The page features a decorative graphic consisting of three overlapping blue circles of varying sizes, arranged in a diagonal line from the top right towards the bottom right. Two thin blue lines intersect at the top left, forming a large 'V' shape that frames the text on the left side of the page.

Team Dec 14-13

CySwap

Design Document

Client: Merry Rankin
Adviser: Professor Mitra

Team Members:
Fabian Briesmoore
Jared Cook
Kyle Johnson
Adam Sunderman

Table of Contents

Functional Requirements	3
Non-Functional Requirements	4
System Block Diagram	5
System Analysis	6
Input/Output Module Specification	7
Request Manager	7
Configuration Manager	8
Verification/Authentication	9
Update Manager	9
Moderator Manager	10
Database Manager	11
Page Generator	12
User Interface Specification	13
Home Screen	13
Category Screen.....	14
Posting Screen	15
Contact Seller Popup	16
Report Screen	17
Create a Post Screen.....	18
Software Specification.....	19

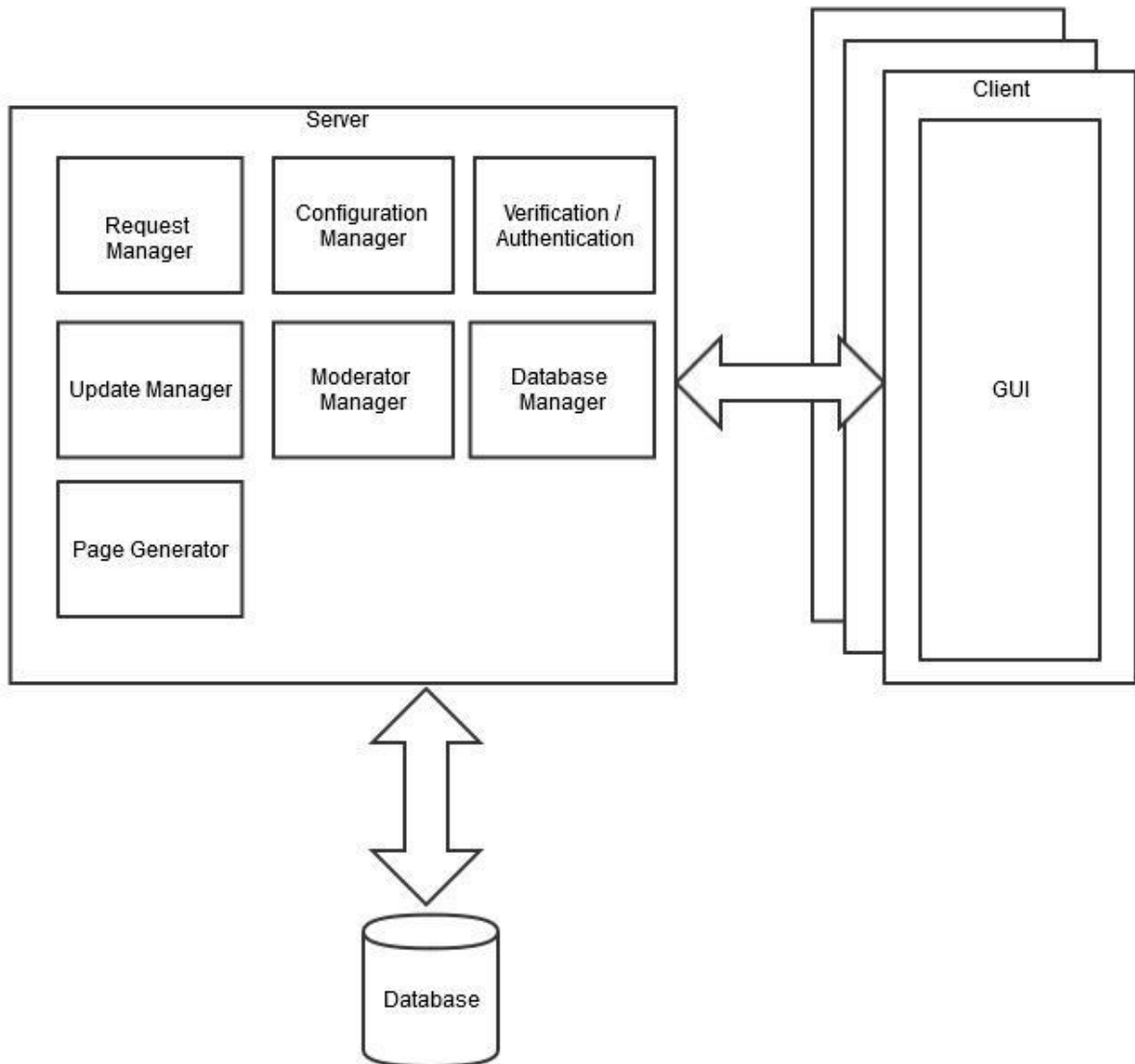
Functional Requirements

FR-1	The system shall allow users to login and logout via Iowa State's NetID system.
FR-2	The system shall allow users to post items for sale.
FR-3	The system shall allow users to search for posts based on specified criteria.
FR-4	The system shall allow users to view item postings.
FR-5	The system shall allow users to post comments on any posting.
FR-6	The system shall allow sellers to mark transactions as completed.
FR-7	The system shall allow users to contact sellers to initiate discussion on completing the transaction.
FR-8	After a transaction has been completed, the system shall allow users (buyer and seller in a specific transaction) to leave positive or negative feedback on one another.
FR-9	The system shall allow users to report posts that violate the posting code of conduct.
FR-10	The system shall allow moderators to view reported postings in a F.I.F.O. ordering.
FR-11	The system shall allow moderators to remove postings from the list of reported postings.
FR-12	The system shall allow moderators to edit or delete postings that violate the posting code of conduct.
FR-13	The system shall allow moderators to suspend account privileges for users who abuse the website's posting code of conduct.
FR-14	The system shall allow admin to assign and remove moderator privileges from NetID accounts.
FR-15	The system shall allow admin assign and remove admin privileges from NetID accounts.
FR-16	The system shall allow admin to view usage statistics.
FR-17	The system shall allow admin to edit content in information sections of the website (i.e. posting code of conduct).
FR-18	The system shall allow admin to configure which site features are enabled and disabled.

Non-Functional Requirements

NFR-1	The system should be seen as visually appealing by a number > 75% of surveyed students.
NFR-2	During high traffic timeframes (beginning of semester) response times should be < 1 second on average. (assuming no network delay on client side)
NFR-3	The site should not be able to have its information maliciously accessed through user input, direct server requests and cookie jacking.

System Block Diagram



The server communicates with the database to get user data (such as account details and settings) and other necessary data. This data, along with other data (possibly from other databases) is used to generate a page. The pages will likely be generated using php. The server is also responsible for authenticating the user. The client will communicate to the server through a session.

System Analysis

The system is designed to allow Iowa State students and faculty to be able to sell and trade books and other commodities with other users. Account details, system settings, posted data, and other necessary data is stored in the MySQL database and accessed by the server. Managing users and other system maintenance will be managed by GSB (Government of the Student Body) representatives. The system will likely be managed by individuals with little technical experience. Consequently, intuitive and simple administrator options will be needed for at least one administrator user. Different levels of administrators will likely be necessary. Administrators will be able to block users, manually adjust user data and postings made by users. In addition to being easily configurable by one or more administrators, the system will need to implement searching and filtering for different items, will be compatible with Iowa State netids, will need to be easily expandable for new categories, will have a feedback system, will need to be secure, and may need to communicate with other databases to get data for different categories.

Though this system is responsible for connecting users who want to sell and buy a common item, it is not responsible for dealing with financial transactions. Users will communicate to each other through email after clicking on a link from the post of the item the buyer is interested in. This is possible because all users that are able to connect to other users will be logged in using their Iowa State email accounts. Viewers of a page that are not logged in will be able to view postings but not be able to see specific information about items, including information necessary to contact the seller. Though the system will be secure with user data, it will not be able to ensure that users are always honest. For this reason, a feedback system will be implemented. Additionally, information advising users to be cautious when making transactions will be posted on the website.

Though this is currently how the system is going to operate, the features that GSB wants could change and better implementations might replace the current system. Because of this, meetings with them are ongoing and system design is still in progress.

Input/Output Module Specification

Request Manager

The request manager will contain the code to accept URL requests and delegate the type of tasks to the different request manager classes (Page Fetcher, Configuration Manager, Moderator Manager, etc.).

Module Function	Input	Output
Accept URLs from clients' browsers	Requested destination URL.	html and javascript for client to render requested page

Test Case	Success	Fail
User clicks a link on current html page.	link exists in routing php. Page loads successfully.	link does not exist in routing php. Page doesn't load successfully.

Configuration Manager

The Configuration Manager will contain the code to manage the configuration of the website from enabled and disabled features to the content displayed on the information pages (such as the posting code of conduct).

Module Function	Input	Output
Modify information content on static pages.	html/javascript object to modify, new content	html and javascript for client to render updated page.
Enable Feature	featureID	html for client to render "success" notification.
Disable Feature	featureID	html for client to render "success" notification

Test Case	Success	Fail
User (admin) modifies static content	Content is in valid format for page object that user is trying to modify (string for text modifications)	Content isn't in valid format for the page object that is being modified.
User (admin) enables feature	Feature becomes enabled on the site.	Feature does not become enabled on the site.
User (admin) disables feature	Feature becomes disabled on the site.	Feature is not disabled on the site.

Verification/Authentication

The Verification/Authentication class will contain the code to interact with Iowa State's NetID login system to authenticate users.

Module Function	Input	Output
User login	email/password	Verification that a user is logged in (Session)

Test Case	Success	Fail
User logs in for the first time (logs in with NetID and our system creates a user with that NetID and stores necessary information on top of that profile).	NetID added to our database. Session is successfully created.	NetID added to our database. Session is not created.
User logs in after their first time (uses profile already in our database).	Session is successfully created.	Session is not created.

Update Manager

The update manager will contain the code for users to create and update information on their posts.

Module Function	Input	Output
Create Post	Post type, post sellerID, Posting fields (varies by type).	HTML and Javascript for client to render posting.
Update Post	Posting ID, modified fields	HTML and Javascript for client to render updated posting.

Test Case	Success	Fail
User creates posting	Associated information is stored in the database.	Associated information is not stored in the database
User updates posting.	Database fields are updated to match the user's update.	Database fields are not updated to match the user's update.

Moderator Manager

The moderator manager will contain the code for moderator functions like report handling and blocking users

Module Function	Input	Output
Block a user.	User's netid	Confirmation that the user has been blocked.
Unblock a user.	User's netid	Confirmation that the user has been unblocked.
Close a reporting case.	A report's status.	Confirmation of the report's new status.

Test Case	Success	Fail
User (admin) blocks a user.	User is blocked from the system.	User can still gain access to the system.
User (admin) unblocks a user.	User is not blocked from the system.	User cannot gain access to the site.
User closes a report.	Report is marked as closed and is archived.	Report is not successfully closed.

Database Manager

The Database Manager will contain code responsible for the retrieval, updating, depositing and deletion of information to the database.

Module Function	Input	Output
Retrieve information from the database.	Location of information to be retrieved.	Information at that location.
Update information in the database.	Which information to be updated and what the new information is.	Confirmation that the database update was successful.
Deposit information from the database.	Information to be stored in the database and in which table.	Confirmation that the database addition was successful.
Deletion of information in the database.	Location of information to be deleted.	Confirmation that the database deletion was successful.

Test Case	Success	Fail
Request to retrieve information from the database.	Correct information is returned.	No information is returned or incorrect information is returned.
Request to update information in the database.	Database successfully updated information.	The database's information was not updated successfully.
Request to put information in the database.	New information is stored in the database.	New information is not stored in the database.
Request for information to be removed from the database.	Information is removed from the database.	Information is still in the database.

Page Generator

The Page Generator will contain code responsible for generating the html and javascript for each page of the site.

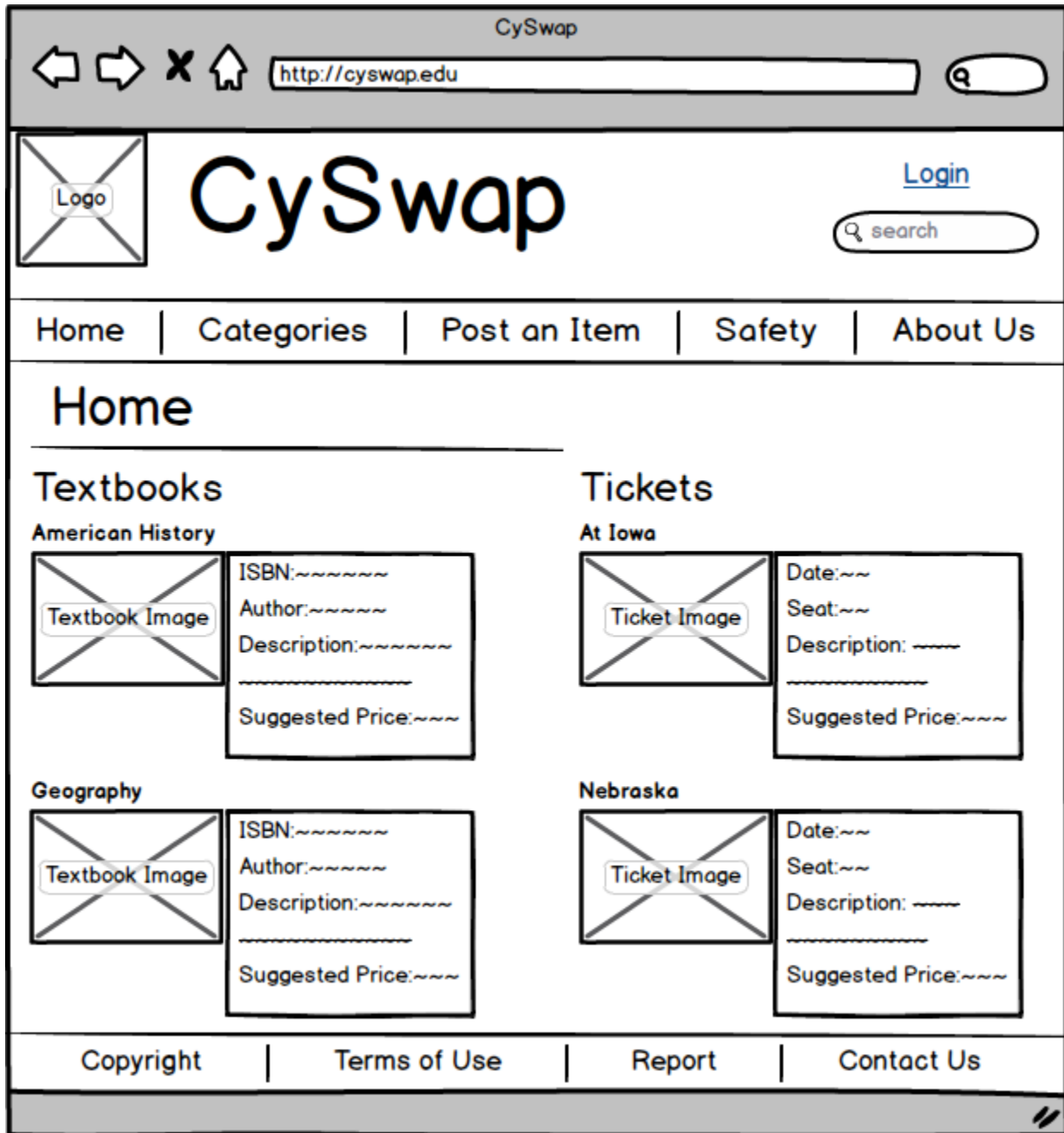
Module Function	Input	Output
Generate the html and javascript for each page of the site.	Requested page.	html and javascript for the requested page.

Test Case	Success	Fail
Home page is requested.	Correct html and javascript is returned.	Html and Javascript is either not returned or is incorrect.
Posting page is requested.	Correct html and javascript is returned.	Html and Javascript is either not returned or is incorrect.
Search page is requested.	Correct html and javascript is returned.	Html and Javascript is either not returned or is incorrect.
Category page is requested	Correct html and javascript is returned.	Html and Javascript is either not returned or is incorrect.
Create a post page is requested	Correct html and javascript is returned.	Html and Javascript is either not returned or is incorrect.
Reporting page is requested	Correct html and javascript is returned.	Html and Javascript is either not returned or is incorrect.

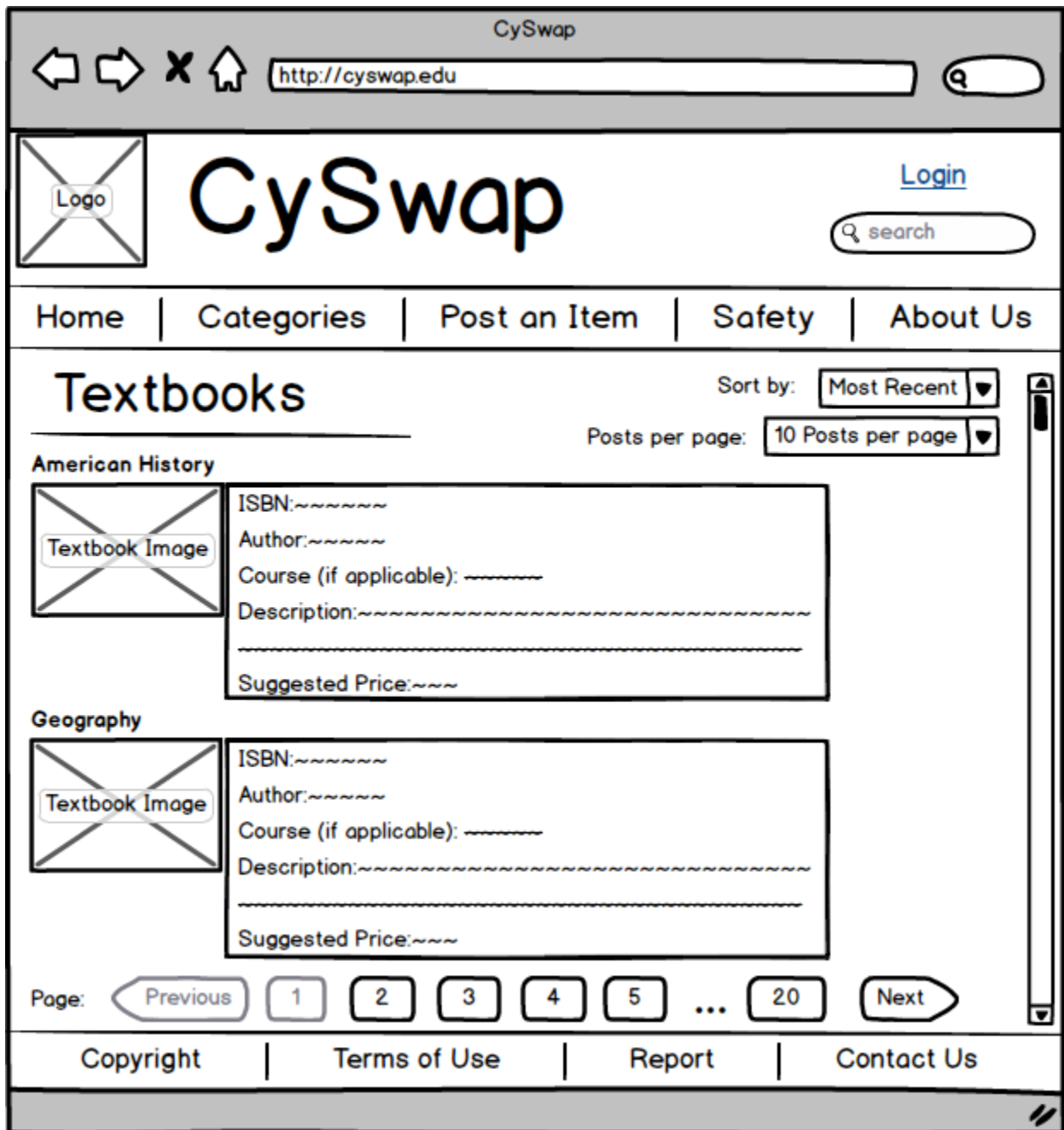
User Interface Specification

This section will show all the different mockups for each screen.

Home Screen



Category Screen



Shows brief descriptions of several postings within a certain category.

Posting Screen

The screenshot shows a web browser window with the address bar displaying 'http://cyswap.edu'. The page header includes the 'CySwap' logo, a 'Login' link, and a search bar. A navigation menu contains 'Home', 'Categories', 'Post an Item', 'Safety', and 'About Us'. The main content area features a listing for 'American History, the Early Years' by seller 'kabernsj'. The listing includes a thumbs-up icon with a count of 3, a thumbs-down icon with a count of 1, a 'Textbook Image' placeholder, and three preview images labeled 'P1', 'P2', and 'P3'. The 'Suggested Price' is listed as '\$37' with a 'Contact Seller' button below it. A detailed description box contains the following text: 'Title: American History The Early Years, Student Edition', 'ISBN: 9780078216114', 'Author: Glencoe', 'Edition: 3rd', 'Course: History 999', and 'Description: This textbook is in great shape, it is like new. It has only been used a few times. There is no writing or highlighting in the textbook.' The footer contains links for 'Copyright', 'Terms of Use', 'Report', and 'Contact Us'.

CySwap

http://cyswap.edu

Logo

CySwap

Login

search

Home | Categories | Post an Item | Safety | About Us

American History, the Early Years

Seller: kabernsj

3 1

Textbook Image

P1
P2
P3

Suggested Price:
\$37

Contact Seller

Title: American History The Early Years, Student Edition
ISBN: 9780078216114
Author: Glencoe
Edition: 3rd
Course: History 999
Description: This textbook is in great shape, it is like new. It has only been used a few times. There is no writing or highlighting in the textbook.

Copyright | Terms of Use | Report | Contact Us

Shows an individual posting.

Contact Seller Popup

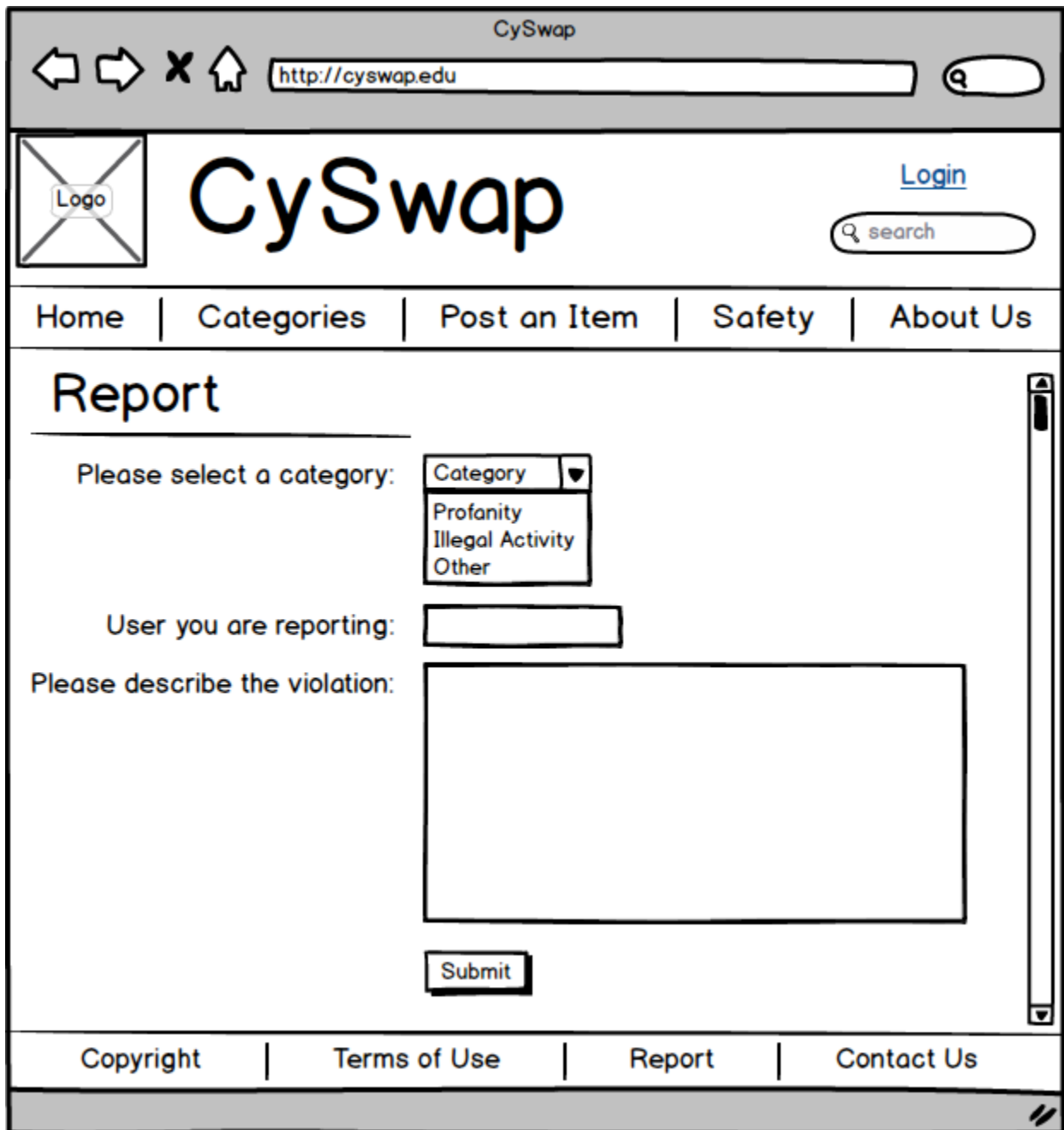
Contact Seller

To: kabernsj

Subject: American History, The Early Years on CySwap

Pop-up that will appear when the 'Contact Seller' button from the previous screen is clicked.

Report Screen



The screenshot shows a web browser window with the address bar displaying "http://cyswap.edu". The page title is "CySwap". The navigation menu includes "Home", "Categories", "Post an Item", "Safety", and "About Us". The main content area is titled "Report" and contains the following form elements:

- A label "Please select a category:" followed by a dropdown menu with options: "Profanity", "Illegal Activity", and "Other".
- A label "User you are reporting:" followed by a text input field.
- A label "Please describe the violation:" followed by a large text area.
- A "Submit" button.

The footer contains links for "Copyright", "Terms of Use", "Report", and "Contact Us".

Screen where a user can report behavior that is a violation of the site's terms of use.

Create a Post Screen

The screenshot shows a web browser window with the address bar containing 'http://cyswap.edu'. The page title is 'CySwap'. The navigation menu includes 'Home', 'Categories', 'Post an Item', 'Safety', and 'About Us'. The main content area is titled 'Post an Item' and contains a form with the following elements:

- A label 'Please select a category:' followed by a dropdown menu with 'Category' selected and a downward arrow. The dropdown list shows 'Textbook' and 'Miscellaneous'.
- An 'ISBN:' label followed by a text input field.
- A 'Description:' label followed by a large text area.
- A 'Submit' button at the bottom of the form.

The footer contains links for 'Copyright', 'Terms of Use', 'Report', and 'Contact Us'.

Screen users will see when they are posting an item.

Software Specification

Languages: PHP, JavaScript, HTML, CSS, and mySQL.

Libraries: Laravel (PHP framework)

Database Server: mySQL

Development Tools: Atlassian Products: Bamboo, Clover, Confluence, Crowd, Jira and Stash.