

# ELO

---

## Application Requirement Specification

09/13/2019

CS 440 Advanced System Design

## List of Responsibility

Originator:

**Jivan Kharel**  
Group Manager

**Mai Trinh**  
Group Member

**Tony Shi**  
Group Member

Reviewers:

**Raven Waters**  
Project Client / Founder

Approvers:

**Clifton Presser**  
Professor

**Rodney Tosten**  
Professor

## Table of Contents

<b>List of Responsibility</b>	<b>2</b>
<b>Table of Contents</b>	<b>3</b>
<b>Document Control</b>	<b>5</b>
Revision History	5
Appendix	5
<b>Introduction</b>	<b>6</b>
Purpose	6
Objectives and Scope	6
System Design	6
User Characteristics	6
Technical Constraints	6
<b>Functional Requirements</b>	<b>7</b>
F-01: Log-in	7
F-02: Sign-up	9
F-03: Rent Book	9
F-04: Buy Book	12
F-05: View Cart & Check Out	14
F-06: Remove from Cart	16
F-07: Sell Book	17
F-08: Create Profile	19
F-09: Log Out	20
F-10: Change Profile Information	21
F-11: In App Messaging	23
F-12: View Current Orders	24
F-13: View Past Orders	25
F-14: View WishList	25
F-15: Troubleshooting & Reports Complaints	26
F-16: Search bar, Sorting & Filter	27
F-17: Contact	29
F-18: Recommendation Section	31
F-19: Push Notifications	32
F-20: Manage payment method	34
F-21: Make in app payments	35
F-22: Reset password / Forgotten Password	37
<b>System Design</b>	<b>40</b>
<b>Test Cases</b>	<b>40</b>
F-01:Log-in & F-02: Sign-up	40
F-05: Add to Cart & F-06: Check out	41
F-03: Rent book	41
F-04: Buy book	41
F-07: Rent out book & F-08: Sell book	41

ELO	4
F-09: Create Profile	41
F-10: Log Out	42
F-11: Change Profile Information	42
F-12: In App Messaging	42
F-13: View Current Orders	42
F-14: View Past Orders	42
F-15: View WishList	42
F-16: Troubleshooting & Reports Complaints	42
F-17: Search with sorting and filter:	42
F-18: Contact:	42
F-19: Recommendation Section:	42
F-20: Push Notifications:	43
F-21: Manage Payment Method:	43
F-22: In-app Payment:	43
F-23: Reset Password/Change password:	43
<b>User Interface</b>	<b>43</b>
<b>Non functional requirements</b>	<b>44</b>
Database	44
Payment Integration	44
Technical and Design Constraints	45
Timeline & Schedule	46
Instruction & User Manual	46
<b>Deliverables</b>	<b>47</b>
<b>Glossary</b>	<b>47</b>
<b>Individual Responsibilities</b>	<b>48</b>
<b>References</b>	<b>48</b>

## 1. Document Control

### 1.1. Revision History

Date	Change Description	Author
12/6/2019	Modification in system design, instruction and user manual, add table and figure labels	Tony Shi
11/20/2019	Added Payment Integration section	Mai Trinh
10/03/2019	Wrote test cases and labelled UI screens for F01,02,03,04,05,06,07,08.	Mai Trinh
10/02/2019	ER Diagram	Jivan Kharel
09/26/2019	Added use cases for F-08	Mai Trinh
09/24/2019	Activity Diagram and screen details for F-17 - F-24	Jivan Kharel
09/23/2019	Added use cases for F-03, F-04, F-05, F-06.	Mai Trinh
09/22/2019	Functionalities,use cases, labeling	Tony Shi
09/22/2019	Added use cases for F-01, F-02, F-07. Added user interface and system design description.	Mai Trinh
09/20/2019	Functionalities	Jivan Kharel
09/16/2019	Added user interface constraints. Added use case for log-in functionality.	Mai Trinh
09/12/2019	Added functionalities, objective and scope, product perspective,user characteristics, constraints, and brief user interface description	Jivan Kharel
09/09/2019	Created the document and outlined document sections	Mai Trinh

Table 1.1.1 Table for revision history

### 1.2. Appendix

#	Term	Definition
1	Users	The people who will be using the application and utilizes its functions
2	Admin/Administrator	A superuser of the application and has control over all functions of the system
3	Web Portal	A specially designed website that brings information from diverse sources (wikipedia)
4	GC	Abbreviation for Gettysburg College

Table 1.2.1 Table for Appendix

## 2. Introduction

### 2.1. Purpose

The purpose of this document is to provide an overview and technical details of the project ELO. This document defines the application's functional, non-functional, user interface, system design and database requirements; target and potential users. This document can also be used as a reference for future implementation of add on functionalities. The main purpose of this document is to present ELO's development team's understanding of the client's needs, while also providing a smooth transition process for future development teams who are going to take over the project. The requirements stated in this document are tentative and can be changed in the future. This requirement specification document also reaffirms the nature of the project: this web application is a prototype version and cannot go into production right after the semester ends.

### 2.2. Objectives and Scope

ELO is a web application that allows Gettysburg College Students to buy, sell, and rent textbooks at affordable prices. The project aims to help the students with the surging textbook prices by being a platform that hosts affordable textbook options. In this prototype version, the application is going to be specific to Gettysburg College uses only.

### 2.3. System Design

The system of ELO will consist of a web portal. The users will need to register to ELO to utilizes features such as buying, selling or renting books. For the purpose of recording user data for better customer service, a database will be used within the system. The web portal will communicate with the database and allow users to retrieve, add, and modify their data. All of the communication with the database requires a secured internet connection.

### 2.4. User Characteristics

There are three different types of users for the web application. One of them is Gettysburg College student who can be an actor in different cases like buying a book, renting a book, or selling a book. Second type of user is the administrator to facilitate the smooth operation of the web portal and integrate different parts of the system. Third kind of users can be stakeholders who are interested in the project.

### 2.5. Technical Constraints

**Wireless connection:** The internet application is a constraint for the ELO app. Since the use of application requires retrieval of data stored on database, it is crucial to have a secure internet connection.

**Possible platforms:** Listed below are the open-source e-commerce and CMS management platform that the development team is considering in production:

- *Magento*: installation requires a php-limit change due to its large size. Modules integration are unrealistic since each module costs an amount of money. *Magento* has a complicated documentation and lots of dependencies.

- *WordPress*: easy to navigate and is a huge help in UI elements and UI design. However, WordPress is more robust in blogging websites than for ecommerce sites.
  - *Joomla*: easy installation and has a lot of free plug-ins to guarantee the functionality aspects of the application. It is also said to be robust for ecommerce platforms.
- Note:** We have decided to move forward with Joomla.

### 3. Functional Requirements

#### 3.1. F-01: Log-in

##### Brief Description

This use case describes how a user logs in to ELO. The actor is a GC student

##### Basic Flow

1. The system requests the actor to enter his/her GC email address and password
2. The actor types in his/her GC email and password
3. The system validates the correct email and password and logs the user into the system

##### Alternative Flow

1. Invalid email: If the actor entered a non GB email, or an incorrect GB email the system displays an error message. The actor can choose to try logging in again (return to basic flow) or cancel log in and end use case.
2. Invalid password: If the actor entered a password that does not meet specified requirements, the system displays an error message and ask the user to re-enter his/her password.

##### Pre-conditions

Actor has to have an existing ELO account created with a valid GC's email address

##### Activity Diagram

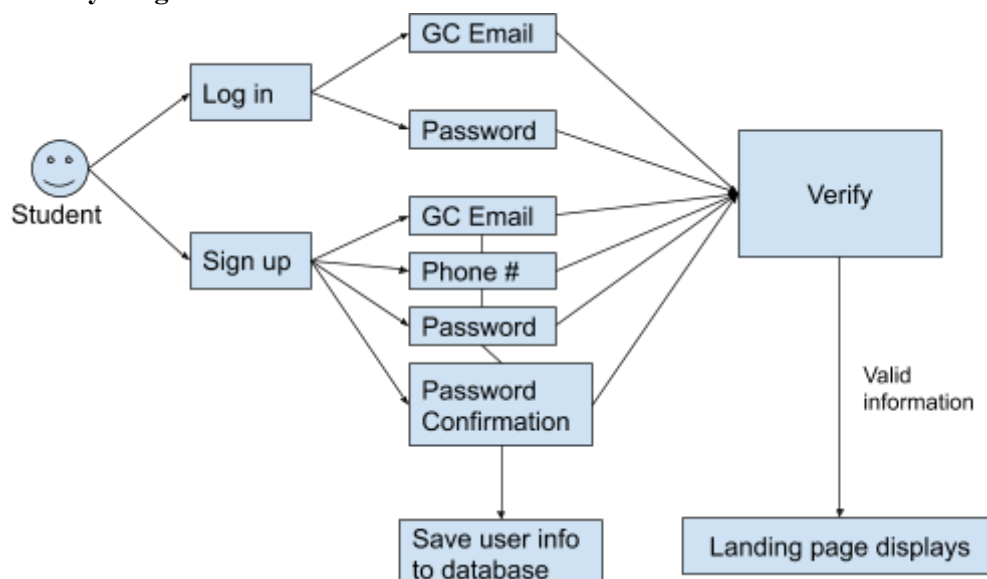


Figure 3.1.1: Activity Diagram for log-in and sign up

##### Screen & Element Details

http://projectcapstone.sites.gettysburg.edu

# ELO

**Log in**

1 Gettysburg College email address

2 Password

3 Log in

4 [Forgot your password?](#)

Or

**Sign up**

5 Gettysburg College email address

6 Phone number

7 Password

8 Confirm Password

9 Sign up

Screen 3.1.1: User interface for log-in and sign up

#	Element	Type	Description
1	Gettysburg College email address (log in section)	Input field	Required. Must verify correct + valid Gettysburg College email address
2	Password (log in section)	Input field	Required. Contain at least one (1) character from three (3) of the following categories: <ul style="list-style-type: none"> <li>• Uppercase letter (A-Z)</li> <li>• Lowercase letter (a-z)</li> <li>• Digit (0-9)</li> <li>• Special character (~`!@#\$%^&amp;*()+= _-{}[]\ :;'",?/&lt;&gt;.,)</li> </ul>
3	Log in	Button	Verify user's information and decide to log in or not
4	Forgot your password	Link	Redirect to forgot your password/reset password page
5	Gettysburg College email address (sign up section)	Input field	Required. Must verify correct + valid Gettysburg College email address
6	Phone number	Numeric field	Optional
7	Password (sign up section)	Input field	Required. Contain at least one (1) character from three (3) of the following categories: <ul style="list-style-type: none"> <li>• Uppercase letter (A-Z)</li> </ul>



			<ul style="list-style-type: none"> <li>• Lowercase letter (a-z)</li> <li>• Digit (0-9)</li> <li>• Special character (~`!@#\$%^&amp;*()+=_ -{}[] :;''"/&lt;&gt;.,)</li> </ul>
8	Confirm password	Input field	Must be similar to Password
9	Sign up	Button	Save user's information + create account in database

Table 3.1.1 : Screen details for log-in and sign-up

### 3.2. F-02: Sign-up

#### Brief Description

This use case describes how a user will sign up to ELO. The actor is a GC student

#### Basic Flow

1. The system requests the actor to enter his/her GC email address, phone number, password, confirm password
2. The actor types in his/her GC email, phone number, password, and confirm password
3. The system validates the valid GC email, password, phone number, save new actor's data and send confirmation email on successful account registration.

#### Alternative Flow

1. Invalid GB email: If the actor entered a non GB email, or an incorrect GB email, the system displays an error message. The actor can choose to re-enter email address and try signing up again.
2. Invalid password: If the actor entered a password that does not meet specified requirements, the system displays an error message and ask the actor to re-enter his/her password.
3. Unmatched password confirmation: if the actor enters a password confirmation that does not match with the entered password, the system displays an error message and ask the actor to confirm password again.
4. Letters input into number field: if the actor enters letters into the phone number field, the system displays an error message and ask the actor to re-enter his/her password.
5. Missing information in one (or more) fields: if the user does not enter one of the fields mentioned above, the system displays an error message and allow the user to enter information in the missing field(s).

#### Pre-conditions

Actor must not have an already existing ELO account

#### Activity Diagram

Can be found in F-01. Refer to Figure 3.1.1

#### Screen & Element Details

Can be found in F-01 since F-01 and F-02 are sharing the same screen. Refer to Table 3.1.1

### 3.3. F-03: Buy Book

#### Brief Description

This use case describes how a user would purchase a book in ELO. The actor is a GC student.

#### Basic Flow

1. The actor navigates to the side bar and click on "Buy a book"
2. The system displays form to buy a book
3. The actor fills in the following information: the title of the book that they want to rent, the edition, the author and the ISBN (optional), and click on "Find your book"
4. The system displays all available book options that matches the actor's request

5. The actor choose the book
6. The actor clicks on “Buy this book”. The book is added to cart. Use case end.

#### Alternative Flow

1. Missing information in one (or more) fields: if the user does not enter one of the fields mentioned above, the system displays an error message and allow the user to enter information in the missing field.
2. No book available: if the book which the actor is looking for is not available, the system redirects to “Book Not Found” page that asks the actor to double check the spelling of the entered fields, or allow actor to search again (return to basic flow), allow the actor to return to home page, or give options for the student to find the book in the library, the college’s bookstore or Amazon.

#### Pre-conditions

Actor must have an ELO account

#### Activity Diagram

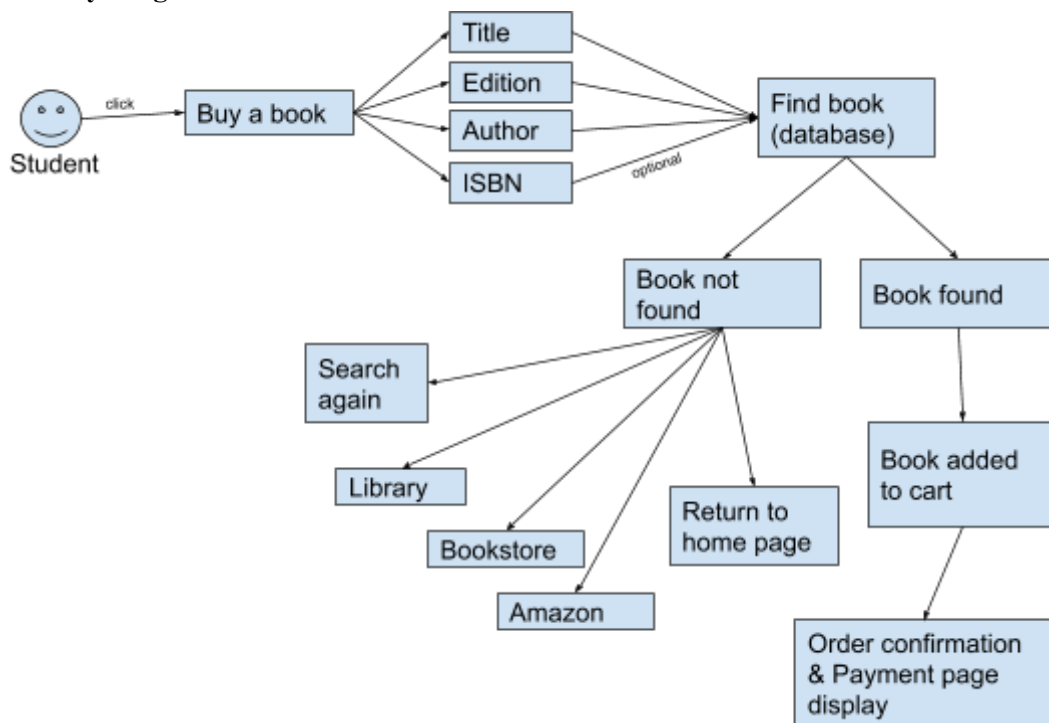
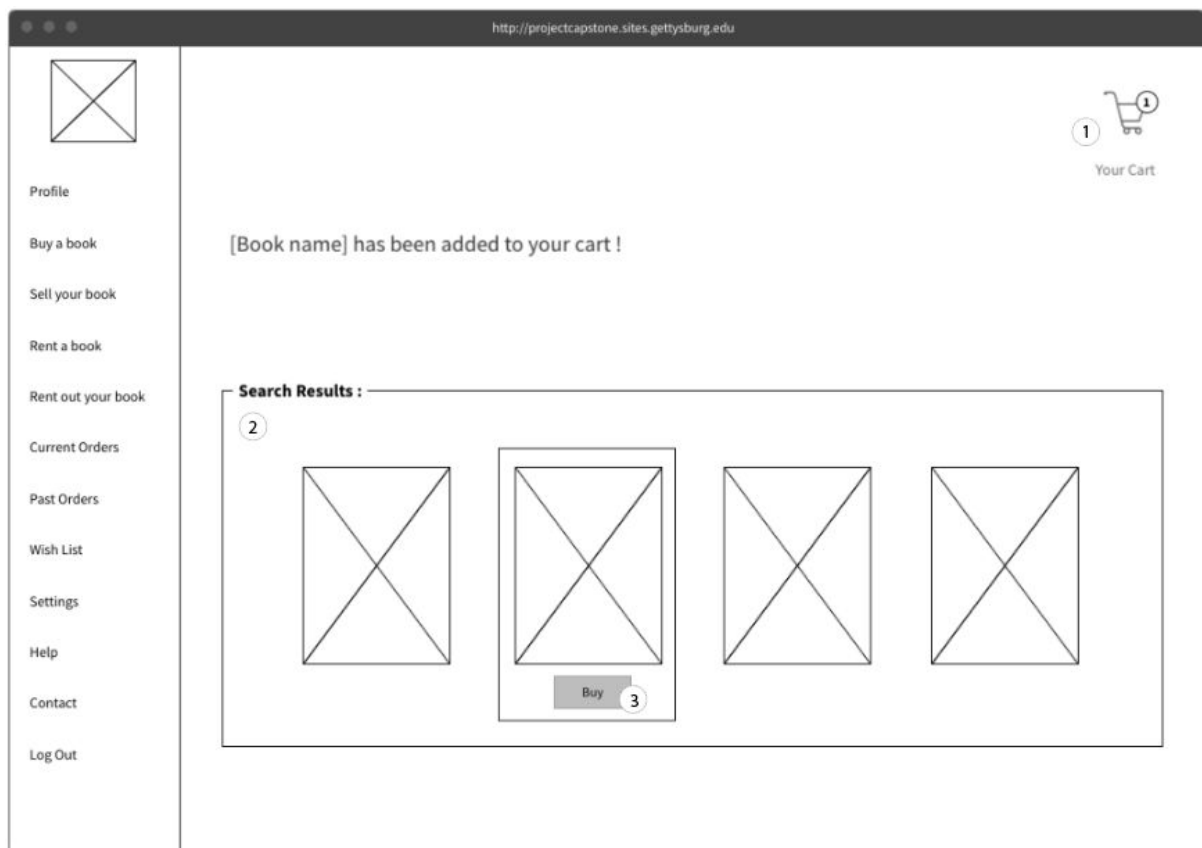


Figure 3.4.1: Activity diagram for buying book

#### Screen & Element Details

User interface of the Buy book form is very similar to that of the Rent book form, therefore can be found in F-03.



Screen 3.4.1: Book added to cart page

#	Element	Type	Description
1	Cart icon	Clickable Icon	Show the number of items currently in cart. On click will redirect the actor to cart page.
2	Search result	Clickable images group	Displays cover photos of available books, and allows actor to choose books
3	Buy	Button	Allows the actor to pick his/book. Increment the number of items in cart.

Table 3.4.1: Screen details for book added to cart page

### 3.4. F-04: View Cart & Check Out

#### Brief Description

This use case describes how a user will view his/her shopping cart. The actor is a GC student.

#### Basic Flow

1. The actor clicks on “Your Cart” icon
2. The system displays Your Cart page.
3. The actor review’s order information, modify quantity, enter payment information and click “Buy”, use case ends.

#### Alternative Flow

1. Missing information in one (or more) fields: if the user does not enter one of the fields mentioned above, the system displays an error message and allow the user to enter information in the missing field.

- Invalid payment information: if any of the information the actor enters is deemed invalid, the system shows an alert and allow the actor to re-enter incorrect information.

### Pre-conditions

The actor must have an ELO account, must be logged in.

### Activity Diagram

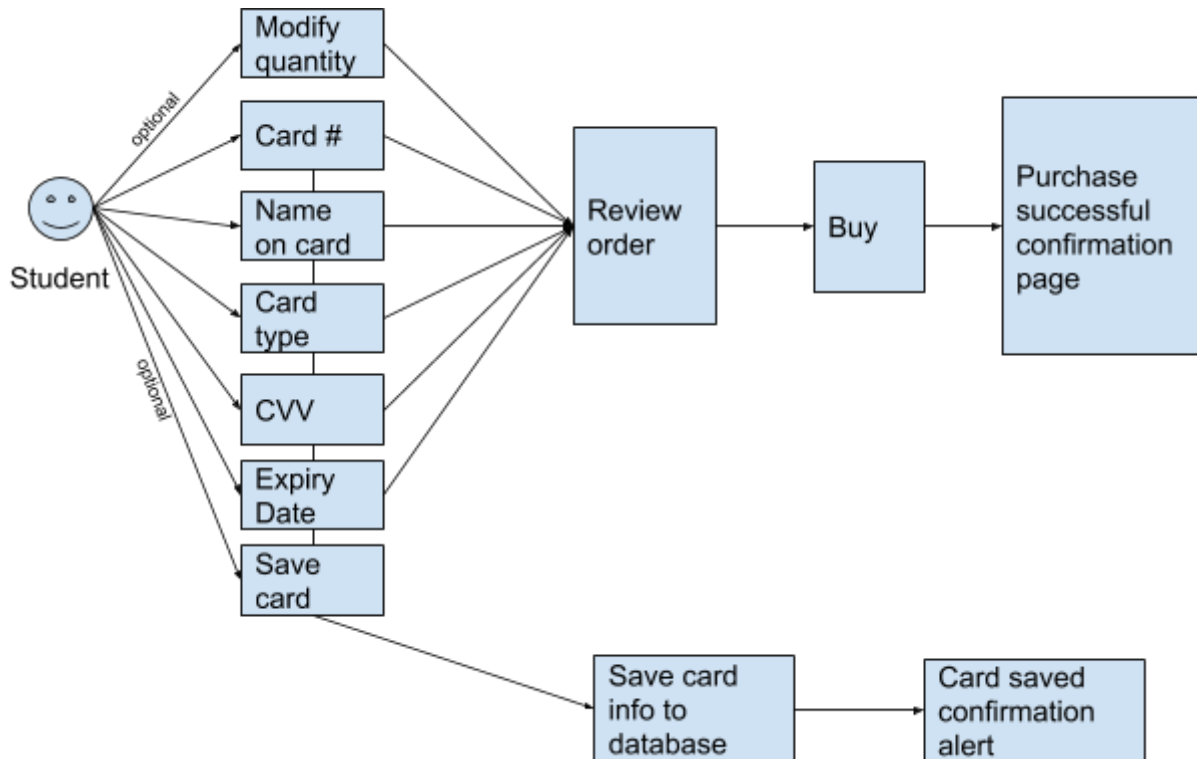
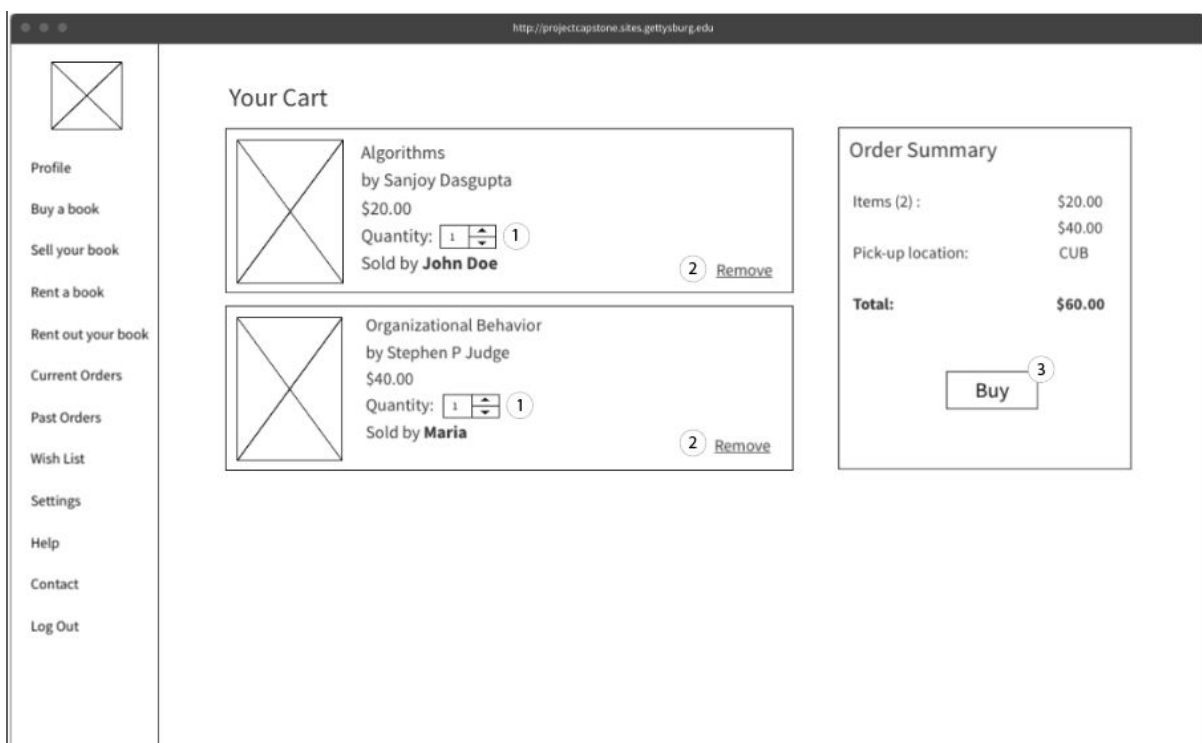


Figure 3.5.1: Activity diagram for view cart and checkout

### Screen & Element Details



Screen 3.5.1 : Screen for cart page

	Element	Type	Description
1	Quantity	Numeric spinner	Range 1-20.
2	Remove	Hyperlink	Remove the item associated with this link
3	Buy	Button	Confirm actor's purchase + send confirmation email + in app push notification + redirect to purchase successful confirmation page (+ integrate payment)

Table 3.5.1: Screen details for Your cart page

### 3.5. F-05: Remove from Cart

#### Brief Description

This use case describes how a user will remove an item from his/her shopping cart. The actor is a GC student.

#### Basic Flow

1. The actor clicks on "Remove" in the product box
2. The system displays a message announcing the item has been removed and displays the remaining items, use case ends.

#### Alternative Flow

None

**Pre-conditions:** Actor must have at least one item in cart.

#### Activity Diagram

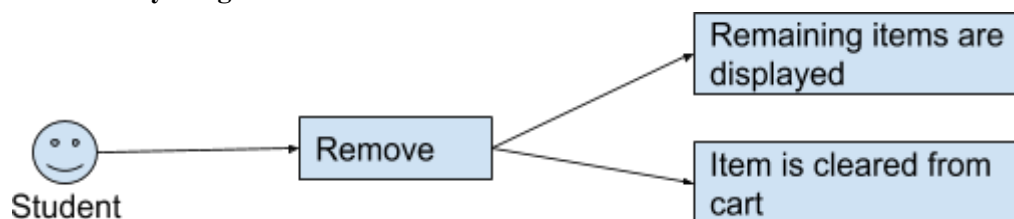
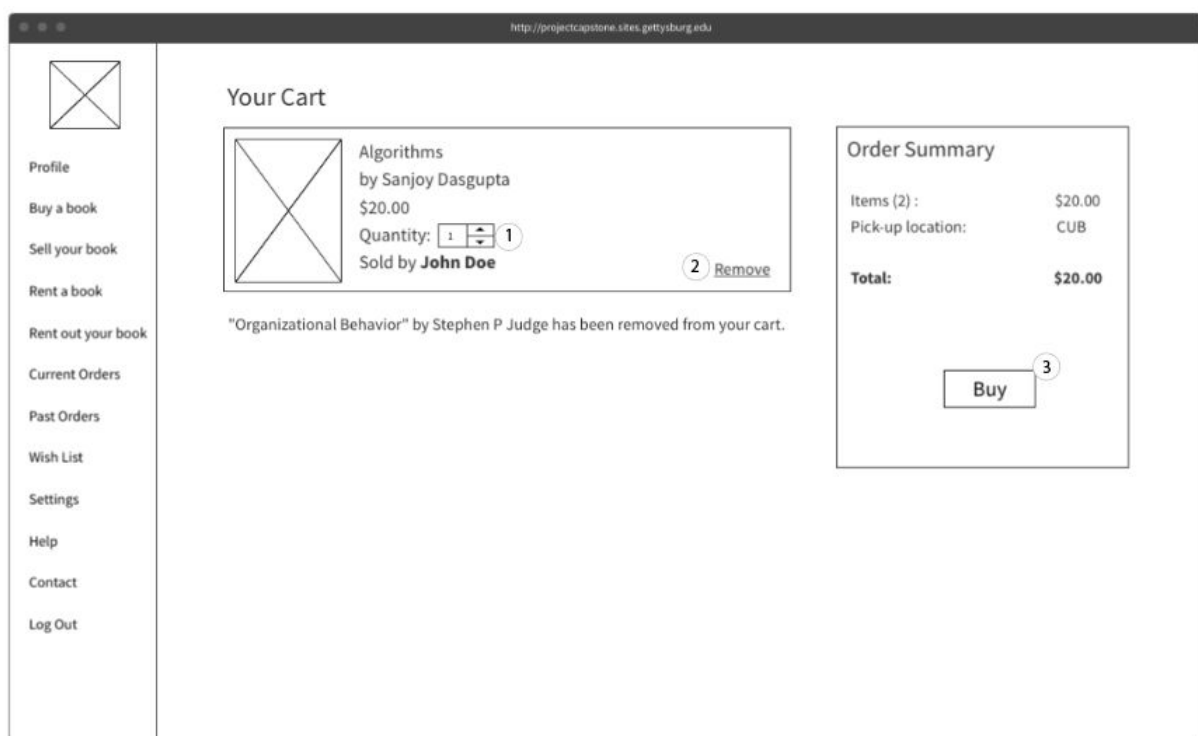


Figure 3.6.1: Activity diagram for removing items from cart

### Screen & Element Details



## Screen 3.6.1: Item removed from cart

Element details are similar to those of F-05: View Cart & Check Out. Refer to Table 3.5.1.

### 3.6. F-06: Sell Book

#### Brief Description

This use case describes how a user will sell his/her book on ELO. The actor is GC student.

#### Basic Flow

1. The actor navigates to the sidebar and click on “Sell your book”
2. The system displays the sell your book form
3. The actor fill in the following fields: book title, book edition, book author, ISBN, book condition, book price, pick up location, and photos of book.

#### Alternative Flow

1. Missing information in one (or more) fields: if the user does not enter one of the fields mentioned above, or did not upload any photos, the system displays an error message and allow the user to enter information in the missing field.
2. If actor wants to cancel the selling process, simply navigate to other buttons on the sidebar.

#### Pre-conditions

User must have a registered ELO account.

#### Activity Diagram

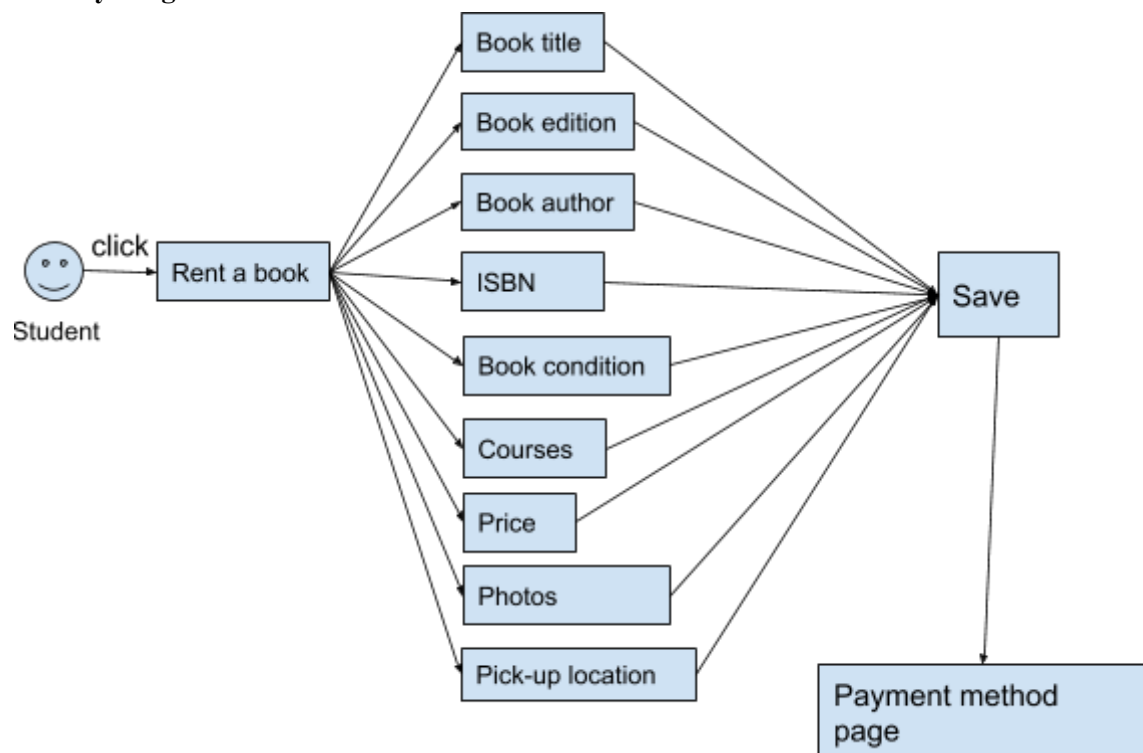


Figure 3.8.1: Activity diagram for selling a book

#### Screen & Element Details

Screen 3.8.1 : User interface for selling book

#	Element	Type	Description
1	Book Title	Text field	Required
2	Book Edition	Input field	Required
3	Book Author	Text field	Required
4	ISBN	Numeric field	Required
5	Book Condition	Select Box	Required. There will be two options in the select box: "New" and "Used". Option "Used" will have a drop down of the following options: "Like New", "Very Good", "Good", "Acceptable"
6	Which courses	Input field	Required
7	Book Price	Numeric field	Required
8	Pick-up Location	Select Box	Required. Options to pick-up are GC's hotspots : CUB, Library, Glatfelter, or Coordinate with Buyer via messaging.
9	Photos of Book	Image gallery	Required. At least 7 photos is required.
10	Save and Continue	Button	Save the book information provided by the user in the database and redirect to payment page.

Table 3.8.1: Screen details for Sell your book page

### 3.7. F-07: Create Profile

#### Brief Description

This use case describes how a user create profile in ELO. The actor is the user ( GC student)

#### Basic Flow

After signing up and logging in the first time, it will automatically lead to the create profile information page.

1. The system asks the actor to select the class year
2. The system asks the actor to select their Major / Minor
3. The system asks the actor to select the courses that they are taking.

Actor can click on the “Later” button, to skip this step and use the other function.

The profile information can be edited in F-11.

#### Alternative Flow

1. Can not find the class year
2. Can not find the Major/ Minor
3. Can not find the course

In all three cases, go to the troubleshooting page (F-16) to report the problem.

#### Pre-conditions

User must have a registered ELO account.

#### Activity Diagram:

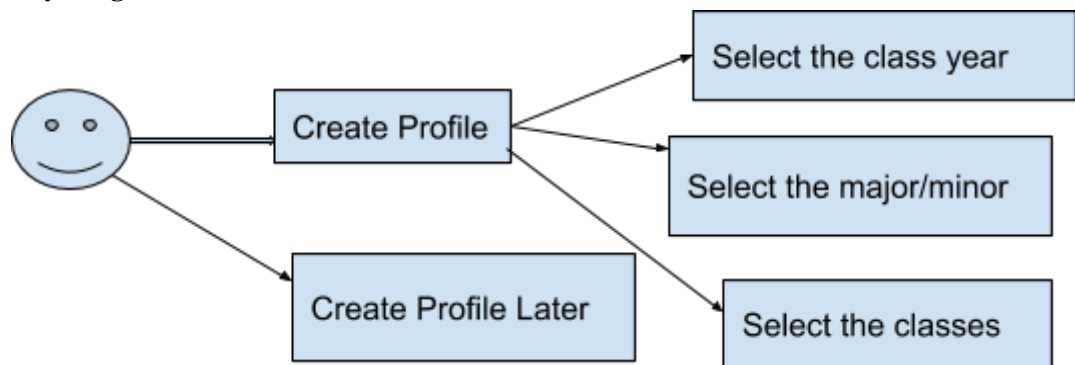


Figure 3.9.1: Activity diagram for creating profile

#### Screen & Element Details:



Select Class Year:

Select Major/Minor:

Major: Computer Science

Minor:

Select Course:

Course: CS440

Course:

Figure 3.9.1: User interface for creating profile

#	Element	Type	Description
1	Class Year	Select Box	Options are the current year up to 4 years after
2	Add Major/Minor	Button	After clicking the button, there will be 2 select box showed up, one let the user choose Major/minor. The other let the user choose the specific major/minor.
3	Add Courses	Button	After clicking the button, there will be 2 select box showed up, one let the user choose the field of the course, and the other let the user choose the specific course.
4	Later	Button	Let the user skip the create profile step.

Table 3.9.1: Screen details for Creating Profile

### 3.8. F-08: Log Out

#### Brief Description

This use case describes how a user log out from their account in ELO. The actor is the user (GC student)

#### Basic Flow

The user click on the log out button, and return to the app as a guest.

#### Alternative Flow

If the user doesn't log in, the log out button will not appear.

#### Pre-conditions

The user is logged in his/her account.

#### Activity Diagram

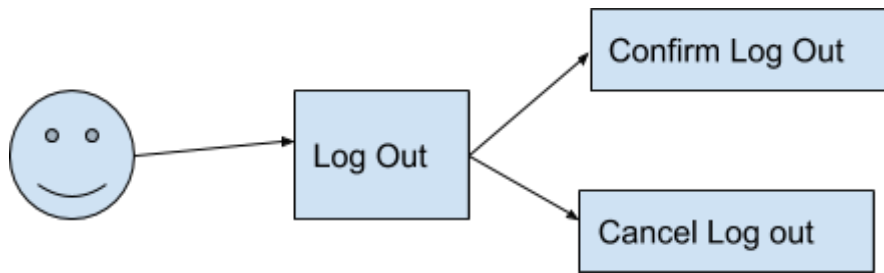


Figure 3.10.1: Activity diagram for log out

#### Screen & Element Details



Figure 3.10.1: User interface for Log out

#	Element	Type	Description
1	Log Out	Button	After clicking it, it will have a confirmation window showed up, and ask if the user confirm to log out.
2	Yes	Button	The user will log out from their account
3	No	Button	The user will go back to the page they are previous at.

Table 3.10.1: Screen details for logging out

### 3.9. F-09: Change Profile Information

#### Brief Description

This use case describes how a user changes their profile information in ELO. The actor is the user (GC student)

#### Basic Flow

Click on the profile, at the bottom will have a button, "change your profile information, and after click that it will lead you to the change information page.

There will be a list of things that can be changed.

1. The phone number
2. The class year
3. The major/minor
4. The course they are taking

The e-mail address can not be changed because it is used as a log in way. The password can be reset in F-23.

#### Alternative Flow

1. Can not find the class year
2. Can not find the Major/ Minor
3. Can not find the course

In all three cases, lead to the troubleshooting page (F-16) to report the problem.

4. The user enters the wrong phone number, it may contain more numbers, wrong than a phone number. The user can not confirm the change of the profile and have an error warning that the format of the phone number is incorrect.

#### Pre-conditions

The user is logged in his / her account.

#### Activity Diagram

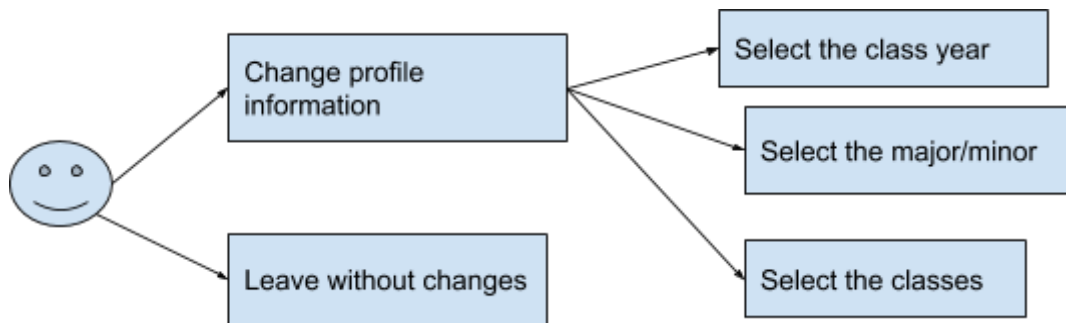


Figure 3.11.1: Activity diagram for change profile information

#### Screen & Element Details

Select Class Year:
 

Select ▼

Select Major/Minor:
 

Add Major

Add Minor

Major: Computer Science
 

Delete

Minor:
 

Select ▼

Delete

Select Course:
 

Search by department

Course: CS440
 

Delete

Course:
 

Select ▼

Delete

Save Changes

Back Without Saving

Screen 3.11.1: User interface for editing profile

#	Element	Type	Description
---	---------	------	-------------

1	Edit profile	Button	Take the user to the profile editing page
2	Class Year	Select Box	Options are the current year up to 4 years after
3	Add Major/Minor	Button	After clicking the button, there will be 2 select box showed up, one let the user choose Major/minor. The other let the user choose the specific major/minor.
4	Add Courses	Button	After clicking the button, there will be 2 select box showed up, one let the user choose the field of the course, and the other let the user choose the specific course.
5	Save changes	Button	Let the user skip the create profile step.
6	Back without saving	Button	Let the user give up the changes and go back to the profile page.

Table 3.11.1: Screen details for Editing Profile

### 3.10. F-10: In App Messaging

#### Brief Description

This use case describes how a user send and receive messages in ELO. The actor is the user (GC student)

#### Basic Flow:

The user can message another user or receive message from another user after two users are involved in a transaction.

#### Alternative Flow:

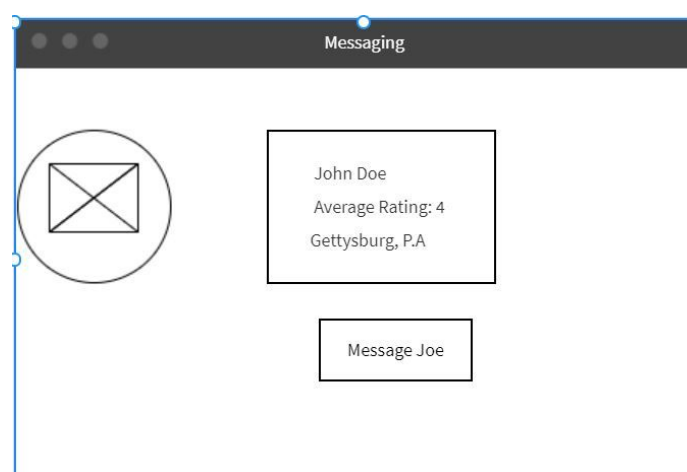
A user cannot message another user if he/she is not involved in a transaction(has not confirmed of buying/renting/selling book) with another user.

#### Pre-conditions

The user must be registered and should have an active transaction.

#### Activity Diagram

#### Screen & Element Details



Screen 3.12.1: User interface for a in app messaging functionality

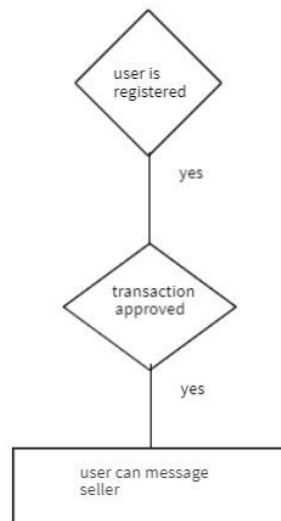


Figure 3.12.1: Activity Diagram for InApp Messaging functionality

#	Element	Type	Description
1	Send	Button	Send the message

Table 3.12.1: Screen details for In App Messaging

### 3.11. F-11: View Current Orders

#### Brief Description

This use case describes how a user view their own current orders in ELO. The actor is the user (GC student)

#### Basic Flow

The current order are the orders that are in process, and the user can see the status of the order, the date of the order, the book is waiting to be transferred between students, etc. The identity of the user will be clarified, as a buyer/borrower or seller/renter. And there will be buttons that can confirm transfer, cancel order, report a problem of the order, etc.

#### Alternative Flow

The order is not show up correctly in the page, report the problem through the troubleshooting(F-16).

#### Pre-conditions

The user is logged in his/her account.

#### Activity Diagram

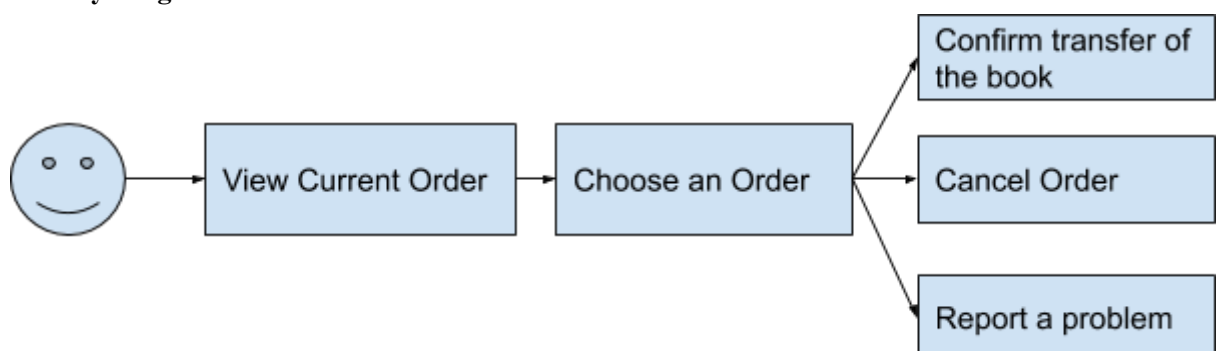


Figure 3.13.1: Activity diagram for view current order

**Screen & Element Details**

#	Element	Type	Description
1	Confirm Transfer	Button	Confirm you rent out / receive the book
2	Cancel Order	Button	Lead to cancel order page
3	Report a Problem	Button	Lead to F-16

Table 3.13.1: Screen details for Current Orders

**3.12. F-12: View Past Orders****Brief Description**

This use case describes how a user view their own past orders in ELO. The actor is the user (GC student)

**Basic Flow**

The past order are the orders that are completed, and the user can see the status of the order, the date of the order, the book pick up date, if the book is rent, the return date, etc. The identity of the user will be clarified, as a buyer/borrower or seller/renter. And there will be buttons that can contact the buyer/borrower or seller/renter, report a problem of the order, etc.

**Alternative Flow**

The order is not show up correctly in the page, report the problem through the troubleshooting(F-16).

**Pre-conditions**

The user is logged in his/her account.

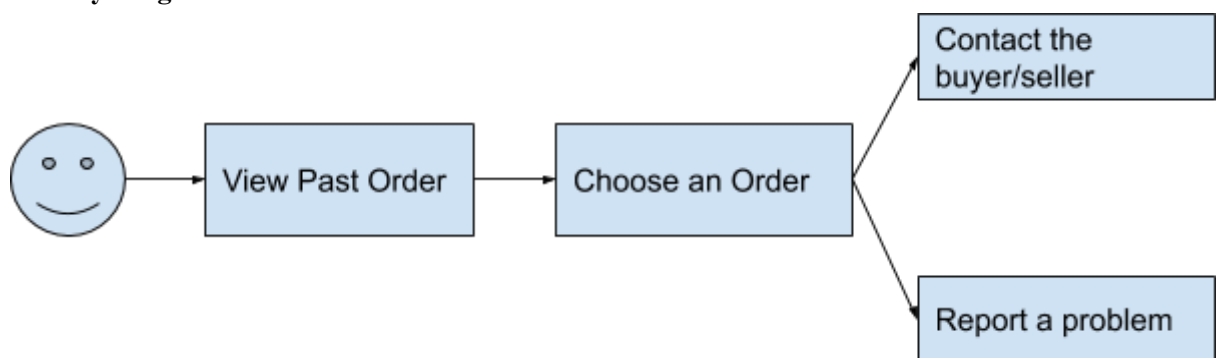
**Activity Diagram**

Figure 3.14.1: Activity diagram for view past order

**Screen & Element Details**

#	Element	Type	Description
1	Contact	Button	Contact the buyer/borrower or seller/renter
2	Report a problem	Button	Lead to F-16

Table 3.14.1: Screen details for Past Orders

**3.13. F-13: View WishList****Brief Description**

This use case describes how a user view their wishlist in ELO. The actor is the user (GC student)

**Basic Flow**

1. The user clicks on the “Wishlist” button on the side bar.

2. The wish list appears, has a list of books, which each contains book picture, title, edition, author, ISBN number, whether the book is still available and the suggested price (lowest price).
3. If the user clicks the book, it will redirect to rent/buy book page.
4. If the user clicks the remove button, it will remove the book from the wishlist.

#### Alternative Flow

1. If the user did not add any book to wish list, it will show that there's nothing in the wish list.

#### Pre-conditions

The user is logged in his/her account.

#### Activity Diagram

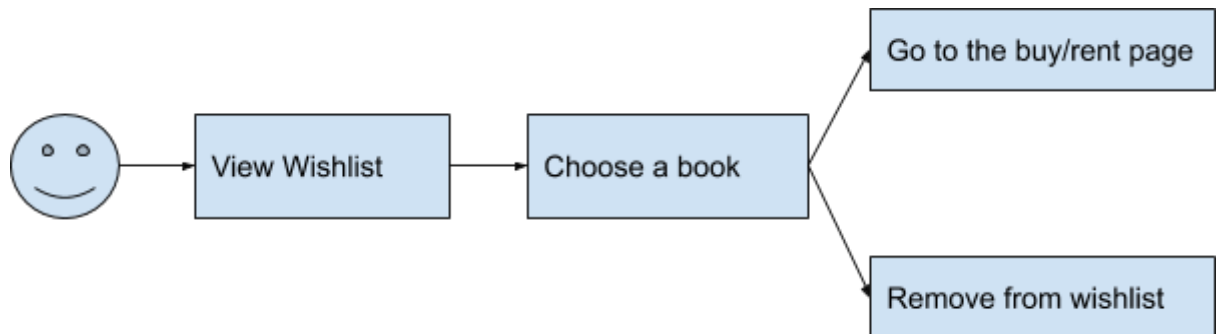
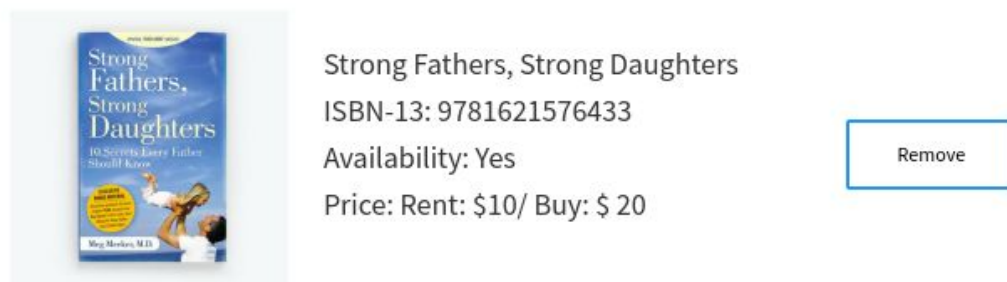


Figure 3.15.1: Activity diagram for view wishlist

#### Screen & Element Details



Screen 3.15.1: User interface for WishList

#	Element	Type	Description
1	Book	Button	Redirected to the book's rent/buy page
2	Remove	Button	Remove the book from the wishlist

Table 3.15.1: Screen details for Wishlist

### 3.14. F-14: Troubleshooting & Reports Complaints

#### Brief Description

This use case describes how a user reports problems and troubleshooting in ELO. The actor is the user (GC student)

#### Basic Flow

1. Click on report a problem.
2. Select from a list of functions, such as find a book , buy a book, edit profile, etc.
3. Type in the detail information of the problem.
4. Click send, and have the page

**Alternative Flow****Pre-conditions**

The user is logged in.

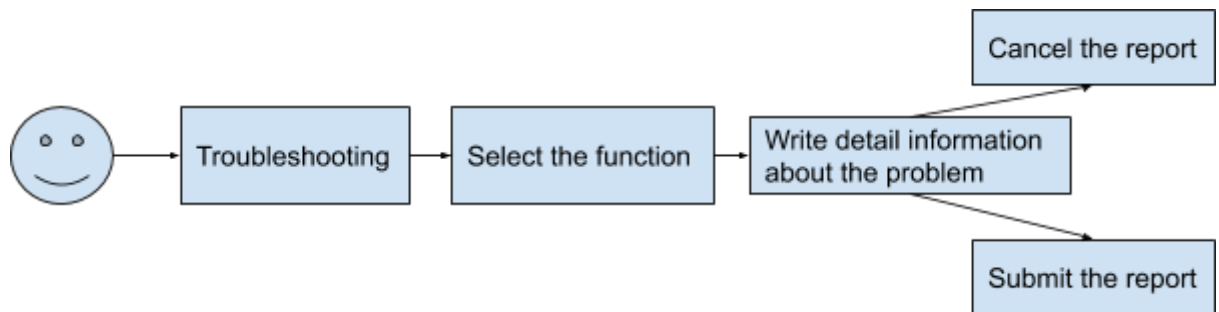
**Activity Diagram**

Figure 3.16.1: Activity diagram for troubleshooting

**Screen & Element Details**

Function you want to report:

Detail:

Screen 3.16.1: User interface for Troubleshooting

#	Element	Type	Description
1	Function you want to report	Select Box	A list of functions in the app, and an “other function”
2	Detail	Text Input	Enter the details of the problem

Table 3.16.1: Screen details for Troubleshooting

### 3.15. F-15: Search bar, Sorting & Filter

**Brief Description**

This use case describes how a user can search for a book item in the search bar with sorting and filtering option. The actor is a Gettysburg College student.

**Basic Flow**

1. The user types the name of the item he, she is looking for in the search bar.
2. The system runs query to find the item or similar item mentioned in the search bar.
3. After the results are displayed, the user can click on sort alphabetically or filter by price range or condition options.



### Alternative Flow

1. If the user enters an invalid entry or user enters the name of the book that the database cannot find, the user will be told that no results are found, and can choose to be notified later if the book is available in the future.
2. In case no results were found the screen displays other options like bookstore, library that the user can try.

**Pre-conditions: None**

### Activity Diagram

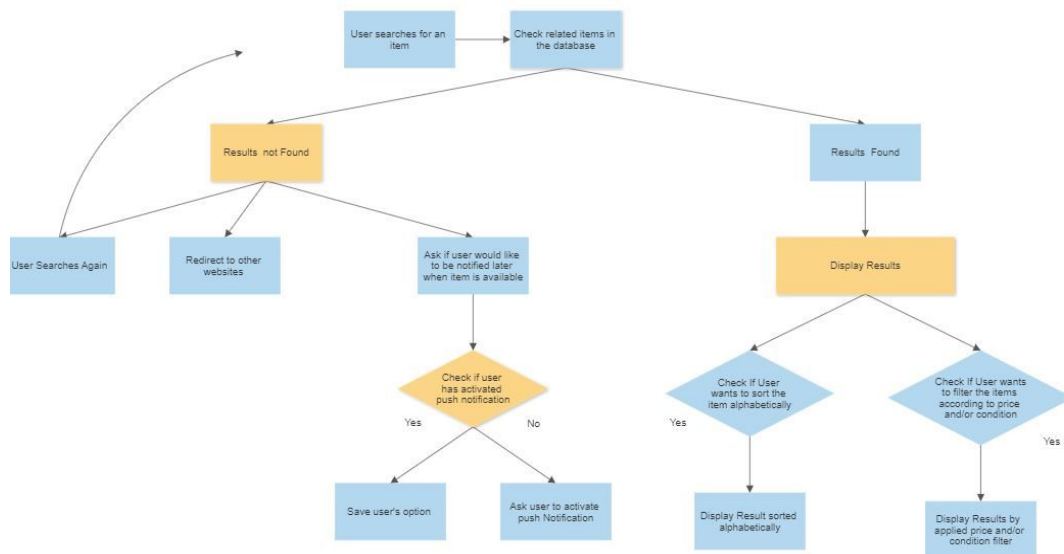


Figure 3.17.1 :Activity Diagram for search functionality with sorting and filter options.

### Screen & Element Details

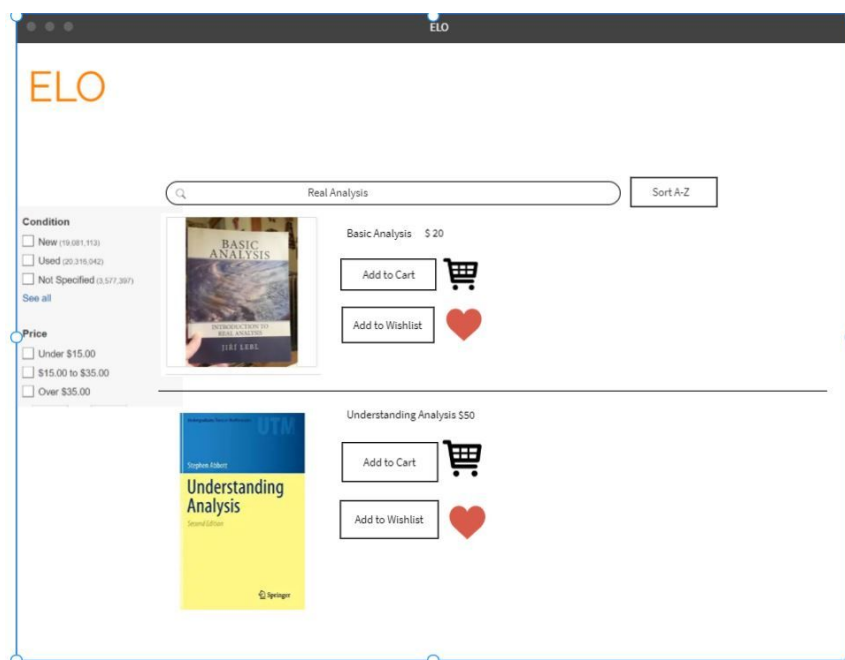


Figure 3.17.11:Screen Sample for search with sort and filter functionality

#	Element	Type	Description
1	Search Bar	Search	The user searches for the item by typing the name of the item as an input to the search bar.
2	Sort A-Z	Button	Upon display of results clicking this button will display all the names of items sorted alphabetically.
3	Add to Cart	Button	Add an item to the user's shopping cart
4	Add to Wishlist	Button	Add an item to the user's wishlist
5	Condition	Checkbox	Filter results according to used condition, new condition, or both if both checkboxes are marked.
6	Price	Checkbox	Filter results by price range.

Table 3.17.1: Screen Element Details for search interface

### 3.16. F-16: Contact

#### Brief Description

This use case allows user to send a message to the ELO team. The actor is Gettysburg College student.

#### Basic Flow

1. The user will be asked to enter their information like name, address, email, and his/her message.
2. The user can then click submit button to send their message.

#### Alternative Flow

1. In case the user enters invalid information, for example invalid email or forgets to put his/her name the user will be prompted to fix the errors until all the information entered is valid.

**Pre-conditions:** None

#### Activity Diagram

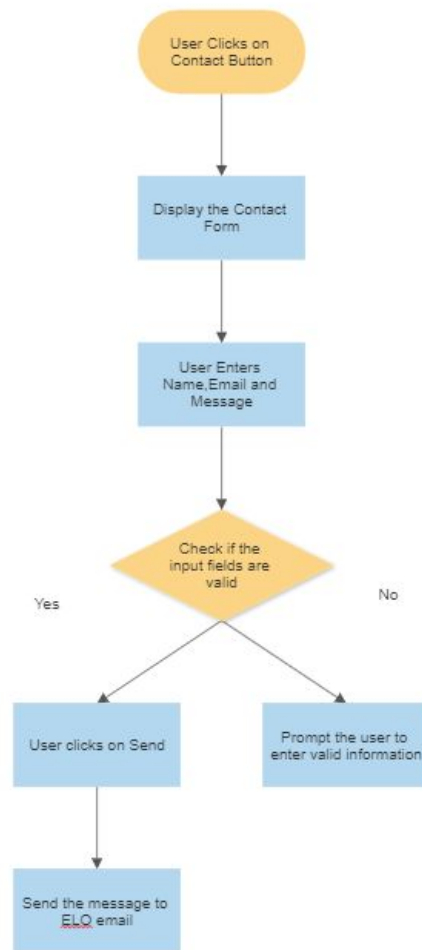


Figure 3.18.1: Activity diagram for Contact

### Screen & Element Details

The user interface for the contact screen features a header with the title "Contact Us" and social media icons for Facebook, Twitter, and Google+. Below the header, there are three input fields: "Your name", "Your mail", and "Your message". The "Your message" field is a text area with a red underline and a double-slash icon. A "SEND" button is located at the bottom right of the form.

Screen 3.18.1: User interface for contact screen

#	Element	Type	Description
1	Name	HTML Input - text	Name of User name - Required
2	Email	HTML Input- text	User email -Required
3	Message	HTML Input -Text	The message user wants to send - Required, and must be non empty
4	Send	HTML Input -Submit	Sends the message to the ELO email

Table 3.18.1: Screen Details for Contact Screen

### 3.17. F-17: Recommendation Section

#### Brief description

This use case is used to suggest books that the user might be interested in based on his/her shopping/browsing history. The actor is Gettysburg College student.

#### Basic Flow

1. If a user is registered with ELO, and has some buying/renting history, he/she shall see some book recommendations on the home screen.

#### Alternative Flow

1. If the user is not active, and has not performed any transaction on ELO, then the recommendation section will not be visible.

#### Pre-conditions

The user should have bought some items previously.

#### Activity Diagram

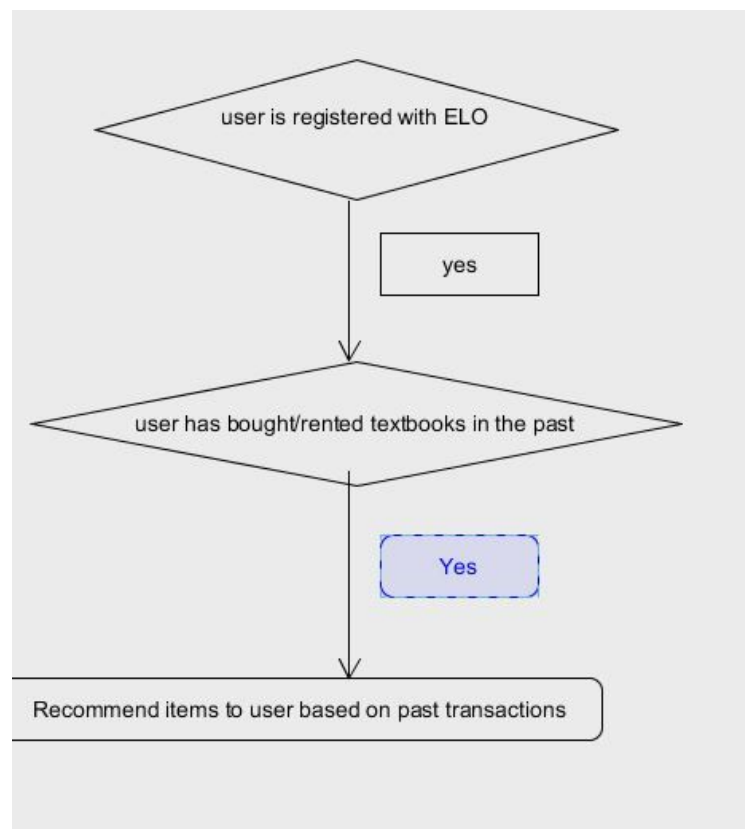
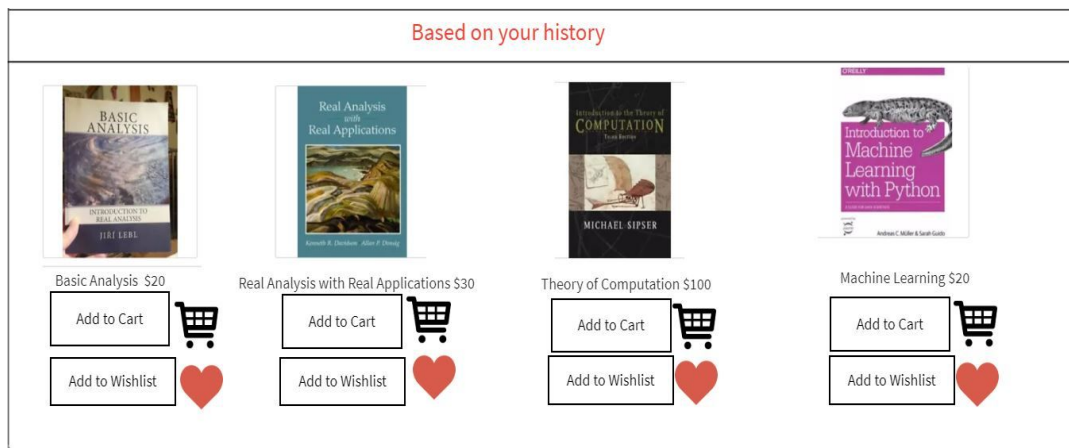


Figure 3.19.1 : Activity Diagram for Recommendation

### Screen & Element Details



Screen 3.19.1: Interface for Recommendation Screen

#	Element	Type	Description
1	Based on your history	Heading	Text
2	Add to Cart	Button	Adds the item to the user's shopping cart
3	Add to Wishlist	Button	Adds the item to the user's wishlist

Table 3.19.1: Screen details for recommendation screen

### 3.18. F-18: Push Notifications

#### Brief Description

The push notification functionality shall notify the user of alerts and messages on their phone.

#### Basic Flow

1. The user if he /she has activated the push notification function will receive notifications on his/her mobile device.

#### Alternative Flow

1. The user will not receive any outside of application notifications if he/she has not activated push notification service.

#### Pre-conditions

User must activate the push notification service.

#### Activity Diagram

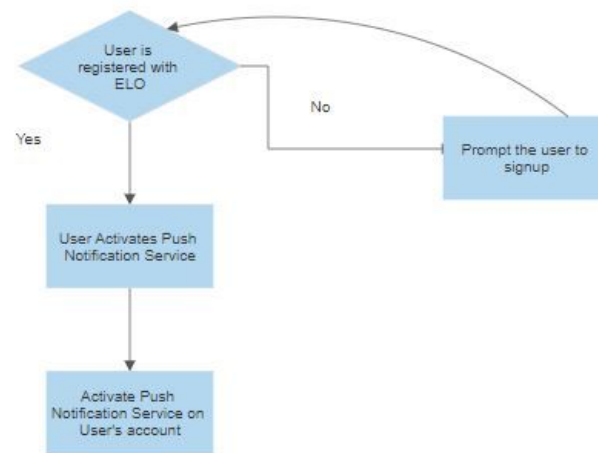
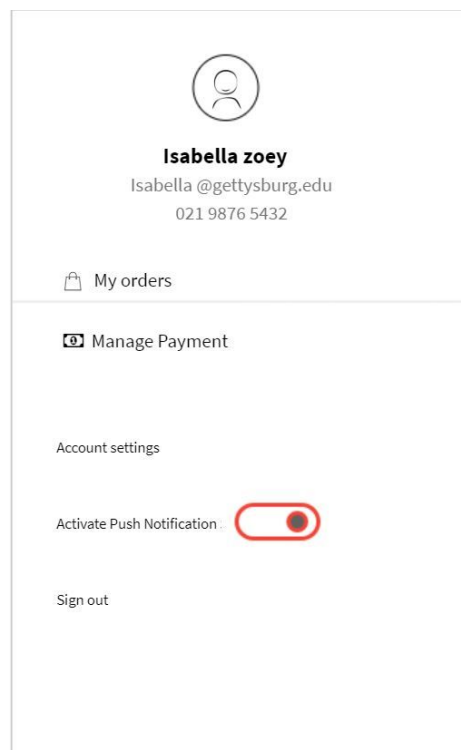


Figure 3.20.1: Activity Diagram for push notifications

### Screen & Element Details

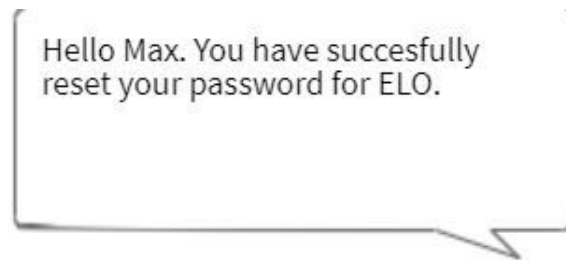


Screen 3.20.1: Interface for activating push notification screen

#	Element	Type	Description
1	My Orders	Button	Displays all the current and past orders of user
2	Manage Payment	Button	Allows user to manage their payment methods
3	Account Settings	Button	Allows user to configure current settings

4	Activate Push Notification	Button	Allows user to activate push notification service
5	Signout	Button	Allows user to sign out of web app

Table 3.20.1 : Screen details for push notification screen



Screen 3.20.1: Sample active push notification scenario

### 3.19. F-19: Manage payment method

#### Brief Description

This functionality shall allow user to add or delete debit or credit cards.

#### Basic Flow

1. Under settings, users can manage payment methods where they can add a payment method, change the default payment method or remove an existing payment method.

#### Alternative Flow

1. Users are not required to add their credit/debit card unless they want to make a transaction.

**Pre-conditions** None

#### Activity Diagram



Figure 3.21.1: Activity Diagram for managing payment

#### Screen & Element Details

Screen 3.21.1: Interface for Manage Payment Screen

#	Element	Type	Description
1	Remove	Button	Removes existing payment method
2	Another Payment Method	Heading	Text
3	Credit Card	Check Box	Lets the user add credit card as another payment method
4	Debit Card	Check Box	Lets the user add debit card from some bank as another payment method

Table 3.21.1: Screen details for manage payment screen

### 3.20. F-20: Make in app payments

#### Brief Description

This functionality shall allow users to make secure in app payment for a transaction.

#### Basic Flow

1. After the user clicks on 'Make Payment', the user will be directed to a secure site where they will make the payment.

#### Alternative Flow

1. If the payment method is not valid, the user will be prompted to make a valid payment method.

#### Pre-conditions

User must have payment method stored before making payment.

#### Activity Diagram



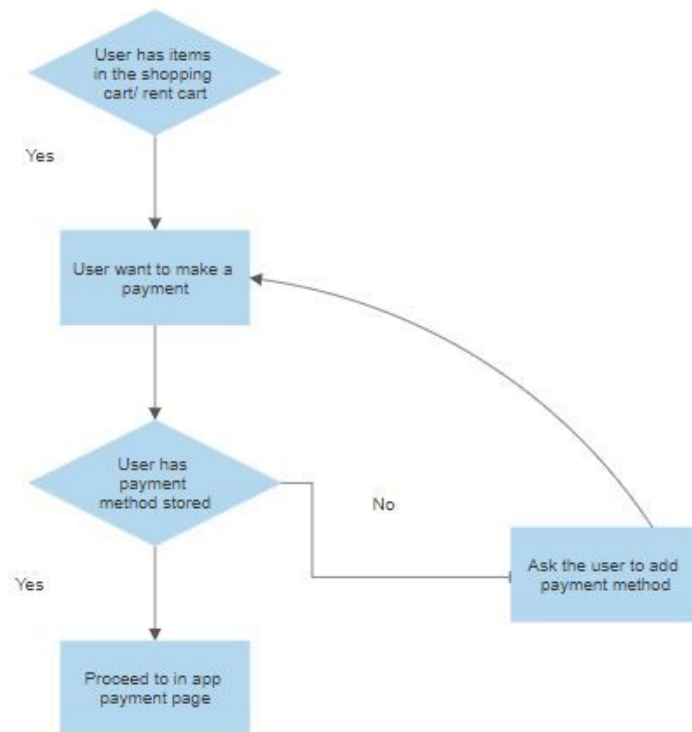


Figure 3.22.1 : Activity diagram for in-app payment

### Screen & Element Details

Pay With

**Your Saved Payment Method:**

☒ Visa ending with 0934  
 CVV:

Name on Card  
 Sam Smith

03/08/17

**Another Payment Method**

☐ Credit Card

☐ Debit Card

Total \$20

Confirm and Continue

Screen 3.22.1: Interface for in app payment

#	Element	Type	Description
1	Remove	Button	Remove the current payment method if the user has any
2	Credit Card	Check Box	Add a new credit card as payment method
3	Debit Card	Check Box	Add a new debit card as payment method

4	Total	Label	Text that shows the total amount due
5	Confirm and Continue	Button	Validates payment information and proceeds

Table 3.22.1: Screen Details for in app payment screen

### 3.21. F-21: Reset password / Forgotten Password

#### Brief description

This functionality shall allow users to change their current password.

#### Basic Flow

1. If the user wants to change their password then he/she enters the current password, enters the new password and reconfirms it before making the final changes.
2. The user receives an email upon password change and also mobile notification if he/she has opted for push notification service.
3. If the user has forgotten his/her current password, then they can reset their current password using their email which will be used to verify user's account before user can reset forgotten password.

#### Alternative Flow

1. If the user wants to change the current password, and does not enter the current password correctly, he/she will not be able to change the password.
2. In case of forgotten password, the user cannot change password without verifying the link sent by email.

#### Pre-conditions

The user must already have an account registered with ELO.

#### Activity Diagram

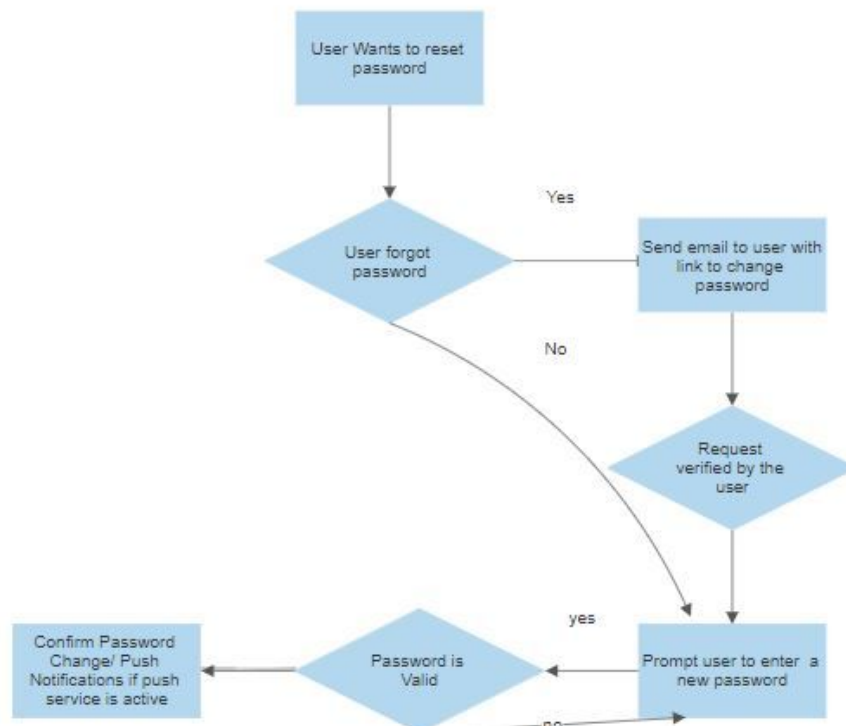


Figure 3.23.1: Activity Diagram for forgot or reset password

**Screen & Element Details**

Change Password

Current Password:

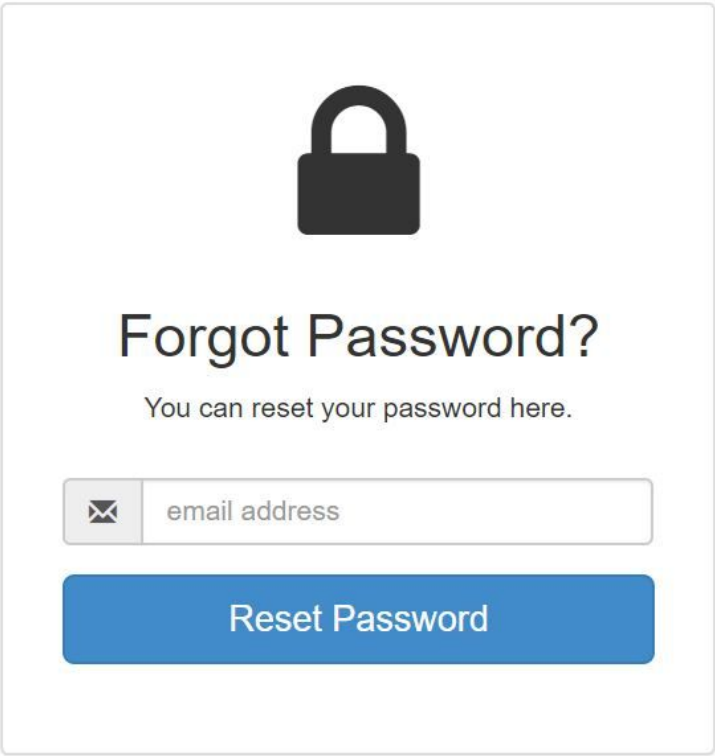
New Password:

Confirm New Password:

Screen 3.23.1: Screen Sample if user wants to change password

#	Element	Type	Description
1	Change Password	Label	Text
2	Current Password	Input	User's current Password
3	New Password	Input	User's new Password
4	Confirm New Password	Input	User's new Password
5	Confirm	Button	Saves the Changes

Table 3.23.1: Details for reset password



Screen 3.23.2: Interface for resetting forgotten password

#	Element	Type	Description
1	Email Address	Text input field	User’s email address
2	Reset Password	Button	Sends the link to the user's email for verification before allowing user to change password

Table 3.23.2: Screen details for resetting forgotten password

**Change Password**

New Password:

Confirm New Password:

Screen 3.23.3: Interface for change password after user verifies the link in email

#	Element	Type	Description
1	New Password	Input	User's new Password
2	Confirm New Password	Input	User's new Password
3	Confirm	Button	Saves the Changes

Table 3.23.3 : Screen details for change password screen

## 4. System Design

- Joomla, the content management system, will be used to hold our front-end of the application.
- Sellacious, the Joomla e-Commerce customizable marketplace software, that build in with multiple functions like, buy and sell books, search product, manage product, etc.
- Blast Chat, an extension that enable online live-chatting in our application
- Paypal, a third party online payments system, that help us process the transactions.
- An administrator account should implemented to be able to login to the system with credentials as an administrator.
- Internet connection is required while using the application for pushing and getting data from database.

## 5. Test Cases

### • Log-in & Sign-up

- Minimum and Maximum lengths should be set for all the text boxes
- Password should be displayed in masked format rather than showing actual text format
- Login credentials in upper case should be treated as invalid

- Validation / Confirmation message should be shown when invalid username and/or password is entered or the fields are left blank
- When logged in user copy the URL and paste in a new browser window, it should redirect to Login page
- After 3 or 5 unsuccessful attempts of login, user should get asked if they want to reset password
- After logout if user clicks on back button user should not be able to login within same session, it should redirect to login page
- **Add to Cart & Check out**
  - After adding items to the cart the user should have an option to continue shopping
  - If the user adds the same item to the cart while continuing to shop, the item count in the shopping cart should get incremented
  - All items and their totals should be displayed in the cart
  - Taxes should be applied for total
  - A user can add more items to the cart - total should reflect the change
  - Update the contents added to the cart - total should reflect the change
  - Remove items from the cart - confirmation message should appear
  - Proceed to checkout - redirect to payment page with correct total
  - Don't check out, close the site and come back later. The site should retain the items in the cart
- **Buy book**
  - Cart icon number should update according to the items in cart
  - Buy button should show up when the user click on the images
  - Message displayed should be correct
  - Required fields should display an alert if not filled out
  - Buy button on click should:
    - Increment number of items in cart
    - Show confirmation message
- **Rent out book & Sell book**
  - Text fields should receive only letters input and numeric fields should only receive number inputs
  - Required fields should display an alert if not filled out
  - Photos of the book should show up in images gallery
  - "Save & Continue" should save the data to the database correctly
  - The date format should be in correct format, and date "to" cannot be in the past or sooner than date "from"
- **Create Profile**
  - Class year is listed from the current year to 4 years later.
  - Major and Minor can be added and deleted.
  - Courses can be searched by the department and deleted.
  - The profile can be saved, or this step can be done later.
- **Log Out**
  - By clicking the logout button, the user can successfully log out from the account and view the app as a guest.
  - By clicking the return button, the user can go back to the page they are previously at.
- **Change Profile Information**
  - Class year is listed from the current year to 4 years later.
  - Major and Minor can be added and deleted.
  - Courses can be searched by the department and deleted.

- The profile can be saved, or cancel the modification.
- **In App Messaging**
  - Message can be received and sent successfully and securely.
- **View Current Orders**
  - The orders are showed up correctly, with the book information, price, the seller/buyer.
  - Click the cancel order will pend a cancel order form to admin, and will be approved by the admin.
  - Click the report a problem will lead to F-16.
- **View Past Orders**
  - The orders are showed up correctly, with the book information, price, the seller/buyer.
  - Click the contact will open the F-12, and start a conversation window with the buyer/seller.
  - Click the report a problem will lead to F-16.
- **View WishList**
  - The books are listed in the wishlist properly, with its title, ISBN number, availability, price for rent/buy. And click on the book can lead to the buy/rent page of the book.
  - The books can be deleted from the wishlist by clicking the remove button.
- **Troubleshooting & Reports Complaints**
  - Functions are listed properly in the select box.
  - The complaints will report to the admin, with user identifying traits, and the detail that the user type in.
- **Search with sorting and filter:**
  - Clicking on sort button or price/condition filters should display the results with all user selected price/conditions marked active.
  - Proper messages should be displayed in case no items are found with the user chosen filters.
- **Contact:**
  - Clicking on the send message without filling name, email, and message should keep prompting the user for valid inputs.
  - Clicking on the send message with name, email, and non-empty message should give an update that the message was sent successfully.
- **Recommendation Section:**
  - When logging in to the website with credentials of a user with no buying/renting history there should be no recommendation section on his/her home screen.
  - When logging in to the website with credentials of a user with some buying/renting history there should be a recommendation section on his/her home screen.
- **Push Notifications:**
  - When logging into the website with credentials of a user with valid username and password, the user should be able to see the push notification functionality on the profile page.
  - If a user has not phone number saved for his/her account the user should be prompted to enter a phone number.
  - If user has a phone number saved for his/her account and activates the push notification service, he/she should get a text message updating the user that the push notification service is activated for his/her account.

- If a user has successfully activated push notification service, he/she should receive text upon actions like changing/resetting password.

- **Manage Payment Method:**

- When logging into the website with credentials of a user with valid username and password, but no payment method stored the user should be directed to manage payment screen(Screen 3.22.1) if he/she decides to buy an item after adding to cart.
- A registered user with payment method stored should see a remove button that allows user to delete existing payment method as shown in the figure.
- Clicking on continue button without valid payment information should keep on prompting user of invalid status, with a message about what went wrong.

- **In-app Payment:**

- After the user has all valid payment method information, the user should be able to choose his/her stored payment method should be able to select his/her stored payment method for making payment as shown in figure 3.22.1.
- A user with no stored payment method should be prompted to enter payment information and should be prompted to enter valid payment method before confirming the purchase.
- The user should be able to see the total amount he/she is being charged as shown in Screen 3.22.1.

- **Reset Password/Change password:**

- A login attempt with incorrect password for more than 3 times should prompt user to reset their password.
- The user, in case of forgotten password, should be prompted to enter the email associated with his/her account, and should be emailed a link where the user can set a new password.
- The password change screen should only follow once the user clicks on the email sent to him/her.
- A registered user can only change the current password only if he/she enters current password correctly and enters a valid new password that is not the same as the last one.
- The user if he/she has activated push notification service should get a text that he/she has changed the password.

## 6. User Interface

- The home page will consist of a sidebar on the left, a main screen area where all available books are displayed. Refer to Screen 3.3.1.
- The pages to **rent a book**, **buy a book**, and **sell a book**, will be in forms format. Refer to Screen 3.3.1, 3.4.1, 3.7.1, 3.8.1.  
**Note:** Due to Sellacious integration, these pages will now be handled in Sellacious (**not in form format anymore**). Please refer to the User Manual for more information.
- There will be a ELO logo, which would be a clickable button that redirects the user to the home page. Refer to Screen 3.3.1.
- A first time user of the application should see the log-in page when he/she opens the application. Refer to Screen 3.1.1.
- Once the user is logged-in, he/she should be able to see the landing page with all available books.



- Every user will have his/her personalized profile page where they can view their information, add new information, or modify existing information. Refer to Screen 3.9.1.

## 7. Non functional requirements

### 7.1. Database

#### General Description

The database for the web application will be hosted on a web server. The web application uses phpMyAdmin to handle the administration of MySQL on the web.

#### ER Diagram

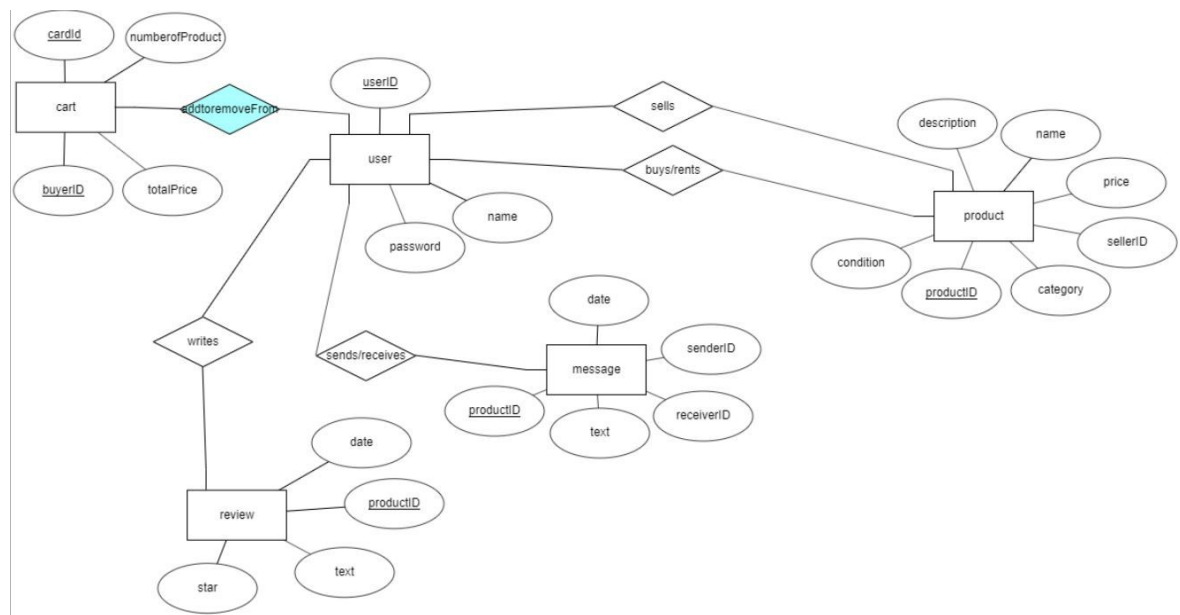


Figure 7.1.1: ER Diagram for database

### 7.2. Payment Integration

The payment functionality will not be implemented in this semester. However, for the client to be able to demonstrate ELO's potential functionalities for future developers and investors, a temporary payment method will be installed. The payment method is PayPal, which is already a built-in module for Sellacious. This method has the lowest risk and highest security standard, and PayPal is a PCI DSS Certified service provider. The details of the implementation is described in the diagram below:

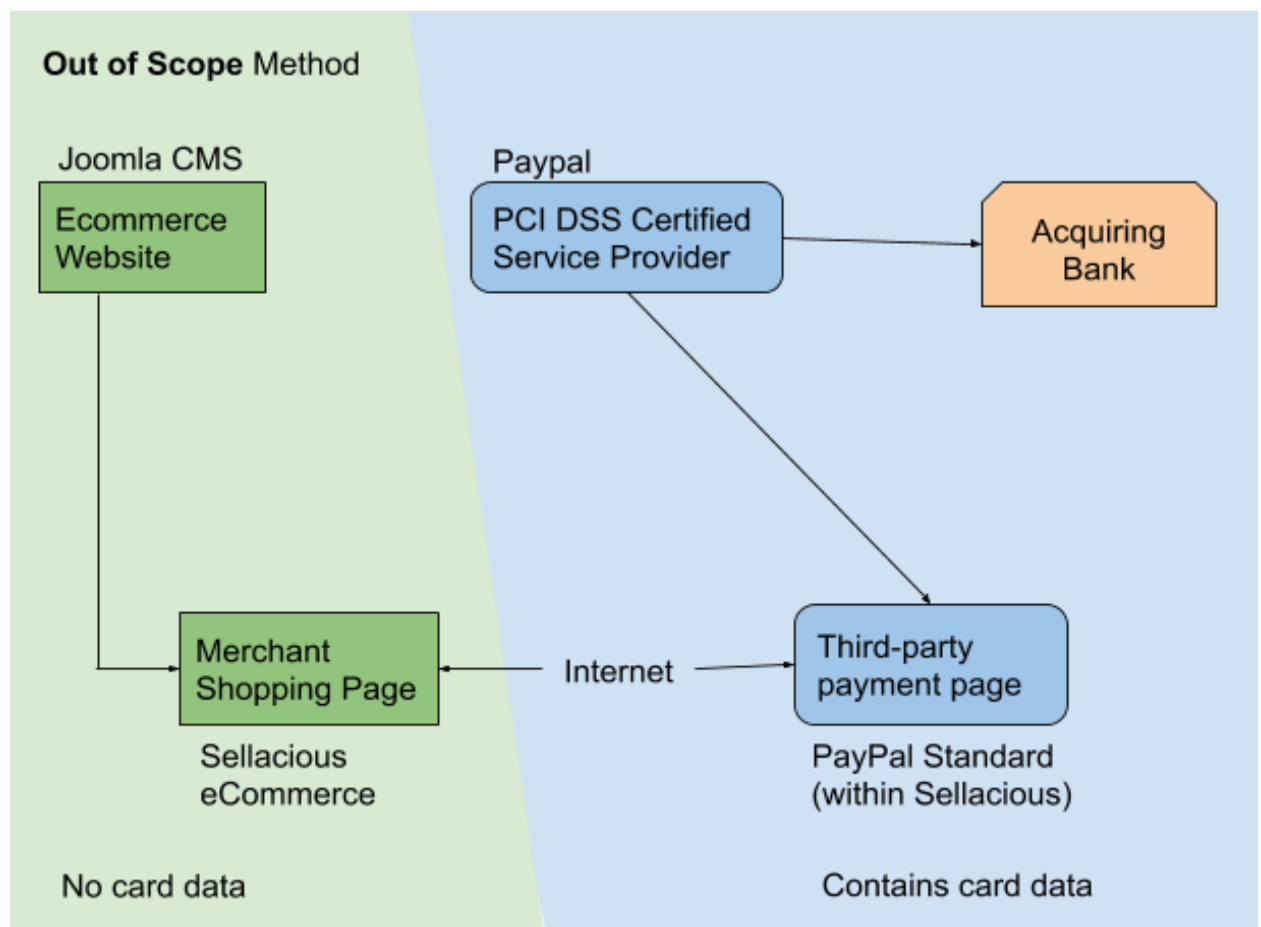


Diagram 7.2.1: Relationship between Joomla, Sellacious and payment provider.

### 7.3. Technical and Design Constraints

#### User Interface

The user interface design is limited by the allowed UI elements of Joomla templates' and the extent to which these elements can be customized. The color coding should adhere to the client's desired color palette, along with black and white.

#### Rent Functionality

Since *Renting Book* is not a customizable module in Sellacious, the functionality will have to be implemented from scratch. During the development process, the development team came to the conclusion that implementing a new Joomla module from scratch is going to be time-consuming and cannot be completed within this semester.

## 7.4. Timeline & Schedule

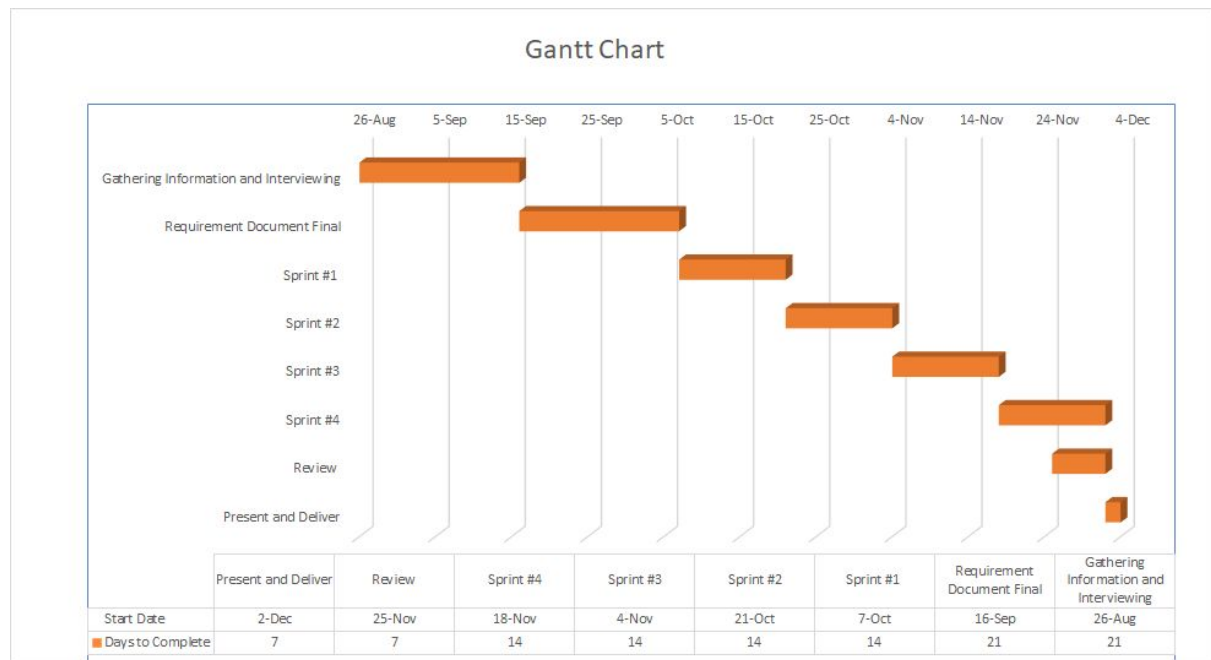


Figure 7.4.1 Chart for the planned timeline

The development approach would be to use agile development for our project. The following table summarizes the sprints with tentative schedule and tasks.

Sprint #	Time Period	Description
1	1 week	Setting up Database and Backend, Landing Page, Login, Signup, Account Verification, Contact Form, Reset Password, Create a Profile, Change Profile Information
2	2 weeks	Form for buying books , selling books, renting books.  Reporting a complaint (Joomla & Sellacious), View Orders in Sellacious, Add to Cart/Remove from Cart in Sellacious,  Wishlist in Sellacious
3	2 weeks	Search bar with sorting and filter option in Sellacious  Set up plugin for payment, Recommendation System, In App Messaging
4	2 weeks	Messaging Continued, Add/delete payment method, Push Notifications
5	1 week	Review and optimize
6	1 week	Present and deliver

Table 7.4.1 Table for explanation of Gantt Chart

## 7.5. Instruction & User Manual

See the separate document: **ELO User Manual**

## 8. Deliverables

- A requirement document that describes the application functional requirements, non-functional requirements, user interface design and database requirements.
- A functioning prototype of a web application that meets the requirements and fully captured the client's idea and vision.
- A detailed user manual at the end for the project for smooth transition and post-handling of the project.
- A poster of the application and project.
- A capstone presentation that gives a quick look on the project.

## 9. Glossary

Term	Description
Web Application	An application program that is stored on a remote server and delivered over the Internet through a browser interface
Actor	Specifies a role played by a user or any other system that interacts with the subject.
UI (User Interface)	The means by which the user and a computer system interact, in particular the use of input devices and software.
Browser Window	The means by which a user can use to access and view websites
ER Diagram	Data modeling diagram that illustrates an information system's entities and the relationships between those entities
Database	A structured set of data held in a computer, especially one that is accessible in various ways.
E-commerce Platform	A software technology solution that allows you to build storefronts (a storefront is any customer or business facing proposition relating to the selling of products and services) Or a software application that allows online businesses to manage their website, marketing, sales, and operations.
Prototype	An early sample, model, or release of a product built to test a concept or process, and generally used to evaluate a new design to enhance precision by system analysts and users.
Agile software development	Refers to a group of software development methodologies based on iterative development, where requirements and solutions evolve through collaboration between self-organizing cross-functional teams.
Sprint	A one time-boxed iteration of a continuous development cycle, where a planned amount of work has to be completed by the team and made ready for review.

Table 9.1 Table for glossary

## 10. Individual Responsibilities

**All members** of the development team will participate in the process of development, design, coding and testing. However, each member will have a concentration that he/she will be putting in extra emphasis on.

- Jivan will be concentrating on the database design, the system requirements and the general development process of the projects.
- Mai will be concentrating on the UI / UX design, testing aspects of the web application and documenting the development process.
- Tony will be concentrating on the implementation of functionalities and technical aspects of web application platform.

## 11. References

Arms, William. *CS 5150: Software Engineering*, Cornell University, 2019, [www.cs.cornell.edu/courses/cs5150/2019sp/lectures.html](http://www.cs.cornell.edu/courses/cs5150/2019sp/lectures.html).

“Flowchart Maker Org Chart Maker Drawing Program Floor Plan Creator.” *SmartDraw*, [www.smartdraw.com/](http://www.smartdraw.com/).

“Lexico” *Oxford*, <https://www.lexico.com/en>

Produl. “Wireframe Tools, Prototyping Tools, UI Mockups, UX Suite.” *MockFlow*, [mockflow.com/](http://mockflow.com/).