**SCHOLAR STACK REQUIREMENTS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Main Task | Sub-Tasks | Developers | Estimated Time(%) | Comments | Date Completed |
| Figma Design Phase | -Figma design for all the pages consisting of the, login page,  Home page,  upload books,  books,  books details,  edit book page,  profile page,  edit profile page,  community,  messaging,  contact us page | Kruti, Diya, Ekom | 10% | - |  |
| Login and Authentication Page | -There should be a login page with options to sign in with google, facebook and github  -There should be a functional sign up page  -Setup functionality for clerk(handles authentication) | Jubril, Kelvin, Daniel | 7% |  |  |
| Home Page | -The navigation bar should include the logo, the names of pages(Home, Books, Community, Upload Books, Profile, Contact Us), user account picture  - A hero section to introduce ScholarStack’s apps purpose -It should have a section for the recently uploaded books on the app and an option to see more  -A section for the books on sale and an option to see more  -A section for the list of free books and an option to see more  - Functionality to be implemented when the see more is clicked, the filters should be preselected in the books page  - Choosing a book from the sections should redirect the user to the book details page  -Actions to retrieve the free books, recently uploaded books and on sale book filters from the database |  |  |  |  |
| Upload Books Page | -A functional form to submit the details of books into the database and app  - Book title form,  - checkmark if the book is free  - Normal Price  - Sale price(optional)  - Book description  - A calender to pick the date uploaded  -a drop down menu to select categories  -A form to add the images from their local source  -A drop down menu to select the languages  - A form to get the location of the book/user  - A submission for the form |  |  |  |  |
| Books Page | -Create database models for the database such as user, books, order, category, language models  - Implementation of the connection to the database and cached storage  - Database actions to retrieve the each book’s info  - Actions created to retrieve all the books from the database  -It should have a side bar with filters including price(low to high, high to low etc), free checkmark, on sale, category, language  -A search bar to locate books based on their titles  - There should be a card consisting of the main features of the book(price, title, location, free tag(if it is free), category) that is being displayed  -Page should consist of a list of book cards that are in the database  -Pagination should be implemented when all book cards do not fit in a page(so page 1, 2, 3 etc)  - The selection of each card should redirect the user to the book details page  - The option to edit a book on the card should be displayed if the user currently logged in uploaded that book which links to the edit book page |  |  |  |  |
| Books Details Page | -Display all the information regarding the book  -Display the images of the books  -The title, book description, price, free(if it is free), sale price, language, category, date uploaded etc  -A message seller option to communicate with the seller that redirects to the messaging page  -A section to display similar or related books(based on the category)  -Backend actions to retrieve the information of the specific book.  -Backend action retrieve similar books based on the category from the database  -option to edit the book details which redirects to the edit book page. |  |  |  |  |
| Edit Book Page | - The same form from the upload books page should be used  - The page should have the options preselected and inserted with the original information in the database to be edited  - A button to submit the edited book |  |  |  |  |
| Profile Page | -Actions to retrieve the specific user’s information from the database  - The image of the user should be displayed  - Display the name, username, location, bio, date joined etc  - A user should be able to edit their profile(edit profile button) which redirects to the edit profile page(only display if it is the logged in user’s profile page)  - A section consisting of the books uploaded by that user  - Database action to retrieve the books uploaded by a user  - A section to display a list of other users in the community  - An option to see more that redirects the user to the community page  - The section should display the users with their first name and last name |  |  |  |  |
| Edit Profile Page | -This page should include a form for the name, username, location, bio option  -Backend action to retrieve the name, username, location, bio from the database  -The form needs the user info to be pre inserted into the form from the database information  -A button to save the information  -Save the information into the user database |  |  |  |  |
| Community Page | -A search feature to find users based on their name  -Action to retrieve all the users from the database  -Display each user card consisting of the profile picture, first name with the last name, username, number of books uploaded etc  - Functionality to redirect a user to the person’s profile page.  - Pagination should be implemented in case all the users cannot fit in one page |  |  |  |  |
| Messaging Page | -Section to display the names of the user’s messaged(my conversations)  - A right section to display the conversation between the users when clicked(from the my conversations section)  -Each message(from both seller and buter) should include the first and last name with their profile picture and the contents of the message sent  -An input section to type the content of the message to be sent(design as needed)  - Save the messages in the database  - Creation of message model in the database  -backend to synchronize the sending and receiving of messages  -notification system to be implemented |  |  |  |  |
| Contact Us Page | -Form for contact us(design accordingly) -Implementation for storage |  |  |  |  |