SW Engineering CSC 648/848 Fall 2018

Project/application Title and Name: GatorList
Team 12:

- Pablo Escriva
- Syed Kazmi
- Stephanie Santana
- Harry Zhang
- Shi Yi Li (Jack)
- Marlo “Mars” Sandoval
- Johnny Huynh

Document Name	“Milestone 1”
Date	10/03/2018
Version	1.0 (first created)

Index of contents

1. Executive Summary

Here at GatorList, we are developing SF State's next biggest buy and sell platform. Right now, there are little alternatives to students posting fliers to sell their products, but that needs to change. We want the community to forget about the days of pesky fliers in the hallway that get ignored most of the time.

We want to bring together buyers and sellers to make transactions hassle free because the SFSU community deserves better. Students are typically busy bodies, so looking for products should be as fast and precise as possible. To do so, products should be categorized accordingly. Also, sellers must be accessible to setup transactions or answer any questions through our messaging system.

If there was one way to describe how special our trading platform is, it would be how we promote ease of use for the user. Our startup team, being students ourselves, know the frustrations of similar online services. We want every aspect to be as simple as possible, from the steps needed to post a listing, searching the catalog, to meeting the buyer or seller.

2. Data Definitions

1. Permission - The amount of access are given to each tier of account
2. Admins - can accept and delete items that are posted
3. User - can post and able to talk to sellers
4. Category - separate items posted for easier search
5. Unregistered User - can search and make post but need to make account before posting on site
6. Message - a system for buyer and seller to talk to each other and set up safe locations for exchange
7. Log-in/Register - a page for user the ability to choose to log in to their account before post or to register if they don't have an account
8. Watch-List/Favorite - able to let user to add an item to a list for future easy access
9. Notification - notify the seller if a buyer had contacted him
10. Items - A post of items being sold by the seller that contains attributes (title, picture, price, description, category, user who posted)
11. Price - selling cost of an item
12. Title - a head of an item being sold
13. Picture - graphical representation of the item being sold
14. Description - a few lines that describe the characteristics of the item being sold
15. Search Bar - allows users to search for keywords for items they wanted to buy

3. Personas and Use Cases

1.0 Persona - Veronica:
Veronica is a freshman at SF State and is very studious, wanting to do well in all of her classes. She also works part-time, but the money she earns barely covers her necessities, so Veronica is very thrifty. Veronica doesn't shop online that much but when she does, she wants a quick and easy experience.

1.1 Use Case - Unregistered User:
Veronica goes to GatorList, and she sees several featured categories and posts on the home page. Veronica goes to the search bar and types an item she's looking for. She sees all the postings related to that item listed in order from cheapest to most expensive, with a picture of the item, the item's title, and the item's price. Veronica sees an item she likes, clicks that item, and is given the item's title, description, category, price, and the item seller. Veronica clicks the item seller's name to contact the seller, and then she is prompted to register to the site or log in as a registered user before she is able to continue. Veronica chooses to register and provides a username and password for the site.

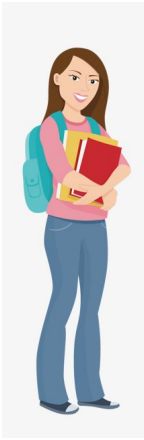
2.0 Persona - Edwin:
Edwin is an art student living on campus and spends most of his free time painting and sculpting. He mostly stays on campus as he doesn't drive and is unfamiliar with the rest of the city. Edwin often shops online for art supplies but he doesn't like spending a lot of time searching and prefers something more convenient.

2.1 Use Case - Registered User:
Edwin is a registered user on GatorList, and when he goes to the site, he logins in using his username and password. After successfully logging in, Edwin sees his dashboard which has his posts, a list of his favorite posts, and his account information. He notices that there are errors in his account info so he edits the mistakes and his account info is updated. Edwin then goes to his favorites list, clicks the item he wants to buy, and is able to message the seller.

3.0 Persona - Luigi:
Luigi is a senior about to graduate later in the semester. Over the past couple of years of attending classes, he has kept all of his textbooks and other school supplies. Luigi is shy so he doesn't know anyone who would want his textbooks and school supplies. Luigi doesn't shop online at all but is willing to try it out.

3.1 Use Case - Posting:
Luigi goes to GatorList but forgets to log in. He sees the home page and clicks the sell button. Luigi is taken to a page that asks him to fill in the details of the item he wants to sell, such as the name, description, picture, price, and category. Once he's done finalizing the details, Edwin clicks to confirm and then he is prompted to register or log in. Edwin chooses to login and the page refreshes with all the item details intact but with his username added under seller. He clicks to confirm again and is taken to his dashboard where he see his item under his list of posts. Edwin also gets a notification that his recent post is being reviewed by the admins before the post becomes public on the platform.

4.0 Persona - Claira:
Claira is the president of a campus club and likes keeping things organized. She also spends a lot of time using social media and often takes pictures to be posted online. Claira likes to shop online but she wishes there was more monitoring of what's posted in the online shops.



4.1 Admins:
Claira is an admin for GatorList, and when she logins into the site, she is given a prompt that asks if she's in shopping mode or admin mode. Claira selects admin mode, and she is taken to the admin dashboard which has a list of all registered users on the site, all public posts sorted by most recent, and all posts waiting to be reviewed. Claira clicks on a post that needs to be reviewed and notices the post contains inappropriate language so she deletes the post. She clicks another post to review and notices the post has inappropriate language and a graphic image so she also deletes the post. Claira eventually finds out one user is making all of the inappropriate posts so she goes to that user's profile and removes the user from the site. After that, Claira clicks a button on the admin dashboard that switches admin mode to shopping mode and Claira sees her shopping dashboard.

4. Functional Requirements

1. Registered and non-registered users shall be able to see the list of postings.
2. Users shall be able to see a list of all the posts from an individual seller.
3. Users shall be able to filter postings by categories.
4. Users shall be able to search for postings.
5. Unregistered users shall be able to register providing name, surname, username, password and optionally with contact details (phone number, email...)
6. Registered users shall be able to keep posts in a list of "Favorites"
7. Registered users shall be able to see their account information
8. Registered and non-registered users shall be able to fill up information for a new post
9. Registered users shall be able to submit a post for reviewing to Admin
10. Registered users shall not be able to revise posts; must be deleted first then submit new post
11. Admin shall be able to approve or deny the posting.
12. Users shall be able to upload at least 1 image and no more than 4 images per posting.
13. Users shall be able to set a price for the item.

14. Users shall be able to set a description for an item that has no more than 500 characters.
15. Registered users shall be able to send a message to another user.
16. Admin shall be able to see a list of all the postings on the site.
17. Admin shall be able to see details about individual postings.
18. Admin shall be able to see a list of all the registered users.
19. Admin shall be able to delete a post.
20. Admin shall be able to delete a user's account.
21. Users shall be able to see information about the team that has created the website as well as information about each of the team members.

5. Non-Functional Requirements

1. Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0 (some may be provided in the class, some may be chosen by the student team but all tools and servers have to be approved by class CTO).
2. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of all major browsers: Mozilla, Safari, Chrome.
3. Selected application functions must render well on mobile devices
4. Data shall be stored in the team's chosen database technology on the team's deployment server.
5. No more than 50 concurrent users shall be accessing the application at any time
6. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
7. The language used shall be English.
8. Application shall be very easy to use and intuitive.
9. Google analytics shall be added
10. No e-mail clients shall be allowed
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated.
12. Site security: basic best practices shall be applied (as covered in the class)
13. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
14. The website shall prominently display the following exact text on all pages *"SFSU-Fulda Software Engineering Project CSC 648-848, Fall 2018. For Demonstration Only"* at the top of the WWW page. (Important so as to not confuse this with a real application).

6. Competitive Analysis

	Letgo	SJSU Bookstore	Amazon	GatorList
Search bar	1	1	1	1
User's Favorite List	0(doesn't exist)	1	1	1
Posting items capability	1	0	1	1
Messaging System(Contacting Buyer or Seller)	1	0	0	2
Log In/Sign up functionality	1	1	1	1

Key: 0 - Not Implemented, 1 - Exists, 2 - Superior implementation

The planned advantages of our planned product to what is already available are including a messaging system for sellers to answer any questions. Our product will provide SFSU students with an actual place to buy and sell items seamlessly, in contrast to another online marketplace such as Amazon, which focuses on the general public. Also, one feature giving us a good advantage vs our competitors involves being able to directly message the buyer and seller for further piece of mind when doing business on our product. Instead of focusing on providing for the general public like the rest of our competition, we focus mainly on SFSU students(or at least students within SF area).

7. High-Level System Architecture

Server Host:
Heroku 1vCPU 512 GB RAM
Operating System:
Ubuntu 18.04
Database:
PostgreSQL 10
Web Server:
Node.js 8.12.0
Server-Side Language:

JavaScript
Additional Technologies:
Dependency: Express, BodyParser, Knex.js
IDE:
VSCode, Sublime, Brackets
PostgreSQL Tools:
pgAdmin, PSequel
Route testing:
Postman
Web Analytics:
Google Analytics
FrontEnd Framework:
Bootstrap
Supported Web Browsers:
Chrome Version 69.0.3497.100
Chrome Version 68.0.3440
Chrome Version 67.0.3396
Firefox Version 62.0.2
Firefox Version 61.0.2
Firefox Version 61.0.1

8. Team

1. Pablo Escriva - Team Lead
2. Syed Kazmi - GitHub Master
3. Stephanie Santana - FrontEnd Leader
4. Harry Zhang
5. Shi Yi Li (Jack) - BackEnd Leader
6. Marlo “Mars” Sandoval
7. Johnny Huynh

9. Checklist

for each item below you must answer with only one of the following: **DONE**; or **ON TRACK** (meaning it will be done on time, and no issues perceived); or **ISSUE** (you have some problems, and then define what is the problem with 1-3 lines)

- 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 and working on it **DONE**

· Team lead ensured that all team members read the final M1 and agree/understand it before submission **DONE**