

1. Executive Summary:

Project name: SFSU Access

In this project, we want to provide SFSU students and faculties a safe, easy, and convenient place to purchase, exchange, and share media contents. At this time, everyone has easy access to at least one electronic device, and people create numerous amounts of media content everyday. These content could have many uses, and the ability to access other people's content opens so much more creative possibilities.

The main function of the project called SFSU Access is providing a platform for SFSU students and faculties to buy, exchange, and share media contents. The platform enforces membership login to prevent inappropriate postings. Members can post their content and other members' contents.

The platform is exclusive for SFSU students and faculty. Targeting a specific user demographic allows us to address the specific needs of the users better, and be more efficient in marketing.

Our team consists of six SFSU CS students. Three of them are on the Back end team with a very experienced leader and git master. Other three are on the Front end team also with a strong leader. Our team is familiar with the technical skills that are being used in this project, hence we are confident about delivering the project on time.

2. Personae and main Use Cases:

Jimmy, a junior student pursuing a Bachelor of Arts in History and works part-time	
<ul style="list-style-type: none">• Has to manage his time between work and school.• Not very good at using online shopping app• Has limited budget for learning material• Not patient with online registration• Since the new semester just started, he wants to get learning materials for his classes as soon as possible	Image here

<ul style="list-style-type: none"> • Wants to buy second-handed books at a lower price • Looking for more class-related materials such as PDF or IMG. 	

Alex – A Senior student who is graduating soon, and currently got an internship at a company	
<p>About Alex:</p> <ul style="list-style-type: none"> · Senior student in SFSU · Has lots of used materials form different classes · Likes online gaming, social media, shopping · Has experience of using online shopping system <p>Goals:</p> <ul style="list-style-type: none"> · Wants to help new students · Wants to make some money with those used materials · Not want to pay shipping fee 	Image here

Katie – A computer science

About Katie:

- Has an excess of digital photographs taken over the years
- Doesn't have any experience with e-commerce
- Inspiring photographer

Goals:

- Boost her portfolio by sharing her work
- Wants to reach an audience that can easily contact her if needed
- Creates connections with those interested in her line of passion

Image here

Use Case 1: Buying second-hand PDF documents

Jimmy enrolled in Greek Mythology in the Spring semester. He was informed that a specific textbook was needed to sufficiently learn and pass the course. Jimmy is already spending his hard earned money from his job on other courses, so he didn't have enough to purchase the textbook until later down the road. Thankfully, he remembered to search on SFSU Access to see if someone was listing a PDF version of the textbook up for sale. He uses the search option to look up the name of the textbook and finds a listing for a decent price. Opening up the listing, he uses the messaging feature to send a message to the seller to inquire about the book. Within a few hours, Jimmy receives a response and was able to confirm how the transaction would work. Jimmy was able to begin reading his new textbook within the day.

Use Case 2: Selling old classwork and notes

Alex is a senior graduating this May. Throughout his years at SF State, he has amassed quite a lot of class materials, including hand-written notes, worksheets, test prep materials, and more. Being adept with the latest technology, he was able to scan all his documents into PDFs and images to list on SFSU Access. He logged in to his account and added each document along with item information, such as pricing and descriptions. Alex is able to make some extra cash from his sales and help fellow and future students with any similar studies.

Use Case 3: Department Faculty case

Use Case 4: Admin case

3. List of main data items and entities

Non-registered users - they can view, search and see details of all the posted items.

Registered users - have all privileges as non-registered users and can post items and send messages.

Administrator - has all privileges as registered users and can approve any item for posting or reject any inappropriate item.

4. Initial list of functional requirements :

Non-registered users

1. The current listing provided by registered users shall be made viewable by non-registered users.
2. Non-registered users shall not engage with any listings until registration has been completed.
3. Non-registered users shall create an account to engage in activity on the service. Registration shall require an SFSU email address.

Registered users

1. Posting items to the store shall be subject to Admin approval.
2. After an item is no longer available, the user shall mark the item as these attributes.
3. A user shall wish to no longer engage in business within the service. They can use the delete user account functionality to remove their account and data.

Administrator

1. Content moderation is needed to protect users. Before a user listing is presented live, it shall be under review with the administrator. The admin shall approve or deny the listing.
2. The administrator reserves the right to delete any post at his or her discretion. This functionality provides that ability.

5. List of non-functional requirements

1. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
2. Selected application functions must render well on mobile devices
3. Data shall be stored in the team's chosen database technology on the team's deployment server.
4. Full resolution free media shall be downloadable directly, and full resolution media for selling shall be obtained after contacting the seller/owner
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 (no localization needed)
8. Application shall be very easy to use and intuitive.

9. Google analytics shall be used
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) for main data items
13. Media formats shall be standard as used in the market today
14. Media material shall be either free or for sale, as determined by media owner
15. Each media material shall have its license info as one of the following: a) free use and modification; b) free but only allowed for SFSU related projects; c) for sale
16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
17. The website shall prominently display the following exact text on all pages *"SFSU Software Engineering Project CSC 648-848, Spring 2020. 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	Amazon	Etsy	SFSU Bookstore	SFSU Access
Messaging	+	+	-	+
Search / Filters	++	+	+	+
Organization-specific	-	-	+	++
Free Items	-	-	-	+
No Listing Fee	-	-	-	+

+ exists; ++ superior; - does not exist

SFSU Access' Competitive Relationships:

6.1) Messaging: Our website will have a messaging feature for registered users, where messages can be sent directly between the seller and buyer. Most platforms' messaging systems are similar.

6.2) Search/Filters: Any user can enter a search term to list items that similarly matches the term, and/or choose a specific category of item to filter the results further.

6.3) Organization-specific: Registering on SFSU Access is only available to San Francisco State University students and faculty, which requires an SFSU email.

6.4) Free Items: Registered users can list items for no price so that others can download their content without payment. No other platform allows this.

6.5) No Listing Fee: SFSU Access does not apply a listing fee nor take a cut of any purchase made. No other platform provides their sellers the full amount they sell for.

7. High-level system architecture and technologies used:

Server Host: AWS

Operating System: Ubuntu (

Database: MySQL

Webserver: Apache

Server-side language: Python

Web framework: React.js

IDE: VScode, intelli j, pycharm

Web Analytics: Google Analytics

8. Team and roles:.

Team leader: Yan

Front team leader: JunMinLi

Front-end team: JunMinLi, Cody and Yan

Back team leader: Kevin Luong

Back-end team: Kevin Luong, Aitor and David

Github master: Kevin Luong

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 **ON TRACK**
- 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**

Objective:

Based on the brief high level project description (posted on iLearn class site) the objective of Milestone 1 (M1) is to:

Develop initial high-level personas and use cases;

From use cases develop high level functional requirements for the application,

List high level architecture, frameworks and tools to be used (generally the same as in M0), and

Get the teamwork going including role assignments (back end, front end leads, github mater)

Learn how to write team reports (one report for the whole team, with participation of all team members)

Note that these are only high-level requirements and specs with the idea to get early feedback and iterate before investing in developing more detailed specs and first prototype in Milestone 2. Future designs can deviate from Milestone 1 in the spirit of iterative SW design and development.

Initial input for your work is the final project high-level description (posted on iLearn), class slides on the topics as well as your SW and tool selection for M0. For use cases and functional specs, feel free to also use your own ideas, research similar applications that already exist, talk to your friends etc. Please consult class material on Use Cases, Requirements and Specs.

This is the first team milestone. The whole student team submits one milestone document for each Milestone 1 – 5, submission details are below.

You will discuss ongoing work on Milestone 1 during team session in each class and you can also send e-mail to instructors with questions.

Expected size of this document is about 7-10 pages, using font and spacing as in this document.

*** Please wrap up M0 before you start on M1 ****

Content and structure for Milestone 1 document for review:

In the document for Milestone 1 (M1) you must have ALL of the following subsections in exact order as below (have a separate numbered section for each) in one PDF file.

1.Executive Summary: Short description of the final product/application and its key advantages, novelty, value (up to 1 page). Make it as an executive summary – think of answering the question of why we should fund this project. We suggest you assign a name to your project for easier reference and good ‘marketing’. This summary should be readable to a general manager/executive that is not a CS specialist and is used to explain and also to advertise/promote your project. Typical outline is: one paragraph on the motivation and importance of the application you are developing, followed by a paragraph on what functions and services your application will be providing and how it helps the users (high level only, no jargon) and optionally what is unique and special in your design. At the end say in one paragraph something about your team (e.g. about your student startup team...).

Project name: SFSU Access

Motivation and importance:

In this project, we want to provide SFSU students and faculties a safe, easy, and convenient place to purchase, exchange, and share media contents. At this time, everyone has easy access to at least one electronic device, and people create numerous amounts of media content everyday. These content could have many uses, and the ability to access other people’s content opens so much more creative possibilities.

Functions and services:

The main function of the project is providing a platform for SFSU students and faculties to buy, exchange, and share media contents. The platform enforces membership login to prevent inappropriate postings. Members can post their content and other members' contents.

Unique and special:

The platform is exclusive for SFSU students and faculty. Targeting a specific user demographic allows us to address the specific needs of the users better, and be more efficient in marketing.

About my team:

My team consists of six SFSU CS students. Three of them are on the Back end team with a very experienced leader and git master. Other three are on the Front end team also with a strong leader. Our team is familiar with the technical skills that are being used in this project, hence we are confident about delivering the project on time.

2. Personae and main Use Cases: Summarize key personas (categories of users) for your application – their general characteristics, goals, skills, pain points related to the application you are developing. About 1/3 of a page per persona – see class notes. (Note: in personas you stay general, in use cases you say how personas will use your app (at high level)). Then provide 4-5 main use cases (one paragraphs for each use case) - see class notes on more detailed format for requirements. Focus only on main use cases. Simple text format is OK and preferable – tell us a story about who and how the application is used. Focus on WHAT users do, their skill level, not on HOW is the SW implemented. NOTE: avoid specific on HOW functions will be done and text resembling user manual: this is supposed to guide the design of the future product and is NOT a description of how the product will work (you don't know that yet) – see class slides for details. Please assign a descriptive title and number to each use case so it can be tracked. Best is to first show all personae then all use cases, do not mix them up.

Persona: Jimmy – A junior student and part time worker

Alex – A Senior student who is graduating soon, and currently got an internship at a company

Katie – A Sophomore photography student that works a full-time retail job

About Jimmy:

- Has to manage his time between work and school.
- Not very good at using online shopping app
- Has limited budget for learning material
- Not patient with online registration

Goals:

- Since the new semester just started, he wants to get learning materials for his classes as soon as possible.
- Wants to buy second-handed books with a lower price.
- Looking for more class-related materials such as PDF or IMG.

About Alex:

- Senior student in SFSU
- Has lots of used materials from different classes
- Like online gaming, social media, shopping
- Have experience of using online shopping system

Goals:

- Want to help new students
- Wants to make some money with those used materials
- Not want to pay shipping fee

About Katie:

- Has an excess of digital photographs taken over the years
- Doesn't have any experience with e-commerce

- Inspiring photographer

Goals:

- Boost her portfolio by sharing her work
- Wants to reach an audience that can easily contact her if needed
- Create connections with those interested in her line of passion

Use Case 1: Buying second-hand PDF documents

Jimmy enrolled in Greek Mythology in the Spring semester. He was informed that a specific textbook was needed to sufficiently learn and pass the course. Jimmy is already spending his hard earned money from his job on other courses, so he didn't have enough to purchase the textbook until later down the road. Thankfully, he remembered to search on [our website name here] to see if someone was listing a PDF version of the textbook up for sale. He uses the search option to look up the name of the textbook and finds a listing for a decent price. Opening up the listing, he uses the messaging feature to send a message to the seller to inquire about the book. Within a few hours, Jimmy receives a response and was able to confirm how the transaction would work. Jimmy was able to begin reading his new textbook within the day.

Use Case 2: Selling old classwork and notes

Alex is a senior graduating this May. Throughout his years at SF State, he has amassed quite a lot of class materials, including hand-written notes, worksheets, test prep materials, and more. Being adept with the latest technology, he was able to scan all his documents into PDFs and images to list on [our website name here]. He logged in to his account and added each document along with item information, such as pricing and descriptions. Alex is able to make some extra cash from his sales and help fellow and future students with any similar studies.

Use Case 3:

3. List of main data items and entities - define main terms, types of users, data structures and "items" or "entities" at high or logical (not implementation) level (e.g. name, meaning, usage,

and NOT how the data is stored in memory) so it is easier to refer to them in the document. Focus on key terms (main data elements/records used in your app, types of users and their privileges etc. These terms and their names must be used consistently from then on in all documents, user interface, in naming SE components and database elements etc. In cases where you attach behavior and privileges to data items (e.g. user types) that also drives the design of the SW. In later milestones you will add more implementation details for each item. You will later expand this section with more details. This will help define planning and design for the DB for example.

Non-registered users - they can view, search and see details of all the posted items.

Registered users - has all privileges as non-registered users and can post items and send messages.

Administrator - has all privileges as registered users and can approve any item for posting or reject any inappropriate item.

4. Initial list of functional requirements – see class notes. This refers to high level functions you plan to develop to the best of your knowledge at this point. Focus on WHAT and not HOW. Keep the user in mind. Develop these functions to be consistent with use cases and requirements above. Number each requirement with unique numeric value and use these numbers consistently from then on. For each functional requirement use 1-3 line description. At this stage no need to prioritize the requirements, collect all ideas from the team. We are looking for 20 or so requirements.

Kevin Luong

Non-registered users

4. View store
 - a. The current listing provided by registered users will be made viewable by non-registered users. Non-registered users may not engage with any listings until registration has been completed.
5. Create user account
 - a. Non-registered users may create an account to engage in activity on the service. Registration will require an SFSU email address

Registered users

4. Post item to store
 - a. Posting items to the store will be subject to Admin approval.
5. Mark as Sold/Unavailable/OutOfStock
 - a. After an item is no longer available, the user may mark the item as these attributes.
6. Delete user account
 - a. A user may wish to no longer engage in business within the service. They can use the delete user account functionality to remove their account and data.

Administrator

3. Approve post
 - a. Content moderation is needed to protect users. Before a user listing is presented live, it will be under review with the administrator. The admin may approve or deny the listing.
4. Delete post
 - a. The administrator reserves the right to delete any post at his or her discretion. This functionality provides that ability.

5. List of non-functional requirements (performance, expected load, security requirements, storage, availability, fault tolerance...). Note that mandatory high level non-functional specs are given in high level document, so for M1 – M5 we request you simply copy them from high level document from iLearn. Please observe and adhere to these non-functional requirements in your design and development from now on – you are not allowed to change them unless you get permission.

18. 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).
19. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
20. Selected application functions must render well on mobile devices
21. Data shall be stored in the team's chosen database technology on the team's deployment server.
22. Full resolution free media shall be downloadable directly, and full resolution media for selling shall be obtained after contacting the seller/owner
23. No more than 50 concurrent users shall be accessing the application at any time
24. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users.
25. The language used shall be English (no localization needed)

26. Application shall be very easy to use and intuitive.
27. Google analytics shall be used
28. No e-mail clients shall be allowed
29. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
30. Site security: basic best practices shall be applied (as covered in the class) for main data items
31. Media formats shall be standard as used in the market today
32. Media material shall be either free or for sale, as determined by media owner
33. Each media material shall have its license info as one of the following: a) free use and modification; b) free but only allowed for SFSU related projects; c) for sale
34. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
35. The website shall prominently display the following exact text on all pages *"SFSU Software Engineering Project CSC 648-848, Spring 2020. For Demonstration Only"* at the top of the WWW page. (Important so as to not confuse this with a real application).

6.Competitive analysis: Find 3-4 competitive products. Present competitors' features vs. your planned ones. First, create a table with key features of competitors vs. yours planed, only very high level, 5-6 entries max (as shown in the class). After the table, you must summarize in one paragraph what are the planned advantages or competitive relationship of your planned product to what is already available. In the table clearly mark your product, e.g. shade its column/data.

Features	Amazon	Etsy	SFSU Bookstore	Our Website
Messaging	+	+	-	+
Search / Filters	++	+	+	+
Organization-specific	-	-	+	++
Free Items	-	-	-	+
No Listing Fee				

+ exists; ++ superior; - does not exist

7.High-level system architecture and technologies used: Briefly provide itemized list of all main SW components such as frameworks, APIs, tools and systems to be used, supported browsers and deployment platform (SW and server) to be used. This list is to be the list of approved tools and systems from M0 (which may be the list you have modified during or after M0 – but get it approved). Any other external code/API/tool must be approved by instructors and you have to justify it.

Server Host: AWS

Operating System: Ubuntu (

Database: MySQL

Webserver: Apache

Server-side language: Python

Web framework: React.js

IDE: VScode, intelli j, pycharm

Web Analytics: Google Analytics

8.Team and roles: list student names, mark their roles (team leader, front and back team lead and github master, document master (optional but recommended, can be team lead), team member front end, team member back end etc.).

Team leader: Yan

Front team leader: JunMinLi

Front-end team: JunMinLi, Cody and Yan

Back team leader: Kevin Luong

Back-end team: Kevin Luong, Aitor and David

Github master: Kevin Luong

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 are working on learning and practicing **ON TRACK**
- 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**

Submission for Milestone 1 document for review – you must follow the instructions below **PRECISELY**:

Teams must collaborate in creating M1 documents by having working M1 documents on their team Github private repository (similar to managing code) in “Milestones” folder so all team members and instructors can access it.

We strongly suggest the following collaborative approach for creation and completion of M1 document (NOTE: creating a team document is similar to creating a code by the

team of programmers):

- Team lead assigns M1 editor (often this can be done by the team lead)
- Team lead/M1 editor assign individual chapters to team members
- M1 editor collect chapters, edits/corrects then integrates them into a well formatted document (with same font and formats)
- M1 editor posts final candidate full document on team repo so that all team members read full document for one more review and any feedback

- M1 editor completes the final version as per feedback

- Team lead submits M1 info for review as per submission instructions.

Submission instructions (below) must be followed precisely and completely or grade penalty will be imposed

The whole student team submits one milestone document for M1, as follows: Team leads will send e-mail with a link (NOT the attached file) pointing directly to M1 Document and not the folder (in PDF format) placed in github folder called “Milestones” to Petkovic@sfsu.edu and CTO Anthony, with e-mail subject line as specified below.

Submission e-mail subject line: MUST be “CSC648-848 Spring 2020 Milestone1 Team N” in the subject line (N is a team number 01, 02...).

e-mail body is to contain brief courtesy text and direct link to actual Milestone 1 document in PDF stored in github Milestones folder

Milestone 1 file name: MUST be

CSC648-848 Spring 2020 Milestone1 Team N.PDF (N is your team number) (We use only PDF so I can send you feedback as yellow sticky notes).

Submission must be done by the deadline specified, any extension has to be approved at least 24 h ahead of the deadline.

M1 document format and structure

Title page MUST include (nicely formatted, larger font please) –“SW Engineering CSC648/848 Spring 2020”

Project/application title and name (you can use the name you chose for your application)

- Team number
- Names of students (team lead first) with e-mail of team lead. Please mark those who are team lead, front end, back end leads and github master
- “Milestone 1”
- Date
- History table (revisions) (Note: you will update this document based on instructors’ feedback so this is important. Generally it should contain two entries: date submitted/ and date revised after instructor comments)

The rest of the document has to contain ALL sections as described above under “Content and structure for Milestone 1 document for review”

Team leads and M1 editors: make sure document is well formatted, reads well, is complete, and looks professional. This will be part of your portfolio and will influence

the grade. Make sure all team members read final version and give comments before submission.

Note that you can use Google docs or similar for creating and editing the document but then we want you to post the document on github so all your code and documentation is there to serve as a portfolio and to allow easy access to all.

Instructor's feedback and creating final Milestone document for Final Project delivery

In the course of developing M1 you can ask instructor questions via e-mail and during team session in the last hour of the class. Upon submission of M1 you will get feedback from instructors by any of the following: e-mail, markings on your document and in class during team meetings. This feedback must be analyzed and taken into account by your team in order to revise your M1 and this must be used subsequently for the rest of the project. Please enter the revision summary in history table.

Instructors will comment from the standpoint of CEO, VP of Marketing (who translates customer and marketing requirements) and CTO (Architecture etc.). You may choose not to agree with the comments. This is OK as long as you justify this and are prepared to live with that design and deliver it. In some cases, instructors may insist on some features or decisions.

Upon getting instructors' feedback on your questions and submitted document, you need to revise your first draft, freeze it (meaning no more changes on this document even if future design changes) and use it as a basis for developing Milestone 2 (M2). The frozen document M1 will be submitted as part of final project delivery in Milestone 5. Do not start working on M2 before you get feedback on M1 and make sure all team members read frozen M1 document.

Future M2 functions and actual SW app may differ from what you proposed in M1, that is normal and in fact expected to happen in the spirit of iterative SW development. In that case there is no need to modify frozen M1 document.

Evaluation and grading

We will grade only each "frozen" milestone when it is submitted with final project at the end of the class ("Milestone 5 folder"), after it has been modified for instructors' feedback. Note that instructors' feedback is NOT graded in order to encourage interaction.

Milestones improperly submitted (e.g. not following ALL required submission rules) will first be returned, and if problems persist 10% penalty will be applied to the grading of that milestone. Only one error in submission in M1 will be "forgiven", any subsequent

problem submission in any milestone documents will be recorded with negative points as outlined under the grading rubric SE Process grade: submissions.

Milestone documents have to have all required sections or negative points will be recorded for later grading (grading rubric: SE Process grade: document quality).

Milestones have to be submitted on time and in a way as specified above. In case of justifiable reasons to delay, permission has to be obtained by e-mailing to Prof. Petkovic Petkovic@sfsu.edu 24 h prior to the deadline. Late submissions with no permission incur 10% penalty on the grading of that milestone.