

College Sale: Online Platform to Sell and Buy

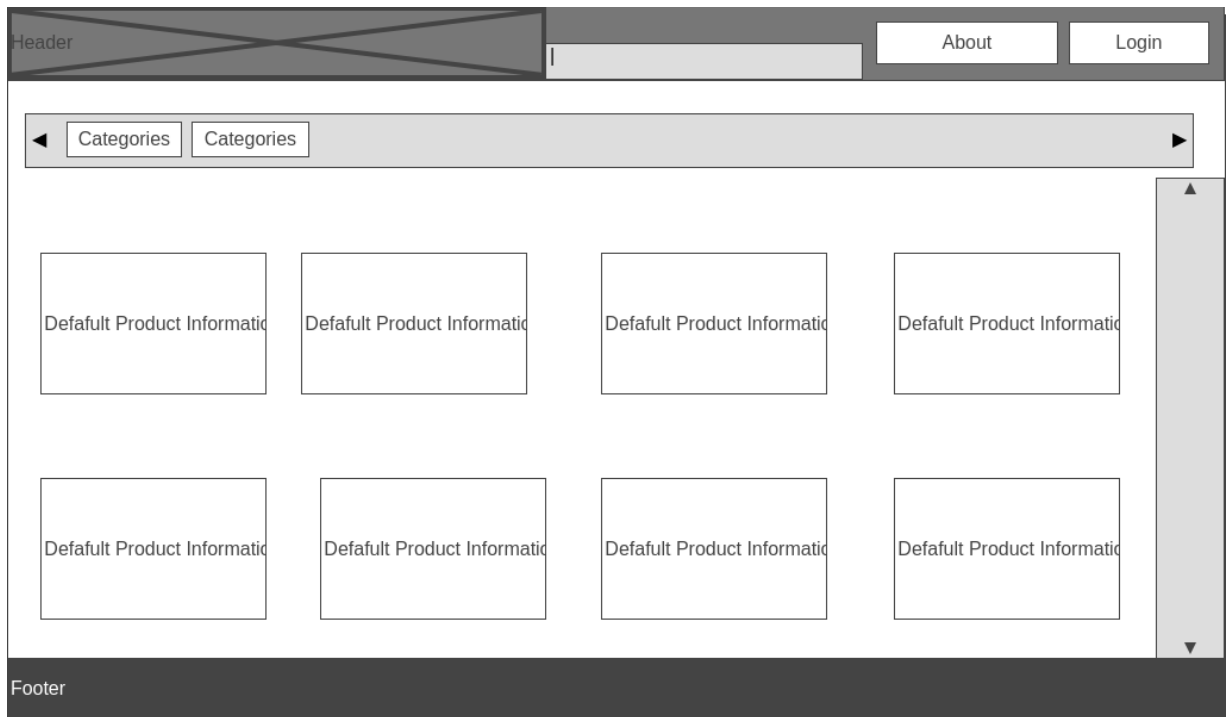
1. Introduction

The idea is to create a portal that would allow university students to sell and buy stuff online. They can use this platform to get connected with other university students living on campus or off-campus to sell second-hand things. People can put their stuff online no longer needed by them or in case they are moving to another town. This could help to build an online marketplace that would encourage students to reuse essential items. It would be easier to kind of exchange and negotiate with students in the same university and make good connections. It would help students who are looking forward to saving money by reusing things required by them in their day to day life.

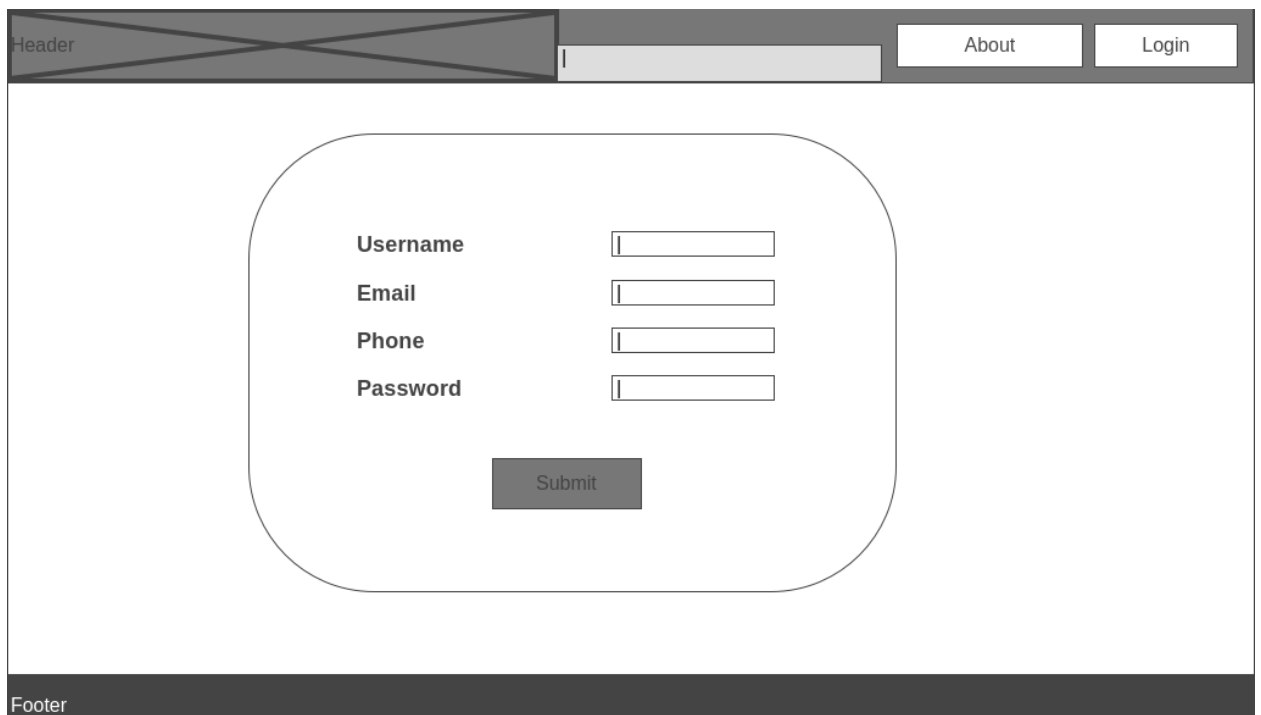
2. Design and Implementation

- Home page – the default page which would pop up. It would have a search bar in the header along with about me page and a button which would redirect user to go to login page
- Registration Page – a simple form which would enable user to fill in details to get registered on the website.
- User's Home Page – it is just a authenticated Home page with User's session. It would have a body with details about different products which have been put on the website for selling purpose.
- Product Page – if a user clicks on a certain product, it would open a new page with product information and a way to contact the owner of the product.

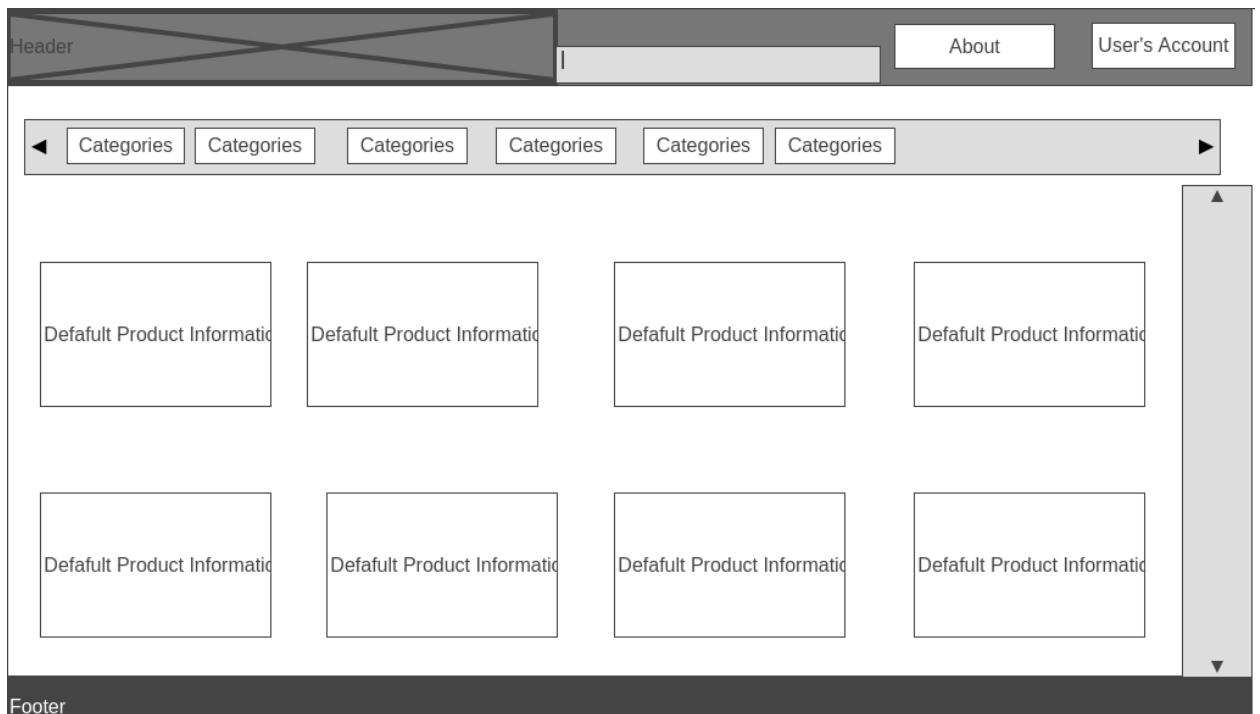
For now, I have only created 5 wireframes. It would also include a way for users to add information about items or things to sell and put it on the website. Also, a login page with just the username and password to fill in. Upon Login button on the website's home page, it would ask user to register if they are a new user.



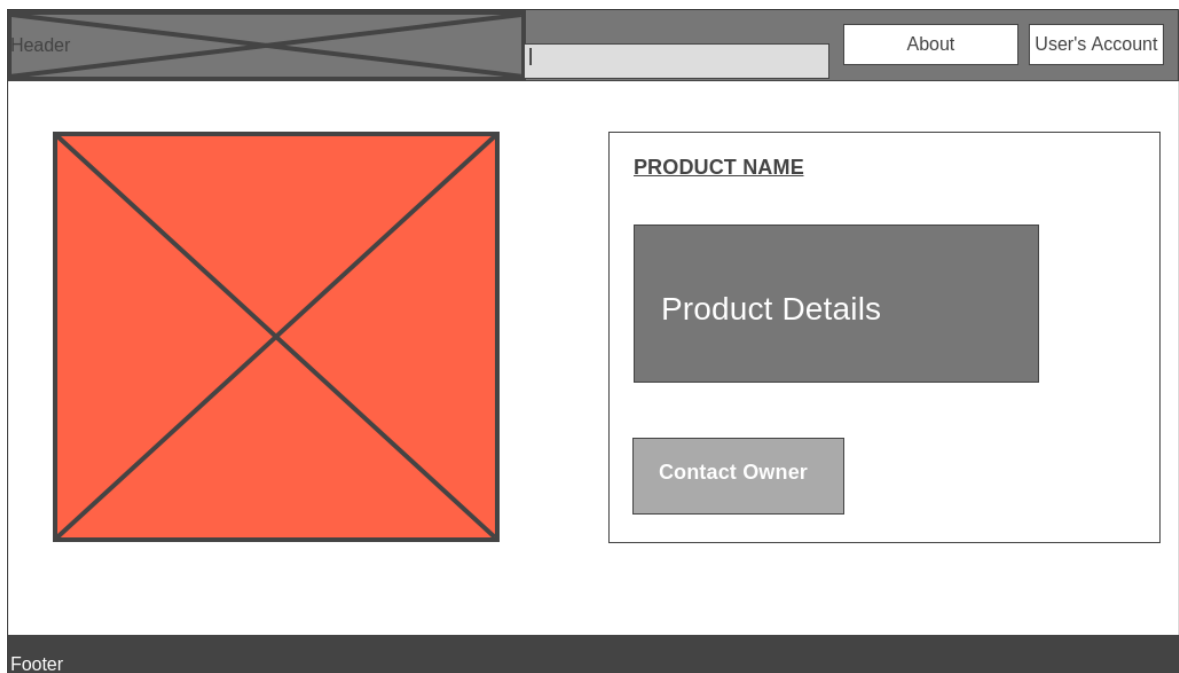
Home page



Registration Page



User's Home Page

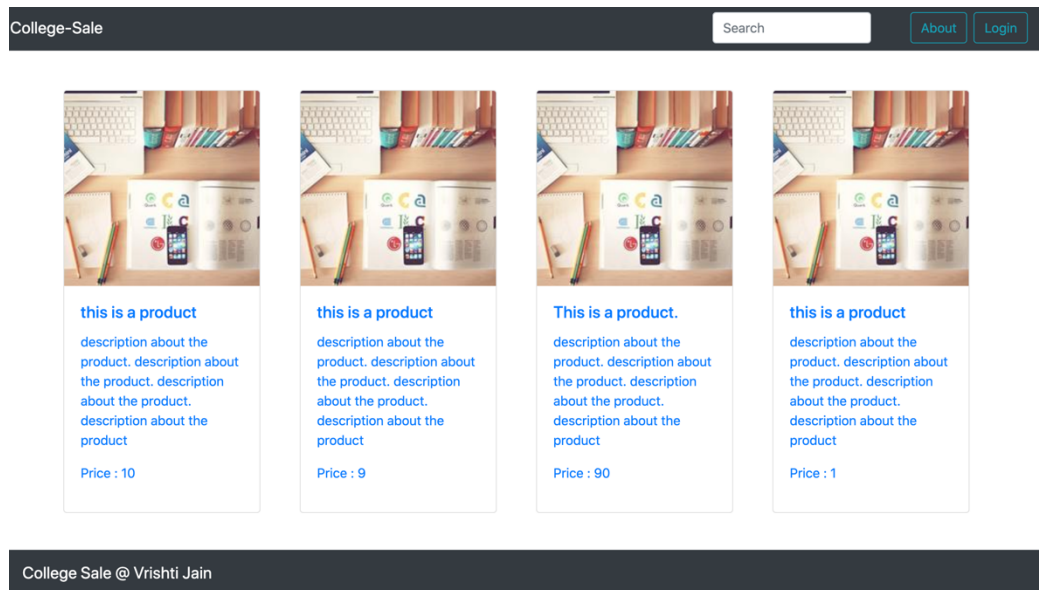


Product page

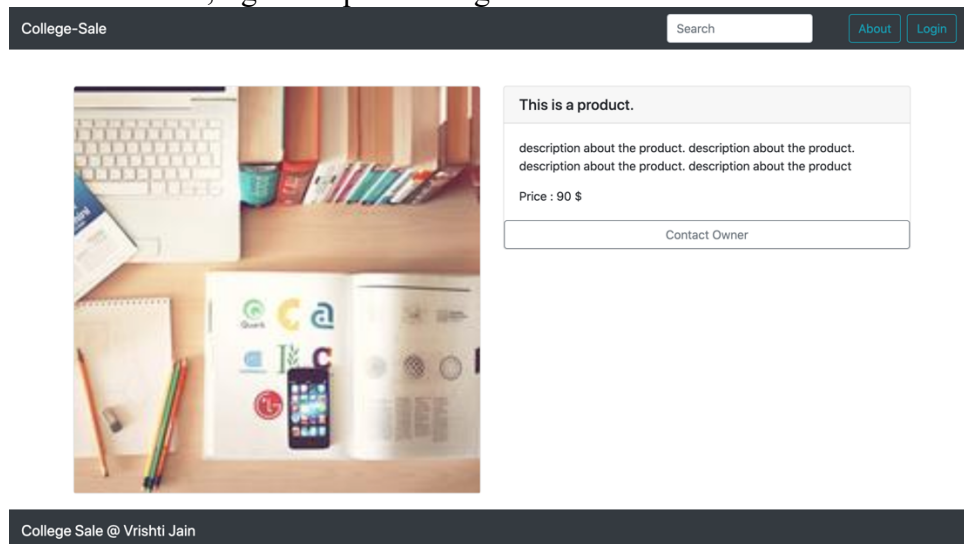
In terms of implementation. I began with dividing my projects into different components and different pages.

Home page: This page consists of various products that have been added by reading the details from a JSON formatted object. Here I placed the random images in Cards as I did not have a database to begin with. I have also added header and footer in the main component so that it would be common in all the components.

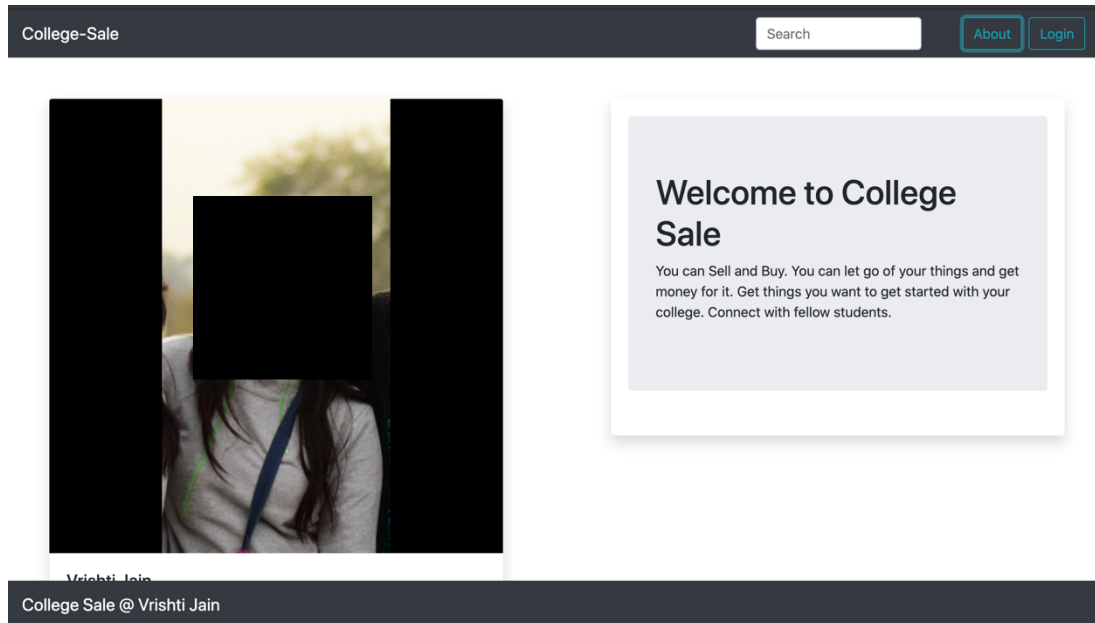
The header consist of a search bar in which users can search and About and Login button which would lead users to respective pages. I used React Bootstrap for all the elements. And also for routing I used react-router-dom.



Product Detail Page: When user clicks on one of these products to explore. They are redirected to a page with information about the product with given product ID. Here I also used Container with one row and two columns. One column for the image of the product and one for its detail part. For both the columns, I gave required margins and used Card Element for both of them.



About page: If a user clicks on About on the navigation bar , they gets redirected to about page with information about the owner and the purpose of the website. This was just general information page. I used Cards again to display information about the owner and Jumbotron from react-bootstrap as it provides user with the key content. It helps in gathering user's attention to this particular component.



Login or Register Page: If the user clicks on the Login button, they are redirected to the Login Page. Here I used Accordion as it allows user to provide a way to restrict Card components to only open one at a time. This is what I intended with the Register and Login. Because the user would either register or login. Initially I thought to make different pages for both of them but while searching components for toggling, I came across this Bootstrap component. Basically If your registration form is open, then the login form is closed and vice versa.

The screenshot shows the 'College-Sale' website header with a search bar and 'About' and 'Login' buttons. Below the header is a 'Register' section with a light gray background. It contains four input fields: 'User Name' (placeholder: 'Enter Username'), 'Email' (placeholder: 'Enter Email'), 'Phone' (placeholder: 'Enter Phone number'), and 'Password' (placeholder: 'Enter Password'). A blue 'Submit' button is located below the password field. At the bottom of the register section is a 'Login' link.

The screenshot shows the 'College-Sale' website header with a search bar and 'About' and 'Login' buttons. Below the header is a 'Login' section with a light gray background. It contains two input fields: 'User Name' (placeholder: 'Enter Username') and 'Password' (placeholder: 'Enter Password'). A blue 'Submit' button is located below the password field. At the top of the login section is a 'Login' button.

3. Conclusion and Results

I have been successfully able to integrate various feature of react together with react-bootstrap to serve the College-Sale project. I have been thinking to implement backend with flask server and MongoDB as Database. As Mongo would allow me to store data in JSON formatted way. It would be efficient in serving the application. But it had been very time consuming to work on both the frontend and the backend simultaneously. I have used JSON formatted object as initial database. I currently do not have a backend server as I invested time in designed UI. Some of the features I would be implementing in the future are:

~To Add a way for users to add product details: This is something that completely went over my head and I forgot to include in the basic requirements as well.

~ Right now, I did not maintained session for each users which is needed for customer based websites. Also, it would require me to have a secured database.

~ I have to build schema of the system. Right now, it's just JSON formatted object. The full implementation would require a secure and reliable database.

~ I intend to implement Flask for the backend and connecting it with the react. That would require me to gain some knowledge about Flask-React integration. I have worked on Angular-Flask stack. But react would require some knowledge to begin with. Flask is easy to integrate with web-based application.

4. References

- <https://wireframe.cc/> - for building the prototypes.
- <https://picsum.photos/> - for generating random images
- <https://react-bootstrap.github.io/> - for different bootstrap elements.