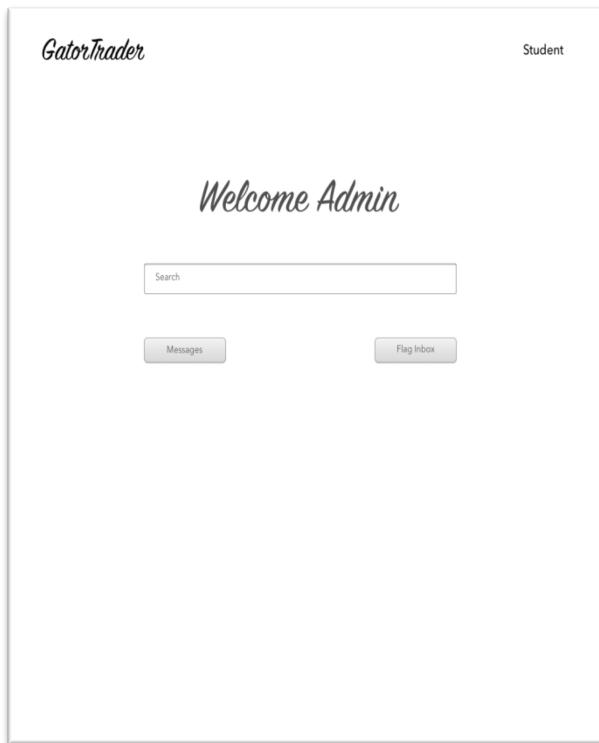
Register

Storyboard 4 - Administrator

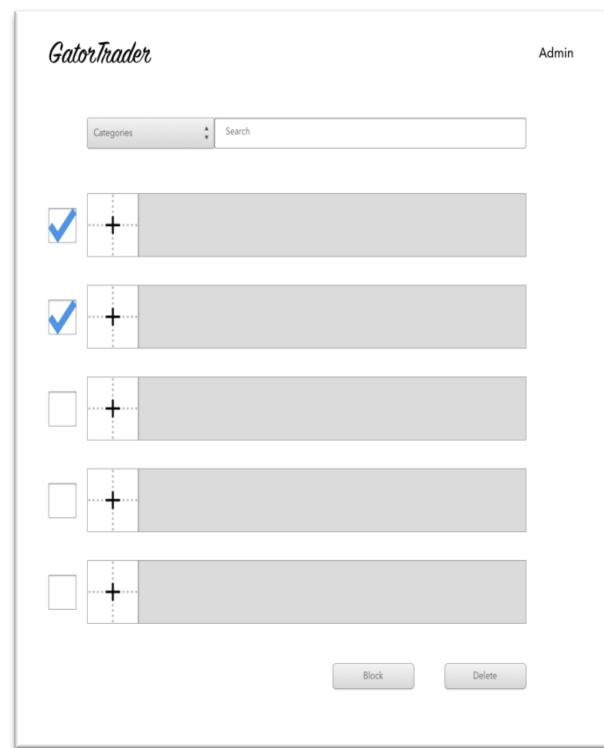
4.1 - Allan is the administrator of the website. When he logs in with his credentials, he is brought to the Admin Account page. He enters a keyword in the search bar and is brought to the Admin Results Page. He is browsing the listings placed on the site, and stumbles upon a vulgar listing.

Allan decides that it is inappropriate. He clicks on the check mark on the item, and clicks the “Delete” button in the bottom right of the screen to delete the item from the listing.

Admin Account page



Admin Results Page



6. High Level System Architecture

The development team is building its product with Amazon Web Service using a Linux, Apache, MySQL, and PHP server, otherwise known as a LAMP Stack. This will provide for optimal uptime, scalability and a greater level of consistency for user's experience. We have chosen to use the Symfony PHP framework with the deployment of this web application as the framework is widely used which has cultivated abundant resources. For design purposes, we have decided that the Bootstrap Stack will allow for the best browsing experience on the four most relevant browsers used, Google Chrome, Firefox, Safari, and Microsoft Edge. To help with design elements other languages in the Bootstrap Stack such as JQuery, JavaScript, and Less will be used sparingly to implement effects.

Amazon Web Services

- Cloud computing platform provided by Amazon. Web services used for remote computing services.

Apache

- Unix based web server that are widely used in HTTP servers on the internet.

Bootstrap

- Open-source front-end web framework for designing websites and web applications.

Linux

- Open sourced operating system modelled for Unix.

LAMP

- Linux Operating System
- Apache
- MySQL Database
- PHP programming

PHP

- Open source scripting language for web development.

SQL

- List of commands used to communicate with a database. SQL statements used to perform updates and retrieve information from a database source.

Symfony

- PHP web application framework and has set of usable PHP components and libraries

Search Architecture

- Search algorithm will be inputted from PHP file to narrow filters from words in database to autofill through each letter inputted in search bar. Terms will come from SQL database system which are organized in different tables based on categories.

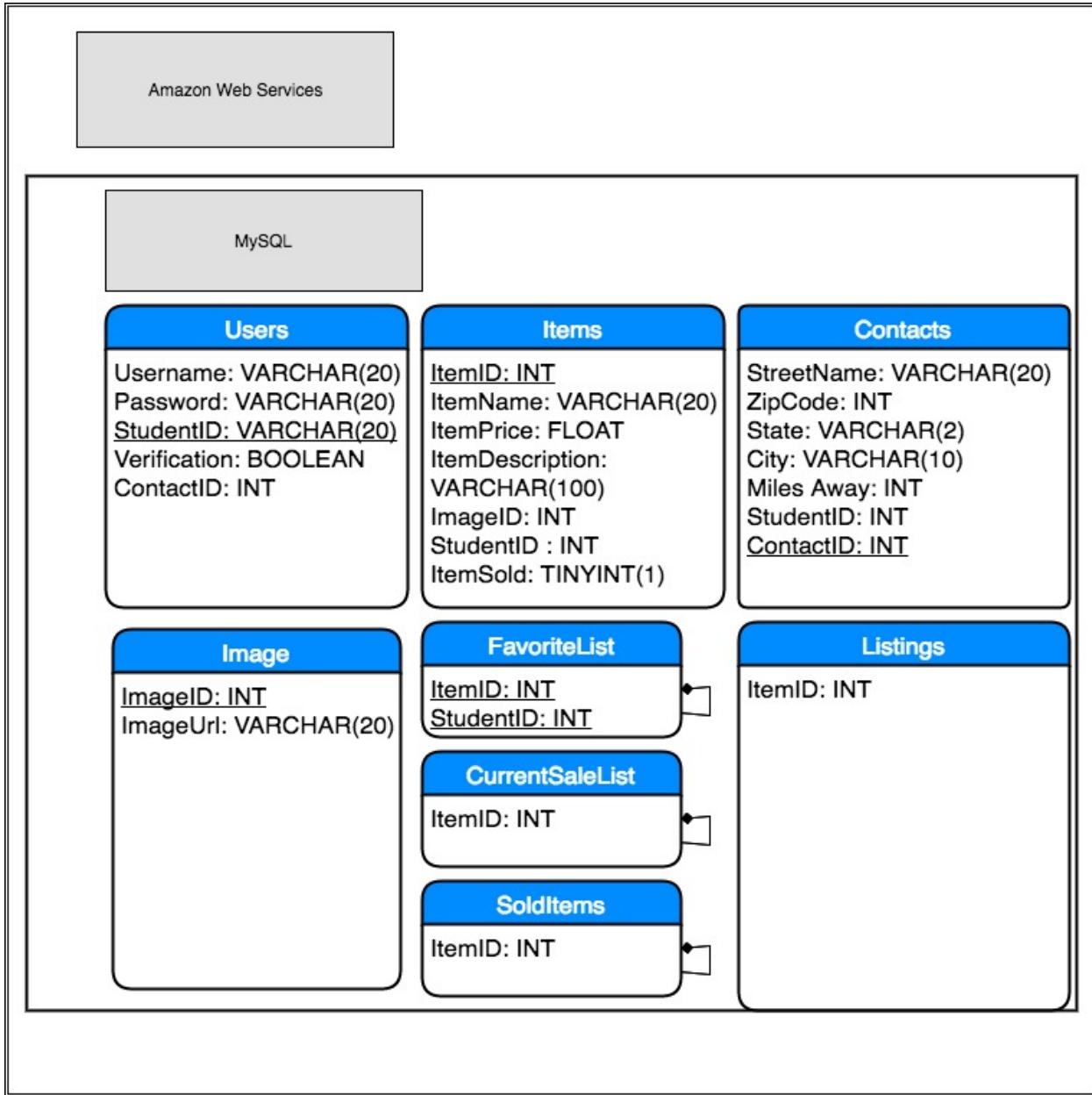
Prioritizing of Items

- Filters will be featured in the application that will prioritize item display. It will be located in the correct categories best fitted.

Image - Files

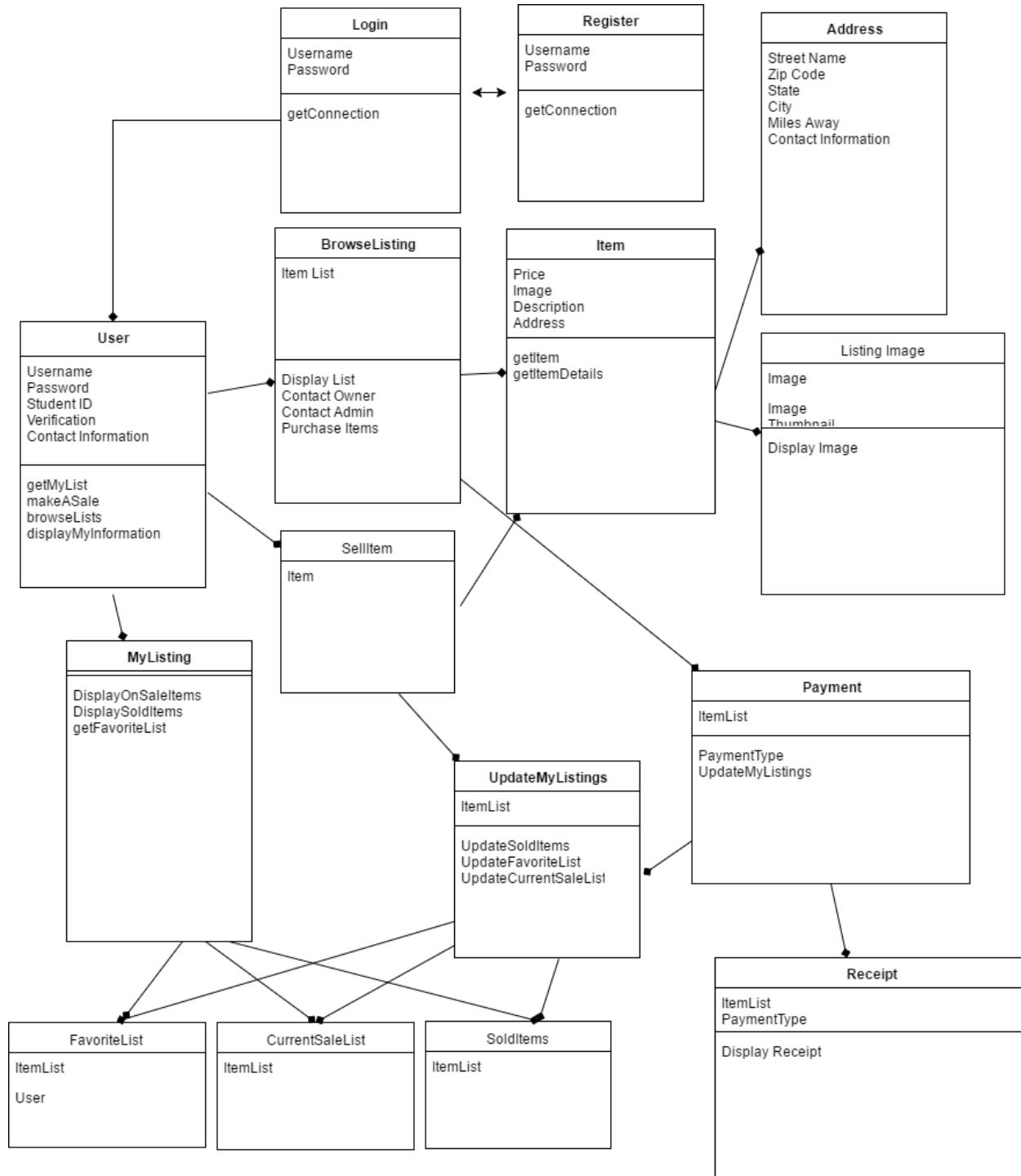
- We will have images in file systems. Files will be saved as thumbnail images or images in the databases. Images will be kept as PNG extention, and will be kept at a size of 300,000 bytes, bit depth will be 32. All images will be fixed sized to be displayed on the website.

Database Organization

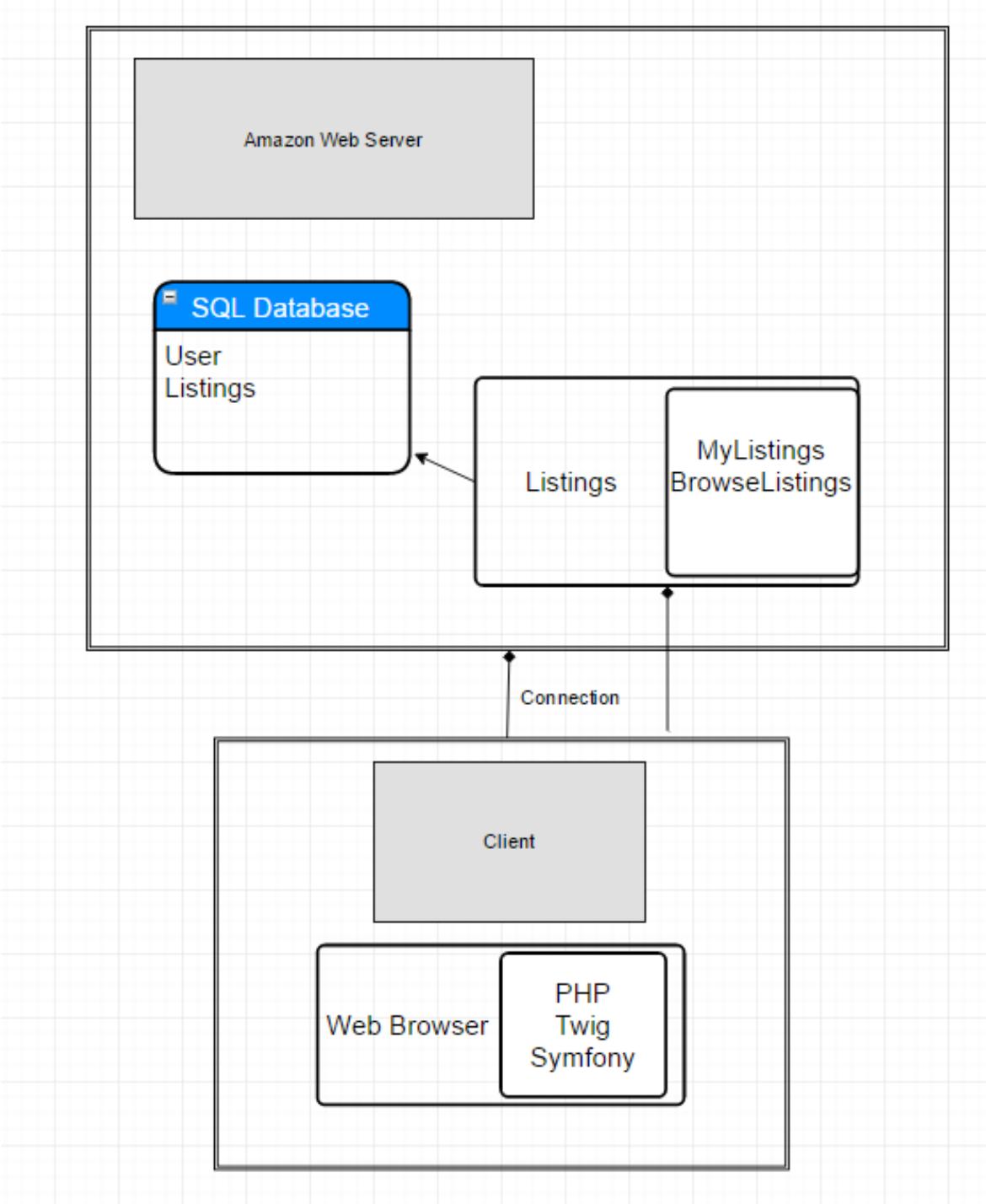


7. High Level UML Diagrams

UML Implementation Diagrams



Component and Deployment Diagrams



8. Key Risks

Risk Type	Risk	Solution
Skills Risk	Varying degrees of skills in programming the front and back end of the project.	Set up a meeting outside of the weekly one, to help familiarize members with the tasks they'll be doing in the project. Share links that are helpful in teaching certain tasks.
Schedule Risk	Team members have other projects they are working on. They may not spend as much time on this project compared to the others.	During the weekly group meetings, assign tasks for each member to accomplish.
Technical Risks	Team members have little experience with server based applications and managing GitHub	Create a wiki to distribute information to the team as technical manuals.
Teamwork Risks	Structural organization issues may become apparent as people begin to work independently on the same task.	Create a systematic way for each engineer to update their progress of the task assigned.
Legal/content risks	Since we don't have and image filtering there could possibly be an issue with copyrights.	Create disclaimers preventing liability and a system for take down requests.

9. Team Organization

Javier Kirksey (CEO) – Project Manager, Data Collection

Responsible for research, data collection, front end management, and team organization. As well as scheduling, and making sure the team functions and operates smoothly.

Evan Terry (CTO) – Technical Lead,

Responsible for keeping our team running efficiently in relation to our technical needs.

Brandon Chiong – Software Engineer, Stack Engineer

Responsible for contributing to and helping create a strong back end for GatorTraders.

Chohee Kim – Software Engineer, Back End Development Coordinator

Responsible for contributing to and helping create a strong back end for GatorTraders.com database.

Kendrick Kwok – Software Engineer, Database Engineer

Responsible for contributing to front end design, as well as relating front end design to system back end. This role oversees the front end and the backend in order to ensure they are both consistent and running smoothly with one another.

Marc Panilio – Software Engineer, Front End Development Coordinator

Responsible for contributing to front end design, as well as relating front end design to system back end. This role oversees the front end and the backend in order to ensure they are both consistent and running smoothly with one another.

Ramsay Wong – Software Engineer, Front End Development

Responsible for contributing to front end design, as well as improving the GatorTraders.com User Interface to ensure a good user experience. Also to collect data that will be used throughout GatorTraders.com

Checklist

1. Updated and improved use cases -----	Complete
2. Updated and improved Data Definitions -----	Complete
3. Updated and improved Functional Specs -----	Complete
4. Updated and improved Non-Functional Specs -----	Complete
5. UI Mockups and Storyboard -----	Complete
6. High Level System Architecture -----	Complete
Database layout -----	Complete
7. High Level UML Diagram -----	Complete
Deployment Diagram -----	Complete
8. Key Risks -----	Complete
9. Team Organization -----	Complete

Revision Table

Date	Revision	Revised by
03/12	Mockup Creation	Evan Terry
03/12	High Level UML Diagrams	Kendrick Kwok
03/13	Updated and improved Data Definitions	Chohee Kim & Evan Terry
03/14	UI Mockups and Storyboard	Marc Panlilio & Kendrick Kwok
03/14	Document Design Changes	Evan Terry
03/14	Logo Design	Kendrick Kwok
03/15	Document Verification	Evan Terry