

Final Project for SW Engineering Class

CSC 648-848 Spring 2020

Team 02

SFSU Access

Team lead: Yanrui Xu

Team lead email: yxu13@mail.sfsu.edu

Back end lead & Github Master: Kevin Luong

Front end lead: JunMinLi

Team members: Cody Xu, Aitor Elvira, David Lin

URL: <http://54.177.237.30>

05/21/2020

Product summary:

The name of our project is called SFSU Access which is exclusive for SFSU students and faculty. In this project, we want to provide SFSU students and faculties a safe, easy, and convenient place to purchase, exchange, and share media contents. The market function allows people to browse media contents posted by other members or themselves. The member login function allows users to post, buy, and download contents after they have logged in to their accounts. The admin function allows website administrators to manage the contents posted by users to make sure the contents are safe and appropriate.

The major committed functions are:

Non-registered users:

1. All listings provided by registered users shall be made viewable by non-registered users.
2. Users shall be able to search and browse all items
3. Users shall be able to view item details after clicking an item
4. Users shall register before being able to purchase or download items.
5. Users shall be able to create an account to engage in activity on the service. Registration shall require an SFSU email address.

Registered users:

1. Users shall be able to do everything that Non-registered users can.
2. Posting items to the store shall be subject to Admin approval.
3. Users shall be able to log in on the website.
4. Users shall be able to send and receive messages.
5. Users shall be able to post items for sale or download.
6. Users shall be able to remove their posted items.
7. Users shall be able to view all posted items and messages

8. Users shall be able to message seller using in-site message prefilled with product info

Administrator:

1. The admin shall be required to approve or deny pending listings.

Website URL: <http://54.177.237.30/>

Milestone documents – M1-M4:

SW Engineering CSC648/848 Spring 2020

M1

SFSU Access

Team 02

Team lead: Yanrui Xu

Team lead email: yxu13@mail.sfsu.edu

Back end lead & Github Master: Kevin Luong

Front end lead: JunMinLi

Team members: Cody Xu, Aitor, David Lin

Milestone 1

03/04/2020

Date submitted: 03/04/2020

History

Date Submitted	Date Revised
03/04/2020	03/12/2020

Content

Executive Summary	2
Personae and main Use Cases	3
List of main data items and entities	7
Initial list of functional requirements	8
List of non-functional requirements	9
Competitive analysis	10
High-level system architecture and technologies used	11
Team and roles	11
Checklist	11

1. Executive Summary

As technology advances, we see that everyone has easy access to at least one electronic device, and people create numerous amounts of media content everyday. These contents could have many uses, and the ability to access other people's content opens so much more creative possibilities. SFSU students and faculty need a safe, easy, and convenient place to purchase, exchange, and share media contents. To this, we created SFSU Access, an online e-commerce platform exclusively designed and specialized for SFSU students and faculty.

The main function of this project is providing an e-commerce platform for SFSU students and faculty to buy, exchange and share media contents. By targeting a specific user demographic allows us to address the specific needs of the users better, and be more efficient in marketing. Users need to provide a valid SFSU ID number to register the website, and contents could be organized according to SFSU course numbers for students to look up easily.

SFSU Access' market function allows people to browse media contents posted by other members or themselves. The member login function allows users to post, buy, and download contents after they have logged in to their accounts. The admin function allows website administrators to manage the contents posted by users to make sure the contents are safe and appropriate.

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 an experienced React 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 is a junior student pursuing a Bachelor of Arts in History and works part-time

- Jimmy has to manage his time between work and school
- Jimmy has a limited budget for his education materials, and wants to get the materials for his classes as soon as possible
- Jimmy lives far away from campus, so his commute takes up a lot of his time
- Jimmy tends to spend his free time reading a historical fiction novels
- Jimmy has basic internet skills, mostly doing online searching and shopping.



Alex is a Senior Electrical Engineering student who is graduating soon, and currently has an internship at a Fortune 1000 company

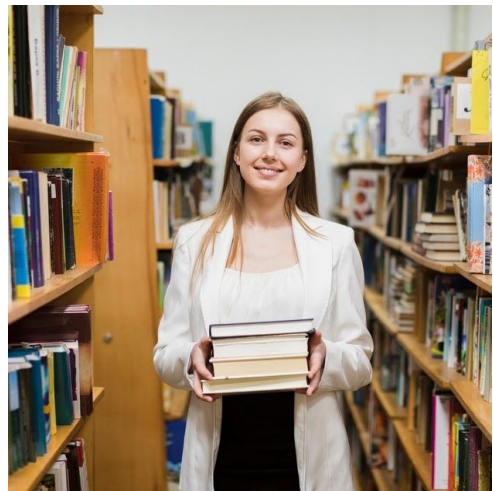
- Alex is a busy person, who currently works 32 hour weeks
- Alex enjoys helping others with their studies, often showing up in the Science labs to tutor others
- Alex is saving up his earnings to use to take care of his parents when they go into retirement
- Alex is a very focused person, usually seen taking notes in every class session
- Alex lives close by campus, so he takes public transportation in support of a better environment



- Alex is familiar with using any online services.
Such as online registration, account management,
upload and download content.

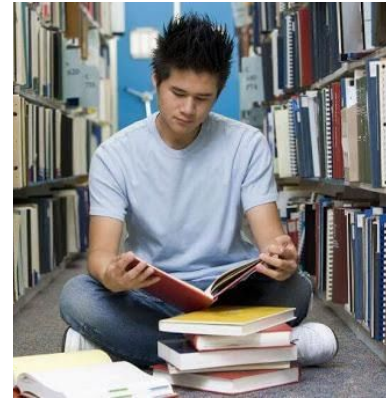
Katie is a Professor in Computer Science who has over ten years of experience in the industry

- Katie is knowledgeable with using online apps
- Katie is knowledgeable of her profession
- Katie has lots of computer science notes, PDF, documents
- Katie cares about her students
- Katie wants to share her own notes with students and inspire people who are interested in CS to join the community
- Katie is selling extra CS materials with a lower price



Kevin is a Computer science student who is graduating soon, and serves as the admin of SFSU Access.

- Kevin has a backlog of list of things to do and is very busy.
- Kevin is always looking for utilities that can allow him to operate SFSU Access more efficiently.
- While he is a Computer Science student, he is still looking for a software solution for managing his content that requires little to no effort to operate



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 couldn't remember the textbook name off the top of his head, so he just searches for his course name and number 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 class guides 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, his personal study guide, tips and tricks, and more. Being adept with the latest technology, He wants to post his materials by scanning all his documents into PDFs and images to list on SFSU Access. He was prompted to create/logged into

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 lists extra materials

Professor Katie is a professor starting her first semester at SFSU. During the first two weeks, she is eager to instruct her new students but many students have not been able to acquire professor Katie's authored textbook or class slides from the Student Bookstore due to long lines or schedule conflicts. Professor Katie seeks to also offer an alternative format for students who are unable to visit the bookstore in person. She wants to post her class slides on SFSU Access and share them with her students, in order to help them to get through the difficulties. This would benefit students and professors to stay on schedule with the course.

Use Case 4: Administrative moderation

Kevin is the server administer and content moderator of SFSU Access. As SFSU Access grows more popular, there is an increasing amount of pending posts awaiting authorization. Kevin needs an easy to use interface to view the large amount of content to approve or deny. SFSU Access provides him with a simple-to-use UI.

3. List of main data items and entities

1) Users: A user is any person that uses SFSU Access

1.1) Non-registered Users are:

- Able to view, search, and see the details of all listed items on the website
- Able to create an SFSU Access account
- Unable to list or purchase items
- Unable to message the sellers of listed items

1.2) Registered Users are:

- Able to perform anything that Non-registered Users can
- Users that have registered an account on the website
- Users that can log in on the website
- Able to create listings of new items, purchase listed items, or contact any sellers on the website when logged in

1.3) Administrators are:

- Users that have privilege to approve or deny listings submitted by Registered Users
- Unable to modify the contents of listings, such as pricing or item descriptions

2) Items:

2.1) An item is any photo, document, video, or other digital media type listed for sale by a Registered User

2.2) A listed item contains a name, price, description, any number of images, category and licensing info

3) Messages:

3.1) Interested buyers can send sellers a message through our messaging feature

3.2) Messaging can be used as a way to communicate conditions of the sale

4) Categories:

- 4.1) Users can choose to search for listed items by filtered categories
- 4.2) Each category generalizes the type of item being sold on the website
- 4.3) Sellers can select which category their items best fit in when listing their products

4. Initial list of functional requirements

Non-registered users

- 1. All listings provided by registered users shall be made viewable by non-registered users.
- 2. Users shall register before being able to purchase or download items.
- 3. Users shall be able to create an account to engage in activity on the service. Registration shall require an SFSU email address.

Registered users

- 1. Users shall be able to do everything that Non-registered users can
- 2. Posting items to the store shall be subject to Admin approval.
- 3. After an item is no longer available, the user shall mark the item as these attributes.
- 4. Users shall be able to log in on the website
- 5. Users shall be able to send and receive messages
- 6. A user shall wish to no longer engage in business within the service. They shall be able to 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	iLearn	CourseHero	Etsy	SFSU Bookstore	SFSU Access
Messaging	+	-	+	-	+
General Search	-	+	+	+	+
SFSU Search	-	-	-	+	+
Organization-specific	+	-	-	+	+
Free Items	N/A	+	-	-	++
No Listing Fee	N/A	N/A	-	N/A	+

+ exists; ++ superior; - does not exist; N/A not applicable

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. Faculties and students can search, post, and download media by SFSU course numbers.

6.4) Free Items: Registered users can list items or purchase/download for no price or payment. Except for some platforms with a content "unlock" model, 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
- Web Server: Apache
- Server-side language: Python/Flask
- Web framework: React.js
- IDE: VScode, JetBrains Suite: WebStorm + 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**

SW Engineering CSC648/848 Spring 2020

M2

SFSU Access

Team 02

Team lead: Yanrui Xu

Team lead email: yxu13@mail.sfsu.edu

Back end lead & Github Master: Kevin Luong

Front end lead: JunMinLi

Team members: Cody Xu, Aitor, David Lin

Milestone 2

03/20/2020

History

Date Submitted	Date Revised
03/20/2020	03/31/2020

Content

List of main data items and entities	2
Functional Requirements - prioritized	4
UI Mockups and Storyboards	6
High level Architecture, Database Organization	12
High Level UML Diagrams	14
Project key risks	15
Project management	17

1. List of main data items and entities

1) Users: A user is any person that uses SFSU Access

1.1) Non-registered Users are:

- Able to view, search, and see the details of all listed items on the website
- Able to create an SFSU Access account
- Unable to list or purchase items
- Unable to message the sellers of listed items

1.2) Registered Users are:

- Able to perform anything that Non-registered Users can
- Users that have registered an account on the website
- Users that can log in on the website
- Able to create listings of new items, purchase listed items, or contact any sellers on the website when logged in

1.3) Administrators are:

- Users that have privilege to approve or deny listings submitted by Registered Users
- Unable to modify the contents of listings, such as pricing or item descriptions

2) Items:

2.1) An item is any photo, document, video, or other digital media type listed for sale by a Registered User

2.2) A listed item contains a name, price, description, any number of images, category and licensing info, such as, but not limited to: All Rights Reserved, No Rights Reserved, and Creative Commons Attribution 4.0 International.

3) Messages:

3.1) A message is a private text communication between two Registered Users

3.2) Interested buyers can send sellers a message through our messaging feature

3.3) Messaging can be used as a way to communicate conditions of the sale

4) Categories:

4.1) Users can choose to search for listed items by filtered categories: Digital media, Books, Images, and Documents

4.2) Each category generalizes the type of item being sold on the website

4.3) Sellers can select which category their items best fit in when listing their products

2. Functional Requirements - prioritized

Priority 1:

Non-registered users:

1. All listings provided by registered users shall be made viewable by non-registered users.
2. Users shall be able to search and browse all items.
3. Users shall register before being able to purchase or download items.
4. Users shall be able to create an account to engage in activity on the service. Registration shall require an SFSU email address.

Registered users:

1. Users shall be able to do everything that Non-registered users can.
2. Posting items to the store shall be subject to Admin approval.
3. Users shall be able to log in on the website.
4. Users shall be able to send and receive messages.
5. Users shall be able to post items for sale or download.

Priority 2:

Registered users:

1. After an item is no longer available, the user shall mark the item as these attributes.

Administrator:

1. The admin shall be able to approve or deny pending listings within 24 hours.
2. The admin shall be able to remove listed items.
3. The admin shall be able to remove registered users from the platform.

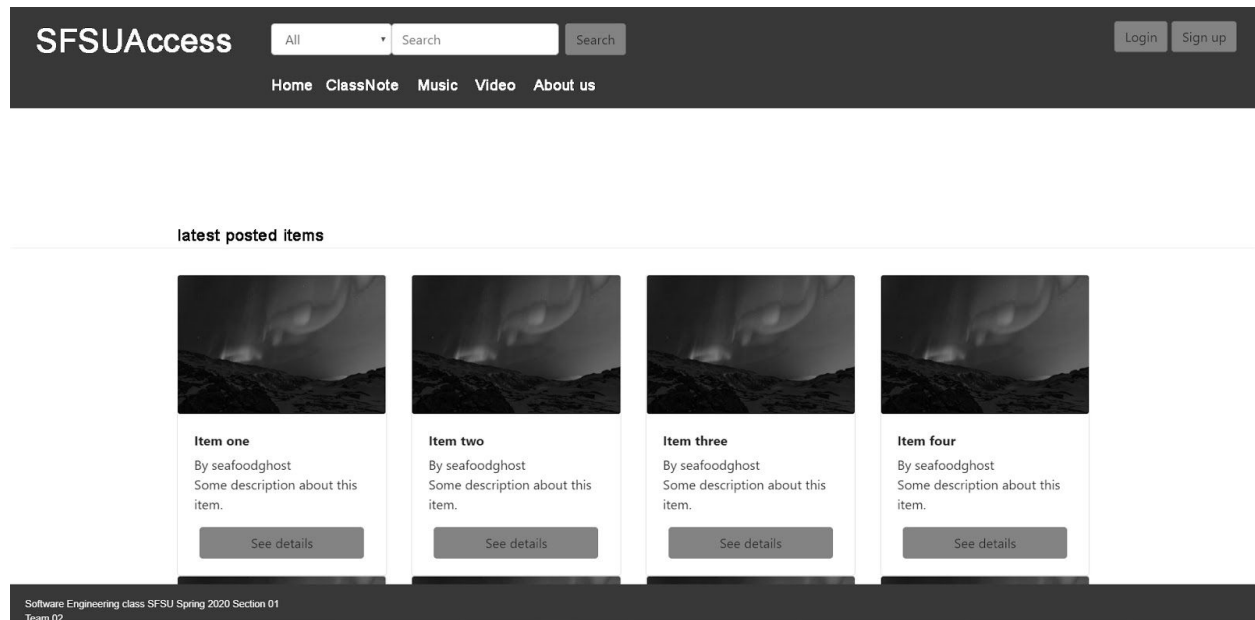
Priority 3:

Registered users:

6. Users shall be able to use the delete user account functionality to remove their account and data, should they no longer wish to use the services.

3. UI Mockups and Storyboards

Basic UI layout



NavBar:

- Home – Link to home page.
- Categories – Any user can search items by categories.
- About us – Link to the about page.
- Search section - Any user can search specific items in different categories by selecting a specific category and typing key words.

Greeting section (This section will be changed according to the current user's status):

- Non-registered Users: Name of our app, SFSU ACCESS, with description. Buttons link to login and sign up pages.
- Registered Users and Administrator: Name of the user. Buttons link to the dashboard and log out.

Content section (This section has different functions which response to current user's input):

- Non-registered Users: Showing a list of items by default, or the search result according to user input.
- Registered Users and Administrator : Able to perform anything that Non-registered Users can. In addition, this section links to the dashboard page for these users where they

can manage their account. For example, removing any posted items, posting items, viewing messages, approving or denying items, etc.

Footer

- Contains the class name, term, section, team.

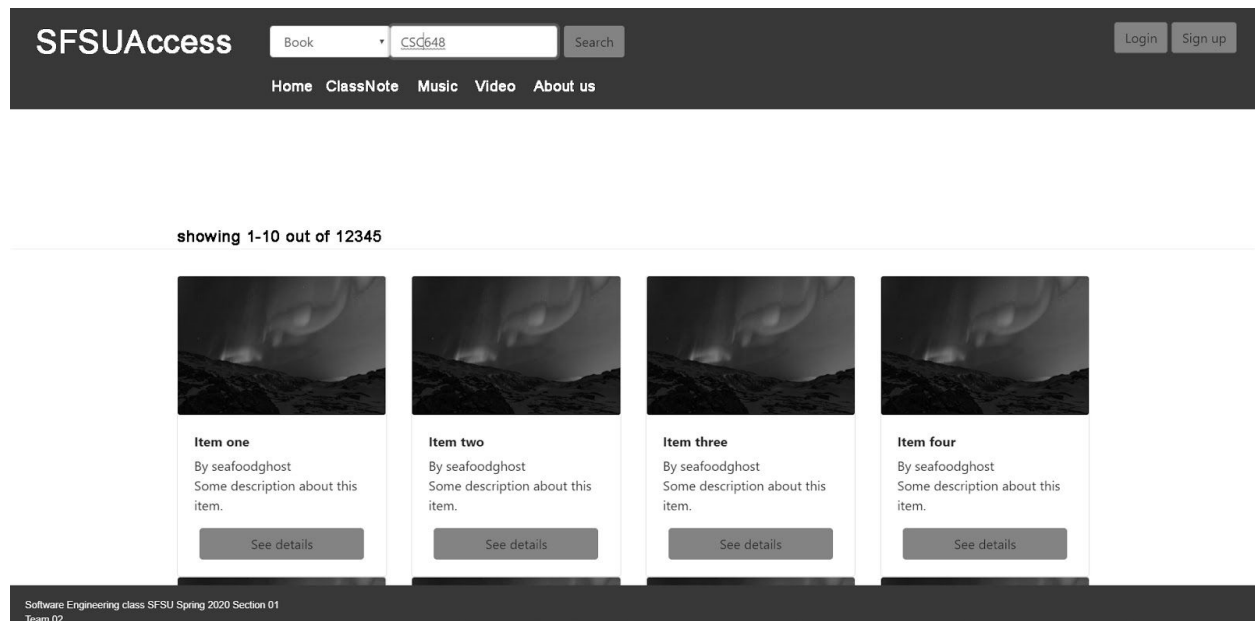
Storyboards

Case 1: A Non-registered User wants to search / purchase an item.

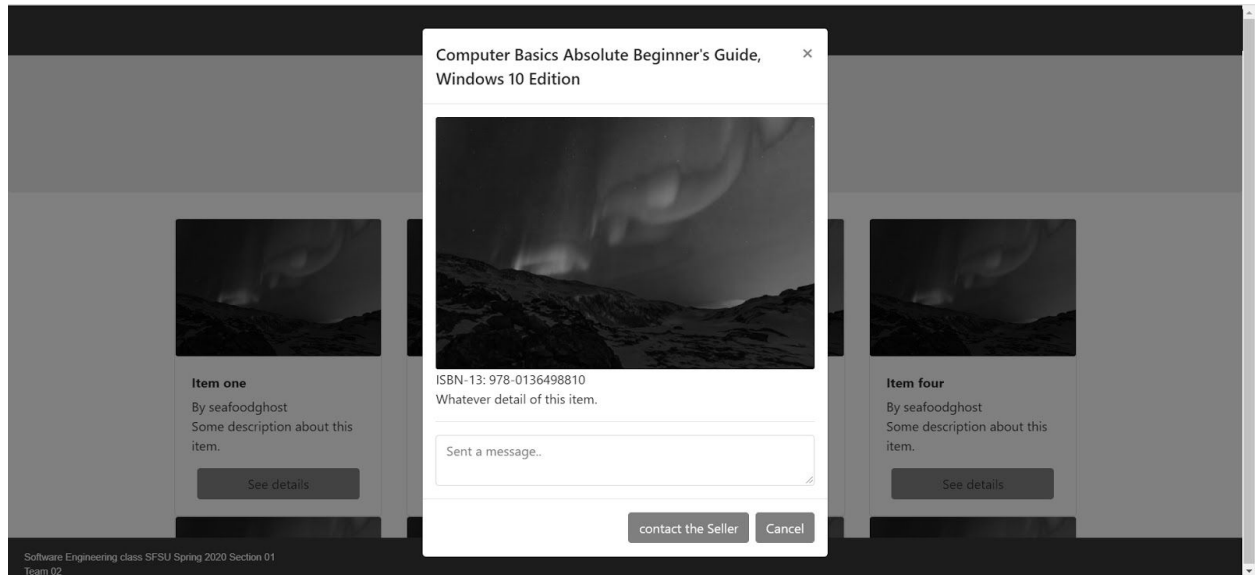
Jimmy (a Non-registered User) wants to search for a textbook for his CSC648 class.

Option one: He does the searching by selecting the Book/ClassNotes option on the navBar.

Option two: He does the searching by using the search section. Selecting the Book option in the category dropdown menu, then typing a relevant term, like “CSC648”, in the searching box.



The corresponding result will be displayed in the content section. Jimmy views the details for each item by clicking the See details button. Here, with a pop up box, he is able view all the details of this item. He is also able to either send a message to the seller or go back to the previous page. If he chooses to contact the seller, he will be prompted to register or log in.

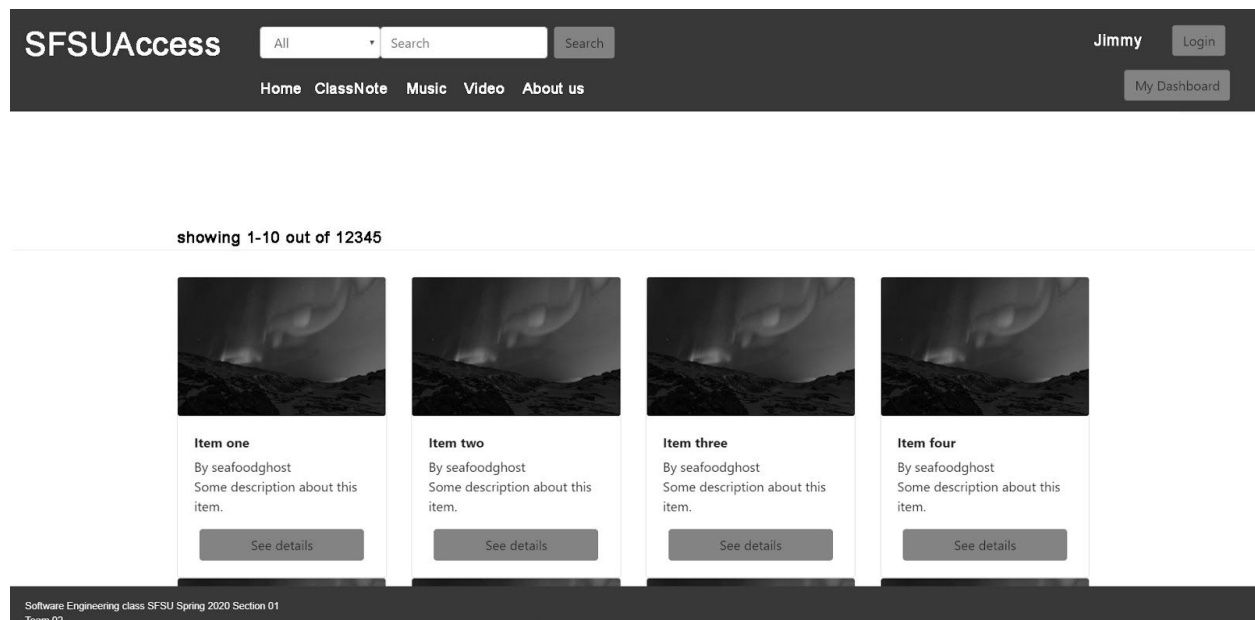


Case 2: A Non-registered User wants to list an item.

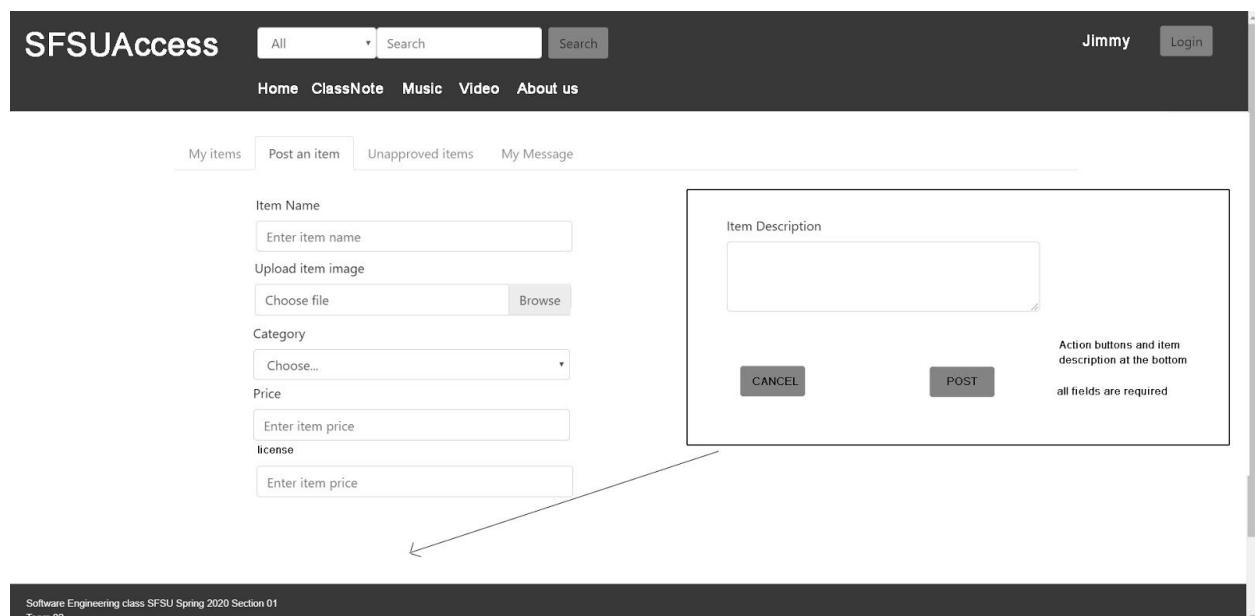
After finishing the semester, Alex (a Non-registered User) wants to sell his book on SFSU Access. He registers an account from the signup page.

The screenshot shows the "Sign up" page of the SFSU Access application. The page has a dark header with the text "Welcome to SFSU Access". The main content area is light gray and features a central "Sign up" section. This section includes a lock icon, the text "Sign up", and four input fields: "User Name *", "Email Address *", "Password *", and "Confirm Password". Below these fields are two buttons: "SIGN UP" and "BACK TO HOMEPAGE". At the bottom of the sign up section, there is a link that says "Already have an account? Sign in". The footer of the page reads "Software Engineering class SFSU Spring 2020 Section 01 Team 02".

Then he goes to his own dashboard by clicking My dashboard button on the home page.




In here, he selects the “Post an item” tab, then fills up the form with the relevant item information, such as name, category, images, pricing, description, and licensing. He is also able to view his items or messages in different tabs in a similar format.



Case 3: Registered Users / Administrators want to manage the account items.

In order to do that, they need to log in from the sign in page.

Welcome to SFSU Access



Sign in

User Name *

Password *

SIGN IN

BACK TO HOMEPAGE

[Forgot password](#)

[Don't have an account? Sign Up](#)

Software Engineering class SFSU Spring 2020 Section 01
Team 02

Registered Users:

Go to the user dashboard. To view posted items, items that are waiting for approval, make a post, read messages.

SFSUAccess

All Search Search

Jimmy Log out

Home ClassNote Music Video About us

My items (20)

Post an item(10) Unapproved items (10) My Message

number of items

Number	Name	Description	Remove item
1	Table cell	Table cell	Delete
2	Table cell	Table cell	Delete
3	Table cell	Table cell	Delete
4	Table cell	Table cell	Delete

Software Engineering class SFSU Spring 2020 Section 01
Team 02

Administrators:

They shall be able to approve or deny listing items, read messages in a similar format to the user dashboard.

SFSUAccess

AllSearchSearch

AdministratorLog outMy Dashboard

HomeClassNoteMusicVideoAbout us

ItemsPost an itemUnapproved itemsMy Message

Number	Name	Description	Approve/Deny item	
1	Table cell	Table cell	Approve	Deny
2	Table cell	Table cell	Approve	Deny
3	Table cell	Table cell	Approve	Deny
4	Table cell	Table cell	Approve	Deny
2	Table cell	Table cell	Approve	Deny
3	Table cell	Table cell	Approve	Deny
4	Table cell	Table cell	Approve	Deny

Software Engineering class SFSU Spring 2020 Section 01
Team 02

4. High level Architecture, Database Organization

DB organization:

Products
id
product_name
product_file_size
product_description
product_author

Users
registered
administrator

Messages
id
message
message_sender
message_receiver

Categories
class_notes_products
video_products
music_products

Licenses
All Rights Reserved
No Rights Reserved
Creative Commons Attribution 4.0 International

Media storage: Any media files related to items uploaded by users of SFSUAccess will be managed in a file system. MySQL will hold the respective relative URI to the item uploaded by the users.

For the purposes of this class project, we will be simulating upload and downloading of video files. The free-tier medium storage on our AWS server is limited on capacity. While we will upload and download audio(Music category) and document files(Class Notes category) to our file system, we will accept Youtube links to “videos files” for the purpose of this simulation.

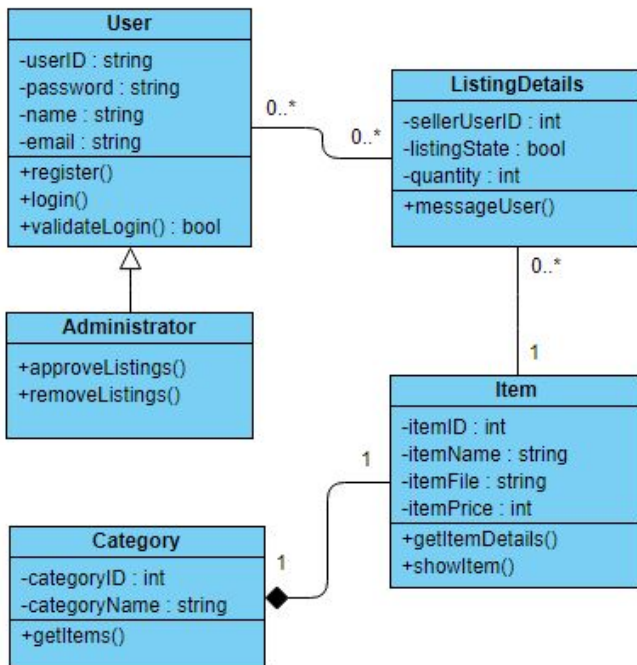
Search/filter architecture and implementation: Right now, our categories have a uniform set of field names. There will be a table associated with each categorie’s table in the database that will keep track of any fields unique to that category. This table will be queried to allow agnostic queries to be made without writing special functions per category to call these unique fields.

API: The only API we have planned at this stage in development are standard queries to the MySQL database.

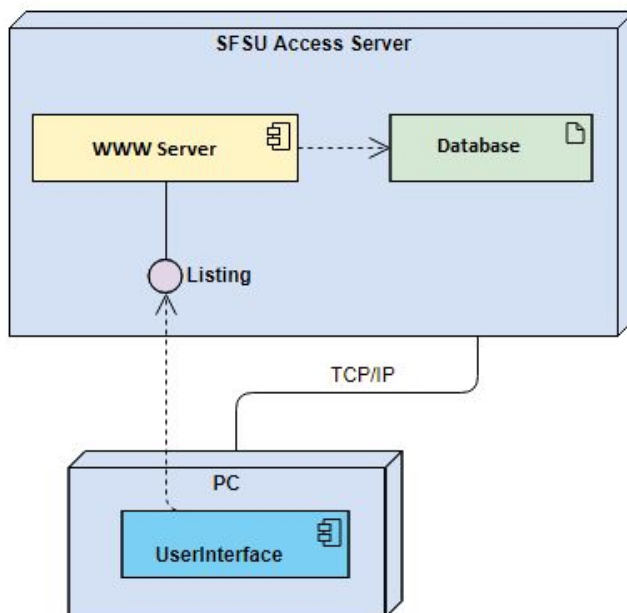
Additional Software Tools: At the beginning of this project, we had chosen Python for back-end / server logic; Amazon’s Web Services for hosting the back-end logic; and React.js for the front-end coding. After getting settled in on the environment, we have no current plans to make any changes this late into the development cycle of our project.

5. High Level UML Diagrams

a) High-level UML class diagrams for implementation classes of core functionality



b) UML Component and deployment diagrams



6. Project key risks

1. Schedule risks:

1.1 - Due to the current pandemic situation worldwide, there may be some changes to the time availability of some of our members. This is due to changes in employment schedules and residential displacement.

Proposed Resolution: Our group has identified these risks and held a discussion regarding any issues that any of our individual members may face. We have since modified our listed Priorities to mainly prioritize core user functionality before any administrative ones, using Workbench if it can not be implemented in time.

If our agreed upon solution is no longer feasible, we will contact the CTO for further action.

2. Technical risks:

2.1 - Due to the current pandemic situation worldwide, there could be unfortunate circumstances in which individuals of the team are unable to access their computers or the Internet and can not seek for these resources outside their shelter.

Proposed Resolution: The group has identified that all team members have mobile data and SMS texting on their smartphones. This will allow those that are affected by the technical risk to contact the rest of the group to communicate their situation. At that time, we should still be able to communicate with text and screenshots through our group chat platform.

3. Legal/content risks:

3.1 - Upload of media content that is not in accordance to copyright laws

While we are planning on including a public facing disclaimer on the public pages of our application regarding the purpose of this project, we are still obligated to follow local and federal regulations regarding the distribution of copyrighted content. We will diligently make sure that any content we as a team upload to the server will be considered “free-use” or media in which we own the rights to, however, there is the potential risk of mis-judging what is or is not allowed to be used on our service.

To minimize this risk, we have two lines of defense:

1. We will discuss this topic with our team to make sure everyone is up to date on this issue and learn how to differentiate media content we can use.
2. In the event a media file is erroneously uploaded to our server that is otherwise not allowed on the service, we will be making use of our Administrator dashboard to approve or reject such content before it is ever live on the service.

7. Project management

Our back-end lead Kevin has organized responsibilities and processes to follow for the team to get started on developing the project. An internal project document has been written to detail many areas of focus. We have a hierarchy of roles assigned to our members, as well as two sub-groups that have unique roles dedicated to that team.

We have researched development environments that make developing more efficient. This led to our team re-factoring our project structure to be in a more organized fashion. We have also meticulously researched any tools that we plan to use to ensure they are used correctly.

We have designated general responsibilities for each team and further clarified responsibilities using examples to ensure everyone understands their role in the team specifically. We have also discussed the responsibilities each team has to the other team by highlighting the importance of proper inter-communication.

We are continuously holding scrums weekly as well as contributing to our internal messaging system daily. All of our communications are recorded and are able to be accessed by anyone in the group. In further milestones, we will begin to practice our internal processes and begin assigning tasks through trello.

CSC 648-848 Spring 2020 team 2

Milestone 3 review summary

Team 2

5/13/2020

Summary of feedback and tasks to do :

Front-end UI

Home page:

- Navbar: Make the logo clickable, when the logo is clicked, show the home page, delete the home button; Make search box to the right, too much stuff on the left right now; separate “Post” button from “Search” a little bit more. Nav bar should be replaced with check buttons. SFSU disclaimer should go above the search box.
- Separate “About us” from navbar, put it to the left of “Log in”
- Display about the app message and recently posted item.
- Pull down menu should have lists of product genre, not type

Item-detail page:

- Add a “Free – download” button when the item is free.
- Cancel button – back to search result.(maybe not needed in our case)
- If the send message was clicked, a pop-up pre-fill window with details, and force users to login or sign up.

Post item page

- All “fields are required” message on the top.
- Putting the Upload file field to the end.
- Different color for cancel button.
- Next to the post item, a dummy text – item might take up to 24 hours to be approved.
- Include “thumbnail” field

Sign up / Sign in page

- Remove the account type selection and back to home page button.

Dashboard page

- “Post an item” should be a button to the right of the dashboard title

- In my message tab, adding what item this message refers to and the date that this message was posted. Similar to my item list.
- Remove “pending items”
- Add a tab for admin to approve / deny items.

Back-end

Github:

- Change the branch name “experimental” to “development”
- Commit messages need to be more specific
- Include header comments in each file

List of tasks the team chose to focus on and implement for final delivery :

Front-end

- Fixing all the UI issues based on the M3 review.
- Implementing the thumbnail, and upload-file system to the app.
- Implementing a loading state and message when the user uploads a file.
- Implementing the approve / deny function for the admin on the dashboard page.

List of final product P1 functions agreed at the meeting :

Home page:

1. Search text box shall only allow up to 40 alphanumeric characters
2. Search and search results shall be responsive

Non-registered users:

1. All listings provided by registered users shall be made viewable by non-registered users.
2. Users shall be able to search and browse all items
3. Users shall be able to view item details after clicking an item
4. Users shall register before being able to purchase or download items.

5. Users shall be able to create an account to engage in activity on the service. Registration shall require an SFSU email address.

Registered users:

1. Users shall be able to do everything that Non-registered users can.
2. Posting items to the store shall be subject to Admin approval.
3. Users shall be able to log in on the website.
4. Users shall be able to send and receive messages.
5. Users shall be able to post items for sale or download.
6. Users shall be able to remove their posted items.
7. Users shall be able to view all posted items and messages
8. Users shall be able to message seller using in-site message prefilled with product info

Administrator:

1. The admin shall be able to approve or deny pending listings.

SW Engineering CSC648/848 Spring 2020

M4

SFSU Access

Team 02

Team lead: Yanrui Xu

Team lead email: yxu13@mail.sfsu.edu

Back end lead & Github Master: Kevin Luong

Front end lead: JunMinLi

Team members: Cody Xu, Aitor Elvira, David Lin

Milestone 4

05/18/2020

History

Date Submitted	Date Revised
05/18/2020	05/21/2020

1. Product summary

The name of our project is called SFSU Access which is exclusive for SFSU students and faculty. In this project, we want to provide SFSU students and faculties a safe, easy, and convenient place to purchase, exchange, and share media contents. The market function allows people to browse media contents posted by other members or themselves. The member login function allows users to post, buy, and download contents after they have logged in to their accounts. The admin function allows website administrators to manage the contents posted by users to make sure the contents are safe and appropriate.

The major committed functions are:

Non-registered users:

1. All listings provided by registered users shall be made viewable by non-registered users.
2. Users shall be able to search and browse all items
3. Users shall be able to view item details after clicking an item
4. Users shall register before being able to purchase or download items.
5. Users shall be able to create an account to engage in activity on the service. Registration shall require an SFSU email address.

Registered users:

1. Users shall be able to do everything that Non-registered users can.
2. Posting items to the store shall be subject to Admin approval.
3. Users shall be able to log in on the website.
4. Users shall be able to send and receive messages.
5. Users shall be able to post items for sale or download.
6. Users shall be able to remove their posted items.
7. Users shall be able to view all posted items and messages

8. Users shall be able to message seller using in-site message prefilled with product info

Administrator:

1. The admin shall be required to approve or deny pending listings.

Website URL: <http://54.177.237.30/>

2. Usability test plan

The major function being tested is upload/posting.

i. Test objectives:

We want to test the process and correct functionality of file uploading and posting. As SFSU Access is an e-commerce website, it is important for our users to be able to upload their files so that we can process their product and list it for other users to see. Should users encounter issues with our file uploading and posting process, they will not be able to create their listing and will be unable to continue with merchandising. Having our valued users turn away from our platform due to usability issues that we could resolve is completely unacceptable.

ii. Test background and setup:

- System setup
 - Any modern browser that can run Javascript, such as Google Chrome. In order to reduce potential interference by browser extensions or add-ons, we recommend launching the browser in Guest mode.
- Starting point
 - To start head over to <http://54.177.237.30/>. This is our homepage. We recommend signing in first. If you do not have an account, just click Sign Up on the top right. Next, click Post Item next to the search bar.
- Intended users
 - We intend for our users to generally be college students, with some faculty in the mix. Our users won't need to be technologically adept at understanding how our platform works, because we've streamlined our posting process.
- URL of system & measurements
 - We are testing our functionality at <http://54.177.237.30/Postitem>. We are measuring the ease of use and user satisfaction with using the function. A small likert scale questionnaire, posted below, will be used to gauge and record those measurements.

iii. Usability task description:

TASK	DESCRIPTION
Task	Post a new item
Machine state	Post item page is not filled

Successful completion criteria	Item is posted
Benchmark	Completed in 1 minute

Task: Please post the given example.jpg image to SFSU Access. You will start on our homepage at <http://54.177.237.30/> while already logged in to a user account. Any extraneous details during the posting process is up to you. You have one minute to post the image.

We would measure effectiveness by determining how many users are able to complete the task on time and record any errors thrown from the form. We expect high effectiveness as our form has minimal restrictions for inputting the required data.

We would measure efficiency by determining how quickly the user completes the posting form. We expect the average time to post an item to not exceed 1 minute, though this may vary due to differing description word counts and time to search and upload files.

Likert scale:

- I found the post item form easy to navigate (check one)

☐ Strongly agree ☐ Agree ☐ Neither agree nor disagree ☐ Disagree ☐ Strongly disagree

- I can browse my file easily ☐ (check one)

☐ Strongly agree ☐ Agree ☐ Neither agree nor disagree ☐ Disagree ☐ Strongly disagree

- I had no issues with filling in the form (check one)

☐ Strongly agree ☐ Agree ☐ Neither agree nor disagree ☐ Disagree ☐ Strongly disagree

3. QA test plan

Test objectives: Perform QA testing on the file upload/post process to ensure there are no issues. Posting should take no longer than one minute on average. Confirm that all character count limits are reasonable, and that the form is secure against abuse.

Setup: (Optional) Register or log in first. Go to <http://54.177.237.30/Postitem> to get started.

Features to be tested: Test the process of uploading/posting files. Test the limits of the upload form.

QA Test Plan Table:

Test #	Test Title	Description	Inputs	Expected correct output	Test Results
1	Uploading unsupported file type: .exe	To test the file type limit of our system, we will upload an .exe file.	Title: Image Magick Category: Other Price: 0 License: Copyrighted Description: "The Image Magick exe installs Image Magic 7.0.10-10-Q16-x64 dll." File: ImageMagick-7.0.10-10-Q16-x64-dll.exe	The listing information should not be inserted into the database and the file upload field will specify it is an invalid file type.	Google Chrome Version 81.0.4044.138 - PASS Microsoft Edge 44.18362.449.0 - PASS
2	Upload a .jpg file with max length character limits	To test the form input character limit, we will upload a .jpg file with the title and description filling out the limit. The limit is currently 30 characters for the title, and 500	Title: Example of Sbox and encryption Category: Notes Price: 0 License: Free use & modification Description: "Many different block ciphers use a special	The listing should be returned with a successful message. Upon reviewing the item in the dashboard's pending items page, all text entered should be present.	Google Chrome Version 81.0.4044.138 - PASS Microsoft Edge 44.18362.449.0 - PASS

		characters for the description.	substitution called an ``S-box". The AES also has these S-boxes, which it terms the ``SubBytes Transformation". S-boxes provide an invertible (reversible) transformation of segments of plaintext during encryption, with the reverse during decryption. With the AES it is a single simple function applied over and over again to each byte during stages of the encryption, returning a byte. For more info, visit: http://www.cs.utsa.edu/~wagner/laws/SBoxes.html ” File: Untitled2.png		
3	Upload a .jpg file after resetting the form once	To test the form’s reset integrity, we will upload a .jpg image and fill in all information as needed, then reset the entire form. Fill in the form again with different	Title: Image QA Category: Other Price: 0 License: Copyrighted Description: “For testing” File: wallpaper.jpg Title: Image QA2 Category: Video	The listing information should be inserted into the database and a success message is shown. All information seen in the dashboard’s pending items page should stay	Google Chrome Version 81.0.4044.138 - PASS Microsoft Edge 44.18362.449.0 - PASS

		information and .jpg file and submit.	Price: 1 License: Free for Project Description: “For testing again” File: 1500x500.jpg	the same as with what was submitted.	
--	--	---	--	--	--

4. Code Review

Developer

CSC648 M4 Code-review



Junmin Li

Thu 5/21/2020 7:49 PM

To: Kevin Luong; cody xu <codyx2019@gmail.com>; David Lin; Aitor Elvira Monsalve; Yanrui Xu

Cc: Junmin Li



Hi all

The following code, the post item function, is ready for M4 code review. Please feel free to make comments.

JunMin

```
//PURPOSE: This page is used for posting items by logged in users.
//          Lazy registration is implemented for non-registered/ non- logged in users.
//AUTHOR: JunMin Li

useEffect (() => {
  axios.get('/api/search').then(response =>{setList(response.data)}).catch(error=>console.log(error));
  console.log("isLoggedIn? "+ user_isloggedin);

  if(typeof cookies.post_item !== 'undefined'){
    setName(cookies.post_item.product_name);
    setCategory(cookies.post_item.product_category);
    setPrice(cookies.post_item.product_price);
    setLicense(cookies.post_item.product_license);
    setDescription(cookies.post_item.product_description);
  }
},[]);

//Reset user input on post item form.
const resetForm = ()=>{
  document.getElementById("itemForm").reset();
  setFileName("Upload File here...");
}
```



David Lin <dlin5@mail.sfsu.edu>

to Junmin ▾

May 21, 2020, 8:13 PM



Hey Junmin

Everything looks good to me. Good use of proper variables and function names -- they really help in identifying what they are for and what they do. That, and your use of comments more ambiguous code makes it easier to read and understand. I think we could work on having better continuity on space usage (such as with the "()" -> "{" and "()" -> "}", though that isn't a big deal. Great job again.

David

...

5. Self-check on best practices for security

Major assets we are protecting:

User information, posted media and media accessibility.

Major threats for each asset above:

Threats for user information includes snooping for user information within information supplied to the client by the server. Threats of unauthorized media accessibility includes directly downloading uploaded assets through means that bypass regular e-commerce processes, in addition, corruption of media.

How we are protecting each asset:

Privacy of users - The application minimizes the usage and accessibility of user information. Only necessary information is sent to the client to allow client-side processing, such as their names. We never send sensitive information, such as emails and passwords, which are only processed server-side. All account passwords are encrypted and stored in the database. The access to the database is also limited, so only authorized administrators are given information to access it.

Media accessibility - Small resolution thumbnails of uploaded files are used when displaying details of the file, such as on the search results and in the item's detail page. The full resolution or original file is presented for download if the item is listed as free, else the media can only be accessed by contacting the seller. The seller must then agree to a deal and perform the transaction themselves. Media is safeguarded by the server itself, which means only media requested by the backend is sent to the frontend. Media that is not free on the website is never returned.

Input data validation list:

Input data is being validated by Formik and Yup custom validation.

Home page:

- Search bar input for up to 40 alphanumeric characters.

Sign in / Sign up:

- All fields are mandatory.
- First name and last name must be 15 characters or less
- Email must be a SFSU email which shall either be “@mail.sfsu.edu” or “@sfsu.edu”.
- PW must be more than 8 characters.
- PW and confirmPW fields must be matched.

Post item:

- All fields are mandatory.
- Product name must be 30 characters or less.
- Price must be a real number.
- File must be picked
- File type must be supported (mp3, mp4, pdf, jpg, png)
- Product description must be 500 characters or less.

6) Self-check: Adherence to original Non-functional specs

List of non-functional requirements

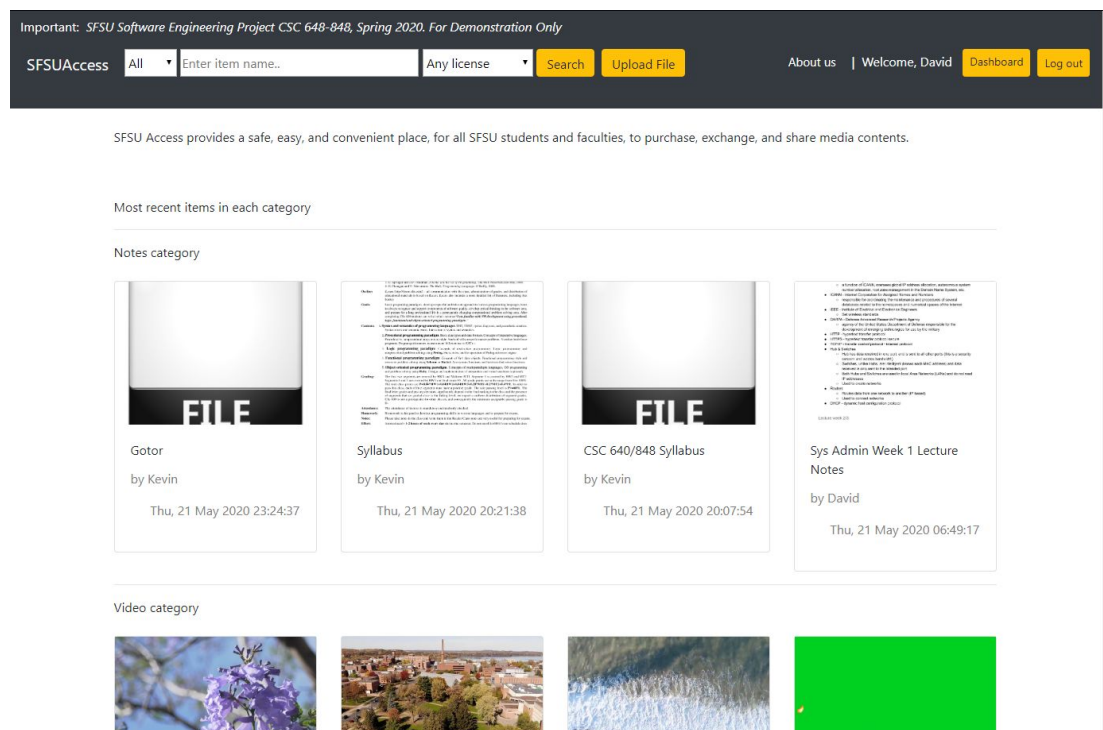
1. Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers. DONE
2. Selected application functions must render well on mobile devices. DONE
3. Data shall be stored in the team's chosen database technology on the team's deployment Server. DONE
4. Full resolution free media shall be downloadable directly, and full resolution media for selling shall be obtained after contacting the seller/owner. DONE
5. No more than 50 concurrent users shall be accessing the application at any time. DONE
6. Privacy of users shall be protected and all privacy policies will be appropriately communicated to the users. DONE
7. The language used shall be English (no localization needed) DONE
8. Application shall be very easy to use and intuitive. DONE
9. Google analytics shall be used DONE
10. No email clients shall be allowed DONE
11. Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI. DONE
12. Site security: basic best practices shall be applied (as covered in the class) for main data items. DONE
13. Media formats shall be standard as used in the market today. DONE
14. Media material shall be either free or for sale, as determined by media owner. DONE

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. DONE

16. Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development. DONE




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). DONE

Product Screenshots:



My Dashboard

[Post an item](#)[My Items](#)[Pending items](#)[My Message](#)

Item	Thumbnail	Details	Remove item
1		Title: Sample ER Diagram Description: An ER diagram of the relationships between courses and students Uploaded: Wed, 20 May 2020 22:06:01 GMT	Remove
2		Title: Explosion green screen effect Description: Free to use explosion green screen effect! Uploaded: Thu, 21 May 2020 05:39:36 GMT	Remove
3		Title: Architecture General Description: CSC 648 lecture notes on Architecture General Uploaded: Thu, 14 May 2020 06:37:22 GMT	Remove

Post an item

All fields are required

Title

Category

Price

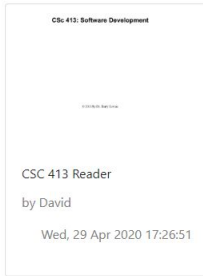
License

Item Description

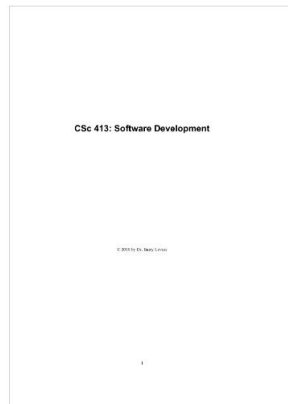
Upload your file.

Notice: all item might take up to 24 hours to be approved.

Results with search key: 'All', '413' and 'Any license'. Showing 1-1 out of 1.



1



CSC 413 Reader

by: David

Category: Notes

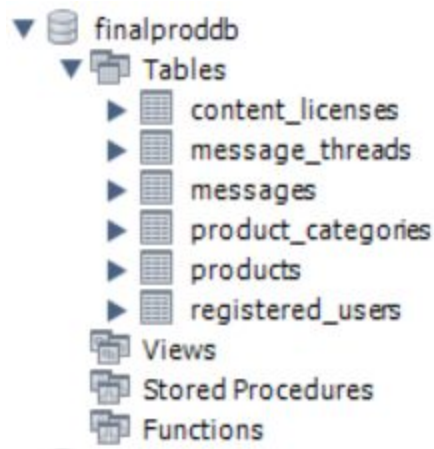
License: Copyrighted

Price: 15

Description: full reader for 413 by Barry Levine. Only \$15 today

Database Organization:

Table list



Product table

	id	product_name	product_category	product_author	product_description	register	product_license	product_status	date_time_added	price
▶	113	sad machine	Music	Kevin	pr	1	Free use & modification	ACTIVE	2020-05-17 19:02:34	0
	125	Sample ER Diagram	Notes	David	An ER diagram of the rel...	13	Free use & modification	ACTIVE	2020-05-20 22:06:01	0
	126	Explosion green scre...	Video	David	Free to use explosion gr...	13	Free use & modification	ACTIVE	2020-05-21 05:39:36	0
	128	z	Video	Kevin	z	1	Copyrighted	DENIED	2020-05-21 20:05:28	0
	128	CSC 640/848 Syllabus	Notes	Kevin	Here is the latest syllabu...	1	Free use & modification	ACTIVE	2020-05-21 20:07:54	0
	130	qwe	Notes	Kevin	s	1	Free to SFSU related projects	DENIED	2020-05-21 20:11:17	0
	131	z	Notes	Kevin	dx	1	Free use & modification	DENIED	2020-05-21 20:11:47	0
	132	Syllabus	Notes	Kevin	some class syllabus	1	Free use & modification	ACTIVE	2020-05-21 20:21:38	0
	133	music test	Music	Kevin	z	1	Free use & modification	DENIED	2020-05-21 20:24:01	0
	134	Architecture General	Notes	David	CSC 648 lecture notes o...	13	Copyrighted	ACTIVE	2020-05-14 06:37:22	0
	135	SF Pier 39 Flag	Video	David	B roll of the pier 39 flag f...	13	Free to SFSU related projects	ACTIVE	2020-05-15 13:40:02	0
	136	SFSU Logo	Other	David	the sfsu logo	13	Copyrighted	ACTIVE	2020-05-21 20:28:32	0
	137	Panorama shot of SF	Video	JunMin	Get a clip of a panoramic...	2	Copyrighted	ACTIVE	2020-05-19 21:27:59	10
	138	Sys Admin Week 1 L...	Notes	David	Here's the notes that I t...	13	Free use & modification	ACTIVE	2020-05-21 06:49:17	1
	139	Post Malone Circles	Music	John	Post Malone performing ...	14	Copyrighted	ACTIVE	2020-05-21 20:57:06	0
	140	Free B Roll footage	Video	John	Use it for your next medi...	14	Free use & modification	ACTIVE	2020-05-21 21:02:44	0
	141	Moon and Wolf wallp...	Other	JunMin	minimalist wallpaper 150...	2	Copyrighted	ACTIVE	2020-05-17 15:30:04	2

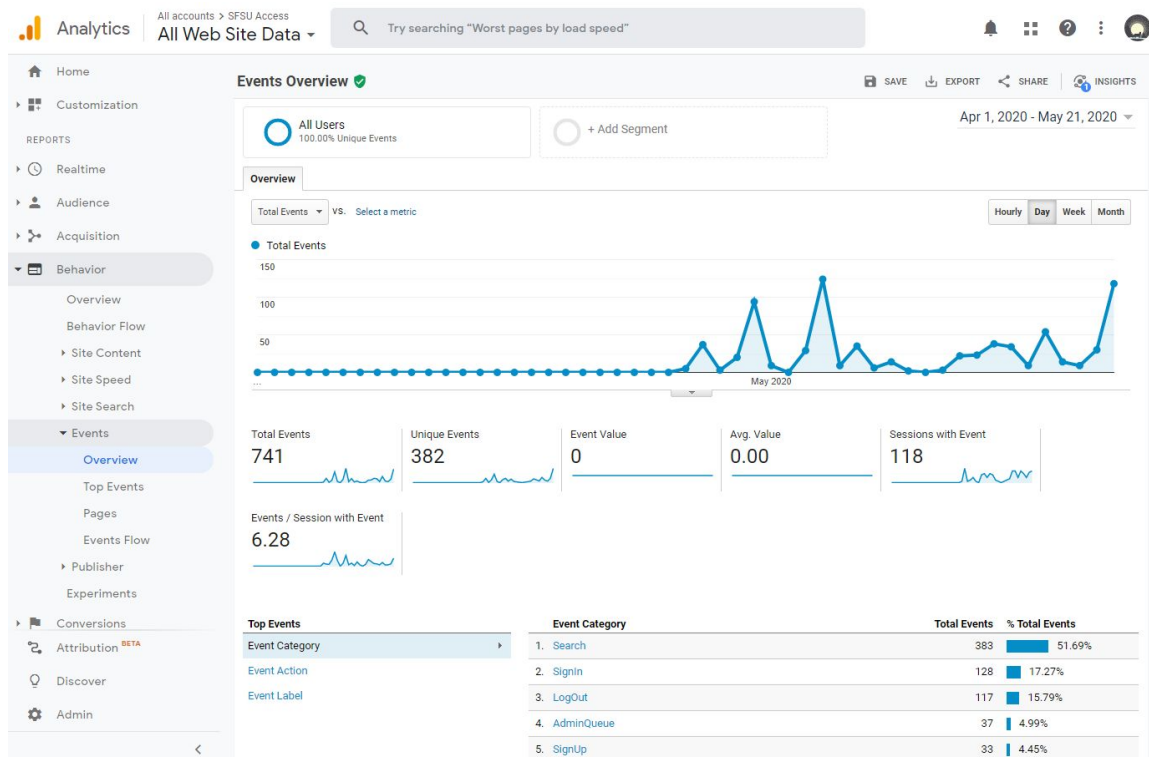
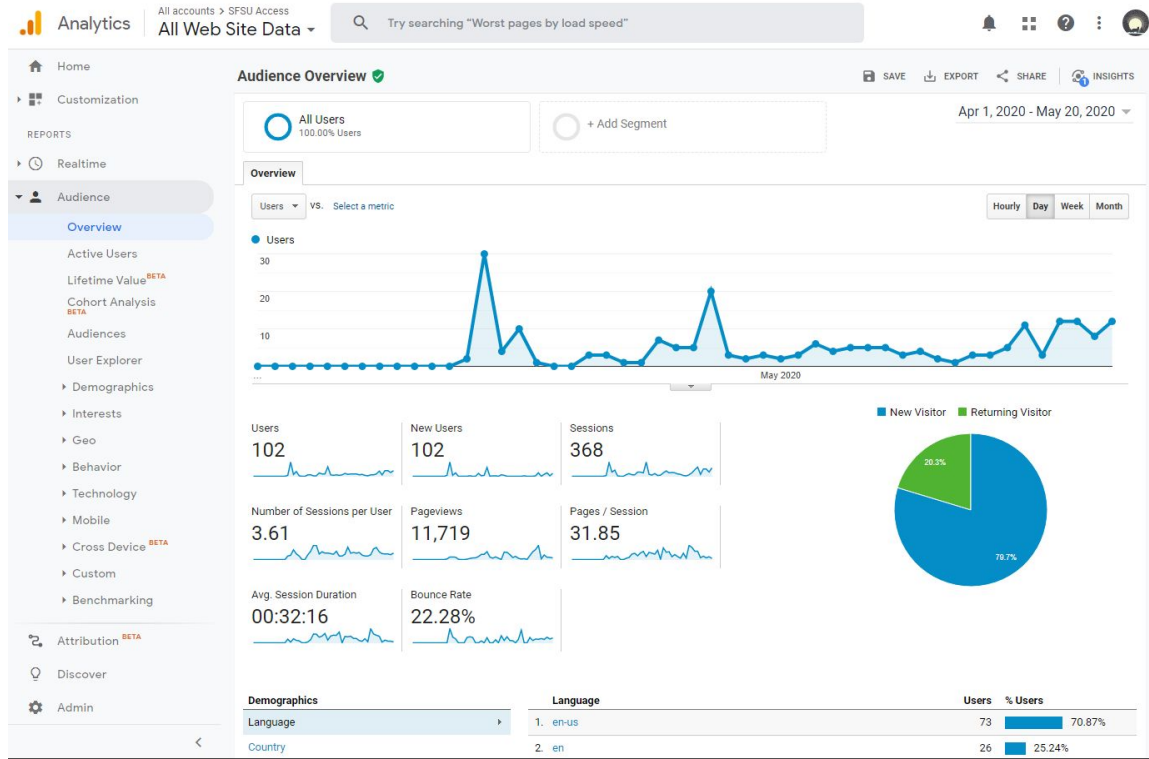
User table

	id	first_name	last_name	email	password	privelege_type
▶	1	Kevin	Luong	kluong4@mail.sfsu.edu	fdd530c759df56a225c5614812e0b2dd	1
	2	JunMin	Li	jli58@mail.sfsu.edu	16c90f148f396e32639c476760e95231	1
	12	bob	singh	bs@mail.sfsu.edu	fdd530c759df56a225c5614812e0b2dd	2
	13	David	Lin	dlin5@mail.sfsu.edu	5f4dcc3b5aa765d61d8327deb882cf99	2
	14	John	Smith	jsmith@sfsu.edu	5f4dcc3b5aa765d61d8327deb882cf99	2
*	NULL	NULL	NULL	NULL	NULL	NULL

Categories table

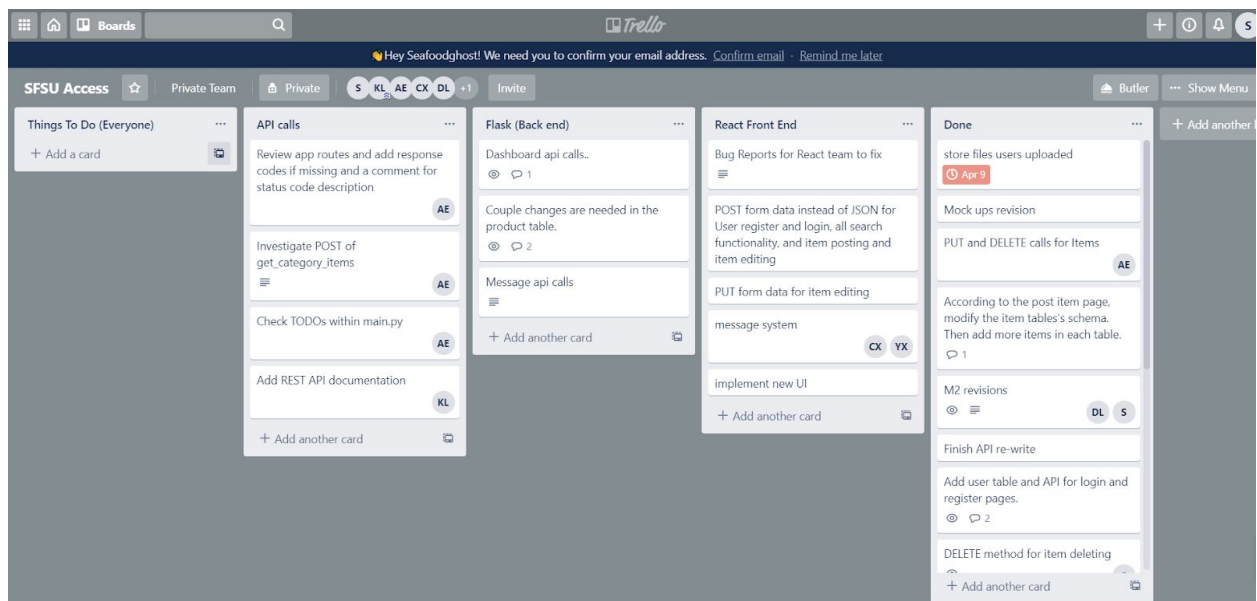
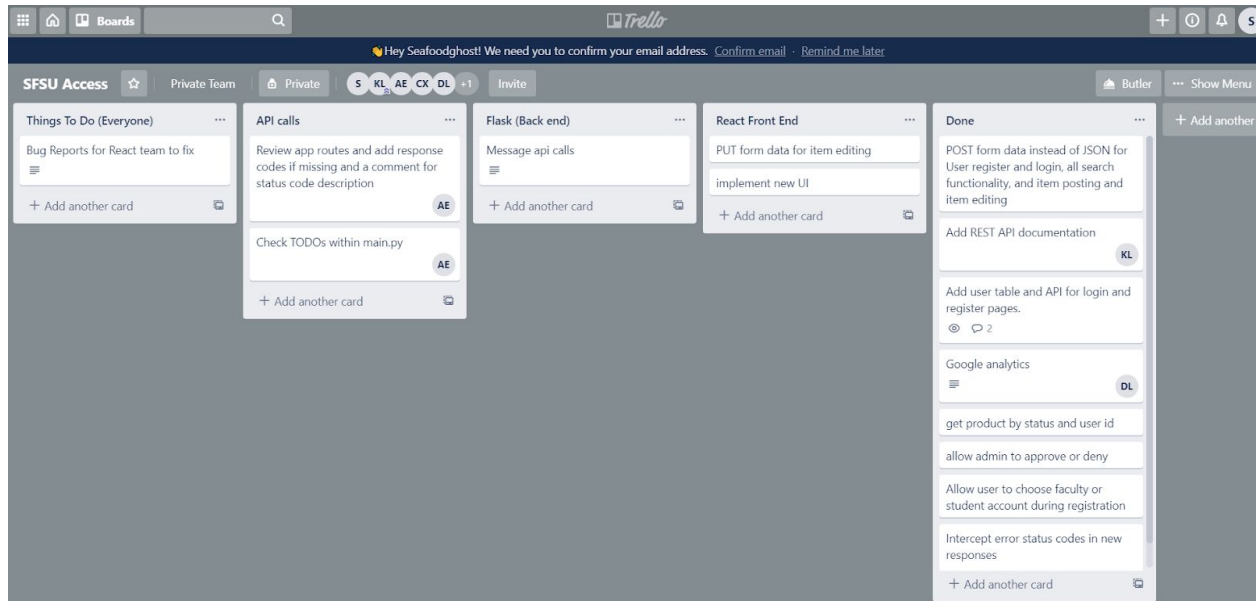
	id	product_category_name
▶	1	Notes
	2	Music
	3	Video
	4	All
•	NULL	NULL

Google analytics plot:



Project management:

We use Trello to manage our project during the development process.



Team member self assessment and contributions:

Kevin's self assessment

KL

Kevin Luong

Thu 5/21/2020 9:09 PM

To: cody xu <codyx2019@gmail.com>; Aitor Elvira Monsalve; Yanrui Xu; Junmin Li; David Lin



- a. His/her contributions to team project and teamwork (technical and any other) in no more than half a page – point format is OK.
 - initiated project structure and wrote internal documentation to onboard group members into the workflow.
 - Wrote the blueprints of the flask app
 - Set up production environment on the Amazon Web Services platform
 - Handled deployment of development code to server
 - Created the database schema
 - Managed members of the back end team (Aitor and Kevin)
 - Handled bug reports from development team related to backend code
- b. Number of submissions he/she made to github team dev. Branch
38 commits.
- c. One brief paragraph on main challenges he/she encountered in team project
Navigating the development workflow and developing new flows that best worked with the pandemic situation as well as the various schedule conflicts we had with one another. One aspect I found challenging is figuring out how to deal with other coding styles that did not reflect the intended usage of our stack choices. For example, I was asked a few times to make changes that would fundamentally go against the REST api guidelines which would create unscaleable code.
- d. One brief paragraph on what would he/she do better next time based on what was learned in the class about SE management and processes

I think that one thing that stood out to me as a team was the lack of discipline of sticking to processes that were developed to make developing easier. For example, we initially set up a bug reports thread on Trello that was used a few times initially but abandoned because of it being a separate service, response time to bugs were slow as it meant logging in to yet another service.

Another issue I think can be prevented next time is increasing the amount of time planning and designing of the prototype/infrastructure. We encountered issues with mis-communication on what is intended in certain app features and why certain design choices were made.

David's self assessment

DL

David Lin

Thu 5/21/2020 10:21 PM

To: codyx2019@gmail.com; Aitor Elvira Monsalve; Yanrui Xu; Junmin Li; Kevin Luong

Cc: David Lin



- a) His/her contributions to team project and teamwork (technical and any other) in no more than half a page – point format is OK.

- google analytics
- backend mysql search results
- navbar responsiveness
- populating website with sample listings
- some bug hunting/fixing

- b) Number of submissions he/she made to github team dev. Branch
18 commits / 14 pull requests

- c) One brief paragraph on main challenges he/she encountered in team project
Initially encountered difficulties setting up the IDE properly. Due to ongoing commitments and the pandemic, time became an issue when ironing out critical bugs. It was also challenging to fix bugs when determining whether it was a backend or frontend issue, and whether or not I knew enough of the framework/language to be able to correctly fix it.

- d) One brief paragraph on what would he/she do better next time based on what was learned in the class about SE management and processes

I think next time I/we need to set a proper deadline for all features and functions, and then get straight into just testing and QA. While many issues were fixed on time for the demo, some were discovered only up to a day beforehand. Another thing is to learn more about both frameworks and languages being used since I had to deal with hunting for bugs and trying to fix them.

Cody's self assessment



cody xu <codyx2019@gmail.com>

Thu 5/21/2020 10:54 PM



To: Yanrui Xu; Junmin Li; Aitor Elvira Monsalve; David Lin

a) His/her contributions to team project and teamwork (technical and any other) in no more than half a page – point format is OK.

- Form Validation check feature.
- Contact seller message system.

b) Number of submissions he/she made to GitHub team dev.
-35 commit to experimental branch.

c) One brief paragraph on main challenges he/she encountered in team project
-One of the challenges I have encountered when implementing the form validation check is deciding how it should be implemented without adding too much code to the original component which will make things a lot more complicated. It took me a few iterations to finally make it work.
Initially I thought of writing the function myself but then I learned there is an easier way of validating a form which is to use the Yup npm library.
It was designed specifically for form validation. Choosing to use a library instead of writing the function saved a lot of time and makes the program a lot easier to understand and debug.

d) One brief paragraph on what would he/she do better next time based on what was learned in the class about SE management and processes

One thing that I learned from this project is to try out different implementations and designs fast. If you're implementing something that doesn't seem to work, try something else instead of wasting time on the original ideas.

JunMin's self assessment



Junmin Li

to me, Kevin, cody, David, Aitor, Junmin

10:14 PM (1 hour ago) ☆ ↶ ⋮

a. My contribution

As the front-end lead of this project, my responsibility is to provide necessary technical support, feedback to our team. In addition, managing the front-end team and communicate with the back-end team to ensure our progress is on track.

- Construct the overall front-end architecture for our project
- Construct, revise the overall UI
- Implement p1 functions, ensure they work properly by doing the necessary testing
- Assign tasks to front-end members, and keep checking their progress
- Regularly communicate with back-end lead about the overall implementation

b. The number of submissions I made to GitHub development branch

44 commits

c. Main challenges

Create an engaging environment within the team by keeping everyone fully participate in the project due to the shelter-in-place situation and schedule conflicts. As we don't meet each other face to face, but only through the chat. Lack of communication happened during the project progress.

d. What would I do better next time based on what was learned in the class about SE management and processes

I think that one thing for sure is project management. Although we use Trello to keep tracking our progress, not everyone has fully participated which led to miscommunication. As a result, when someone tried to implement certain functions on one specific page, sometimes, I, who made the UI, need to explain why / how the figure was created in a certain way from the beginning. Based on this experience, I should communicate with my teammates more frequently and put everyone on the same page first, before we work on the details individually.

Yan's self assessment



Yanrui Xu

Thu 5/21/2020 11:39 PM



To: Junmin Li; Kevin Luong; cody xu <codyx2019@gmail.com>; David Lin; Aitor Elvira Monsalve

a) His/her contributions to team project and teamwork (technical and any other) in no more than half a page – point format is OK.

- Decided project tool list
- Created milestone documentations
- Arranged team meetings and established team work/communication platforms
- Presented at final demo

b) Number of submissions he/she made to github team dev. Branch

11 commits.

c) One brief paragraph on main challenges he/she encountered in team project

It is hard to have every member contribute to the project on the same degree. I think that comes from some people may be weaker in coding or tasks are not assigned clearly.

d) One brief paragraph on what would he/she do better next time based on what was learned in the class about SE management and processes

This was my first time being a team lead of six people and working on a project at this scale. Looking back now I think I would be clearer on short-term checkpoints and keep constant communication with team members. I would also assign tasks more specifically and constantly check with each member's progress.



Aitor Elvira Monsalve

Fri 5/22/2020 1:47 AM

To: Yanrui Xu



a) His/her contributions to team project and teamwork (technical and any other) in no more than half a page – point format is OK.

- Messaging API.

- Database Queries.

b) Number of submissions he/she made to GitHub team dev.

- 5 commit to experimental branch.

c) One brief paragraph on main challenges he/she encountered in team project

- One of the challenges I have encountered when working from my home country. Because of time difference it made harder to be available on certain times

and it was harder to coordinate with my team members. I also had to learn how to use MySQL with Python as it was the first time I used it.

d) One brief paragraph on what would he/she do better next time based on what was learned in the class about SE management and processes

I will try to make coordination with my teammates a little bit easier so everything could get done faster.