Project Vision, Requirements, and Design Document

for

Online Bookstore

Prepared by Robert Herber
Lawrence Ferretti
Nick Murphy
Naga Panganamamula

Section W01 Group 1
Software Engineering SWE 6623
Spring 2017

Kennesaw State University

28 February 2017

Table of Contents

Table of Contents 3
Revision History 4
1. Project Vision and Business Requirements 5
    1.1. Background and Business Opportunity 5
    1.2. Business Objectives and Success Criteria 5
    1.3. Project Risks 5
2. Vision and Scope of the Solution 6
    2.1. Vision Statement 6
    2.2. Major Features 7
    2.3. Assumptions and Dependencies 8
    2.4. Scope of Initial Release 8
    2.5. Scope of Subsequent Releases 8
    2.6. Limitations and Exclusions 8
3. Business Context 9
    3.1. Stakeholder Profiles 9
    3.2. Project Schedule 15
        3.2.1 Project Activity Tree 25
        3.2.2 Gantt Chart 29
4. Requirements 31
    4.1. User Stories 31
    4.2. Use Case Diagram 38
    4.3. Written Use Cases 38
5. Preliminary and Detailed Design 48
    5.1. Project Environment 48
        5.1.1 Operating Environment 48
        5.1.2 System Interactions and Information Flow 48
    5.2. UML Activity Diagrams 49
    5.3. UML Class Diagrams 57
    5.4. Software Modules and Components 58
    5.5. Wireframe Diagrams 59
6. Implementation and Unit Testing 69
    6.1. Development Tools and Environment 69
    6.2. Unit Testing 69
7. Integration and System Testing 69
    7.1 Project Integration 69
    7.2 Software Testing 69
        7.2.1 Test Plan & Schedule 69
        7.2.2 Test Items 70
        7.2.3 Item Pass/Fail Criteria 72
        7.2.4 Test Deliverables 73
8. Project Retrospective 113
    8.1 Project Results 113

8.1.1 Successes 113
8.1.2 Failures 113
8.1.3 Technical Challenges 113
8.2 Project Implementation Analysis 113
8.3 Lessons Learned(a.k.a What Can We Do Better Next Time?) 114
9. Appendices 110
9.1. Meeting Minutes 114
9.2 Other Supporting Documents (including source code) 124

Revision History

Name	Date	Reason For Changes	Version
Initialization	2/2/17	Initial version	1.0
Revision 1	4/4/17	Revised document for resubmission	1.1

1. Project Vision and Business Requirements (LF)

The online bookstore is a web-based software application that will provide online access for students to be able to purchase or rent textbooks for their courses and have those textbooks be delivered to their front door.

1.1. Background and Business Opportunity (LF)

With a brick-and-mortar bookstore, the sales are limited to specific times during the day and is only available to customers that are able to travel to it. In the modern age, the brick-and-mortar store is in a losing battle with the online stores such as Amazon.

The creation of an online bookstore for the school will increase the potential sales while at the same time limit overhead costs. The online store will be available to customers 24/7 and the warehouse will ship directly to their homes. Customers will no longer be limited by geographic regions and can purchase (or rent) the textbooks at their leisure. Having the online option for the bookstore will allow the school to directly compete with the digital markets.

1.2. Business Objectives and Success Criteria (LF)

By creating the online bookstore, it will increase the overall net profit of the school. The current market trends show that the online availability of the bookstore will generate 60% additional profit when compared to just having the brick-and-mortar bookstore. The overhead costs to create, maintain, and enhance the online bookstore will put the school at seeing a 40% increase in annual bookstore revenue.

The development costs for this venture are minimal and the estimated time to completion is May 1, 2017.

1.3. Project Risks (RH)

The risk assessment shown in the table below attempts to recognize the risks associated with the development of the online bookstore and the mitigation strategies that will be used to counter the risks.

The impact of the risks are described as being low (L), medium (M), or high (H). A low impact risk means that the risk is negligible and will not have a noticeable effect on the project's development. A medium impact risk means that the risk is marginal but manageable. A risk with a medium impact could cause a delay in the project's development. A high impact risk means that the risk is catastrophic and could cause major problems and setbacks in the project's development.

A risk's probability is shown as a percentage. For example, a risk probability score of 25% means that there is a one in four chance of the risk occurring.

Risk	Impact/Probability	Mitigation Plan
Lack of user involvement	M/25%	Conduct a minimum of biweekly stakeholder meetings.
Incorrect/incomplete requirements	H/30%	Validate requirements with the stakeholders.
Constantly changing requirements	M/40%	Validate requirements with the stakeholders. Develop prototypes to show the customer.
Unrealistic customer expectations	M/20%	Create a project vision and scope. Develop prototypes to show customer.
Unmet project deadlines	M/15%	Develop a project schedule. Reevaluate project schedule on a biweekly basis.
Developers adding extra functionality (“feature creep”)	L/10%	Prioritize requirements.
Conflicting development ideas between the developers	L/20%	Conduct a minimum of weekly developer meetings to ensure development understanding.

2. Vision and Scope of the Solution

2.1. Vision Statement (NM)

Vision Statement: Provide Kennesaw State University with an online textbook purchasing system which will increase the university bookstore's revenue and market access with minimal overhead, while giving customers a more convenient way to purchase textbooks. Currently, Kennesaw State is only able to sell textbooks via a physical bookstore on-campus. While its location and official sponsorship by the University affords it certain benefits, these are also its limitations. Students that are living off-campus, distance-learning students, and non-students have a harder time getting to the brick-and-mortar location, and are more likely to purchase their textbooks elsewhere. Even for on-campus students, online textbook services such as Amazon are often seen as a more convenient alternative to the official bookstore.

With an Online Bookstore system in place, the University is capable of recapturing some of the

business it has lost to other online shopping systems. It also allows for expansion into previously untapped populations such as non-students and long-distance-learning students. In addition, as the Online Bookstore system will use existing data from the University and will conform to its standards, this project should be able to add this potential revenue increase with minimal additional overhead to Kennesaw State.

Many customers are already accustomed to the convenience of online textbook shopping. The benefit to them will be the University's more specific knowledge of their needs, compared to other existing services. Students will be able to type in their professor's name or the course title and quickly see what textbooks are required for them to purchase – this is a feature that Amazon or eBay can not provide. Additionally, students with sufficient Financial Aid awards will be able to use those funds to purchase textbooks through the Online Bookstore system

2.2. Major Features (NM)

1. Search for textbooks
Users will be able to search for textbooks by title, author, assigned professor(S), assigned courses, or by keyword(s) in the product description. The user will also be able to sort results alphabetically if he/she desires.

2. View textbook information
Users will be able to click on a book from the search results page and be taken to a details page to view more detailed information on that item. The information that will be displayed on the results page is the name of the book, the author, ISBN, a description of the item, the cover image, the price (there will be different prices depending on the format of the book; all of which will be shown in the details page), the format of the book (including new, used, rent, or ebook), the associated courses and sections with the book (with whether the book is recommended or required for that course), and the associated professor(s) with the book.

3. Purchase textbooks
Users will be able to add and remove any quantity of textbooks to their shopping carts (if there are enough of the items in stock). Users will then be able to checkout and pay for their order using one of three methods including credit card (Visa and MasterCard), PayPal, and by financial aid (which will require the user to be a student, be logged in, and have enough financial aid to cover the entire order). At checkout, the system will apply a 7% GA sales tax and a $14.99 shipping fee if shipping is involved.

4. Receive order confirmation
Users will receive order confirmation upon placing a successful order. A receipt will be displayed on the user’s screen, sent to the user’s email, and will be saved to a shared folder on the server. The receipt will show the order summary (items purchased, price of each, tax, shipping fee, and total cost), an ebook download link (if applicable), and the rental due date (if applicable).

5. Send updates to concerned parties
The online bookstore system will also update relevant files after purchases. The inventory file will be updated appropriately after an order has been placed. If the user is a student and used financial aid to pay for their order, then the user’s financial aid balance will be updated as well. A copy of all order receipts will be saved on a shared folder on the server that is accessible by the

administrative assistant.

2.3. Assumptions and Dependencies (NM)

As this will be an extension of the Kennesaw State University's existing bookstore, the University has played a crucial role in the development and release of the Online Bookstore. The bookstore manager has served as the main point of contact, providing the initial requirements and additional feedback while also passing on additional information and concerns from other members of the University.

The Online Bookstore application will depend heavily on CSV files provided by the Registrar of the University. These data files contain all the information needed to run the Online Bookstore system, such as textbook names, associated courses and professors, the quantity available, and the prices. In order to accept payments, simulated versions of PayPal, the University's Financial Aid system, and a credit card processing portal will be utilized as well.

2.4. Scope of Initial Release (NM)

The initial release will be the entire Online Bookstore system as is described and designed throughout this document and containing the major features listed in Section 2.2. This is the minimum viable product needed for the client to fully evaluate the system and provide their final feedback on the development team's work.

2.5. Scope of Subsequent Releases (NM)

Subsequent releases will consist of fixes for any software errors that may have been missed in the first release's testing phase. If there were any misunderstandings regarding how the system should function in the first version, those misconceptions would also be corrected in the subsequent releases.

Additional online bookstore features will also be coming with subsequent releases. These features include adding miscellaneous items to the bookstore including tee-shirts, coffee mugs, general school supplies, and backpacks. Other future features including various quality of life features that are focused on the staff. One such feature is automatically printing order receipts for the warehouse manager.

2.6. Limitations and Exclusions (NM)

This Online Bookstore system is intended to be a class project, not a real world working system. Therefore, many features will be simulated rather than truly functional. The most notable mock-up feature is payment processing: while proof-of-concept pages will be created for credit cards, PayPal, and the University's Financial Aid system, none of them will actually connect to those real world services.

As there will be no administrator panel, there will be no method for uploading a new CSV to update the quantity and selection of textbooks in the Online Bookstore. In addition, while a PDF copy of each order’s receipt will be generated and saved on the server, it will not be placed on a

network shared folder for the fictional warehouse manager and administrative assistant. Finally, even though a user will be able to complete a simulated order on the Online Bookstore, no textbooks will actually be sent to the Shipping Address they provided – nor will they receive any digital textbooks.

3. Business Context

3.1. Stakeholder Profiles (RH and LF)

3.1.1 John the Bookstore Manager

Roles: John is the manager of the bookstore and he is the customer and an indirect user of the online bookstore.

Responsibilities: John is responsible for the bookstore and all of the business done by the bookstore. He is also responsible for paying for the development of the online bookstore.

Attitudes: John is likely to be positive and open about the Online Bookstore development because he is the one ordering it. However, if features need to be cut, if the project is delayed, or if development goes over budget, he may become frustrated.

Major Interests: John is interesting in attracting more customers to the bookstore by allowing them to purchase books online instead of having to purchase books from the physical bookstore. He believes that expanding the bookstore to an online version will accomplish two major goals. The first goal is that the online bookstore will make life more convenient for all students (especially online students and students who live far away from campus) by allowing them to purchase their books without having to travel the on-campus bookstore and by having their books shipped directly to them. The second goal the online bookstore will accomplish is increasing the overall revenue to the bookstore. John believes that the convenience offered by the online bookstore will encourage more students to buy their books from the online bookstore rather than from other online retailers.

Concerns: John is concerned about the online bookstore being developed before the beginning of next semester. He wants the online bookstore to be operational before next semester so the students can begin to purchase their books online. John is also concerned about the online bookstore being able to handle payments correctly. If the the online bookstore is unable to handle payments correctly, than the online bookstore will lose business.

Technical Proficiency/Constraints: John is familiar with using existing online shopping applications and expects the online bookstore to offer a similar experience to users. However, John is not familiar with the technical specifications of the online bookstore and instead defers to Dilbert's (from UITS) technical expertise.

Stakeholder Requests:

1. John wants an online bookstore built so that users can buy their books without physically going to the bookstore.
2. John wants “to make life easier” for students when they are buying their books.
3. John wants students to be able to search for books by title, author, keyword, ISBN, course section associated with the book, and by the professor teaching the course section.
4. John wants the search results to show the book's title, author, price, and a picture of the book.
    4.1. He wants the details page to show the book’s title, author, ISBN, description, format, price, course sections associated with the book, and the professors who teach the course sections associated with the book, and if the book is required or recommended for a particular course section.
5. John wants the online bookstore to show the four different formats of each book.
    5.1. Not every book is available in all four formats.
    5.2. The different formats all have different prices and quantities.
    5.3. EBooks are available for download immediately
        5.3.1. A link to the download will be available on the receipt that is shown directly after purchase and on the receipt emailed to the purchaser.
6. John wants users to be able to add and/or remove any amount of any format of books to and from their cart.
    6.1. A user should not be able to add a book that is out of stock to their cart.
    6.2. A user should not be able to add more than the remaining quantity of books to his/her cart.
    6.3. The shopping cart should show subtotal, name of items, picture, format, and quantity.
7. John wants users to be able to checkout and pay for their books.
    7.1. He wants payment in the form of Paypal, Visa, Mastercard, or Financial Aid
        7.1.1. Financial aid can only be used if the student has enough to cover the cost of the entire purchase.
        7.1.2. If using financial aid, the student must log into the system using their KSU email.
    7.2. John wants the online bookstore to allow user to pay for their order using only one payment method per checkout session.
8. John wants a user to be able to checkout as a guest without having to log on.
9. John wants anyone who wants to buy books to be able to use the Online Bookstore, not just students.
    9.1. The online bookstore should be accessible on all devices.
10. John wants the online bookstore to only sell books when the on campus bookstore is selling books.
    10.1. Both stores will stop selling books on the 15th day of class.
11. John wants the online bookstore to have a Help/FAQ page.
12. John wants a GA sales tax of 7% be added to the total cost of an order regardless of where the user lives and where the items are being shipped to.

13. John wants the online bookstore complete by the last day of class.

3.1.2 Jenny the Administrative Assistant

Roles: Jenny is the bookstore's administrative assistant and an indirect user of the Online Bookstore.

Responsibilities: Jenny prints receipts from the shared folder on her desktop to give to the warehouse manager for shipping.

Attitudes: Jenny is likely to be accepting of the new Online Bookstore as it includes the feature of saving all of the Online Bookstore orders to the shared folder on her desktop. Although the introduction of the Online Bookstore does increase her workload (because before the Online Bookstore, there were no online orders), the workload is small because it only involves printing the order receipts and giving them to the warehouse manager. Jenny may become frustrated if the order receipts are not properly saved to the shared folder on her desktop.

Major Interests: Jenny is interesting in having her shared folder on her desktop contain all of the orders from the Online Bookstore. This is important to Jenny because it allows her to keep track of all Online Bookstore orders in a single location, thus making it easier for her to print them and give them to the warehouse manager for shipping.

Concerns: Jenny is concerned with the physical size of the printed receipt. She would like a printed receipt to fit on one page in order to save paper.

Technical Proficiency/Constraints: Jenny is familiar with her desktop computer and printer. She understands how to open the shared folder on her desktop containing the order receipts and she understands how to print them.

Stakeholder Requests:

1. Jenny wants the Online Bookstore to generate a receipt of each order which includes the order's customer information (name, email, phone number, billing address (street address, PO box (if applicable) city, state, postal code), the order details (products ordered and quantity of each product ordered), and shipping address (street address, PO box (if applicable), city, state, postal code)).
    1.1. She wants this receipt to be saved into a shared folder that is accessible on her desktop.
    1.2. She wants to be able to print the receipt in order to give it to Joe.

3.1.3 Joe the Warehouse Manager

Roles: Joe is the warehouse manager and an indirect user of the Online Bookstore.

Responsibilities: Joe is responsible for managing the warehouse where all of the Online Bookstore's inventory will be held. He is responsible for picking up the printed orders from the administrative assistant, packing the ordered items, and shipping the ordered items.

Attitudes: Joe is likely to be open to the introduction of the Online Bookstore. Although it does increase his workload as he will have to package and ship more items, it also increases his job security.

Major Interests: Joe is interested in the success of the Online Bookstore because it will increase business to his warehouse. The most important feature of the Online Bookstore to Joe is the shipping address information as it must be a valid USPS shipping address.

Concerns: Joe uses USPS priority mail for shipping, therefore, Joe is concerned with the possibility of an order not containing a valid USPS shipping address.

Technical Proficiency/Constraints: Joe's work does not involve any technical skills, therefore he has no technical constraints.

Stakeholder Requests:

1. Joe wants to be able to physically (meaning in person) pick up (as in acquire, but the process of acquiring the invoices does actually involve literally "picking up" the invoices) the previous day's order invoices from Jenny.
2. Joe wants the shipping address on each order receipt to be a valid shipping address that USPS priority mail can ship to.
3. Joe wants the USPS priority mail shipping fee of $14.99 to be paid for by the user.
    3.1. No fee for e-books

3.1.4 Dilbert from UITS

Roles: Dilbert is the technical advisor and an Indirect user of the Online Bookstore.

Responsibilities: Dilbert is from UITS and is the technical advisor for the project. He controls the server where the Online Bookstore will be located and has the information regarding the domain of the Online Bookstore.

Attitudes: Dilbert is likely to be open and accepting of the new Online Bookstore. He may become upset if the Online Bookstore severely impairs the performance of his server.

Major Interests: It is important to Dilbert that the Online Bookstore conforms to the KSU branding and style guidelines.

Concerns: Dilbert's concerns include the Online Bookstore not running properly on his server and the Online Bookstore's conformance to the KSU branding the style guidelines.

Technical Proficiency/Constraints: Dilbert is trained and educated in a technical field and is

the technical advisor to the project. He knows all of the information regarding the server and the domain of the Online Bookstore. He is also able to provide advice for the implementation of the Online Bookstore.

Stakeholder Requests:

1. Dilbert wants the Online Bookstore to conform to the KSU branding and style guidelines.

3.1.5 Ian B. Greedy the Bursar

Roles: Ian B. Greedy is the bursar and an indirect user of the system.

Responsibilities: Ian is responsible for all of the financial aid information on the students.

Attitudes: Ian is likely to be welcoming of the Online Bookstore as it will increase the flow of money into the school and Ian is known to be greedy. Ian could potentially become upset if the Online Bookstore does not process financial aid payments correctly.

Major Interests: Ian is most interested in the Online Bookstore's ability to accept payment in the form of financial aid.

Concerns: Ian is concerned with the Online Bookstore's ability to properly access and update a student's financial aid information. The a student uses financial aid to pay for an order, that must be reflected in the student's financial aid balance.

Technical Proficiency/Constraints: Ian is provides all of the student's financial aid information in the form of a csv data file. The Online Bookstore must be able to read and update a student's balance from Ian's provided data file.

Stakeholder Requests:

1. Ian B. Greedy wants students to only be able to use Financial Aid to pay for their textbook order if their Financial Aid Award is sufficient to cover the entire purchase.
2. Wants students' financial aid information available in online bookstore system.
3. Ian B. Greedy wants the student's financial aid award to be decreased by the total cost of their order once they purchase their textbooks.

3.1.6 Bill Bigsby the Registrar:

Roles: Bill Bigsby is the registrar and an indirect user of the Online Bookstore.

Responsibilities: Bill is responsible for providing the course information and the professor teaching the courses to the Online Bookstore.

Attitudes: Bill is likely to be accepting of the new Online Bookstore as he is a kind person and welcomes any new additions that make the student's lives easier.

Major Interests: Bill is interesting the Online Bookstore's ability to allow users to search by course name and/or by professor name.

Concerns: Bill is concerned with the Online Bookstore's ability to properly associate a book with the professors and courses it is associated with.

Technical Proficiency/Constraints: Bill provides all of the book, professor, and course associations in the form of a csv data file. Therefore, the Online Bookstore must be able to read information from the csv data file he provides.

Stakeholder Requests:

1. Bill wants his text file to be used for all course, section, professor, and textbook associations.
    1.1. The text file will not change throughout the semester.

3.1.7 The User

Roles: The user is a direct user of the Online Bookstore.

Responsibilities: The user is responsible for actually using the Online Bookstore.

Attitudes: The user is likely to be welcoming to the Online Bookstore as it gives the user another convenient way to buy his/her books.

Major Interests: The user is most interested in being able to buy his/her books online without going to the physical bookstore and having the books shipped to his/her shipping address. The user is interesting in being able to search for books in a variety of ways, add and/or remove books from his/her cart, and pay for the order.

Concerns: The user is concerned about the Online Bookstore properly handling his/her payment. Because the user is inputting payment information that connects the user's financial institution to the Online Bookstore, the user wants to make sure that the Online Bookstore does not overcharge him/her.

Technical Proficiency/Constraints: The user's technical proficiency varies from user to user. One user could be very technically proficient while another user could be not technically proficient at all. Therefore, the Online Bookstore must not require much technical proficiency to use in case a user is not technically proficient.

Stakeholder Requests:

1. The user wants to be able to search for books by title, author, ISBN, keyword, course section associated with book, and professor.
2. The user wants to be able to sort his/her search results alphabetically by textbook name, assigning professor, and author, as well as by price.

3. The user wants to be able to add any amount of books in any format (new, used, rental, or EBook) to my cart.
4. The user wants to be able to click on a book from his/her search results and see its details including title, author, description, ISBN, associated courses (including professor's name, section, and whether the book is required or recommended (or both)), picture of the book cover, price, format (new, used, rental, or ebook), and number of books in stock
5. The user wants to be able to rent a book if the book is available for rent.
    5.1. The due date should be displayed on the receipt.
6. The user wants to be able to buy a used copy of a book if available.
7. The user wants to be able to buy an EBook version of a book if available.
    7.1. The user wants to be able to download his/her EBook immediately after purchase.
        7.1.1. The link will be shown on the receipt both in the online bookstore receipt shown after purchase and on the receipt sent to the user's email address.
8. The user wants to be able to add or remove any quantity of books in any format from his/her cart.
9. The user wants to able to checkout and pay for his/her books.
    9.1. Be able to use PayPal to pay for his/her books
    9.2. Be able to use Visa to pay for his/her books
    9.3. Be able to use MasterCard to pay for his/her books
    9.4. Be able to use my financial aid to pay for his/her books.
10. The user wants to be able to see his/her financial aid balance before they purchase their textbooks, as well as during the checkout process.
11. The user wants to be able to checkout as a guest without having to log on if he/she wants.
12. The user wants the books he/she ordered shipped to his/her address (even if he/she use a PO box).
13. The user want his/her books shipped to his/her vacation house in Guam.
14. The user wants a receipt emailed to his/her email address after his/her order has been placed.
    14.1. User also wants the receipt displayed on the screen directly after checkout.
15. The user wants the system to remember his/her shopping cart selections even if he/she closes his/her browser.
16. The user wants to be able to have separate billing and shipping addresses if he/she wants.

3.2. Project Schedule (RH and NM)

The development of the online bookstore will follow a waterfall development process. This means that the development will be broken down into distinct and separate phases with the

previous phase being a prerequisite for the following phase. Because of this, the following work breakdown structure separate each phase of the waterfall cycle into its own distinct phase with each phase complete with its own steps, activities, and milestones. Because there are three developers, up to three activities can be worked on simultaneously (assuming all prerequisites are met). Please see the activity graphs and Gantt chart to view prerequisites and which steps/activities can be worked on simultaneously.

The cost shown in the work breakdown structure below is measured in person business days. One person day is defined as 2 hours of work (this measures out to 10 person hours of work per week per developer assuming each developer works 2 hours a day for 5 days a week). The online bookstore project was introduced to the developers on 2/1/17 and is due on 5/1/17. This gives (64 business days between 2/1/17 and 5/1/17 times 3 group members) 192 total person days (which equates to 384 total person hours) spread over 64 real time business days.

Work Breakdown Structure and Milestones

Phase 1 Requirements Documentation Cost: 29 - 34			
Step 1.1 Elicit Requirements Cost: 10			
Activity 1.1.1 Interview stakeholders. Cost: 5			
Activity 1.1.2 Create Project Vision Statement, project scope, business objectives, stakeholder profiles, and risk assessment. Cost: 5			
	Step 1.2 Write Requirements Documentation Cost: 18		
Activity 1.2.1 Write User Stories. Cost: 5			
Activity 1.2.2 Create Use Case Diagram. Cost: 3			
Activity 1.2.3 Write written use cases. Cost: 10			
		Step 1.3 Create Project Schedule Cost: 13	
Activity 1.3.1 Create Work Breakdown Structure. Cost: 7			
Activity 1.3.2 Create Project Activity Graph. Cost: 3			
Activity 1.3.3 Create Gantt Chart. Cost: 3			
			Step 1.4 Validate

	Requirements **Cost: 1-6**
Activity 1.4.1 Hold customer meeting to validate requirements. Cost: 1	
Activity 1.4.2 Revise Requirements documentation based on customer feedback. Cost: 0-5	

Milestones for Phase 1	
M1.1	Interviews with stakeholders complete
M1.2	Project vision statement, scope, risk assessment, stakeholder profiles, and business objective are written
M1.3	User stories are written and finalized
M1.4	Use case diagram is drawn and complete
M1.5	Written use cases are complete
M1.6	Work breakdown structure complete
M1.7	Project activity graph drawn and complete
M1.8	Gantt Chart drawn and complete
M1.9	Requirements are validated to be complete with the customer
M1.10	Requirements are revised

Description of Phase 1:

Phase 1 is the requirements step in the waterfall process model. As described above, this phase will consist of 4 steps. Step 1 is to elicit the requirements of the proposed system from the customer. This step includes the activities of interviewing the various stakeholders and documenting their requests and information on the proposed project. The next activity in this step is to create the formal documentation that captures the main ideas, goals, and risks of the project. This includes writing the vision statement with the customers, defining the scope of the project, creating a risk assessment for the project, writing out the stakeholder profiles for the major stakeholders involved with the project, and defining the business objectives of the project.

The next step in phase 1 is writing the formal requirements documentation. This step starts with writing out user stories for the project. User stories are short stories that explain what a stakeholder would like the system to accomplish. These stories are written from the stakeholder requests (found in the stakeholder profiles) and (if the stakeholders themselves do not write the user stories) are validated with the stakeholders. The next step in creating the requirements documentation is drawing out a use

case diagram. This is a high level visual depiction of the system with the associated actors and use cases. After the use case diagram is drawn, the activity of writing the written use cases begins. Written use cases are detailed descriptions of all of the use cases depicted in the use case diagram.

Creating the project schedule is the next step in phase 1. The first activity in creating a project schedule is to create what is known as a work breakdown structure. A work breakdown structure (abbreviated as WBS) is a written structure that breaks the work of a project down into distinct phases, steps, and activity. Each phase has steps and each step has activities. Each activity, step, and phase has a cost (this cost can be in units of time, unit, money, or whatever the project manager wants) associated with it. WBSs also have milestones associated with them. A milestone is an important point in time where certain criteria are met. The above chart is an example of a WBS. The next activity in creating a project schedule is to make an activity graph of the work breakdown structure. An activity graph is a visual depiction of the work breakdown structure complete with nodes that represent milestones, lines that represent activities and steps, and the costs associated with reaching each node. Activity graphs can also show which milestones are dependent on the completion of other milestones (these are known as prerequisites) and they can also show which activities can be worked on simultaneously. The last activity is creating a Gantt chart. This is another graphical representation of the project's timeline. A Gantt chart is similar to an activity graph in that it shows activities and milestones throughout time, but it also shows who is working at any given time which can allow for a visual representation of human resource constraints.

The last step in phase 1 is validating the requirements with the stakeholders. After all the requirements documentation is created and written, it is important to go over this documentation with the stakeholders to ensure that it is indeed what they are wanting from the project. Without validation from the stakeholders, the project runs a higher risk of incomplete and/or incorrect requirements which could end up being very costly (in both time and money) if the project is developed from the wrong requirements.

<table>
<tr><th colspan="3">Phase 2 Design the Project
Cost: 24</th></tr>
<tr><td>Step 2.1 Describe Operating Environment
Cost: 3</td><td colspan="2"></td></tr>
<tr><td colspan="3">Activity 2.1.1 Define the system's availability, reliability, performance, and integrity requirements.
Cost: 1</td></tr>
<tr><td colspan="3">Activity 2.1.2 Create system interactions and information flow diagrams.
Cost: 2</td></tr>
<tr><td></td><td>Step 2.2 Software Modules and Components
Cost: 6</td><td></td></tr>
<tr><td colspan="3">Activity 2.2.1 Breakdown Project into software modules
Cost: 2</td></tr>
<tr><td colspan="3">Activity 2.2.2 Define program software needs</td></tr>
</table>

Cost: 2		
Activity 2.2.3 Create module decomposition diagram Cost: 2		
	Step 2.3 Design Database and Schema (Optional) Cost: 6	
Activity 2.3.1 Research viability of using a database and, if chosen to use one, choose a Database Cost: 1		
Activity 2.3.2 Create database ER diagram Cost: 3		
Activity 2.3.3 Create Database Schema Cost: 2		
		Step 2.4 Design Project Diagrams Cost: 9
Activity 2.4.1 Create UML Activity Diagrams Cost: 3		
Activity 2.4.2 Create UML Class Diagrams Cost: 3		
Activity 2.4.3 Create Wireframe Diagrams Cost: 3		

Milestones for Phase 2	
M2.1	The operating environment has been defined.
M2.2	Project has been broken down into modules and a module decomposition diagram has been created.
M2.3	Database has been chosen and ER diagram and schema created. (If applicable)
M2.4	UML Activity diagram created and validated with the stakeholders.
M2.5	UML Class diagram created and validated with the stakeholders..
M2.6	Wireframe diagram created and validated with the stakeholders..
M2.7	Project modeling complete and validated with the stakeholders..

Description of Phase 2:

Phase 2 is designing the project. This phase directly correlates to the design step in the waterfall model. This phase takes the requirements from phase 1 and designs the project based on the documentation

produced in phase 1. This phase has four steps which each step having several activities. Step one in designing the project is to describe the operating environment. This step's first activity is defining the system's availability, reliability, performance, and integrity requirements. The online bookstore will have environmental constraints expected of it (such as how many users it can accommodate at one time, the system's up-time, responsiveness, etc.) and those constraints and expectations will be defined in this activity. The next activity in this step is to create system interactions and information flow diagrams.

The next step in phase 2 is decompose the project into distinct software modules and components. This step includes the activities of breaking down the project into smaller pieces, defining the software needs of those smaller pieces, and creating a module decomposition diagram of those pieces to use as a visual aid. Breaking down the project into smaller modules helps with work assignments and incremental implementation of the project.

Database design is an optional step in the design of the project. The online bookstore can be developed using many different methods and using a database to store data is one method. In this step, the development team will research the viability of using a database to implement the online bookstore. If using a database is deemed unnecessary, this step will be skipped. However, if the development team decides that a database will be used to implement the online bookstore, this step will include the activities of choosing which database to use (as each database system has its own strengths and weaknesses), creating an ER diagram of the database tables, and then creating a database schema from the ER diagram.

The last step in phase 2 is designing project diagrams. This step includes the activity of creating a UML activity diagram. This diagram is a high level diagram the shows the flow of events from one activity to another; similar to a flow chart. The next activity in this step is to create a UML class diagram. A class diagram is a static representation of the system that shows entities (complete with attributes and methods associated with the entity) and their relationships to one another. The last activity in step 4 is creating a wireframe diagram. A wireframe diagram is a visual representation of how the user interface will look. This is similar to a prototype and is useful because a wireframe diagram can be shown to the stakeholders so they can see how, in the case of the online bookstore, the web page will be laid out.

<table>
<tr><th colspan="3">Phase 3 Code
Cost: 25</th></tr>
<tr><td>Step 3.1 Implement Database (Optional)
Cost: 4</td><td colspan="2"></td></tr>
<tr><td colspan="3">Activity 3.1.1 Install Database into server
Cost: 1</td></tr>
<tr><td colspan="3">Activity 3.1.2 Create tables and relationships in the database
Cost: 2</td></tr>
<tr><td colspan="3">Activity 3.1.3 Import Data into the database
Cost: 1</td></tr>
<tr><td></td><td>Step 3.2 Implement functions
Cost: 15</td><td></td></tr>
</table>

Activity 3.2.1 Design searching functions Cost: 1	
Activity 3.2.2 Implement searching functions Cost: 2	
Activity 3.2.3 Design login system and cart functions Cost: 2	
Activity 3.2.4 Implement login system and cart functions (including adding and removing items and reviewing cart) Cost: 4	
Activity 3.2.5 Design checkout, payment, and receipt functions Cost: 2	
Activity 3.2.6 Implement checkout, payment, and receipt functions Cost: 4	
Activity 3.2.7 Conduct unit testing Cost: 3	
	Step 3.3 Implement HTML and CSS **Cost: 6**
Activity 3.3.1 Implement HTML Cost: 3	
Activity 3.3.2 Implement CSS Cost: 3	

Milestones for Phase 3	
M3.1	Database is installed on the server
M3.2	Tables and relationships are created in the database
M3.3	Data is imported into the database.
M3.4	Search functions are implemented
M3.5	Login system and Cart functions are implemented
M3.6	Checkout, payment, and receipt functions are implemented
M3.7	HTML is implemented
M3.8	CSS is implemented
M3.9	Project code implementation and unit testing complete and validated with the stakeholders.

Description of Phase 3:

Phase 3 is the actual coding implementation of the Online Bookstore project and is correlated to the code step in the waterfall process. This phase is broken down into three steps (one of which is depending on if a database was chosen to be used in phase 2). Each step has its own activities. The first step is dependent on if a database is going to be used. If so, this step involves the activities of installing the database software onto the server where the online bookstore will be hosted, creating the tables and relationships (based off of the scheme that was created in phase 2) in the database, and then importing the data from the .csv files into the database. If a database is not being used in the implementation, this step is skipped.

The next step in coding is implementing the functions of the Online bookstore. This is where all of the functionality of the system will be coded. The first activity of this step is to design the searching functions. There are three major searching functions: search by keyword (which will be used to search by title, author, ISBN, description keyword), search by professor, and search by course. The next activity is implementing the previously designed search functions. The third activity of this step is designing login system and the cart functions. These functions include adding items to the cart, removing items from the cart, and viewing the cart. The login system includes logging in, authenticating user information, and logging out. The next activity is implementing the designed login system and cart functions. Designing the checkout, payment, and receipt functions is the next activity. These functions include the viewing cart functions, entering shipping and billing information, payment functions (which include authenticating payments through paypal if using paypal, the user's bank if using Visa or MasterCard, or the registrar if using financial aid), and receipt functions (which include generating the receipt, displaying the receipt on the screen for the user, email the receipt to the user, and saving the receipt in the shared folder for the administrative assistant). The next activity is implementing the designed checkout, payment, and receipt functions.

The last step in phase 3 is implementing the HTML and CSS for the web page. The step will take the design of the layout of the webpage from the wireframe diagrams and implement them using HTML and CSS. The step consists of two activities. The first activity is implement HTML for the webpage and the second activity is implement CSS for the webpage.

Phase 4 Test **Cost: 15**		
Step 4.1 Function Testing **Cost: 7**		
Activity 4.1.1 Design function tests Cost: 2		
Activity 4.1.2 Implement function tests Cost: 2		
Activity 4.1.3 Debug and verify functions with function tests. Cost: 3		
	Step 4.2 System Testing **Cost: 7**	
Activity 4.2.1 Design system tests Cost: 2		
Activity 4.2.2 Implement system tests		

Cost: 2	
Activity 4.2.3 Debug and verify system functionality with system tests Cost: 3	
	Step 4.3 Acceptance Testing **Cost: 1**
Activity 4.3.1 Hold acceptance testing presentation with stakeholders Cost: 0.5	
Activity 4.3.2 Validate system with stakeholders Cost: 0.5	

Milestones for Phase 4	
M4.1	All function tests are designed and coded
M4.2	Functions are verified with function tests.
M4.3	All system tests are designed and coded
M4.4	System functionality is verified with system tests.
M4.5	System is validated through acceptance testing.

Description of Phase 4:

Phase 4 is testing the Online Bookstore. This phase is correlated with the testing step in the waterfall process and is the last phase of the project. The phase consists of three steps. The first step is function testing. Designing the function tests is the first activity in this step. This activity involves designing detailed and effective tests for each function implemented in phase 3. The next activity in this step is implemented the tests designed in the previous activity. After the tests have been implemented, verification and debugging of the functions based on the test results begin.

The next step in phase 4 is system testing. This step tests the system as a whole and ensures that all of the functions work together correctly when required. The step includes the activities of designing system tests, implementing the tests, and verifying system functionality and debugging based on the system tests.

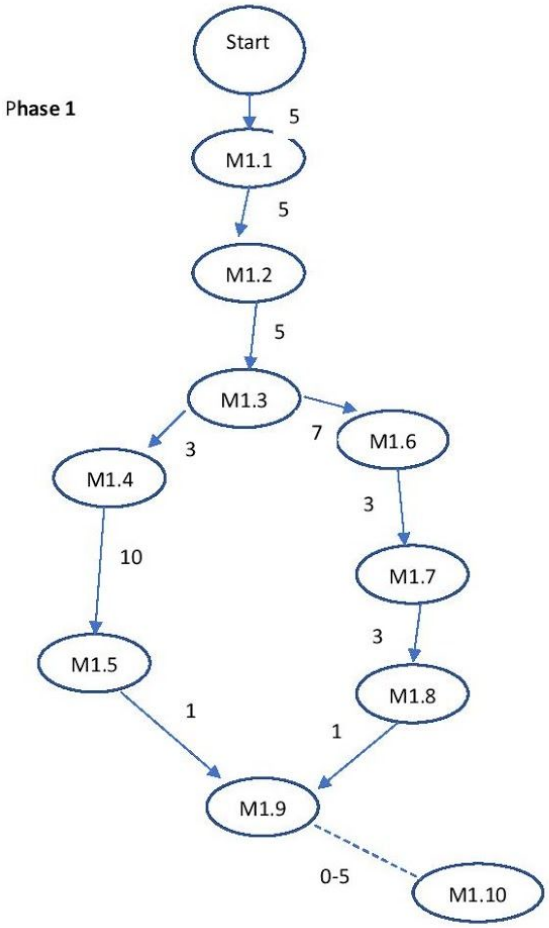
The last step in phase 4 is acceptance testing. After the Online Bookstore has been thoroughly tested on both a function level and a system level and has been deemed worthy of completion, acceptance testing begins. This step involves holding a meeting with the stakeholders and presenting the system to them. This phase is complete once the system has been validated by the stakeholders after successful acceptance testing.

3.2.1 Project Activity Tree (RH)

Below are the activity graphs for the above work breakdown structure. Each activity graph is associated with the phase above (the activity graph labeled “phase 1” below is the activity graph for the phase 1 WBS above). Each node of the graphs represent a milestone from the milestone charts above (M1.1 on the activity graphs is associated with M1.1 in the milestone charts above).

Each node also has the cost (in person business days as explained above) associated with reaching that milestone (the cost is the number located next to the arrow that points to a node). The arrows in between the milestone nodes are representative of the activities required to achieve the milestone and also indicate prerequisites (such as the arrow between milestone M1.1 and M1.2 shows that M1.1 is a prerequisite for M1.2).

Each activity graph also depicts the critical path and the slack time associated with the branches that are not the critical path. The critical path is the longest path (in cost) where there is no flexibility in scheduling. If there is a delay in the critical path, then the entire project becomes delayed. Each path that is not the critical path has slack time. Slack time is the amount of time that the particular path can be delayed without delaying the rest of the project (in other words, if a path has a slack time of 3 person days, then that path can be behind by up to 3 person days without affecting the schedule of the rest of the project).



Critical Path

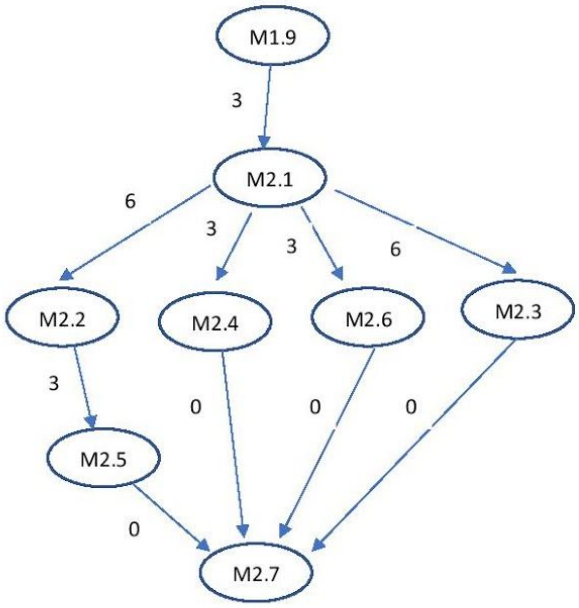
Start -> M1.1 -> M1.2 -> M1.3 -> M1.4 -> M1.5 -> M1.9 = 29 Person days

Start -> M1.1 -> M1.2 -> M1.3 -> M1.6 -> M1.7 -> M1.8 -> M1.9 = 29 Person days

Slack Time

There is 0 slack time for this phase.

Phase 2



Critical Path

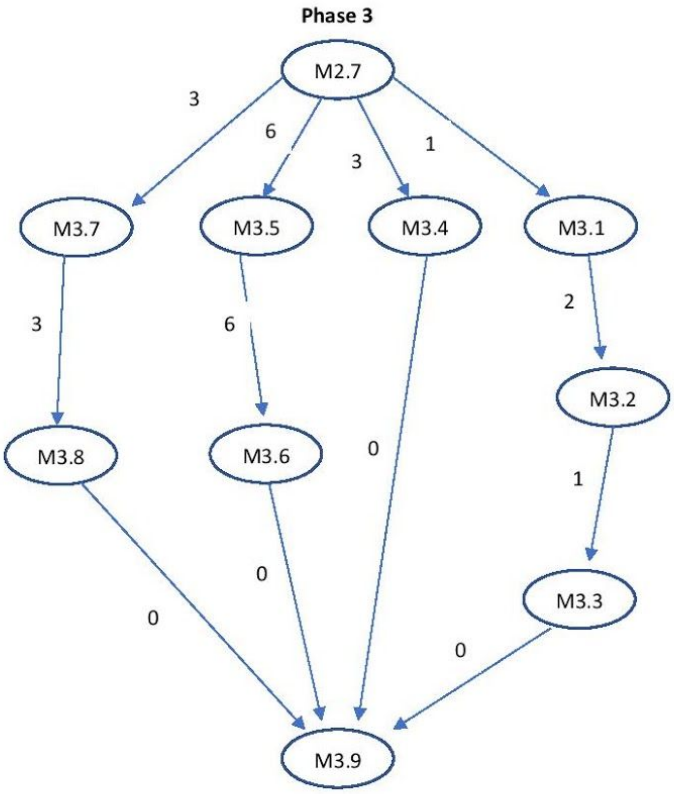
M1.9 -> M2.1 -> M2.2 -> M2.5 -> M2.7 = 12 Person days

Slack Time

M2.1 -> M2.4 -> M2.7 = 6 Person Days

M2.1 -> M2.6 -> M2.7 = 6 Person Days

M2.1 -> M2.3 -> M2.7 = 3 Person Days



Critical Path

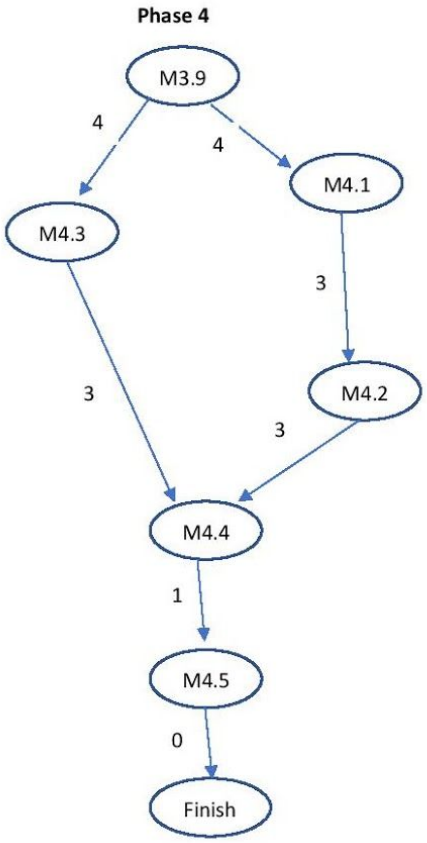
M2.7 -> M3.5 -> M3.6 -> M3.9 = 12 Person days

Slack Time

M2.7 -> M3.7 -> M3.8 -> M3.9 = 6 Person Days

M2.7 -> M3.4 -> M3.9 = 9 Person Days

M2.7 -> M3.1 -> M3.2 -> M3.3 -> M3.9 = 8 Person Days



Critical Path

M3.9 -> M4.1 -> M4.2 -> M4.4 -> M4.5 -> Finish = 11 Person days

Slack Time

M3.9 -> M4.3 -> M4.4 = 3 Person Days

3.2.2 Gantt Chart (NM)

Below is the projected Gantt Chart for the entire project, as well as close-ups of each section (corresponding to the same-numbered phase in the Work Breakdown Structure) for easier viewing. Each task from the Work Breakdown Structure is represented on one row, along with their associated cost, estimated versus actual time needed to complete, any preceding steps that need to be completed before the task in question can be started, and who is currently assigned to work on the task.

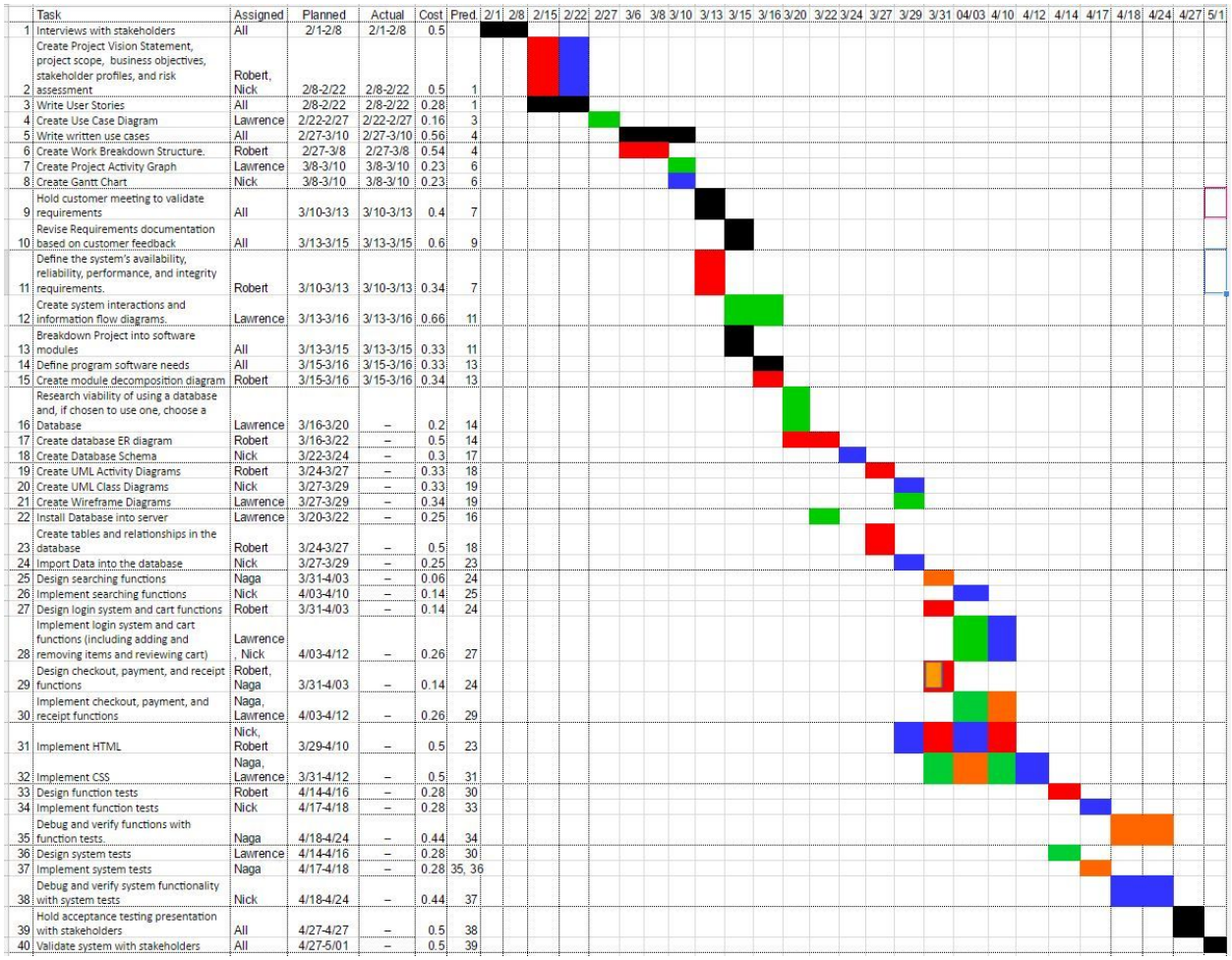
Task costs are represented as a percentage of the total cost for that section. As an example, Step 1.1 has two tasks, each costing 5 units for a total cost of 10. In the Gantt Chart, these are represented as 0.5 each, adding up to 1.

The estimated (as well as actual) time to completion is a numerical range represented by generic “day” numbers rather than specific dates. Any tasks that have not been undertaken yet (such as the coding section) are noted by a single dash, “-”. This range is also represented visually using colored blocks on the far right of the Gantt Chart - the legend indicates which color represents which team member.

Prerequisite steps to the current task are listed by their row number in Gantt Chart, rather than their step identification in the Work Breakdown Structure.

Due to the addition of a team member part way through the project, the timeline for the latter portion of the Gantt Chart had to be revised. Normally, we would need to include additional time for training and integrating the new team member; however, we are fortunate in that they have been working on an identical project (and thus should not need to be “brought up to speed” on requirements and specifications), as well as the fact that we have not begun coding yet (and thus our new team member does not need additional time to learn our code base). Therefore, we have optimistically reduced our development time from the original estimate.

	Task	Assigned	Planned	Actual	Cost	Pred.
1	Interviews with stakeholders	All	2/1-2/8	2/1-2/8	0.5	
2	Create Project Vision Statement, project scope, business objectives, stakeholder profiles, and risk assessment	Robert, Nick	2/8-2/22	2/8-2/22	0.5	1
3	Write User Stories	All	2/8-2/22	2/8-2/22	0.28	1
4	Create Use Case Diagram	Lawrence	2/22-2/27	2/22-2/27	0.16	3
5	Write written use cases	All	2/27-3/10	2/27-3/10	0.56	4
6	Create Work Breakdown Structure.	Robert	2/27-3/8	2/27-3/8	0.54	4
7	Create Project Activity Graph	Lawrence	3/8-3/10	3/8-3/10	0.23	6
8	Create Gantt Chart	Nick	3/8-3/10	3/8-3/10	0.23	6
9	Hold customer meeting to validate requirements	All	3/10-3/13	3/10-3/13	0.4	7
10	Revise Requirements documentation based on customer feedback	All	3/13-3/15	3/13-3/15	0.6	9
11	Define the system's availability, reliability, performance, and integrity requirements.	Robert	3/10-3/13	3/10-3/13	0.34	7
12	Create system interactions and information flow diagrams.	Lawrence	3/13-3/16	3/13-3/16	0.66	11
13	Breakdown Project into software modules	All	3/13-3/15	3/13-3/15	0.33	11
14	Define program software needs	All	3/15-3/16	3/15-3/16	0.33	13
15	Create module decomposition diagram	Robert	3/15-3/16	3/15-3/16	0.34	13
16	Research viability of using a database and, if chosen to use one, choose a Database	Lawrence	3/16-3/20	–	0.2	14
17	Create database ER diagram	Robert	3/16-3/22	–	0.5	14
18	Create Database Schema	Nick	3/22-3/24	–	0.3	17
19	Create UML Activity Diagrams	Robert	3/24-3/27	–	0.33	18
20	Create UML Class Diagrams	Nick	3/27-3/29	–	0.33	19
21	Create Wireframe Diagrams	Lawrence	3/27-3/29	–	0.34	19
22	Install Database into server	Lawrence	3/20-3/22	–	0.25	16
23	Create tables and relationships in the database	Robert	3/24-3/27	–	0.5	18
24	Import Data into the database	Nick	3/27-3/29	–	0.25	23
25	Design searching functions	Naga	3/31-4/03	–	0.06	24
26	Implement searching functions	Nick	4/03-4/10	–	0.14	25
27	Design login system and cart functions	Robert	3/31-4/03	–	0.14	24
28	Implement login system and cart functions (including adding and removing items and reviewing cart)	Lawrence, Nick	4/03-4/12	–	0.26	27
29	Design checkout, payment, and receipt functions	Robert, Naga	3/31-4/03	–	0.14	24
30	Implement checkout, payment, and receipt functions	Naga, Lawrence	4/03-4/12	–	0.26	29
31	Implement HTML	Nick, Robert	3/29-4/10	–	0.5	23
32	Implement CSS	Naga, Lawrence	3/31-4/12	–	0.5	31
33	Design function tests	Robert	4/14-4/16	–	0.28	30
34	Implement function tests	Nick	4/17-4/18	–	0.28	33
35	Debug and verify functions with function tests.	Naga	4/18-4/24	–	0.44	34
36	Design system tests	Lawrence	4/14-4/16	–	0.28	30
37	Implement system tests	Naga	4/17-4/18	–	0.28	35, 36
38	Debug and verify system functionality with system tests	Nick	4/18-4/24	–	0.44	37
39	Hold acceptance testing presentation with stakeholders	All	4/27-4/27	–	0.5	38
40	Validate system with stakeholders	All	4/27-5/01	–	0.5	39



All Team:
Lawrence:
Nick:
Naga:
Robert:

4. Requirements

4.1. User Stories

1. Searching and Search Results:

1.1
As a user, I want to be able to search for books by title. (RH)
Estimate: 5
Priority: 8

1.2
As a user, I want to be able to search for books by keyword. (RH)
Estimate: 1
Priority: 8

1.3
As a user, I want to be able to search for books by ISBN. (RH)
Estimate: 1
Priority: 8

1.4
As a user, I want to be able to search for books by author. (RH)
Estimate: 1
Priority: 8

1.5
As a user, I want to be able to search for books by the course section number associated with the book. (RH)
Estimate: 2
Priority: 8

1.6
As a user, I want to be able to search for books by the professor teaching the course section associated with the book. (RH)
Estimate: 2
Priority: 8

1.7
As the bookstore manager, I want all search results to display a picture of the book, the title of the book, the author of the book, the ISBN of the book, and the quantity of the books in stock (a book out of stock will still be displayed in the search results). (RH)
Estimate: 5
Priority: 5

1.8
As a user, I want to be able to sort my search results alphabetically by textbook name, assigning professor, and author, as well as by price. (NM)
Estimate: 4
Priority: 2

1.9
As a user, I want to be able to click on a book from my search results and see its details including title, author, description, ISBN, associated courses (including professor's name, section, and whether the book is required or recommended (or both)), picture of the book cover, price, format (new, used, rental, or ebook), and number of books in stock. (RH)
Estimate: 5
Priority: 7

2. Shopping Cart:

2.1
As a user, I want to be able to add any amount of books in any format (including new, used, rentals, and EBooks) to my cart. (RH)
Estimate: 7
Priority: 10

2.2
As a user, I want to be able to remove items from my cart. (RH)
Estimate: 3
Priority: 7

2.3
As the bookstore manager, I want the online bookstore to prevent users from being able to add a book that is out of stock to their cart. The system should also prevent users from adding more than the remaining quantity of books to their cart. (RH)
Estimate: 4
Priority: 7

2.4
As a bookstore manager, I want the shopping cart to show the subtotal when there are items in it. The shopping cart will also display the name of each item, a picture of each item, the version of each item(new, used, rental, e-book), and the quantity of each item. (LF)
Estimate: 5
Priority: 9

2.5
As a customer, I want my shopping cart to remember my selections even if I close my browser and come back later to finish shopping. (NM)
Estimate: 8
Priority: 5

3. Checkout and Payment:

3.1
As the bookstore manager, I want to accept payment in the form of PayPal, Visa, MasterCard, or Financial Aid. (RH)
Estimate: 5
Priority: 9

3.1.1 Financial Aid

3.1.2
As the bookstore manager, if a student wants to use financial aid to pay for their order, I want them to be required to log into the online bookstore using their KSU email in order to
verify that they have sufficient funds. (NM)
Estimate: 5
Priority: 6

3.1.3
As the Bursar, I want to be able to send a student's financial aid information to the online bookstore system. (NM)
Estimate: 5
Priority: 8

3.1.4
As the Bursar, I want a student’s financial aid balance to be updated after each purchase (using financial aid as payment) from the online bookstore. (RH)

Estimate: 3
Priority: 5

3.1.5

As the Bookstore manager, I want students to be able to use their financial aid to purchase
items on the online bookstore only if they have enough financial aid to cover the entire order. (NM)
Estimate: 2
Priority: 5

3.1.6

As a user, I want to be able to see my current available financial aid balance before and during checkout. (RH)
Estimate: 3
Priority: 2

3.2

As the bookstore manager, I do not want users to be able to use more than one payment method per cart checkout session. (RH)
Estimate: 3
Priority: 5

3.3

As the bookstore manager, during checkout I want the customer to be required to fill in a valid billing address, shipping address, and credit card information . Valid credit card information is defined as having a 16 digit credit card number, 3 digit verification number, and an expiration date that is later than the date of the online bookstore order. A valid address will have a street number and name, PO box (if applicable), city, state, and zip code. (LF)
Estimate: 7
Priority: 10

3.4

As the bookstore manager, I want the checkout to include the 7% Georgia tax for all sales, regardless of the purchaser's actual location or shipping address. (LF)
Estimate: 3
Priority: 10

3.5

As a warehouse manager, I want every order that requires shipping to be a flat fee of $14.99 using USPS Priority mail. There is no fee for e-books. (LF)
Estimate: 4
Priority: 10

3.6
As a user, I want to my receipt to display on my screen directly after checkout. (NM)
Estimate: 1
Priority: 7

3.7
As a user, I want to receive an email containing my receipt after purchase. (NM)
Estimate: 1
Priority: 9

3.8
As a user, I want my receipt - whether it is received by email or displayed on the screen after my purchase - to contain a link to download any e-book that I purchase in the online bookstore. (NM)
Estimate: 4
Priority: 6

3.9
As a user, I want my receipt - whether it is received by email or displayed on the screen after my purchase - to display the due date for any rental books I rented. (NM)
Estimate: 1
Priority: 5

3.10
As a user, I want to be able to checkout as a guest and not be required to register or log in. (LF)
Estimate: 1
Priority: 3

3.11
As a user, I want to be able to have separate shipping and billing addresses if desired. (NM)
Estimate: 2
Priority: 5

4. Processing and Handling:

4.1
As the administrative assistant, I want the online bookstore to generate a receipt which is defined as a document that contains an order's date, item(s) purchased, customer name,

customer contact information (phone, billing address, and email), rental due date (if applicable), e-book download link (if applicable), and customer shipping address. (RH)
Estimate: 5
Priority: 7

4.2
As the administrative assistant, I want a copy of the receipt saved into a shared folder that is accessible on my desktop. (RH)
Estimate: 5
Priority: 7

4.3
As the warehouse manager, I want each receipt to have a valid shipping address (customer name, PO box number (if applicable), street address, city, state, postal code) that USPS can ship to associated with the purchase. (RH)
Estimate: 3
Priority: 9

4.4
As the bookstore manager, I want each format of the book (new, used, rental, and EBook) to have its own distinct price and inventory quantity. (RH)
Estimate: 2
Priority: 8

4.5
As a warehouse manager, I want to only ship items where the USPS ships (exclusively ship to US and US territories). (LF)
Estimate: 3
Priority: 10

5. General:

5.1
As the bookstore manager, I want the online bookstore to be accessible and useable on all devices (meaning desktops, laptops, tablets, and mobile phones). (LF)
Estimate: 4
Priority: 7

5.2
As a bookstore manager, I want to only sell books for a given semester from the first day of phase 1 registration to the 15th day of classes. (LF)

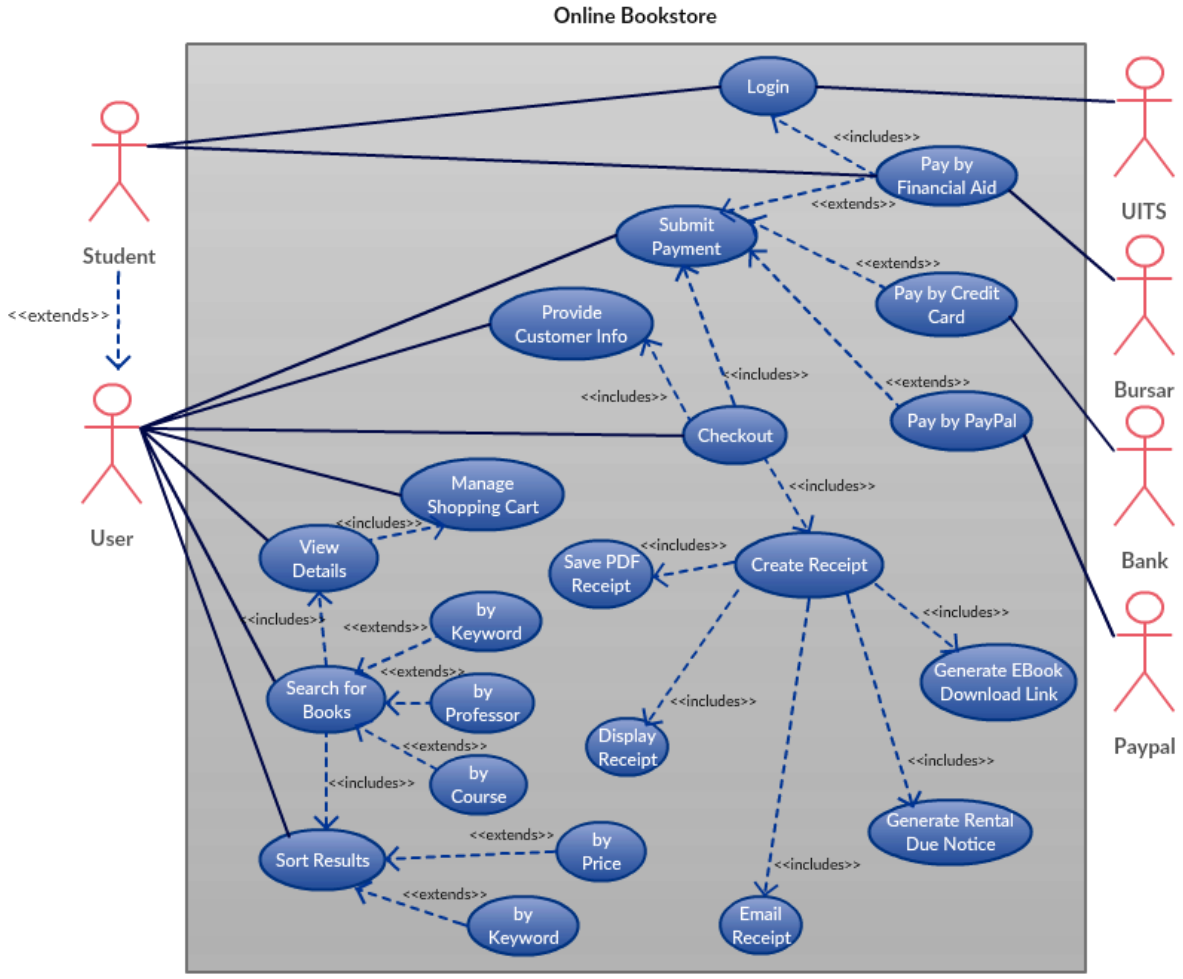
Estimate: 5
Priority: 10

5.3
As a UITS member, I want the online store to conform to KSU branding/style guidelines and standard web practices. (LF)
Estimate: 10
Priority: 10

5.4
As a bookstore manager, I want the website to have a FAQ/Help page. (LF)
Estimate: 6
Priority: 9

5.5
As a bookstore manager, I want the online store to use the CSV file from the registrar to update the books, courses and sections, and include professors for the given course for the semester. (LF)
Estimate: 7
Priority: 10

4.2. Use Case Diagram (LF)



4.3. Written Use Cases

4.3.1 Search

Author: Robert Herber

Date: 3/10/2017

Primary Actor: User

Summary: A user utilizes the Online Bookstore's search feature to search for books matching the specified criteria. A user can search by keyword (either title, author, ISBN, or description), course, or professor. The system returns a result page with all of the books that match the

search criteria.

Assumptions:

1. User is on homepage where the search options are

Preconditions:

1. The user has navigated to the Online Bookstore's web address
2. The user has located the search for books area on the Online Bookstore's front page

Postconditions:

If books matching the search criteria were found:

1. A list of all the books that matched the supplied search criteria will be displayed on the screen for the user
2. The user will be able to click on a book from the results to view detailed information on the book

If no books matching the search criteria were found:

1. The Online Bookstore displays "No results found" on the screen for the user

Normal Flow:

1. The user enters text into the search by text search bar
2. The user clicks the "search" button to start the search

Subflow 1:

1. The user picks a course name from the search by course name drop down box
2. The user clicks the "search" button to start the search

Subflow 2:

1. The user picks a professor from the search by professor drop down box
2. The user clicks the "search" button to start the search
3. The Online Bookstore searches through the .csv file that contains the book data and retrieves all of the books that match the search criteria
    a. The system opens the .csv file containing all the book data
    b. The system parses through the .csv file to find all books containing the passed in search criteria from the search bar
    c. The search results are alphabetized by title
    d. The alphabetized results are loaded into the search results page
4. The Online Bookstore displays the search results page from step 3 on the screen for the user

Exceptional Flow:

1. Same as normal flow
2. Same as normal flow
3. Same as normal flow
4. The Online Bookstore searches through the .csv file that contains the book data and finds no books matching the search criteria
5. The Online Bookstore displays "No results found" on the screen for the user

4.3.2 Sort

Author: Lawrence Ferretti

Date: 3/12/2017

Actor: User

Summary: The user can sort the search results by price, title, or author. Sorting by price will allow the user to sort by the highest to lowest price or lowest to highest price. Sorting by title, or author will allow the user to sort alphabetically from A to Z or from Z to A.

Flow of Events:

1. The user clicks on the header to either sort by:
    a. Price from high to low
    b. Price from low to high
    c. Author from A to Z
    d. Author from Z to A
    e. Title from A to Z
    f. Title from Z to A
2. The results will be sorted based on the input from the user
3. The sorted results will be displayed

Exceptional Flow:

1. If the search results contain multiple results with the same price, the sort will then arrange the multiples in alphabetical order from A to Z by default.
2. If the search results contain multiple results with the same title, or author, the sort will not change the arrangement of the multiples.
3. The sort drop-down menu does not appear for the user to interact with it.

Triggers: The sort drop-down menu will appear after the system has populated search results. Once the sort drop-down menu has appeared, the user can then interact with it and choose an option for sorting.

Assumptions: The sort drop-down has displayed correctly and is interactable with the user after the search results yield is greater than one.

Preconditions: The use case 4.3.1(search) has been completed and the books matching the search criteria were found.

Postconditions: The search results will be displayed in a sorted order, based on the input from the user.

4.3.3 Checkout

Author: Nick Murphy

Date: 3/13/2017

Primary Actors: User
Student

Additional Actors: Bank/Financial Institution
PayPal
Bursar/University Financial Aid System

Summary: A user purchases their textbooks via the Online Bookstore's provided payment methods (Credit Card, PayPal, or Financial Aid) and receives confirmation. The user has at least one textbook in their shopping cart. The "Checkout Now" button will only be visible when there is at least one textbook in the user's shopping cart – at which point, the user can click the button to start the process of checking out.

Assumptions:

1. Student users have a Financial Aid award from Kennesaw State University.

Preconditions:

1. The user is on the Shopping Cart page of the Online Bookstore.
2. The user has completed the **Shopping Cart** use case and has at least one textbook in their shopping cart.

Triggers: None.

Normal Flow:

1. The user clicks the "Checkout Now" button.
2. The Online Bookstore system uses the CSV database to calculate the subtotal cost for the textbook order, then applies the 7% Georgia Sales Tax
3. if there any non-digital books in the order, the Online Bookstore system applies a $14.95 flat fee to the subtotal
4. The system displays a summary of the order including the number, quantity, type and price of each book, shipping costs, tax, and an order total - as well as a “Pay Now” button.
5. The user clicks the "Pay Now" button
6. The system displays the shipping and billing address entry page with the following entry fields: Name, Email Address, Shipping Address, Billing Address and a checkbox to "Use Shipping Address as Billing Address.

7. The user enters their name in the Name entry field
8. The user enters their email address in the Email Address entry field
9. The user enters their shipping address information in the Shipping Address entry fields

Subflow 2a:

- The user clicks the checkbox with the "Use Shipping Address as Billing Address" label
- The Online Bookstore copies the information entered into the Shipping Address entry fields into the identical Billing Address entry fields

Subflow 2b:

- The user enters a different address in the Billing Address entry fields compared to the values in the Shipping Address fields

10. The user clicks the "Next" button
11. The Online Bookstore places the user-entered information into temporary storage
12. The Online Bookstore displays a page with three button options:
    - "Pay with Credit Card"
    - "Pay With PayPal"
    - "Pay with Financial Aid".

Subflow 3a:

a. The user clicks the "Pay with Credit Card" button
b. The Online Bookstore displays a "Submit" button, two radio buttons labeled "Visa" and "Mastercard" as well as three text fields labeled:

- "Credit Card Number"
- "Credit Card Expiration Date"
- "Credit Card CVC"

c. The user enters their Credit Card Number into the labeled field
d. The user enters their Credit Card Expiration Date into the labeled field
e. The user enters their Credit Card CVC into the field

Subflow 3.1a:

- The user clicks the "Visa" radio button

Subflow 3.2a:

- The user clicks the “MasterCard” radio button

f. The user clicks the "Submit Button"
g. The Online Bookstore send the information (along with the final total generated in Step 2) to the user's bank or financial institution
h. The Online Bookstore receives a transaction confirmation from the financial institution gateway.

Subflow 3b:

a. The user clicks the "Pay with PayPal" button
b. The Online Bookstore sends the user to PayPal's payment system
c. The user completes the payment process defined by PayPal
d. The Online Bookstore receives a transaction confirmation from PayPal

Subflow 3c:

a. The user clicks the "Pay with Financial Aid" button
b. The Online Bookstore send the user to the University's Financial Aid system
c. The user logs in using their University username and password
d. The user completes the payment process defined by the University
e. The Online Bookstore receives a transaction confirmation and the total amount of financial aid available for that user back from the University

f. The Online Bookstore verifies that the amount of financial aid available is greater than or equal to the final order total generated in Step 3 of the Normal Flow

13. The Online Bookstore deducts the quantity of physical textbooks ordered from the CSV
14. The Online Bookstore generates a receipt of the order including the number, quantity, type and price of each book, tax, and an order total

Subflow 4a:

a. The Online Bookstore also includes a shipping rate if any textbooks purchased were physical

Subflow 4b:

a. The Online Bookstore also includes a shipping rate and a return date if any textbooks purchased were rentals

Subflow 4c:

a. The Online Bookstore also includes download links for all digital textbooks

15. The Online Bookstore saves the receipt created in Step 13 in PDF form to a predefined location on the server
16. The Online Bookstore sends an email to the email address provided in Step 7, which contains the receipt created in Step 13
17. The Online Bookstore displays a page to the user with a confirmation of the order and the receipt information generated in Step 13.
18. The Online Bookstore removes all textbooks from the user's shopping cart
19. The Online Bookstore clears out all user-related data that was temporarily stored to complete the Checkout process

Exceptional Flow:

1. **All**: If the user does not have enough money to complete the textbook purchase, the Online Bookstore displays an error message informing the user of insufficient funds
2. **All**: If the user's shipping address is outside the United States or its territories (or is an invalid address entirely), the Online Bookstore displays an error message informing the user that the Shipping Address provided can not be used, and does not allow progression to the next step in the flow.
3. **Subflow 3a**: If the Credit Card information is rejected by the bank or financial institution, the Online Bookstore displays an error message informing the user that their credit card information is invalid.
4. **All**: If the Online Bookstore is unable to access the outside payment system (due to issues with the outside payment system), the Online Bookstore displays an error message informing the user that the payment process can not be completed at this time due to an inability to connect to (**Subflow 3a**: the user's bank/financial institution; **Subflow 3b**: PayPal; **Subflow 3c**: the University's Financial Aid system).
5. **All**: If the user clicks the "Back" Button on their browser after selecting a payment option but before they have actually paid, the Online Bookstore goes to Step 14 of the Normal Flow.
6. **All**: If the user clicks the "Back" Button on their browser after clicking the "Pay Now" button but before choosing a payment option, the Online Bookstore goes to Step 6 of the Normal Flow.
7. **All**: If the user clicks the "Back" Button on their browser before clicking the "Pay Now" button, the use case ends.

Postconditions:

1. A receipt has been generated, saved to the server as a PDF, displayed for the user to print, and sent to the user's email address if provided
2. The Online Bookstore has received a guarantee of payment for the textbooks from the user's payment provider (either their bank/financial institution, PayPal, or the university itself)
3. The quantity of textbooks that were ordered (excluding digital textbooks) has been deducted from the Online Bookstore's inventory
4. All user information and textbooks in the user's shopping cart have been cleared from the Online Bookstore System

4.3.4 Login

Author: Robert Herber

Date: 4/13/2017

Primary Actor: User

Summary: A student who wishes to use financial aid to pay for their order is required to login. The login system will take the student's entered information and will validate it against school records. I student who enters information that is not consistent with school records will not be able to log in.

Assumptions:
1. The user is a student with a valid login ID and password.

Preconditions:
1. The student is not already logged in.
2. The student has added items to their cart and is at the "how would you like to pay?" page.

Postconditions:
If success:
1. The student's login information is authenticated.
2. The student is now logged in and can use financial aid to pay for orders.

If failure:
1. The system will deny the student's login request.

Normal Flow:
1. The student clicks the "Financial Aid" button from the "How would you like to pay?" page
2. The system displays the login page on the screen
3. The student enters his/her username and password and clicks the login button
4. The system parses the data file containing the student login information and finds the student's entered information matches with the information in the file
5. The student is now logged into the system and the system displays the financial aid

summary page

Exceptional Flow:

1. Same as normal flow
2. Same as normal flow
3. Same as normal flow
4. The system parses the data file containing the student login information and finds the student's entered information does not match with the information in the file
5. The system denies the student's login request and displays the login failure page

4.3.5 Add Item to the Shopping Cart

Author: Robert Herber

Date: 4/13/2017

Primary Actor: User

Summary: The user will add an item to the shopping cart. The user can also change the quantity of items in the shopping cart.

Assumptions:

1. The Search for Books use case and View Details use case did not enter an exceptional flow

Preconditions:

1. The user has completed the Search for Books use case and View Details use case

Postconditions:

1. The user has added an item and/or changed the quantity of the an item to his/her shopping cart

Normal Flow:

1. The user clicks on the radial button next to the format of the book he/she would like to add to his/her cart
2. The user clicks on the "Add to Cart" button
3. The system parses the inventory file to ensure there are enough of the selected item in stock
4. The system adds the selected item to the user's cart

Alternate Flow 1:

1. Same as normal flow
2. Same as normal flow
3. Same as normal flow
4. Same as normal flow

5. The user clicks the shopping cart dropdown menu
6. The user clicks on the quantity box next to an item in the cart
7. The user enters a number for the quantity of the item that he/she wants
8. The system parses the inventory file to ensure there are enough of the selected item in stock
9. The system updates the shopping cart to reflect the new quantity

Exceptional Flow:
1. Same as normal flow
2. Same as normal flow
3. Same as normal flow
4. The system finds that there are not enough items in stock and displays “not enough items in stock” to the user. The item is not added to the user’s shopping cart

4.3.6 Remove Item From Cart

Author: Robert Herber

Date: 4/13/2017

Primary Actor: User

Summary: A user removes item(s) from his or her shopping cart.

Preconditions:
1. The user must have at least 1 item in his or her cart

Postconditions:
1. The user has removed the desired amount of items from his/her cart

Normal Flow:
1. The user clicks the shopping cart drop down menu
2. The user clicks the quantity box next to the item to be removed
3. The user enters a quantity less than the current quantity in the box
4. The user clicks the “Update Cart” button
5. The system updates the shopping cart to reflect the changes

Exceptional Flow:
1. Same as normal flow
2. Same as normal flow
3. The user enters a negative value into the quantity box
4. The user clicks “Update Cart”
5. The system displays “Invalid quantity”

4.3.7 View Details

Author: Robert Herber

Date: 3/28/2017

Primary Actor: User

Summary: A user clicks on a search result to view more details about the selected item. The user is then directed to a details page that displays the title, author, ISBN, description, associated courses (including professor's name, section, and whether the book is required or recommended (or both)), picture of the book cover, price, format (new, used, rental, or ebook), and number of books in stock.

Assumptions:

1. The search for books use case (use case 4.3.1) returned matching books and did not enter an exceptional flow.

Preconditions:

1. The user has completed a search for books use case (use case 4.3.1).
2. The search for books for books use case results are displaying on the screen.

Postconditions:

1. A view details page is displaying on the user's screen.

Triggers: The user has clicked on a search result.

Normal Flow:

1. The user browses through the list of items returned from the search for books use case (use case 4.3.1).
2. The user picks an item where he/she would like to view more details.
3. The user clicks on the item's name or picture.
    a. The system parses the .csv file containing the inventory information and loads the information associated with the clicked on item into an array.
4. The system directs the user's web browser to the view details page
    a. The system loops through the array with the item's information and displays it on the screen
5. The view details page displays the item's title, author, ISBN, description, associated courses (including professor's name, section, and whether the book is required or recommended (or both)), picture of the book cover, price, format (new, used, rental, or ebook), and number of books in stock on the user's screen.

5. Preliminary and Detailed Design

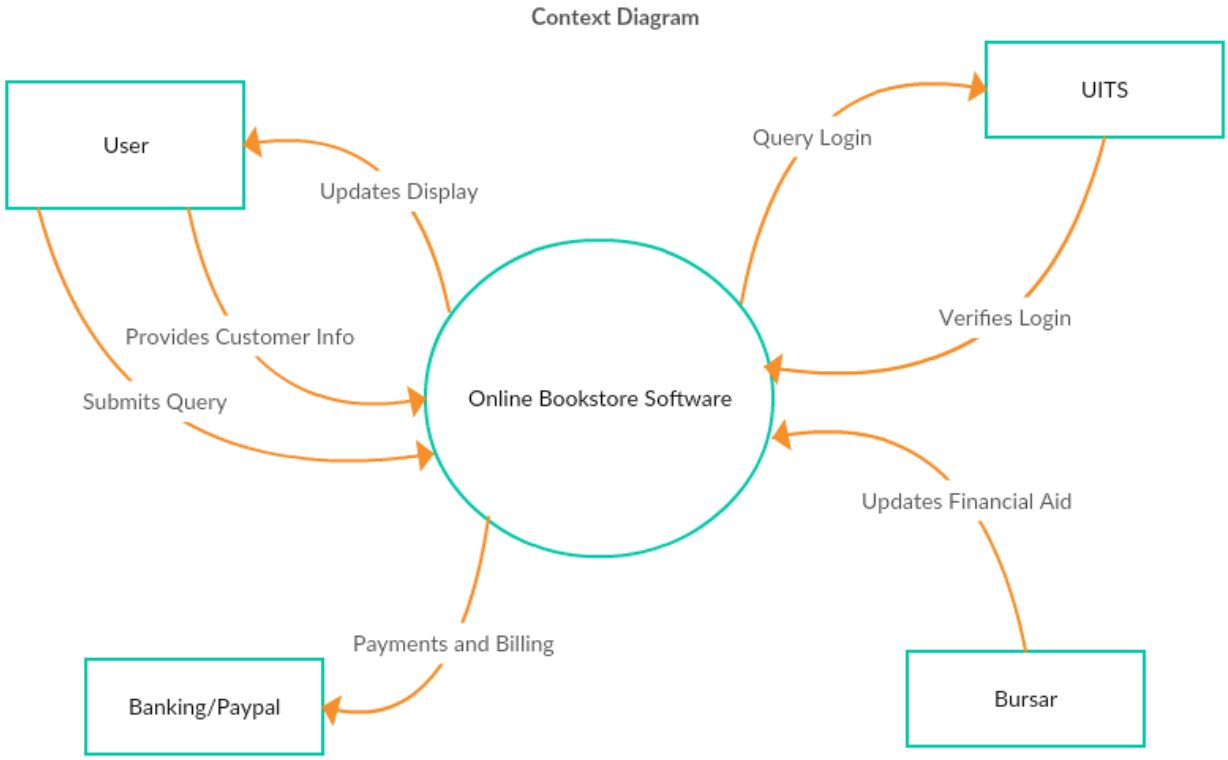
5.1 Project Environment

5.1.1 Operating Environment

The system will be a multi-tier architecture and be available globally to customers via the world wide web. All that is needed to access the system is either a mobile device or computer and an internet connection. The data for the system is only derived from a csv file and the system does not use a database like every single application that exists in this cloud-based modern age. The timing and connection of the users is of little consequence, the first ones to successfully place an order will receive the items. If the supplies are low, the person with the most consistent and fastest connection speed will be able to purchase while someone with a poor connection might not be able to buy before the item is out of stock.

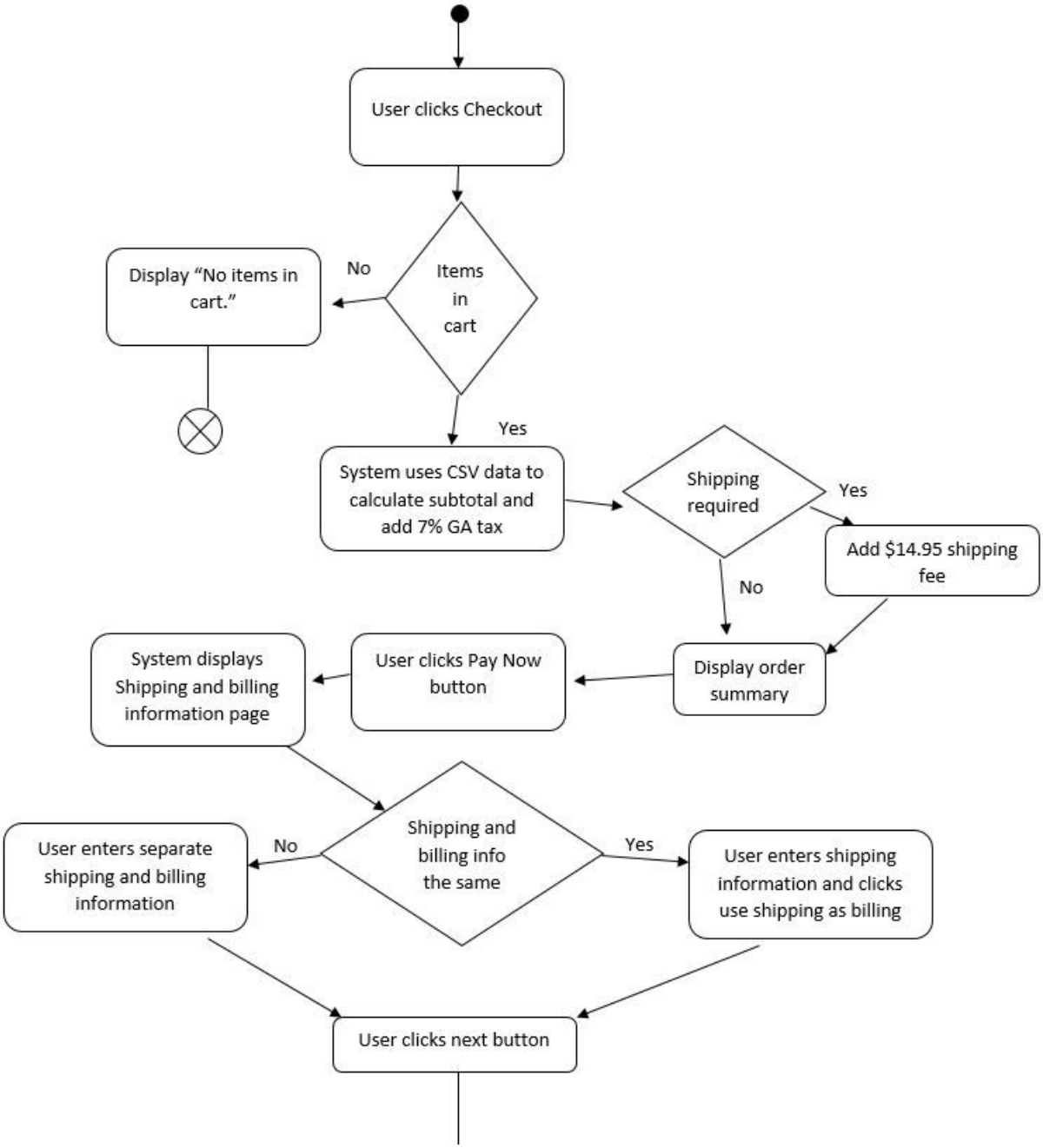
If the system experiences a loss of connection, then the system will be unavailable to customers until the connection is restored. There are no data protection requirements needed, but for security there is a login to the system that will allow students to access their financial aid balances and use those funds to pay for the items.

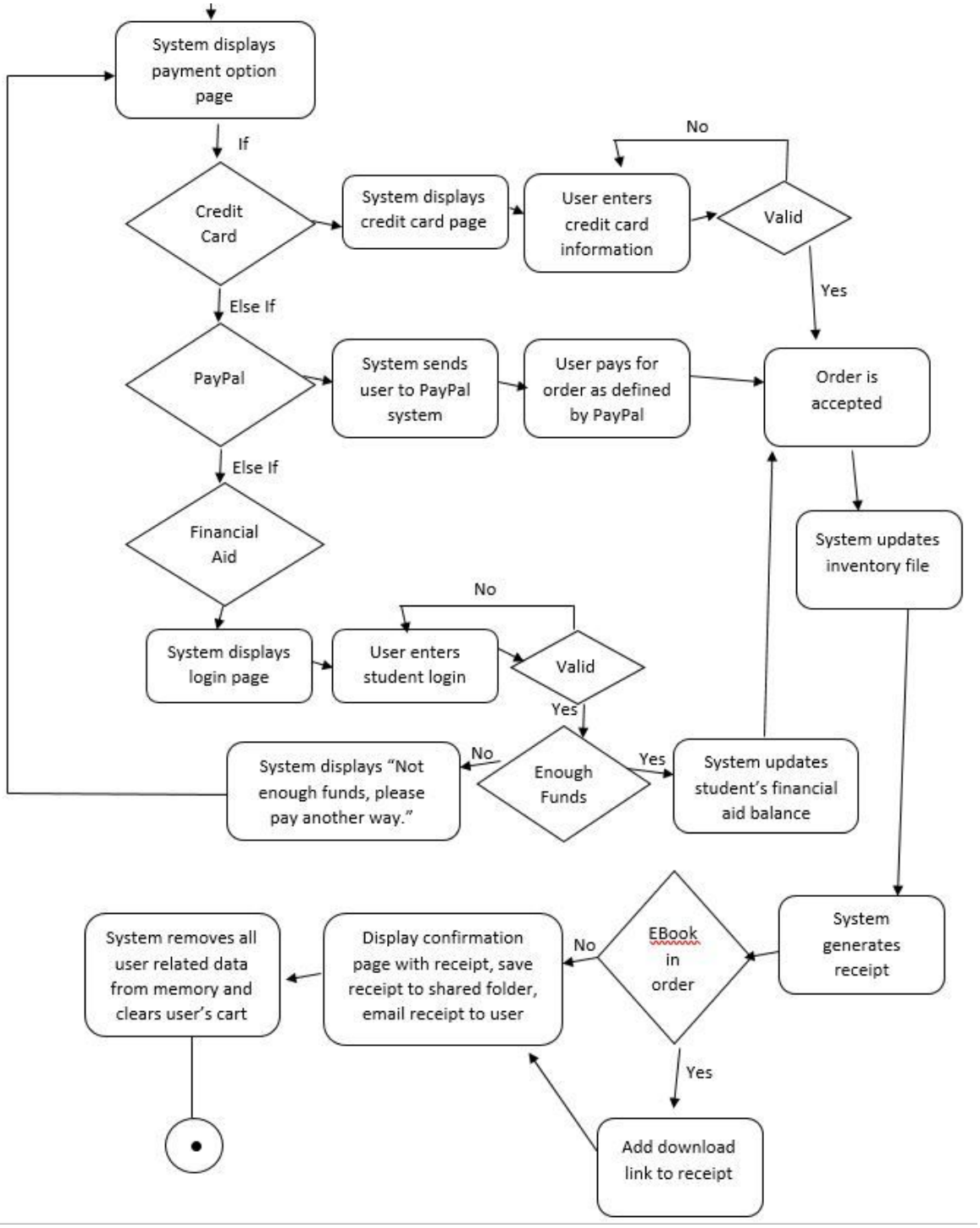
5.1.2 System Interactions and Information Flow



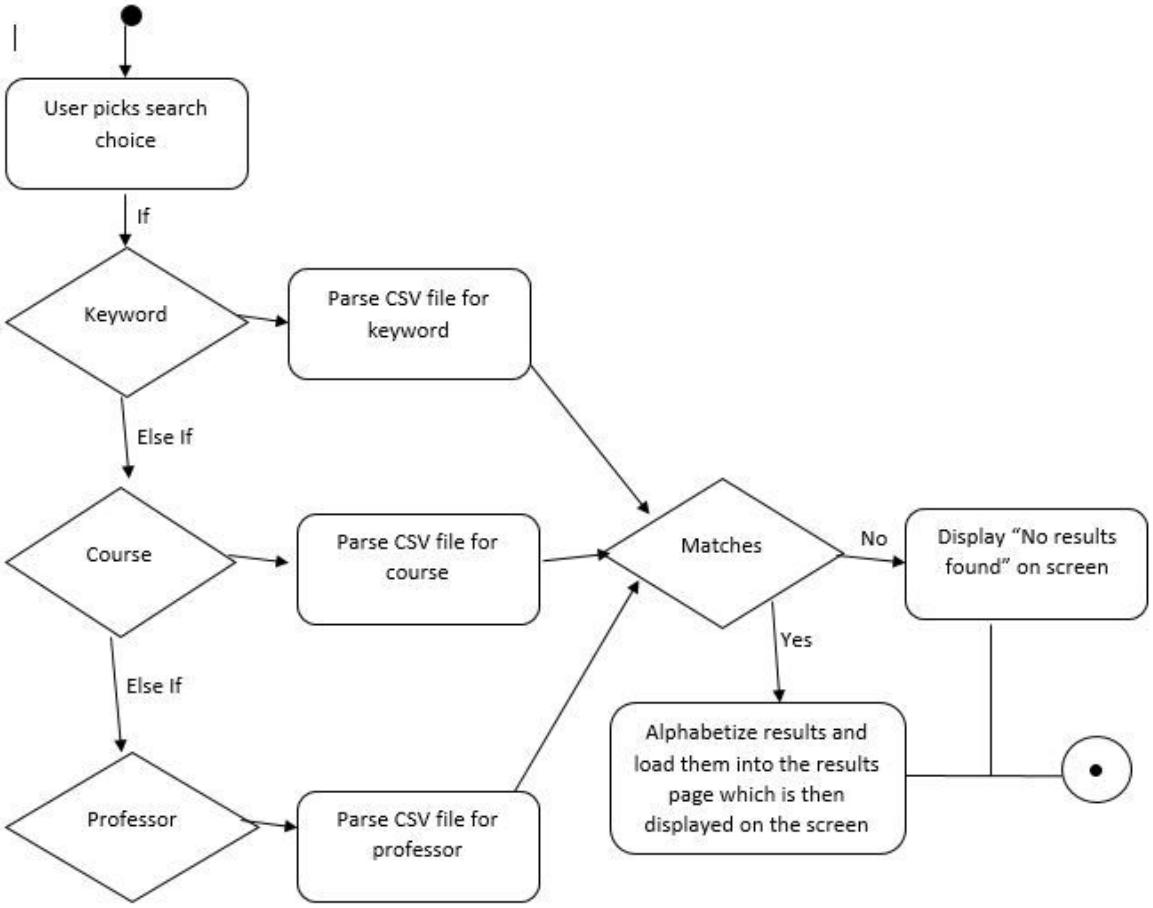
5.2 UML Activity Diagrams (RH)

5.2.1 Checkout:

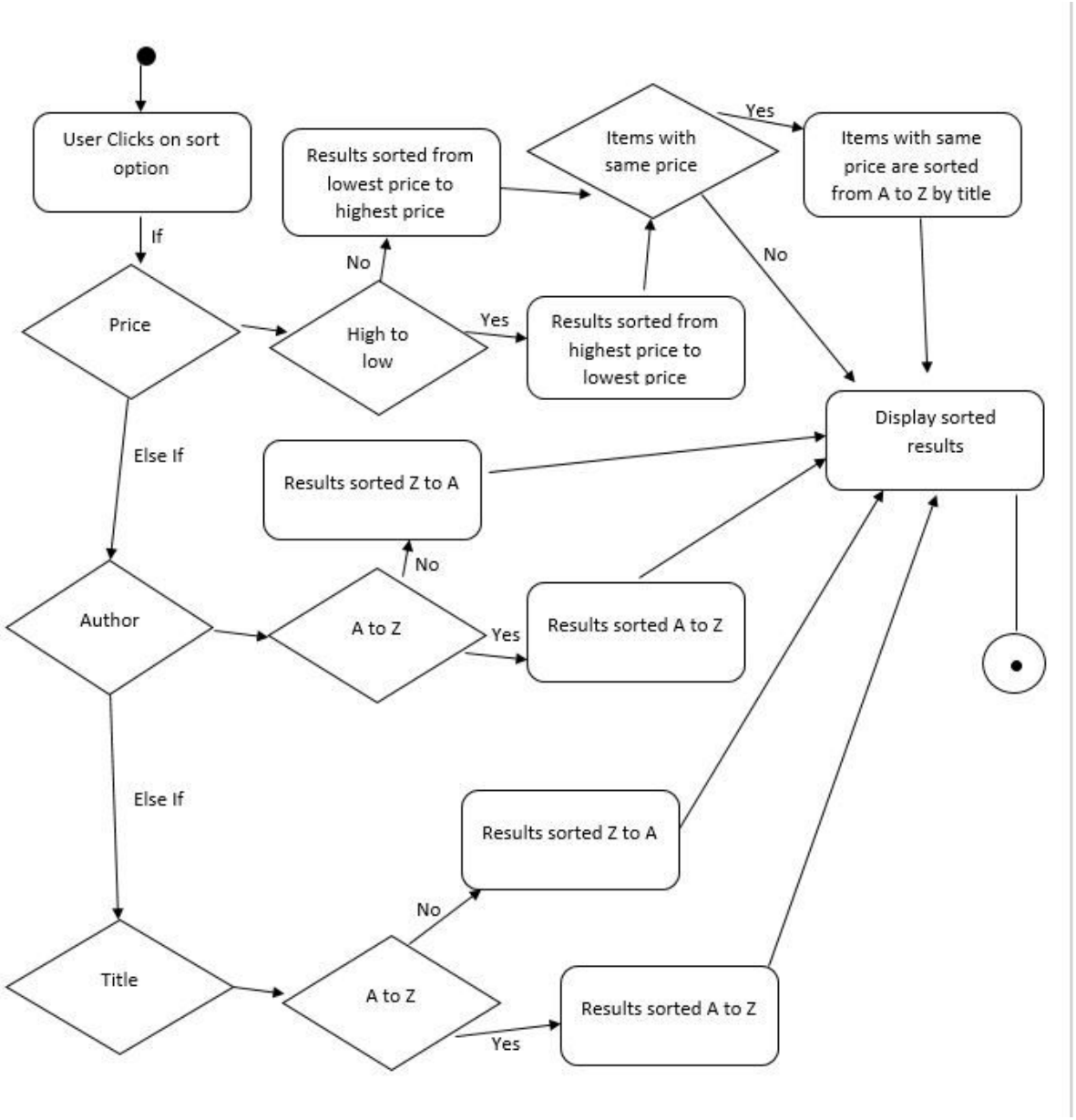



System displays payment option page
If
Credit Card
System displays credit card page
User enters credit card information
No
Valid
Yes
Else If
PayPal
System sends user to PayPal system
User pays for order as defined by PayPal
Order is accepted
Else If
Financial Aid
System updates inventory file
No
System displays login page
User enters student login
Valid
Yes
No
Enough Funds
Yes
System displays "Not enough funds, please pay another way."
System updates student's financial aid balance
System generates receipt
EBook in order
No
Display confirmation page with receipt, save receipt to shared folder, email receipt to user
System removes all user related data from memory and clears user's cart
Yes
Add download link to receipt

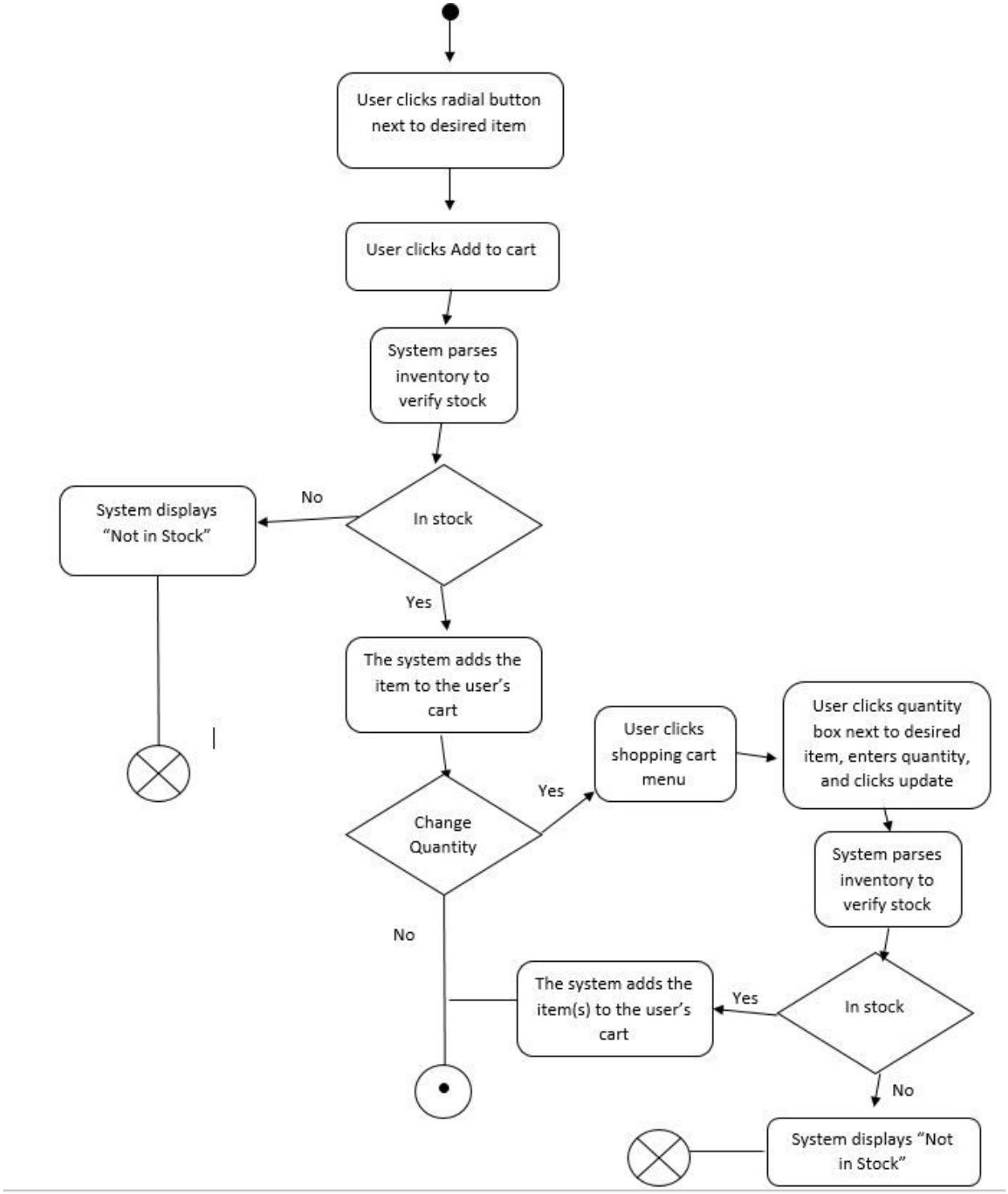
5.2.2 Search:


User picks search choice
If
Keyword
Parse CSV file for keyword
Else If
Course
Parse CSV file for course
Else If
Professor
Parse CSV file for professor
Matches
No
Display "No results found" on screen
Yes
Alphabetize results and load them into the results page which is then displayed on the screen

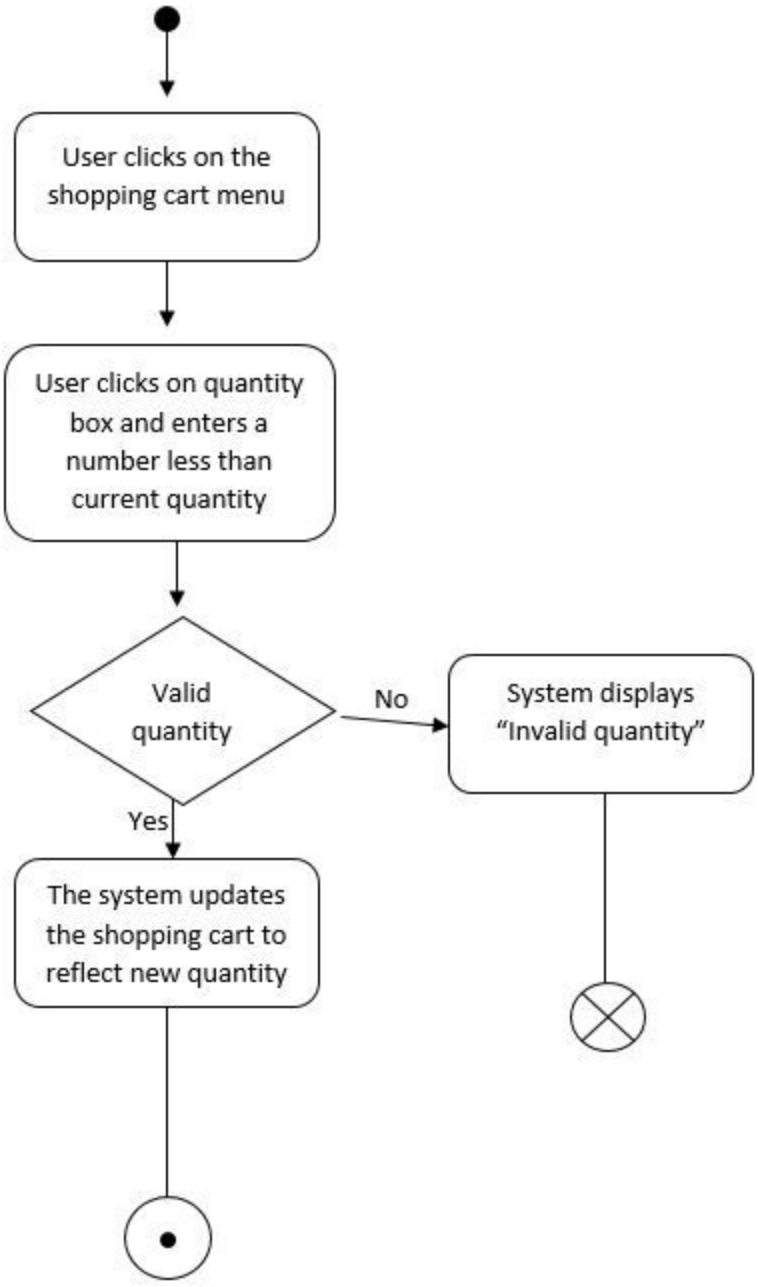
5.2.3 Sort:


User Clicks on sort option
If
Price
High to low
No
Yes
Results sorted from lowest price to highest price
Items with same price
Yes
No
Items with same price are sorted from A to Z by title
Results sorted from highest price to lowest price
Display sorted results
Else If
Results sorted Z to A
No
Author
A to Z
Yes
Results sorted A to Z
Else If
Results sorted Z to A
No
Title
A to Z
Yes
Results sorted A to Z

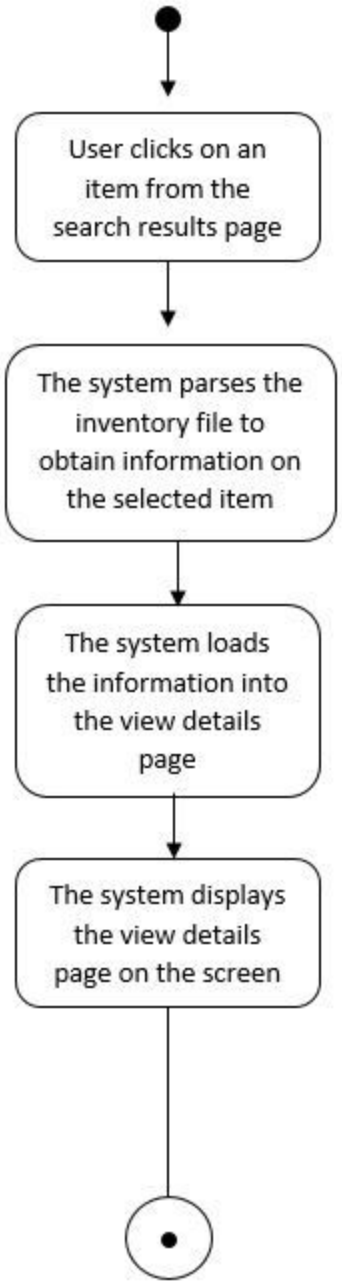
5.2.4 Add Item to Cart



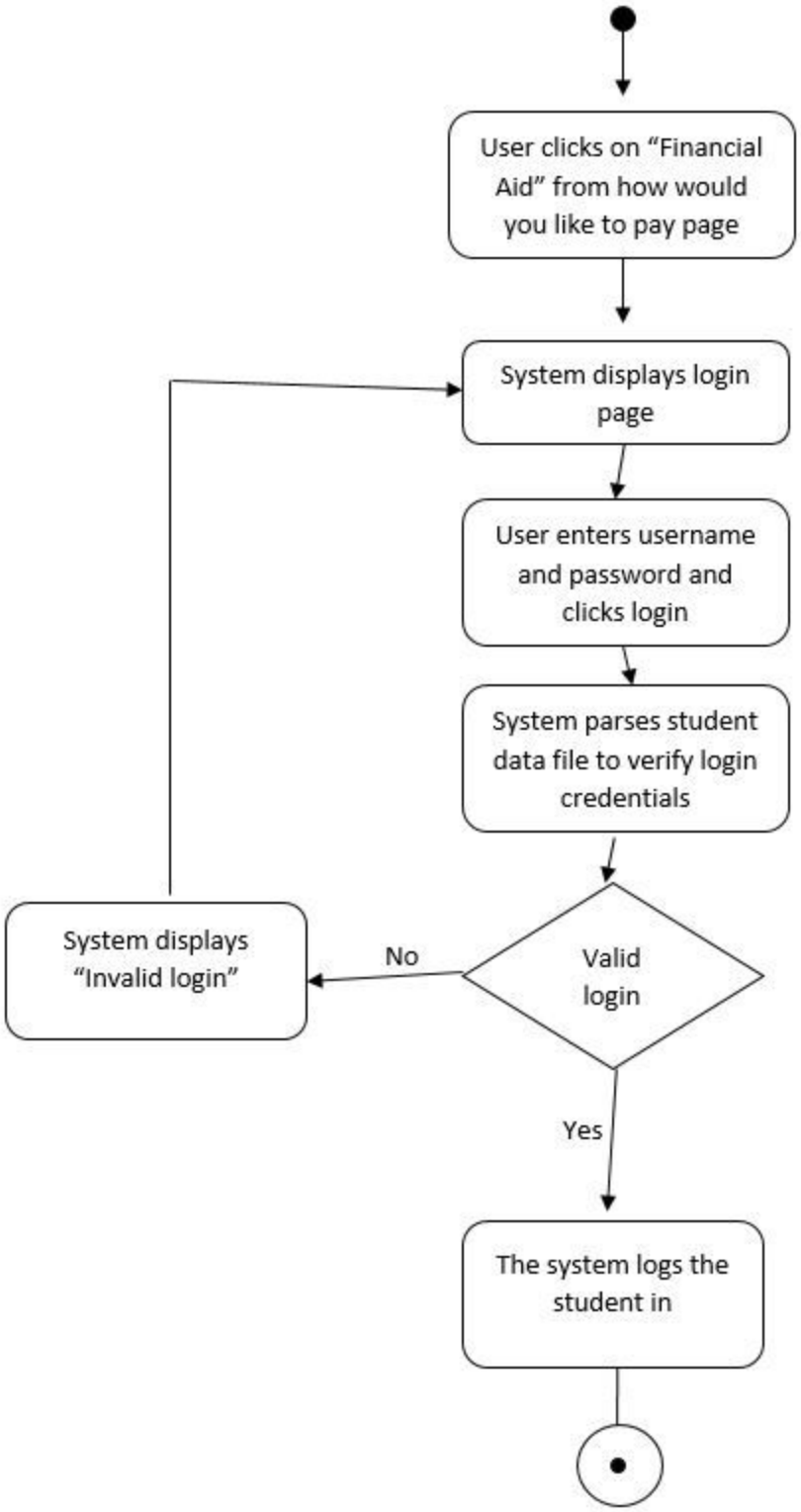
5.2.5 Remove Item from Cart



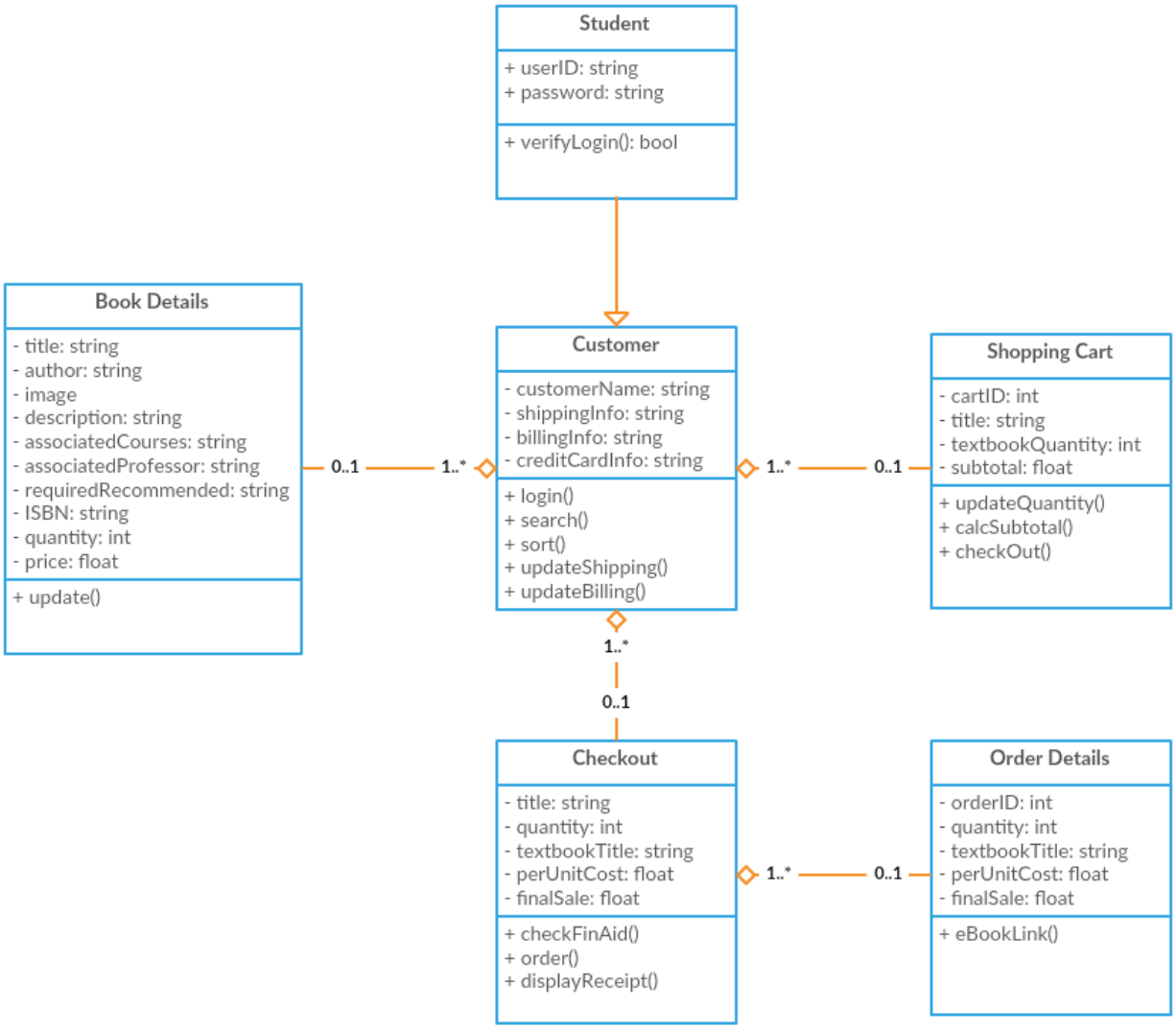
5.2.6 View Details



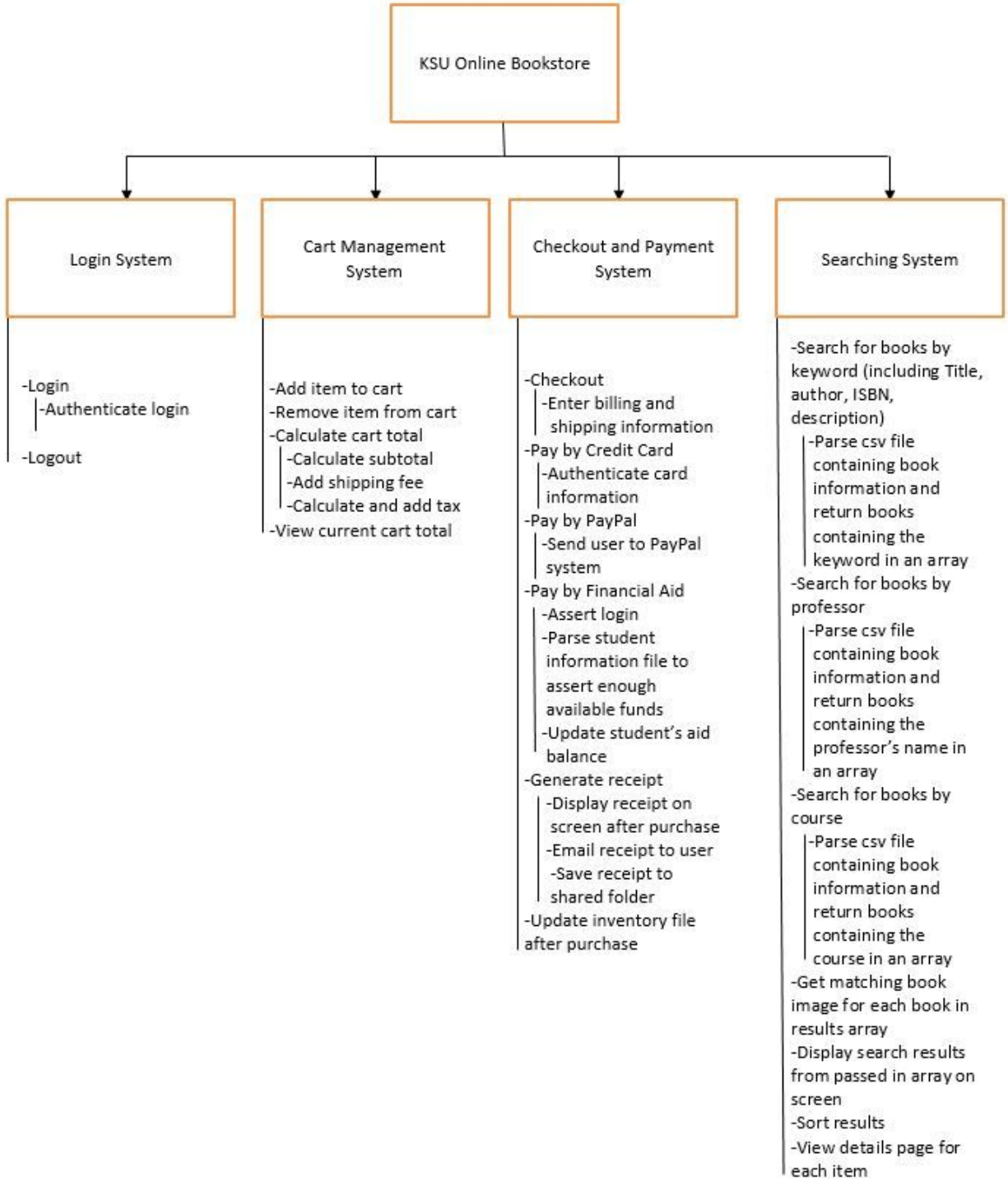
5.2.7 Login


User clicks on "Financial Aid" from how would you like to pay page
System displays login page
User enters username and password and clicks login
System parses student data file to verify login credentials
Valid login
No
System displays "Invalid login"
Yes
The system logs the student in

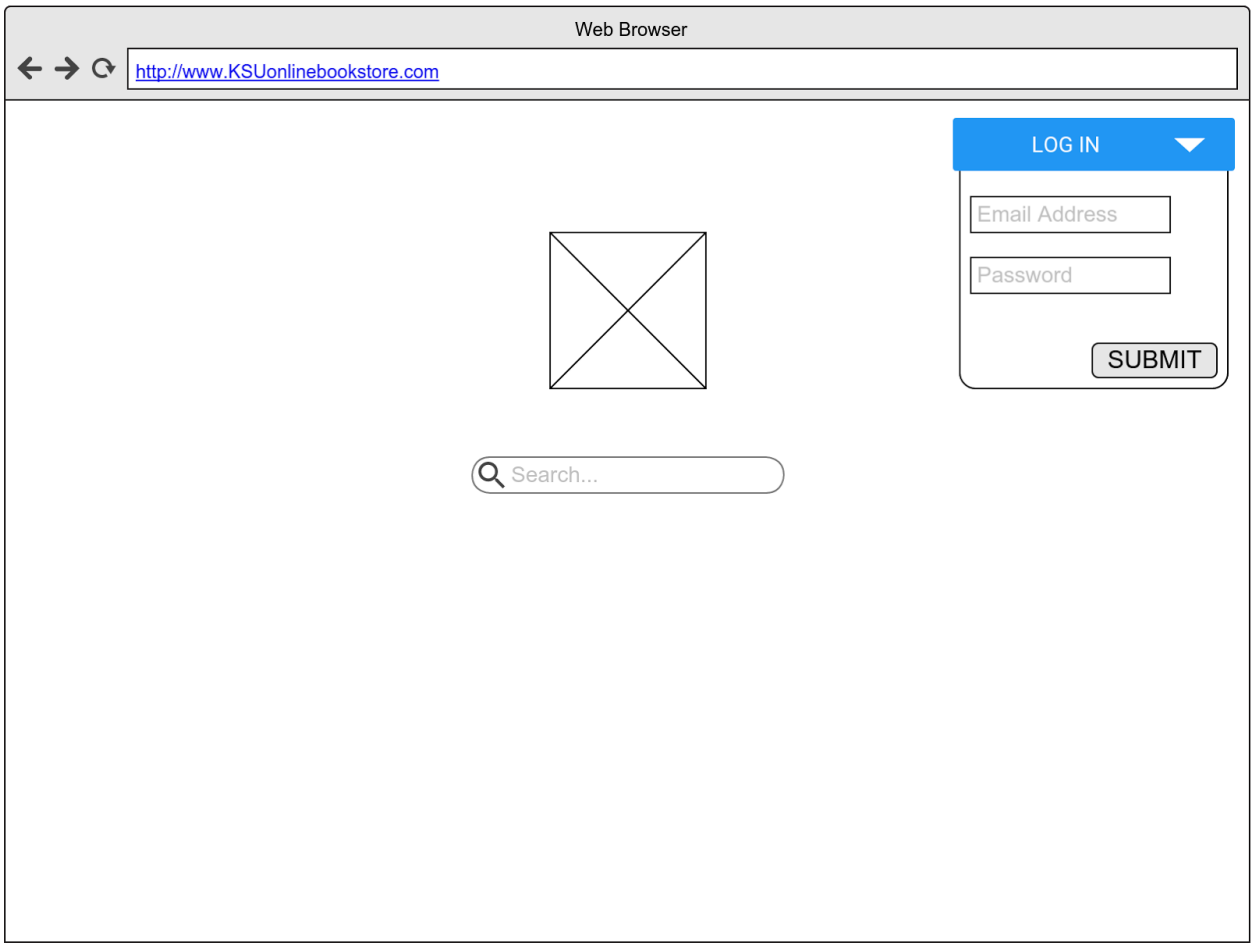
5.3 UML Class Diagrams



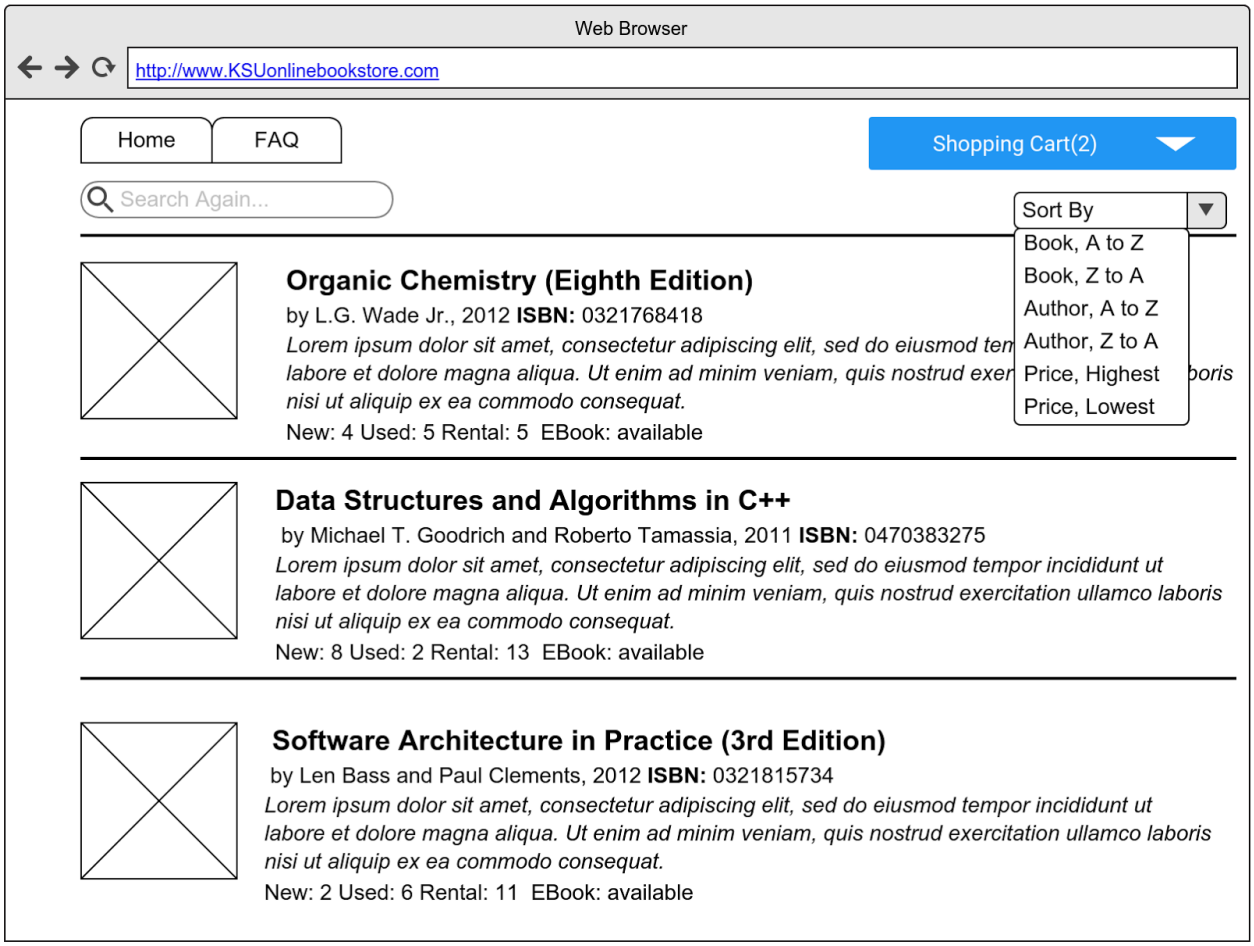
5.4 Software Modules and Components



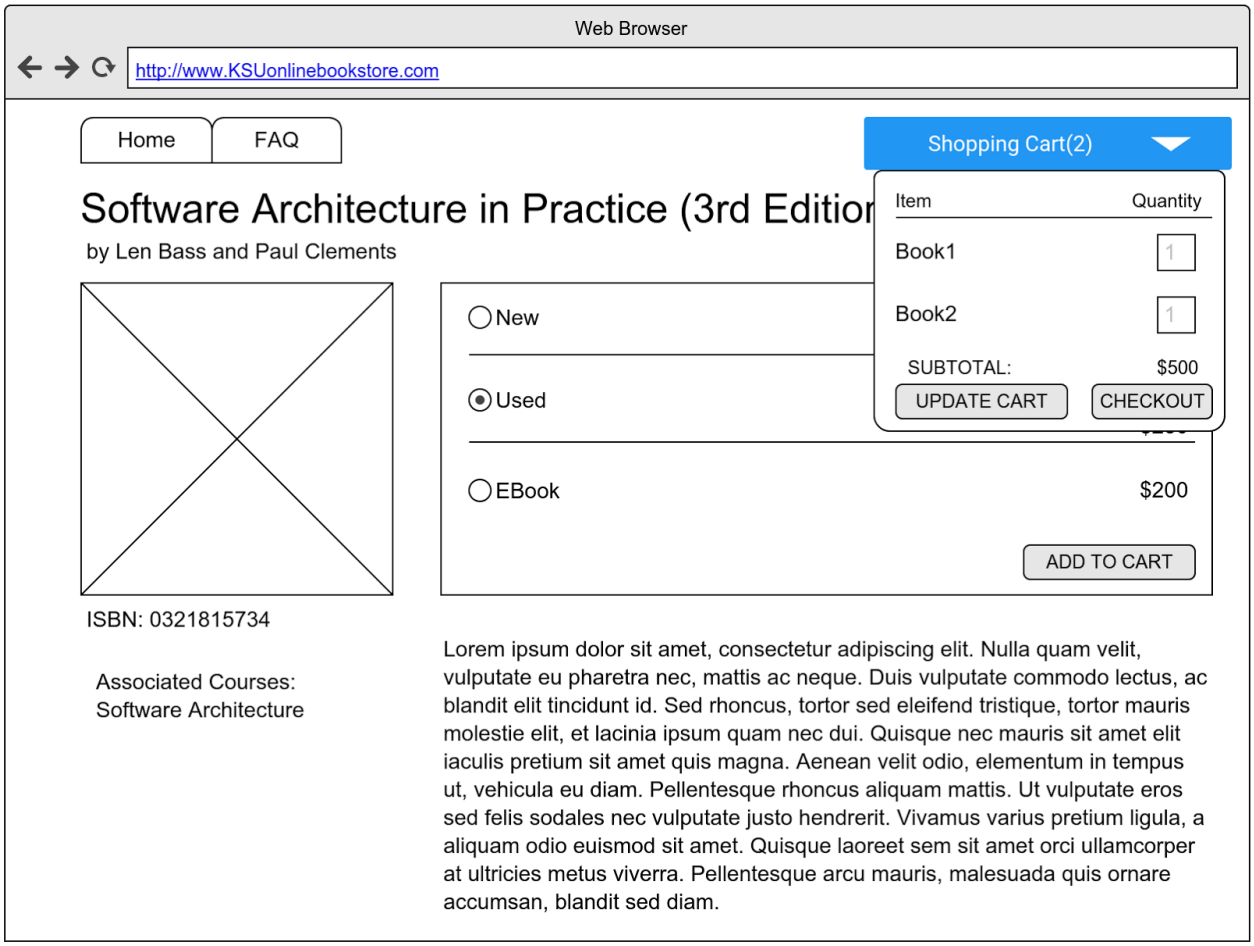
5.5 Wireframe Diagrams



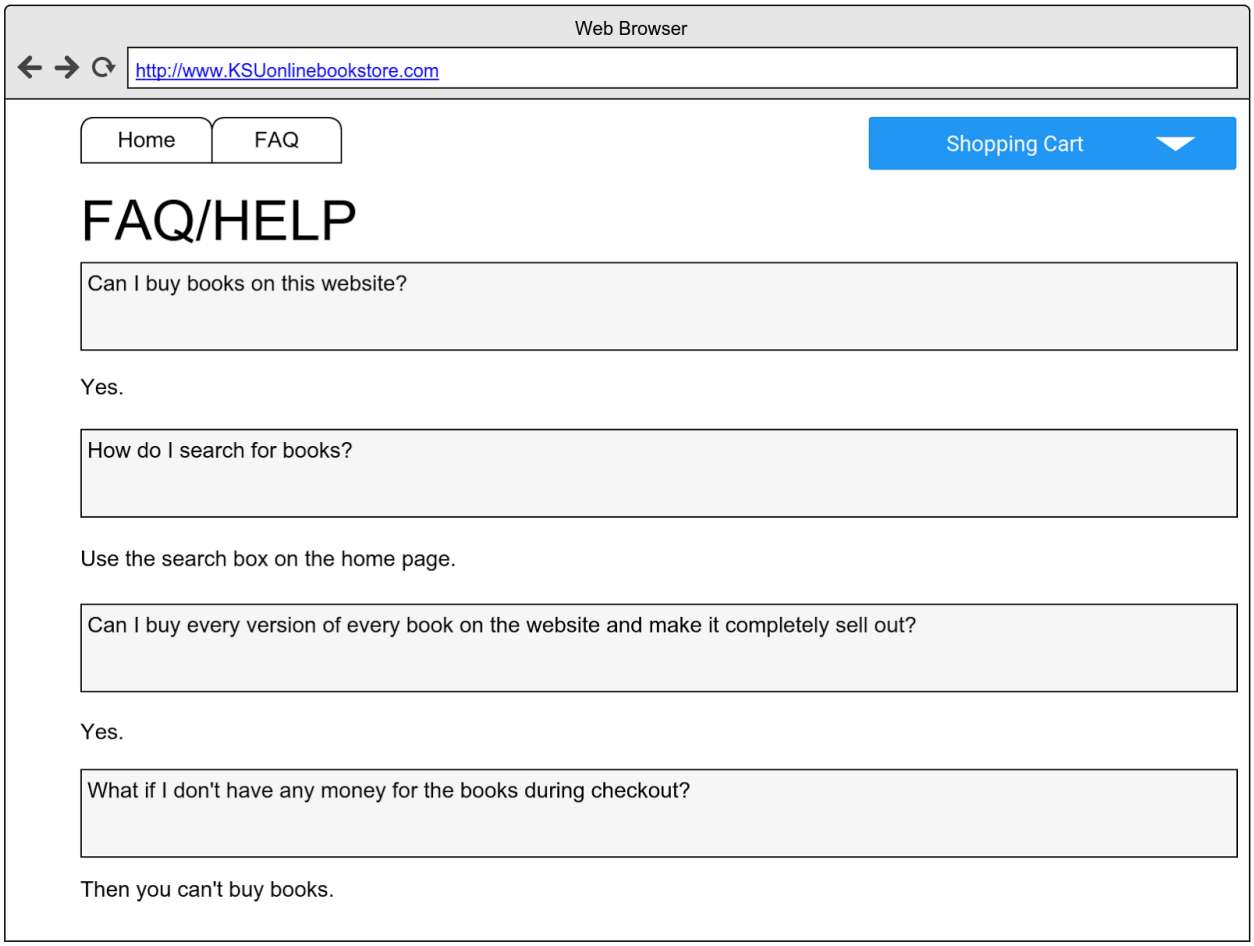
This is the home page of the system. The user can interact with the search box and the log in drop-down menu if they wish. The placeholder is for the KSU logo that UITS will provide for the system to comply with the KSU guidelines. (LF)



This is the search result screen. Once the user searches from the home screen or searches again this screen, they are taken to this search result screen. The search results show the picture, title, author, ISBN, quantity, and price. The user can interact with the shopping cart via its drop-down menu, the search box to search again, or the sort drop-down menu to choose to sort the results. They can also interact with the tabs at the top left of the screen to go to the home screen or the FAQ/Help page. (LF)

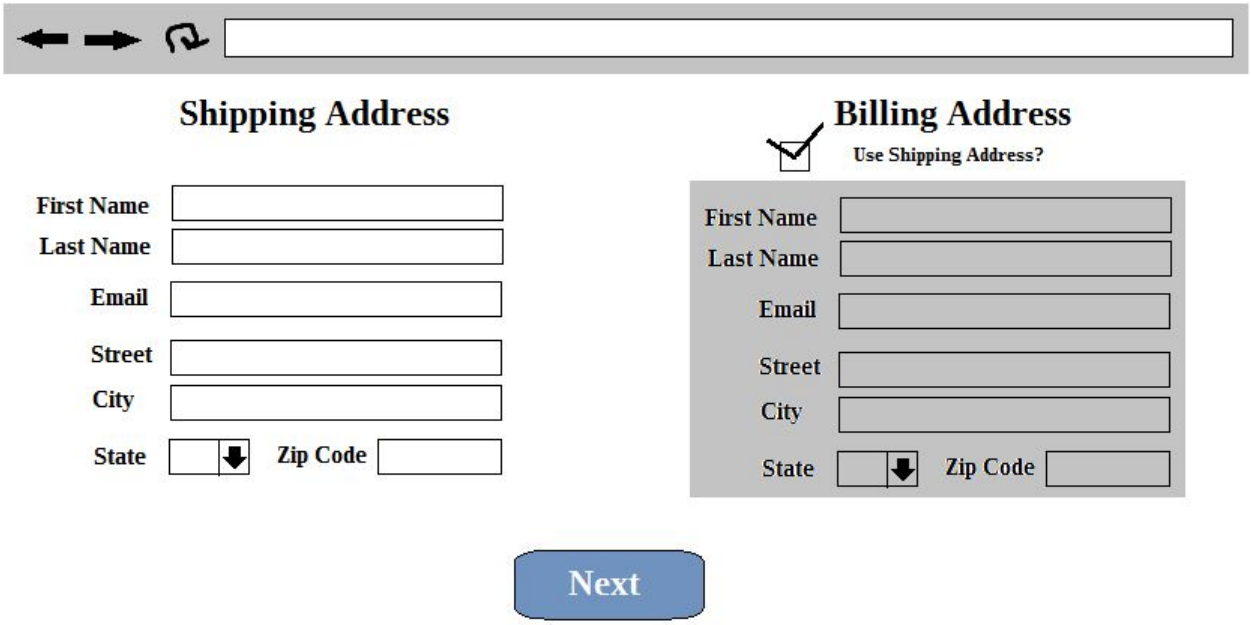


This is the book description page. After getting to the search results page, the user can click on a result to then be taken to this page. The user can view the book picture, the book title, the ISBN, the author, the format, the number in stock, the price, the associated courses, and the description. The placeholder in the top left of the screen is for the book picture. The user can interact with the shopping cart drop-down menu as well as the Home/FAQ tabs to navigate around the website. (LF)



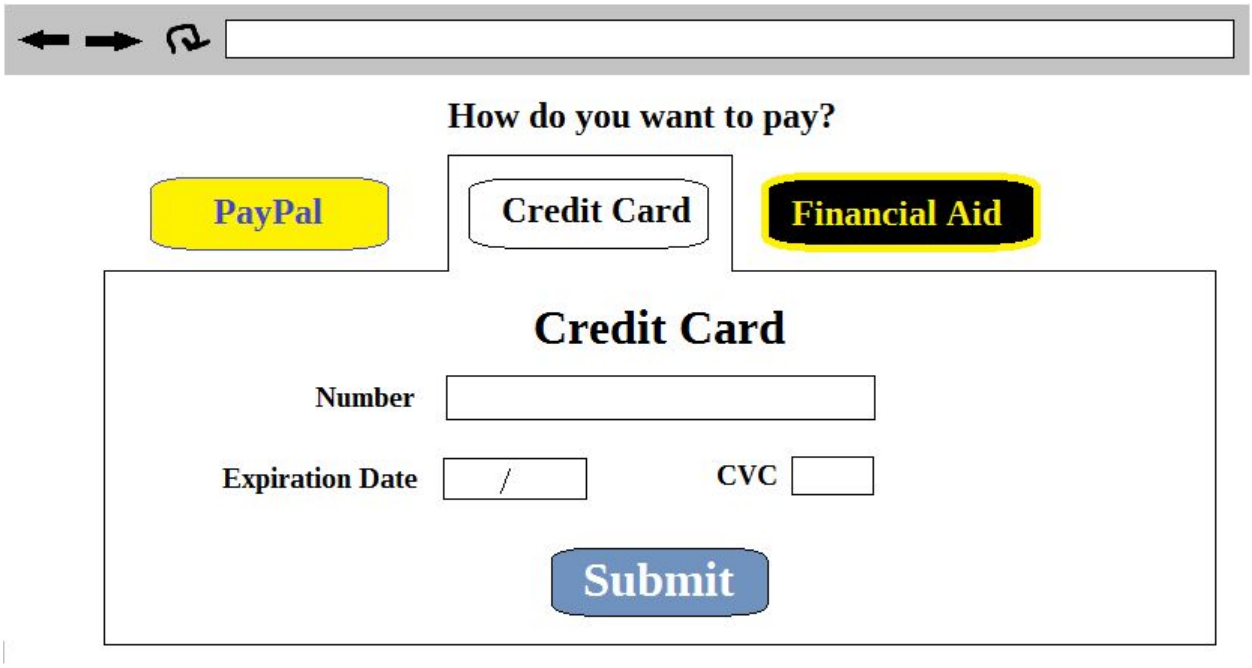
This is the FAQ/Help screen. The user can view the most common questions asked about the online bookstore. The user can interact with the shopping cart drop-down menu as well as the Home/FAQ tabs to navigate around the website. The placeholder is for the KSU logo that UITS will provide for the system to comply with the KSU guidelines.(LF)

This is the Checkout summary page. This allows the user to see what books they've ordered, what type of books (purchased, digital, or rentals) those are, how many of each book they selected and their associated cost. It also shows them how much shipping is (if applicable) and the final amount they will owe the bookstore. The user can interact with the "Pay Now" button to move to the address entry page.

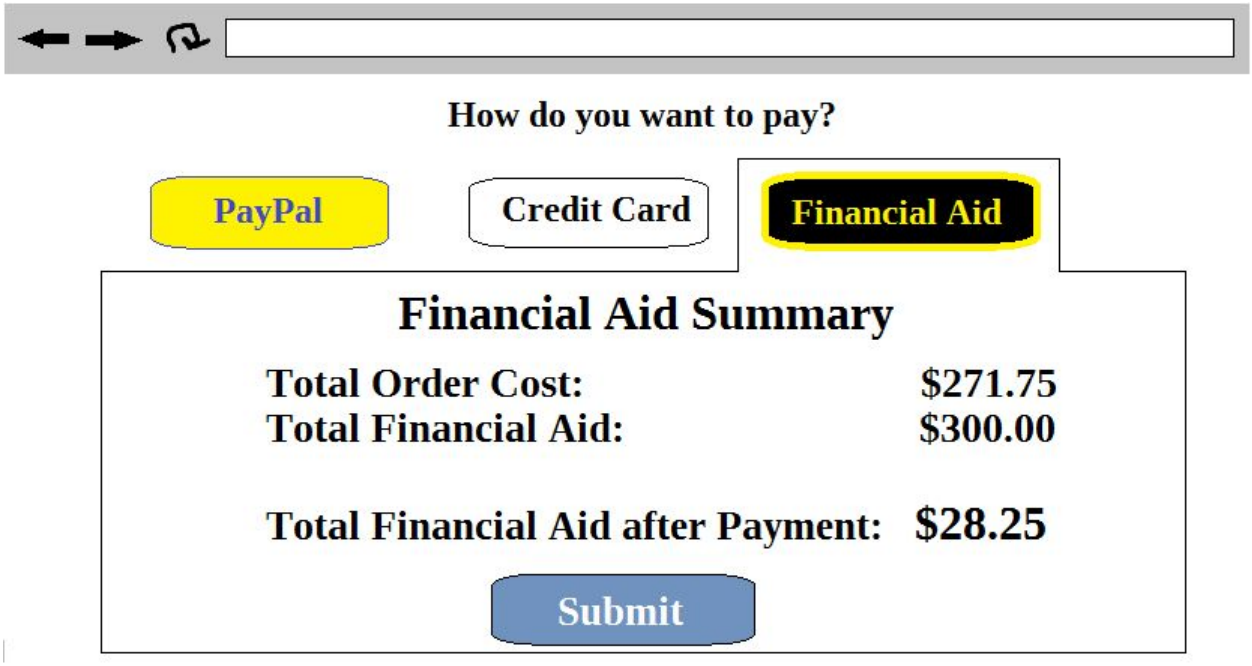


This is the address entry page for the Checkout process. Customers can type their name and email address as well as the street, city, and zip code they want to ship their textbooks to. (they can also interact with the drop-down menu to select the state they want to ship to). They can repeat this process for the matching fields in the billing address section, or they can interact with the “Use Shipping Address?” checkbox to have that information copied over. The user can interact with the “Next” button to move to payment selection page.

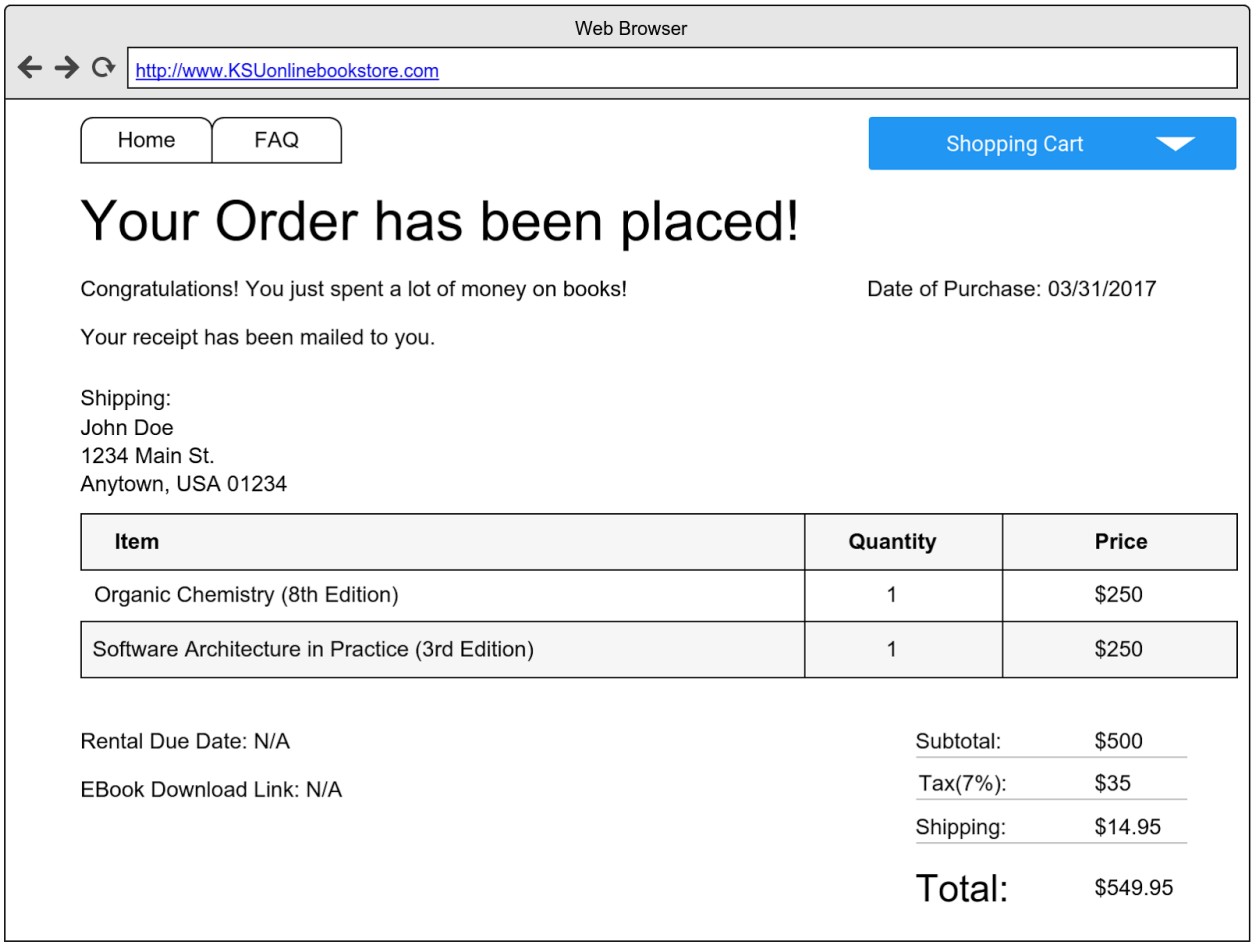
This is the payment selection page for the Checkout process. Users can interact with the “Pay with PayPal” button to be sent to PayPal’s site to finish paying for their books. Once there, they can enter their PayPal username and password and interact with the “Submit” button to authorize payment for their textbook order.



This is the credit card processing page for the Checkout process. Users can also interact with the “Pay by Credit Card” to be sent to the Credit Card entry page. Users can type their credit card number, expiration date, and CVC number, then interact with the “Submit” button to send their information to their bank. If it is successful, then the user is directed to the receipt page.



This is the financial aid summary page for the Checkout process. Users can also interact with the “Pay with Financial Aid” button: if they are not logged in, they will be sent to the login page. If they are logged in, they will sent to the Financial Aid summary page. Students can review the amount of funds they have, as well as how much they would have left if they use their financial aid awards to pay for their textbook order. After they interact with the “Submit” button, if they have enough money to cover the order, they will be taken to the receipt page.



This is the receipt page. The user can see their receipt after they successfully placed an order. The user can also interact with the Home/FAQ tabs to navigate the website and the user can interact with the shopping cart drop-down menu. (LF)

6. Implementation and Unit Testing

6.1. Development Tools and Environment

All development of the online bookstore was done on a remote hhcard server. Each developer used an IDE of their choice (including Notepad++, Microsoft Visual Studio 2017, Sublime Text Editor, vim, emacs, and Eclipse) and uploaded the file to the remote server (using file transfer protocol software such as Filezilla and WinSCP) after each development section. The programming languages used includes PHP, JQuery, HTML, and CSS. PHP was used for all of the page logic and processes. JQuery, HTML, and CSS were used (in conjunction with AJAX library and the Twitter bootstrap) for all page design layout and scripting.

6.2. Unit Testing

All unit testing was done at the time code was written. A minimal number of functions had to be written as most code was written in procedural format and imbedded within the written HTML for the page. The functions that were written (such as checkexpdate() and getGUID()) were tested with several random input values to ensure that they worked correctly. These tests were all done in the browser with user input directly into the browser page. All function source code is located in the appendix of this document. All test cases and procedures used in unit testing are included and detailed in the testing procedures section (section 7.3) starting on page 78.

7. Testing

7.1. Project Integration

All code written by each developer was integrated into the system as soon as the developer saved the file and uploaded it to the remote server. The newly uploaded file would either create a new file/page on the server (if that file did not exists previously), or the file would replace the previous version of that file. In order to prevent unexpected issues with uploading files, a backup of all files on the remote server is downloaded onto a local computer every night.

Informal testing of newly integrated code was done immediately by the developer who wrote the code and uploaded it. This informal testing ensured that the code was integrated appropriately. Formal testing of all integrated code occurred during the testing procedures detailed in section 7.3 starting on page 78.

7.2 Software Testing and Testing Plan (NP)

Overview
This test plan describes the testing approach and overall framework that will drive the testing of

the KSU Online Bookstore site.
The document introduces:
· Test Strategy: rules the test will be based on, including the givens of the project (e.g.: start / end dates, objectives, assumptions); description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
· Execution Strategy: describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
· Test Management: process to handle the logistics of the test and all the events that come up during execution (risk and mitigation). This standard specifies the following test plan outline:

7.2.1. Test Plan Identifier:

1.2.820.163704.1.3.1.9

7.2.2. References.

Applicable Standards:
IEEE Standard for Software Test Documentation, ANSI/IEEE Std. 829-1983.

7.2.3. Introduction.

This test plan is for testing the online bookstore application supporting the following objectives.
(1) To detail the activities required to prepare for and support the test.
(2) To communicate to all responsible teams the tasks which they are to perform and the schedule to be followed in performing the tasks.
(3) To define the sources of information used to prepare the test plan.
(4) To define the test tools and environment needed to conduct the test.
(5) References to lower level test plans
(6) References to related documents such as project authorization, project plan, QA plan, configuration management plan, relevant policies, relevant standards
(7) Summary of the items and features to be tested

7.2.4. Test Items

(1) Search by Title
(2) Search by author
(3) Search by ISBN
(4) Search by Professor
(5) Search by keyword
(6) Sort A-Z
(7) Sort Z-A
(8) Shopping Cart
(9) Checkout
(10) Processing and Handling
(11) Receipt Generation

References: Requirements specification, design specification, bug report to test items.

7.2.5. Software Risks Issues

(1) Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the scheduled start date.

(2) Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve.

(3) Natural disasters

(4) Delayed Testing Due To new Issues

Risk impact analysis table:

Risk	**Probability**	**Impact**
Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the scheduled start date.	High	High
Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve.	Medium	High

Natural disasters	Low	Medium
Delayed Testing Due To new Issues	Medium	High

7.2.6. Features to be Tested

(1) Search Functionality
(2) Sort Functionality
(3) Shopping Cart Functionality
(4) Check out Functionality
(5) Processing and Handling
(6) Receipt Generation Functionality

7.2.7. Features Not to Be Tested

(1) Performance issues like scalability are not tested due to lack of schedule.

7.2.8. Approach

Unit Testing
Individuals will test the components they are working on during development in an ad-hoc manner.

Integration Testing
A detailed set of test cases will be created and executed. These test cases will completely and at the very least cover all requirements detailed in the SRS. A test matrix will verify the coverage. The tests will be performed manually through the user interface against the fully integrated system containing all components.

Test Procedures
Test Procedure contains all the test steps that are to be followed in order to verify and validate each and every functionality of the application using various combinations of test data.

7.2.9. Item Pass/Fail Criteria

(1) The results of each test case will be considered “approved” if the results meet the expected results description in the test case.

7.2.10. Suspension Criteria and Resumption Requirements

(1) Test case execution will be suspended if a critical failure that impedes the ability or value in performing the associated tests is discovered.
(2) Test case execution will be resumed when a developer thinks the problem causing suspension has been fixed. All test cases that exercise the modified portions of the project will be re-executed.

7.2.11. Test Deliverables

(1) A test report will be included in the project deliverables. This report will contain the set of test cases, a history of all formal test executions and a summary of the final state of the test suite.

7.2.12. Testing Tasks

(1) Develop Test Procedures
(2) Execute tests
(3) Report defects
(4) Retest after bug fixes
(5) Complete test report

7.2.13. Environmental Needs

Hardware

Any Operating system computer with a broadband connection

Software

Any Browser

Microsoft Office (Word, Excel) for reports and defect tracking

Security

The test environment doesn't have any specific security requirements.

Tools

Excel or Word document can be used as a tool to document test procedures

7.2.14. Staffing and Training Needs

(1) Anyone who can work on Excel or word.
(2) Anyone who is trained on testing.

7.2.15. Responsibilities

(1) All developers are responsible for the completion of all unit and integration testing tasks.
(2) Project Manager is responsible for approving the Test Plan and the Test Procedures.
(3) Project Manager is also responsible for critiquing the demonstrations and final acceptance of all work products.

7.2.16. Schedule Planning Risks and Contingencies

Risk	**Probability**	**Impact**	**Mitigation Plan**
Testing schedule is tight. If the start of the testing is delayed due to design tasks, the test cannot be extended beyond the scheduled start date.	High	High	The testing team can control the preparation tasks (in advance) and the early communication with involved parties. • Some buffer has been added to the schedule for contingencies, although not as much as best practices advise.

Defects are found at a late stage of the cycle or at a late cycle; defects discovered late are most likely be due to unclear specifications and are time consuming to resolve.	Medium	High	Defect management plan is in place to ensure prompt communication and fixing of issues.
Natural disasters	Low	Medium	Teams and responsibilities have been spread to two different geographic areas. In a catastrophic event in one of the areas, there will resources in the other areas needed to continue (although at a slower pace) the testing activities.

Delayed Testing Due To new Issues	Medium	High	During testing, there is a good chance that some “new” defects may be identified and may become an issue that will take time to resolve. There are defects that can be raised during testing because of unclear document specification. These defects can yield to an issue that will need time to be resolved. If these issues become showstoppers, it will greatly impact on the overall project schedule. If new defects are discovered, the defect management and issue management procedures are in place to immediately provide a resolution.

7.2.17. Approvals

Name	Signature	Date
Robert	Robert Herber	4/24/17
Nicholas	Nicholas Murphy	4/24/17

Lawrence	Lawrence Ferretti	4/24/17

7.2.18. Glossary

SRS:
A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and may include a set of use cases that describe user interactions that the software must provide.

A software bug is an error, flaw, failure or fault in a computer program or system that causes it to produce an incorrect or unexpected result, or to behave in unintended ways.

Deliverable:
Any product that must be delivered to someone other than the author of the product. Examples of deliverables are documentation, code and the system.

Expected result :
A description of the test object's expected status or behaviour after the test steps are completed. Part of the test case.

Failure:
Deviation of the component or system under test from its expected result.

Functional testing:
Testing of the system's functionality and behaviour; the opposite of non-functional testing.

Integration testing:
A test level meant to show that the system's components work with one another. The goal is to find problems in interfaces and communication between components.

Risk:
A factor that could result in future negative consequences. Is usually expressed in terms of impact and likelihood.

Validation:
Tests designed to demonstrate that the developers have built the correct system. Contrast with verification, which means testing that the system has been built correctly. A large number of validation activities take place during acceptance testing.

Verification:
Tests designed to demonstrate that the developers have built the system correctly. Contrast with validation, which means testing that the correct system has been built. A large number of verification activities take place during component testing.

7.3 Testing Procedures (RH)

7.3.1 Homepage Content

Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. None

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	In a browser, type http://hhcard.org/nmurphy1/bookstore/ into the address bar and press enter.	The bookstore homepage is displayed with KSU logo in the upper middle, searching options in the middle, Home and FAQ tabs in the top left corner, and the shopping cart in the top right.		

7.3.2 Search by Title
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)
2. "Title" is selected in the search by dropbox.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Ensure that the test prerequisite that "Title" is selected in the search dropdown box is met and enter "eng" into the search bar and click search.	5 results should appear all with "eng" somewhere in the title.		
2	Enter "asdf" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
3	Click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
4	Enter "1234" into the search bar and click search.	"No Results Found" should be displayed.		
5	Enter "soft" into the search again bar in the upper left corner and click search.	2 results with "soft" in the title should appear.		
6	Enter "SOFT" into the search again bar in the upper left corner and	2 results with "soft" in the title should appear.		

	click search.			
7	Enter “!@#$” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
8	Enter “e” into the search again bar in the upper left corner and click search.	43 results with “e” in the title should appear.		
9	Enter “E” into the search again bar in the upper left corner and click search.	43 results with “e” in the title should appear.		
10	Enter “Lartigue” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
11	Enter “978-1449691998” into search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
12	Enter “Baase” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
13	Enter “Engineering Fluid Mechanics” into the search again bar in the upper left corner and click search.	2 results “Engineering Fluid Mechanics” and “Engineering Fluid Mechanics Looseleaf” should appear.		

7.3.3 Search by Author
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)
2. "Author" is selected in the search by dropbox.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Ensure that the test prerequisite that "Author" is selected in the search dropdown box is met and enter "eng" into the search bar and click search.	"No Results Found" should be displayed.		
2	Enter "asdf" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
3	Click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
4	Enter "1234" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
5	Enter "soft" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
6	Enter "SOFT" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
7	Enter "!@#$" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
8	Enter "e" into the search again bar in the upper left corner and click search.	38 results with "e" in the author's name should appear.		

9	Enter “E” into the search again bar in the upper left corner and click search.	38 results with “e” in the author’s name should appear.		
10	Enter “Lartigue” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
11	Enter “978-1449691998” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
12	Enter “Baase” into the search again bar in the upper left corner and click search.	1 result with “Baase” in the author’s name will appear.		
13	Enter “Engineering Fluid Mechanics” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
14	Enter “Kai” into the search again bar in the upper left corner and click search.	1 result with “Kai” in the author’s name will appear.		
15	Enter “Dan” into the search again bar in the upper left corner and click search.	2 results with “Dan” in the author’s name will appear.		

7.3.4 Search by Description
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)
2. “Description” is selected in the search by dropbox.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Ensure that the test prerequisite that “Description” is selected in the search dropdown box is met and enter “eng” into the search bar and click search.	7 results with “eng” in the description will appear.		
2	Enter “asdf” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
3	Click search with nothing in the search again bar in the upper left corner.	“No Results Found” should be displayed.		
4	Enter “1234” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
5	Enter “soft” into the search again bar in the upper left corner and click search.	2 results with “soft” in the description will appear.		
6	Enter “SOFT” into the search again bar in the upper left corner and click search.	2 results with “SOFT” in the description will appear.		
7	Enter “!@#$” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
8	Enter “e” into the search again bar in the upper left corner and click search.	44 results with “e” in the description will appear.		

9	Enter “E” into the search again bar in the upper left corner and click search.	44 results with “E” in the description will appear.		
10	Enter “Lartigue” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
11	Enter “978-1449691998” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
12	Enter “Baase” into the search again bar in the upper left corner and click search.	1 result with “Baase” in the description will appear.		
13	Enter “Engineering Fluid Mechanics” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
14	Enter “!” into the search again bar in the upper left corner and click search.	1 result with “!” in the description will appear.		
15	Enter “Dan” into the search again bar in the upper left corner and click search.	1 results with “Dan” in the description will appear.		

7.3.5 Search by Course
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)

2. "Course" is selected in the search by dropbox.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Ensure that the test prerequisite that "Course" is selected in the search dropdown box is met and enter "eng" into the search bar and click search.	8 results with "eng" in the course title will appear.		
2	Enter "asdf" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
3	Click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
4	Enter "1234" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
5	Enter "soft" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
6	Enter "SOFT" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
7	Enter "!@#$" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
8	Enter "e" into the search again bar in the upper left corner and click search.	24 results with "e" in the course title will appear.		

9	Enter “E” into the search again bar in the upper left corner and click search.	24 results with “E” in the course title will appear.		
10	Enter “Lartigue” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
11	Enter “978-1449691998” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
12	Enter “Baase” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
13	Enter “Engineering Fluid Mechanics” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
14	Enter “CS” into the search again bar in the upper left corner and click search.	11 results with “CS” in the title will appear.		
15	Enter “CSE 3642” into the search again bar in the upper left corner and click search.	1 results with “CSE 3642” as the course title will appear.		
16	Enter “CSE3642” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
17	Enter “3642” into the search again bar in the upper left corner and click search.	1 results with “3642” in the course title will appear.		

7.3.6 Search by Professor

Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)
2. "Professor" is selected in the search by dropbox.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Ensure that the test prerequisite that "Professor" is selected in the search dropdown box is met and enter "eng" into the search bar and click search.	2 results with "eng" in the Professor's name will appear.		
2	Enter "asdf" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
3	Click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
4	Enter "1234" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
5	Enter "soft" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
6	Enter "SOFT" into the search again bar in the upper left corner and	"No Results Found" should be displayed.		

	click search.			
7	Enter “!@#$” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
8	Enter “e” into the search again bar in the upper left corner and click search.	32 results with “e” in the professor’s name should appear.		
9	Enter “E” into the search again bar in the upper left corner and click search.	32 results with “e” in the professor’s name should appear.		
10	Enter “Lartigue” into the search again bar in the upper left corner and click search.	1 result with “Lartigue” in the professor’s name will appear.		
11	Enter “978-1449691998” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
12	Enter “Baase” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
13	Enter “Engineering Fluid Mechanics” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
14	Enter “Kai” into the search again bar in the upper left corner and click search.	2 result with “Kai” in the professor’s name will appear.		
15	Enter “Dan” into the search again bar in the upper left corner and click search.	2 results with “Dan” in the professor’s name will appear.		

7.3.7 Search by ISBN
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)
2. "ISBN" is selected in the search by dropbox.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Ensure that the test prerequisite that "ISBN" is selected in the search dropdown box is met and enter "eng" into the search bar and click search.	"No Results Found" should be displayed.		
2	Enter "asdf" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
3	Click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
4	Enter "1234" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
5	Enter "soft" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		

6	Enter “SOFT” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
7	Enter “!@#$” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
8	Enter “e” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
9	Enter “E” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
10	Enter “Lartigue” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
11	Enter “978-1449691998” into the search again bar in the upper left corner and click search.	1 result with “978-1449691998” in the ISBN will appear.		
12	Enter “Baase” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
13	Enter “Engineering Fluid Mechanics” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
14	Enter “-” into the search again bar in the upper left corner and click search.	44 results with “-” in the ISBN will appear.		
15	Enter “978” into the	44 results with “978” in the		

	search again bar in the upper left corner and click search.	ISBN will appear.		
16	Enter “978-” into the search again bar in the upper left corner and click search.	44 results with “978-” in the ISBN will appear.		
17	Enter “1449691998” into the search again bar in the upper left corner and click search.	1 result with “1449691998” in the ISBN will appear.		
18	Enter “-1449691998” into the search again bar in the upper left corner and click search.	1 result with “-1449691998” in the ISBN will appear.		

7.3.8 Sort by Title A - Z
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	5 results will appear.		
2	At the results page, select “A-Z” in the sort by drop down menu in the top right of the	5 results will appear with “Basics of Engineering Economy 2nd Edition” listed first and “Essentials of Software		

	screen.	Engineering 3rd Edition" listed last.		
3	With "Author" selected in the search by menu, enter "e" into the search again bar in the upper left corner and click search.	38 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by author.	
4	At the results page, select "A-Z" in the sort by drop down menu in the top right of the screen.	38 results will appear with "A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition" listed first and "Writer's Reference" listed last.		
5	With "Title" selected in the search by menu, click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
6	At the results page, select "A-Z" in the sort by drop down menu in the top right of the screen.	"No Results Found" should be displayed.		
7	With "Title" selected in the search by menu, enter "1234" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
8	At the results page, select "A-Z" in the sort by drop down menu in the top right of the screen.	"No Results Found" should be displayed.		
9	With "Professor" selected in the search by menu, enter "Zeynep" into the	2 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by	

	search again bar in the upper left corner and click search.		professor.	
10	At the results page, select “A-Z” in the sort by drop down menu in the top right of the screen.	2 results will appear with “Basics of Engineering Economy 2nd Edition” at the top and and “Essentials of Economics 3rd Edition” at the bottom.		
11	With “Course” selected in the search by menu, enter “CSE” into the search again bar in the upper left corner and click search.	6 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by course.	
12	At the results page, select “A-Z” in the sort by drop down menu in the top right of the screen.	6 results will appear with “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” at the top and and “Vistas, 4th Edition, Supersite Plus Code (supersite & WebSAM & vtext) - Code Only” at the bottom.		
13	With “ISBN” selected in the search by menu, enter “978” into the search again bar in the upper left corner and click search.	44 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by ISBN.	
14	At the results page, select “A-Z” in the sort by drop down menu in the top right of the screen.	44 results will appear with “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” at the top and and “Writer’s Reference” at the bottom.		
15	With “Description” selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.	44 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by description.	

16	At the results page, select “A-Z” in the sort by drop down menu in the top right of the screen.	44 results will appear with “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” at the top and and “Writer’s Reference” at the bottom.		

7.3.9 Sort by Title Z - A
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Each step starts at Homepage Content (test 6.2.1)

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	5 results will appear.		
2	At the results page, select “Z-A” in the sort by drop down menu in the top right of the screen.	5 results will appear with “Essentials of Software Engineering 3rd Edition” listed first and “Basics of Engineering Economy 2nd Edition” listed last.		
3	With “Author” selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.	38 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by author.	
4	At the results page, select “Z-A” in the sort by drop down menu in the top right of the	38 results will appear with “Writer’s Reference” listed first and “A Gift of Fire: Social, Legal, and Ethical Issues for		

	screen.	Computing Technology 4th Edition" listed last.		
5	With "Title" selected in the search by menu, click search with nothing in the search again bar in the upper left corner.	"No Results Found" should be displayed.		
6	At the results page, select "Z-A" in the sort by drop down menu in the top right of the screen.	"No Results Found" should be displayed.		
7	With "Title" selected in the search by menu, enter "1234" into the search again bar in the upper left corner and click search.	"No Results Found" should be displayed.		
8	At the results page, select "Z-A" in the sort by drop down menu in the top right of the screen.	"No Results Found" should be displayed.		
9	With "Professor" selected in the search by menu, enter "Zeynep" into the search again bar in the upper left corner and click search.	2 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by professor.	
10	At the results page, select "Z-A" in the sort by drop down menu in the top right of the screen.	2 results will appear with "Essentials of Economics 3rd Edition" at the top and and "Basics of Engineering Economy 2nd Edition" at the bottom.		
11	With "Course" selected in the search by menu, enter "CSE" into the	6 results will appear.	This ensures that sort still correctly sorts by title when items are	

	search again bar in the upper left corner and click search.		searched for by course.	
12	At the results page, select “Z-A” in the sort by drop down menu in the top right of the screen.	6 results will appear with “Vistas, 4th Edition, Supersite Plus Code (supersite & WebSAM & vtext) - Code Only” at the top and and “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” at the bottom.		
13	With “ISBN” selected in the search by menu, enter “978” into the search again bar in the upper left corner and click search.	44 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by ISBN.	
14	At the results page, select “Z-A” in the sort by drop down menu in the top right of the screen.	44 results will appear with “Writer’s Reference” at the top and and “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” at the bottom.		
15	With “Description” selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.	44 results will appear.	This ensures that sort still correctly sorts by title when items are searched for by description.	
16	At the results page, select “Z-A” in the sort by drop down menu in the top right of the screen.	44 results will appear with “Writer’s Reference” at the top and and “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” at the bottom.		

7.3.10 Sort by Price, Low to High
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	5 results will appear		
2	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	5 results will appear with “Essentials of Software Engineering 3rd Edition” listed first and “Engineering Fluid Mechanics” listed last.		
3	With “Author” selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.	38 results will appear		
4	At the results page, select “Price, Low to	38 results will appear with “Latin for the New Millennium” listed		

	High” in the sort by drop down menu in the top right of the screen.	first and “Chemistry” listed last.		
5	With “Title” selected in the search by menu, click search with nothing in the search again bar in the upper left corner.	“No Results Found” should be displayed.		
6	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	“No Results Found” should be displayed.		
7	With “Title” selected in the search by menu, enter “1234” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		
8	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	“No Results Found” should be displayed.		
9	With “Professor” selected in the search by menu, enter “Zeynep” into the search again bar in the upper left corner and click search.	2 results will appear		
10	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	2 results will appear with “Basics of Engineering Economy 2nd Edition” at the top and and “Essentials of Economics 3rd Edition” at the bottom.		
11	With “Course” selected in the search by menu, enter “CSE” into the	6 results will appear		

	search again bar in the upper left corner and click search.			
12	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	6 results will appear with “Fundamentals of Computer Programming with C#” at the top and and “Java Illuminated 4th Edition” at the bottom.		
13	With “ISBN” selected in the search by menu, enter “978” into the search again bar in the upper left corner and click search.	44 results will appear		
14	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	44 results will appear with “Fundamentals of Computer Programming with C#” at the top and and “Chemistry” at the bottom.		
15	With “Description” selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.	44 results will appear.		
16	At the results page, select “Price, Low to High” in the sort by drop down menu in the top right of the screen.	44 results will appear with “Fundamentals of Computer Programming with C#” at the top and and “Chemistry” at the bottom.		

7.3.11 Sort by Price, High to Low
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	5 results will appear		
2	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	5 results will appear with “Engineering Fluid Mechanics” listed first and “Essentials of Software Engineering 3rd Edition” listed last.		
3	With “Author” selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.	38 results will appear		
4	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	38 results will appear with “Chemistry” listed first and “Latin for the New Millennium” listed last.		
5	With “Title” selected in the search by menu, click search with nothing in the search again bar in the upper left corner.	“No Results Found” should be displayed.		
6	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	“No Results Found” should be displayed.		
7	With “Title” selected in the search by menu, enter “1234” into the search again bar in the upper left corner and click search.	“No Results Found” should be displayed.		

8	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	“No Results Found” should be displayed.		
9	With “Professor” selected in the search by menu, enter “Zeynep” into the search again bar in the upper left corner and click search.	2 results will appear		
10	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	2 results will appear with “Essentials of Economics 3rd Edition” at the top and and “Basics of Engineering Economy 2nd Edition” at the bottom.		
11	With “Course” selected in the search by menu, enter “CSE” into the search again bar in the upper left corner and click search.	6 results will appear		
12	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	6 results will appear with “Java Illuminated 4th Edition” at the top and and “Fundamentals of Computer Programming with C#” at the bottom.		
13	With “ISBN” selected in the search by menu, enter “978” into the search again bar in the upper left corner and click search.	44 results will appear		
14	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	44 results will appear with “Chemistry” at the top and and “Fundamentals of Computer Programming with C#” at the bottom.		
15	With “Description”	44 results will appear		

	selected in the search by menu, enter “e” into the search again bar in the upper left corner and click search.			
16	At the results page, select “Price, High to Low” in the sort by drop down menu in the top right of the screen.	44 results will appear with “Chemistry” at the top and and “Fundamentals of Computer Programming with C#” at the bottom.		

7.3.12 View Details
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	The search results page will appear.		
2	At the results page, click on the underlined title “Basics of Engineering Economy 2nd Edition.”	The view details page with the details on “Basics of Engineering Economy 2nd Edition” will be displayed. It has a new price of $160.00, used of $120, rental $82.00, and ebook at $131.80. Its ISBN is 978-0073376356 and it is required for Dr. Zeynep Arzu Kelani's ECON 2107, Section 3		

3	Click the “Home” tab in the top left corner.	Homepage content will appear.		
4	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	The search results page will appear.		
5	At the results page, click picture to the left of the underlined title “Basics of Engineering Economy 2nd Edition.”	The view details page with the details on “Basics of Engineering Economy 2nd Edition” will be displayed. It has a new price of $160.00, used of $120, rental $82.00, and ebook at $131.80. Its ISBN is 978-0073376356 and it is required for Dr. Zeynep Arzu Kelani's ECON 2107, Section 3		
6	Click the “Home” tab in the top left corner.	Homepage content will appear.		
7	With “Professor” selected in the search by menu, enter “Zeynep” into the search bar and click search.	The search results page will appear.		
8	At the results page, click on the underlined title of the second result “Essentials of Economics 3rd Edition.”	The view details page with the details on “Essentials of Economics 3rd Edition” will be displayed. It has a new price of $213.15, used at $159.85, rental at $109.00, and ebook at $175.40. Its ISBN is 978-0073511450 and it is required for Dr. Zeynep Arzu Kelani's ECON 2107, Section 3 class.		
9	Click the “Home” tab in the top left corner.	Homepage content will appear.		
10	With “Professor” selected in the search by menu, enter	The search results page will appear.		

	"Zeynep" into the search bar and click search.			
11	At the results page, click on the picture to the left of the underlined title of the second result "Essentials of Economics 3rd Edition."	The view details page with the details on "Essentials of Economics 3rd Edition" will be displayed. It has a new price of $213.15, used at $159.85, rental at $109.00, and ebook at $175.40. Its ISBN is 978-0073511450 and it is required for Dr. Zeynep Arzu Kelani's ECON 2107, Section 3 class.		
12	Click the "Home" tab in the top left corner.	Homepage content will appear.		
13	With "Course" selected in the search by menu, enter "CSE" into the search bar and click search.	The search results page will appear.		
14	At the results page, click on the underlined title of the first result "A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition."	The view details page with the details on "A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition." will be displayed. With new price at $93.57, used at $85.89, rental at $28.19, and ebook at $66.99. Its ISBN is 978-0132492676. It is required for Dr. Manijeh Chavoshi's CSE 3642, section 2 class.		

7.3.13 Add and Remove Item(s) to and from Cart
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content (test 6.2.1)

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	The search results page will appear.		
2	At the results page, click on the underlined title “Basics of Engineering Economy 2nd Edition.”	The view details page for the item will appear.		
3	From the view details page, click on the radial button to the left of “New” and then click “Add to Cart.”	The “Your Shopping Cart” page will appear with the item “Basics of Engineering Economy 2nd Edition” with a quantity of 1, a $160.00 Per Unit Cost, and a total cost of $160.00. The shopping cart drop down menu in the top right corner of the screen will now display a “1” in parentheses.		
4	Remove the item from the cart by clicking the down arrow in the quantity box until quantity is 0 and click “update cart”	The item will be removed from the cart. The “Your Shopping Cart” page will refresh and display “Your Shopping Cart is Empty” and the shopping cart menu in the top right will display a (0) in parentheses.		
5	Click the “Home” tab in the top left corner.	Homepage content will appear.		
6	With “ISBN” selected in the search by menu, enter “978-0547152813” into the search bar and click search.	The search results page will appear.	Attempting to add a book to the cart that is out of stock.	
7	At the results page, click picture to the left of the underlined title “General, Organic, and	The view details page for the item will appear.		

	Biological Chemistry."			
8	At the view details page, attempt to add a used version of the book to the cart.	There will not be a radial button next to "Used" because there are no used copies currently in stock. Therefore, a used copy of this book cannot be added to the cart.		
9	Click the "Home" tab in the top left corner.	Homepage content will appear.		
10	With "Professor" selected in the search by menu, enter "Zeynep" into the search bar and click search.	The search results page will appear.		
11	At the results page, click on the underlined title of the second result "Essentials of Economics 3rd Edition."	The view details page for the item will appear.		
12	At the view details page, click on the radial button next to "eBook" and then click "Add to Cart."	The "Your Shopping Cart" page will appear with 1 item in the cart.		
13	At the "Your shopping Cart" page, click on quantity box next to the book and enter "2" then click update cart.	The quantity displayed in the quantity box will now show "2" and the shopping cart menu in the top right corner will now display a "2" in parentheses.		
14	Remove both items from the cart by clicking the down arrow in the quantity box until quantity is 0 and click "update cart"	The items will be removed from the cart. The "Your Shopping Cart" page will refresh and display "Your Shopping Cart is Empty" and the shopping cart menu in the top right will display a (0) in parentheses.		

15	Click the “Home” tab in the top left corner.	Homepage content will appear.		
16	With “Course” selected in the search by menu, enter “CSE” into the search bar and click search.	The search results page will appear.		
17	At the results page, click on the underlined title of the first result “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition.”	The view details page for the item will appear.		
18	At the view details page, click on the radial button next to “Rental” then click “Add to Cart.”	The item will be added to the cart.		
19	At the “Your Shopping Cart” page, in the upper left hand corner of the page, ensure that “title” is selected and type “e” into the search bar and click “search.”	The search results page will appear.		
20	At the results page, click on “Accounting: Tools for Business Decision Making.”	The view details page will appear.		
21	At the details page, click on the button next to “new” and click “Add to Cart.”	The “Your Shopping Cart” page will appear with both “A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition” and “Accounting: Tools for Business Decision Making” in it. Both have a quantity of 1. The shopping cart menu in the upper right hand corner will now display “2.”		

22	Remove both items from the cart by clicking the down arrow in the quantity box next to both items until quantity is 0 and click “update cart”	The items will be removed from the cart. The “Your Shopping Cart” page will refresh and display “Your Shopping Cart is Empty” and the shopping cart menu in the top right will display a (0) in parentheses.		

7.3.14 Input Shipping and Billing Information
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	The search results page will appear.		
2	At the results page, click on the underlined title “Basics of Engineering Economy 2nd Edition.”	The view details page will appear.		
3	From the view details page, click on the radial button to the left of “New” and then click “Add to Cart.”	The item will be added to the cart.		
4	From the “Your	The order summary page will		

	Shopping Cart" page, click "Check Out."	appear.		
5	From the Order Summary page, click "Pay Now."	The "Step 1:Shipping and Billing Address" page will appear.		
6	With no input in any input box, click "next."	The "Shipping and Billing Options" page will not change and all empty input boxes will be red.		
7	Under shipping address, type "Bob" in the "First Name" text box and click "next."	The "Shipping and Billing Options" page will not change and all empty input boxes will be red. Any input box with input will not be red.		
8	Under shipping address, type "Bob" in the first name box, "Dillion" in the Last Name box, "bob@gmail.com" in the email box, "111 fake street" in the street box, select "GA" from the state box, type "30179" in the zip code box, ensure that the "Use Shipping Address?" check box is NOT checked, and click "next" with all billing information empty.	The "Shipping and Billing Options" page will not change and all empty input boxes will be red. Any input box with input will not be red.		
9	Type "abc" in the "Zip Code" box.	The typed letter will be instantly deleted and the zip code box will remain empty.		
10	Under shipping address, type "Bob" in the first name box, "Dillion" in the Last Name box, "bob.bob" in the email box, "111 fake street" in the street	The "Shipping and Billing Options" page will not change and the email box will be red. Any input box with input will not be red.		

	box, select “GA” from the state box, type “30179” in the zip code box, ensure that the “Use Shipping Address?” check box is checked, and click “next.”			
11	Under shipping address, type “Bob” in the first name box, “Dillion” in the Last Name box, “bob@gmail.com” in the email box, “111 fake street” in the street box, select “GA” from the state box, type “123” in the zip code box, ensure that the “Use Shipping Address?” check box is checked, and click “next.”	The “Shipping and Billing Options” page will not change and the zip code box will be red. Any input box with input will not be red.		
12	Under shipping address, type “Bob” in the first name box, “Dillion” in the Last Name box, “bob@gmail.com” in the email box, “111 fake street” in the street box, select “GA” from the state box, type “30179” in the zip code box, ensure that the “Use Shipping Address?” check box is checked, and click “next.”	The “Step 2: Payment Options” page will appear.		
13	Click the back button in the browser.	The “Step 1: Shipping and Billing Options” will appear.		
14	Under shipping	The “Step 2: Payment Options”		

	address, type “Bob” in the first name box, “Dillion” in the Last Name box, “bob@gmail.com” in the email box, “111 fake street” in the street box, select “GA” from the state box, type “30179” in the zip code box, ensure that the “Use Shipping Address?” check box is NOT checked. Under billing address, type “David” in the first name box, “Bowe” in the Last Name box, “david@gmail.com” in the email box, “112 fake street” in the street box, select “GA” from the state box, type “30179” in the zip code box, then click “Next.”	page will appear.		

7.3.15 Payment
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content

Test steps:

#	Required Actions	Expected Results	Comments	
1	With “Title” selected in the search by menu, enter “eng” into the search bar and click search.	The search results page will appear.		

2	At the results page, click on the underlined title “Basics of Engineering Economy 2nd Edition.”	The view details page will appear.		
3	From the view details page, click on the radial button to the left of “New” and then click “Add to Cart.”	The “Your Shopping Cart” page will appear with the item in the cart.		
4	From the “Your Shopping Cart” page, click “Check Out.”	The order summary page will appear.		
5	From the Order Summary page, click “Pay Now.”	The “Step1: Shipping and Billing Information” page will appear.		
6	Under shipping address, type “Bob” in the first name box, “Dillion” in the Last Name box, “bob@gmail.com” in the email box, “111 fake street” in the street box, select “GA” from the state box, type “30179” in the zip code box, ensure that the “Use Shipping Address?” check box is checked, and click “next.”	The “Step2: Payment options” page will appear.		
7	Click on the “PayPal” button and with no input in any input box, click “pay.”	The “Step 2: Payment Options” page will still be displayed and all empty input boxes will be red.		
8	Click on the “Credit Card” button and with no input in any input box, click “pay.”	The “Step 2: Payment Options” page will still be displayed and all empty input boxes will be red.		
9	Click on the “Financial	The “Step 2: Payment Options”		

	Aid" button and with no input in any input box, click "Login."	page will still be displayed and all empty input boxes will be red. "Invalid Username/Password" error will appear.		
10	Click on the "PayPal" button and input an invalid PayPal Username and password and click "pay."	The "Step 2: Payment Options" page will still be displayed and "Invalid Username/Password" will be displayed.		
11	Click on the "PayPal" button and input a correct PayPal Username and password and click "pay."	The information is sent to PayPal for processing and is accepted. The confirmation page displaying the receipt is now displaying. The shopping cart is now empty.		
12	Click on the "home" tab in the top left corner.	Homepage content will appear.		
13	Repeat steps 1 - 6 to add an item to the cart.	The "Step2: Payment options" page will appear.		
14	Click on the "Credit Card" button and attempt to enter "abc" into the "Card Number" input box.	The typed letter will be instantly deleted and the card number box will remain empty.		
15	Click on the "Credit Card" button and attempt to enter a date previous to today's date into the "Exp. Date" input box.	The date will automatically jump forward to a date after today's date making it impossible for a user to provide an already passed expiration date.		
16	Click on the "Credit Card" button and attempt to enter "abc" into the "CVC" input box.	The typed letter will be instantly deleted and the CVC box will remain empty.		
17	Click on the "Credit Card" button and type "Bob Dole" in the	The "Step 2: Payment Options" page will still be displayed and Card Number box will be red		

	“Name on Card” box, “123” in the “Card Number” box, “4/21” in the “Exp. Date” menu, and “123” in the “CVC” box, and click “Pay.”	with a tooltip saying the “card number must be 16 numbers long” will be displayed.		
18	Click on the “Credit Card” button and type “Bob Dole” in the “Name on Card” box, “1234567890123456” in the “Card Number” box, “4/21” in the “Exp. Date” menu, and “1” in the “CVC” box, and click “Pay.”	The “Step 2: Payment Options” page will still be displayed and CSV box will be red with a tooltip saying the “CVC must be 3 numbers long” will be displayed.		
19	Click on the “Credit Card” button and type “Bob Dole” in the “Name on Card” box, “1234567890123456” in the “Card Number” box, “4/21” in the “Exp. Date” menu, and “123” in the “CVC” box, and click “Pay.” Any valid credit card information will work.	The information is sent to the credit card company for processing and is accepted. The confirmation page displaying the receipt is now displaying. The shopping cart is now empty.		
20	Click on the “home” tab in the top left corner.	Homepage content will appear.		
21	Repeat steps 1 - 6 to add an item to the cart.	The “Step2: Payment options” page will appear.		
22	Click on the “Financial Aid” button and input a correct Username and password (for testing, use “jwins22” and “123456789”) and click “Login.” At the “Financial Aid Summary” page, ensure there is NOT	The “Financial Aid Summary” page will still be displayed and will be showing a negative balance after the purchase. The “Pay” button will not be visible due to the negative balance after the order. Therefore, the student cannot use financial aid for this order.		

	ENOUGH funds to pay for the order (add more items to the cart if needed), and attempt to click “Pay.”			
23	Click on the “Financial Aid” button and input an incorrect Username and password (for testing, use “abc” and “123”) and click “Login.”	The “Step 2: Payment Options” page will still be displayed and “Invalid Username/Password” will be displayed.		
24	Click on the “Financial Aid” button and input a correct Username and password (for testing, use “jwins22” and “123456789”) and click “Login.” At the “Financial Aid Summary” page, ensure there is ENOUGH funds to pay for the order, click “Pay.”	The information is validated with the student data file for processing and is accepted. The confirmation page displaying the receipt is now displaying. The shopping cart is now empty. The cost of the items have been deducted from the student’s financial aid balance.		

7.3.16 Receipt
Created 4/19/17
by Robert Herber

Test date: 4/21/17
Results: Pass
Tested by: Robert Herber

Test prerequisites:

1. Homepage Content.

Test steps:

#	Required Actions	Expected Results	Comments	✓
1	Search “e” by title.	The search results page will appear.		
2	At the results page, click on “A Gift of Fire:	The view details page will appear.		

	Social, Legal, and Ethical Issues for Computing Technology 4th Edition."			
3	At the view details page, click the radial button next to the new version of the book and click "add to the cart."	The "Your shopping cart" page will appear with a new version of the item in the cart.		
4	Repeat steps 1 through 3 three more times to add 1 rental, 1 used, and 1 eBook version of the book to the cart.	The "Your shopping cart" page will appear with a new, used, rental, and eBook version of the item in the cart.		
5	At the shopping cart page, click checkout.	The "Order Summary" page will appear.		
6	At the order summary page, click pay now.	The "Step 1: Shipping and Billing Options" page will appear.		
7	Under shipping address, type "Bob" in the first name box, "Dillion" in the Last Name box, "bob@gmail.com" in the email box, "111 fake street" in the street box, select "GA" from the state box, type "30179" in the zip code box, ensure that the "Use Shipping Address?" check box is checked, and click "next."	The "Step 2: Payment Options" page will appear.		
8	Click on the "Credit Card" button and type "Bob Dole" in the "Name on Card" box, "1234567890123456" in the "Card Number" box, "4/21" in the "Exp.	The order confirmation page with the receipt is now being displayed. The 4 items with the quantity and price are now being displayed along with the subtotal, tax, shipping fee, and total cost. The rental due date is		

	Date" menu, and "123" in the "CVC" box, and click "Pay."	14/08/2017. There is an ebook download link that is clickable. A copy of the receipt has been emailed to the proved email address. The shopping cart is not empty and the shopping cart drop down menu now shows a "0" in parentheses.		
9	At the order confirmation page, click the ebook download link.	The "Your Ebook has been downloaded!" page is being displayed along with the GUID for the ebook.		
10	Click the "Home" tab in the top left corner.	Homepage content will appear		
11	Search "e" by title.	The search results page will appear.		
12	At the results page, click on "A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology 4th Edition."	The view details page will appear.		
13	At the view details page, click the radial button next to the eBook version of the book and click "add to the cart."	The "your shopping cart page" will appear with the ebook in it.		
14	At the shopping cart page, click checkout.	The Order summary page will appear.		
15	At the order summary page, click pay now.	The "Step 1: Shipping and Billing Options" page will appear.		
16	Under shipping address, type "Bob" in the first name box, "Dillion" in the Last Name box, "bob@gmail.com" in the email box, "111	The "Step 2: Payment Options" page will appear.		

	fake street" in the street box, select "GA" from the state box, type "30179" in the zip code box, ensure that the "Use Shipping Address?" check box is checked, and click "next."			
17	Click on the "Credit Card" button and type "Bob Dole" in the "Name on Card" box, "1234567890123456" in the "Card Number" box, "4/21" in the "Exp. Date" menu, and "123" in the "CVC" box, and click "Pay."	The order confirmation page with the receipt is now being displayed. The 1 item with the quantity and price is now being displayed along with the subtotal, tax, and total cost. The rental due date is N/A. The ebook download available and clickable. A copy of the receipt has been emailed to the proved email address. Ensure that there is no shipping fee due to only being ebooks in the order. The shopping cart is not empty and the shopping cart drop down menu now shows a "0" in parentheses.		

8. Project Retrospective

8.1. Project Results

The project document was completed and an application was built using the requirements and specifications given by the stakeholders. The application has been validated and verified per the specifications so the project can be deemed a success.

8.1.1 Successes

Recording the entire initial meeting with the stakeholders via meeting minutes set up our project for success. We were able to go back and constantly check our documentation to the stakeholder requests and it allowed us to stay within the scope of the project. Creating the stakeholder profiles and the written use cases during the requirements gathering sped up the rest of the project development.

8.1.2 Failures

Having to review sections and modify them slowed the project development time. With the waterfall method one cannot progress in the development before the previous stage is completed so if we needed to change an item in a previous step, we would have to change everything that had progressed since that change.

8.1.3 Technical Challenges

With each member of the team in a different location, working on the code for the project was not as easy as if we were all in the same building using the same network or database. We also decided on using PHP as the programming language as it suited the application requirements and we were familiar with the language. It would not have made sense for us to program the application in C or C++.

8.2. Project Implementation Analysis

Our team was well organized and used clear communication throughout the project development. We created a space for the group to be in constant communication and set weekly deadlines that were not included by the professor. Everyone worked well together and each pulled their own weight on the project. Below, we have created charts for the hours worked per person and the average lines of code per person. We divided the work throughout the semester, but everyone would still critique and help out each other when needed. The bugs found in the testing from the previous section were corrected and the application runs as intended.

Hours Worked	
Robert Herber	100
Lawrence Ferretti	94
Nicholas Murphy	98
Naga Panganamamula (2nd half of semester)	45
Total	**337**

Lines of Code Written	
Robert Herber	543
Lawrence Ferretti	458
Nicholas Murphy	972
Naga Panganamamula (2nd half of semester)	422
Total	**2395**

8.3. Lessons Learned (a.k.a. What Can We Do Better Next Time?)

The project would have been much more streamlined and efficient if we had not been forced to use the waterfall method. Having each stage of the process need to be completely finished before the next one can start is not viable in the business world. With the waterfall process, the project would not have survived the constantly changing markets and requirements in the real world. The cost and time required with using the waterfall method would have become a liability for the company and more than likely would not have made it to production.

9. Appendices

9.1. Meeting Minutes

Initial customer meeting 2/2/17

Description:

Online version of campus bookstore that students can order from and have the books shipped to them.

Customer: Jon the bookstore manager

- Hired us to solve a problem
- Make life easier for students to buy books
- No easy way for online students to buy books
- Make online bookstore
- Order books
- Have shipped to them
- On campus and online bookstore completely separate

Warehouse manager is Joe

- Joe gets orders and ships them out

Administrative assistant is Jenny 8675309

- Office hours are office hours

Books orders are placed:

- Books associated with course, some required and recommended
- Some courses may have no books,
- Some can have both required and recommended
- New books, used books, rental books, and eBooks
- Prices and quantity are different for each
- EBook will get email with link to download
- Physical purchase/rent will be shipped for 14.99$ usps priority mail
- Ebook no shipping charge.
- Only ship in US and US territory
- No international shipping
- There can be multiple sections of a class.
- Sections can have separate textbooks for the same class but different teachers
- Students can search for books by title, authors, professors
- “Show me all books taught by professor smith”
- Search by course
- Show books for each section
- Search by keyword “find all books about microbiology”
- Search by ISBN
- Only deal with 1 semester at a time
- Stops selling books by 15th class day
- Website can only order books when book selling is going on
- List of books and courses from registrar. Bill Bigsby (the registrar)
- Data file contains all books, professors, csv file contains all info.
- Amount does not change. Sell out and it is out
- 4 different prices for the 4 formats.
- Folder full of files that contain covers for all the book. Named based on ISBN.
- Search results will also show picture of the book
- Add book to shopping cart, can checkout. Put things in and take things out
- Shows subtotal
- Enter billing address, shipping address, process credit card, send them receipt
- Can pay with PayPal (simulated), VISA and MasterCard a valid credit card will be 16 digit , cc verification = 777
- PayPal valid email and PW 12345678 anything else is invalid
- Students can pay for books with fin aid award. Will have to log in using Kennesaw email and PW Simulated if it is enough to cover cost
- Aid comes from IB Greedy (Bursar). He knows if student has fin aid
- Do not have to be a student or be logged in store use system. Can checkout as a guest
- Conform to KSU branding, style guidelines. See UITS Gilbert can help you.

- Website does not care about rental returns
- Website will not tell you when the rental is due. Good idea to print due date
- Online store will not take a return
- Due last day of class
- Talk to Dilbert for domain URL
- System will NOT retain credit card info
- People will pay GA sales tax no matter where they live
- 7% sales tax
- No comments section or review
- Books for a course will not change. List will never change
- Jenny's area code is 404
- Help and FAQ page with contact info
- No discounts
- No mixing payment options
- Don't worry about saving info
- Typical shopping cart stuff
- Don't store stuff separately from banner
- Email receipt
- Will give folder later
- Nothing coming until after midterm (files)
- No canceling orders
- We get a receipt too. Prints invoice. System save invoice in a folder, jenny goes to the folder by date and print out invoice and give them to joe.
- Dilbert sets up folder. Shared folder.
- Only 1 shipping option
- Conform to KSU branding guidelines
- Would be nice if they could sort search results
- Site should tell them the stock of book. Can't put book in cart if not in stock
- No tracking number
- Ask Dilbert about hosting the website. Has server farm.
- Should be available 24/7
- Should be visible on all devices
- Can ship to Guam.
- Anyone can buy Ebook, does not require shipping
- Customers can mix and match different types of books in the same cart. Can have 2 copies of the same book
- .txt file from registrar
- PO box can be shipped to
- Folder on Jenny's desktop where all invoices are saved
- No admin login.
- Dilbert's number 555 IDONT CARE
- Joe doesn't have computer. Jenny prints previous days invoices and gives them to Joe, he goes to warehouse and boxes them up and ships them out

- Can't put in incorrectly formatted address (need zip, and city and state)
- If book is out of stock, still display in search results. Will show quantity
- Customers can have different shipping and billing addresses.
- No language supports
- 2 day priority shipping
- Cart should remember items in it if the customer quits and then returns
- Can sort by required/recommended. Details will show all courses that use that book
- Create meeting minutes(?)

Key Stakeholders

1. Jon the Bookstore Manager (Tues/Thurs 1-3pm J302)
2. Jenny the Assistant (404-867-5309) (Tues/Thurs 1-3pm J302)
3. Joe the Warehouse Manager (Tues/Thurs 1-3pm J302)
4. Bill Bixby the Registrar (Tues/Thurs 1-3pm J302)
5. I.B. Greedy the Bursar (Tues/Thurs 1-3pm J302)
6. Dilbert from UITS (helps with guidelines for web practices/KSU branding) (Tues/Thurs 1-3pm J302)
7. The user

Important Dates:

Project is due last day of class

Meeting Minutes 2/21/17

Group meeting

- Clarified user stories
    - Critiques/proofread user stores.
- Prepared for 10am meeting with professor
- Discussed finishing stakeholder requests
- Set up study session for midterm
- Discussed filling in any requirement gaps.

Meeting Minutes 2/22/17

Meeting with Stakeholders and Professor

- **Technical Details
    - Implementation, not design
- Jon Bookstore Manager Questions
- Invoice Layout
    - Receipt != Invoice (Precise wording)

        - Receipt: Shipping Info, Items, Details, Price, Rental Notice (Last Day), eBook Link
    - (Packing Slip, not Invoice)
        - Invoice = What someone owes you (Billing Address, Sales Tax, etc)
    - Receipt: Columns format
        - Packing Slip = What the Warehouse Manager uses
    - Receipt used by Jenny & Joe
- Design Decision, up to us at this point:
    - Specific details (folder name, etc) up to us to decide later on.
    - Named a certain thing, stored manually

        - *Incredible Hulk creation - TV vs. Comics*
        - *Gamma Radiation, either way*

    - Non-system implementation
    - Kennesaw.edu Style and Branding, find via Google
    - Database -> Implementation question (Database not necessary)
    - UITS Webserver (perfect, no need to manage by UITS)

- Financial Aid - how is it displayed?
    - Via login - fake text file will contain this information in implementation
- Nice feature - Display financial aid to customer before checkout and during
    - Now a requirement
    - Decrement the aid when used as well
- Aimed at undergraduates,
    - Don't force team to use technologies that they don't know
- User Stories/Stakeholder Summaries
    - Don't duplicate if two stakeholders have same objective
    - Changing quantity (checking out if quantity is over limit) lots more detail
    - RH5: Reword (combine increase & decrease, change language)
    - RH7: Add distinct inventory to end
    - Define invalid information
    - More duplicates
    - Details missing on financial aid
    - We need to be able to “check off” items as we develop
    - Stakeholder and relationship to the project (why are they relevant)
    - After turning in, revise into stakeholder profiles
    - Use template (leave areas we don't use, blank)
    - Again, "removing books" phrasing needs to change
    - More goldplating on Nick’s part
    - Wording: reject orders -> “don't want them to enter that information”
    - No shipping cost if e-book
- Make sure stakeholder requests and user stories match each other

- (Stakeholders Requests should have been written first)
- Fix grouping (not by team member, but by stakeholder)

- Deciding time for next meeting (Wait to hear by back from Lawrence)

- End Meeting

Meeting Minutes 3/2/17

- Group discovered that their meeting with the professor was actually scheduled for yesterday
    - Panic ensued
- Group discussed sections 1 and 2 in the template provided by the professor.
    - Nick agreed to complete section 2 and section 1 will be completed by Lawrence and Robert
- Group agreed that the next meeting with the professor/customer should be set for Thursday, 3/16/17 at 11am EST.
- An email was sent to the professor requesting for that meeting to be scheduled.

Meeting Minutes 3/9/17

- Follow template, don't change fonts
- Don't do a database, we could do the entire project with javascanner
- Allocate time for test procedures and design
- Don't put cost in the nodes - nevermind go ahead and do it that way
- Need critical path and slack time
- Remove spacing on estimates for user stories
- Get user stories from customer requests in stakeholders profiles
    - Write down all requests
- One use case each for turn-in on Tuesday
- Gantt is hard to see, use template for the Gantt chart just uploaded on D2L
    - Can't have overlapping tasks > number of people in the group
- Use case diagram
    - database not an actor, part of system
    - open arrow to extend
    - arrow from extend going wrong way
    - all searches extend search
    - 15-20 bubbles
    - does not want use case diagrams for each use case
    - one use case diagram for entire system
        - either do big diagram or break them up
    - do all extensions
    - checkout includes pay for books by different types
    - Going to be a large diagram
    - Each group member must have 1 written use case.

Meeting Minutes 3/13/17

- Sections 1 or 2.
    - Not 1 and 2 sentences
    - Wish he never said that
    - 1. reason the business is doing it
    - cares more about 2.1 (user vision)
    - Major features = high level top level features. Summaries of major features
- pencast for Gantt chart
- Use case diagram
    - big
    - gone way to far down rabbit hole
    - search breakdown are extends not include
    - change display to view details page
    - don't need arrow going to everything
    - don't need quantity or picture
    - checkout is the biggest use case
        - includes generating receipts
        - includes updating inventory
        - paypal, visa, mastercard etc are extends
    - generating receipts would include emailing receipt and showing rental due date
    - warehouse manager never touches the system, delete him. Never touches any use cases.
    - UITIS guy needs to be deleted too
    - uploading csv file not included in system
    - bookstore manager never touches the system, not an actor
    - use case not stakeholder request.
    - Use case is something an actor actually does with the system
    - Bursar would be an actor
    - use case diagram actors would include
        - user, possible jenny, bank, paypal, bursar
    - add login (Actor: UITIS guy for authenticating login)
- written use case
    - don't add obvious stuff to assumptions
    - don't need trigger for search
    - sort
        - small little thing
        - 1 on flow may be precondition
        - assumption less strict than precondition
        - not going to show professor in search results unless search by professor
        - thumbnail title, description
        - doing them pretty well
        - detailed and specific actions and interactions
    - 4pm another meeting
        - shorter meeting
- Friday - Presentations due

- Written Use Case
    - Don't need a trigger for search
    - Flow of Event steps should be summary
    - Step 1 - user types a phrase in the search box
    - missed step 2 (my bad)
    - Step 3 - Catalog returns results (use step 4, but needs more detail - "the system searches through the file that contains the book information...")
    - Assumption - user is on the homepage where the box is displayed
    - Precondition is absolute, assumption is just an assumption and is less strict
- Sort Use Case
    - Remove step 1
    - Remove professor from sort
    - Step 3 - user clicks on header...
- 17 to 20 minute presentation due on Friday
- Start coding next Wednesday

Meeting Minutes 3/16/17

- Revise Gantt chart to include real dates
    - otherwise looking good
    - Add black color to legend
- Use case diagram
    - Problems with extends and includes
    - Checkout does not make sense
    - checkout includes providing payment
    - Payment is a line going from user as well
    - Detail is in the written description
    - Nothing in diagram shows sequence of events
    - Checking out includes generating receipt
    - Save receipt in shared folder, email, and display on screen are all extends
        - All includes rental note and ebook link
    - More involved in checking out
        - update inventory (is very detailed)
        - 2 things floating off by themselves
    - Made up creating account
    - Need login use case
    - Pay by fin aid includes the necessity to log in
    - Separate student actor for login
    - Student extends customer
- Checkout written use case

Meeting Minutes 3/29/17

- Grading
    - Customer introduction for 1.2. Dont make up marketing BS
    - Not vision statement he is looking for. Should be longer. Merge additional details into vision statement

- Don't give bullet points. Rewrite 2.2
- Scope = all features included in release
- Sell tshirts and coffee mugs, next year, print packing slips for JOe next year in scope of subsequent releases
- Work on 2.5
- Add approved by customer to WBS for most things
- Add more detail to HTML and CSS, add unit testing to coding
- Checkout includes all the extra use cases, not pay for order.
- For search, 1 is not a step, dont need sub bullets
- Subflow needs to be inside normal flow

Doc is loosely based on memory jogger
Doesn't want BS like he told us to do before on business requirements?
Use customer's intro for 1.2
Not the vision statement he's looking for
-shouldn't be one sentence
Vision statement should set the stage, the stakeholder's vision
Merge vision sections together into one statement?
No bullet points on major features
-bullet points are crutch of a writer, engineers are not brief
-looking for a paragraph or two on each feature
Rewrite 2.2 completely
2.4 - doesn't like the phrases "proof-of-concept" "minimum viable product"
2.5 - things to implement next year
Stakeholder requests should all be documented and answered, even if answer is no
No bugs in software
No problems with stakeholder requests
Project schedule - 2.3 missing review with customer
-need to include approved by customer for created objects
3.3 - code not realistic, need more time?
-unit testing needs to be in development, not phase 4
-write test plan and test procedures, and get those approved
-have a cleaned up version of complete activity tree
Gantt - they don't line up, keep periods above each phase
Use case diagram - checkout includes modify shipping
-checkout includes generate receipts, could stop right there
-provide customer info instead of modify address
Search written use case - Normal flow 1 is not a step
-start at step 2
-don't do sub-bullets? Make them all numbers instead
-aren't sub flows if they are separate, sub flow is inside another flow
- make alt flow instead
Checkout - subflow 3a, indentation is all messed up, needs to be corrected
-he hates bullets

-too complicated, could be different use cases
-can chain use cases
-avoid miracle steps, subflow 3c.b.
-almost(?) too detailed
-focus on interactions
-exceptional flow - is an entire flow, not a list of potential things to go wrong
-meeting on thursday 1:30 EST

-Hypothetical "year from now" release
***E.g. "Sell coffee mugs & tickets in addition to textbook"
--Even ones we said we're not going to do in stakeholder requests.
Stakeholder Profiles:
*Move requests from John to relevant
stakeholders
Project Schedule
*Review with customer
*"Created" -> "Approved by Customer"
*HTML/CSS -> More time
--More detail for HTML (what pages, etc)
*Unit test goes in Coding
--Write 'Test Plan'
--Move "All 4 phases" Activity Tree up from appendix

Gantt Chart

-Repeat headers for sub-sections to indicate scale
Use Case Diagram

-Checkout still "wrong"
  *Syntax/logic mistakes
**Add line
Written Use Cases

Search
--Step 1 is not a step - step 2 is
--Subflows are inside another flow
Checkout
---
*Subflow has to have more than one option (instead, another step with a conditional statement)
--No Sub-Sub Flows
*Subflow 3a -> move to separate use cases
Chain Use Cases
Still has "Miracle" steps

--Logged in versus not logged in at checkout
**After logging in, show summary and offer option to pay
Make it a little less technical -
interaction instead (Subflow 3C)
Exceptional Flow
--Actual flow, not "pulling out errors" - Use Case can't be completed because... [exits abnormally]
---Remove "Back Button" steps
---Exceptional Flow 1, Exception FLow 2

Meeting Minutes 3/30/17

- Module decomposition diagram
    - Not to list all features, that is a feature tree/fishbone diagram
    - Purpose is to identify the pieces of code
    - Everything should be pieces of code
    - Top level is like a java package
    - class called paypal, financial aid, credit card
    - No software to managing inventory system
    - Parsing updating inventory file
    - retrieve book images
- Wireframe diagrams
    - Pictures may be too small on results
    - Wasting too much space with logo
    - Add dimensions to boxes

Meeting Minutes 4/19/17

- Can enter all 3 boxes at the same time, but combines them. User is dumb. Doesn't like the interface. Give choice to display search. Need to do last name first for professor. Price is not listed on search results.
- search shouldn't combine the boxes
- doesn't like the interface
- search by name- last name first
- search results - price not listed, don't display long descriptions
- link should be underlined
- pics are links as well
- tooltips not helpful
- shopping cart should be own page, even though no issue previously with the wireframes?
- details page - book pic too small
- name on credit card isn't there
- no up-down clicker on verification code
- doesn't like error message
- couldn't get receipt to work

- need more testing
- test cases are explicit
- Too much description
- All thumbnail books should be clickable
- View details
    - Doesn't like the shopping cart
- Work on testing, follow lecture

9.2. Other Supporting Documents

- This should be 20.25/25
    - Professor said he fixed the grade to 20.25/25 (over email) on 3/29/17

XXXXXXXX Title goes here if long and long and long
XXXXXXXX by John Smith, 2015, Random House 978-123456-789-0
XXXXXXXX This is the really long description. Lorem ipsum dolor sit amet. Lorem….
XXXXXXXX New: 4 Used: 5 Rental: 5 EBook: available

Written Deliverable #1 Grade Form — SWE 6623 –Software Engineering I
Last Update: March 22, 2017

Section: W01 **Group:** 1 **Evaluation date:** 3/29/2017 **Grade:** 20.25/25

Introduction, Vision, Scope	1.5 ○0 NONE ☒1 POOR ☒2 GOOD ○3 EXCELLENT	**Use Case Diagram**	○0 NONE ○1 POOR ○2 AVERAGE ☒3 GOOD ○4 EXCELLENT
Stakeholder Profiles	○0 NONE ○1 POOR ○2 AVERAGE ○3 GOOD ☒4 EXCELLENT	*UML Syntax*	○ Poor/Missing ○ Fair ☒ Good
All Major Stakeholders	○ Poor/Missing ○ Fair ☒ Good	*Logical Structure*	○ Poor/Missing ☒ Fair ○ Good
All Stakeholder Requests	○ Poor/Missing ○ Fair ☒ Good	**Written Use Cases** 2.5	○0 NONE ○1 POOR ☒2 AVERAGE ☒3 GOOD ○4 EXCELLENT
User Stories	○0 NONE ○1 POOR ○2 GOOD ☒3 EXCELLENT	*Use Case Format, Style*	○ Poor/Missing ○ Fair ☒ Good
Syntax, Structure and Content (INVEST)	○ Poor/Missing ○ Fair ○ Good	*Completeness*	○ Poor/Missing ○ Fair ☒ Good
Completeness	○ Poor/Missing ○ Fair ○ Good	*Correctness*	○ Poor/Missing ☒ Fair ○ Good
Project Plan 3.25	○0 NONE ○1 POOR ○2 AVERAGE ☒3 GOOD ☒4 EXCELLENT	**Miscellaneous**	○0 NONE ○1 POOR ○2 GOOD ☒3 EXCELLENT
Work Breakdown Structure	○ Poor/Missing ○ Fair ☒ Good	*Meeting Minutes, Other Docs*	○ Poor/Missing ○ Fair ☒ Good
Milestones	○ Poor/Missing ○ Fair ☒ Good	*Charts Legible, Labeled*	○ No ○ Partially ☒ Yes
Activity Tree	○ Poor/Missing ○ Fair ☒ Good	*Follows required template*	○ No ○ Partially ☒ Yes
Gantt Chart (ungraded)	○ Poor/Missing ☒ Fair ☒ Good	*Submitted as a PDF*	○ No ☒ Yes

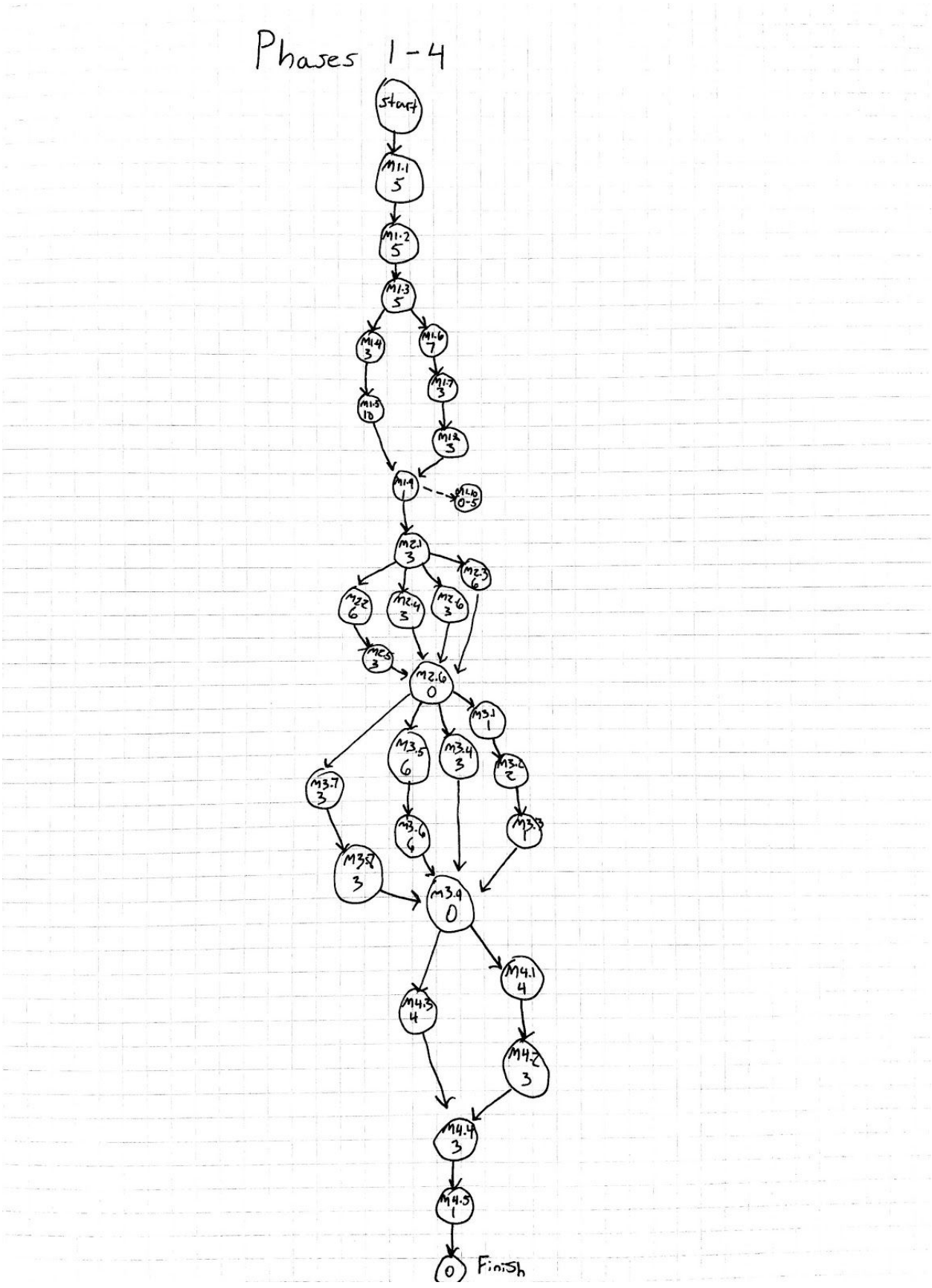
Comments:

Verbal feedback delivered during virtual office hours on 3/29/2017.

Changes made for resubmission:

1. Updated and rewrote much of section 2 (Vision and Scope of the Solution).
2. Deleted unnecessary Gantt charts that did not have the headers.
3. Corrected and updated the use case diagram.

4. Corrected and updated the “Search” and the “Checkout” written use cases.


Phases 1-4
Start
M1.1 5
M1.2 5
M1.3 5
M1.4 3
M1.6 7
M1.7 3
M1.8 10
M1.8 3
M1.9
M1.10 0-5
M2.1 3
M2.2 6
M2.3 6
M2.4 3
M2.5 3
M2.6 0
M3.1 1
M3.2 2
M3.3 1
M3.4 3
M3.5 6
M3.6 4
M3.7 3
M3.8 3
M3.9 0
M4.1 4
M4.2 3
M4.3 4
M4.4 3
M4.5 1
0 Finish

```css
body{
    font-family: 'Nunito', sans-serif !important;
}

h1, h2, h3, h4, h5{
    font-family: 'Roboto Condensed', sans-serif !important;
}

a{
    text-decoration:underline !important;
}

input:invalid {
    border: 2px solid red;
}


.table-striped>tbody>tr:first-child, .panel-default>.panel-heading {
    background-color: #FEBC11 !important;
}

.tab{
    border: 2px solid black;
    border-radius: 10px 10px 0px 0px;
    display: inline-block;
    padding: 0.2em 1em;
    font-size: larger;
    text-decoration: none !important;
}

.tab a, .btn {
    text-decoration:none !important;
}


.spaceFromTop{
    margin-top: 15%;
}


.checkoutIcon{
    height: 120px;
    width: 75px;
}

.searchIcon{
    height: 185px;
    width: 110px;
}

.detailIcon{
    height: 330px;
    width: 235px;
}


.largerFont{
    font-size:larger;
}


#header{
    margin-top: 1em;
}

#content, #header, .row{
```

```
    max-width: 100%;
}

#ksulogo{
    height: 45px;
    width: auto;
}
```

```
/*!
 * Bootstrap v3.3.7 (http://getbootstrap.com)
 * Copyright 2011-2016 Twitter, Inc.
 * Licensed under MIT (https://github.com/twbs/bootstrap/blob/master/LICENSE)
 *//*! normalize.css v3.0.3 | MIT License | github.com/necolas/normalize.css
 */html{font-family:sans-serif;-webkit-text-size-adjust:100%;-ms-text-size-adjust:100%}body{margi
n:0}article,aside,details,figcaption,figure,footer,header,hgroup,main,menu,nav,section,summary{d
isplay:block}audio,canvas,progress,video{display:inline-block;vertical-align:baseline}audio:not(
[controls]){display:none;height:0}[hidden],template{display:none}a{background-color:transparent}
a:active,a:hover{outline:0}abbr[title]{border-bottom:1px
dotted}b,strong{font-weight:700}dfn{font-style:italic}h1{margin:.67em
0;font-size:2em}mark{color:#000;background:#ff0}small{font-size:80%}sub,sup{position:relative;fo
nt-size:75%;line-height:0;vertical-align:baseline}sup{top:-.5em}sub{bottom:-.25em}img{border:0}s
vg:not(:root){overflow:hidden}figure{margin:1em
40px}hr{height:0;-webkit-box-sizing:content-box;-moz-box-sizing:content-box;box-sizing:content-b
ox}pre{overflow:auto}code,kbd,pre,samp{font-family:monospace,monospace;font-size:1em}button,inpu
t,optgroup,select,textarea{margin:0;font:inherit;color:inherit}button{overflow:visible}button,se
lect{text-transform:none}button,html
input[type=button],input[type=reset],input[type=submit]{-webkit-appearance:button;cursor:pointer
}button[disabled],html
input[disabled]{cursor:default}button::-moz-focus-inner,input::-moz-focus-inner{padding:0;border
:0}input{line-height:normal}input[type=checkbox],input[type=radio]{-webkit-box-sizing:border-box
;-moz-box-sizing:border-box;box-sizing:border-box;padding:0}input[type=number]::-webkit-inner-sp
in-button,input[type=number]::-webkit-outer-spin-button{height:auto}input[type=search]{-webkit-b
ox-sizing:content-box;-moz-box-sizing:content-box;box-sizing:content-box;-webkit-appearance:text
field}input[type=search]::-webkit-search-cancel-button,input[type=search]::-webkit-search-decora
tion{-webkit-appearance:none}fieldset{padding:.35em .625em .75em;margin:0 2px;border:1px solid
silver}legend{padding:0;border:0}textarea{overflow:auto}optgroup{font-weight:700}table{border-sp
acing:0;border-collapse:collapse}td,th{padding:0}/*! Source:
https://github.com/h5bp/html5-boilerplate/blob/master/src/css/main.css */@media
print{*,:after,:before{color:#000!important;text-shadow:none!important;background:0
0!important;-webkit-box-shadow:none!important;box-shadow:none!important}a,a:visited{text-decorat
ion:underline}a[href]:after{content:" (" attr(href) ")"}abbr[title]:after{content:" ("
attr(title)
")"}a[href^="javascript:"]:after,a[href^="#"]:after{content:""}blockquote,pre{border:1px solid
#999;page-break-inside:avoid}thead{display:table-header-group}img,tr{page-break-inside:avoid}img
{max-width:100%!important}h2,h3,p{orphans:3;widows:3}h2,h3{page-break-after:avoid}.navbar{displa
y:none}.btn>.caret,.dropup>.btn>.caret{border-top-color:#000!important}.label{border:1px solid
#000}.table{border-collapse:collapse!important}.table td,.table
th{background-color:#fff!important}.table-bordered td,.table-bordered th{border:1px solid
#ddd!important}}@font-face{font-family:'Glyphicons
Halflings';src:url(../fonts/glyphicons-halflings-regular.eot);src:url(../fonts/glyphicons-halfli
ngs-regular.eot?#iefix)
format('embedded-opentype'),url(../fonts/glyphicons-halflings-regular.woff2)
format('woff2'),url(../fonts/glyphicons-halflings-regular.woff)
format('woff'),url(../fonts/glyphicons-halflings-regular.ttf)
format('truetype'),url(../fonts/glyphicons-halflings-regular.svg#glyphicons_halflingsregular)
format('svg')}.glyphicon{position:relative;top:1px;display:inline-block;font-family:'Glyphicons
Halflings';font-style:normal;font-weight:400;line-height:1;-webkit-font-smoothing:antialiased;-m
oz-osx-font-smoothing:grayscale}.glyphicon-asterisk:before{content:"\002a"}.glyphicon-plus:befor
e{content:"\002b"}.glyphicon-eur:before,.glyphicon-euro:before{content:"\20ac"}.glyphicon-minus:
before{content:"\2212"}.glyphicon-cloud:before{content:"\2601"}.glyphicon-envelope:before{conten
t:"\2709"}.glyphicon-pencil:before{content:"\270f"}.glyphicon-glass:before{content:"\e001"}.glyp
hicon-music:before{content:"\e002"}.glyphicon-search:before{content:"\e003"}.glyphicon-heart:bef
ore{content:"\e005"}.glyphicon-star:before{content:"\e006"}.glyphicon-star-empty:before{content:
"\e007"}.glyphicon-user:before{content:"\e008"}.glyphicon-film:before{content:"\e009"}.glyphicon
-th-large:before{content:"\e010"}.glyphicon-th:before{content:"\e011"}.glyphicon-th-list:before{
content:"\e012"}.glyphicon-ok:before{content:"\e013"}.glyphicon-remove:before{content:"\e014"}.g
lyphicon-zoom-in:before{content:"\e015"}.glyphicon-zoom-out:before{content:"\e016"}.glyphicon-of
f:before{content:"\e017"}.glyphicon-signal:before{content:"\e018"}.glyphicon-cog:before{content:
"\e019"}.glyphicon-trash:before{content:"\e020"}.glyphicon-home:before{content:"\e021"}.glyphico
n-file:before{content:"\e022"}.glyphicon-time:before{content:"\e023"}.glyphicon-road:before{cont
ent:"\e024"}.glyphicon-download-alt:before{content:"\e025"}.glyphicon-download:before{content:"\
e026"}.glyphicon-upload:before{content:"\e027"}.glyphicon-inbox:before{content:"\e028"}.glyphico
n-play-circle:before{content:"\e029"}.glyphicon-repeat:before{content:"\e030"}.glyphicon-refresh
:before{content:"\e031"}.glyphicon-list-alt:before{content:"\e032"}.glyphicon-lock:before{conten
```

t:"\e033"}.glyphicon-flag:before{content:"\e034"}.glyphicon-headphones:before{content:"\e035"}.glyphicon-volume-off:before{content:"\e036"}.glyphicon-volume-down:before{content:"\e037"}.glyphicon-volume-up:before{content:"\e038"}.glyphicon-qrcode:before{content:"\e039"}.glyphicon-barcode:before{content:"\e040"}.glyphicon-tag:before{content:"\e041"}.glyphicon-tags:before{content:"\e042"}.glyphicon-book:before{content:"\e043"}.glyphicon-bookmark:before{content:"\e044"}.glyphicon-print:before{content:"\e045"}.glyphicon-camera:before{content:"\e046"}.glyphicon-font:before{content:"\e047"}.glyphicon-bold:before{content:"\e048"}.glyphicon-italic:before{content:"\e049"}.glyphicon-text-height:before{content:"\e050"}.glyphicon-text-width:before{content:"\e051"}.glyphicon-align-left:before{content:"\e052"}.glyphicon-align-center:before{content:"\e053"}.glyphicon-align-right:before{content:"\e054"}.glyphicon-align-justify:before{content:"\e055"}.glyphicon-list:before{content:"\e056"}.glyphicon-indent-left:before{content:"\e057"}.glyphicon-indent-right:before{content:"\e058"}.glyphicon-facetime-video:before{content:"\e059"}.glyphicon-picture:before{content:"\e060"}.glyphicon-map-marker:before{content:"\e062"}.glyphicon-adjust:before{content:"\e063"}.glyphicon-tint:before{content:"\e064"}.glyphicon-edit:before{content:"\e065"}.glyphicon-share:before{content:"\e066"}.glyphicon-check:before{content:"\e067"}.glyphicon-move:before{content:"\e068"}.glyphicon-step-backward:before{content:"\e069"}.glyphicon-fast-backward:before{content:"\e070"}.glyphicon-backward:before{content:"\e071"}.glyphicon-play:before{content:"\e072"}.glyphicon-pause:before{content:"\e073"}.glyphicon-stop:before{content:"\e074"}.glyphicon-forward:before{content:"\e075"}.glyphicon-fast-forward:before{content:"\e076"}.glyphicon-step-forward:before{content:"\e077"}.glyphicon-eject:before{content:"\e078"}.glyphicon-chevron-left:before{content:"\e079"}.glyphicon-chevron-right:before{content:"\e080"}.glyphicon-plus-sign:before{content:"\e081"}.glyphicon-minus-sign:before{content:"\e082"}.glyphicon-remove-sign:before{content:"\e083"}.glyphicon-ok-sign:before{content:"\e084"}.glyphicon-question-sign:before{content:"\e085"}.glyphicon-info-sign:before{content:"\e086"}.glyphicon-screenshot:before{content:"\e087"}.glyphicon-remove-circle:before{content:"\e088"}.glyphicon-ok-circle:before{content:"\e089"}.glyphicon-ban-circle:before{content:"\e090"}.glyphicon-arrow-left:before{content:"\e091"}.glyphicon-arrow-right:before{content:"\e092"}.glyphicon-arrow-up:before{content:"\e093"}.glyphicon-arrow-down:before{content:"\e094"}.glyphicon-share-alt:before{content:"\e095"}.glyphicon-resize-full:before{content:"\e096"}.glyphicon-resize-small:before{content:"\e097"}.glyphicon-exclamation-sign:before{content:"\e101"}.glyphicon-gift:before{content:"\e102"}.glyphicon-leaf:before{content:"\e103"}.glyphicon-fire:before{content:"\e104"}.glyphicon-eye-open:before{content:"\e105"}.glyphicon-eye-close:before{content:"\e106"}.glyphicon-warning-sign:before{content:"\e107"}.glyphicon-plane:before{content:"\e108"}.glyphicon-calendar:before{content:"\e109"}.glyphicon-random:before{content:"\e110"}.glyphicon-comment:before{content:"\e111"}.glyphicon-magnet:before{content:"\e112"}.glyphicon-chevron-up:before{content:"\e113"}.glyphicon-chevron-down:before{content:"\e114"}.glyphicon-retweet:before{content:"\e115"}.glyphicon-shopping-cart:before{content:"\e116"}.glyphicon-folder-close:before{content:"\e117"}.glyphicon-folder-open:before{content:"\e118"}.glyphicon-resize-vertical:before{content:"\e119"}.glyphicon-resize-horizontal:before{content:"\e120"}.glyphicon-hdd:before{content:"\e121"}.glyphicon-bullhorn:before{content:"\e122"}.glyphicon-bell:before{content:"\e123"}.glyphicon-certificate:before{content:"\e124"}.glyphicon-thumbs-up:before{content:"\e125"}.glyphicon-thumbs-down:before{content:"\e126"}.glyphicon-hand-right:before{content:"\e127"}.glyphicon-hand-left:before{content:"\e128"}.glyphicon-hand-up:before{content:"\e129"}.glyphicon-hand-down:before{content:"\e130"}.glyphicon-circle-arrow-right:before{content:"\e131"}.glyphicon-circle-arrow-left:before{content:"\e132"}.glyphicon-circle-arrow-up:before{content:"\e133"}.glyphicon-circle-arrow-down:before{content:"\e134"}.glyphicon-globe:before{content:"\e135"}.glyphicon-wrench:before{content:"\e136"}.glyphicon-tasks:before{content:"\e137"}.glyphicon-filter:before{content:"\e138"}.glyphicon-briefcase:before{content:"\e139"}.glyphicon-fullscreen:before{content:"\e140"}.glyphicon-dashboard:before{content:"\e141"}.glyphicon-paperclip:before{content:"\e142"}.glyphicon-heart-empty:before{content:"\e143"}.glyphicon-link:before{content:"\e144"}.glyphicon-phone:before{content:"\e145"}.glyphicon-pushpin:before{content:"\e146"}.glyphicon-usd:before{content:"\e148"}.glyphicon-gbp:before{content:"\e149"}.glyphicon-sort:before{content:"\e150"}.glyphicon-sort-by-alphabet:before{content:"\e151"}.glyphicon-sort-by-alphabet-alt:before{content:"\e152"}.glyphicon-sort-by-order:before{content:"\e153"}.glyphicon-sort-by-order-alt:before{content:"\e154"}.glyphicon-sort-by-attributes:before{content:"\e155"}.glyphicon-sort-by-attributes-alt:before{content:"\e156"}.glyphicon-unchecked:before{content:"\e157"}.glyphicon-expand:before{content:"\e158"}.glyphicon-collapse-down:before{content:"\e159"}.glyphicon-collapse-up:before{content:"\e160"}.glyphicon-log-in:before{content:"\e161"}.glyphicon-flash:before{content:"\e162"}.glyphicon-log-out:before{content:"\e163"}.glyphicon-new-window:before{content:"\e164"}.glyphicon-record:before{content:"\e165"}.glyphicon-save:before{content:"\e166"}.glyphicon-open:before{content:"\e167"}.glyphicon-saved:before{content:"\e168"}.glyphicon-import:before{content:"\e169"}.glyphicon-export:before{content:"\e170"}.glyphicon-send:before{content:"\e171"}.glyphicon-floppy-disk:before{content:"\e172"}.glyphicon-floppy-saved:before{content:"\e173"}.glyphicon-floppy-remove:before{content:"\e174"}.glyphicon-floppy-save:before{content:"\e175"}.glyphicon-floppy-open:before{content:"\e176"}.glyphicon-credit-card:before{content:"\e177"}.glyphicon-transfer:before{content:"\e178"}.glyphicon-cutlery:before{content:"\e179"}.glyphicon-header:before{content:"\e180"}.glyphicon-compressed:before{content:"\e181"}.glyphicon-earphone:before{content:"\e182"}.glyph

```
small,h4 .small,h4 small,h5 .small,h5 small,h6 .small,h6
small{font-weight:400;line-height:1;color:#777}.h1,.h2,.h3,h1,h2,h3{margin-top:20px;margin-bottom:10px}.h1 .small,.h1 small,.h2 .small,.h2 small,.h3 .small,.h3 small,h1 .small,h1 small,h2 .small,h2 small,h3 .small,h3
small{font-size:65%}.h4,.h5,.h6,h4,h5,h6{margin-top:10px;margin-bottom:10px}.h4 .small,.h4 small,.h5 .small,.h5 small,.h6 .small,.h6 small,h4 .small,h4 small,h5 .small,h5 small,h6 .small,h6
small{font-size:75%}.h1,h1{font-size:36px}.h2,h2{font-size:30px}.h3,h3{font-size:24px}.h4,h4{font-size:18px}.h5,h5{font-size:14px}.h6,h6{font-size:12px}p{margin:0 0 10px}.lead{margin-bottom:20px;font-size:16px;font-weight:300;line-height:1.4}@media (min-width:768px){.lead{font-size:21px}}.small,small{font-size:85%}.mark,mark{padding:.2em;background-color:#fcf8e3}.text-left{text-align:left}.text-right{text-align:right}.text-center{text-align:center}.text-justify{text-align:justify}.text-nowrap{white-space:nowrap}.text-lowercase{text-transform:lowercase}.text-uppercase{text-transform:uppercase}.text-capitalize{text-transform:capitalize}.text-muted{color:#777}.text-primary{color:#337ab7}a.text-primary:focus,a.text-primary:hover{color:#286090}.text-success{color:#3c763d}a.text-success:focus,a.text-success:hover{color:#2b542c}.text-info{color:#31708f}a.text-info:focus,a.text-info:hover{color:#245269}.text-warning{color:#8a6d3b}a.text-warning:focus,a.text-warning:hover{color:#66512c}.text-danger{color:#a94442}a.text-danger:focus,a.text-danger:hover{color:#843534}.bg-primary{color:#fff;background-color:#337ab7}a.bg-primary:focus,a.bg-primary:hover{background-color:#286090}.bg-success{background-color:#dff0d8}a.bg-success:focus,a.bg-success:hover{background-color:#c1e2b3}.bg-info{background-color:#d9edf7}a.bg-info:focus,a.bg-info:hover{background-color:#afd9ee}.bg-warning{background-color:#fcf8e3}a.bg-warning:focus,a.bg-warning:hover{background-color:#f7ecb5}.bg-danger{background-color:#f2dede}a.bg-danger:focus,a.bg-danger:hover{background-color:#e4b9b9}.page-header{padding-bottom:9px;margin:40px 0 20px;border-bottom:1px solid #eee}ol,ul{margin-top:0;margin-bottom:10px}ol ol,ol ul,ul ol,ul ul{margin-bottom:0}.list-unstyled{padding-left:0;list-style:none}.list-inline{padding-left:0;margin-left:-5px;list-style:none}.list-inline>li{display:inline-block;padding-right:5px;padding-left:5px}dl{margin-top:0;margin-bottom:20px}dd,dt{line-height:1.42857143}dt{font-weight:700}dd{margin-left:0}@media (min-width:768px){.dl-horizontal dt{float:left;width:160px;overflow:hidden;clear:left;text-align:right;text-overflow:ellipsis;white-space:nowrap}.dl-horizontal dd{margin-left:180px}}abbr[data-original-title],abbr[title]{cursor:help;border-bottom:1px dotted #777}.initialism{font-size:90%;text-transform:uppercase}blockquote{padding:10px 20px;margin:0 0 20px;font-size:17.5px;border-left:5px solid #eee}blockquote ol:last-child,blockquote p:last-child,blockquote ul:last-child{margin-bottom:0}blockquote .small,blockquote footer,blockquote small{display:block;font-size:80%;line-height:1.42857143;color:#777}blockquote .small:before,blockquote footer:before,blockquote small:before{content:'\2014 \00A0'}.blockquote-reverse,blockquote.pull-right{padding-right:15px;padding-left:0;text-align:right;border-right:5px solid #eee;border-left:0}.blockquote-reverse .small:before,.blockquote-reverse footer:before,.blockquote-reverse small:before,blockquote.pull-right .small:before,blockquote.pull-right footer:before,blockquote.pull-right small:before{content:''}.blockquote-reverse .small:after,.blockquote-reverse footer:after,.blockquote-reverse small:after,blockquote.pull-right .small:after,blockquote.pull-right footer:after,blockquote.pull-right small:after{content:'\00A0 \2014'}address{margin-bottom:20px;font-style:normal;line-height:1.42857143}code,kbd,pre,samp{font-family:Menlo,Monaco,Consolas,"Courier New",monospace}code{padding:2px 4px;font-size:90%;color:#c7254e;background-color:#f9f2f4;border-radius:4px}kbd{padding:2px 4px;font-size:90%;color:#fff;background-color:#333;border-radius:3px;-webkit-box-shadow:inset 0 -1px 0 rgba(0,0,0,.25);box-shadow:inset 0 -1px 0 rgba(0,0,0,.25)}kbd kbd{padding:0;font-size:100%;font-weight:700;-webkit-box-shadow:none;box-shadow:none}pre{display:block;padding:9.5px;margin:0 0 10px;font-size:13px;line-height:1.42857143;color:#333;word-break:break-all;word-wrap:break-word;background-color:#f5f5f5;border:1px solid #ccc;border-radius:4px}pre code{padding:0;font-size:inherit;color:inherit;white-space:pre-wrap;background-color:transparent;border-radius:0}.pre-scrollable{max-height:340px;overflow-y:scroll}.container{padding-right:15px;padding-left:15px;margin-right:auto;margin-left:auto}@media (min-width:768px){.container{width:750px}}@media (min-width:992px){.container{width:970px}}@media (min-width:1200px){.container{width:1170px}}.container-fluid{padding-right:15px;padding-left:15px;margin-right:auto;margin-left:auto}.row{margin-right:-15px;margin-left:-15px}.col-lg-1,.col-lg-10,.col-lg-11,.col-lg-12,.col-lg-2,.col-lg-3,.col-lg-4,.col-lg-5,.col-lg-6,.col-lg-7,.col-lg-8,.col-lg-9,.col-md-1,.col-md-10,.col-md-11,.col-md-12,.col-md-2,.col-md-3,.col-md-4,.col-md-5,.col-md-6,.col-md-7,.col-md-8,.col-md-9,.col-sm-1,.col-sm-10,.col-sm-11,.col-sm-12,.col-sm-2,.col-s
```

```
.radio-inline{cursor:not-allowed}.checkbox.disabled label,.radio.disabled
label,fieldset[disabled] .checkbox label,fieldset[disabled] .radio
label{cursor:not-allowed}.form-control-static{min-height:34px;padding-top:7px;padding-bottom:7px
;margin-bottom:0}.form-control-static.input-lg,.form-control-static.input-sm{padding-right:0;pad
ding-left:0}.input-sm{height:30px;padding:5px
10px;font-size:12px;line-height:1.5;border-radius:3px}select.input-sm{height:30px;line-height:30
px}select[multiple].input-sm,textarea.input-sm{height:auto}.form-group-sm
.form-control{height:30px;padding:5px
10px;font-size:12px;line-height:1.5;border-radius:3px}.form-group-sm
select.form-control{height:30px;line-height:30px}.form-group-sm
select[multiple].form-control,.form-group-sm textarea.form-control{height:auto}.form-group-sm
.form-control-static{height:30px;min-height:32px;padding:6px
10px;font-size:12px;line-height:1.5}.input-lg{height:46px;padding:10px
16px;font-size:18px;line-height:1.3333333;border-radius:6px}select.input-lg{height:46px;line-hei
ght:46px}select[multiple].input-lg,textarea.input-lg{height:auto}.form-group-lg
.form-control{height:46px;padding:10px
16px;font-size:18px;line-height:1.3333333;border-radius:6px}.form-group-lg
select.form-control{height:46px;line-height:46px}.form-group-lg
select[multiple].form-control,.form-group-lg textarea.form-control{height:auto}.form-group-lg
.form-control-static{height:46px;min-height:38px;padding:11px
16px;font-size:18px;line-height:1.3333333}.has-feedback{position:relative}.has-feedback
.form-control{padding-right:42.5px}.form-control-feedback{position:absolute;top:0;right:0;z-inde
x:2;display:block;width:34px;height:34px;line-height:34px;text-align:center;pointer-events:none}
.form-group-lg
.form-control+.form-control-feedback,.input-group-lg+.form-control-feedback,.input-lg+.form-cont
rol-feedback{width:46px;height:46px;line-height:46px}.form-group-sm
.form-control+.form-control-feedback,.input-group-sm+.form-control-feedback,.input-sm+.form-cont
rol-feedback{width:30px;height:30px;line-height:30px}.has-success .checkbox,.has-success
.checkbox-inline,.has-success .control-label,.has-success .help-block,.has-success
.radio,.has-success .radio-inline,.has-success.checkbox label,.has-success.checkbox-inline
label,.has-success.radio label,.has-success.radio-inline label{color:#3c763d}.has-success
.form-control{border-color:#3c763d;-webkit-box-shadow:inset 0 1px 1px
rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075)}.has-success
.form-control:focus{border-color:#2b542c;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0
0 6px #67b168;box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #67b168}.has-success
.input-group-addon{color:#3c763d;background-color:#dff0d8;border-color:#3c763d}.has-success
.form-control-feedback{color:#3c763d}.has-warning .checkbox,.has-warning
.checkbox-inline,.has-warning .control-label,.has-warning .help-block,.has-warning
.radio,.has-warning .radio-inline,.has-warning.checkbox label,.has-warning.checkbox-inline
label,.has-warning.radio label,.has-warning.radio-inline label{color:#8a6d3b}.has-warning
.form-control{border-color:#8a6d3b;-webkit-box-shadow:inset 0 1px 1px
rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075)}.has-warning
.form-control:focus{border-color:#66512c;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0
0 6px #c0a16b;box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #c0a16b}.has-warning
.input-group-addon{color:#8a6d3b;background-color:#fcf8e3;border-color:#8a6d3b}.has-warning
.form-control-feedback{color:#8a6d3b}.has-error .checkbox,.has-error
.checkbox-inline,.has-error .control-label,.has-error .help-block,.has-error .radio,.has-error
.radio-inline,.has-error.checkbox label,.has-error.checkbox-inline label,.has-error.radio
label,.has-error.radio-inline label{color:#a94442}.has-error
.form-control{border-color:#a94442;-webkit-box-shadow:inset 0 1px 1px
rgba(0,0,0,.075);box-shadow:inset 0 1px 1px rgba(0,0,0,.075)}.has-error
.form-control:focus{border-color:#843534;-webkit-box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0
0 6px #ce8483;box-shadow:inset 0 1px 1px rgba(0,0,0,.075),0 0 6px #ce8483}.has-error
.input-group-addon{color:#a94442;background-color:#f2dede;border-color:#a94442}.has-error
.form-control-feedback{color:#a94442}.has-feedback
label~.form-control-feedback{top:25px}.has-feedback
label.sr-only~.form-control-feedback{top:0}.help-block{display:block;margin-top:5px;margin-botto
m:10px;color:#737373}@media (min-width:768px){.form-inline
.form-group{display:inline-block;margin-bottom:0;vertical-align:middle}.form-inline
.form-control{display:inline-block;width:auto;vertical-align:middle}.form-inline
.form-control-static{display:inline-block}.form-inline
.input-group{display:inline-table;vertical-align:middle}.form-inline .input-group
.form-control,.form-inline .input-group .input-group-addon,.form-inline .input-group
.input-group-btn{width:auto}.form-inline .input-group>.form-control{width:100%}.form-inline
.control-label{margin-bottom:0;vertical-align:middle}.form-inline .checkbox,.form-inline
.radio{display:inline-block;margin-top:0;margin-bottom:0;vertical-align:middle}.form-inline
```

```
.alert-link{font-weight:700}.alert>p,.alert>ul{margin-bottom:0}.alert>p+p{margin-top:5px}.alert-
dismissable,.alert-dismissible{padding-right:35px}.alert-dismissable .close,.alert-dismissible
.close{position:relative;top:-2px;right:-21px;color:inherit}.alert-success{color:#3c763d;backgro
und-color:#dff0d8;border-color:#d6e9c6}.alert-success
hr{border-top-color:#c9e2b3}.alert-success
.alert-link{color:#2b542c}.alert-info{color:#31708f;background-color:#d9edf7;border-color:#bce8f
1}.alert-info hr{border-top-color:#a6e1ec}.alert-info
.alert-link{color:#245269}.alert-warning{color:#8a6d3b;background-color:#fcf8e3;border-color:#fa
ebcc}.alert-warning hr{border-top-color:#f7e1b5}.alert-warning
.alert-link{color:#66512c}.alert-danger{color:#a94442;background-color:#f2dede;border-color:#ebc
cd1}.alert-danger hr{border-top-color:#e4b9c0}.alert-danger
.alert-link{color:#843534}@-webkit-keyframes
progress-bar-stripes{from{background-position:40px 0}to{background-position:0 0}}@-o-keyframes
progress-bar-stripes{from{background-position:40px 0}to{background-position:0 0}}@keyframes
progress-bar-stripes{from{background-position:40px 0}to{background-position:0
0}}.progress{height:20px;margin-bottom:20px;overflow:hidden;background-color:#f5f5f5;border-radi
us:4px;-webkit-box-shadow:inset 0 1px 2px rgba(0,0,0,.1);box-shadow:inset 0 1px 2px
rgba(0,0,0,.1)}.progress-bar{float:left;width:0;height:100%;font-size:12px;line-height:20px;colo
r:#fff;text-align:center;background-color:#337ab7;-webkit-box-shadow:inset 0 -1px 0
rgba(0,0,0,.15);box-shadow:inset 0 -1px 0 rgba(0,0,0,.15);-webkit-transition:width .6s
ease;-o-transition:width .6s ease;transition:width .6s
ease}.progress-bar-striped,.progress-striped
.progress-bar{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent);-webkit-background-size:40px 40px;background-size:40px
40px}.progress-bar.active,.progress.active
.progress-bar{-webkit-animation:progress-bar-stripes 2s linear
infinite;-o-animation:progress-bar-stripes 2s linear infinite;animation:progress-bar-stripes
2s linear infinite}.progress-bar-success{background-color:#5cb85c}.progress-striped
.progress-bar-success{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent)}.progress-bar-info{background-color:#5bc0de}.progress-striped
.progress-bar-info{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent)}.progress-bar-warning{background-color:#f0ad4e}.progress-striped
.progress-bar-warning{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent)}.progress-bar-danger{background-color:#d9534f}.progress-striped
.progress-bar-danger{background-image:-webkit-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent
75%,transparent);background-image:-o-linear-gradient(45deg,rgba(255,255,255,.15)
25%,transparent 25%,transparent 50%,rgba(255,255,255,.15) 50%,rgba(255,255,255,.15)
75%,transparent 75%,transparent);background-image:linear-gradient(45deg,rgba(255,255,255,.15)
```

```php
<?php
session_start();
setlocale(LC_MONETARY,"en_US");


$myfile = fopen("data/students.txt", "r") or die("Unable to open file!");
$studentsString = "";


while(!feof($myfile)) {
  $studentsArray[] = explode(' ', fgets($myfile));
}
fclose($myfile);




if(ISSET($_SESSION["finaid_user"])){

    $funds = 0;
    for($i = 0; $i < count($studentsArray); $i++){
        if(strcmp($studentsArray[$i][2], $_SESSION["finaid_user"]) == 0){
            $funds = $studentsArray[$i][4];
            break;
        }
    }

?>


<h1>Financial Aid Summary</h1>

<form method="post" action="code/receipt">
<table class="table">
    <tr>
        <td class="text-left"><b>Total Cost</b></td>
        <td><?php echo money_format("$%!n", $_SESSION["ordertotal"]); ?></td>
    </tr>

    <tr>
        <td class="text-left"><b>Total Financial Aid</b></td>
        <td><?php echo money_format("$%!n", $funds); ?></td>
    </tr>

    <tr>
        <td> </td>
        <td> </td>
    </tr>

        <tr>
        <td class="largerFont text-left"><b>Total Financial Aid After Payment</b></td>
        <td><?php echo money_format("$%!n", $funds - $_SESSION["ordertotal"]); ?></td>
    </tr>

</table>

    <?php
        if($funds - $_SESSION["ordertotal"] > 0){
    ?>
    <input type="submit" class="btn btn-info btn-lg" value="Pay">
    <?php
        }
    ?>
    </form>
```

```
<?php
}
else{
?>


<h1>Financial Aid Login</h1>

<form method="post" action="code/login">
        <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">User Name:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="finaid_user" required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Password:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="password" class="form-control" name="finaid_password" required>
    </div>


    <input type="submit" class="btn btn-info btn-lg" value="Login">
    </form>

<?php
}
?>
```

```php
<?php
session_start();
setlocale(LC_MONETARY,"en_US");

$pagetitle = "KSU Bookstore - Checkout";
include_once("sections/header.php");


$arrayOfBooks = array_map('str_getcsv', file('books.csv'));
$cartlabels = array("New", "Used", "Rental", "eBook");


$physicalFlag = 0;
$finalSales = 0;

$booksDataArray = array();
$order = array();
$orderKeys = array();

foreach ($_SESSION["cart"] as $orderLine){
    $order[] = explode(",", $orderLine);
}

foreach ($arrayOfBooks as $bookLine){

    foreach ($order as $data){
        if($bookLine[0] == $data[0]){

            $orderKeys[] = count($booksDataArray);
            $booksDataArray[] = $bookLine;


        }
    }
}

?>



<div id="content">

<h1>Your Order Summary</h1>

<div class="col-xs-8 col-xs-offset-2">

<table class="table table-striped">
<tr>
    <th><!-- Image Goes Here --></th>
    <th>Textbook Title</th>
    <th>Type</th>
    <th>Quantity</th>
    <th>Per Unit Cost</th>
    <th>Total Cost</th>

</tr>

<?php

for($i = 0; $i < count($order); $i++){

    ?>

<tr>
    <td><img src="images/<?php echo $order[$i][0]; ?>.jpg" class="checkoutIcon"></td>
```

```
    <td class="text-left"><?php echo $booksDataArray[$orderKeys[$i]][1]; ?></td>
    <td class="text-left"><?php echo $cartlabels[$order[$i][1]]; ?></td>
    <td class="text-left"><?php echo $order[$i][2]; ?></td>
    <td class="text-left"><?php echo money_format("$%!n", $booksDataArray[$orderKeys[$i]][13 +
    $order[$i][1]]); ?></td>
    <td class="text-left"><?php echo money_format("$%!n", $order[$i][2] *
    $booksDataArray[$orderKeys[$i]][13 + $order[$i][1]]); ?></td>

    <?php

        if($order[$i][1] != 3){ //If we've ordered at least one physical book
            $physicalFlag = 1;
        }
        $finalSales += $order[$i][2] * $booksDataArray[$orderKeys[$i]][13 + $order[$i][1]];
    ?>


</tr>

<?php

}


?>

</table>




<table class="table">
<tr>
    <td class="largerFont text-left"><b>Subtotal</b></td>
    <td class="largerFont text-center"><?php echo money_format("$%!n", $finalSales); ?></td>
</tr>
<tr>
    <td class="largerFont text-left"><b>GA Sales Tax</b></td>
    <td class="largerFont text-center">
    <?php
        echo "+" . money_format("$%!n", $finalSales * 0.07);
        $finalSales *= 1.07;
    ?>
    </td>
</tr>
<?php
if($physicalFlag == 1){
    $finalSales += 14.95;
?>
<tr>
    <td class="largerFont text-left"><b>Shipping & Handling</b></td>
    <td class="largerFont text-center">+$14.95</td>
</tr>
<?php
}
?>

<tr>
    <td> </td>
    <td> </td>
<tr>

<tr>
    <td class="largerFont text-left"><b>Total Cost</b></td>
    <td class="largerFont text-center"><b><?php echo money_format("$%!n", $finalSales);
```

```
    ?></b></td>
    <?php $_SESSION["ordertotal"] = $finalSales; ?>
</tr>

</table>



<a href="shipping" class="btn btn-info btn-lg">Pay Now</a>

</div>


</div>




<?php

//Storing this here since I'm lazy


include_once("sections/footer.php");
?>
```

```php
<?php
session_start();
setlocale(LC_MONETARY,"en_US");

$arrayOfBooks = array_map('str_getcsv', file('books.csv'));
$cartlabels = array("New", "Used", "Rental", "eBook");


$physicalFlag = 0;
$rentalFlag = 0;
$finalSales = 0;

$booksDataArray = array();
$finalorder = array();
$orderKeys = array();

if(count($_SESSION["cart"]) > 0){
    foreach ($_SESSION["cart"] as $orderLine){
        $finalorder[] = explode(",", $orderLine);
    }
}


foreach ($arrayOfBooks as $bookLine){

    foreach ($finalorder as $data){
        if($bookLine[0] == $data[0]){

            $orderKeys[] = count($booksDataArray);
            $booksDataArray[] = $bookLine;


        }
    }
}

//Copy all session data here
$userInfo = $_SESSION;

//Clearing out session


$pagetitle = "KSU Bookstore - Confirmation";
$nocart = 1;
include_once("sections/header.php");

?>



<div id="content">

<?php
    include_once("code/displayreceipt.php");
?>

</div>




<?php

$_SESSION = array();
```

```php
include_once("sections/footer.php");
?>
```

```php
<?php


if(!ISSET($_GET["id"])){
    header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] . '/nmurphy1/bookstore/');
    die("Invalid ID");
}

$pagetitle = "KSU Bookstore";
include_once("sections/header.php");

//Get all matching items from shopping cart
$order = array();

if(count($_SESSION["cart"]) > 0){
    foreach ($_SESSION["cart"] as $orderLine){
        if (strpos($orderLine, $_GET["id"]) !== false) {
            $order[] = explode(",", $orderLine);
        }
    }
}

?>



<div id="content">


<?php


function processQuantity($textbookQuantity){

    if(strtoupper($textbookQuantity) == "INF"){
        return "UNLIMITED";
    }

    if($textbookQuantity <= 0){
        return "OUT OF STOCK";
    }

    return $textbookQuantity;

}


$arrayOfBooks = array();


//Super-useful - handles commas in quoted sections with no additional parsing
$arrayOfBooks = array_map('str_getcsv', file('books.csv'));

$imageArray = array();
$titleArray = array();
$authorArray = array();
$description = "";
$associated_courses = "";
$associated_professor="";

//New, Used, Rental, eBook
$quantities = array();
$prices = array();
```

```
//Loop through each line of the CSV
foreach ($arrayOfBooks as $bookLine){

    //2nd column contains the title - if our search term is in it, we extract title, image, and
    description
    if(strpos($bookLine[0], $_GET["id"]) !== false){
        $titleArray[] = "{$bookLine[1]}";
        $imageArray[] = "<img class='detailIcon' src='images/{$bookLine[0]}.jpg'>";
        $description = "<p>" . utf8_encode($bookLine[17]) . "</p>";
        $associated_courses = "{$bookLine[4]}";
        $associated_professor = "{$bookLine[6]}";
        $requiredRecommended = "{$bookLine[8]}";
        $courseSection = "{$bookLine[5]}";
        $ISBN = "{$bookLine[0]}";

        if(strtoupper($bookLine[2]) != "BLANK"){
            $authorArray[] = "by " . $bookLine[2];
        }
        else{
            $authorArray[] = " ";
        }


        for($i = 0; $i < 4; $i++){
            $quantities[] = $bookLine[9 + $i];
            $prices[] = $bookLine[13 + $i];
        }


        if($bookLine[12] == 999999){
            $quantityString .= "<b>Available</b>";
        }
        else{
            $quantityString .= "<b>Not Available</b>";
        }


        if(strtoupper($requiredRecommended) == "REQUIRED"){
            $requiredRecommended = "<b class='largerFont'>$requiredRecommended</b>";
        }
        else{
            $requiredRecommended = "<i>$requiredRecommended</i>";
        }
    }

}


//If we found something
if(count($titleArray) > 0){

    echo "<hr class='clearfix'>";

    //Display results
    for($i = 0; $i < count($titleArray); $i++){

        ?>

        <div class="row">
            <div class="col-xs-10 col-xs-offset-1 text-left">
                <h2><?php echo $titleArray[$i]; ?></h2>
                <?php echo $authorArray[$i]; ?>
            </div>

        </div>
```

```
        <div class="row">
            <div class="col-xs-3 col-xs-offset-1">
                <?php echo $imageArray[$i];?>

                <p class="text-left">
                    <b>ISBN</b>: <br>
                    <?php echo $ISBN; ?>
                    <br>
                </p>
                <h3 class="text-left">Related Courses:</h3>

                <p class="text-left">
                    <?php
                    echo $requiredRecommended . " for Dr. $associated_professor" . "'s
                    <u>$associated_courses</u>, Section $courseSection";
                    ?>
                    <br>
                </p>


            </div>

            <div class="col-xs-6 text-left">

            <div class="panel panel-default">

                <div class="panel-heading">
                    <h2>Select a Textbook Format</h2>
                </div>

                <div class="panel-body">

                <form method="post" action="code/updatecart">

                <?php
                    setlocale(LC_MONETARY,"en_US");
                    $labels = array("New", "Used", "Rental", "eBook");
                    for($i = 0; $i < 4; $i++){


                ?>

                        <div class="col-xs-3 radio">
                        <?php
                            $showCheckbox = 0;
                            foreach($order as $checkQuantity){
                                if($checkQuantity[1] == $i){
                                    $showCheckbox = $checkQuantity[2];
                                }
                            }

                            if($showCheckbox < $quantities[$i]){
                                echo "<input type='radio' name='booktype' value='{$i}'
                                required> ";
                            }

                        ?>
                        <?php echo $labels[$i]; ?>
                        </div>

                        <div class="col-xs-8 text-right">
                        <?php echo money_format("$%!n", $prices[$i]); ?>
                        </div>
```

```
                    <br class="clearfix">
                    <hr>
                <?php
                    }
                ?>


                <input type="hidden" name="action" value="add">
                <input type="hidden" name="id" value="<?php echo $_GET['id']; ?>">
                <input type="submit" class="btn btn-default" value="Add to Cart">

                </form>

                </div>


            </div>




                <?php echo '<p>Description:</p>'.$description; ?>


            </div>

        </div>
        <hr>

        <?php

    }
}
else{
    echo "No Results Found.";
    echo "<hr class='clearfix'>";

}


?>









</div>




<?php
include_once("sections/footer.php");
?>
```

```php
<?php

session_start();


$database = "books.csv";
if(!file_exists($database)){
    $database = "../books.csv";
}

    $arrayOfBooks = array_map('str_getcsv', file($database));



$cartlabels = array("New", "Used", "Rental", "eBook");

$order = array();
$totalItems = 0;

if(count($_SESSION["cart"]) > 0 && !ISSET($nocart)){
    foreach ($_SESSION["cart"] as $orderLine){
        $order[] = explode(",", $orderLine);
        $totalItems += $order[count($order) - 1][2];
    }
}

?>


<button class="btn btn-info btn-lg text-center dropdown-toggle" data-toggle='dropdown'>
    Shopping Cart (<?php echo $totalItems; ?>) <span class="glyphicon
    glyphicon-chevron-down"></span>
</button>
    <div class='dropdown-menu dropdown-menu-right'>

<table class="table table-striped">

    <tr>
        <th>
        Item
        </th>
        <th>
        Type
        </th>
        <th>
        Quantity
        </th>

    </tr>


<?php


$cart_count = 0;
foreach ($order as $cartValues){

    echo "<tr>";



    $bookArray = array();
    foreach ($arrayOfBooks as $bookLine){

        if($bookLine[0] == $cartValues[0]){
```

```
            $bookArray = $bookLine;
        }

    }


    echo "<td>{$bookArray[1]}</td>";

    echo "<td>{$cartlabels[$cartValues[1]]}</td>";

    echo "<td>";
        echo "<input type='number' id='cartvalue$cart_count' value='{$cartValues[2]}' min='0'
        max='{$bookArray[9 + $cartValues[1]]}'>";
        echo "<input type='hidden' id='carttype$cart_count' value='{$cartValues[1]}'>";
        echo "<input type='hidden' id='cartid$cart_count' value='{$cartValues[0]}'>";
    echo "</td>";



    echo "</tr>";

    $cart_count++;
}


?>

        </td>
    </tr>


    <tr>

<?php
    if(count($_SESSION["cart"]) > 0){
?>
        <td colspan="2" class="text-center">
            <button id="updatecart" class="btn btn-default">Update Cart</button>
        </td>

        <td>
            <a class="btn btn-default" href="checkout">Checkout</a>
        </td>

        </tr>
        <tr>
            <td colspan="3">
                <a href="shoppingcart" class="largerFont"><b>Go to your Shopping Cart  </b></a>
            </td>
        </tr>

<?php
}
else{
    echo "<td colspan='3' class='text-center'><b>Nothing in Cart!</b></td>";

}
?>

    </tr>
</table>

</div>
```

```
<div class="col-xs-8 col-xs-offset-2">

<h1 class="text-left">Your Order has been placed!</h1>

<?php
    $cartlabels = array("New", "Used", "Rental", "eBook");
?>

<div class="row">
    <div class="col-xs-6 text-left">
        Congratulations! You just spent a lot of money on textbooks!
    </div>
    <div class="col-xs-6 text-center">
        Date of Purchase: <?php echo date("d/m/Y"); ?>
    </div>
</div>

<br class="clearfix">

<div class="row">
    <div class="col-xs-6 text-left">
        <p>Your receipt has been emailed to you.</p>
        <p>Shipping:</p>
        <?php
            echo $userInfo["shipping-billing"]["firstname_s"] . " " .
            $userInfo["shipping-billing"]["lastname_s"] . "<br>";
            echo $userInfo["shipping-billing"]["street_s"] . "<br>";
            echo $userInfo["shipping-billing"]["city_s"] . ", " .
            $userInfo["shipping-billing"]["state_s"] . " " .
            $userInfo["shipping-billing"]["zipcode_s"] . "<br>";

        ?>

    </div>
    <div class="col-xs-6 text-center">

    </div>
</div>

<table class="table table-striped">
<tr>
    <th>Item</th>
    <th>Type</th>
    <th>Quantity</th>
    <th>Price</th>

</tr>

<?php

for($i = 0; $i < count($finalorder); $i++){

    ?>

<tr>
    <td class="text-left"><?php echo $booksDataArray[$orderKeys[$i]][1]; ?></td>
    <td class="text-left"><?php echo $cartlabels[$finalorder[$i][1]]; ?></td>
    <td class="text-left"><?php echo $finalorder[$i][2]; ?></td>
    <td class="text-left"><?php echo money_format("$%!n", $finalorder[$i][2] *
    $booksDataArray[$orderKeys[$i]][13 + $finalorder[$i][1]]); ?></td>

    <?php

        if($finalorder[$i][1] != 3){ //If we've ordered at least one physical book
            $physicalFlag = 1;
```

```
        }
        if($finalorder[$i][1] == 2){ //Rental
            $rentalFlag = 1;
        }
        if($finalorder[$i][1] == 3){ //Ebook
            $ebookderpFlag = 1;
        }
        $finalSales += $finalorder[$i][2] * $booksDataArray[$orderKeys[$i]][13 +
        $finalorder[$i][1]];
    ?>


</tr>

<?php

}


?>

</table>




<div class="col-xs-8 text-left">
Rental Due Date: <?php
if($rentalFlag == 1){
    echo "14/08/2017";
}
else{
    echo "N/A";
}

?>
<br><br>
EBook Download Link:
<?php
    if($ebookderpFlag == 1){
        $GUID = sprintf('%04X%04X-%04X-%04X-%04X-%04X%04X%04X', mt_rand(0, 65535), mt_rand(0,
        65535), mt_rand(0, 65535), mt_rand(16384, 20479), mt_rand(32768, 49151), mt_rand(0,
        65535), mt_rand(0, 65535), mt_rand(0, 65535));

        echo "<a
        href=\"http://www.hhcard.org/nmurphy1/bookstore/ebookLink?GUID=$GUID\">Download your
        ebook here!</a>";
    } else {
        echo 'N/A';
    }
?>
</div>


<div class="col-xs-4">
<table class="table">
<tr>
    <td class="largerFont text-left"><b>Subtotal</b></td>
    <td class="largerFont text-center"><?php echo money_format("$%!n", $finalSales); ?></td>
</tr>
<tr>
    <td class="largerFont text-left"><b>Tax (7%)</b></td>
    <td class="largerFont text-center">
    <?php
```

```
        echo "+" . money_format("$%!n", $finalSales * 0.07);
        $finalSales *= 1.07;
    ?>
    </td>
</tr>
<?php
if($physicalFlag == 1){
    $finalSales += 14.99;
?>
<tr>
    <td class="largerFont text-left"><b>Shipping</b></td>
    <td class="largerFont text-center">+$14.99</td>
</tr>
<?php
}
?>

<tr>
    <td> </td>
    <td> </td>
<tr>

<tr>
    <td class="largerFont text-left"><b class="largerFont">Total:</b></td>
    <td class="largerFont text-center"><b><?php echo money_format("$%!n", $finalSales);
    ?></b></td>
</tr>

</table>
</div>

</div>
```

```php
<?php
    $pagetitle = "Ebook Download";
    include_once("sections/header.php");
?>
<?php

/*
function getGUID(){ //Creates GUID
    if (function_exists('com_create_guid')){
        return com_create_guid();
    }else{
        mt_srand((double)microtime()*10000);//optional for php 4.2.0 and up.
        $charid = strtoupper(md5(uniqid(rand(), true)));
        $hyphen = chr(45);// "-"
        $uuid = substr($charid, 0, 8).$hyphen
            .substr($charid, 8, 4).$hyphen
            .substr($charid,12, 4).$hyphen
            .substr($charid,16, 4).$hyphen
            .substr($charid,20,12);
        return $uuid;
    }
}
*/
?>
    <div class="col-xs-8 col-xs-offset-2 text-center">
        <?php
            $GUID = $_GET["GUID"];
        ?>
        <br><br>
        <h1>Your Ebook has been downloaded!<br>GUID Number: <?php echo $GUID; ?></h1>
    </div>



<?php
    include_once("sections/footer.php");
?>
```

```
<?php
$pagetitle = "KSU Bookstore - FAQs";
include_once("sections/header.php");
?>



<div id="content" class="text-left">



<div class="col-xs-10 col-xs-offset-1">

<h1>FAQ/HELP</h1>

<div class="panel panel-default">
  <div class="panel-heading">Can I buy books on this website?</div>
</div>

Yes.

<hr>



<div class="panel panel-default">
  <div class="panel-heading">How do I search for books?</div>
</div>

Use the search box on the homepage

<hr>




<div class="panel panel-default">
  <div class="panel-heading">Can I buy every version of every book on the website and make it
  completely sold out?</div>
</div>

Yes.


<hr>



<div class="panel panel-default">
  <div class="panel-heading">What if I don't have any money for the books during checkout?</div>
</div>

Then you can't buy books.




</div>

</div>
```

```php
<?php
include_once("sections/footer.php");
?>
```

```
<script src="js/jquery.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<link rel="stylesheet" href="css/bootstrap.min.css">


<script>

$( document ).ready(function() {
        $("#updatecart").click(function(){
            var i = 0;
        var updateString = "";
        var separator = "";

        while($("#cartvalue" + i).length != 0){

            if($("#cartvalue" + i).val() > 0){

                updateString += separator + $("#cartid" + i).val() + "," + $("#carttype" +
                i).val() + "," + $("#cartvalue" + i).val();
                separator = ";";
            }

            i++;
        }

            $.ajax({
                    url: 'code/updatequantity',
                    type: 'post',
                    data: { 'shoppingcart': updateString},
                    success: function (data, status)
                    {
                    if (status == "success")
                    {
                        document.getElementById("shopping_cart_section").innerHTML = data;
                location.reload();
                    }
                    },
                    error: function (xhr, desc, err){}
            }); // end ajax call


    });
});

</script>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en-US">
<head>
<meta charset="UTF-8">
<title><?php echo $pagetitle; ?></title>
<link rel="icon" type="image/png" href="/nmurphy1/bookstore/favicon.ico">
<link href="css/bootstrap.min.css" type="text/css" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Nunito:400,400i,700" rel="stylesheet">
<link href="https://fonts.googleapis.com/css?family=Roboto+Condensed:700" rel="stylesheet">
<link href="css/bookstore.css" type="text/css" rel="stylesheet">
</head>

<body class="text-center">



<div class="row" id="header">
    <div class="col-xs-4">
    <p class="tab">
        <a href="/nmurphy1/bookstore/">Home</a>
    </p>
    <p class="tab">
        <a href="faqs">FAQs</a>
    </p>
    </div>


    <div class="col-xs-4 text-center">
        <img src="images/ksu.png" id="ksulogo" alt="KSU Logo">
    </div>


    <div class="col-xs-4 text-right dropdown" id="shopping_cart_section">
        <?php include_once("code/displaycart.php"); ?>
    </div>

</div>
```

```
<?php
$pagetitle = "KSU Bookstore";
include_once("sections/header.php");
?>




<div id="content" class="text-center spaceFromTop"">

<h1>KSU Bookstore</h1>

<hr>

<form method="get" action="results" class="col-xs-4 col-xs-offset-4">

    <div class="form-group text-left col-md-9">
        <input name="search" class="form-control glyphicon" type="search" placeholder="&#57347;
        Keyword..."/>
    </div>
    <div class="form-group text-center col-md-3">
        <input type="submit" class="btn btn-default btn-lg" value="Search">
    </div>


    <div class="form-group text-left col-md-3">
        <label class="PhilosopherBoldItalic textSize5">Search by:</label>
    </div>
    <div class="form-group text-left col-md-6">
        <select class="form-control " name="type" id="type">
            <option value="1">Title</option>
            <option value="2">Author</option>
            <option value="17">Description</option>
            <option value="4">Course</option>
            <option value="6">Professor</option>
            <option value="0">ISBN</option>
        </select>
    </div>




<!--
    <div class="form-group text-left col-md-2">
        <label class="PhilosopherBoldItalic textSize5">Professor:</label>
    </div>
    <div class="form-group text-left col-md-4">
        <select class="form-control" name="Professor">
        <option> </option>
<?php
    $arrayOfBooks = array_map('str_getcsv', file('books.csv'));
    foreach($arrayOfBooks as $book){
        $professor[] = $book[6];
    }

    $professor = array_unique($professor);
    sort($professor);
    foreach($professor as $thisProfessor){
        echo '<option>'. $thisProfessor .'</option>';
    }
?>
        </select>
    </div>
```

```
    <div class="form-group text-left col-md-2">
        <label class="PhilosopherBoldItalic textSize5">Course:</label>
    </div>
    <div class="form-group text-left col-md-4">
        <select class="form-control" name="Course">
        <option> </option>
<?php
    $arrayOfBooks = array_map('str_getcsv', file('books.csv'));
    foreach($arrayOfBooks as $book){
        $course[] = $book[4];
    }

    $course = array_unique($course);
    sort($course);
    foreach($course as $thisCourse){
        echo '<option>'. $thisCourse .'</option>';
    }
?>
        </select>
    </div>

-->


</form>









</div>




<?php
include_once("sections/footer.php");
?>
```

```php
<?php


session_start();


$myfile = fopen("../data/students.txt", "r") or die("Unable to open file!");


while(!feof($myfile)) {
  $studentsArray[] = explode(' ', fgets($myfile));
}
fclose($myfile);


if(ISSET($_POST["finaid_user"])){


    for($j = 0; $j < count($studentsArray); $j++){

        if(strcmp($studentsArray[$j][2], $_POST["finaid_user"]) == 0 &&
        strcmp($studentsArray[$j][3], $_POST["finaid_password"]) == 0){
            $_SESSION["finaid_user"] = $_POST["finaid_user"];
            $_SESSION["finaid_password"] = $_POST["finaid_password"];
            header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
            '/nmurphy1/bookstore/payment?finaid=1');
            die("");
        }
    }


    header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
    '/nmurphy1/bookstore/payment?finaid=2');
    die("");


}
else{

    header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] . '/nmurphy1/bookstore/payment');
    die("Invalid");

}







?>
```

```php
<?php
session_start();
setlocale(LC_MONETARY,"en_US");

$pagetitle = "KSU Bookstore - Payment";
include_once("sections/header.php");


//Save Shipping info
if(ISSET($_POST["firstname_s"])){
    $_SESSION['shipping-billing'] = $_POST;
}

$active = "";
if(ISSET($_GET["finaid"])){
    $active = "active ";
}

?>



<div id="content">

<h1>Step 2: Payment Options</h1>

<hr>

<div class="row">

<ul class="nav nav-tabs">
  <li class="col-md-2 col-md-offset-3 col-xs-12"><button class="btn btn-warning btn-lg"
  data-toggle="tab" href="#paypal">PayPal</button></li>
  <li class="col-md-2 col-xs-12"><button class="btn btn-secondary btn-lg" data-toggle="tab"
  href="#creditcard">Credit Card</button></li>
  <li class="<?php echo $active; ?>col-md-2 col-xs-12"><button class="btn btn-danger btn-lg"
  data-toggle="tab" href="#financialaid">Financial Aid</button></li>
</ul>

</div>

</hr>

<div class="tab-content">
  <div id="paypal" class="tab-pane fade col-xs-6 col-xs-offset-3">

    <br><br>
    <h2>PayPal</h2>

    <form method="post" action="code/receipt">
        <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">User Name:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="paypaluser" title="Enter your PayPal user
        name." required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Password:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="password" class="form-control" name="paypalpassword" title="Enter your
        PayPal password." required>
    </div>
```

```
    <div class="form-group text-center col-md-12">
        <input type="submit" class="btn btn-info btn-lg" value="Pay">
    </div>
    </form>









</div>
<div id="creditcard" class="tab-pane fade col-xs-6 col-xs-offset-3">


    <br><br>
    <h2>Visa or MasterCard</h2>

    <form method="post" action="code/receipt">
    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Name on Card:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="nameoncard" title="Name as it appears on
        the card.")"  required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Card Number:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" maxlength="16" name="cardnumber"
        pattern="[0-9]{16}" title="Credit card must be 16 numbers."
        onkeyup="this.value=this.value.replace(/[^\d]/g,'')"  required>
    </div>


    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Exp. Date:</label>
    </div>
    <div class="form-group text-left col-md-4">
        <select type="text" class="form-control" style="max-width:5em !important;
        display:inline-block !important" name="expdate1" id="expdate1"
        onchange="checkExpiration()" required>
            <option> </option>
            <?php
            for($i = 1; $i <= 12; $i++){
                $display = $i;
                if($i < 10){
                    $display = "0" . $i;
                }
                $selected = "";
                if($i == date('n')){
                    $selected = " selected";
                }
                echo "<option value='$display'$selected>$display</option>";

            }
            ?>
        </select>
        /
        <select type="text" class="form-control" style="max-width:5em !important;
        display:inline-block !important" name="expdate2" id="expdate2"
        onchange="checkExpiration()" required>
```

```
            <option value='17'>17</option>
            <?php
            for($i = 18; $i <= 32; $i++){
                echo "<option value='$i'>$i</option>";

            }
            ?>
        </select>
    </div>
    <div class="form-group text-center col-md-2">
        <label class="PhilosopherBoldItalic textSize5">CVC:</label>
    </div>
    <div class="form-group text-left col-md-2">
        <input type="text" maxlength="3" class="form-control" name="cvc" pattern="[0-9]{3}"
        title="CVC must be 3 numbers" onkeyup="this.value=this.value.replace(/[^\d]/g,'')"
        required>
    </div>

    <div class="form-group text-center col-md-12">
        <input type="submit" class="btn btn-info btn-lg" value="Pay">
    </div>
    </form>



  </div>
  <div id="financialaid" class="tab-pane fade in <?php echo $active; ?>col-xs-6 col-xs-offset-3">

    <br><br>
    <?php
        if($_GET["finaid"] > 1){
            echo "<b style='color:red; font-size:larger'>Invalid User Name and/or Password!</b>";
        }

    ?>
    <?php include_once("code/checkfinaid.php"); ?>


  </div>
</div>



</div>


<script>

function checkExpiration(){

    var first = document.getElementById("expdate1");
    var second = document.getElementById("expdate2");

    if(second.value == 17){
        var d = new Date();
        var n = d.getMonth() + 1;
        if(first.value <= n){
            var n2 = n + 1;
            if(n2 < 10){
                n2 = "0" + n2;
            }
            first.value = n2;
        }
    }
```

```
}


</script>

<?php
include_once("sections/footer.php");
?>
```

```php
<?php
session_start();

//Subtract from financial aid
if(ISSET($_SESSION["finaid_user"])){


    $myfile = fopen("../data/students.txt", "r") or die("Unable to open file!");

    while(!feof($myfile)) {
        $studentsArray[] = explode(' ', fgets($myfile));
    }
    fclose($myfile);


    $newStudentRecords = "";
    for($i = 0; $i < count($studentsArray); $i++){
        if(strcmp($studentsArray[$i][2], $_SESSION["finaid_user"]) == 0){
            $studentsArray[$i][4] = $studentsArray[$i][4] - $_SESSION["ordertotal"];
        }


    $newStudentRecords .= $studentsArray[$i][0] . " " . $studentsArray[$i][1] . " " .
    $studentsArray[$i][2] . " " . $studentsArray[$i][3] . " " . trim($studentsArray[$i][4]) .
    "\n";
    }

    //write
    $mywfile = fopen("../data/students.txt", "w") or die("Unable to open file!");
    fwrite($mywfile, trim($newStudentRecords));
    fclose($mywfile);
}


//=============================================
//Subtract from Inventory

$arrayOfBooks = array_map('str_getcsv', file('../books.csv'));

foreach ($_SESSION["cart"] as $orderLine){
    $order[] = explode(",", $orderLine);
}


//Updating
for($i = 0; $i < count($arrayOfBooks); $i++){

    foreach($order as $orderLine){
        if($orderLine[0] == $arrayOfBooks[$i][0] && $orderLine[1] != 3){ //Don't update eBooks
        quantity

            //Removing quantity from that section
            $arrayOfBooks[$i][9 + $orderLine[1]] = $arrayOfBooks[$i][9 + $orderLine[1]] -
            $orderLine[2];

        }
    }
}


//Actually writing
$csvfile = fopen("../books.csv","w");
foreach ($arrayOfBooks as $bookLine) {
       fputcsv($csvfile, $bookLine);
 }
```

```php
fclose($csvfile);




//================================================
//Send Email

$email_from = "spsu_bookstore@spsu.edu";
$email_to = $_SESSION["shipping-billing"]["email_s"];
$email_subject = "SPSU Bookstore - Group Project Test";



//=====================================
$cartlabels = array("New", "Used", "Rental", "eBook");

$physicalFlag = 0;
$rentalFlag = 0;
$finalSales = 0;

$booksDataArray = array();
$finalorder = array();
$orderKeys = array();

foreach ($_SESSION["cart"] as $orderLine){
    $finalorder[] = explode(",", $orderLine);
}


foreach ($arrayOfBooks as $bookLine){
    foreach ($finalorder as $data){
        if($bookLine[0] == $data[0]){
            $orderKeys[] = count($booksDataArray);
            $booksDataArray[] = $bookLine;
        }
    }
}
//Copy all session data here
$userInfo = $_SESSION;
//=====================================


//Save receipt data to email body
ob_start();
include 'displayreceipt.php';
$body = ob_get_contents();
ob_end_clean();


$headers = "MIME-Version: 1.0" . "\r\n";
$headers .= "Content-type:text/html;charset=UTF-8" . "\r\n";
$headers .= 'From: '.$email_from."\r\n".
'Reply-To: '.$email_from."\r\n" .
'X-Mailer: PHP/' . phpversion();

@mail($email_to, $email_subject, $body, $headers);



//================================================
//Redirect


header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] . '/nmurphy1/bookstore/confirmation');
```

```
die("Invalid");

?>
```

```
<?php
header('Content-Type: text/html; charset=UTF-8');
setlocale(LC_MONETARY,"en_US");

$type = 1;
if(ISSET($_GET["type"])){
    $type = $_GET["type"];
}

$pagetitle = "KSU Bookstore";
include_once("sections/header.php");
?>



<div id="content">
<br>

<form method="get" action="results">

    <div class="text-left row">
    <div class="col-xs-3 col-xs-offset-1">
    <input name="search" class="form-control glyphicon" type="search" placeholder="&#57347;
    Search Again"/>
    </div>

    <div class="col-xs-2">
    <input type="submit" class="btn btn-default" value="Search">
    </div>

    <div class="col-xs-2 col-xs-offset-4 text-right">
        <select name="sort" class="form-control" onchange="sortResults(this.value)">
            <option value="" disabled selected>[Sort By]</option>
            <option value="0">A - Z</option>
            <option value="1">Z - A</option>
            <option value="2">Price, Low to High</option>
            <option value="3">Price, High to Low</option>
        </select>
    </div>
    </div>

    <br class="clearfix">

    <div class="text-left row">

        <div class="form-group text-left col-xs-1 col-xs-offset-1">
            <label class="PhilosopherBoldItalic textSize5">Search by:</label>
        </div>
        <div class="form-group text-left col-xs-2">
            <select class="form-control" name="type" id="type">
            <?php
                $searchlabels = array("Title", "Author", "Description", "Course", "Professor",
                "ISBN");
                $searchid = array(1, 2, 17, 4, 6, 0);

                for($j = 0; $j < count($searchlabels); $j++){

                    $selected = "";
                    if($type == $searchid[$j]){
                        $selected = " selected";
                    }
                    echo "<option value='" . $searchid[$j]. "'$selected>" . $searchlabels[$j] .
                    "</option>\n";
                }
            ?>
```

```
            </select>
        </div>

    </div>



</form>



<?php


function processQuantity($textbookQuantity){

    if(strtoupper($textbookQuantity) == "INF"){
        return "UNLIMITED";
    }

    if($textbookQuantity <= 0){
        return "OUT OF STOCK";
    }

    return $textbookQuantity;

}


$arrayOfBooks = array();


//Super-useful - handles commas in quoted sections with no additional parsing
$arrayOfBooks = array_map('str_getcsv', file('books.csv'));

$resultsArray = array();

//Loop through each line of the CSV
foreach ($arrayOfBooks as $bookLine){

    //2nd column contains the title - if our search term is in it, we extract title, image, and
    description
    if( stripos($bookLine[$type], $_GET["search"]) !== false){

        $entry = array();
        $highestPrice = 0;
        $lowestPrice = 10000;

        $entry[0] = $bookLine[0]; //ID
        $entry[1] = "{$bookLine[1]}"; //Title
        $entry[2] = " <b>ISBN: </b>{$bookLine[0]}";
        $entry[3] = "<img class='searchIcon' src='images/{$bookLine[0]}.jpg'
        title='{$bookLine[1]}'>";

        if(strlen($bookLine[17]) > 350){
            $bookLine[17] = substr($bookLine[17], 0, 350) . "[<i>...</i>]";
        }
        $entry[4] = "<p>" . utf8_encode($bookLine[17]) . "</p>"; //Description

        if(strtoupper($bookLine[2]) != "BLANK"){
            $entry[5] = "by " . $bookLine[2];
        }
        else{
            $entry[5] = " ";
        }
```

```
    $quantityString = "New: ";
    $quantityString .= "<b>" . processQuantity($bookLine[9]) . "</b>";
    if(processQuantity($bookLine[9]) != "OUT OF STOCK" && $bookLine[9] > 0){
        $quantityString .= " for <u> " . money_format("$%!n", $bookLine[13]) . "</u> | ";
    }
    else{
        $quantityString .= " | ";
    }

    $quantityString .= "Used: ";
    $quantityString .= "<b>" . processQuantity($bookLine[10]) . "</b>";
    if(processQuantity($bookLine[10]) != "OUT OF STOCK" && $bookLine[10] > 0){
        $quantityString .= " for <u> " . money_format("$%!n", $bookLine[14]) . "</u> | ";
    }
    else{
        $quantityString .= " | ";
    }

    $quantityString .= "Rental: ";
    $quantityString .= "<b>" . processQuantity($bookLine[11]) . "</b>";
    if(processQuantity($bookLine[11]) != "OUT OF STOCK" && $bookLine[11] > 0){
        $quantityString .= " for <u> " . money_format("$%!n", $bookLine[15]) . "</u> | ";
    }
    else{
        $quantityString .= " | ";
    }

    $quantityString .= "eBook: ";
    if($bookLine[12] > 0){
        $quantityString .= "<b>Available</b> for <u> " . money_format("$%!n",
        $bookLine[16]) . "</u>";
    }
    else{
        $quantityString .= "<b>Not Available</b>";
    }

    $entry[6] = "<span class='largerFont'>{$quantityString}</span>";


    //Get the highest/lowest price for sorting purposes
    for($h = 13; $h < 17; $h++){
        if($bookLine[$h - 4] > 0){ //Don't show out-of-stock prices
        if($bookLine[$h] > $highestPrice){
            $highestPrice = $bookLine[$h];
        }
        if($bookLine[$h] < $lowestPrice){
            $lowestPrice = $bookLine[$h];
        }
        }
    }
    $entry[7] = $highestPrice;
    $entry[8] = $lowestPrice;
    if(stripos($bookLine[8], "Required") !== false){
        $entry[9] = "<b style='font-size:larger'>" . $bookLine[8] . "</b>"; //Recommended
    }
    else{
        $entry[9] = "<i>" . $bookLine[8] . "</i>";
    }

    $entry[9] .= " for Dr. {$bookLine[6]}'s <u>{$bookLine[4]}</u>, section {$bookLine[5]}";


    $resultsArray[] = $entry;
}
```

```
}


//If we found something
if(count($resultsArray) > 0){



    //TODO: Make more flexible
    if($_GET["sort"] == 0){ //A-Z

        $sortByTextbook = array();
        foreach ($resultsArray as $textbooks) {
            $sortByTextbook[] = $textbooks[1];
        }

        array_multisort($resultsArray, SORT_ASC, SORT_LOCALE_STRING, $sortByTextbook);
    }
    else if($_GET["sort"] == 1){ //Z-A

        $sortByTextbook = array();
        foreach ($resultsArray as $textbooks) {
            $sortByTextbook[] = $textbooks[1];
        }

        array_multisort($resultsArray, SORT_DESC, SORT_LOCALE_STRING, $sortByTextbook);
        $resultsArray = array_reverse($resultsArray);
    }
    else if($_GET["sort"] == 2){ //Lowest to Highest Price

        $sortByPrice = array();
        foreach ($resultsArray as $prices) {
            $sortByPrice[] = $prices[8];
        }

        array_multisort($resultsArray, SORT_ASC, SORT_NUMERIC, $sortByPrice);
    }
    else if($_GET["sort"] == 3){ //Highest to Lowest Price

        $sortByPrice = array();
        foreach ($resultsArray as $prices) {
            $sortByPrice[] = $prices[7];
        }


        array_multisort($resultsArray, SORT_ASC, SORT_NUMERIC, $sortByPrice);
        $resultsArray = array_reverse($resultsArray);

    }







    echo "We found <b>" . count($resultsArray) . "</b> result(s):";
    echo "<hr class='clearfix'>";

    //Display results
    for($i = 0; $i < count($resultsArray); $i++){

        ?>
```

```
        <div class="row">
            <div class="col-xs-2">
                <a href="details?id=<?php echo $resultsArray[$i][0]; ?>"><?php echo
                $resultsArray[$i][3];?></a>
            </div>

            <div class="col-xs-9 text-left">
                <h2><a href="details?id=<?php echo $resultsArray[$i][0]; ?>"><?php echo
                $resultsArray[$i][1]; ?></a></h2>
                <?php echo $resultsArray[$i][5]; ?>
                <?php echo $resultsArray[$i][2]; ?>
                <br><br>

                <?php
                echo $resultsArray[$i][4];
                ?>
                <br>
                <?php echo $resultsArray[$i][9]; ?>
                <hr>
                <?php echo $resultsArray[$i][6]; ?>
            </div>

        </div>
        <hr>

        <?php

    }
}
else{
    echo "No Results Found.";
    echo "<hr class='clearfix'>";

}


?>









</div>


<script>
function sortResults(sortvalue){

    window.location.href = "results?search=<?php echo str_replace(" ", "+", $_GET["search"]);
    ?>&type=<?php echo $type; ?>&sort=" + sortvalue;
}
</script>

<?php



include_once("sections/footer.php");
?>
```

```
<?php
session_start();
setlocale(LC_MONETARY,"en_US");

$pagetitle = "KSU Bookstore - Shipping";
include_once("sections/header.php");

?>



<div id="content">

<br><br>
<h1>Step 1: Shipping & Billing Options</h1>

<div class="row">

<form method="post" action="payment">
<div class="col-xs-5 col-xs-offset-1">

<h2>Shipping Address</h2>
<br><br><br>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">First Name:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="firstname_s" title="Enter your first
        name." required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Last Name:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="lastname_s" title="Enter your last name."
        required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Email:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="email" class="form-control" name="email_s" title="Enter your email
        address." required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Street:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="street_s" title="Enter your complete
        street address." required>
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">City:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="city_s" title="Enter your city." required>
    </div>


    <div class="form-group text-left col-md-4">
```

```
        <label class="PhilosopherBoldItalic textSize5">State:</label>
    </div>
    <div class="form-group text-left col-md-2">
        <select class="form-control" name="state_s" title="Enter your state." required>
            <?php include("code/states.php"); ?>
        </select>
    </div>
    <div class="form-group text-center col-md-3">
        <label class="PhilosopherBoldItalic textSize5">Zip Code:</label>
    </div>
    <div class="form-group text-left col-md-3">
        <input type="text" class="form-control" maxlength="5" name="zipcode_s"
        pattern="[0-9]{5}" title="Zip code must be 5 numbers."
        onkeyup="this.value=this.value.replace(/[^\d]/g,'')"  required>
    </div>


</div>



<div class="col-xs-5 col-xs-offset-1">

<h2>Billing Address</h2>

<input name="use_shipping" type="checkbox" data-toggle="collapse" data-target="#billcontainer"
checked> <b>Use Shipping Address?</b>
<br><br>

    <div id="billcontainer" class="collapse">


    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">First Name:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="firstname_b" title="Enter your first
        name.">
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Last Name:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="lastname_b" title="Enter your last name.">
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Email:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="email" class="form-control" name="email_b" title="Enter your email
        address.">
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">Street:</label>
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="street_b" title="Enter your complete
        street address.">
    </div>

    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">City:</label>
```

```
    </div>
    <div class="form-group text-left col-md-8">
        <input type="text" class="form-control" name="city_b" title="Enter your city.">
    </div>


    <div class="form-group text-left col-md-4">
        <label class="PhilosopherBoldItalic textSize5">State:</label>
    </div>
    <div class="form-group text-left col-md-2">
        <select class="form-control" name="state_b" title="Enter your state.">
            <?php include("code/states.php"); ?>
        </select>
    </div>
    <div class="form-group text-center col-md-3">
        <label class="PhilosopherBoldItalic textSize5">Zip Code:</label>
    </div>
    <div class="form-group text-left col-md-3">
        <input type="text" class="form-control" maxlength="5" name="zipcode_b"
        pattern="[0-9]{5}" title="Zip code must be 5 numbers."
        onkeyup="this.value=this.value.replace(/[^\d]/g,'')">
    </div>


    </div>
</div>
</div>


<div class="row">
    <hr>
    <button type="submit" class="btn btn-info btn-lg">Next</button>
</div>

</form>

</div>




<?php
include_once("sections/footer.php");
?>
```

```php
<?php
session_start();
setlocale(LC_MONETARY,"en_US");

$type = 1;
if(ISSET($_GET["type"])){
    $type = $_GET["type"];
}

$pagetitle = "KSU Bookstore - Shopping Cart";
include_once("sections/header.php");


$arrayOfBooks = array_map('str_getcsv', file('books.csv'));
$cartlabels = array("New", "Used", "Rental", "eBook");


$physicalFlag = 0;
$finalSales = 0;

$booksDataArray = array();
$order = array();
$orderKeys = array();

if(count($_SESSION["cart"]) > 0){
foreach ($_SESSION["cart"] as $orderLine){
    $order[] = explode(",", $orderLine);
}
}

foreach ($arrayOfBooks as $bookLine){

    foreach ($order as $data){
        if($bookLine[0] == $data[0]){

            $orderKeys[] = count($booksDataArray);
            $booksDataArray[] = $bookLine;


        }
    }
}

?>



<div id="content">

<br>

<form method="get" action="results">

    <div class="text-left row">
    <div class="col-xs-3 col-xs-offset-1">
    <input name="search" class="form-control glyphicon" type="search" placeholder="&#57347;
    Search Again"/>
    </div>

    <div class="col-xs-2">
    <input type="submit" class="btn btn-default" value="Search">
    </div>

    </div>
```

```
    <br class="clearfix">

    <div class="text-left row">

        <div class="form-group text-left col-xs-1 col-xs-offset-1">
            <label class="PhilosopherBoldItalic textSize5">Search by:</label>
        </div>
        <div class="form-group text-left col-xs-2">
            <select class="form-control" name="type" id="type">
            <?php
                $searchlabels = array("Title", "Author", "Description", "Course", "Professor",
                "ISBN");
                $searchid = array(1, 2, 17, 4, 6, 0);

                for($j = 0; $j < count($searchlabels); $j++){

                    $selected = "";
                    if($type == $searchid[$j]){
                        $selected = " selected";
                    }
                    echo "<option value='" . $searchid[$j]. "'$selected>" . $searchlabels[$j] .
                    "</option>\n";
                }
            ?>
            </select>
        </div>

    </div>



</form>

<h1>Your Shopping Cart</h1>

<div class="col-xs-8 col-xs-offset-2">

<?php
if(count($order) > 0){
?>


<table class="table table-striped">
<tr>
    <th><!-- Image Goes Here --></th>
    <th>Textbook Title</th>
    <th>Type</th>
    <th>Quantity</th>
    <th>Per Unit Cost</th>
    <th>Total Cost</th>

</tr>

<?php

for($i = 0; $i < count($order); $i++){

    ?>

<tr>
    <td><a href="details?id=<?php echo $booksDataArray[$orderKeys[$i]][0]; ?>"><img
    src="images/<?php echo $order[$i][0]; ?>.jpg" title="<?php echo
    $booksDataArray[$orderKeys[$i]][1]; ?>" class="checkoutIcon"></a></td>
    <td class="text-left"><a href="details?id=<?php echo $booksDataArray[$orderKeys[$i]][0];
    ?>"><?php echo $booksDataArray[$orderKeys[$i]][1]; ?></a></td>
```

```
    <td class="text-left"><?php echo $cartlabels[$order[$i][1]]; ?></td>
    <td class="text-left">
        <?php
            echo "<input type='number' id='cartvalueS$i' value='{$order[$i][2]}' min='0'
            max='{$booksDataArray[$orderKeys[$i]][9 + $order[$i][1]]}'>";
            echo "<input type='hidden' id='carttypeS$i' value='{$order[$i][1]}'>";
            echo "<input type='hidden' id='cartidS$i' value='{$order[$i][0]}'>";

        ?>
    </td>
    <td class="text-left"><?php echo money_format("$%!n", $booksDataArray[$orderKeys[$i]][13 +
    $order[$i][1]]); ?></td>
    <td class="text-left"><?php echo money_format("$%!n", $order[$i][2] *
    $booksDataArray[$orderKeys[$i]][13 + $order[$i][1]]); ?></td>

    <?php

        if($order[$i][1] != 3){ //If we've ordered at least one physical book
            $physicalFlag = 1;
        }
        $finalSales += $order[$i][2] * $booksDataArray[$orderKeys[$i]][13 + $order[$i][1]];
    ?>


</tr>

<?php

}


?>

</table>

<table class="table">
<tr>
    <td class="largerFont text-left"><b>Subtotal</b></td>
    <td class="largerFont text-center"><?php echo money_format("$%!n", $finalSales); ?></td>
</tr>
</table>




<button id="updatecartS" class="btn btn-default">Update Cart</button>
<a href="checkout" class="btn btn-info btn-lg">Check Out</a>
</form>

<?php
}
else{

    echo "<br>Your Shopping Cart is Empty";
}

?>

</div>


</div>

<?php
include_once("sections/footer.php");
```

```
?>

<script>

$( document ).ready(function() {
        $("#updatecartS").click(function(){
            var i = 0;
        var updateString = "";
        var separator = "";

        while($("#cartvalueS" + i).length != 0){

            if($("#cartvalueS" + i).val() > 0){

                updateString += separator + $("#cartidS" + i).val() + "," + $("#carttypeS" +
                i).val() + "," + $("#cartvalueS" + i).val();
                separator = ";";
            }

            i++;
        }

            $.ajax({
                    url: 'code/updatequantity',
                    type: 'post',
                    data: { 'shoppingcart': updateString},
                    success: function (data, status)
                    {
                    if (status == "success")
                    {
                        location.reload();
                    }
                    },
                    error: function (xhr, desc, err){}
            }); // end ajax call


    });
});

</script>
```

```
<option>  </option>
<option value="AL">AL</option>
<option value="AK">AK</option>
<option value="AZ">AZ</option>
<option value="AR">AR</option>
<option value="CA">CA</option>
<option value="CO">CO</option>
<option value="CT">CT</option>
<option value="DE">DE</option>
<option value="FL">FL</option>
<option value="GA">GA</option>
<option value="HI">HI</option>
<option value="ID">ID</option>
<option value="IL">IL</option>
<option value="IN">IN</option>
<option value="IA">IA</option>
<option value="KS">KS</option>
<option value="KY">KY</option>
<option value="LA">LA</option>
<option value="ME">ME</option>
<option value="MD">MD</option>
<option value="MA">MA</option>
<option value="MI">MI</option>
<option value="MN">MN</option>
<option value="MS">MS</option>
<option value="MO">MO</option>
<option value="MT">MT</option>
<option value="NE">NE</option>
<option value="NV">NV</option>
<option value="NH">NH</option>
<option value="NJ">NJ</option>
<option value="NM">NM</option>
<option value="NY">NY</option>
<option value="NC">NC</option>
<option value="ND">ND</option>
<option value="OH">OH</option>
<option value="OK">OK</option>
<option value="OR">OR</option>
<option value="PA">PA</option>
<option value="RI">RI</option>
<option value="SC">SC</option>
<option value="SD">SD</option>
<option value="TN">TN</option>
<option value="TX">TX</option>
<option value="UT">UT</option>
<option value="VT">VT</option>
<option value="VA">VA</option>
<option value="WA">WA</option>
<option value="WV">WV</option>
<option value="WI">WI</option>
<option value="WY">WY</option>
<option value="American Samoa">AS</option>
<option value="District of Columbia">DC</option>
<option value="Federated States of Micronesia">FM</option>
<option value="Guam">GU</option>
<option value="Marshall Islands">MH</option>
<option value="Northern Mariana Islands">MP</option>
<option value="Palau">PW</option>
<option value="Puerto Rico">PR</option>
<option value="Virgin Islands">VI</option>
```

```php
<?php
session_start();



if(ISSET($_POST["action"])){

    if($_POST["action"] == "add"){

        $id = $_POST["id"];
        $booktype = $_POST["booktype"];
        $quantity = 1;

        $foundit = 0;
        for($i = 0; $i < count($_SESSION["cart"]); $i++){
            if(strpos($_SESSION["cart"][$i], $id) !== false && strpos($_SESSION["cart"][$i],
            "," . $booktype . ",") !== false){

                $updateArray = explode(",", $_SESSION["cart"][$i]);
                $updateArray[2]++;
                $_SESSION["cart"][$i] = $updateArray[0] . "," . $updateArray[1] . "," .
                $updateArray[2];


                $foundit = 1;

            }
        }


        if($foundit == 0){
            $_SESSION["cart"][] = $id . "," . $booktype . "," . $quantity;
        }


        header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
        '/nmurphy1/bookstore/shoppingcart');
        die("Invalid ID");
    }
}


?>
```

```php
<?php

session_start();

$_SESSION = array();


$shoppingcart = $_POST["shoppingcart"];


$shoppingcartArray = explode(";", $shoppingcart);


if (strpos($shoppingcart, ';') === false) {
    $shoppingcartArray[0] = $shoppingcart;
}

if(strlen($shoppingcart) > 2){
    for($i = 0; $i < count($shoppingcartArray); $i++){
        $_SESSION["cart"][] = $shoppingcartArray[$i];
    }
}

include_once("displaycart.php");

?>
```