

Software Engineering CSC 648/848 Fall 2019  
Gator- Trader

Section 2  
Team 10

TEAM LEAD  
Ibraheem Chaudry

## Table of Contents

<b>1. Executive Summary.....</b>	<b>3</b>
<b>2. Personae and Use Cases .....</b>	<b>4</b>
Personae #1:.....	4
Personae #2:.....	4
<b>Goals and scenario: .....</b>	<b>5</b>
Use Case #1:.....	5
Use Case #2:.....	5
<b>3. List of main data items and entities.....</b>	<b>5</b>
* Users .....	5
* Admins .....	6
* Sales item.....	6
<b>4. Initial list of functional requirements.....</b>	<b>6</b>
Unregistered users.....	6
Registered Users .....	7
Administrator .....	7
<b>5. List of non-functional requirements.....</b>	<b>7</b>
<b>6. Competitive Analysis .....</b>	<b>8</b>
Services offered: .....	8
Price: .....	8
Time to get product:.....	8
Safety: .....	9
Product handling: .....	9
<b>7. High-level system architecture and technologies used:.....</b>	<b>9</b>
Frameworks: .....	9
Date Base: .....	9
API: .....	9
Tools: .....	9
Browsers:.....	10
Deployment Platforms: .....	10
Additional Technologies: .....	10
<b>8. Team and Roles:.....</b>	<b>10</b>
<b>9. Checklist.....</b>	<b>10</b>

# 1. Executive Summary

San Francisco State University caters a population of approximately 30,000 scholars, of which a good percentage consists of students housed at on-campus residential areas. Complementary to the large student body, the university offers 118 different bachelor's degrees, 94 master's degrees, and 5 doctoral degrees among six academic colleges. To fulfill their needs students depend on on-campus and off-campus resources such as the SF State bookstore, eBay, amazon, etc. However these resources are not always efficient in terms of cost and availability. SF State has always harbored a spirit of investing in its students, through provision of convenience and creating an environment that promotes education. Our team proposes to develop an application named 'Gator Trader' that will serve as an online e-commerce website specifically for SF State students. This web application will allow students to buy and sell items and will serve as a single centralized platform where needs will be met.

Gator Trader will allow SF State students to buy and sell items ranging across a number of categories i.e. books, furniture, electronics etc. Valid Students will be able to make posts of items they want to list for sale and browse already listed items to find products they may need to purchase. A potential buyer can direct message the seller and they can arrange to meet at a safe location on campus. All posts will be managed and scrutinized by an administrator to ensure that no university policy is violated and vulgarity is avoided. Gator Trader's model is essentially similar to that of eBay or craigslist but what makes it unique is that it's a localized solution specifically for SF State students which gives it a competitive edge. Students would be able to browse books for specific SFSU classes and transactions can be completed in no time through meetups on-campus. Other university specific services can also be made available to students through this web application e.g. tutoring.

Our startup team consists of engineers skilled in full-stack web development work and have completed e-commerce application projects in the past. We have competency in both front-end and back-end development and can guarantee the delivery of a quality product.

## 2. Personae and Use Cases

### Personae #1:

Beth- full time student at SFSU

- Works stressful part-time job just to be able to afford SF rent
- In debt with student loans and is very frugal financially
- Despite busy schedule she is 4.0 college student with academic honors
- Motivated and positive person
- Just moved to new apartment
- Is about to start a new semester
- Free time is a privilege for her, only free during the hours when most stores are closed!
- Does not own a car



### Goals and Scenario:

Beth wants to obtain the textbooks she needs and completely renovate her apartment before the semester starts and do so without breaking the bank! A friend tells her about a local SFSU market website she found.

### Personae #2:

David – part time faculty at SFSU

- Currently teaches Computer Science at the campus
- Plans of moving away from campus housing to a condominium downtown that he has already purchased
- Being a hoarder, he has loads of stuff to sell (His current fiancée says he needs to get rid of this “collection of useless junk”)
- Does not prefer shipping items
- Doesn’t have time to auction his goods on sites like “eBay”
- Loves to help out students
- Social individual who loves bargaining



### Goals and scenario:

David reluctantly wants to get rid of the contents in his apartment so he can appease his fiancée and move into his new place by the coming week. He remembers a student made SFSU market website mentioned in his department's message board and decides to check it out.

### Use Case #1:

After a busy shift at work Beth navigates to the SFSU market webpage to solve the problem described above. She searches for the first item she needs, a textbook, by selecting the appropriate category and typing in its name. A neatly displayed page with multiple results is then shown with a picture, price and other details posted by the seller. Upon clicking on one of the search results and selecting message seller, a simple account creation/login prompt is shown which takes Beth little time to complete. After filling this out, she messages the seller and organizes a meeting at 9pm after her last class of the evening at a safe location on campus. Satisfied with the ease of this process, Beth navigates back to the home page and starts looking for everything else she needs.

### Use Case #2:

David browses to the SFSU market website in hopes of both making a few extra bucks and helping out some students during his moving out process as described above. David decides on a first item to sell, his old Java for Dummies textbook, and clicks on the sell link on the frontpage. He then fills out the required information presented and clicks 'post item' upon completion. Following a prompt to login/register David then repeats the process for all the rest of the items he has. The following afternoon David returns to the website and logs in. He is redirected to a page that shows all of his items have been approved by the system administrator and have been posted to the site. In addition to this, he has multiple new message notifications from people interested in his goods. After a short message exchange from one of these individuals, David schedules a meeting to sell his textbook at 9pm that evening.

## 3. List of main data items and entities

The user entity will be used to store information for all registered individuals who will post on the site. Each user is required to input a name, email, password, and username to be able to sell and buy. The email will be used to authenticate the user as an SFSU student. Each user will be required to add a username and password, the username which will be displayed on each post for privacy.

### \* Users

- Name
- Email
- Password

- Username

The admin entity will be used as a website administrator who will review all posts before being public. Admin accounts will be created for admin use only. Admins are required to set the account with a valid name, which will be used for the statuses of posts. Email address and password are required to sign in to the admin account.

## \* Admins

- Name
- Email
- Password

Sales item entity is used to store all the information about the item being sold. Each item is required an item name and a category which it will be posted under for faster search. Price is required to post the item. Picture is required for the condition of the item being sold. Description of the item condition, and information if needed. Item location is optional for posting user's privacy. The meeting location will be required to inform the buyer user of the meeting location

## \* Sales item

- Name
- Category
- Price
- Picture
- Description (Optional)
- Item Location (Optional)
- Meeting Location

# 4. Initial list of functional requirements

## Unregistered users

1. Unregistered users shall be able to register for an account.
2. Unregistered users shall be able to browse through the website.
3. Unregistered users shall be able to filter search results.
4. Unregistered users shall be able to use the search bar to search for items.
5. Unregistered users shall be able to see item/product's images.
6. Unregistered users shall be able to see the price/ cost of a product.
7. Unregistered users shall be able to see the current status(New, used) of an item/ product.

## Registered Users

1. Registered users shall be able to log into the website.
2. Registered users shall be able to sell products onto the website.
3. Registered users shall be able to do whatever an unregistered user can do.
4. Registered users shall be able to contact other registered users about products.
5. Registered users shall be able to propose to admin about putting their product on the site.
6. Registered users shall be able to take down their post.
7. Registered users shall be able to receive messages from other registered users.

## Administrator

1. Admin shall be able to do whatever a registered user can do.
2. Admin shall be able to accept proposals from registered users.
3. Admin shall be able to reject proposals from registered users.
4. Admin shall be able to see the proposals from registered users.
5. Admin shall be able to see the info of a registered user.
6. Admin shall be able to check the status of proposals.

## 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

	Craigslist	Ebay	SFSU Bookstore	Gater Trader
Range of products	+	+	-	++
Price	++	++	+	++
Time to get product	+	-	++	+
Safety	-	++	++	+
Buyer handle product	+	-	+	+
		++ = Superior		
		+ = Good/Fair		
		- = Minimal or not existant		

Our service will provide a similar service to craigslist but for San Francisco State University students only. Our goal is to offer a more directed selection of products that will be more relevant to the students using the service. While you are allowed to sell items on eBay and craigslist, it is generally a hassle and potentially dangerous to set up an appointment with a stranger on craigslist and it costs time and money to ship items using eBay. Our service will suggest places and times to meetup that are public and safe. We also offer more products than the on campus bookstore as well as offer services such as tutoring and personal training. The SFSU bookstore only sells at full price or a large markup on used books, our service will allow students to sell their used books for more than the book buyback offered by the bookstore.

### Services offered:

eBay and craigslist has a wide range of products.  
Bookstore and we will have more limited range of products aimed towards students.  
We will also allow for advertising services like tutoring or general handyman work.

### Price:

eBay, craigslist and our site will be for second hand products at a reduced price.  
Bookstore sells new products at full price.

### Time to get product:

eBay you have to wait for a product to be shipped and could take a few days/weeks.  
Craigslist, bookstore and our website will allow for in person pickup.



## Safety:

Craigslist has the potential to lead to an unsafe situation.

Our site will have safeguards to use to allow for a safe meetup (safe, well-lit location suggestions)

Bookstore and eBay safe to buy from.

## Product handling:

Craigslist, bookstore and our site allow buyer to see/handle product before purchase.

eBay does not allow the buyer to see/handle product before purchase.

# 7. High-level system architecture and technologies used:

## Frameworks:

- Bootstrap
- Node.js 10.16.3
- Express.js

## Date Base:

- MySQL 8.0.17

## API:

- JavaScript

## Tools:

- Git & GitHub
- Visual Studio Code
- IntelliJ
- Brackets
- Sublime text

### Browsers:

- Google Chrome
- Firefox

### Deployment Platforms:

- AWS EC2
- AWS Linux AMI

### Additional Technologies:

- Web
- Google Analytics

## 8. Team and Roles:

Name	Role
Ibraheem Chaudry	Team Lead
Tom Sechrist	GitHub Master, Back-end Engineer
Saleh Zahran	Document Master, Back-end Engineer
Alexander Beers	Back-end Lead
Paul Lueng	Front-end Engineer
Lance Santos	Front-end Lead

## 9. Checklist

Task	Status
Team found a time slot to meet outside of the class	ON TRACK

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