SW Engineering CSC648-848 Spring 2025

RedJr: SFSU Marketplace

Ria Thakker (Team Lead, rthakker@sfsu.edu), Rohith Gannoju (Backend Lead), Danny Duenas (Frontend Lead), James Richards (GitHub Master), Eric/Po-Han Chen

Milestone 1

History

Date Submitted	03/12/2025
Date Revised	TBD

01. Executive Summary

In today's fast-paced university environment, students and faculty often struggle to find affordable and convenient ways to buy and sell essential items like textbooks, electronics, furniture, and learning materials. While general online marketplaces exist, they are not tailored to students, and can have unrelated listings, scams, and unreliable sellers. RedJr is a platform designed specifically for the San Francisco State University (SFSU) community, offering an easy-to-use and trusted marketplace where students and faculty can connect to buy and sell items.

RedJr provides essential e-commerce functionalities, including item browsing, keyword-based searching, and direct communication between buyers and sellers. Sellers can effortlessly upload product listings and manage them through a personalized dashboard, while buyers can explore verified items with confidence. A key differentiator of our platform is its high degree of safety due to SFSU-only verification and administrator monitoring of all content. To further distinguish itself from existing platforms, RedJr features student-specific tools such as the ability to rent items from other members of the SFSU community, the ability to set up sales on campus at specific locations, and a rating system for both buyers and sellers to provide feedback on their experiences.

RedJr is a project built by SFSU students for the SFSU community. It's developed by a passionate team: Ria Thakker (Team Lead), Rohith Gannoju (Backend Lead), Danny Duenas (Frontend Lead), James Richards (GitHub Master), and Eric Chen (Frontend Contributor). As students ourselves, we are familiar with the challenges faced by college students. Our mission is to reduce waste by promoting selling and giving away old items and to provide an affordable, reliable solution for buying and selling needs to the campus community.

02. Personae

Persona 1

Griffey is a third-year student at San Francisco State University who considers himself financially and tech-savvy, as well as organized. He is competent when it comes to websites and can usually figure out the proper navigation of it if the website is structured properly.
Griffey loves making extra money whenever possible and always looks for ways to get the best value for his items. When selling he is usually willing to let an item sit until it goes for the price listed.
A pain point of his is that it is the end of the semester and he needs to sell his dorm items quickly before moving out. He doesn't want to deal with the hassle of storing or transporting large furniture and electronics. Also Griffey is currently without a vehicle and does not want to use public transportation to take large item to meet with a prospective buyer

The goal of using our app to list his items for sell is that because the app of for students attending his college it is likely that there are students in need of similar items he owns and being in close proximity they can purchase and retrieve the item in a quick fashion.

Persona 2

Emily is an 18-year-old freshman at San Francisco State University, majoring in Business Administration. She recently moved from Seattle, Washington, and is adjusting to college life and living independently in a campus dorm. While excited about this new chapter, she quickly realizes her dorm lacks essential furniture like a desk, chair, and shelves, making it difficult to study and stay organized. She also wants to add personality to her space without overspending. Emily is tech-savvy and frequently shops on Amazon, Facebook Marketplace, and Depop, but she prefers simple, easy-to-navigate websites. She needs a reliable and affordable way to buy second-hand furniture and textbooks from other students to stay within her budget. Without a car, she requires a platform that allows her to coordinate safe, on campus meetups for item pickup. Her main goals include finding reasonably priced textbooks and furniture through a trusted SFSU exclusive marketplace that ensures legitimate listings and safe transactions. However, she has concerns about scams, misleading posts, and the inability to inspect items before purchase. Transportation limitations also make it difficult for her to pick up larger items or travel far for transactions.

Persona 3

Mike is a second-year computer science student at SFSU who is an administrator for the RedJr site. He is a developer, so is tech-savvy and intimate with the details of the website, its structure, and has access to internal documentation that will assist him with his job.
Mike's job is to go through all new listings that are posted to the site and approve them for public viewing. One of his pain points is that it is tedious to manually view and approve each post, so he wants the process to be as fast and uncomplicated as possible.

Persona 4

Background:
Olivia is a 20-year-old junior at SFSU majoring in Environmental Studies. Passionate about sustainability, she actively participates in campus recycling initiatives. She frequently buys and sells used textbooks, electronics, and everyday items to promote eco-friendly practices and save money.

Goals:

- Find quality second-hand items while reducing waste.
- Earn extra cash by selling pre-owned belongings.
- Use a secure, faculty-supported platform to ensure trustworthy transactions with fellow students.

Pain Points:

- Chaotic, unmoderated online marketplaces leading to unreliable deals.
- Difficulty assessing the true condition of items without trusted reviews.
- Risk of scams and miscommunications during exchanges.

Persona 5

Jake is a senior Computer Science student at San Francisco State University who loves making extra money through buying and selling items. He frequently purchases underpriced goods, refurbishes them, and resells them at a higher price. Always on the lookout for deals, he refreshes the website constantly to catch new listings before others.

Jake values a smooth selling process and prefers a clean, intuitive interface. While he usually waits for the best offer, he sometimes needs to sell items quickly, especially as graduation approaches. A major pain point for him is dealing with unreliable buyers who back out last minute.

His goal is to efficiently buy and sell within the SFSU community, avoiding the hassle of external marketplaces while ensuring quick and profitable transactions.

03. Use Cases

Use Case 1 - Selling Items

Griffey is a student **seller** on the website because the semester is over and he needs to move out of his dorm. He is looking to list large items that he does not wish to take with him for sale to San Francisco State **buyers**. After arriving at the homepage and logging into his personal account, he navigates to his dashboard and visits the **Listings** page, which displays the items he has listed for sale.
On this page, he clicks **New Listing** and enters the item details, including:

- Item Name
- Category (Furniture, Tech, Clothing, etc.)
- Brief Description
- Condition (Brand New, Used but Like New, Some wearing, etc.)
- Photos of the Item
- Asking Price

After entering all the required information, he clicks **Publish** and the item is now listed for sale on the site. To keep track of buyers interested in his items, Griffey can visit the listings page. If a buyer has messaged him about an item, the listing will display the number of interested buyers. Clicking on this number takes him to the **messages** page, where he can view messages related to that particular item. Griffey can also go directly to the Messages page from the **navigation bar**, where he can see all messages he has received, including ones from potential buyers. If Griffey

and a buyer agree on a price and pickup location through messaging, he marks the item as **sold** so that no other buyers can message about it.

Use Case 2 - Buying Items

Emily recently moved into an SFSU dorm and needs a desk and chair for her room. She wants to find a second-hand setfrom another student instead of buying new.

1. Emily logs into the SFSU Marketplace website using her student email login.
2. She browses listings and selects the "Furniture" category.
3. She uses filters to narrow her search by price range and location (prefers sellers on campus).
4. She finds a listing for a desk and chair that fits her budget.
5. She clicks the listing to view photos, description, and seller details.
6. Emily uses the built-in messaging system to ask the seller about the furniture's condition and if it's still available.
7. The seller responds, and they agree to meet at a designated on-campus location.
8. Emily meets the seller, checks the condition of the furniture, and decides to buy it.
9. She pays via Venmo or cash, takes the items, and the seller marks the listing as "sold".

Use Case 3 - Administrator Functions

Mike logs in to his administrator account on the site and views his admin dashboard, where he is able to use the navigation bar to view all the item listings that are pending approval. At the end of each item listing, he is able to approve or deny any listing. If a listing is denied, he gives feedback to the seller and a reason for the denial. Items that are approved get removed from view and Mike is able to approve or deny the next item on the list.
Once he is done approving new public listings, Mike is able to navigate to his messages where he can respond to support queries and review chat logs for appropriate content / reliability if he chooses. If he sees something that is not appropriate or a violation of the site terms, he is able to report or ban a user for a specified period of time or permanently.

Use Case 4 - Creating Account

Steps:

1. Discovery: Olivia notices a flyer on campus promoting the marketplace and hears positive feedback from her peers.
2. Registration: She visits the marketplace website, registers using her university email, and creates her profile.
3. Interaction: Olivia browses listings for used textbooks and gadgets, then contacts a seller through the platform's messaging system to inquire about an item's condition and negotiate a fair price.

4. Transaction: They agree on a price and arrange a convenient campus meetup to exchange the item.
5. Feedback: After a smooth transaction, Olivia leaves a review for the seller, reinforcing community trust and encouraging high-quality listings.

Use Case 5 - Getting updated listings

Jake logs into his account, navigates to his **seller dashboard**, and visits the **Listings** page. He clicks **New Listing**, enters details (name, category, description, condition, photos, price), and publishes it.

To track interest, he checks the number of interested buyers on his listings and accesses the **Messages** page to negotiate. Once a deal is made, he marks the item as **Sold** to prevent further inquiries.

At the same time, Jake refreshes the site frequently to catch new listings. He buys low-priced items, refurbishes them, and relists them at a higher price, ensuring steady profits while making transactions quick and convenient.

04. Data Items & Entities

User	**Description**
Administrator	User responsible for system maintenance, views a different UI compared to other users
Buyer	A user that has made an offer on an item and can rent listings
Seller	A user that has items up for sale or has put a listing up for rent

Entities	**Description**
item	Can be listed and bought from user has description of what it is
offer	Bid on an item that's up to be sold
Rental item	An item that is only being offered for a

	temporary time then returned to seller
Sale	A pre-arranged meetup for buyers and sellers to exchange items and payment. Based on the outcome/fulfillment of both parties, the users' ratings are impacted.
Seller Rating	Score out of 5 buyers give based on quality of transaction process
Buyer Rating	Score out of 5 sellers give based on quality of transaction process

Data Structure	**Description**
Messaging	Log of chats sent between buyer and seller users
Locations	List of on-campus areas that a sale/pickup is able to be fulfilled
Items bought/sold	Structure contains list of items a user has purchased and sold on the site

05. Functional Requirements

1. Unregistered users shall only be able to access the register/login/about pages.
2. To register, a user shall verify their sfsu.edu email.
3. To register, a user shall specify a school-appropriate username of their choice.
4. To register, a user shall create a password of 8 characters that contains at least 1 special character.
5. A registered user shall be able to browse and search for items listed on the website and descriptions of the items.
6. A registered user shall be able to upload item listings and descriptions for sale.
7. A registered user shall be able to make an offer to purchase an item of their choice.
8. A registered user shall be able to edit their item listings after uploading them.
9. A registered user shall be able to edit an offer they have made on an item.

10. A registered user shall be able to cancel an item listing and remove it from public view even if it is not sold.
11. A registered user shall be able to cancel an offer on an item at any time before purchase is finalized.
12. A registered user shall be able to message another registered user.
13. A registered user shall have a dashboard for their posted items and view messages from prospective buyers.
14. Registered users who have agreed to a purchase shall be able to create a sale that contains the meeting location on campus and time.
15. A registered user who has sold an item shall be able to provide feedback and contribute to the rating of the user who has bought their item after a sale.
16. A registered user who has bought an item shall be able to provide feedback and contribute to the rating of the user who has sold the item after a sale.
17. A registered user shall be able to mark items as sold and remove them from public view after a sale is finalized.
18. A registered user shall be able to report an item to an administrator if they feel the item violates a term or condition of the website.
19. A site administrator shall be able to approve all items uploaded to the site before they are available for public view.
20. A site administrator shall be able to delete items deemed inappropriate from public view.
21. A site administrator shall be able to delete/ban users that have been reported, violating platform policies.
22. A site administrator shall be able to view the chat logs of registered users.

06. Non-functional requirements

1. Application shall be developed, tested and deployed using tools and cloud servers approved by Class CTO and as agreed in M0
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. All or selected application functions shall render well on mobile devices (no native app to be developed)
4. Posting of sales information and messaging to sellers shall be limited only to SFSU students
5. Critical data shall be stored in the database on the team's deployment server.
6. No more than 50 concurrent users shall be accessing the application at any time
7. Privacy of users shall be protected
8. The language used shall be English (no localization needed)
9. Application shall be very easy to use and intuitive
10. Application shall follow established architecture patterns
11. Application code and its repository shall be easy to inspect and maintain

12. . Google analytics shall be used
13. No e-mail clients or chat services shall be allowed. Interested users can only message to sellers via in-site messaging. One round of messaging (from user to seller) is enough for this application
14. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
15. Site security: basic best practices shall be applied (as covered in the class) for main data items
16. Media formats shall be standard as used in the market today
17. Modern SE processes and tools shall be used as specified in the class, including collaborative and continuous SW development and GenAI tools
18. The application UI (WWW and mobile) shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2025. For Demonstration Only" at the top of the WWW page Nav bar.

07. Competitive Analysis

Features	facebook Marketplace	craigslist	Offerup	LetGo	RedJr
User Verification	+	-	+	+	++
User Interface	+	+	+	+	++
Internal Messaging	+	-	-	+	++
Admin Communication and customer support	+	+	-	+	++
Community Integration	-	-	-	-	++
Safety & trust	+	-	+	+	++

+ **Has feature, - feature not included, ++ superior**

Analysis Summary

Craigslist lacks an internal messaging system, requiring users to communicate via email, whereas platforms like Facebook Marketplace, OfferUp, and Letgo provide built-in messaging

for convenience. Our website will include an internal messaging system to streamline communication between users. Also, craigslist design is very minimal compared to Facebook Marketplace and OfferUp's more modern interfaces, which require photos for listings and offer a smoother user experience. A common complaint among Craigslist users is the inability to edit listings, an issue we aim to resolve. User verification is a major weakness on Craigslist, Facebook Marketplace, OfferUp, and Letgo, as they only require an email for registration, allowing scammers to create temporary or fake accounts. For our website, we will require an official SFSU email, ensuring that each user is a verified student or faculty member, reducing fake accounts and scams. Because our website is only SFSU students or faculty, we can have more responsive admin support, unlike larger platforms where direct communication with administrators is often slow or ineffective. Unlike these broader marketplaces, our site is specifically designed for the SFSU community, allowing us to adjust features to user needs and provide a safer, more efficient experience.

08. High-level System Architecture & Technologies Used

- Server Host: Amazon Web Services
- Operating System: Ubuntu 16.04 Server
- Database: MySQL v 8.0.41
- Web Server: Apache2
- Server-Side Language: Python
- Additional Technologies: Web Framework: Flask
    - IDE: JetBrains PyCharm, Visual Studio Code
    - Web Analytics: Google Analytics
    - SSL Cert: Let's Encrypt (Cert Bot)
    - HTML, CSS, Javascript (frontend)
- Browsers Supported: Google Chrome 134.0, Firefox 136.0

09. Use of GenAI Tools

- Tool used: ChatGPT-4-Turbo
- Tasks used:
    - Executive Summary: HIGH usefulness. Streamlined the writing process, and only minor modifications were needed to increase brevity and specificity with regards to the SFSU-related functions and some requirements. I didn't have to spend more than 15 minutes on the executive summary because its content was useful on the first go-around.
    - Personae, Use Cases: MEDIUM usefulness. ChatGPT provides a solid starting point for defining user personas and creating structured use cases. However, some personas fell short in addressing the specific challenges faced by SFSU students, so I had to refine the descriptions.

- Example prompts:
    - Can you generate an executive summary for a project with the following requirements [Copied & pasted functional requirements from project overview, excluded for space]

      The executive summary must contain the following: [Description of executive summary from M1 overview, excluded for space]

10. Team & Roles

Team Member	**Email**	**Role**
Ria Thakker	rthakker@sfsu.edu	Team Lead
Rohith Gannoju	rgannoju@sfsu.edu	Backend Lead
Danny Duenas	dduenas@sfsu.edu	Frontend Lead
James Richards	jrichards@sfsu.edu	GitHub Master
Eric/Po-Han Chen	pchen@sfsu.edu	Frontend Contributor

11. Team Lead Checklist

- So far, all team members are fully engaged and attending team sessions when required
    - DONE
- Team found a time slot to meet outside of the class
    - DONE - Fridays 3pm - 4/5pm
- 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
    - ON TRACK
- Team reviewed class slides on requirements and use cases before drafting Milestone 1
    - DONE
- Team reviewed non-functional requirements from “How to start…” document and developed Milestone 1 consistently
    - DONE
- Team lead checked Milestone 1 document for quality, completeness, formatting and compliance with instructions before the submission
    - DONE
- Team lead ensured that all team members read the final M1 and agree/understand it before submission

    - DONE
- Team shared and discussed experience with GenAI tools among themselves
    - DONE
- GitHub organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)
    - ON TRACK