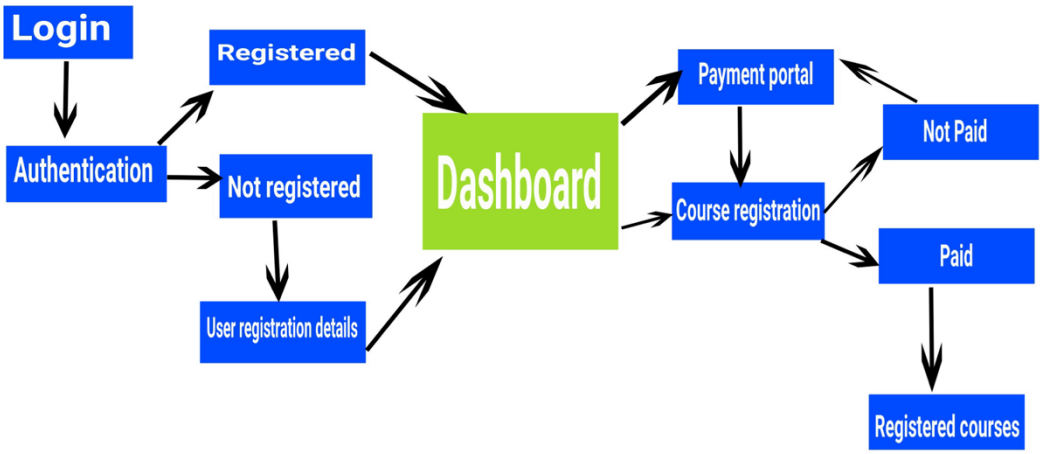
DESIGN/DEVELOPMENT OF COURSE REGISTRATION WEB APPLICATION

GROUP SEVEN (7)

S/N	NAME	MATRIC NUMBER
1.	DAVID TIMOTHY ADINOYI	2021/2/81357CS
2.	JOY GRACE UGBENYO	2019/1/76593CS
3.	JAMES PIOPANA NOTHINGBAD	2021/2/80252CS
4.	TOCHUKWU GLORY OBI	2019/1/79465CS
5.	JIMOH ABDULHAMEED	2019/1/77878CS
6.	ADAMA DAMIAN OJONIMI	2021/2/79607CS

MARCH 27TH, 2024

INFORMATION ARCHITECTURE



OVERVIEW OF THE COURSE REGISTRATION WEB APPLICATION

The application streamlines course registration, offering an intuitive interface for students to manage course registration considering the levels in the school and make registration payment online. It ensures that registration payment is made before course registration.

SYSTEM ARCHITECTURE

Utilize HTML, CSS, JavaScript and MongoDb for interactive and responsive user interface.

USER ROLES AND PERMISSIONS

Student

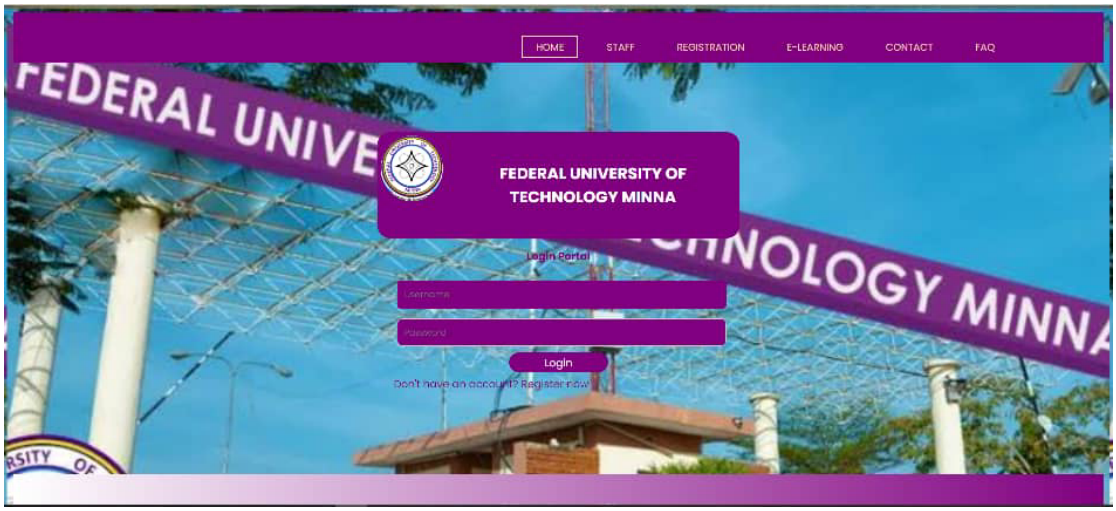
- User registration and Login
- Registration (school fees) payment
- Course Registration
- Print school fees receipts
- Automatic generation of schedules based on course availability and student preference.

Admin

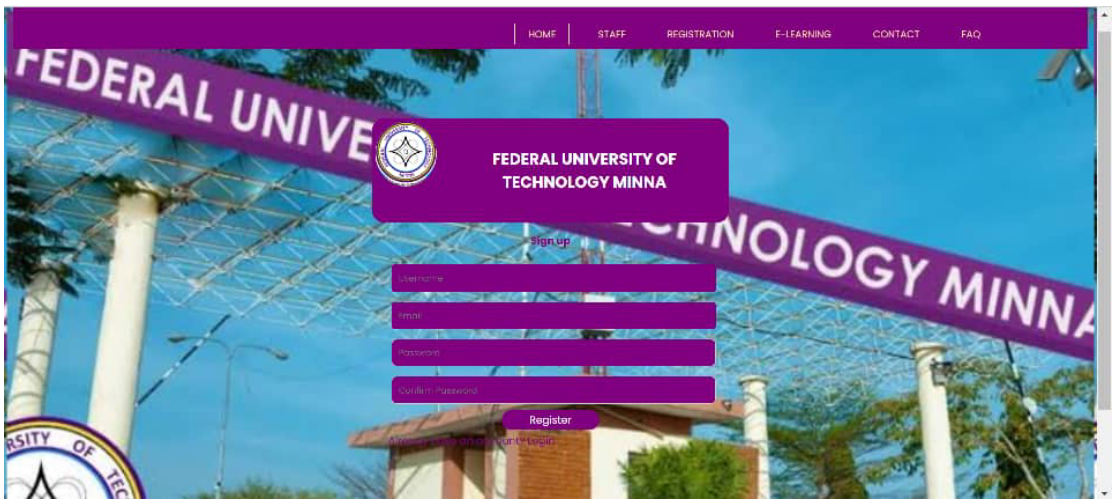
- Manage database

FEATURES AND FUNCTIONALITIES

User Login / Registration



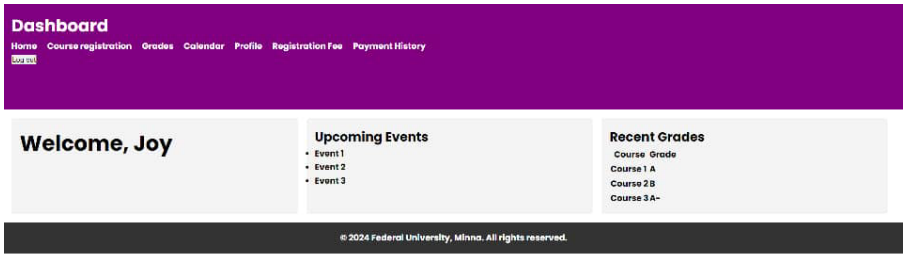
Login page



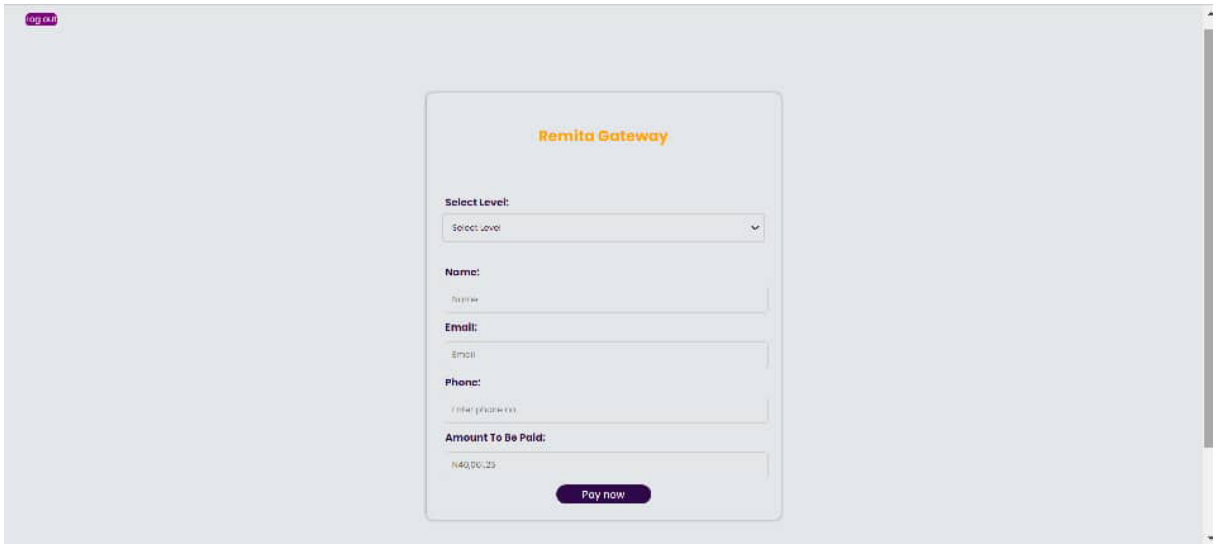
Sign up (user registration page)

It allows students to register and also Login with user name and strong passwords to ensure authorized access to their student portal. During registration the system detects weak password. Student must register using a strong password which contains numbers, alphabets, special characters and should be at least 8 characters long.

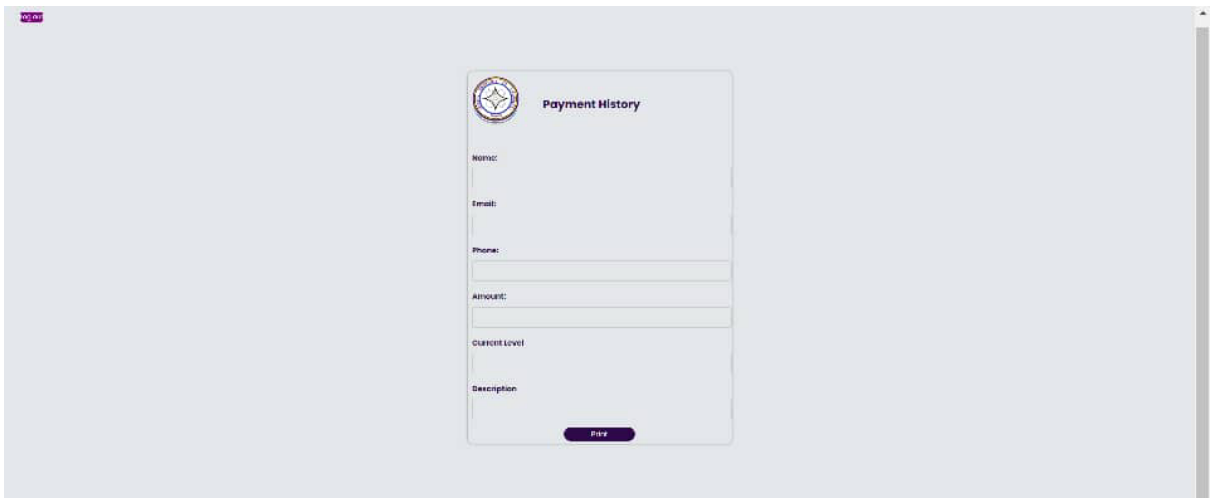
Payment Portal and Processing



Dash board



Payment portal



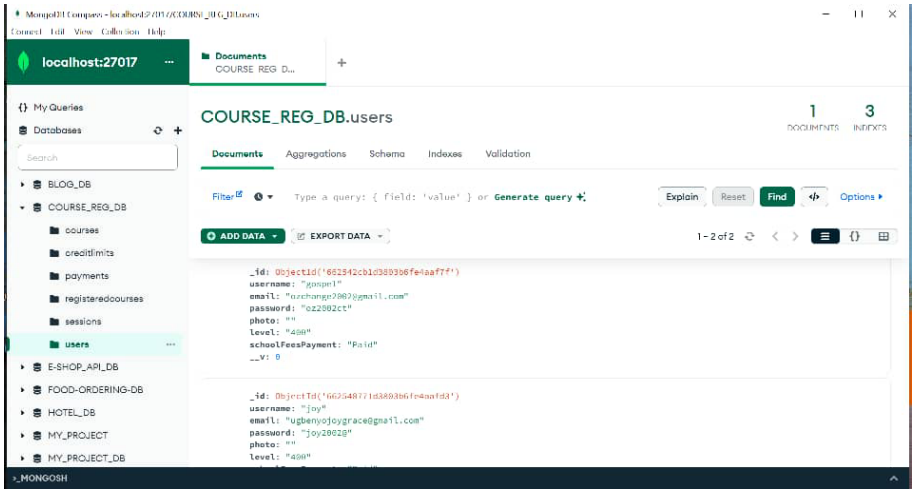
Payment registration

The payment portal (Registration page) allows students to pay school fees online. The students can also print out the receipt after payment has been made

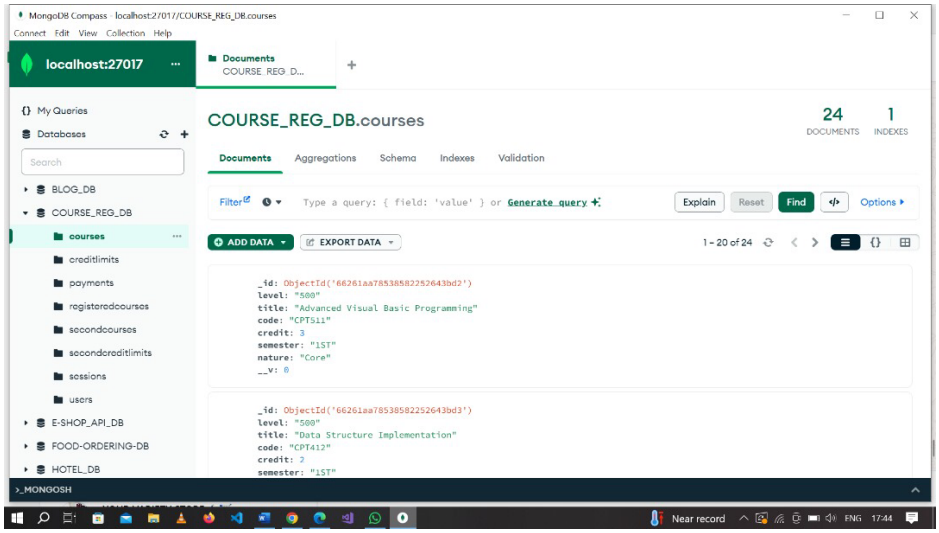
Course Registration

The course registration page contains the list of courses for each level from 100-500. The courses are automatic generation of schedules based on course availability and student preference. The course registration page allows student to register courses online. Students can select and remove course from the page. The website also ensures that students can't register for a course twice. Students are only allowed to register a maximum of 10 credit unit load per semester

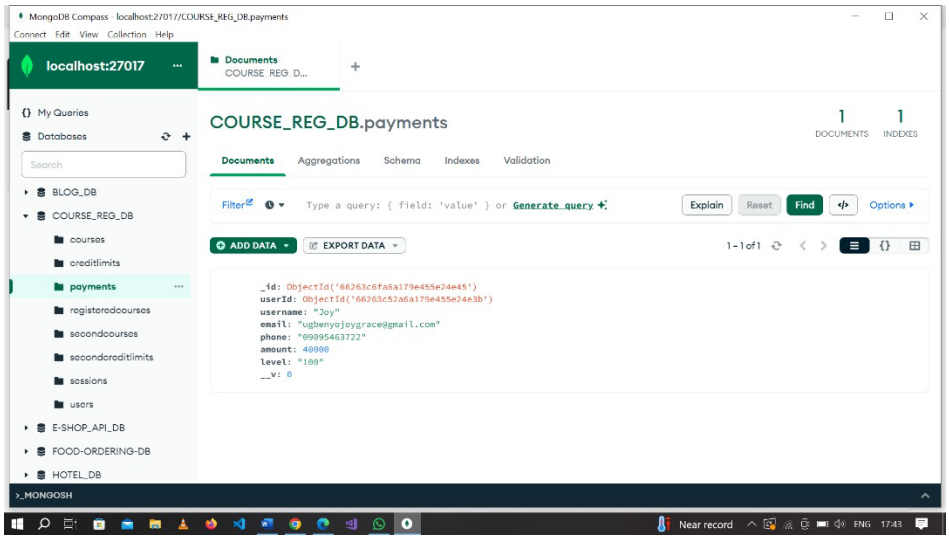
Other functionalities



Database showing list of registered users, their email and password



Database showing list of registered courses



Database showing student payment receipts

USAGE

- Open the login.html file in a web browser.
- For registered users input Login details which is your user name and password. For users that has not been registered click on the link to register and input your username and password. Note: your password must contain letters, numbers, special characters and should be At least 8 characters long.
- The Login page directs users to their dashboard which contains registration page, course registration page, etc.
- To register for course, click on Course registration. Note that you cannot register for a course without making registration fee payment.
- Click (this prompt you to make registration fee payment and redirect you to a payment gateway to make payment). Select your level and fill in your details for name, email and phone number .
- Click on payment confirmation after a successful registration payment
- Click on Level to select the particular course for your course registration.
- After selecting your courses click on print.
- Click on logout

SECURITY

- User authentication measure
- Authorization control
- Accessibility

THREAT MODELING

We identified the security vulnerabilities in our web application design, this is to ensure that we work to fix them when developing and writing the code for the backend.

DIAGRAM

This will cover the scope and the assets of our website

Scope:

• Student profiles and accounts.
• Course lists and descriptions.
• Online registration.
• Course Selection.
• Payment processing.
• Authentication and authorization.
• Access control and permission.

Assets:

• Student personal and academic data (names, IDs, course enrollment).
• Website infrastructure (servers, databases, network).
• Course list information.
• Student accounts and login credentials.
• Payment processing information.

IDENTIFY THREATS

This will cover the threat actors and the possible threats.

Threat Actors:

• Unauthorized users (hackers, script kiddies)
• Malicious insiders (disgruntled employees, students)
• Automated bots/scanners (vulnerability scanners, brute-force attacks)
• Denial of Service (DoS) or Distributed Denial of Service (DDoS) attackers

Possible threats we identified:

• Students could use weak and guessable passwords, this is prone to brute-force attacks.

• Lack of input validation, the system won't detect if the password or username inputted is the right one.
• Improper access control, the system won't be able to detect legitimate users and so illegitimate users can access a students portal.
• Non-repudiation attacks could be carried out, where a person can claim to have registered when they haven't.

MITIGATION

The security mechanisms we put in place:

• The system detects weak passwords and guessable password during user's registration.

• A standard input validation procedure was put in place to ensure the right username or password is being used to login.

• Students username names and passwords are stored in a secure database, to ensure that who ever is logging in has proper access.

• After students have paid and registered their courses, their information is saved to the secure database, this way registration won't be carried out again to avoid repeated transactions.

• To avoid non-repudiation attacks, a database was created to store the logs of sessions carried out on the website.

VALIDATION

Have we acted on each of the previous steps? Yes we have.

Policies to be implemented

1. Data Privacy and Protection:
    - Adhere to data protection regulations by ensuring that user data is collected, stored, and processed securely.
    - Use encryption for sensitive data such as passwords and personal information(password s and sensitive information should not be visible to unauthorized users)
    - Users must use a strong password consisting of. Special characters, alphanumeric letters and a least 8 characters long

2. User Authentication and Authorization (e.g., password hashing, multi-factor authentication):
    - Ensure that only authorized users can access the website and their portal.
    - Define user roles and set up appropriate access controls based on the roles

3. Course Enrollment Rules:
    - Define and enforce rules for course enrollments. Eg, prerequisites, maximum enrollment limits: a student should not register a maximum of 10 credit unit load per semester) to ensure fairness and efficiency in registration

- Student should not be allowed to register for a course twice.

4. UX/UI best practices: Ensure responsiveness for different devices (e.g., desktop, mobile)
5. Performance and Scalability:Design the application to handle a large number of concurrent users during peak times (e.g., registration periods).
6. Testing and Quality Assurance:Conduct comprehensive testing (e.g., unit tests, integration tests, usability tests) to identify and fix bugs before deployment.
7. User Policies and Terms of Service:Define clear user policies, terms of service, and code of conduct to govern the use of the platform and set expectations for users.

Troubleshooting Mechanisms

Here is a comprehensive mechanism for troubleshooting the course registration web application:

1. Identify the problem: First, determine what the issue is and where it is occurring. Is it affecting all users or just some? Is it happening on a specific page or throughout the entire application?

2. Gather information: Next, gather as much information as possible about the issue. This might include error messages, screenshots, or user feedback.

3. Check the logs: Review the server logs and application logs for any errors or warnings that might be related to the issue.

4. Try to isolate the problem by testing different scenarios. For example, try registering for a course using different browsers or devices to see if the issue is specific to one type of user or device.

5. Check the database: Make sure that the database is functioning properly and that there are no issues with the data. Check for errors or inconsistencies in the database.

CONCLUSION

This design documentation is a comprehensive resource for understanding and utilizing the Course Registration Web Application. The application enhances the efficiency and accessibility, ensuring that users adhere to payment registration before Courses Registration.

The modular design, incorporating frontend technologies like HTML, CSS, JavaScript and Mongodb ensures scalability, responsiveness and a seamless user experience.

In essence, the course Registration Web Application stands as an effective tool for modernizing and optimizing course management processes, fostering a more streamlined, secure and user-friendly educational environment.