**Flash Carma**

by Adrian West, Brandon Sitz,

Joseph Ramos-Garcia, and Josh Moore

### 

### Evolution of Flash Carma

Since our initial proposal, Flash Carma’s design and features have been mostly consistent with the exception of a small number of changes. Some minor changes that occurred were changes in priority, such as the search feature, which consists of a text box and button in the navigation bar that would allow the user to search for the title of a study deck from their collection of study decks. Instead of being implemented during the school semester- which was our initial plan- the search feature will be added after the semester has finished. The “forgot password” feature- which is a clickable link on the Login Page that will direct the user to a page dedicated for changing their password and will likely use some form of email verification- that was mentioned in our requirements document has also had its priority changed. The “forgot password” feature will now be added after the semester instead of during it. The most significant change to our project was a change in Flash Carma’s user interface. In our proposal we planned on having a significant number of dialogue boxes that would pop up and give the user some information or provide the user with some capability. An example of this was the method in which the user would create a new profile. Our initial idea was that the user would click a button on the Login page, and a dialogue box would appear with fields for the user’s information (e.g., email address, password, username, etc.). However, the more this matter was discussed, it became evident that abundant dialogue boxes would likely make the process more difficult for the user and lower the user’s enjoyment instead of raising it. So, alternatives to the excessive dialogue boxes were explored and changes were made. In the case of creating a new profile, we decided to keep the button but direct the user to another page designated for creating a new profile.

### What We Have Accomplished

The pages for Flash Carma use a variety of functions that add new data to our database, retrieve data from it, or modify the existing data within it. Implementation of the functions for user authentication, creating a new user in the database, retrieving study decks from the database, retrieving flash cards from the database, deleting a study deck, and deleting a flash card have all been completed. The functionality of the Results page is complete. The only thing that remains for it is to improve it aesthetically. The Navigation Bar, Login page, Sign Up page, Edit Study Deck page, Study Session page, and View Study Decks page are all partially complete. The Navigation Bar is fully functional; however, it does still require an updated copy of the Flash Carma logo and for the routing to the other pages to be implemented. The Login page, Sign Up page, Edit Study Deck page, and Study Session page all need implementation for particular functions that interact with the database. Currently, certain visual aspects for the Edit page and Study page are finished, however, the database interaction is still being worked on. The View Study Decks page has pseudo code and some functions implemented but is still being implemented at this time.

### What We Still Need to Accomplish

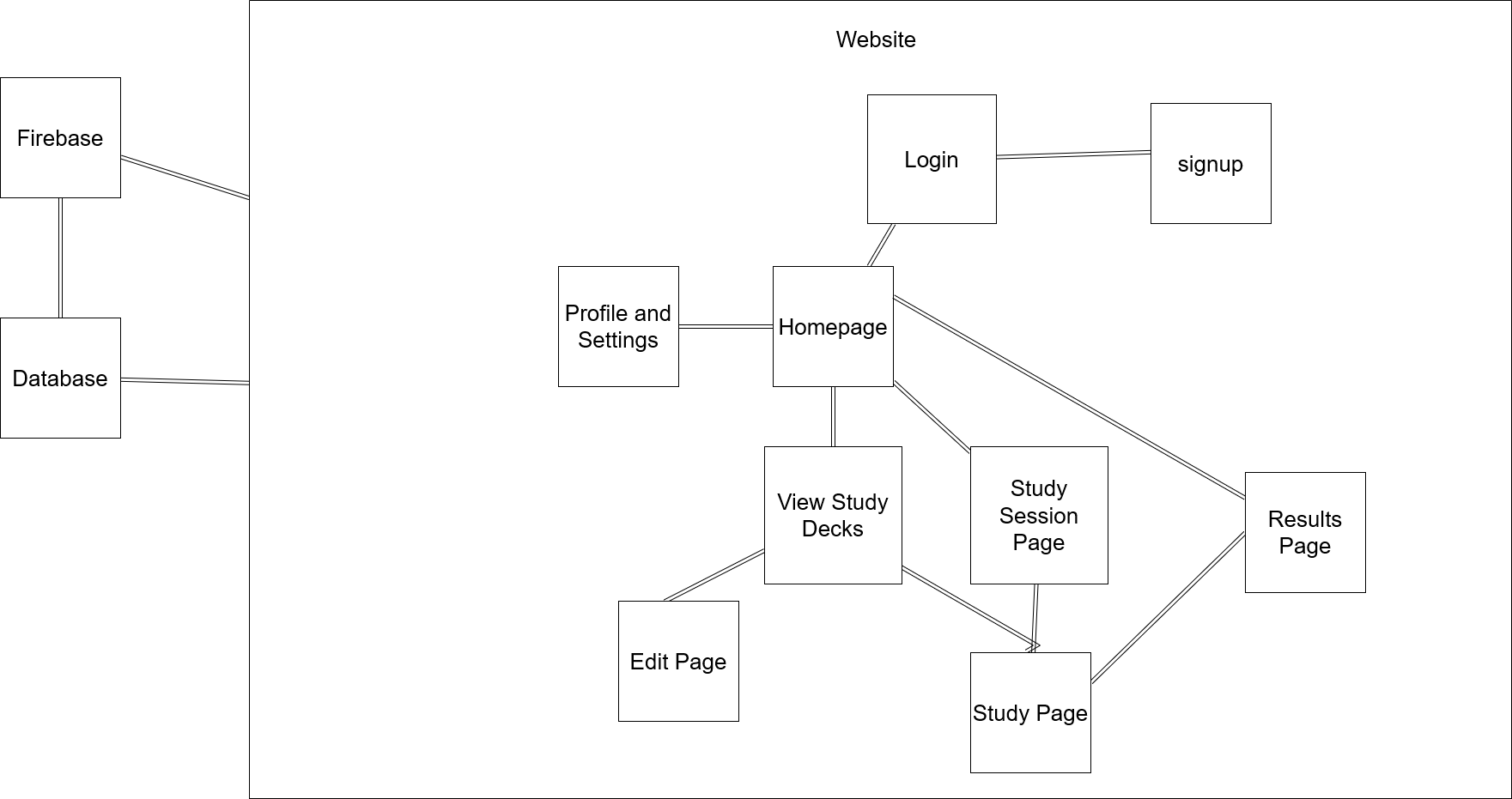
As mentioned earlier, the View Study Decks page is currently being implemented. Once completed, using the View Study Decks page, the user will be able to create a study deck by clicking a Create Study Deck button. If the user clicks on an already-existing study deck, there will be a drop down with three options. These options will allow the user to Edit, Study, or Delete a study deck. When the user clicks Edit, they will be sent to the Edit page, where they will be able to add new flash cards or delete existing flash cards within that particular study deck. The Edit page will display the study deck’s name. If the user wishes to create a new flash card, there will be two textboxes where the user can fill in entries for the front and back sides of their flash card. There will be a button for the user to click in order to add the flash card to the study deck. After the button is clicked, the new flash card will be displayed with the others in the study deck, and the database will be updated to include this new flash card. The user can also click a button on an individual flash card to delete said flash card. The function that deletes the flash card will use the flash card’s ID to delete it from the database. On the View Study Decks page, if the user clicks the Study option for a study deck, they will be sent to an instance of a study session that uses the flash cards from that study deck. At any given moment during a study session, the Study page will display the study deck’s name, the user’s current score for the study session, what number flash card they are on out of the total number of flash cards in the study deck, a button to stop the study session, the current flash card, and two buttons for whether the user got the question correct or incorrect. If the user clicks the flash card, an animation of the flash card flipping will play, and the other side of the flash card will be displayed. If the user clicks the Correct button, points will be added to their score for the study session. If the user clicks the Incorrect button, their points will not change. Once the user clicks either of these buttons, the next flash card in the study deck will be displayed. If the user clicks Correct at least twice in a row, a Correct Answer Streak will begin, and an icon indicating this streak will appear beside the number of the flash card that the user is on. Inside the icon will be the number of consecutive correct answers they have given. If the user clicks either Correct or Incorrect on the final flash card, they will be directed to the Results page, which will show them the results of their study session. If the user clicks Incorrect, they will lose their streak. When the user either finishes or cancels the study session, the user’s score will be added to their cumulative score, and the results for the user’s session will be displayed in the Results page. On the View Study Decks page, if the user clicks Delete, the selected study deck and its cards will be deleted.

A few other pages for the application still need to be implemented. The Profile Page, which will display the user’s information, allows the user to change their information and possibly edit settings related to how the user interface is displayed. Additionally, the Study Deck Selection Page, which will allow the user to select a study deck for a study session, still needs to be implemented. The Study Deck Selection Page will have a similar format to the View Study Decks Page but will have slightly different functionality. A few functions associated with the database also need to be implemented, including updating a flashcard, updating a study deck, retrieving user information for the profile page, and updating user information from the profile page. The majority of the database functions are complete. However, the implementation for the pages interacting with the database still needs to be done.

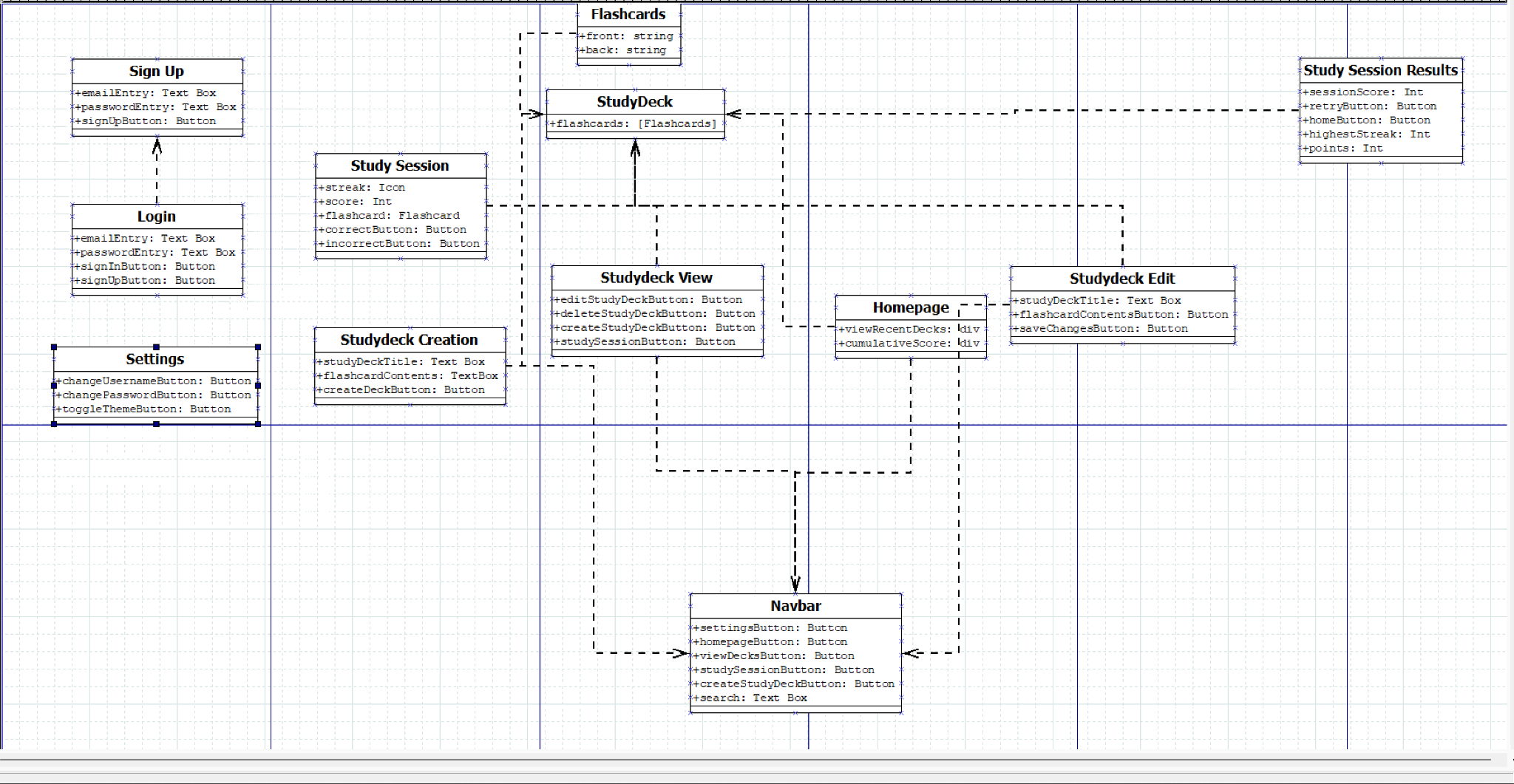
### Unresolved Issues

One issue that will have a major impact on the project’s success is passing data between separate pages. An example of this is passing the user’s study session score, and the user’s correct answer streak from the Study Page to the Results Page. Since the Results Page is only accessed after completing a study session, one potential solution to this issue is to make the Results Page a child component of the Study Page. The Study Page would then pass the necessary data as properties to the Results Page, which would be rendered on the Results Page. Another example of this issue is passing the selected study deck to another page. One solution to this problem is to use the React Context API. The React Context API allows us to store the study deck name in a global state that any component can access. Once a study deck is selected, we pass the study deck id to the global context, and the selected page will have access to the selected study deck.

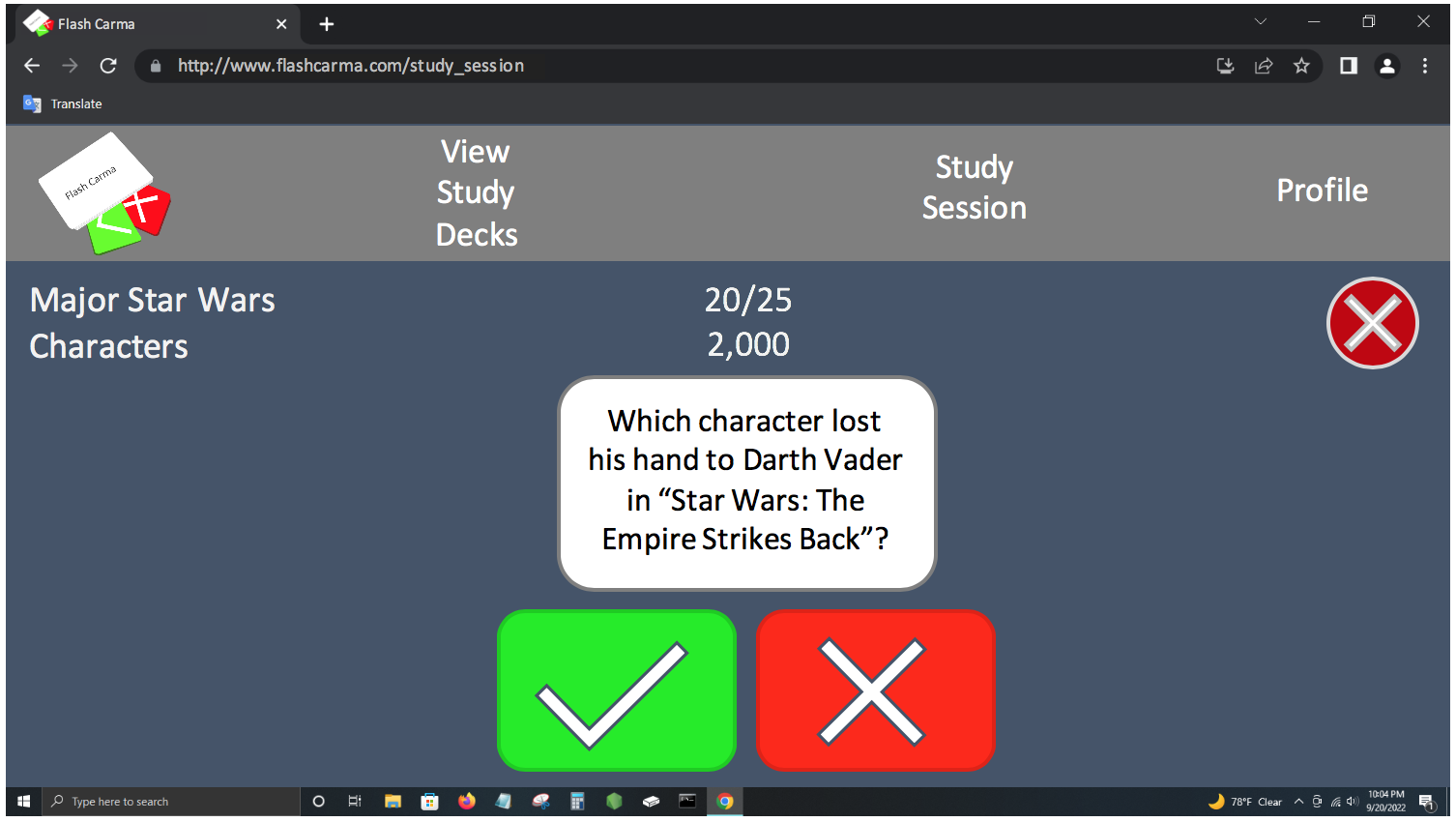
### Updated Charts and Diagrams



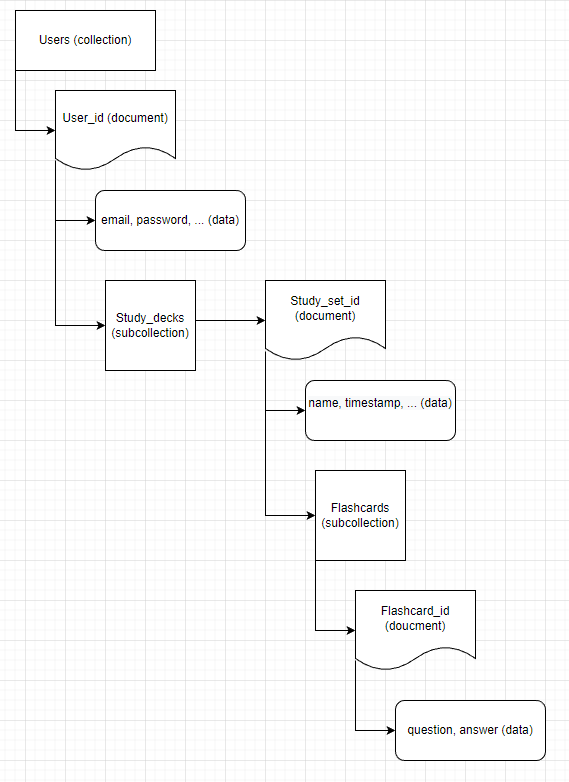
*Figure 1: Updated Block Diagram*



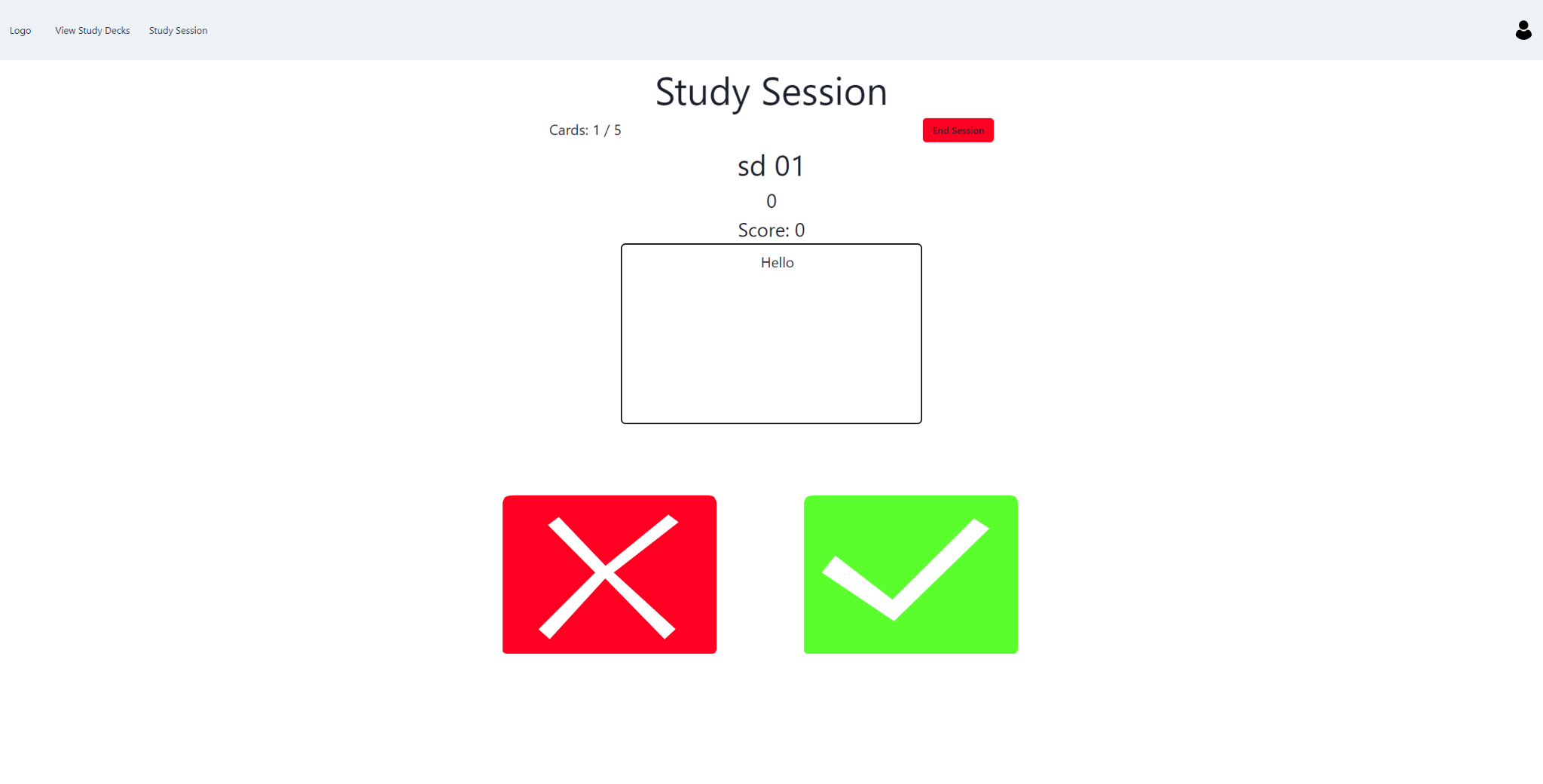
*Figure 2: Updated Component Diagram*



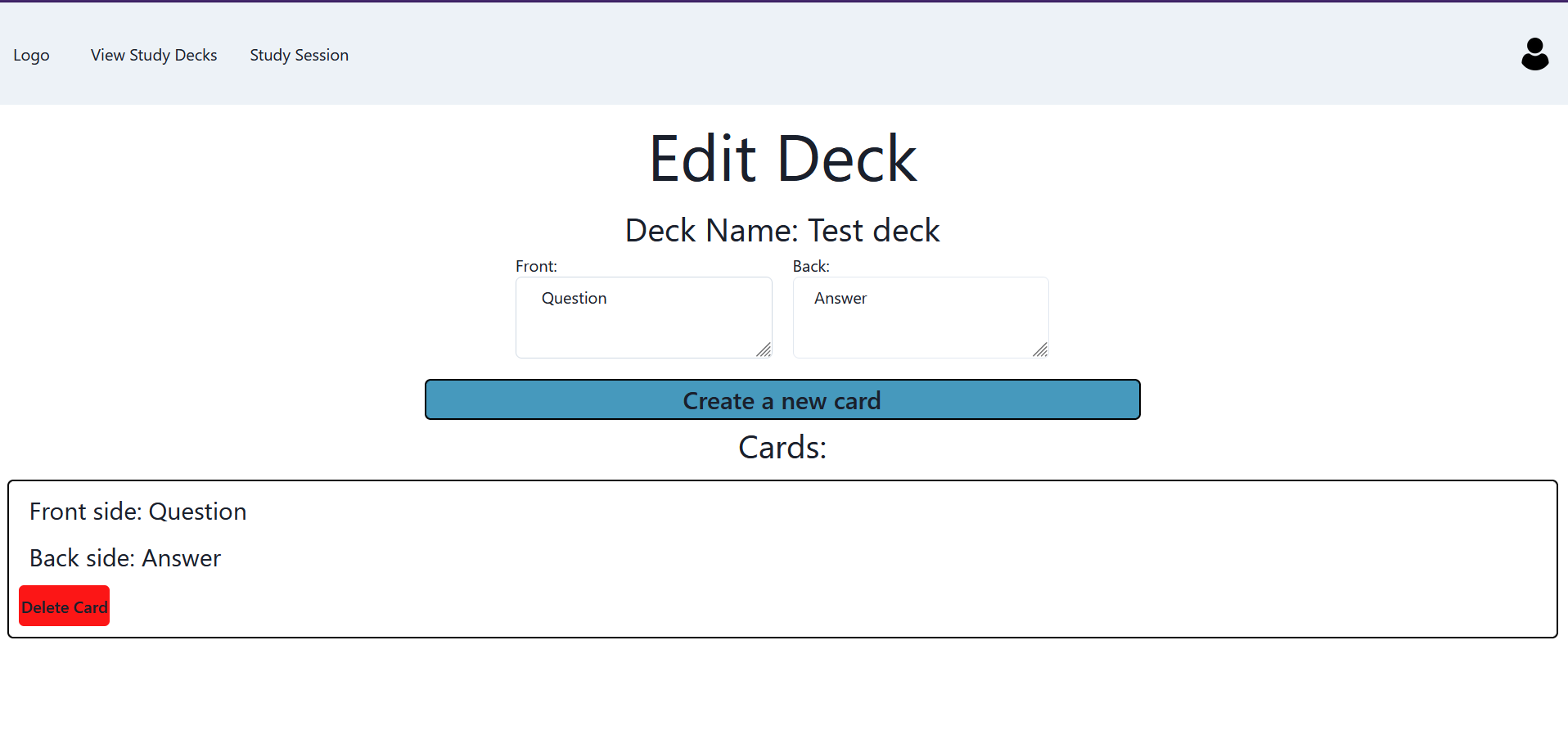
*Figure 3: Updated User Interface Storyboard*



*Figure 4: Updated Data Model*



*Figure 5: Visual for Current Study Page*



*Figure 6: Visual for Current Edit Page*