

COOL AS CODE

GROUP 13



**A. INTRODUCTION**

Our Web-app initiative intends to improve the process of obtaining financial aid by compiling complete information on scholarships, grants, and bursaries, as well as their respective eligibility requirements. In today’s increasingly competitive academic environment, accessing financial aid can be daunting and time-consuming. Our approach aims to expedite this process, allowing individuals to pursue their educational and career goals without financial restraints.

**Objectives.**

Create a consolidated platform for people requesting financial assistance.

Provide extensive information about numerous scholarship, grant, and bursary options.

Clearly state the eligibility requirements and application procedures for each opportunity.

Improve access and openness in the financial aid landscape.

**Significant**.

This project is significant because it addresses a common issue faced by students seeking to further their education. By offering a user-friendly interface and comprehensive database, our Web-app aims to democratize access to financial aid, leveling the playing field and empowering individuals from diverse backgrounds to pursue their dreams. Furthermore, by facilitating the connection between aspiring individuals and available financial resources, our project helps to foster talent, innovation, and socioeconomic mobility in our community.

**B. DEVELOPMENT METHODOLOGY**

Our project has chosen the Agile software development methodology iterative and flexible nature, which is perfectly aligned with our project objectives and team dynamics. Here’s how Agile helps us achieve our objectives.

Agile enables us to adapt to changing requirements and priorities with ease. Given the dynamic nature of the financial aid landscape and our users’ changing needs, the ability to pivot quickly is critical. Agile allows us to incorporate feedback and make changes throughout the development process, ensuring that our web application remains relevant and effective.

Agile encourages close collaboration between team members, stakeholders, and end users. This collaborative approach promotes a common understanding of the project’s objectives and encourages open communication. Agile allows us to stay on track with user needs and project goals by encouraging regular interactions and feedback loops, resulting in a product that better meets our target audience’s expectations.

Agile breaks the project down into digestible chunks, which enables us to gradually deliver value to our users. We may prioritize additions according to their relevance and viability with this iterative strategy, which starts with a minimal viable product (MVP) and adds functionality progressively in response to user feedback and evolving requirements. Agile improves our ability to confirm assumptions, reduce risks, and provide measurable results by producing useful functionality in brief iterations.   
  
Agile offers the structure and guiding principles needed to accomplish our project's objectives, which include creating a high-caliber web application that is focused on the user and meets the changing and complicated needs of our target market. It is a perfect fit for our team and project because of its emphasis on adaptability, flexibility, cooperation, and iterative development.

**C. APPLICATION ARCHITECTURE AND DESIGN**

The web application has a three-tier architecture, which includes the Presentation layer, Application layer, and Data layer. Each layer has a specific purpose and interacts with the others to provide the application’s functionality.

1.THE PRESENTATION LAYER : Manages user interactions and displays the user interface.

It consists of Web pages, UI templates, and client-side scripts (HTML, CSS, JavaScript).

This layer communicates directly with the user’s web browser.

The user interacts with the application via a web browser, sending requests to the server for data and functionality.

2.APPLICATION LAYER : The application layer consists of the business logic and application functionality.

It handles user requests, executes operations, and coordinates interactions between the Presentation and Data Layers.

Controllers, service classes, and application logic are all essential components.

This layer manages authentication, authorization, validation, and business rules.

It communicates with the data layer, retrieving and manipulating data as needed.

3. DATA LAYER : The data layer manages data storage and retrieval.

It consists of a database management system (DBMS) and related components.

Databases, tables, indexes, and data access objects are all essential components.

This layer stores and retrieves data required for the application’s operations.

It communicates with the Application Layer to perform CRUD (Create, Read, Update, and Delete) operations on data.

**D. APPLICATION SCENARIO AND USER SCENARIOS**

Our Web-app includes a variety of features and capabilities that make it easier to find financial assistance. Here’s a summary of some key features and the APIs consumed to support them:

1. Users can search for scholarships, grants, and bursaries using a variety of parameters, including keywords, eligibility conditions, and application deadlines.

To retrieve pertinent information, the program uses APIs provided by scholarship databases and financial assistance agencies.

2. Each financial aid opportunity is presented with complete information such as eligibility requirements, application procedures, deadlines, and contact information.

3.: The application makes personalized suggestions based on the user’s profile, academic background, and interests.

APIs consumed: Machine learning algorithms analyze user data and recommend the most relevant financial aid opportunities.

4. Users are notified about forthcoming deadlines and required activities. APIs used include integration with email and notification systems for sending automated updates to users.

USER SCENARIOS :

1.Scenario 1: SEARCHING FOR SCHOLARSHIPS

Sibusisiwe, a senior in high school, wants to find scholarships to cover her upcoming college expenses.

User Interactions: Sibusisiwe types keywords about her interests and academic achievements into the search bar.

Application behavior: The application retrieves relevant scholarship opportunities from its database and external APIs and displays them to Sibusisiwe along with detailed information and eligibility requirements.

2.Scenario 2: **RECEIVING PERSONALIZED RECOMMENDATIONS**

Zenande isn’t sure where to begin her search for financial aid.

User Interaction: Zenande completes a brief questionnaire about her academic background, interests, and financial situation.

Application behavior: The application analyzes Zenande’s responses and makes personalized recommendations based on her profile.

By providing these features and capabilities, our Web application not only simplifies the process of obtaining financial assistance, but it also increases user engagement and, as a result, empowers individuals to achieve their educational and professional goals.

**E. TESTING AND DEPLOYMENT**

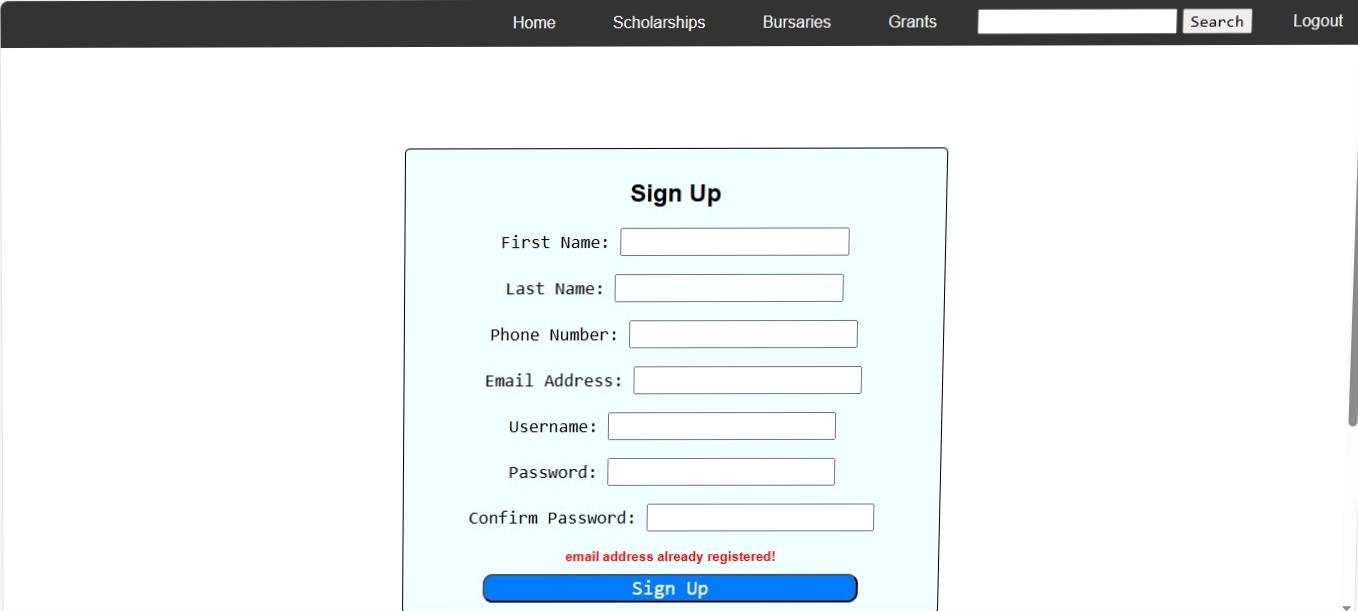
TESTING STRATEGIES

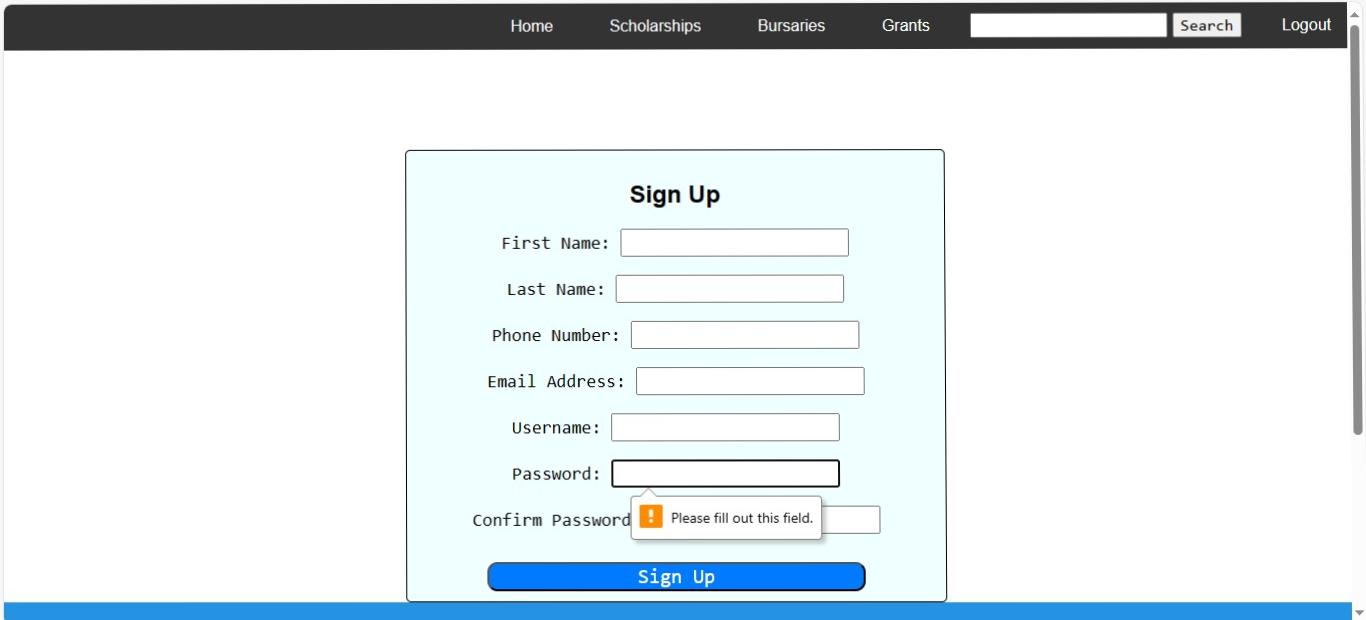
1.SECURITY TESTING :

This is a security testing to test that the system does not allow a user to login without inserting correct username and password. In order for the user to login, they have to insert a correct username and password. This ensures that the only who have signed up and recognized the system can be able have access to the website.

2.2VALIDATION TESTING :

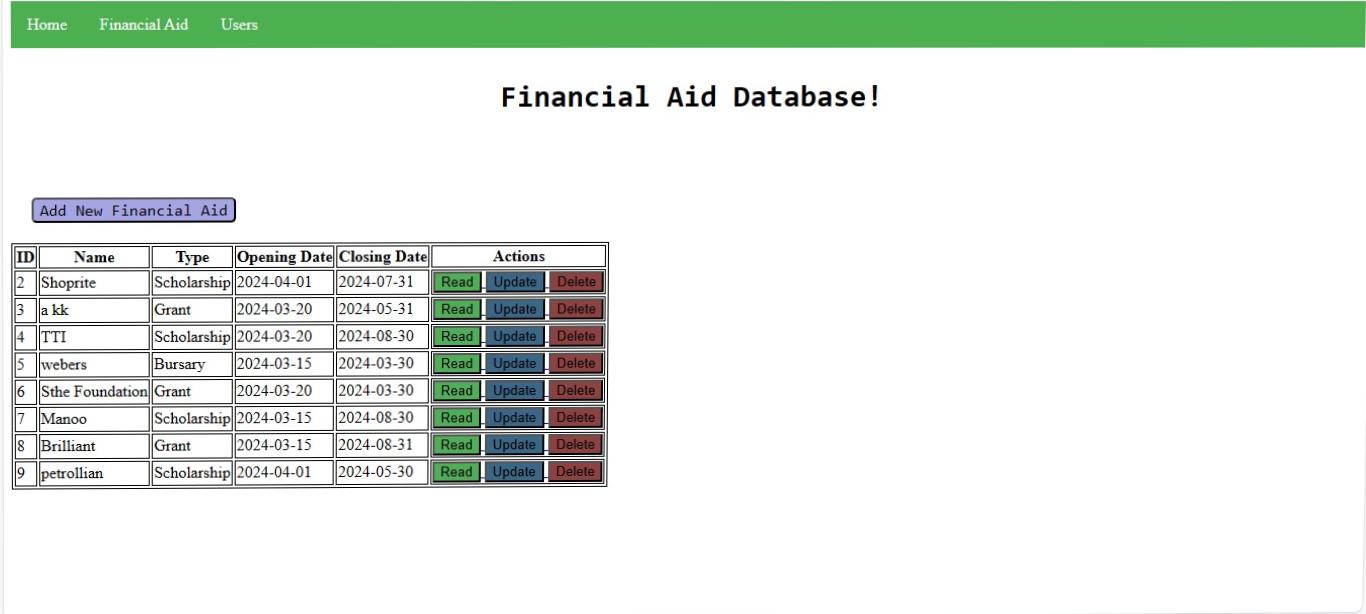
This is the type of testing done to check if the user has typed information in the fields given. Validation testing ensures that the required fields are filled out and that the data entered meets specified criteria. We have used Validation testing on the sign up page to ensure that the user has filled in all the required fields after clicking on the sign up button. We also did the duplicate email validation testing to verify that the system correctly detects and prevents the creation of multiple accounts with the same email address.

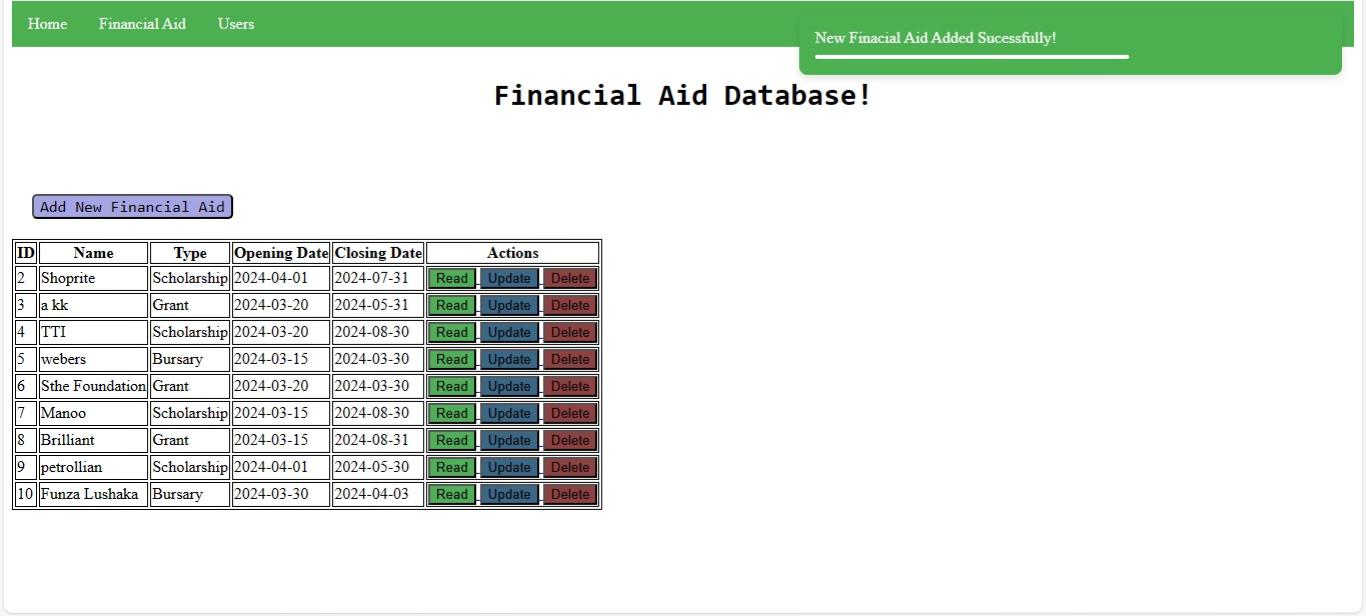




3. SYSTEM TESTING :

All components of the application are tested together to ensure proper functionality.

Under system testing we tested the database function in terms of adding new financial aids, deleting existing financial aids and updating them. We added a new financial aid and then a pop-up message got displayed to state that the financial aid has been added and it got added to the table which displays all the existing financial aids. When the existing financial aid is deleted, it is removed from the database and also from the table which displays all the existing financial aids.



DEPLOYMENT PROCESS

1. FINALIZING THE APPLICATION FOR RELEASE :

All testing phases have been completed, and any issues found have been addressed.

The application is thoroughly reviewed to ensure that it meets quality, reliability, and security requirements.

2. DEPLOYMENT ENVIRONMENTS :

Multiple deployment environments are used, such as development, staging, and production.

The development environment is used for continuous development and testing.

Staging environment: simulates the production environment for final testing prior to deployment.

Production environment: A live environment that is accessible to end users.

3. CONTINUOUS INTEGRATION AND DEPLOYMENT (CI/CD) PIPELINES:

CI/CD pipelines automate application development, testing, and deployment.

Continuous integration ensures that code changes are consistently merged and tested.

Continuous deployment automates the deployment process, allowing for seamless updates to the application.

4. DEPLOYMENT PROCESS :

Code changes are committed to a version control system.

CI/CD pipelines automate builds and tests for each commit.

Once the tests are completed successfully, the application is deployed to the staging environment for final testing.

After stakeholder approval and successful UAT, the application is deployed to production.

**F. CONCLUSION**

In conclusion, this documentation has provided a comprehensive overview of our Web application project aimed at simplifying the process of finding financial assistance. We discussed the application architecture’s features, testing strategies, and deployment process, emphasizing their importance in meeting the needs of individuals seeking financial aid for education and career advancement.

Throughout the development process, we prioritized user-centric design, dependability, and security, using Agile methodologies and continuous integration and deployment pipelines to ensure high-quality results. Our application, which uses a three-tier architecture and personalized recommendations, provides a user-friendly platform that enables individuals to access relevant financial aid opportunities and pursue their goals with confidence.

RE**FERENCES**

1. [S Al-Saqqa](https://scholar.google.com/citations?user=lBI3MlgAAAAJ&hl=en&oi=sra), [S Sawalha](https://scholar.google.com/citations?user=LTSzal4AAAAJ&hl=en&oi=sra) (2020), AMM Hamed, [H Abushama](https://scholar.google.com/citations?user=MNfIqA8AAAAJ&hl=en&oi=sra) (2013), [P Abrahamsson](https://scholar.google.com/citations?user=A-CX3y4AAAAJ&hl=en&oi=sra),  [MT Siponen](https://scholar.google.com/citations?user=lG0VhooAAAAJ&hl=en&oi=sra)  (2010).

2. SQL Alchemy

3. bcrypt

4. Jinja2:

5. SQLite