

SW Engineering CSC648/848 Fall 2018

EduGator Team 03

Milestone 1
October 4th, 2018

Team Members

James Andrews jandrew3@mail.sfsu.edu (Team Leader)
Ian Dennis
Ivan Varela
Chris Johansen
Liyao Jiang
Shirin Namiranian
Kenneth Surban

Revision Number	Description	Date
1.0	Initial Version	10/04/2018
2.0	Revised based on comments	10/10/2018

1. Executive summary

As a student, the process of buying and selling books, old room furniture, or even getting tutoring can be a hassle. You can check Amazon or eBay for your textbook, but can't be sure if you are getting the right volume you need for the class, or if it's the best price. EduGator aims to make this process of buying and selling items more streamlined for the SFSU campus. Since the application will be tailored to SFSU students, we will have books sortable by SFSU course they pertain to, and also sort other items into categories that students would find useful. Additionally, we plan to make the process of finding a tutor as easy as possible by allowing SFSU community members themselves create listings to tutor their fellow students in courses offered by the university.

EduGator aims to allow students to buy and sell items such as electronics and furniture, but our main focus will be on allowing students to find books relevant to SFSU courses and allow students who need additional help to find tutors within our community. Books will have attributes that link to a specific course, and our tutoring service will allow users to sell their services for that course.

We chose to focus mainly on books and the tutoring service because we feel that is a critical piece missing in the marketplace for SFSU students. We will also provide the option to sell and buy other items, so users won't be limited to just books and tutoring, and more importantly, so that our product can grow to become the new hub for the SFSU student marketplace.

Our product simplifies the process of buying and selling items to SFSU students. When it comes to books, users will have identifiers that connect them to specific SFSU courses, which facilitates SFSU students finding books for their classes. These identifiers will also be used for students looking for and selling tutoring services.

EduGator has an innovative focus on selling books and tutoring services. The tutoring will be specific to the SFSU campus, allowing our product to hit a niche market that has yet to be serviced, and allows us to serve SFSU student community as closely as possible.

Competitor apps include Amazon and Ebay, who focus broadly on selling all types of items, and do not specifically target SFSU community needs. We plan to tailor our service to align with the needs of a SFSU student.

Our team consists of seven computer science students from SFSU that want to simplify the process of selling items and buying items for our fellow students, and approve the overall quality of campus life here at SFSU.

2. Personas and Use Cases

2.1 About Elisa - "I love to explore and try new things"

Elisa is a freshman business major student in SFSU. She is new to the city, lives in an apartment close to school and wants to settle down quickly to college life. She likes shopping online and social media a lot. She is a fast-paced, socialized person, and makes friends easily. Obtain everything for her new apartment and school materials at a reasonable price and quickly. Likes to share good deals she finds online with her friends.



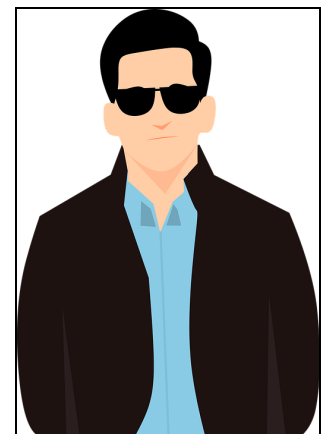
2.2 About Mike - "I love technology"

Mike is a busy SFSU student who is majoring in computer science and is proficient with technology. His phone is with him 24/7 and he never fails to respond to his emails in a timely manner, especially since he spends most of his time on his laptop. Mike is a hard worker and loves jobs where he can work remotely. Mike wants to do a great job as an admin of EduGator so that he can be recommend for other admin jobs in the future. He wants to have power over postings because he hates other websites that have unhelpful and useless postings.



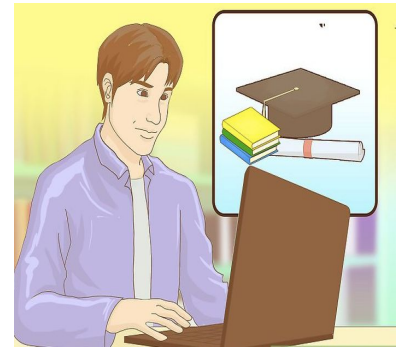
2.3 About Tyler - "I love nice things and money"

Tyler is from Los Angeles, CA. He is a 23 year old student studying Accounting at San Francisco State University. He is in his junior year and has been at SFSU for 5 years. He lives in an off-campus apartment and is financially dependent on his upper middle class parents. Tyler has a bad addiction at spending his monthly allowance for school. He will browse the Internet, look for good deals, and make spontaneous purchases. Tyler is really in to being always up to date with his technology and loves getting the latest Apple product. Tyler often finds himself having a lot of tech, items, and old books he wants to get rid of but wants some money out of it. Tyler wants a platform where he can sell his mountain of unused belongings and old books to offset his bad spending habits and make extra money.



2.4 About Arnold - "I love having enough money for rent"

Arnold is a 21 year old student who is a senior at SFSU and is majoring in computer science. He loves computer science, math, and physics; he has such a passion for those topics that he enjoys teaching and tutoring students in those topics. Ever since moving to San Francisco Arnold worries about not having enough money to pay his rent, he's always on the lookout for ways to make money. He wants to be able to easily make money while he is studying and needs a job that doesn't require much travel to be flexible with his class schedule.



Use Case 1 - Unregistered user browsing items and buying items

On a Saturday morning, Elisa is browsing EduGator, a sell and buy website for SFSU students which a friend told her about. She wants to buy a textbook for her business class, a brand new one is very expensive for her so she wants to find a used textbook. She is just checking out the site without registering first. She searches for textbooks by typing in her class number into the search bar, and surprisingly she finds the class book that is sold by a former student at an incredibly low price. She tries to contact the seller and is prompted to register or login. She creates her account, leaves a message to the seller, and the seller contacts her and they make a deal the same day.

Use Case 2 - Admin managing posts and banning users

Mike is an admin for EduGator. As an admin Mike can't buy or sell items or services, his only responsibility is to administer EduGator. When administering EduGator Mike is responsible for the status of items and services posted. When an item or service is posted by a seller Mike receives a notification at his e-mail address that an item or service is waiting to be reviewed. That is to say that each item or service are bound to a flag. When an item or service is posted the default value of the flag is pending. When a flag is pending the item or service is not visible to potential buyers. For the flag of an item or service to be set to live Mike must review the item or service and approve it. If the item or service are not up to Mike's standards he can contact the seller and ask them to change the posting, or if the posting is inappropriate Mike can delete the posting and ban the user from creating further postings. If by some chance Mike does approve an inappropriate posting, users can flag the posting as inappropriate, in which case Mike will receive a notification at his e-mail address, Mike can then delete the posting.

Use Case 3 - Seller checking the status of his posts and messages from buyers

Tyler finds a buy and sell site exclusively for SFSU students. Without registering, he is able to browse through the site. He sees the site is free and easy to use. Tyler wants to make a post for the items he wants to sell. He plans to sell his Apple Series 1 Watch and some old textbooks. He is prompted with a registration login screen. He signs up and creates a post, including the name of the item, a photo, a description, the category, and the condition. For the books, he includes the same: name, photo, description, category, condition, and also the book's course title. After some time, he wants to check the status of his post. He goes to our site, logs in, and sees the condition of his post. He checks back the next day and sees on a dashboard someone messaged him about his book.

Use Case 4 - Seller posting items

For Arnold to pay his rent and tuition, he needs to make money while he is at school. He needs a job with flexible hours to work and have enough time to study. Arnold is feeling a little worried. Thus he has to offer some services to other students for making money. He posts on EduGator in the tutoring section, he creates posts in the Math, Physics, and Computer Science sections. Arnold can choose how much he charges each student per hour for his tutoring services. After his post has been up for a couple of hours EduGator notifies him that he got a request from Suzy who is a junior at SFSU and majoring in computer science, to tutor her about the Java language for her CS413 class. Since Arnold got connected with Suzy through the website, he is able to help her for an hour. He feels safe and secure that Suzy will pay him money for his time helping her with Java. Arnold can sit down and enjoy the rest of his evening by checking his bank account and not worrying about how to pay his rent and bills.

3. Data Description

3.1 Unregistered Users:

Can browse items, course books, and tutoring services. Does not need to register or login to do this.

3.2 Registered Users:

Can browse, post, view the approval status of, remove, and send messages to other users about items, course books, and tutoring services. They need to be registered and logged in to access this functionality for their account. Each account is also linked to a user registration record that the registered user completed when initially created their account.

3.3 Admin:

An employee of the company hired to change the approval status of or remove a course book, item, or tutor service that a registered user has posted. Needs to be logged in.

3.4 Item:

An item available to be purchased, other than a course book. Has a name photo, price, description, category, condition, approval status, and is associated with the registered user who posted it.

3.5 Course Book:

A book associated with a particular course at SFSU. Has a name, photo, price, description, course, condition, approval status, and is associated with the user who posted it.

3.6 Tutor Service:

A post by a registered user to tutor for a specific course. It has a price, description, course, and approval status.

3.7 Approval Status:

The approval status of a particular tutor service, item, or course book. Viewable by the admin and the registered user who posted the item for sale. Only the admin can change this value.

3.8 Course:

The name of a course at SFSU. Each course book and tutoring service has a course associated with.

3.9 Category:

A classification for items which enables them to be sorted into different groups.

3.10 Message:

A message sent between two registered users. It is used for registered users to communicate about tutoring services, course books, and items.

3.11 User Registration Record

A record that an unregistered user completes to become a registered user on the site. The user registration record includes the name of the user, the user's password, and the user's email.

4. High-Level Functional Requirements

4.1 Search Bar:

Registered and unregistered users shall be able to search items by name, category, and condition. Registered and unregistered users shall be able to search books by name, course, based on the type of item being searched. Registered and unregistered users shall be able to search by course for tutors.

Admins shall be able to view items, books, and tutoring service posts that have a pending status.

4.2 Filtering types:

Unregistered, registered, and admin users shall be able to filter by type of item, price, and category.

4.3 Item filters:

Registered, unregistered, and admin users shall be able to filter type of item by: furniture, books, general items, etc.

4.4 Tutoring:

Registered, unregistered, and admin users shall see tutors related to the book, professor, or class number they are searching.

4.5 Browsing:

Registered and unregistered users shall see a list of items/services to browse that other users have posted with approved status.

Admin users shall see a list of items/services to browse that have not yet been approved(pending status).

4.6 Posting Items/services:

Registered users shall be able to post their personal items to be sold. Registered users shall be able to set the price, condition, type, and name of the item. Posted items and tutoring services shall be subject to approval via administrators of the site.

4.7 Messaging for Buying Items/Services:

Registered users shall be able to contact other users to buy items/services they have posted.

4.8 Admin controls:

Admins shall be the only users to approve posts. Admins shall be able to remove posts.

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

	Ebay	OfferUp	EduGator
Demographic Emphasized Services	+	++	++
Tutoring	-	-	++
Browse	++	++	+
Filtering	++	+	++
Text Search	++	+	+

Our future product's competitive advantage lies in it's tailored service to the SFSU campus community. We offer a tutoring marketplace for SFSU students, which is unique facet of our product as is evidenced in the competitive analysis chart. While we may not have as strong browsing, searching, and filtering capabilities as our competitors, we will focus on our niche target demographic of SFSU students, while still delivering the capability to browse, filter, and search through our content.

7. High-Level System Architecture

Server Host: Heroku 1vCPU 512 MB RAM

Operating System: Ubuntu 18.04 Server

Database: PostgreSQL 10.1

Web Server: Node 8.0.0

Server-Side Language: Nodejs

Additional Technologies:

Web Framework: Expressjs

Text Editor: Atom

Web Analytics: Google Analytics

ReactJs: 16.5

jQuery: 3.3.1

pg-promise: 8.4.6

Styled-components: 3.4.9

Bootstrap: 4.1.3

8. Team

- James Andrews - Team Lead
- Ivan Varela - Front-End Lead
- Shirin Namiranian- Front-End Developer
- Kenneth Surban - Front-End Developer
- Ian Dennis - Back-End Lead
- Liyao Jiang - Back-End Developer
- Chris Johansen- 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 and working on it - **DONE**
- Team lead ensured that all team members read the final M1 and agree/understand it before submission - **DONE**