

Hokie For U

Final Product Report



Virginia Tech Department of Computer Science
Connecting Hokies, facilitating jobs, and fostering a vibrant community.

Group 11

Akshay Reddy Narra

Charan Teja Chelle

Samhitha Pentaparthys

Siva Kumar Reddy Sangu

Siva Sagar Kolichana

Sushma Kumari Kakarla

CS 5934 | Fall 2023

Product Description

HokieForU is an innovative platform designed for the Virginia Tech community, providing a space for Hokies to both seek employment opportunities and actively engage in community service. It goes beyond a typical job portal by fostering a spirit of collaboration and support among Hokies. Users can offer or find a wide range of services, from household tasks to tutoring, creating a network that benefits both job seekers and local communities. By connecting Hokies with local needs, HokieForU aims to empower individuals to make a positive impact on their neighborhoods while also facilitating employment opportunities within the community. It's a unique blend of job platform and community service hub, reflecting the diverse talents and generosity of the Virginia Tech community.

Product Functionalities

Home Page:

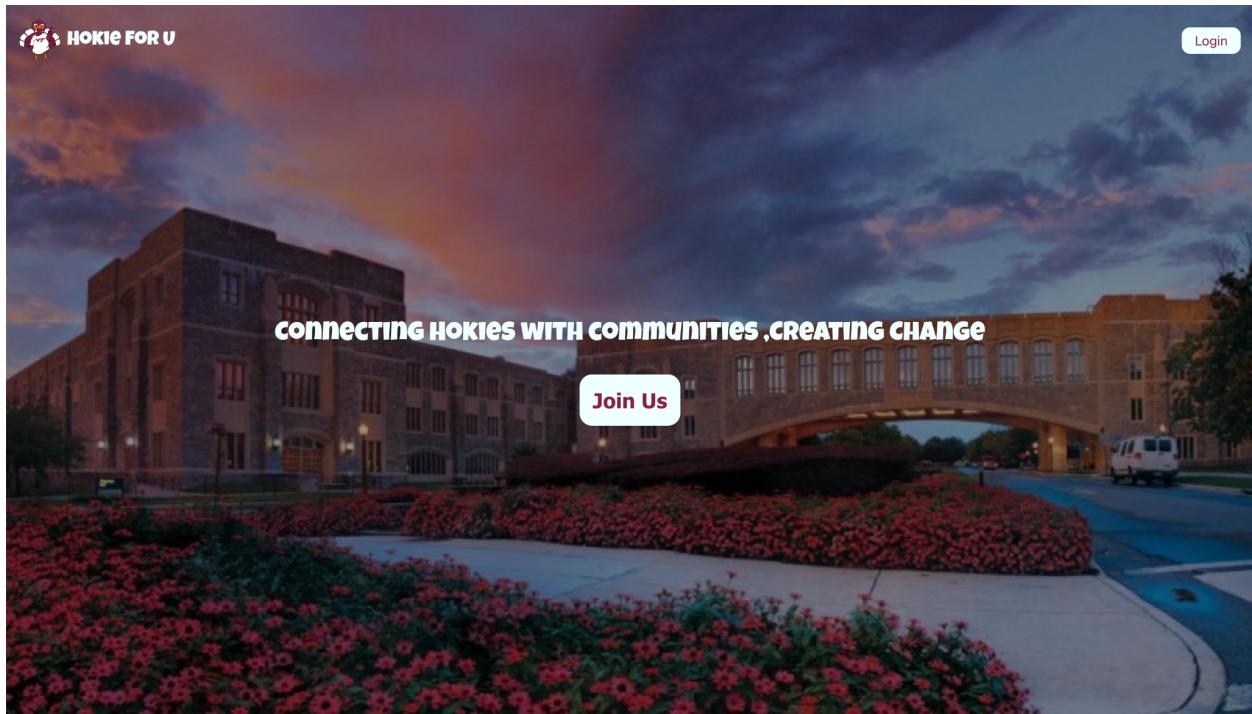


Figure 1: Landing page of the website

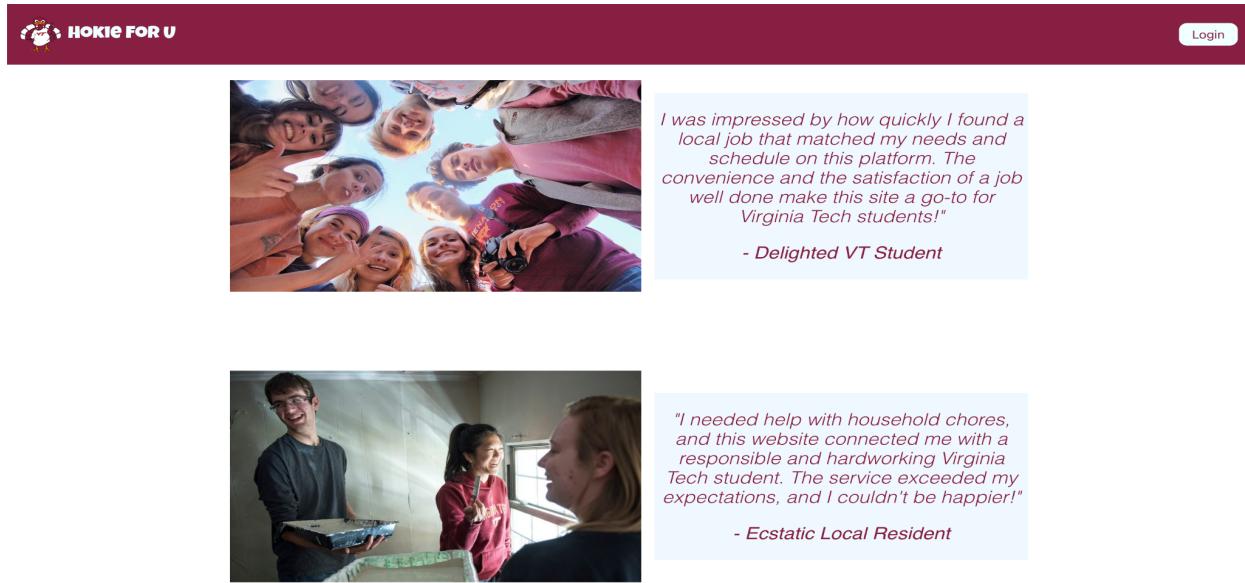


Figure 2: Stories on Home Page



Figure 3: Contact Information and Links for Social Media Websites

The three images presented above depict the central hub of the application, which is the homepage. This page serves as the virtual gateway to the application, encapsulating its primary mission and offering insight into the core functionalities it provides. On this homepage, you'll find compelling stories that narrate how this product has positively impacted not only individuals but also the communities it serves. These real-life anecdotes vividly illustrate the practical

significance of the application, shedding light on the tangible benefits it brings to people's lives. This storytelling element not only informs users about the application's capabilities but also fosters a sense of connection and trust, showing that it has already made a difference in various ways.

In addition to the impactful stories, the homepage features quick links to the application's social media profiles like Facebook, Twitter, Instagram, and LinkedIn, making it easy for users to stay connected and engaged. These links facilitate social interaction and keep the user community informed about updates, news, and events related to the application. Moreover, users can seamlessly access the application by either logging in if they are existing users or registering if they are new to the platform. The inclusion of these login and registration options on the homepage ensures a user-friendly experience and demonstrates the application's commitment to being accessible to a broad audience. To further enhance user engagement and support, contact information in the form of a phone number and email address is readily available, enabling users to reach out for assistance or inquiries.

Registration Page:

The registration page for Hokie For U. The page has a maroon header with the logo 'HOKIE FOR U' and a 'Login' button. The main content area has a background image of a stone wall with the Virginia Tech VT logo. The registration form includes fields for Full Name, Username, Email, Phone Number, Password, Confirm Password, and Gender (Male, Female, Prefer not to say). A red error message 'Please fill out this field.' is displayed above the gender selection. A large orange 'Register' button is at the bottom right.

Figure 4: Registration page for Hokie For U

The registration page, on the other hand, is geared towards individuals who are new to the application and wish to create an account. During the registration process, users provide essential information such as their email, username, and a password of their choice. This page is the

gateway to joining our user community and benefiting from the features and services the application has to offer. Additionally, users have the convenience of registering or signing in with their Google accounts, streamlining the process and enhancing user-friendliness. By keeping these pages separate, we maintain clarity and ease of use for both new and returning users, ensuring a positive experience at every stage of their interaction with our application.

Login Page:

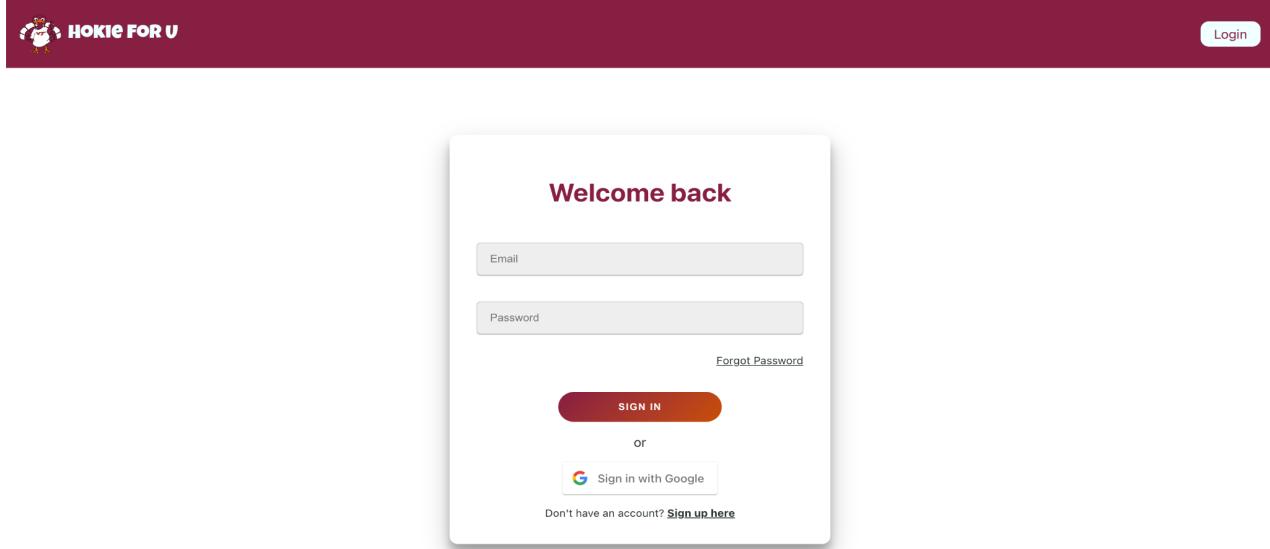


Figure 5: Login Page for Hokie For U

Our application has distinct pages for both login and registration, each serving its unique purpose. The login page is designed to offer existing users a secure and straightforward means of accessing their accounts. On this page, users enter their registered credentials, including email or username and password. Furthermore, we've implemented an additional feature that allows users to reset their password in case they forget it. This provides a safety net for users who may encounter login issues, ensuring a smooth and frustration-free experience.

Dashboard Page:

The dashboard page functions as the central hub in a job listing application, providing users with a comprehensive overview of all available job opportunities stored in the database. A significant feature is the "Post a Job" button, empowering users to contribute new job listings to the platform. This interactive feature encourages users to actively participate in the job listing community, expanding the range of available opportunities.

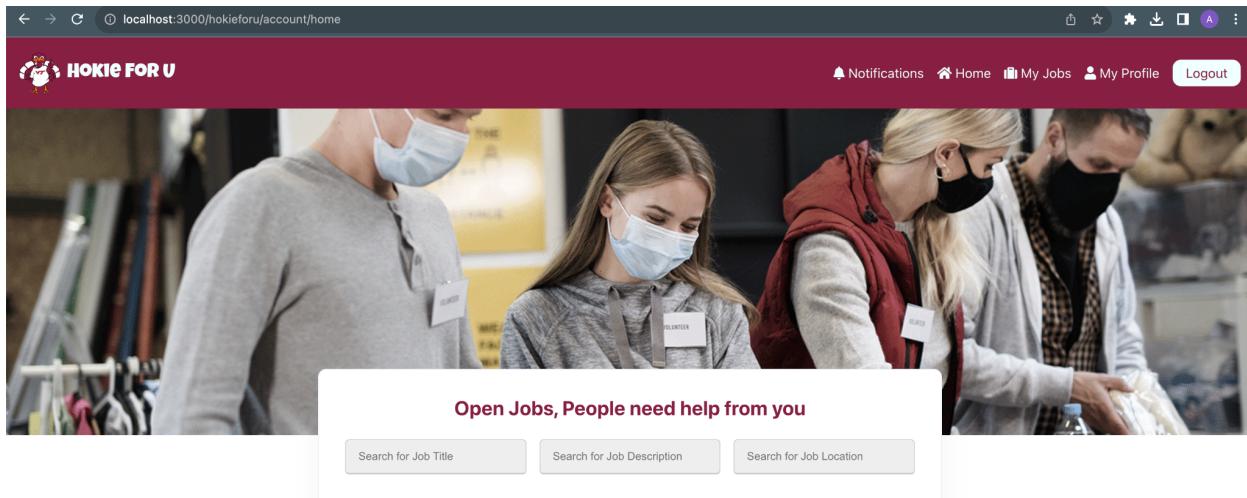


Figure 6: Dashboard page containing all available jobs

This screenshot shows the same dashboard as Figure 6, but it includes a section titled 'Available Jobs' below the search bar. It displays three job listings in boxes:

- Community Center Service Work**
Location: Blacksburg Community Center, VA, 24060
Pay: \$20/hour
Job Date: Not Provided
Posted By: akshayreddy@vt.edu
Job Details: Need help for a weekend game
[Apply this Job](#)
- Babysitter / Nanny**
Location: 820 Patrick Henry Drive, Blacksburg
Pay: \$9/hour
Job Date: Not Provided
Posted By: sivasagar@vt.edu
Job Details: I have meeting on 11/30/2023 but I cannot leave my kid alone. he is a great kid. He is quiet and
[Apply this Job](#)
- Community Center Service Work**
Location: Blacksburg Community Center, VA, 24060
Pay: \$12/hour
Job Date: 12/2/2023
Posted By: akshayreddy@vt.edu
Job Details: Testing
[Apply this Job](#)

A 'Post a job' button is located in the top right corner of the 'Available Jobs' section. At the bottom, there are navigation arrows and a page number indicator (<< 1 2 3 >>).

Figure 7: Dashboard page showcasing about Filter Criteria and Pagination concepts

Each job listing is presented in an individualized box format, containing essential information for potential applicants, such as the job title, location, pay, and date of posting. Additionally, a detailed job description is provided to offer users a deeper understanding of the available roles. Including the email and contact details of the user who posted the job facilitates direct communication and streamlines the application process.

To improve the user experience and simplify job searches, the dashboard page incorporates filter criteria. Users can input specific terms like job title, description, or location to customize displayed results according to their preferences. This thoughtful integration of search filters not only saves time but also ensures users can quickly and efficiently find the most relevant opportunities.

Post a Job Form Page:



The screenshot shows the "Compose a Job" form on a website. At the top left is the logo "HOKIE FOR U" with a cartoon owl icon. At the top right are links for "Notifications", "My Jobs", "My Profile", and "Logout". The main title "Compose a Job" is centered above the form fields. The form includes fields for "Job Title" (with a placeholder "Enter job title"), "Job Description" (with a rich text editor toolbar and a placeholder "Enter Job Description"), "Contact Details" (with a placeholder "Enter Phone Number"), "Job Location" (with a placeholder "Enter Job Location"), and "Job Pay" (with a placeholder "Enter Job Pay").

Figure 8: Post a Job Form

Our application features a "Post a Job" page with a user-friendly form for job listings. This form includes fields for title, description, date, pay, and location, providing essential job details. When submitted, the data is securely stored in our database, streamlining the job posting process and making it easier for both job posters and job seekers to manage listings and find suitable opportunities. This page serves as a key feature, supporting efficient data management for a seamless user experience.

Profile Page:

The "Update Profile" functionality is a critical feature within our application that allows users to modify their personal information. This feature enhances the user experience by enabling them to keep their profile details up to date. The "Update Profile" page comprises several fields,

including First Name, Last Name, Contact Number, Email, and Profile Picture, each of which contributes to a comprehensive user profile. Users can edit their First Name, Last Name, and Contact Number to ensure that their personal information remains accurate. To initiate updates, users are presented with input fields for each of these attributes. A well-designed form structure guides users through the process, with error handling and validation to guarantee the integrity of the data. The "Update Profile" functionality allows users to view their current email address but does not permit them to modify it. This ensures the security and consistency of user accounts. In the case where email address changes are necessary, users can utilize an alternative method, such as contacting support or customer service. Users have the option to upload or update their profile picture. This image provides a visual representation of the user and personalizes their profile. The file upload feature supports various image formats, ensuring flexibility and ease of use.

The screenshot shows the 'My Profile' page of the HokieFORU application. At the top, there is a header bar with the URL 'localhost:3000/hokieforu/account/myprofile'. Below the header is a dark red navigation bar containing the 'HOKIE FOR U' logo, notifications, home, my jobs, my profile, and logout links. The main content area has a white background. It features a 'My Profile' heading. Below it are four input fields: 'First Name' (Akshay Reddy), 'Last Name' (Narra), 'Contact Number' (undefined), and 'Email' (akshayreddy@vt.edu). To the left of the email field is a 'Profile' section showing a cartoon rooster icon with the text 'HOKIE FOR U' and a camera icon. At the bottom right is a 'Save' button.

Figure 9: My Profile page

My Jobs Page:

The "My Jobs" page functions as a personalized area within the application, displaying jobs either posted or selected by a specific user. This page is divided into two main parts: "Posted Jobs" and "Picked Jobs."

The "Posted Jobs" section exclusively exhibits the jobs posted by the particular user. In this segment, users can view detailed information about each job, encompassing the job title, location, pay, and a comprehensive job description. This layout enables users to easily manage and monitor the opportunities they have shared on the platform.

Figure 10: My Jobs page showing both User Posted and Picked Jobs

Conversely, the "Picked Jobs" section highlights jobs chosen or favorited by the user. Similar to the "Posted Jobs" area, the "Picked Jobs" section provides in-depth details about each selected job, including the job title, location, pay, as well as the email and contact details of the user who posted the job. Additionally, a job description is presented to offer a thorough understanding of the selected opportunities.

With these two subcomponents, the "My Jobs" page creates a user-focused experience, allowing individuals to conveniently access and oversee both the jobs they have posted and those they have chosen to pick or favor. This dual categorization ensures a clear and well-organized presentation of the user's engagement with the job platform, facilitating efficient navigation and interaction with the posted and picked job listings.

Email Notifications:

The application incorporates automated email notifications to enhance user engagement and communication. Upon successfully posting a job, the system sends a confirmation email to the user, titled "Your job is successfully posted." This email provides a detailed summary of the posted job, including key details like title, description, pay, location, date, and contact information. This feature ensures users have a comprehensive record of their posted jobs, contributing to a positive user experience.

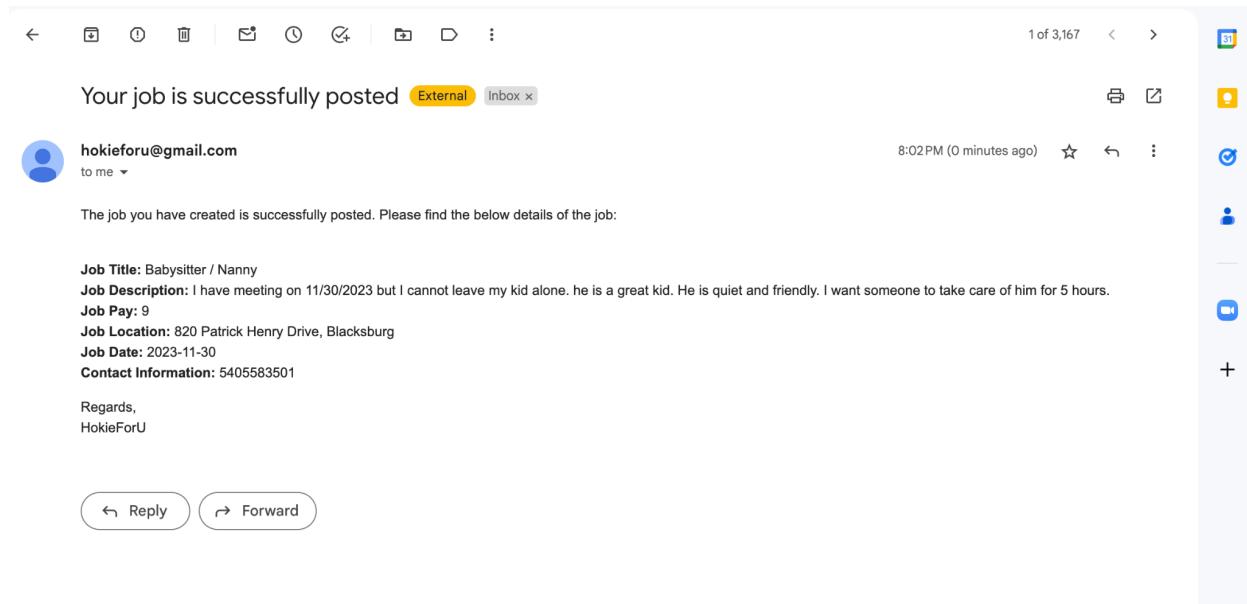


Figure 11: Email when Job is posted

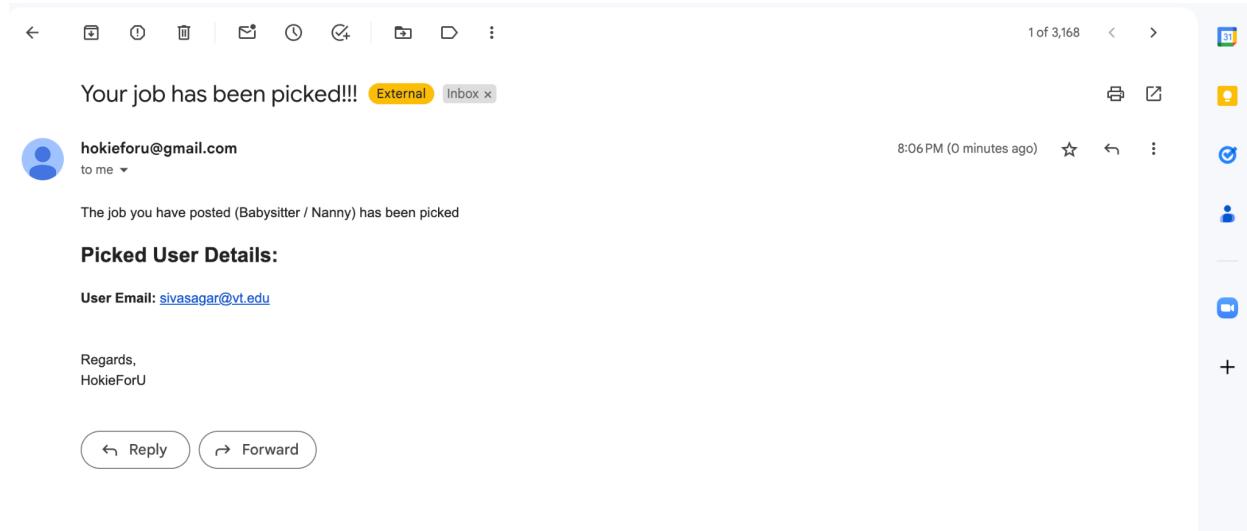


Figure 12: Email when Job is picked

Additionally, when a user's posted job is picked by another user, an automated email alert is triggered with the subject "Your job has been picked!!!". This email promptly informs the original job poster about the selected job, including the title and the user who picked it. This immediate notification fosters efficient communication and transparency within the platform, allowing users to stay informed about the status of their posted jobs.

Both types of emails are sent from the recognizable email address "hokieforu@gmail.com," ensuring a consistent and trustworthy sender for these notifications. This approach contributes to the legitimacy of the communication process and establishes a reliable connection between the platform and its users.

Design

Domain Model:

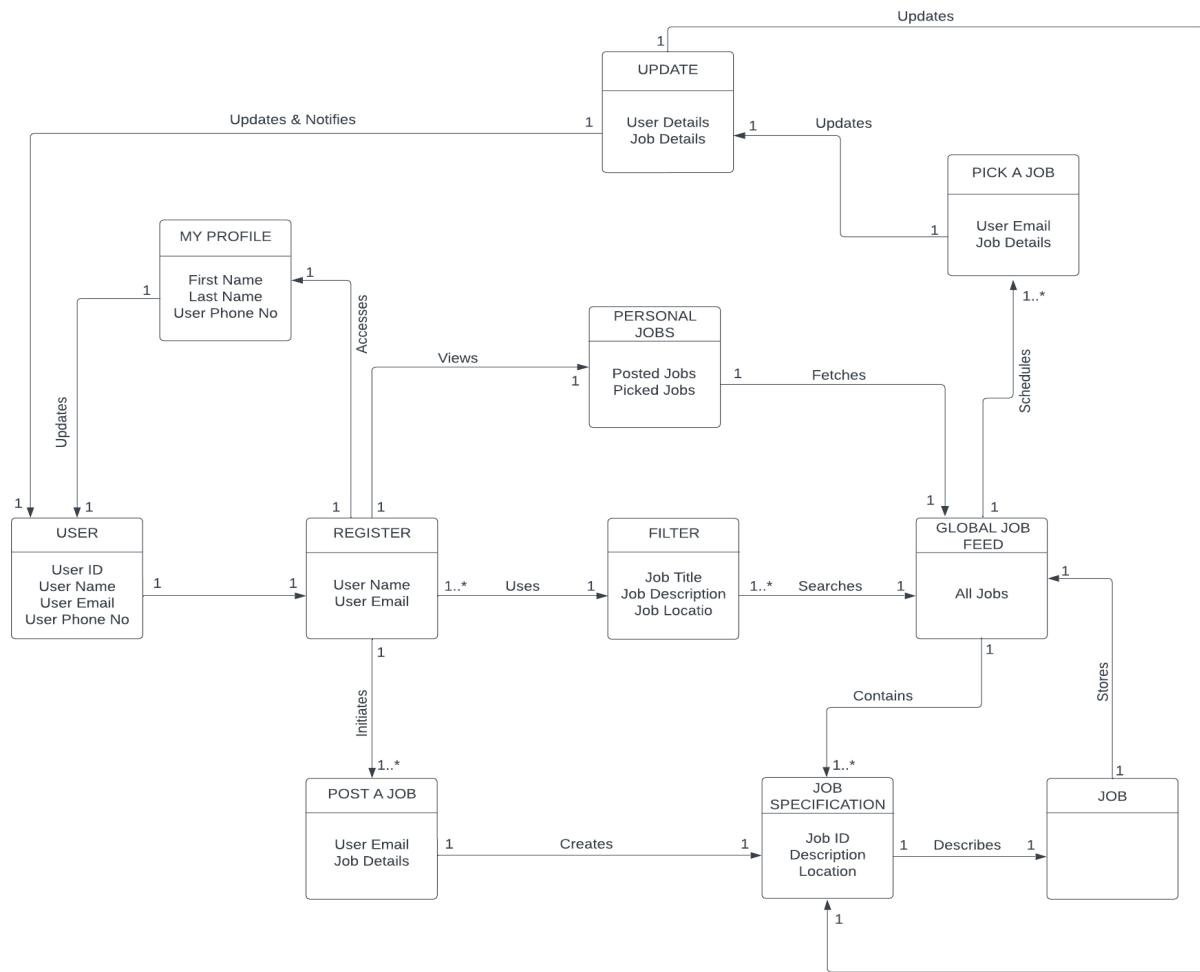


Figure 13: Domain Model of Hokie For U application

Interaction Diagrams:

Instead of showing the interaction diagrams for all the use cases, we showcased the crucial functional use cases for the report.

UseCase 1: User trying to Post a Job

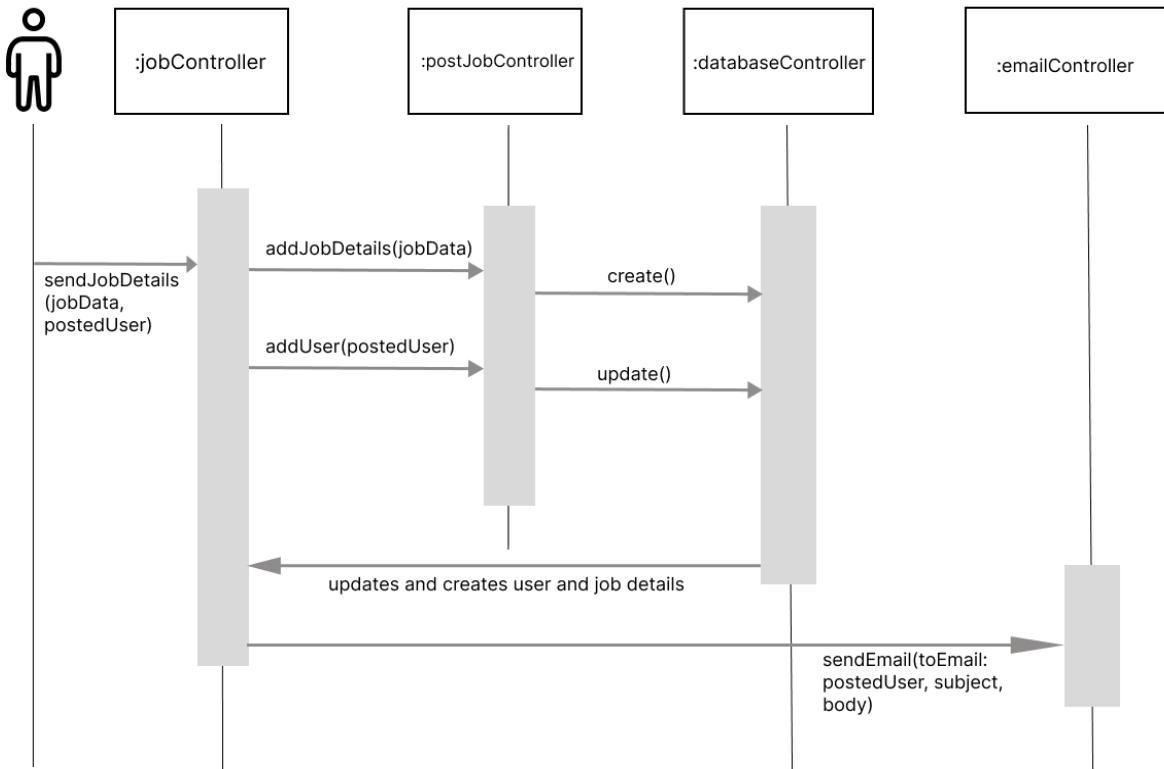


Figure 14: Interaction diagram for Usecase where User posts a job

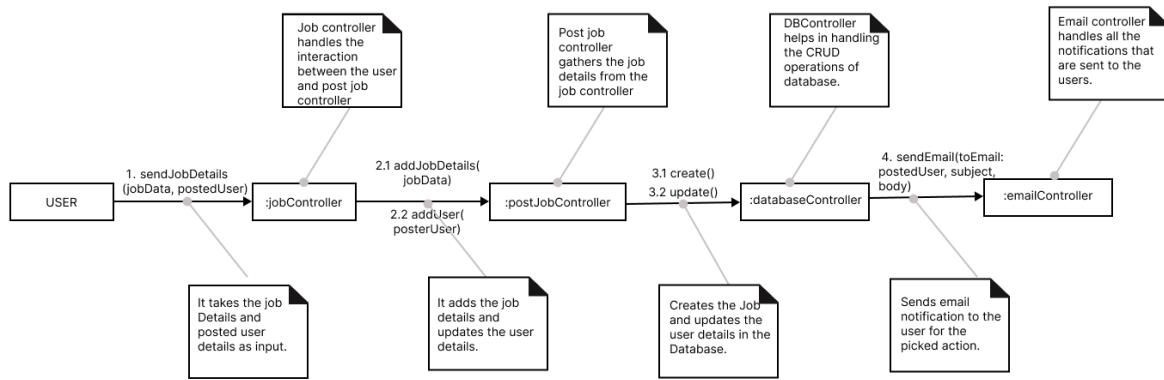


Figure 15: Collaboration diagram for Usecase where User posts a job

UseCase 2: User Trying to Pick a Job

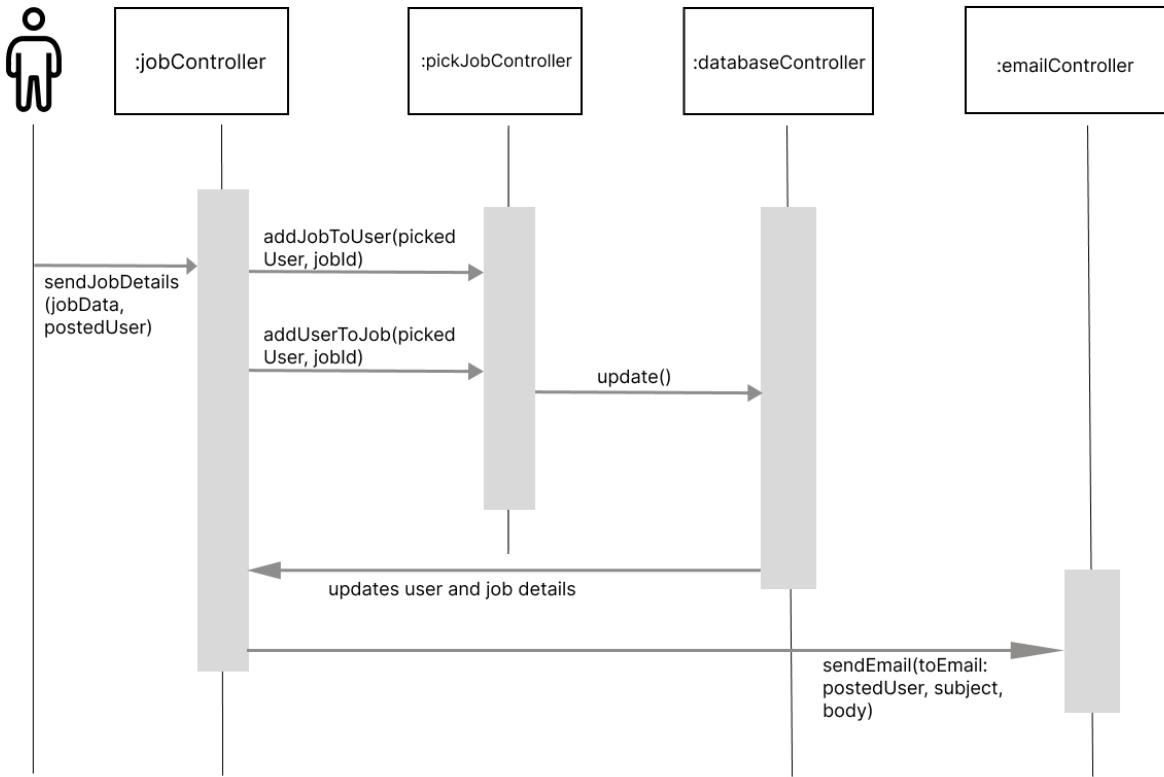


Figure 16: Interaction diagram for Usecase where User picks a job

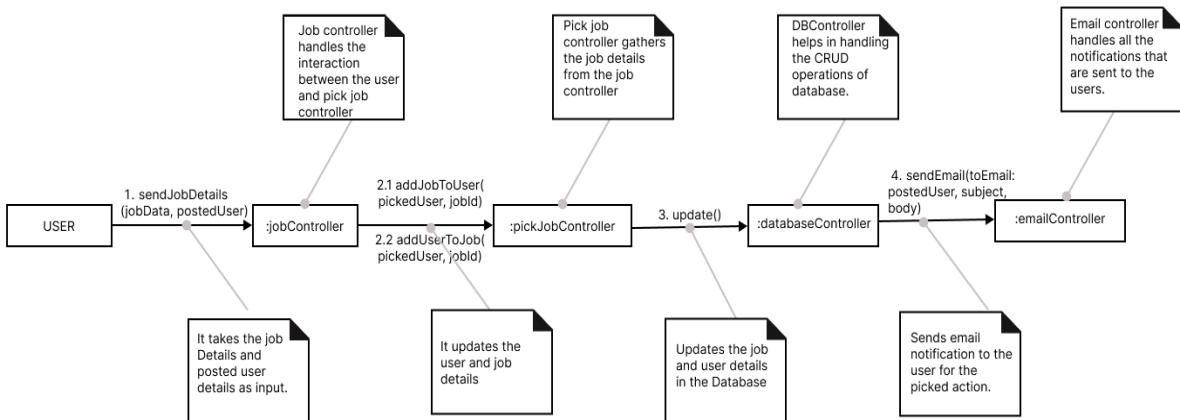


Figure 17: Collaboration diagram for Usecase where User picks a job

UseCase 3: Show all the jobs posted by the user for My Jobs Component

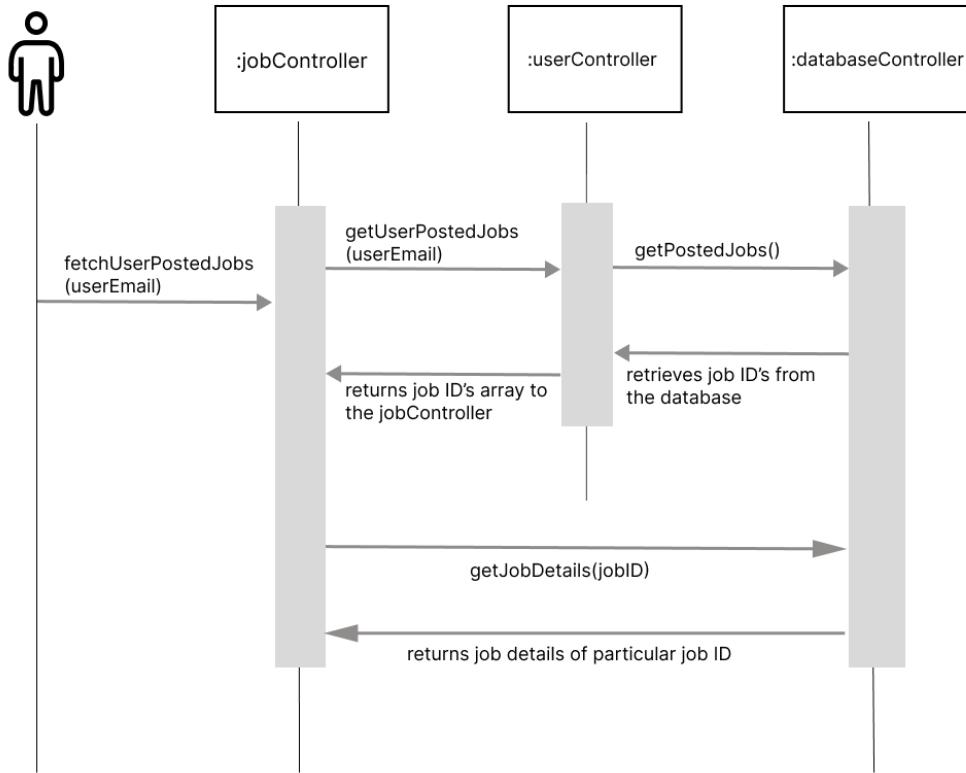


Figure 18: Interaction diagram for Usecase where User views all the posted jobs

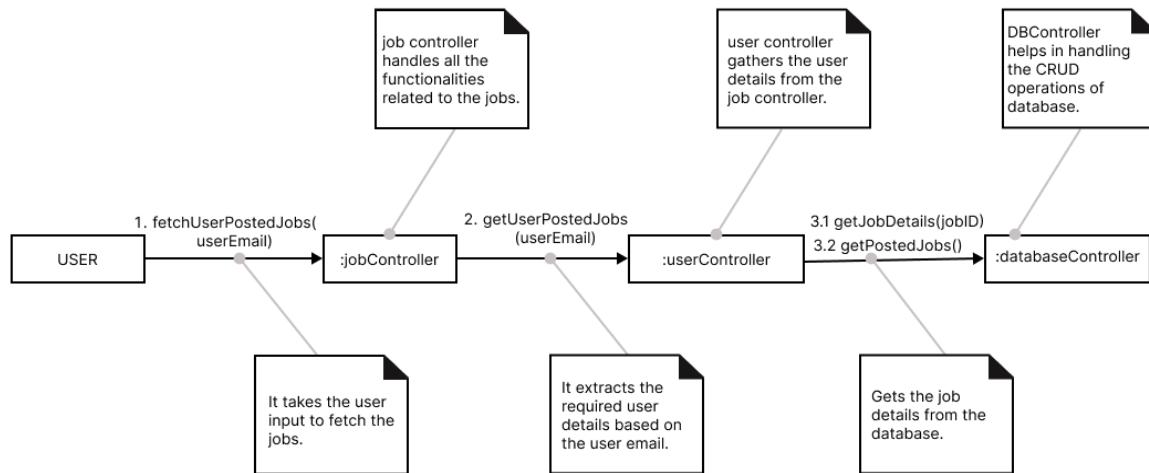


Figure 19: Collaboration diagram for Usecase where User views all the posted jobs

Design Class Diagram:

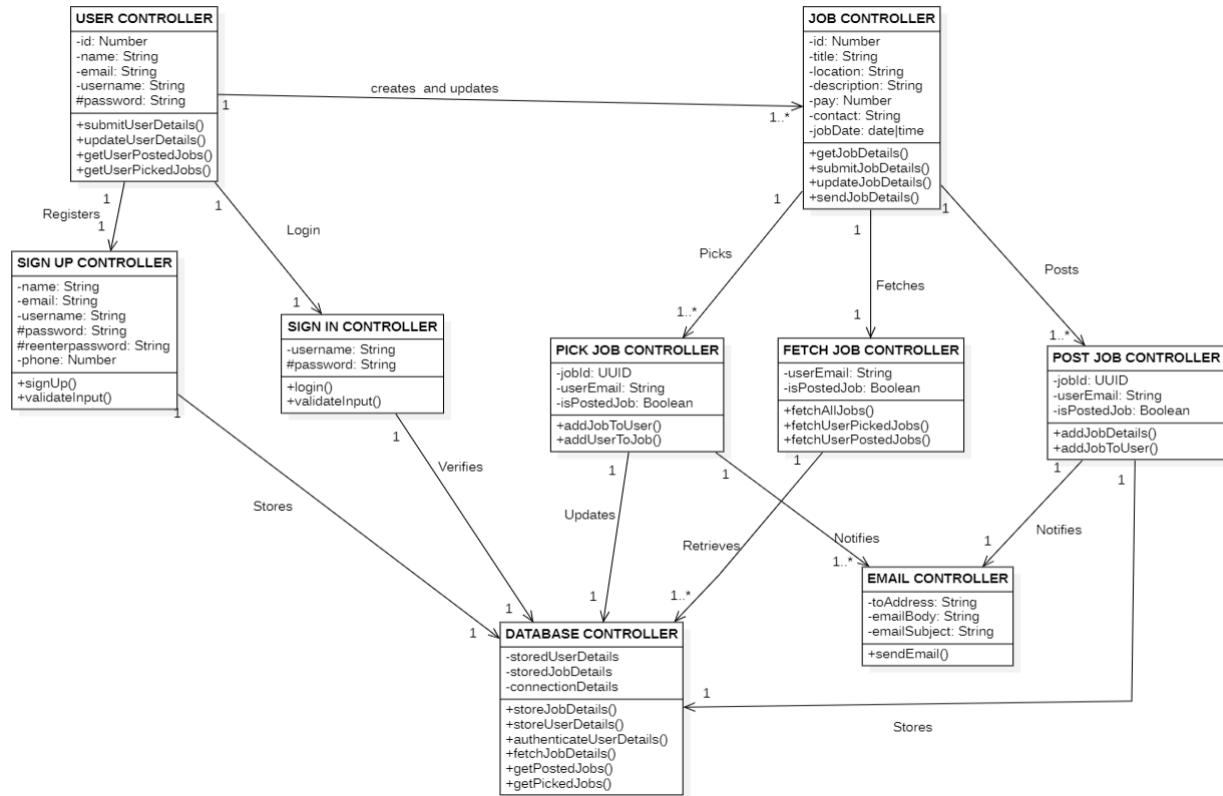


Figure 20: Design Class diagram for Hokie For U application

Collaboration Spaces:

Our team leverages three crucial tools - GitHub, JIRA, and Confluence - to amplify our productivity and streamline project management and software development processes.

Jira

JIRA, a cornerstone of our Agile methodologies, empowers us to manage our projects using Scrum boards, allowing for the efficient prioritization of tasks and incremental improvements. Custom workflows tailored to our unique processes ensure the smooth progression of work, eliminating bottlenecks. Issue tracking capabilities expedite the resolution of bugs and tasks, contributing to improved productivity. Additionally, customizable reporting and dashboards provide insights crucial for decision-making and resource allocation.

Projects / HokieForYou

Backlog

PLANNING: Timeline, Backlog, Board, Reports, Issues.

DEVELOPMENT: Code, Security (NEW), Releases.

OPERATIONS: Deployments, On-call.

Project pages: You're in a team-managed project. Learn more.

HOK Sprint 1 9 Oct – 23 Oct (9 issues)

- HOK-12 Develop a basic layout webpage for home page
- HOK-2 Create a database schema for user profiles, job postings etc
- HOK-11 Develop front end web pages for login and registration.
- HOK-1 Implement user registration and login functionality using secure authentication methods.
- HOK-13 Develop a component inside the webpage to allow user to update the profile
- HOK-14 Develop a component inside the webpage to allow users to post a job
- HOK-27 Login - Password hash
- HOK-32 Login, Register - Mobile view
- HOK-35 Existing user - Register

HOK Sprint 2 24 Oct – 6 Nov (0 issues)

HOK Sprint 3 5 Nov – 19 Nov (0 issues)

HOK Sprint 4 11 Nov – 25 Nov (0 issues)

+ Create issue

Figure 21: Sprint Backlogs in the JIRA Software

Projects / HokieForYou

HOK Sprint 1

PLANNING: Timeline, Backlog, Board, Reports, Issues.

DEVELOPMENT: Code, Security (NEW), Releases.

OPERATIONS: Deployments, On-call.

Project pages: You're in a team-managed project. Learn more.

TO DO 1

IN PROGRESS

TESTING

DONE 21

GROUP BY Subtask ▾ Import work Insights

Front End Dev - Post a Job

- HOK-37
- HOK-36

Basic UI Testing

- HOK-27
- HOK-32
- HOK-35

Everything else 4 issues

- Develop a component inside the webpage to allow user to update the profile
 - HOK-13
- Login - Password hash
 - HOK-27
- Login, Register - Mobile view
 - HOK-32
- Existing user - Register
 - HOK-35

Figure 22: Sprint Dashboards in the JIRA Platform

GitHub

The screenshot shows the GitHub repository page for 'hokie-for-u'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Security, and Insights. Below the navigation is a search bar and a button to add a file. The main content area shows a 'Switch branches/tags' dropdown set to 'dev', showing 5 branches and 0 tags. It lists branches like 'main', 'Test-Sushma', 'dev', 'dev-akshay', and 'dev-sam'. A commit log shows recent changes, including one from 'sivakumareddy07/dev-akshay' adding a Phase 2 document and fixing bugs. To the right, there's an 'About' section describing it as a 'Job posting application' with 13 commits, and sections for Releases, Packages, and Contributors.

Figure 23: Git Repository of our project

In GitHub, we embrace a collaborative approach by having each team member create their individual branches where they work on specific tasks or features. This strategy allows us to maintain a clear and organized codebase, reducing the likelihood of conflicts and errors. Once our work on these branches is complete, we employ the process of merging, incorporating our changes and improvements into the main branch. This workflow aligns with the principles of version control, enhancing code quality and accelerating our development process. Additionally, we capitalize on GitHub's versatile features, including pull requests and automated workflows via GitHub Actions, which not only streamline collaboration but also contribute to the overall efficiency of our development efforts. The project management functionalities, such as issue tracking, are instrumental in keeping our team organized and fully focused on the attainment of our project goals.

Confluence

Confluence, our collaborative documentation platform, plays a pivotal role in facilitating knowledge sharing and document management. By creating and sharing project documentation, knowledge bases, and wikis, we ensure that all team members have access to up-to-date information. Real-time collaborative features in Confluence simplify document creation, allowing our teams to work efficiently. The integration with JIRA further ensures that our project

documentation aligns seamlessly with project progress, thus keeping documentation current and aligned with the project's status.

The screenshot shows the Confluence software interface. At the top, there is a navigation bar with links for Home, Recent, Spaces, Teams, Apps, Templates, Invite people, and Create. A search bar is located at the top right. On the left, there is a sidebar for the space 'HokieForYou' with sections for All content, Blogs, Automation, Space settings, and SHORTCUTS. Under CONTENT, there are three items: Deliverable Phase 2 Ana..., Deliverable Phase 2 (De..., and Deliverable Phase 1, which is highlighted with a blue selection bar. The main content area displays a page titled 'Deliverable Phase 1'. It includes a header 'Project name - Hokie For U', a section titled 'Problem statement and brief project description', and a 'Problem Statement' section. The 'Problem Statement' section contains text about the need for accessible and reliable services in local communities. Below it is a 'Project Description' section with text about how Hokies can help provide various services like household chores, babysitting, grocery pickup, and tutoring. The page is owned by Akshay Reddy Narra and was last updated less than a minute ago.

Figure 24: Confluence software used by the team for document management.

Technology Stack:

Our application was built on the MERN stack, a powerful and versatile technology stack renowned for its ability to streamline the development of modern, robust web applications. MERN stands for MongoDB, Express.js, React, and Node.js, each contributing essential components to different aspects of our application.

MongoDB (M): MongoDB served as our application's database, providing a flexible and scalable NoSQL data storage solution. Its document-oriented structure allowed for efficient handling of diverse data types, fostering adaptability as our application evolved.

Express.js (E): Express.js, a minimalistic and flexible Node.js web application framework, formed the backbone of our backend. It simplified the development of robust APIs, handling routing, middleware, and HTTP requests efficiently. Express.js seamlessly integrated with Node.js to create a cohesive server-side environment.

React (R): React was our frontend library of choice, enabling the creation of dynamic and interactive user interfaces. Its component-based architecture facilitated modular development,

making it easier to manage and update different parts of the application. React's virtual DOM ensured optimal rendering performance, contributing to a smooth user experience.

Node.js (N): Node.js served as the runtime environment for our backend, leveraging its asynchronous, event-driven architecture. Node.js excelled in handling concurrent connections, making it well-suited for building scalable and real-time applications. It seamlessly integrated with Express.js to create a robust server-side foundation.

This MERN stack architecture provided a cohesive and end-to-end solution for our application's development. It ensured consistency throughout the tech stack, from the server-side to the user interface, simplifying the development process and fostering efficient collaboration among team members. The MERN stack's versatility, scalability, and ease of use aligned perfectly with our project goals, allowing us to deliver a seamless and engaging user experience.

Need for this System:

Numerous students express a desire to engage in community service and make a positive impact on social issues. Our platform simplifies the process of finding and participating in community service opportunities for students. By restricting access to Virginia Tech students, we enhance the precision and effectiveness of matching students with suitable opportunities. This approach increases the likelihood of student interest, ultimately saving time and effort for both organizations and students. Our application bridges the gap between Virginia Tech students who want to volunteer or work in community service roles and local organizations seeking enthusiastic volunteers.

Examples:

Websites like Handshake or even general job boards often have job listings.

Criticism:

1. Because these platforms are not designed for students or community service, the filtering choices may be inadequate for such opportunities.
2. It's worth noting that the existing system primarily focuses on job listings with longer durations and is not tailored to community service positions, which are typically short-term and oriented toward volunteer work.
3. The current systems only provide descriptions of the job and application links, they do not provide contact and scheduling features that are offered by our application.

Retrospection

What Went Well:

1. The team made significant progress in front-end development, successfully implementing crucial pages such as the homepage, login, registration, and "Post a Job" functionality.
2. The team seamlessly connected with the database, completing the design of the schema without encountering major issues.
3. Improved communication strategies enhanced collaboration, ensuring a synchronized workflow and facilitating efficient progress tracking.
4. Substantial advancements were made in back-end development, with a focus on building the core functionality of the web application.
5. Development activities were completed ahead of schedule, showcasing the team's efficiency and effective time management.
6. The team demonstrated unwavering commitment by successfully completing all planned user stories, aligning with the objectives set for each sprint.
7. Additional time dedicated to testing contributed to the delivery of a higher-quality product, ensuring a positive end-user experience.
8. The team encountered minimal obstacles, and any challenges that arose were promptly addressed, allowing for a smoother development process.
9. Sprint planning was executed meticulously, resulting in the timely completion of development tasks and paving the way for thorough testing.
10. The overall efficiency and cohesive collaboration of the team played a pivotal role in achieving the goals set for the project release.

What Went Wrong:

1. In Sprint 1, the team faced unexpected technical challenges, requiring extra time and resources for resolution. This affected the original timeline for backend development.
2. While there was an overall improvement in communication during Sprint 2, there were instances of breakdowns in communication during Sprint 1. This led to misunderstandings and necessitated some rework.
3. Despite completing development tasks early in Sprint 3, there were delays in the testing phase because the test cases were not automated. This impacted the overall testing timeline.

Learning Curve

1. The project provided an opportunity to deepen technical skills, especially in the areas of frontend and backend development.
2. The use of Agile methodology, as reflected in the sprint structure, improved the team's adaptability and responsiveness to changing project requirements.
3. The project highlighted the significance of effective communication within the team.
4. Team members learned to navigate unexpected obstacles and adjust project plans to ensure successful outcomes.
5. The ability to allocate resources efficiently and optimize work schedules contributed to meeting project deadlines showcasing time management by Sprint 3.

Areas of Improvements

Following in-depth discussions among team members, the team collectively devised the following solutions. While these solutions may not eliminate all challenges, they are anticipated to mitigate some of the issues encountered by the team.

1. Before project initiation, conduct a thorough analysis of potential technical risks. This includes identifying challenges related to backend development, database integration, and any other technical aspects. Develop strategies to mitigate these risks, ensuring a more proactive and prepared approach to unforeseen issues.
2. Establish clear and effective communication protocols from the beginning of the project. This involves defining channels for regular updates, clarifying roles and responsibilities, and implementing a structured issue-resolution process. Improved communication helps prevent misunderstandings and ensures swift problem resolution.
3. Recognizing the importance of efficient testing, initiate the automation of test cases as part of the early development stages. Automated testing not only expedites the testing process but also helps identify and address issues promptly, reducing delays in the testing phase.
4. Acknowledge the inevitability of unexpected technical challenges by incorporating buffers in the development timeline. This ensures that unforeseen issues can be addressed without causing significant disruptions to the project schedule, promoting a more realistic and adaptable timeline.
5. Improve the logging of work to maintain better transparency and track progress effectively.

Future Enhancements

1. Strengthen the communication features of the application. Implement real-time chat functionality or alerts to improve communication between service seekers and providers. Clear and efficient communication is vital for the success of a service-oriented platform.
2. Implement data analytics tools to gain valuable insights into user behavior, preferences, and trends.
3. Integrate geolocation services to enable users to discover and post jobs based on their proximity. This feature can enhance the local community aspect of the platform.
4. Implement an automated job matching system that suggests relevant jobs to users based on their preferences, skills, and previous interactions on the platform.
5. Implement a new functionality focusing on subscribe options for job listing updates.

GitHub:

The link below is for our Git Repository.

<https://github.com/sivakumarreddy07/Hokie-For-U>