Senior Project I

Inquisitive Study Site (ISS)

Fall 2024, 11/19/24

Caleb Rachocki
Caleb Ruby
Caleb Massey
Ibrahim Alani

INSTRUCTOR COMMENTS / EVALUATION: 94

Overall a great specification document that included the required components.

Great "Overview" section

"Tester" section is well done.

Dynamic modeling and Dataflow were well presented.

Acceptance criteria need revision.

Contents

ABSTRACT:

The Inquisitive Study Site (ISS) is a web-based platform where users can create, discover, and study personalized "study sets" composed of terms and definitions. Our intent is to provide a free, streamlined alternative to similar popular resources, such as Quizlet and Brainly. We will offer quintessential learning features such as the ability to create and use these sets with our built-in tools, such as flashcards, practice exams, and adaptive learning methods. In addition to the use of one's own sets, the ISS will act as a freely accessible library of community-generated content, with all data stored in an off-site SQL database. The platform will only require a web-enabled device with standard input and output hardware.

The role of this document is to explicitly describe what the ISS intends to do and how we as the developers intend to implement these features. In this document, we will further discuss the SQL server, the uses and intended implementation of the SQL server, the hosting of the ISS website, and its functional and nonfunctional capabilities. This document will also outline the testing criteria, operating environment, user interaction with the website, integration with the back-end systems, and present dataflow diagrams and system models.

Another function of this document will be as a binding agreement between students and the professor(s) that we, as the developers of this project, will commit ourselves to work on the project to the best of our ability, and implement everything as described here. The intended audience of this document are those who desire to learn about the ISS and its inner functions, all developers working on the project, and the professor(s) reviewing the documentation.

Commented [TK1]: long sentence, maybe break it up into two separate ones.

Commented [TK2]: comma is not needed

Commented [TK3]: "as well as, and" you only need one conjugate in the items in a series

SYSTEM DESCRIPTION:

Overview:

The ISS website will compose of a frontend for the users to easily access and utilize all study sets that have been stored via our SQL database, and backend logic handling access to data, as well as our built-in learning tools. When users first visit the ISS webpage, they will see the home page, which features buttons linking to key functionalities such as "Login," "Home," “About,” “Explore,” and "Search." The home page will also feature a guide for new users, a hub to access one’s personal library of study sets, and a history of recently studied sets.

When a user clicks the “Search” button, they will be presented with a search bar in which they can enter a quick text search. Upon pressing enter on the keyboard, the user will be directed to the “Explore” page, with the results displayed for their entered input. The “Explore” page, beyond searching by text, offers filters using the website’s “tag” system, and the ability to sort sets based on things like creation date and number of recent visits.

After selecting a study set, the user will be taken to an external webpage, where they will be able to utilize the study set. The user will be presented with options regarding how they use the study set, namely studying via flashcards, practice exams and quizzes, basic true and false, etc. If the user of a set is also the creator of that set, the options to delete or edit the set will be displayed.

When a user clicks the “Login/Sign Up” button, they will be presented with a screen to input a username and password to sign into their account. If a user does not have an account, they will also have the opportunity to create an account. No two users may have the same username, and the user will be notified upon attempting to create an account with an identical name to another. A valid password must also meet certain criteria. Failure to meet this criteria

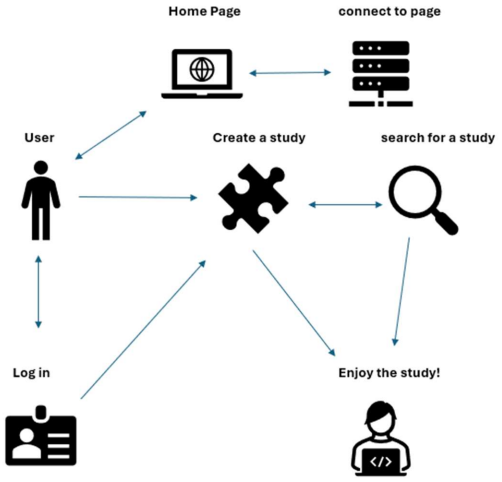
will similarly invalidate the account creation and notify the user of the error. After creating an account, the user will have permission to create their own study sets. When already logged in, the "Login/Sign Up" button will be replaced by a text label matching the username of the current user. Pressing this button will display available account actions, such as "logout" and "delete."

Commented [TK4]: separate these sentences to make it more coherent

Under the "Home" tab, users may enter their "Library", where they may select the "Create" button. After clicking this button, they will be directed to a separate webpage where they will be able to name the study set and add terms and definitions. A save button will finalize the creation of the set, saving all the entered data and generating the creation timestamp associated with the set.

Commented [TK5]: the creation*

Figure 1: General Overview of the ISS Project's Functionality



Environments and Constraints:

End User Profile:

The majority of our users will likely be students of a variety of ages, though our target demographic is high school and college-age students. With this in mind, we intend to keep the interface simple, clean, and self-explanatory to ensure students of all ages can use the ISS website. Users will only need to possess basic reading comprehension and direction following skills to properly use the ISS website.

Commented [TK6]: add hyphen

User Interaction:

Users will interact with our website via mouse and keyboard, or any equivalent means of pointing, making selections, and entering text. All output from the website will be visual, requiring a screen of some type.

Hardware Constraints:

Hardware constraints regarding the user are simple: a web connected device with the ability to select buttons and enter text. This includes touchscreen mobile devices such as tablets and smartphones, as well as personal computers equipped with a mouse and keyboard. As our service is fully online, users with slow internet speed may suffer long wait times, but the type of wireless network will not be relevant. Users will be able to access our website through a home Wi-Fi network, public Wi-Fi, cellular data, a personal hotspot, or any other common means of wireless connection.

Software Constraints:

The ISS website is designed to be compatible with all commonly used devices and browsers. It will support modern versions of major web browsers, including Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari, ensuring a consistent experience across platforms. The website will also be responsive, allowing seamless functionality on both desktop and mobile devices with varying screen sizes. Additionally, the system will require minimal processing power and memory, making it accessible on standard PCs, laptops, and smartphones without performance degradation. Users with damaged or atypical hardware devices may run into issues navigating our site.

Time Constraints:

Performance and Response Time Constraints:

Precise response times are not a requirement for this project, nor can we guarantee fast loading times as the user's internet speed will greatly impact the speed at which our elements and webpages are loaded. However, streamlined design and use is a priority for this project. Since we cannot guarantee quick response times because of user-sided hardware constraints, we will do all that we can to cut down on delays when we are able.

One area of import in this regard is our search features. When developing our search features, we will select efficient search algorithms and focus on making queries efficient as well as accurate, with the aim of improving our website's responsiveness and ease of use as much as possible.

Commented [TK7]: no comma

Commented [TK8]: possessive

Production time constraints:

The time constraints of the ISS project are limited to a single academic semester, which is approximately fifteen weeks. Developers of the project are free to work on their parts of the ISS project at any given point, either separately or in a group environment. The final product will be presented at the end of the specified fifteen weeks. Given the development is taking place over an academic semester, all team members will have their priorities split between various other classes taking place at that time; it will be important that everyone take on their assigned workload and leadership responsibilities to the best of their abilities.

Cost Constraints:

The cost of the ISS project will primarily come from the server. Since we plan to utilize a single server for both hosting and data storage, this will cut the cost of a server in half. The team has discussed utilizing a Raspberry Pi to host the server and database from for sake of the presentation. Should the ISS website receive high traffic and usage, we will separate the hosting and database onto two different servers. We are planning to utilize a *Raspberry Pi 4 Model B* for demonstration purposes; the approximate cost in USD as of writing is $64.99 (raspberry).

Other Concerns:

Our primary concern is the size of this project; none of us have done anything to this scale before. Furthermore, we must also figure out how each piece of the project will interact with one another. One of our other concerns is the management of our languages, and how we can get them to interact with each other. Our final concern is keeping the server active and managing its processes.

Commented [TK9]: possessive

Acceptance Testing Criteria:

Testers:

Each team member will also act as a tester and share responsibility for testing the various features of the product. The team will test the ISS individually, as well as in a group setting where we can discuss the performance and possible solutions to any issues found to ensure we have effective and consistent debugging throughout the development of the ISS.

Some key testing criteria:

- Account actions
    - The team will test to ensure that users are able to create and login to their account and their account only
- Study set actions
    - We will test to ensure that only users with an account have access to creation and editing tools, and that they only may do so on their own study sets
- Data Security
    - We must ensure that important data, such as passwords, stored in the SQL database is both encrypted and sorted properly, such that no one can gain access to features, information, or privileges linked with someone else's account

Commented [TK10]: capitalize

Once the final product is complete, we will work through a full feature checklist to ensure that we have implemented all mechanics as intended. The SQL database will be checked regularly for correct data and relations. This will be done with admin accounts, which will allow one to display all stored data, and to edit, add, and remove study sets regardless of who created them.

Developers may also ask third parties for input on the appearance of the ISS webpages, to ensure that they're both appealing and easy to navigate. We will not be asking anyone aside from ourselves to actually test the website until the product is complete. Once the product is completed, we may again seek outside help to test its basic functions. Through such unbiased testing, we seek to ensure the ISS is appealing, easy to navigate, and simple to use from the perspective of a potential user. Outside testers may consist of our classmates, professors, or family members. There will be no online resources used in testing.

We will also ensure that each feature is tested and reviewed by at least one person who did not implement that section of code to ensure that no features suffer from any kind of bias or oversite that may crop up from our natural perspective limitations. This peer review offers a "chance to ensure that other developers aren't checking in poor quality code", as well as to keep all members of the team familiar with each component of the project (Neale, 2023).

Commented [TK11]: is this finished?

Criteria for User Acceptance:

Successful user acceptance occurs when the non-account user can access study sets that others have made. A second stage of successful user acceptance occurs when they're able to create an account to make their own study sets for personal use, or for other users of the ISS. User acceptance also depends on the ability to access the project from any computer that has access to the internet.

- Guests
    - View Study Sets
    - Account Creation
- Account Users

    - Guest Privileges, i.e., only creators of a set can alter them
    - Account Login
    - Study Set Creation, Alterations, and Deletion
    - Account Logout
    - Account Deletion
- Admins (Developers)
    - Guest + Account User Privileges
    - Quality Control of Study Sets + Accounts
    - Tag Generation
    - General Database Management, i.e., overview of all study sets and accounts

Integration of Separate Parts and Installation:

The ISS website software will be loaded onto and hosted from the Raspberry Pi, as will the database. The Raspberry Pi will communicate to the internet through an ethernet or Wi-Fi connection. Users will then be able to connect to our web domain through their browser.

SYSTEM MODELING:

Functional:

Once the user finds and accesses the ISS website from the internet, they will be presented with the ISS homepage. This homepage will have some buttons and options for the user to choose from. These options are displayed in the diagram below.

Figure 1: Simplified general flow of the ISS final product.

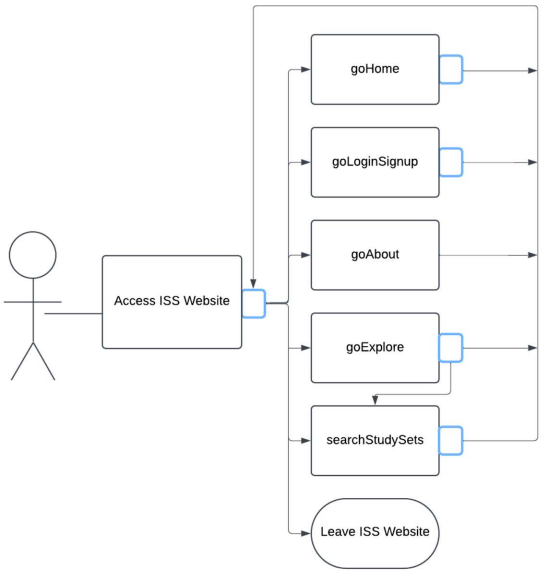


Figure 1 shows an abstract representation of choices the user may make once entering the website. The user will first be directed to the homepage, but they will always have the option to go to any other page via a navigation header, including the home page itself. Each page will have its own options and elements displayed once the user selects that page from the header.

Run Scenario;

- The user accesses the ISS website
- The user will be presented with the homepage and its options (detailed below)
- The user may select one of the options from the navigation header, then use their desired features from their selected page
    - Home
        - Quick-start guide
            - simple explanation of how to use the core features of our site
        - Library
            - list of study-sets created by this user
        - Study history
            - list of sets studied by this user
    - Explore
        - Full featured search, displaying list of study-sets matching search criteria
    - About
        - Detailed information on the website and it's operation
    - Search
        - Not a dedicated page, rather a search bar in which the user may quickly enter a text search
        - Will redirect to the explore page with the entered search text and the results

- Login / Signup
  - Sign Up
    - Option to enter a new username and password to generate a new account
  - Login
    - Option to enter the username and password for an existing account
  - For an existing account
    - Account options, such as edit username and password or delete account

Class Diagrams:

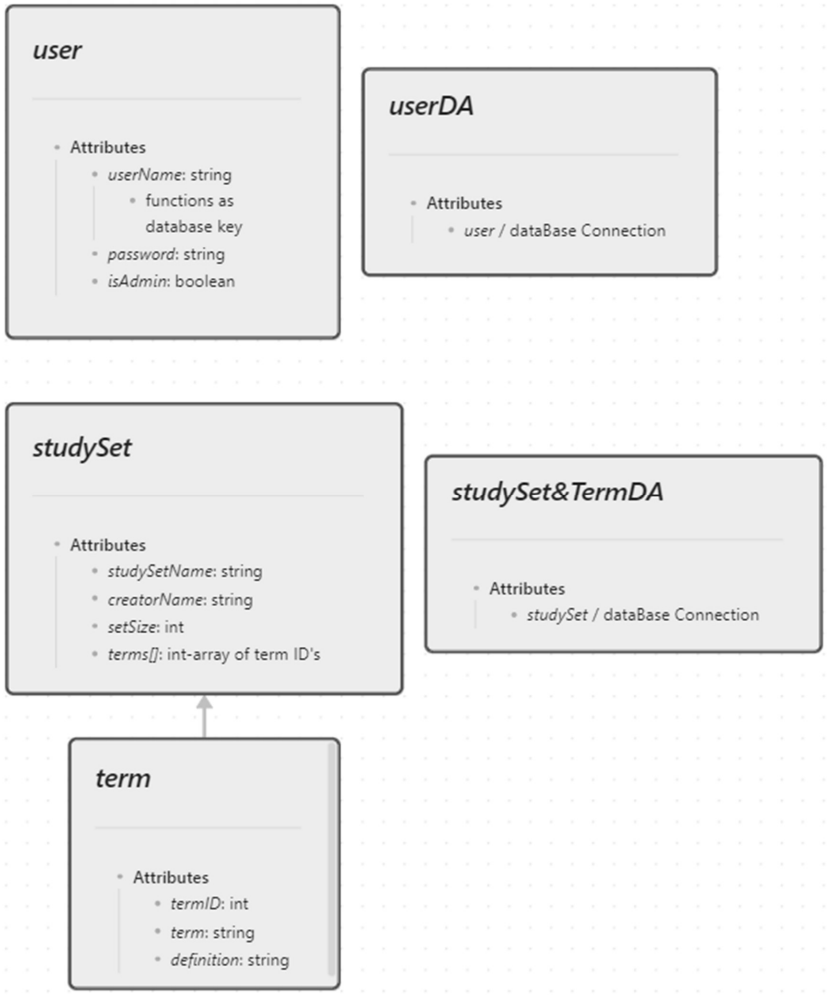


Figure 1: Class Diagram Showing Entity Classes and Database Relations

User Class:

- Username- a string that functions as a unique key in the database for identifying users.
- Password- a string used for user authentication.
- isAdmin- a Boolean indicating whether the user has admin privileges, such as the ability to edit or delete a study-set created by another user.

UserDA:

- This data access class signifies a relation between the user class and database, facilitating storage and retrieval of user data.

StudySet Class:

- StudySetName- a string representing the name of the study set and operating as a key by which data can be fetched from the database for the appropriate set.
- CreatorName- a string storing the name of the user who created the study set.
- SetSize- an integer representing the size of the study set.
- Terms [setSize]- an integer array containing the IDs of the terms within the study set.
    - These ID's can be used as keys linked to the corresponding terms in the database table for this study-set.

StudySet&TermDA:

- This data access class represents a relationship between the StudySet class and corresponding database tables.

Term Class (inherited by StudySet):

- TermID- an integer uniquely identifying a term.
- Term- a string containing the content of the term.
- Definition - a string containing the definition corresponding to the term.

Dynamic - Statechart:

Figure 2: Upper-Level Website Navigation

- Upon First Visit, user will be presented with the home page. At all points in time, a navigation header will be visible, displaying the tabs “Home”, “About”, “Login/Sign Up”, “Search”, and “Explore”.
    - *For simplicity’s sake, this will not be mentioned as a possible transition from every state; it should be implied that the user may select any of these options from any other state, demonstrated by Figure 2.

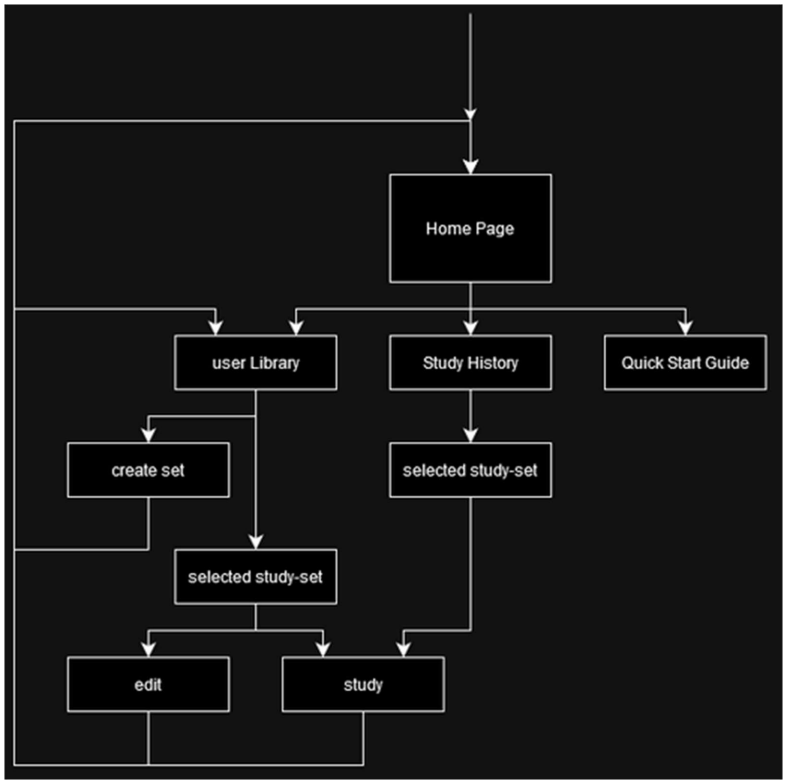


Figure 3: Home Page State Navigation

- Figure 3 displays the dynamic structure of the home page.
- From the homepage, the website has three distinct states available to the user:
    - The user Library will display the sets created on this account, as well as the option to create a new set
        - Upon creating a new set, the user may return to their library, or to the home page.
        - Upon selecting a study set from their library the user may edit or study the set.

- After editing, the user may return to their library or to their homepage.
- After studying, the user may return to their library or their homepage.

- Upon selecting “Study History,” a list of recently studied sets will be displayed.
  - A set may be selected and studied. When complete, the user may navigate back to their homepage.

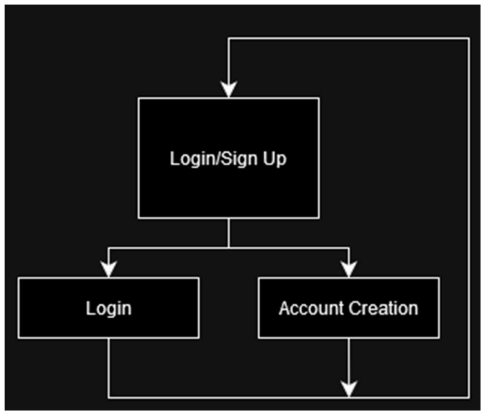


Figure 4: Login/Sign Up State Navigation

- Figure 4 displays the dynamic structure of the login and signup page.
- From this page, users can:
    - Log in with their credentials
        - if successful, they proceed to the Login/Sign Up page, which now displays account information and options.
    - Create an account
        - if successful, they proceed to the Login/Sign Up page, which now displays account information and options

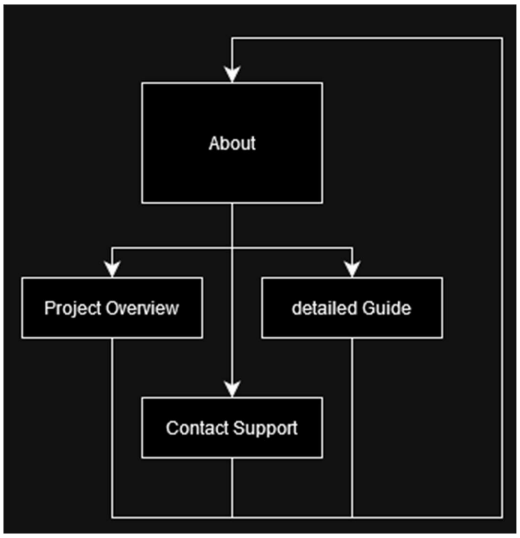


Figure 5: About Page State Navigation

- Figure 5 displays the dynamic structure of the about page.
- This page provides static content detailing website information, goals, and relevant content.

- This is the main state that presents an overview of the site. It offers options:
    - Project overview:
        - Users can navigate to a high-level overview of the project, providing insight into its purpose and key features.
    - Detailed guide:
        - Users can select a more in-depth guide that offers comprehensive information on website features
- Contact support:
    - From either the project overview or detailed guide, users have the option to reach out for support.
        - After contacting support, users can navigate back to any of the previous states to continue exploring or reading.

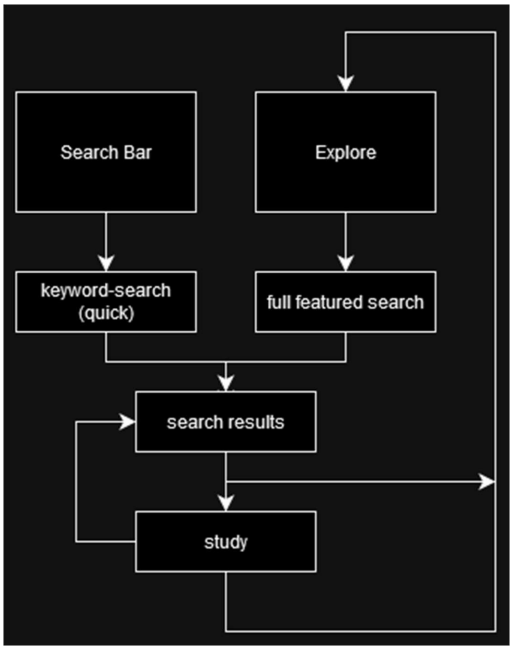


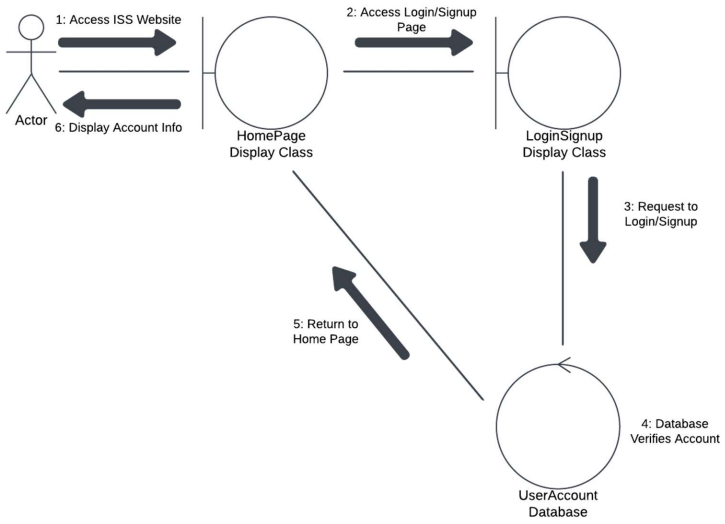
Figure 6: Search & Explore State Navigation

- Figure 6 displays the dynamic structure of the search and explore page.
- Search state:
    - allows for quick keyword-based searches, which redirects to the explore page, displaying generated results relevant to the user's input.
- Explore state:
    - Enables users to conduct a more in-depth search for discovering content based on filters and sorting options
- Search results:
    - Displays the outcome of either a keyword-based search or a full featured search, users can select items from these results to study, or return to the explore page to search again.

- Study state:
    - When an item is selected from the search results:
        - The user enters the study state, once finished, users can return to their search results and explore additional content or navigate back to the main page.

DATAFLOW DIAGRAMS:

The following collaboration diagram shows how the user will interact with two classes and the database of the ISS when making an account, or logging into a pre-exiting account.



The *LoginSignup* use case is described in the following steps;

1. The user accesses the Inquisitive Study Site. The user will have the option to choose which webpage to go to in the top bar of the home page. The home page will display

login status, as well as previous study sets the user has accessed. It will also display the option to edit their own study sets.

2. The user will choose to access the Login / Signup webpage. Here they will have the option to sign into a pre-existing account or make a new one. The user will have two text boxes to fill out, *Username* and *Password*. If the user enters username data that is not associated with an account, they will be able to click the C*reate Account* button. If they enter username data that is associated with an account, they will not be able to create the account.
3. The LoginSignup webpage will verify the entered data meets our security criteria, then send the sign-in request to the database.
4. While the user remains on the LoginSignup webpage, the database will be queried for the matching username and password combination. If the combination is not found, or if it does not meet the security criteria, the user will remain on the LoginSignup webpage with an error, still not signed into an account.
5. If the user's account data is valid, either by successfully logging into or creating an account, they will be directed to the home page
6. From the home page, the user will be able to see their previously accessed study sets, as well as access their own personal study sets.

COMPONENTS / TOOLS NEEDED:

- *Raspberry Pi*
    - Small computer capable of storing, running, and hosting the source code for our website, as well as a database capable of storing data for a small set of users.
- Linux (Ubuntu)
    - Distribution of the Linux operation system to be used on the Raspberry Pi for our webhosting needs. Chosen for readily available and efficient webhosting tools.
- GitHub
    - Means to simplify and unify our version control and better track implementation progress.

References:

Neale, T. (2023, September 12). The importance of code reviews. Surviving Software. https://survivingsoftware.dev/the-importance-of-code-reviews/

Raspberry Pi 4 Model B; Broadcom BCM2711 Quad Core Cortex-A72 (ARM V8) 64-bit SOC; 8GB DDR4 RAM. Micro Center. (n.d.). https://www.microcenter.com/product/622539/pi4modelB8gb?src=raspberrypi

Appendix: Writing Center Report

- Grammar:
  - - Commas: There are a few places throughout the report that do not need commas. Be sure that when you put a comma, it is combining two independent clauses. I added comments in some places where I noticed it, but review the rest of the document for this mistake.
  - - Conjunctions: Several sentences have items in a series where the words "as well as" and "and" are both used. This is not needed. Only use one or the other before the last item in the series.
  - - Hyphen: I only noticed one word that needs a hyphen. I added a comment on the document where this was.
  - - Contractions: I had a question about the use of contractions in the document. Be sure that these are allowed.
- Miscellaneous:
  - - In the abstract, there is a long sentence that could be split up. I recommend splitting this into two sentences.
  - - Filler words: There's a sentence that says, "both username and password," but the "both" is not needed since "and" already implies that.
  - - Coherence: A few sentences lack coherence, and the wording could be spruced up a bit.
  - - Vocabulary: The vocabulary used in this document is great.

Appendix: TECHNICAL GLOSSARY

Account User – Users who access the website and choose to make an account, thus allowing them to create their own study sets to work with.

Back-End - The part of a website or application that users do not see; it includes servers, databases, and application logic that process data.

Customization - The ability for users to modify or personalize their study sets and preferences according to their individual needs.

Database - An organized collection of structured information or data, typically stored electronically, that can be easily accessed, managed, and updated.

Encryption - The process of converting information into a code to prevent unauthorized access, ensuring data security, especially for user account information.

Flashcards – Simple study aids that have a question or term on one side and the corresponding answer or definition on the other, used for self-testing and learning.

Front-End - The visible part of a website or application that users interact with, including layout, design, and user interface elements.

Subject Tags – Refers to the specific tag a study set must receive on its creation, informing users of the type of content this set is capable of providing, i.e., “Biology,” “Chemistry,” “Mathematics,” and any other basic terms describing academic fields of study.

Study Set – Refers to a data set uploaded by a user to be utilized in the website’s study tools. This data set is meant to be filled with a given subject’s material, i.e. questions and their associated answers. The ISS will store this data in the SQL database.

HTML – The mark-up language in which many websites, including ISS, are written in. This provides the appearance and functionality of websites, and is often referred to as the "front-end."

SQL Database – Refers to the project's off-site data storage for the study sets, as well as user account information (MySQL).

Study Tools – Refers to the traditional study tools that existed before any online tools did, i.e., flash cards (showing a question on one side, the answer on the other), customized practice exams, multiple choice questions, true/false questions, etc.

Version Control - A system that records changes to files over time so that specific versions can be recalled later, facilitating collaboration among multiple developers.

Website Admins – Refers to the developers and moderators of the website, who ensure the content of the website is both appropriate and maintained.

Appendix: TEAM DETAILS

Each Team member added content to the body of the paper and helped to create any visual aids displayed here. There also was significant overlap between each of our work, and nothing was completed solely by one person. Below are some substantial, specific contributions made by each member:

- Caleb Ruby
    - Did a significant portion of the proof reading and subsequent editing for the entire document
    - Led the creation of the dynamic state diagrams
- Caleb Massey
    - Wrote the initial draft for the majority of this document
    - Led the creation of many the functional and interface diagrams
- Caleb Rachocki
    - Fleshed out many of the sections found to be lacking in content and clarity.
    - Created a mock-up website to help us model and visualize the navigation which really helped to make some of the diagrams
- Ibrahim Alani
    - Led creation of the class entity and relation diagrams
    - Led the description of each diagram and the textual explanation of mechanical details.

Appendix: WORKFLOW AUTHENTICATION

I, Caleb Massey, testify that any and all work on this document, as well as future work on the project, was done so by adhering to the requirements description of the document.

SIGNATURE: 

Date: 10/17/24

I, Caleb Rachocki, testify that any and all work on this document, as well as future work on the project, was done so by adhering to the requirements description of the document.

Caleb Rachocki

SIGNATURE: 

Date: 10/17/24

I, Caleb Ruby, testify that any and all work on this document, as well as future work on the project, was done so by adhering to the requirements description of the document.

SIGNATURE:

Date: 10/17/24 

I, Ibrahim K. Alani, testify that any and all work on this document, as well as future work on the project, was done so by adhering to the requirements description of the document.

SIGNATURE: 

Date: 10/17/24