California University of Pennsylvania

CSC 490: Senior Project

CalU Book Exchange

Specifications Document

Nick Cavalancia

Joe Rimsky

Ryan Dean

INSTRUCTOR COMMENTS/EVALUATION PAGE

Table of Contents

ABSTRACT

The CalU Book Exchange will provide a way for students of California University of Pennsylvania an alternative, cheaper method to acquire required reading and supplementary material for their classes. The service will be a website and will be accessible by both mobile and desktop devices that have an internet connection. This service is meant to be an online marketplace where students can buy and sell books at exchange rates of their choosing. This document will detail user's interactions with the service as well as any hardware, software and system requirements. Program maintenance and testing will also be covered.

DOCUMENT DESCRIPTION

Purpose and Use

This document is meant to serve as a detailed description of the various details of the project and the specifications needed for the project to operate correctly. This is meant to serve as a contract, between developer and client, and to explain the scope of the project. This document will explain how the product will function and what is required for it to function correctly. This will allow for the client to see if what is written here will satisfy their needs. On the development side of things, this will provide an easily definable scope to ensure that the product is completed accurately and successfully.

Intended Audience

This document is meant for the client and the developers. For the client, this document is a way to ensure that the product will satisfy their needs. The constraints section provides information to ensure that the product will fit within the client's set guidelines. This section will cover the cost as well as other aspects that the client would take interest in. The acceptance portion will go over what the client needs before using the application. If there is a change in the needs of a client, this document should be rewritten and must be accepted by everyone involved. Until then, this document is a binding agreement and is the basis for all development.

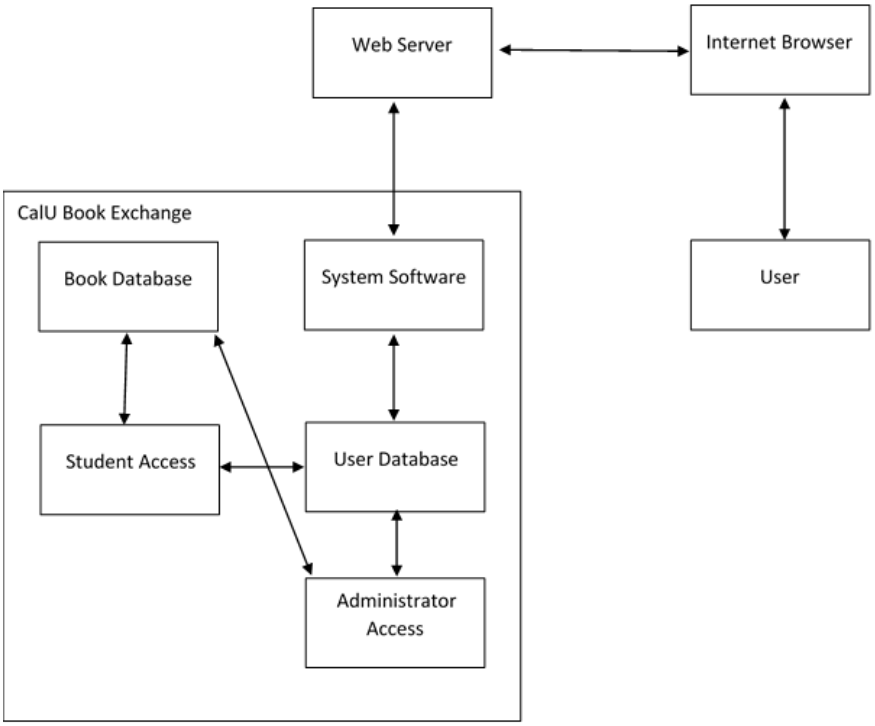
The developers will use the information contained in this document when writing the code and testing the application throughout its lifetime. The constraints section will provide the developers with the knowledge needed to create a service that will fit within the specified guidelines. Additionally, the data flow, use case and entity diagrams should be used to produce a quality product in a structured, object-oriented, and logical manner.

System Description

Overview

The CalU Book Exchange software is meant to aid students. That aid comes in the form of a marketplace for students to buy and sell books to each other at agreed upon rates. College can already be expensive to attend. This service is meant to lighten the financial load of students by giving them a way to sell and also earn money back. This is a web-based service that students will be able to access through their CalU username and password. The service will have to communicate across the internet and store information on a server hosted by the university (figure 1).

Figure 1: Block Diagram of Program Management System



Environment and Constraints

End User Profile

For this service, there are two categories of users: the admin and the user that can operate as both a buyer and a seller. The administrator will have full access to every function within the

application. Functions for the administrator category will be to remove listings, feedback and accounts. Additional users can be granted administrator access for extra assistance in maintenance. Access rights can only be modified by the lead administrator for security reasons.

The administrator will start by interacting with the application and logging into the service. Once validated, the administrator will then have access to all of the administrator functions. The administrator will be able to view all accounts on the service, as well as all postings and feedback that have been created.

User interaction will begin by starting the application and logging into the service. Once logged into the account, the user will be able to edit their profile, post what book(s) they are selling, and view other users' posts for purchasing purposes. A user will only be permitted to leave feedback for transactions in which they participated. A user will only be permitted to use the functions given to them unless they were promoted to an admin account.

User Interaction

Student Account:

The CalU Book Exchange's end user will be the students of California University of Pennsylvania. This gives a large poll of people from different backgrounds that could be end users. The user only requires a valid CalU login username and password to access the service.

Admin Account:

Administrator accounts will be overseers of the service. These will be far fewer in number than the student accounts. The primary purpose of administrator accounts is to keep the service running as it should.

Shared Interactions:

The interactions in this section of the service will be shared by both administrators and students. The only requirement to use this service is a device capable of accessing the internet. The user then must travel to the CalU Book Exchange Website where they will be prompted to login. Upon entering login information, the system will be able to login to the appropriate account. The account itself will be flagged as a student or administrator. Both users will be capable of removing themselves from the service, leaving and removing their own feedback and accessing personal account information.

Hardware Constraints

To run the server, one should have a computer with the hardware capable of running web server software, such as Apache or NGINX. Minimum requirements for PHP installation are also necessary. There should be enough space on the server's hard drive for the database to be stored as well. Compatible hardware for users includes computer or mobile device with a functioning internet connection.

Software Constraints

In order to run the software, the system must be able to run PHP and a web server software, such as Apache. For users, the ability to access one of the following web browsers: Internet Explorer, Google Chrome, or Mozilla Firefox. The project will be created with these three browsers in mind.

Time Constraints

Logging onto the system requires the authentication of a username and password combination that is stored on a file on the server. This process will take less than a second from start to finish of entering the file and returning an authorization value for acceptance or denial. Accessing the server's database for listings should also take no more than a second to display

onto the formatted webpage. Updating the database with new postings and user accounts will take a few seconds as well.

Since communication will be based off of SMS messages and emails outside of the server, effective communication relies mainly on how fast the chosen service is able to transmit the data between recipients. Normally this should not take any longer than a few seconds to a couple minutes, but occasionally there will be delays. Network outages or flooded servers can cause messages to be delayed by hours or days depending on the severity.

Installing the application only requires the system files to be placed into the desired directory to act as a server. Any additional files that need created can be done as pre-compiled binary packages so there is no time constraint from a user standpoint. The total estimated time for installation should not be any more than 5-10 minutes.

Cost Constraints

The service's cost can be broken down into two core components. The first is the cost of hosting the website, and the second is the cost on maintaining it. The first is a simple fee. The second is more complex as the system requires people to be administrators and work to keep the system clean. These two core components make up the cost of the service. A third cost is having a device of some sort with the ability to connect to the internet.

Other Concerns

Designing a proper user interface will be another constraint. Implementing a Graphical User Interface (GUI) is one subject that we have a lack of experience in as a group. We feel that the concept of a GUI was not touched on as much with more modern languages. A useful interface is going to take some time to design and code, so this going to be of concern.

Acceptance Test Criteria

Testers

The CalU Book Exchange will be tested by the team as it is being developed. Two team members will code the section and the third teammate will test the functionality of that section of code and look for errors. Team members will rotate roles in order to allow for each member to both code and test various sections of the project.

Additionally, the team will find a test group of at least 5 users that are unfamiliar with the project. These users will be able to use all modules of the software and will be asked to report back any issues found with the software and their input will be used to make corrections.

In both cases, actual data will be used to test the various modules of the software. The only functions of the software that will not be accessible are any modules that allow user payment information to be collected. These sections will not need tested because they are to be hosted by third parties, such as PayPal or a major credit card company, and their reliability will be taken into consideration.

Criteria for User acceptance

If all of the conditions to follow are met, the CalU Book Exchange will be considered a successful and complete project:

1. The Client will be able to access a full release version on any device.
2. The User Interface is easily/completely navigable.
3. Users are able to create/modify their account
4. Users are able to view/modify their account information
5. Users (sellers) are able to create/edit items listed for sale
6. Users are able to search for appropriate materials
7. Users are able to purchase desired material

8. Users (buyers and sellers) are able to leave appropriate feedback
9. Users are able to mark items sold
10. Users are correctly alerted when any of the above is completed.
11. Administrator is able to delete/modify items for sale.
12. Administrator is able to delete/modify feedback.
13. Administrator is able to remove users.
14. All information is displayed correctly on the appropriate web pages.
15. Integration with CalU course requirements/information is finalized.

Integration of Separate Parts and Installation

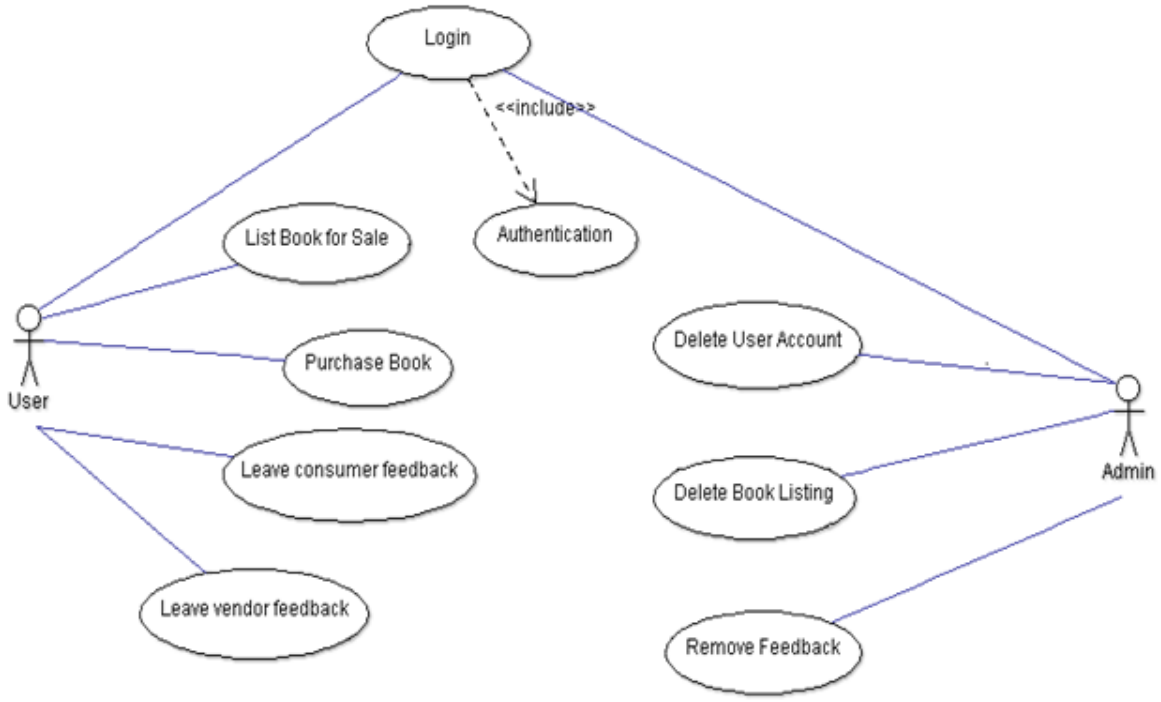
The CalU Book exchange is a web-based service and will be hosted on a web server that is administered by a third party. The only installation that must be completed by the client is a compatible web browser that was listed earlier in the document.

Installation of the server-side software will be completed by the development team. The appropriate files will be placed on the web server and the proper installation requirements will be completed. After the files are correctly uploaded, the software administrator will complete a one-time system set up to gain access to the system. Then the administrator will set up an administrator username and password. After this setup is complete, the administrator will then be able to “turn on” access to outside users and enable new account registration.

System Modeling

Functional: Use Cases and Scenarios

Figure 2 – Use Case Diagram



Login Scenario

The login will accept the user's email address as a username and a password to access the website.

Normal Scenario

Steps:

1. User/Admin provides username (email address)
2. Authentication verifies correct username was entered
3. User/Admin provides password
4. Authentication matches the password and username in the database
5. User/Admin will be logged in.

Exception Scenario

Steps:

1. User/Admin provides username (email address) and password
2. Authentication detects username and/or password is invalid
3. A message is displayed explaining that the username and/or password is invalid

List Book for Sale Scenario

List book for sale will allow the user to list a book for sale.

Normal Scenario

Steps:

1. User will provide the name and/or ISBN of the book
2. User will list the price and condition of the book
3. User will upload an image of the book
4. Listing will be available for view in the appropriate section of the website

Exception Scenario

Steps:

1. User attempts to list a book to sell
2. User does not provide all the required information
3. An error message is displayed instructing that all required fields must be filled

Purchase Book Scenario

Purchase book will allow a user to purchase a book.

Normal Scenario

Steps:

1. User will be able to search for a specific book by name/course/etc.
2. If desired, a price filter can be used
3. Users will select the book to be purchased
4. A notification will be sent to the seller
5. The buyer/seller will receive contact information for the other party
6. A payment method will be agreed upon
7. Users will meet in person to exchange the book and/or money
8. Consumer will be asked to leave feedback on the vendor
9. Listing will be marked as sold after transaction is complete

Exception Scenario

Steps:

1. Buyer/seller will commit to buying a book/selling a book
2. The book is no longer needed/for sale
3. The book will be marked for sale and re-listed on the website

Leave Consumer Feedback Scenario

Vendors will be able to leave feedback on the consumer.

Normal Scenario

Steps:

1. After the transaction, the vendor will be asked to leave feedback
2. Feedback will affect the reputation of the user receiving feedback

Exception Scenario

Steps:

1. After the transaction, the vendor will be asked to leave feedback
2. Feedback is not left

Leave Vendor Feedback Scenario

Consumers will be able to leave feedback on the vendor.

Normal Scenario

Steps:

1. After the transaction, the consumer will be asked to leave feedback
2. Feedback will affect the reputation of the user receiving feedback

Exception Scenario

Steps:

1. After the transaction, the consumer will be asked to leave feedback
2. Feedback is not left

Delete User Account Scenario

The Admin will be able to delete a user account that is found in violation of rules.

Normal Scenario

Steps:

1. Admin will receive notification of account in question from user input.
2. Admin will select the user account to be deleted.

3. The Admin will provide a reason for deletion
4. The user will be notified via email that their account was deleted.
5. The user account will be removed from the database.

Exception Scenario

Steps:

1. The Admin will determine if an account is to be deleted
2. The Admin will select the user account to be deleted
3. The Admin provides a reason for deletion
4. A notification is sent to the Admin telling them the account was already removed/does not exist

Delete Book Listing Scenario

The admin will be able to delete book listings that do not comply with the standards of the application.

Normal Scenario

Steps:

1. Admin will receive notification of listing in question from user input.
2. Admin will select appropriate book.
3. The Admin will then provide a reason for the deletion
4. The user will be notified of the deletion of their listing with the reason it was removed.
5. The listing will be removed from the database.

Exception Scenario

Steps:

1. The Admin will determine if the listing is to be deleted
2. The Admin will select the book listing to be deleted
3. The Admin provides a reason for deletion
4. A notification is sent to the Admin telling them the book listing was already removed.

Remove Feedback Scenario

The Admin will be able to delete Feedback that does not comply with the standards of the application.

Normal Scenario

Steps:

1. Admin will receive notification of feedback in question from user input.
2. The Feedback will be selected by the Admin.
3. The Admin will provide a reason as to why this Feedback was deleted.
4. This reason will be relayed to the account of the person whose Feedback was deleted.
5. The Feedback will be deleted.

Exception Scenario

Steps:

5. The Admin will determine if the feedback is to be deleted
6. The Admin will select the feedback to be deleted
7. The Admin provides a reason for deletion

8. A notification is sent to the Admin telling them the feedback was already removed/does not exist

Entity: Class Diagrams

User Class	Student Class	Post Class
Username : String Password : String Category : Integer Permissions : List	Name : String Location : String Feedback : integer	Title : String Price : Integer Description : String
VerifyUser() AddUser() ModifyUser() ModifyPost() DeleteUser() DeletePosting() GetUserName() GetUserCategory() GetFeedback()	AddStudent() LeaveFeedback() GetCategory()	AddPost() UpdatePost() GetPost()

Figure 3 – CalU Book Exchange Class Diagram

Class Descriptions

User Class

The user class is used to store admin and student information in the server. Admins will have access to adding and deleting users/postings.

- Username, data type: string – this data will hold the name of the account
- Password, data type: string – this data will hold the secure copy of the password to compare to the typed password on login
- Category, data type: integer – this data will hold the value that will determine whether the user is a student or administrator
- Permissions, data type: list – this will hold the permissions given to each user
- VerifyUser() - verifies the username and password the user enters
- AddUser() - adds a user to the system

- ModifyUser() - modifies a user's account
- ModifyPost() - modifies an already listed post
- DeleteUser() - deletes a user from the system
- DeletePosting() - deletes a posting from the system
- GetUserName() - gets a user's username
- GetUserCatagory() - gets a user's category (student or admin)
- GetFeedback() - retrieves user feedback

Student Class

The student class is used to store student information in the database that users can view and change with the given functions.

- Name, data type: string - this data holds the name of the user for contact information
- Location, data type: string – this data holds the location of the user for posting purposes
- Feedback, data type: integer – this data will hold the feedback rating of the user (1-10)
- AddStudent() - adds a student user to the system
- LeaveFeedback() - allows user to give feedback on another user
- GetCategory() - returns the category of the student (admin or student)

Post Class

The post class is used to store the posts made by users in the database.

- Title, data type: string – this data holds the title of the post
- Price, data type: integer – this data holds the price listed for the post
- Description, data type: string – this data holds the description of the post

- AddPost() - adds a post to the system
- UpdatePost() - updates a post on the system
- GetPost() - returns a post from the system

Dynamic: State Chart

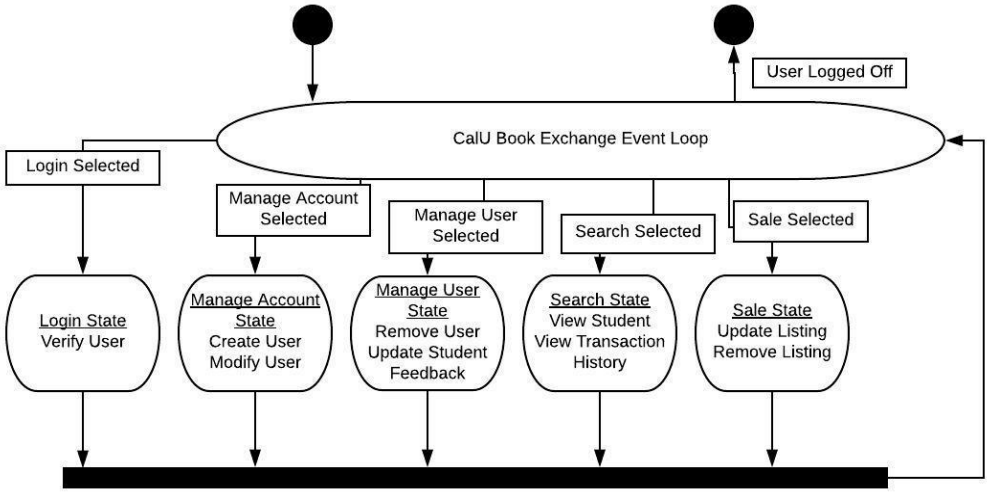


Figure 4 – CalU Book Exchange Dynamic State Chart

States

There are 6 states that control the program. The user's input will be responsible for selecting what state the system will go to. The following are the states of the system as shown in figure 4.

Default State:

The default state is the homepage of the website. If a user is not logged in, a log in page will be displayed asking the user for their credentials. When the user is logged in, the system displays a home page with options that are available to that type of user (admin or student).

Login State:

The login state is the result of an attempt to login to the website from the default state. Authentication will occur using the information provided by the user to verify that their account exists, and the username and password are a match. Under this login state we also have the similar create account option where the user creates a new account. The Login State will notify the user if they must enter in their information again due to a failed login.

Manage Account State:

The Manage Account State is achieved by administrators. This state is for the administrators to review and delete users that have not complied with community standards. In this state Administrators also are given the options to send warnings and remove feedback.

Manage User State:

This state is for Users to update their own information. The Manage User state will have easily modifiable fields such as payment information and contact information. The User will also be able to change their password.

Search State:

The Search State is when a user accesses the search functionality to look for a book. The user can search the website using ISBN, course, instructor, and book title.

Sale State:

The Sale State is achieved when the finalization of a sale must take place. The state is meant to have two parts. The first is the original agreement. This is when a transaction is decided to occur. The second part is the finalization. This is for after the transaction has occurred. The finalization step removes the listing and prompts both users for feedback.

Events

Verify User Event:

User verification happens anytime a user attempts to login. The system takes the username and password entered and compares those to the information that is stored in the user database. The username is the first to be used. It is searched for within the user database. If found, the passwords are compared, and if matching, the user is logged in. Upon failure, a reason is determined based off if a username was found. If a username was not found the user is formed that no such account exists. If a username is found, but the passwords do not match, the user is informed that the login attempt as failed due to an incorrect password.

Create User Event:

This event occurs when a user first decides to use the system. The first step is to create an account. The Create User Event ask for the user to enter a valid CalU email as their username. The first step is to verify the email they entered is a @calu.edu email. The second is to determine if the username already exists within the database.

Remove User Event:

This event is incurred by an Admin. The system processes the account the admin has marked for deletion. After deletion the system notifies the account's owner through email. The reason for termination will be written within the email.

Modify User Event:

This event occurs when the admin is modifying a user's profile, whether it's feedback or profile activity. Also, a student can use this when modifying their own profile. Modifying the user will be done on the system file that saves the information for each user.

View Student Event:

Viewing student information is done by the system when an admin or student account is requesting information on a user. Also, this will be used when viewing posts to see who made the post to the system.

Update Student Event:

Updating a student is done by the student account. A student can use this when requesting to update their profile information. This update is done by the system and will update the user's profile on the system file for use throughout the website.

View Transaction history Event:

Viewing transaction history is performed by the system after a user views another user's profile. The system will display the title of the listing and whether or not the user sold or purchased this listing. Users can use this information when viewing a post made by a user.

Update Listing Event:

Updating a listing can be done by either an admin or student account. Updating the listing will cause the entry on the server end to be changed for public viewing.

Remove Listing Event:

Removing a listing is done by the administrator to keep the system up to date. The system will delete this post from the list, which will keep it from being viewed on the public website.

Feedback Event:

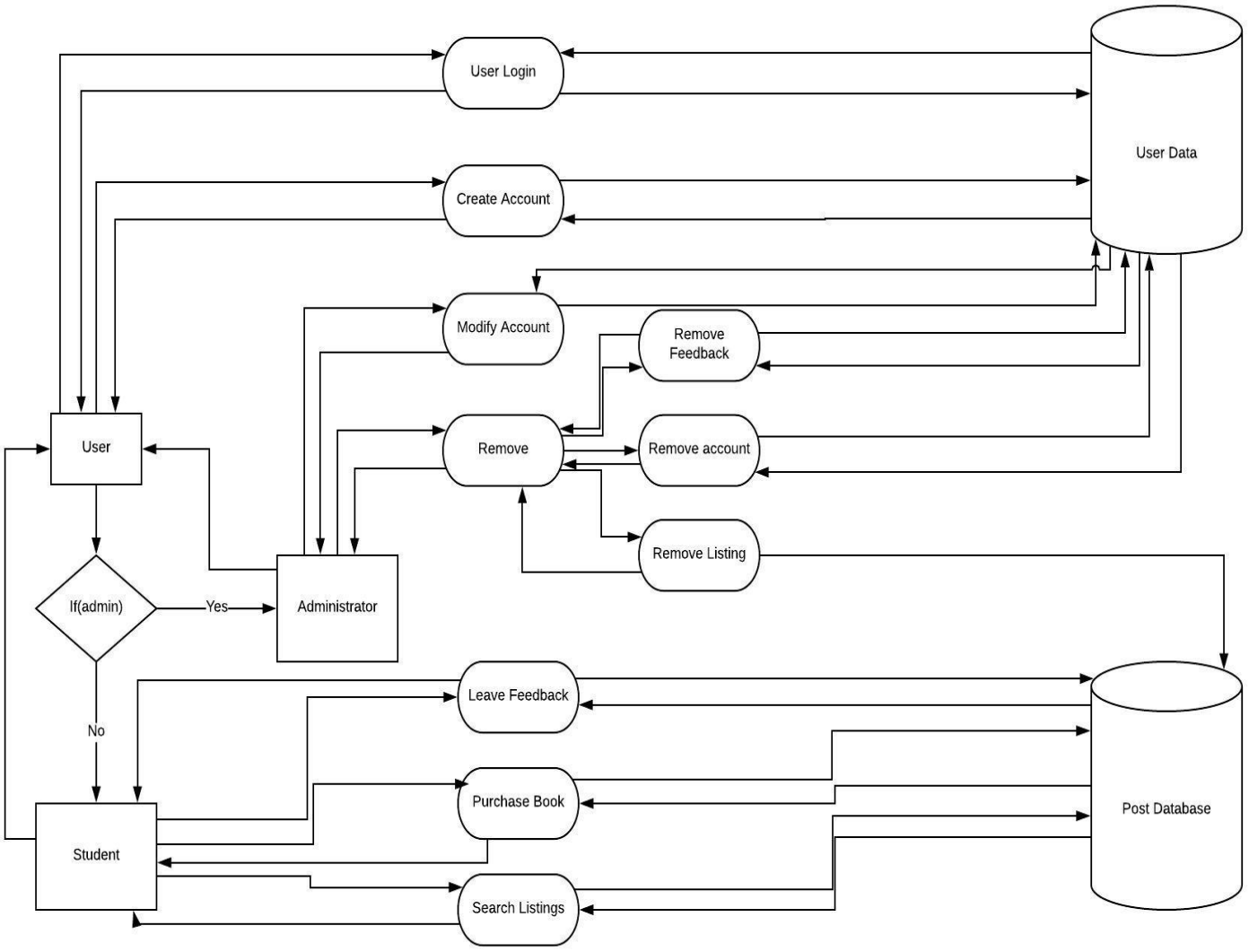
Feedback is left by a user for public viewing. The user is initially set to a feedback of NULL until they have completed a sale/purchase and the other user leaves feedback on their account. The system will display this information when users are viewing a post. Once a user leaves feedback, the system will modify the receiving user's feedback variable for future use.

Transitions

Transitions happen when a user selects a system function. The first step in a transition is to detect if the user has the appropriate permissions for the task they have selected. While most of the CalU Book Exchange functions will be only displayed to the appropriate user, if a user has temporarily lost privileges in the service due to misconduct, this will detect it.

Data Flow Diagrams

Figure 5 – CalU Book Exchange Data Flow Diagram



Components/Tools Needed for Implementation

Hardware Requirements

- Device with internet access
- Device with a display
- Input devices keyboard/mouse

Software Requirements

- Internet Browser

Appendix A: Technical Glossary

Program – The Computer Science Program offered at California University of Pennsylvania

System – The CalU Book Exchange system designed for the university.

Apache - Apache HTTP Server – free web server software.

GUI - Graphical User Interface – a user interface that uses graphics instead of a console interface.

Integer - data type that stores whole numbers.

List - data type that stores connected lists of information

PHP - Hypertext Preprocessor – a scripting language that can be embedded in html.

String - data type that consists of sequential characters.

Appendix B: Team Details

Nick Cavalancia is responsible for overseeing the workflow of the proposal and specifications documents. He is in charge of the creation of the documents as well as assigning tasks to each group member for efficient work flow. Ryan Dean is the leader for the design and requirements documents and their creation. This phase of the assignment will take place in the second half of the 2018 fall semester. He will be responsible for the organization of the documents and assigning tasks to each member.

In the spring 2019 semester, two additional phases will be worked through. Joe Rimsky will be responsible for leading the development and testing sections of the application during the

first half of the semester. Responsibilities will be equivalent to those of the other group members. As a group, we will all be contributing equally on the user manual and final presentation documents. This will take place in the second half of the spring semester and will finish off the project.

Appendix C: Writing Center Report

I, the undersigned, certify that this document has been reviewed by a member of the staff at the California University of Pennsylvania Writing Center.

Signature: _____ Date: _____

Appendix D: Workflow Authentication

I, Nick Cavalancia, certify that I have performed the actions specified in this document.

Signature: _____ Date: _____

I, Ryan Dean, certify that I have performed the actions specified in this document.

Signature: _____ Date: _____

I, Joe Rimsky, certify that I have performed the actions specified in this document.

Signature: _____ Date: _____