

SW Engineering CSC648/848 Fall 2019

“Gator Store”

Section 02

Milestone 1

Team 12

Beibei Jiang - Team Lead / Github Master / Back-end Developer

Email: bjiang1@mail.sfsu.edu

Jinying Ren - Front-end Lead

Jesse Smick - Back-end Lead

Angelo Solitario - Front-end Developer

Melissa Endang - Front-end Developer

Jinghan Cao - Back-end Developer

Submitted Date	10/2/2019
Revised and Frozen Date	10/4/2019

1. Executive Summary

SFSU students have the need for buying and selling resources. Seniors about to graduate need to sell items such as books, electronic products and furniture in a short time. These things are exactly what current students and especially for those who have economic burden seek. “Gator Store” aims to offer easy, economic experience for and only for SFSU student from one seamless shopping website.

In our website, buyers can easily browse items by category, can search product using its name or class number, can sort products as well. After clicking on one particular item, they can see its picture, description and other information. Moreover, buyers can communicate with seller via in-site message box after logging in.

As for sellers, they are allowed to post information as well as edit posted items. Everything posted will show up only after it is approved by website admin who deals with improper post request regularly and also make sure in-site communication are healthy and smooth. In addition, lazy registration will be implemented which means anyone can browse and search easily without signing up or logging in until he or she wants to post, send messages or purchase.

One of the unique features of our website is allowing users to search listings not only according to the name but also its related course number. For example, a student is going to take csc510 and would like to buy a textbook. When referring to other shopping website, the student should provide the book name together with maybe other information like author name and book version. On our website, she/he can easily type csc510 in the search box and satisfied results show up immediately. On top of that, meetup spots are limited to a 0.5 mile radius near campus under security camera. Students get products easily, fast and safely compared with other websites.

Our team consists of six current SFSU students who are dedicated to solving student online trade. You can find more information about us in about page.

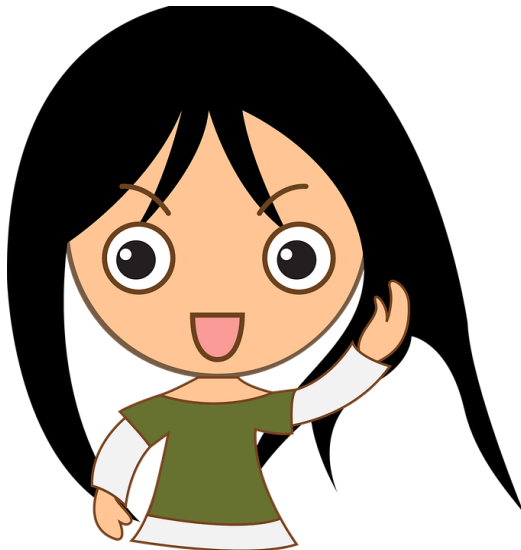
2. Personae and main Use Cases

2.1 Personae



Ali – Admin

Ali is an SFSU student and an administrator on the SFSU buy/sell website. He is very skilled and has a thorough understanding of the website, software, and browsers. He is very organized when it comes to events, files, and a system that makes sense for everyone.



Bella- Buyer

Bella is a fulltime student and who is also working full time. She is a hardworking person. Because she needs to help her parents with paying the bills, she

does not have enough money to pay the full price for the book she needs for one of her classes.



Silvester – Seller

Silvester is a student who is graduating soon. He has no siblings or anybody that can use all his textbooks and things from his dorm. He needs to sell or get rid of his things very soon before he moves. Otherwise, he will need to pay extra to move them to his new place.

2.2 Use Cases

Managing the listings

Ali logs in on the website to check on the listings and approve all the listings. He can browse the listings starting from the most recent listing. While browsing, he realized that a user is trying to post very offensive and inappropriate stuff on the website. Ali proceeded to reject the posting and it will not be visible to the user.

After some time, the same user is trying again to post the same listing. Ali rejected the listing and sent a warning to the registered user that he/she may be banned for multiple and repeat attempts.

Buying

Bella decided to visit the SFSU buy/sell website to see if the book is listed. There she was able to filter the listings by classes offered in SFSU. She found the book and proceeded to contact the seller and buy the book. Before completing the

message, she is prompted to register. After contacting the seller, she was able to acquire the book and a new friend.

Selling

Silvester decided to go to the SFSU buy/sell website because he knows that students will benefit from the low prices of textbooks and things. Silvester is browsing the website and he easily found the button for posting a listing on the website. He filled up the form needed to sell an item and then he was prompted to register. He then proceeded to register and sign up to post and sell. It was easy for him to post pictures of his items. Now, Silvester can go back and check if anybody is interested in buying his stuff.

Editing listings

Silvester has posted many listings on the website most of them sell in a short amount of time. However, one item will not sell for some reason. He decided to check the listing using his dashboard and realized that he did not describe the item well enough. He decided to edit the listing's description to better describe the item. After changing it, he received offers for the item.

Browsing

Bella was not looking for anything specific this time. She is using a different computer and forgot her password. Being lazy, she decided not to log in. She was able to browse the listings without the trouble of having to log in or register. She was able to compare the prices of items that have multiple listings. She visited the website a couple of times and was able to browse freely.

3. List of main data items and entities

- **Unregistered Users:** Users who have not registered or log in to the account. They shall be able to search and browse items, but they cannot post new items or in-site messages to other users.

- **Registered Users:** Users who have accounts and are logged in (can be buyers or sellers). They can browse, search, post and remove items, and they can send in-site messages to interact with other users.
 - Email
 - Username
 - Password

- **Admin:** The admin user has access to the website's administrator panel. The admin keeps the website running efficiently and securely. Their tasks are to manage user accounts, posts request and content.
 - Username
 - Password
 - Remove posts
 - Remove items
 - Issue warnings
 - Ban from posting
 - Contact sellers or buyers

- **In-site Messages:** Communication between registered users (buyer, seller, and admin). They can contact each other with text and pictures about items' problems or questions.

- **Items:** Items posted by registered users (sellers) for sale. Items include:
 - Name
 - Price
 - Description
 - Pictures
 - Condition
 - Distance from campus
 - Category

- **Category:** Users shall be able to look for specific kinds of items efficiently through category (books, electronics, school supplies, accessories, etc).

- **Search:** Users shall be able to find items that they want efficiently with items' names or keywords.

- **Login:** Users can login to their personal account page to access their information.

- **Registered users account:** It records the name, email, address, and purchase details of the registered users.

4. Initial list of functional requirements

Unregistered Users

1. Users shall be able to create an account
2. Users shall be able to view listed items for sale
3. Users shall be able to browse through category
4. Users shall be able to browse through tag
5. Users shall be able to see the remaining amount of items
6. Users shall be able to be redirected to register page if they want to post items for sale

Registered Users

7. +Unregistered Users' functions
8. Users shall be able to log into the website
9. Users shall be able to have their own dashboard
10. Users shall be able to post items attached with descriptions for sale on this website.
11. Users shall be able to post items attached with images for sale on this website.
12. Users shall be able to post items attached with prices for sale on this website.
13. Users shall be able to post items attached with tags for sale on this website.
14. Users' dashboard shall be able to view the in-site messages
15. Users' dashboard shall be able to view past posted items
16. Users' dashboard shall be able to modify past posted items
17. Users' dashboard shall be able to delete past posted items
18. Users' posted items shall be reviewed by an administrator before it is officially posted on website
19. Users' posted items shall be reviewed again if they update the information about items.
20. Users shall be able to send in-site messages among each other.
21. Users shall be able to view approval results from an administrator regarding their proposed items for sale
22. Users shall be able to choose the location within campus to trade off-line

23. Users shall be able to receive confirmation message sent to their in-site message box

Administrators

24. Administrators shall be able to log in the website
25. Administrators shall be able to have administrator dashboard
26. Administrators shall be able to review the items before they are officially posted
27. Administrators shall be required to approve items before they are officially posted
28. Administrators shall be able to send messages to sellers if the items are against the policies

5. List of 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 two major browsers
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 email clients shall be allowed
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
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 Software Engineering Project CSC 648-848, Fall 2019. For Demonstration Only" at the top of the WWW page. (Important so as to not confuse this with a real application).

6. Competitive analysis

Features	Craigslist	Ebay	Facebook Market	SFSU Store
Browse by category	+	+	+	+
Communicate via in-site messaging	+	+	++	+
One-click access to SFSU books	-	-	-	++
Safe meetup spots on campus	-	-	-	++
	key: - = don't have + = has ++ = does well			

We plan on making it easy to browse and submit posts before signing up is needed. A key feature we have over the competition is having predetermined safe meetup spots on campus to recommend to the users. Another feature we have over other sites is easy access to books for SFSU classes. Overall we've strategized to leverage the inefficiencies of existing platforms to bring a competitive advantage into the market.

7. High-level system architecture and technologies used

Server Host: Google Compute Engine 1vCPU 0.6GB RAM

Operating System: Ubuntu Server 18.04 LTS

Database: MySQL v5.7.x

Web Server: NGINX 1.17.x

Front-end Language: bootstrap, css, html, javaScript

Server-Side Language Engine: Node.js v10.x

Server-Side Language: JavaScript

Additional Technologies:

- Web Framework: Express.js v4.x
- Web Analytics: Google Analytics
- VirtualBox v6.x
- Vagrant v2.x
- Ansible v2.x

8. Team and roles

Beibei Jiang - Team Lead / Github Master / Back-end developer

Jinying Ren - Front-end Lead

Jesse Smick - Back-end Lead

Angelo Solitario - Front-end Developer

Melissa Endang - Front-end Developer

Jinghan Cao - Back-end Developer

9. Checklist

- 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

DONE

- 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