BABCOCK UNIVERSITY, ILISHAN-REMO,

OGUN STATE, NIGERIA

TECHNICAL REPORT OF STUDENT INDUSTRIAL WORK EXPERIENCE SCHEME (SIWES)

CARRIED OUT AT

VAS2NET TECHNOLOGIES LIMITED

PRESENTED TO THE

DEPARTMENT OF SOFTWARE ENGINEERING

BY

ANYANWU, CHUKWUEMEKA CLINTON

22/0319

SOFTWARE ENGINEERING

FROM JANUARY 15TH 2024 TO JUNE 25TH 2024

CERTIFICATION

This is to certify that I, Anyanwu Chukwuemeka Clinton, hereby declare that the information in this report was written by me. It is true that I did the student industrial work experience scheme (SIWES) otherwise known as Industrial Training (IT) between January 15th and June 25th 2024.

Student's Matric No: 22/0319

Student's Signature:…….

SIWES Coordinator/Supervisor:…….

Oral IT-Defense Coordinator:……..

SIWES Class Code: SE/D159

Student Name: ANYANWU, CHUKWUEMEKA CLINTON

Matric: 22/0319

Email: chrisemeka096@gmail.com

Level: 300 Level

Course of study: Software Engineering

Mobile number: 08127984934

Company name: Vas2Net Technologies Limited

Postal Address & traceable location of company: 5, Stella Sholanke Street, Ajao Estate, Lagos, Nigeria.

Geographical state: Lagos State

Company Email: hr@vas2nets.com

Company Phone Number: +234 813 419 6851

Company website: https://www.vas2nets.com/

Staff strength (Number of staff): 50-100

Name of Industry-Based Unit Head (immediate supervisor): Mr Rasaq Olamilekan Sulaimon

Mobile number (Supervisor): +2347030373137

ACKNOWLEDGEMENT

I give my everlasting thanks to God for his strength and guidance these past 6 months. This past 6 months has been an tough but eye opening one where I got to have an experience of the technological world outside the four corners of the classroom and it's all through God's grace in my life.

I would also like to thank my parents who helped me get the chance to study and Babcock University and also an interview at Vas2Net Technology Limited.

Also this 6 months would have been a fruitless one without the opportunity granted me by Vas2Net Technology Limited, from granting audience for an interview; to accepting me and providing a convenient environment for me to work, learn and grow, I am truly grateful. Also to my Indutry Based Supervisor Mr. Rasaq Olamilekan Sulaimon for his technical support and constant supervision which contributed immensely to my personal development. I also thank him for his guidance which was a remarkable force that enabled me to succ**es**sfully complete the internship program.

To the rest of the staff in Vas2Net Technologies Limited, their support and guidance which helped me to overcome the challenges I faced during the Six months of my SIWES.

Finally, I would love to appreciate Babcock University for the opportunity and also the staff. Especially the lecturers in the Department of Software Engineering, Babcock University, Dr. Aaron Izaang who is our Student Industrial Work Experience Scheme (SIWES) coordinator, and Mr Idepefo Felix the university-based SIWES Supervisor who took out time to inspect me physically on-site and also give me words of advice despite his busy schedule.

TABLE OF CONTENT

CHAPTER FIVE: SUMMARY AND CONCLUSION

CHAPTER ONE: BRIEF HISTORY OF SIWES

1.1 **BRIEF HISTORY OF SIWES**

The Student Industrial Work Experience Scheme (SIWES) was conceived in the early 1970s as a revolutionary initiative to bridge the gap between academic theory and practical industry experience for students in Nigerian tertiary institutions. Before its establishment, there was a growing concern among employers and industry stakeholders that graduates lacked the necessary hands-on skills and real-world exposure to excel in the workforce.

In recognition of this deficiency, the Industrial Training Fund (ITF), established in 1971, pioneered SIWES in 1973. The primary objective was to provide students with an opportunity to gain practical experience in their respective fields, complementing their classroom-based learning with valuable on-the-job training. SIWES aimed to equip students with the practical knowledge, technical expertise, and problem-solving abilities required to thrive in their future careers by immersing them in industrial settings.

The implementation of SIWES marked a significant shift in the Nigerian education system, acknowledging the importance of aligning academic curricula with the dynamic needs of the labor market. Previously, there was a disconnect between the theoretical concepts taught in educational institutions and the practical realities of the workplace. SIWES bridged this gap by creating a structured framework for students to apply their academic knowledge in real-world scenarios, working alongside industry professionals and utilizing specialized equipment and machinery often unavailable in traditional academic settings.

Over the years, SIWES has evolved from its initial focus on technical skills development to a more comprehensive experiential learning program. In addition to fostering practical competencies, the scheme now emphasizes cultivating leadership qualities and career readiness. SIWES aims to nurture students' leadership potential through mentorship opportunities with industry leaders, enabling them to develop the skills necessary to become future leaders in their respective fields.

Today, participation in SIWES is mandatory for obtaining diplomas and degrees in various disciplines at most Nigerian higher education institutions. The program is managed by the ITF in collaboration with educational bodies such as the National Universities Commission (NUC), the National Board for Technical Education (NBTE), and the National Commission for Colleges of Education (NCCE). Funded by the Federal Government of Nigeria, SIWES serves undergraduate students across a wide range of fields, including agriculture, engineering, technology, environmental science, education, medical science, and pure and applied sciences. With the duration differing for different Higher Institutions, Universities have a duration of 6 months, Polytechnics and College of Education being a month duration. (ITF, 2015)

By fostering a synergy between academic institutions and industry partners, SIWES has become a catalyst for nurturing well-rounded professionals capable of meeting the ever-evolving demands of the job market.

1.2 AIMS AND OBJECTIVES OF SIWES

Aims:

1. To produce a highly skilled and industry-ready workforce by bridging the gap between theoretical knowledge and practical application.
2. To foster collaboration between educational institutions and industry partners, aligning academic curricula with the dynamic needs of the labor market.
3. To promote human capital development and enhance the employability of Nigerian graduates through hands-on industrial experience.
4. To contribute to the overall economic growth and development of Nigeria by providing organizations with a pool of well-trained and competent professionals.

Objectives:

1. To provide students with real-world exposure to industrial processes, equipment, and working environments relevant to their fields of study.
2. To develop practical skills, problem-solving abilities, and technical competencies in students through structured on-the-job training.
3. To cultivate leadership qualities, teamwork, and professional ethics among students, preparing them for future managerial and leadership roles.
4. To facilitate knowledge transfer and the exchange of ideas between academic institutions and industry professionals, fostering continuous improvement and innovation.

5. To promote the adoption of global best practices in skills training and human capital development, ensuring that Nigerian graduates are competitive in the international job market.

6. To foster a culture of commitment, loyalty, integrity, professionalism, creativity, efficiency, and effectiveness among students, preparing them to be valuable assets to their future employers.

7. To establish and maintain strong partnerships with industries, organizations, and stakeholders to ensure the relevance and effectiveness of the SIWES program.

8. To continuously evaluate and improve the SIWES program based on feedback from industry partners, educational institutions, and students, ensuring its alignment with evolving industry needs and best practices.

CHAPTER TWO: COMPANY PROFILE

2.1 BRIEF HISTORY OF VAS2NET TECHNOLOGIES LIMITED

Vas2Net Technology Limited is a telecommunication Software Solution technology company that provides its customers and clients with cutting-edge solutions to meet all their mobile digital services and digital banking requirements.

We value our customers' lives by providing easier, safer, and more efficient services.

Vas2Net Limited Technologies is a company that thrives on bringing our customers' businesses to the technological space to enable them to take their business to greater heights because we believe that through technology we can move forward than we ever had.

Vas2Net Technology Limited has branches in Nigeria, Ghana, Cameroon, South Africa, and Canada with Nigeria as the headquarters.

2.2 VISION OF VAS2NET TECHNOLOGIES LIMITED

To be the preferred mobile and digital technology solution provider out of Africa, with an ecosystem that improves digital banking and value-added services globally.

2.3 MISSION OF VAS2NET TECHNOLOGIES LIMITED

To deliver an integrated value-driven proposition that combines digital services, content, and digital banking seamlessly to customers.

2.4 VALUES OF VAS2NET TECHNOLOGIES LIMITED

1. Creativity and Innovation
2. Excellence (Reputation)

3. Passion
4. Teamwork and Respect
5. Courage

2.5 **SERVICES PROVIDED**

Vas2Net is a telecommunication Software Solution Technology Company focused on delivering to customer professional services to enhance their business. The services provided are:

1. Airtime
2. Data
3. USSD
4. BVN/NIN verification
5. Short code
6. Utility Bill Services
7. A2P SMS
8. Web/Software Development
9. RCS
10. Caller Ring Back Tone
11. Interactive Voice Response

CHAPTER THREE: RESPONSIBILITIES AND PARTICIPATION

3.1 RESPONSIBILITIES AND PARTICIPATION

In the 6 months I spent in Vas2Net Technologies Limited, I was responsible for:

1. collating and sending a daily report of network data for different network services like Airtel, MTN, GLO, and 9mobile
2. testing out website dashboard for our customer onboarding website
3. Developing the frontend for the automation website for collating the daily reports.
4. Learning and practicing the programming language assigned to me.

Picture 3.1.1 me at my workstation

Picture 3.1.2 me and my company based supervisor

CHAPTER FOUR: WORK EXPERIENCE AND KNOWLEDGE GAINED

4.1 ACTIVITIES DURING SIWES

I spent the totality of my 6 months learning about the field of Full Stack Engineering. I came into Vas2Net Technology Limited only knowing about Frontend Engineering to a very limited knowledge of only HTML, CSS, and JavaScript, to which was taught to me in the class when I took the course Introduction to Website Development (SENG 102) in Babcock University. During my SIWES I took up several online courses, videos and materials recommended or given to me by Vas2Net Technologies Limited to bolster my knowledge on Frontend Engineering and Backend Engineering to develop me into a well-rounded Full Stack Engineer. Below I will explain the different new technologies I learned during the 6 months SIWES:

4.2 INTRODUCTION TO PHP

What is PHP? PHP stands for Hypertext Preprocessor, it is an open-source, server-side, HTML embedded scripting language used to create a dynamic Web pages. It's a server-side, meaning the code is executed on the server, generating HTML, which is then sent to client. PHP can be embedded directly into HTML, making it easy to add functionality to web pages. PHP can be used with various databases like MySQL, PostgreSQL, etc. and on different platforms like Windows, Linux, MacOS, etc.

4.2.1 BASIC SYNTAX

The basic syntax of PHP starts with the php tag where all code is to be placed in between this tag

i.e.

```
 <?php
// PHP code goes here
?>
```

Semicolons end every line of code or else a SyntaxType error is encountered

i.e.

```
<?php
      Echo “Hello, World”;
      $x = 5;
?>
```

The variables start with a dollar sign ($), followed by the name of the variable. Like other programming languages variable names are case-sensitive.

```
<?php
      $txt = “Hello”;
      $x = 5;
      $y = 20;
?>
```

The datatypes supported by PHP are:

i. Strings
ii. Integers
iii. Floats
iv. Booleans
v. Arrays
vi. Objects
vii. NULL

e.g.

```
<?php
      $string = “Hello”;



      $integer = 42;
      $float = true;
      $array = array[“apple”, “banana”, “cherry”];
      $null = NULL;
?>
```

PHP like other programming language can perform several operations like Arithmetic, Logical and Comparison.

Arithmetic Operations:

```
<?php
      $a = 10;
      $b = 5;
      echo $a + $b;  // Addition: 15
      echo $a - $b;  // Subtraction: 5
      echo $a * $b;  // Multiplication: 50
      echo $a / $b;  // Division: 2
      echo $a % $b;  // Modulus: 0


?>
```

Logical Operations:

```
<?php
      $a = true;
      $b = false;
      var_dump($a && $b);  // Returns false (AND)
      var_dump($a || $b);  // Returns true (OR)
      var_dump(!$a);       // Returns false (NOT)
?>
```

Comparison Operations:

```
<?php
	$x = 5;
	$y = "5";



	var_dump($x == $y);   // Returns true (equal value)



	var_dump($x === $y);  // Returns false (not identical - different types)
	var_dump($x != $y);   // Returns false (not equal)
	var_dump($x !== $y);  // Returns true (not identical)
?>
```

PHP also has Control Structures like other programming languages:

If, else, elseif:

```
<?php
	$t = 20;
	if ($t < 20) {
	    echo "It's cold";
	} elseif ($t < 30) {
	    echo "It's nice";
	} else {
	    echo "It's hot";
	}
?>
```

Switch:

```
<?php
	$favcolor = "red";
	switch ($favcolor) {
	    case "red":
```

```
            echo "Your favorite color is red!";
            break;
        case "blue":
            echo "Your favorite color is blue!";
            break;
        default:


     echo "Your favorite color is neither red nor blue!";
    }
?>
```

While and do-while loops:

```
<?php
    $x = 1;
    while ($x <= 5) {
        echo "The number is: $x <br>";
        $x++;
    }

    $y = 1;
    do {
        echo "The number is: $y <br>";
        $y++;
    } while ($y <= 5);
?>
```

For and Foreach loops:

```
<?php
    for ($x = 0; $x <= 10; $x++) {
```

```
        echo "The number is: $x <br>";
    }

    $colors = array("red", "green", "blue");
    foreach ($colors as $value) {
        echo "$value <br>";
    }
?>
```

Functions:

PHP supports User-Defined functions and also has its own built-in functions that can be called and used by the user at any point.

User-Defined Functions:

```
<?php

    function writeMsg() {
        echo "Hello world!";
    }

    writeMsg();  // Call the function
    ?>
    Built-in Functions:
    <?php
    echo strlen("Hello world!");  // Outputs 12
    echo    str_word_count("Hello    world!");        //    Outputs    2ccs
    ?>
```

Arrays:

1. **Indexed Arrays**

```
<?php
      $cars = array("Volvo", "BMW", "Toyota");
      echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] .
      ".";
?>
```

2. **Associative Arrays**

```
<?php
      $age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
      echo "Peter is " . $age['Peter'] . " years old.";
?>
```

3. **Multidimensional Arrays:**

```
<?php
      $cars = array (
          array("Volvo",22,18),
          array("BMW",15,13),
          array("Saab",5,2)
      );

      echo    $cars[0][0].":    In    stock:    ".$cars[0][1].",    sold:
      ".$cars[0][2].".<br>";d
?>
```

Super Global Variables: This are predefined variables that are always accessible, regardless of scope. They hold information about server environment, user input and other important data. The global. The global variables are; `$GLOBALS, $_SERVER, $_GET, $_POST, $_FILES, $_COOKIE, $_SESSION, $_REQUEST, $_ENV`.

1. $GLOBALS: An associative array containing references to all variables currently defined in the script's global scope.

e.g.

```
<?php
      $x = 10;
      $y = 20;
      function sum() {
    $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}
      sum();
      echo $z; // Outputs 30
?>
```

2. $_SERVER: Contains information about headers, paths, and script locations

e.g.

```
<?php
      echo $_SERVER['PHP_SELF']; // The filename of the currently
      executing script
      echo $_SERVER['SERVER_NAME']; // The name of the server host
      echo $_SERVER['HTTP_HOST']; // The Host header from the current
      request
      echo $_SERVER['HTTP_USER_AGENT']; // The user agent string of the
      user's browser


?>
```

3. $_GET: An associative array of variables passed to the current script via URL parameters.

e.g.

```
<?php
      // For a URL like: test.php?name=John&age=30
      echo $_GET['name']; // Outputs: John
```

```
echo $_GET['age']; // Outputs: 30
?>
```

4. $_POST: An associative array of variables passed to the current script via the HTTP POST method.

   e.g.

```
<?php
    // Assuming a form was submitted with these fields
    echo $_POST['username'];
    echo $_POST['password'];
?>
```

5. $_FILES: An associative array of items uploaded to the current script via the HTTP POST method.

   e.g.

```
<?php
    echo $_FILES['file']['name']; // The original name of the file on
    the client machine
    echo $_FILES['file']['type']; // The mime type of the file
    echo $_FILES['file']['size']; // The size of the file in bytes
?>
```

6. $_COOKIE: An associative array of variables passes to the current script via HTTP Cookies.

   e.g.

```
<?php

    echo $_COOKIE['user']; // Outputs the value of the cookie named
'user'
```

```
?>
```

7. $_SESSION: An associative array containing session variables available to the current script.

   e.g.

```
<?php
      session_start();
      $_SESSION['username'] = 'John';
      echo $_SESSION['username']; // Outputs: John
?>
```

8. $_REQUEST: An associative array that by default contains the contents of `$_GET`, `$_POST`, and `$_COOKIE`.

   e.g.

```
<?php
      echo $_REQUEST['username']; // Will contain the username whether it
      was sent via GET, POST, or COOKIE
?>
```

9. $_ENV: An associative array of variables passes to the current script via the environment method.

   e.g.

```
<?php
      echo $_ENV['HOME']; // Might output something like /home/username
      ?>
```

WORKING WITH FORMS

HTML FORM:

The form structure is like normal HTML form structure, where you can insert into the HTML PHP code using the php tag.

```
<form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>
```

PHP FORM HANDLING:

```
<?php
      if ($_SERVER["REQUEST_METHOD"] == "POST") {
          $name = $_POST['fname'];
          if (empty($name)) {
              echo "Name is empty";
          } else {
              echo "Hi, " . $name;
          }


      }
?>
```

INCLUDING FILES

You can import and export files between one another using the include and require keywords.

e.g.

```
<?php
      include 'filename.php';//export the filename.php
      ?>
```

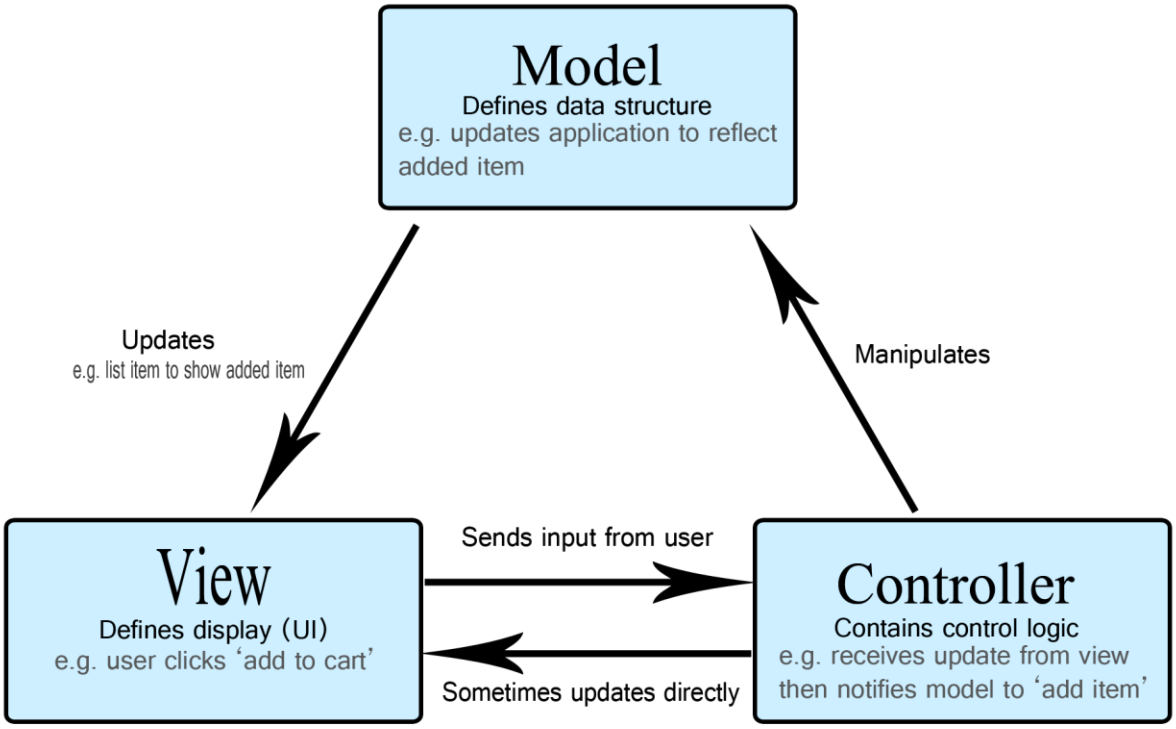
```
<?php
      require 'filename.php';//import the filename.php
?>
```

4.3. **MVC (Model, View, Controller) ARCHITECTURE:**

Model-View-Controller (MVC) is an architectural pattern in software design, particularly for developing user interfaces. The concept of MVC was introduced to me in a course at Babcock University Software Architecture and Design (SENG 307). MVC is an architectural pattern that splits an application into 3 interconnected components namely: Model, View, and Controller.

- Model manages the data, logic, and rules of the application
- View handles the display and presentation of data to the user
- Controller acts as an intermediary between the Model and View, processing user input and managing application flow.

This helps in the separation of application logic from the user interface and enables independent development, testing and maintenance of each component.



Picture 4.3.1 MVC Architecture

https://developer.mozilla.org/en-US/docs/Glossary/MVC

I learned a PHP framework called **CodeIgniter** using YouTube Videos and Materials recommended by my company (https://www.hostinger.com/tutorials/codeigniter-tutorial) which makes adoption of the MVC framework in PHP easier and provides some more functionalities to make coding easier.

4.4 **INTRODUCTION TO REACT.JS**

React is a popular JavaScript Library for building for building user interfaces especially Single Page Application (SPA).I used the Udemy course “The Ultimate React Course 2024: React, Next.js, Redux &More” recommended to me by supervisor. React was developed and is managed by Facebook. React make creating user interfaces easier where it combines the HTML like code into your JavaScript files directly through the use of a syntax extension called JSX (JavaScript XML).

React is Component based, declarative, efficient, and flexible. Component based where the UI is divided into reusable, self-contained pieces. Declarative where you describe what you want, and React handles the DOM manipulation. Efficient where it uses a virtual DOM for rendering therefore React is not destructive in nature. React can be used for both Mobile and Web development therefore making it flexible.

Example of how a component is written:

jsx

```
function Welcome(props) {
  return <h1>Hello, {props.name}</h1>;
}
```

As you can notice components are written like similarly to how a normal JavaScript function is written.

Things to note above is that the HTML part of the code is written in between the return () while JavaScript codes are written outside the return statement.

Counter Component

```
function Counter() {



const [count, setCount] = useState(0); // javascript code
  return (
    <div>
      <p>You clicked {count} times</p> //html code
      <button onClick={() => setCount(count + 1)}>Click me</button>
    </div>
  );
}
```

There can only be one parent element in a return statement e.g. like in the example above there is only one parent div element in which every other element is a child of. But an exception can be made using the shorthand syntax `<></>`

e.g.

```
function MyComponent() {
  return (
    <>
      <div>First parent div</div>
      <div>Second parent div</div>
    </>
  );
}
```

From the above there are now two independent div element in the return statement.

Some key concept I learned are:

1. State: State is like a container that holds data that may change over time. Whenever a state changes the components re-renders. A state is instantiated using the useState hook. The

useState hook consists of two things; the initial variable and the setter function. The initail variable is what holds the initial data passed in the useState parenthesis and the setter function is what is used to set the new value of the state.

e.g.

```
function Counter() {
  const [count, setCount] = useState(0); // instance of a state that holds the count data, count is the initail variable and setCount is the setter function
  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button> // setCount is setting the new value of count as the initial value of count plus one(1).
    </div>
  );
}
```

2. Props: Props pass data from parent to child component. Props are passed to components like HTML Attributes.

   Parent Component

```
function Parentcomponent(){
      const [score, setScore]  = useState(5);
      return(
            <Childcomponent    studentScore  =  score/>  //passing  the component
```

```
      )
}
Child Component
function Childcomponent({studentScore}){ //receiving the prop



return(
            <p> The student score is {studentScore} </p> //using the prop
      )
}
```

3. Event Handling: React uses synthetic events, which are wrappers around native browser events.

```
function Counter() {
  const [count, setCount] = useState(0); // instance of a state that holds
the count data, count is the initail variable and setCount is the setter
function
  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>Click me</button> //
the on click event
    </div>
  );
}
```

Unlike native JavaScript, it is called like an attribute on as part of the `addEventListner()` function.

4. Conditional Rendering: We use JavaScript conditions to render different components or elements based on the application's state.

   e.g

```
function Greeting(isLoggedIn) {
  return isLoggedIn ? <UserGreeting /> : <GuestGreeting />;



}
```

   If `isLoggedIn` is true then `UserGreeting` component is rendered but if `isLoggedIn` is false `GuesGreeting` component is rendered.

5. Form Handling: Unlike native JavaScript where form inputs handling there state, in React the state is handled by React where the input can now “controlled inputs”.

   e.g

```
function NameForm() {



const [name, setName] = useState('');


  function handleSubmit(event) {
    alert('A name was submitted: ' + name);
    event.preventDefault();
  }


  return (
    <form onSubmit={handleSubmit}>
      <label>
        Name:
        <input      type="text"      value={name}      onChange={e      =>
setName(e.target.value)} />
```

```
            </label>
            <input type="submit" value="Submit" />
          </form>
        );
      }
```

4.5 **Introduction to MySQL**

MySQL is a popular open-source relational database management system (RDBMS) that uses SQL (Structured Query Language) for managing and manipulating databases. It provides a standardized way to interact with databases, allowing users to define, manipulate, and retrieve data. I learnt about firstly in class from the course Database System Design, Implementation and Management (COSC 333) but was further reexamined and retaught to me by a fellow intern at the office.

Common MySQL Tasks

1. Creating and modifying database schema
2. Inserting, updating, and deleting data
3. Querying and retrieving data
4. Managing database transactions
5. Defining and enforcing data integrity constraints

Basic MySQL Commands

1. SELECT

Used to retrieve data from one or more tables.

```sql
```

```
SELECT column1, column2, ... FROM table_name;

SELECT * FROM table_name;
```

```

2. INSERT

Used to add new records into a table.

```sql
INSERT INTO table_name (column1, column2, column3, ...) VALUES (value1,
value2, value3, ...);
INSERT INTO table_name VALUES (value1, value2, value3, ...);
```

3. UPDATE

Used to modify existing records in a table.

```sql

```
UPDATE table_name SET column1 = value1, column2 = value2, ... WHERE
condition;
```

Example:

```sql
UPDATE Customers SET ContactName = Chukwuemeka, City = Lagos WHERE CustomerID
= 1;
```

4. DELETE

Used to remove records from a table.

```sql
DELETE FROM table_name WHERE condition;
```

Example:

```sql
DELETE FROM Customers WHERE CustomerName = Chukwuemeka;
```

5. CREATE TABLE

Used to create a new table in the database.

```sql
CREATE TABLE table_name (
    column1 datatype constraints,
    column2 datatype constraints,
    ...
);
```

Example:

```sql
CREATE TABLE Persons (
    PersonID INT AUTO_INCREMENT PRIMARY KEY,
    LastName VARCHAR(255),
    FirstName VARCHAR(255),
    Address VARCHAR(255),
    City VARCHAR(255)
);
```

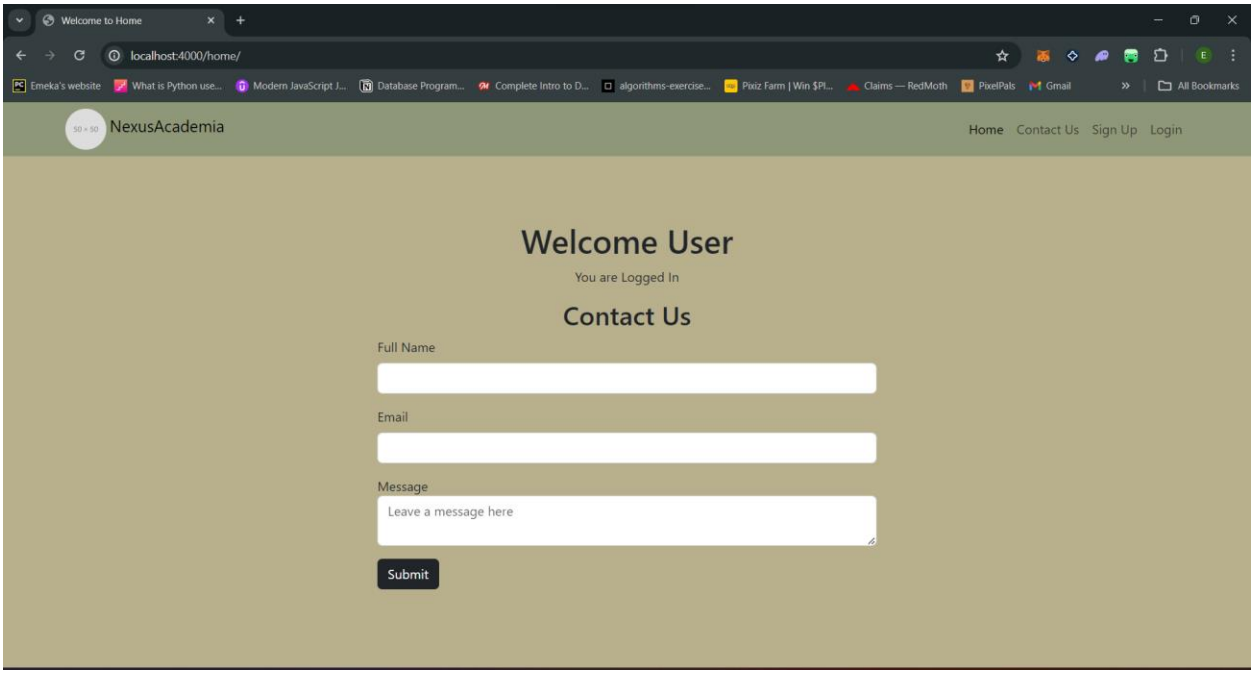
4.6 **Certification**

The API Academy's API Design course provides in-depth training on creating APIs that are user-friendly, scalable, and secure. It covers key concepts like RESTful API design, microservices architecture, and the application of industry best practices. The course also delves into the importance of consistency in API design, versioning strategies, and how to handle various data formats. Additionally, it addresses security practices necessary to protect APIs from common vulnerabilities, making it a comprehensive guide for anyone looking to master API design.

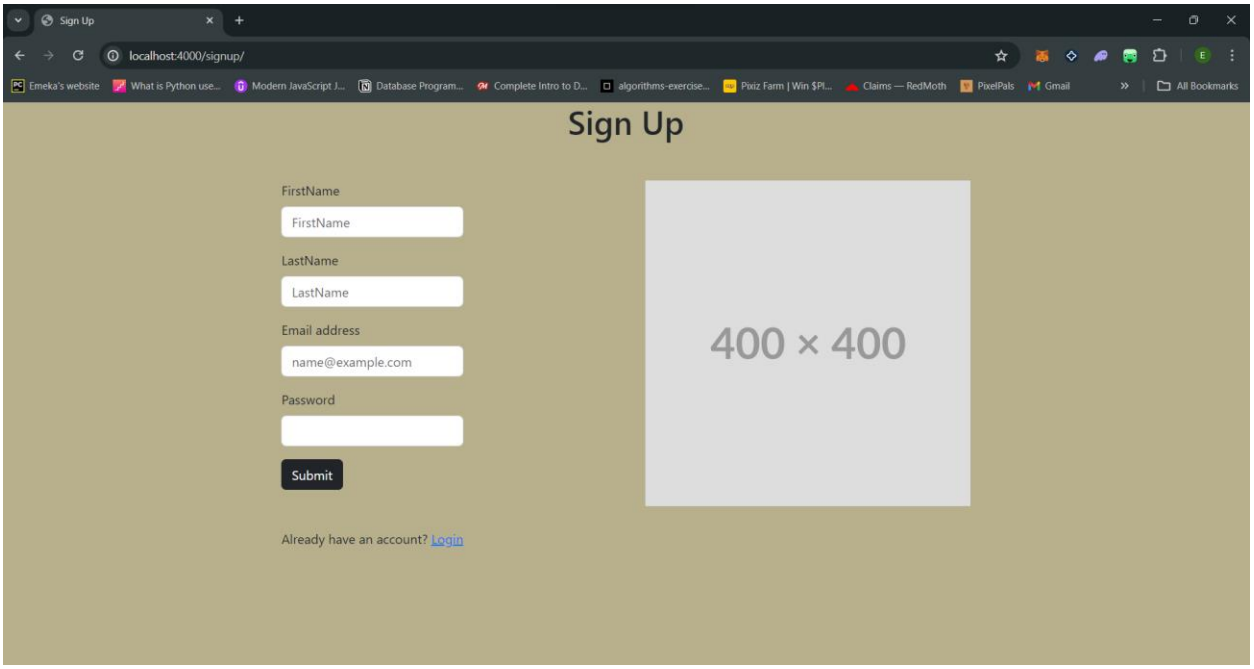
4.7 **Projects**

4.7.1 **Nexus Academia Website**

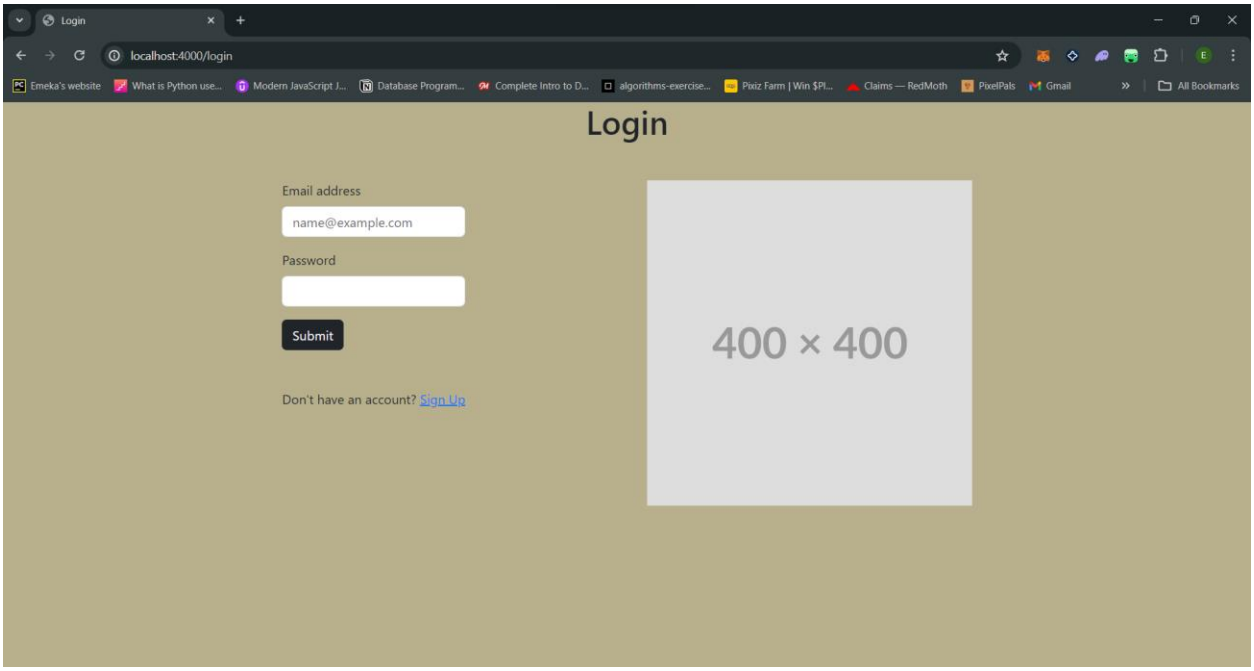
This is a school website I built fully in PHP with Bootstrap for the styling. It is a simple dashboard which is used where the user can login and signup for the school. It allows for login for the school website administrator. Also it allows for already logged in user to send message to administrator. The database language used is MySQL on XAMPP.



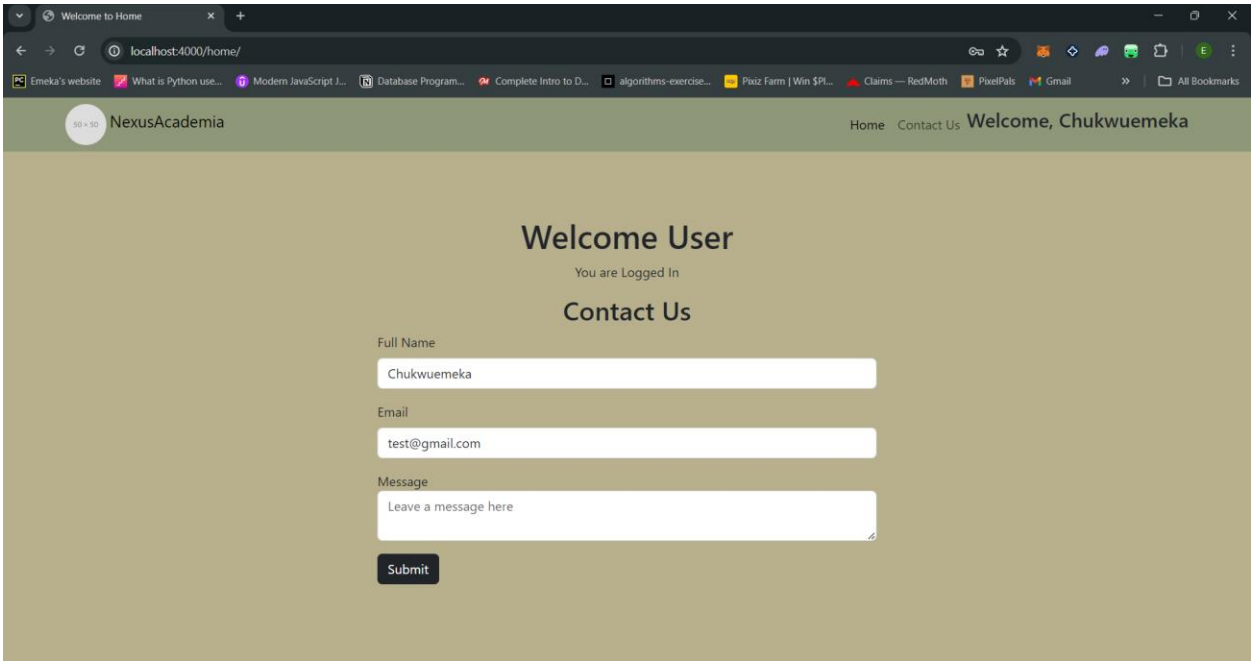
Picture 4.7.1.1 Nexus Academia home dashboard



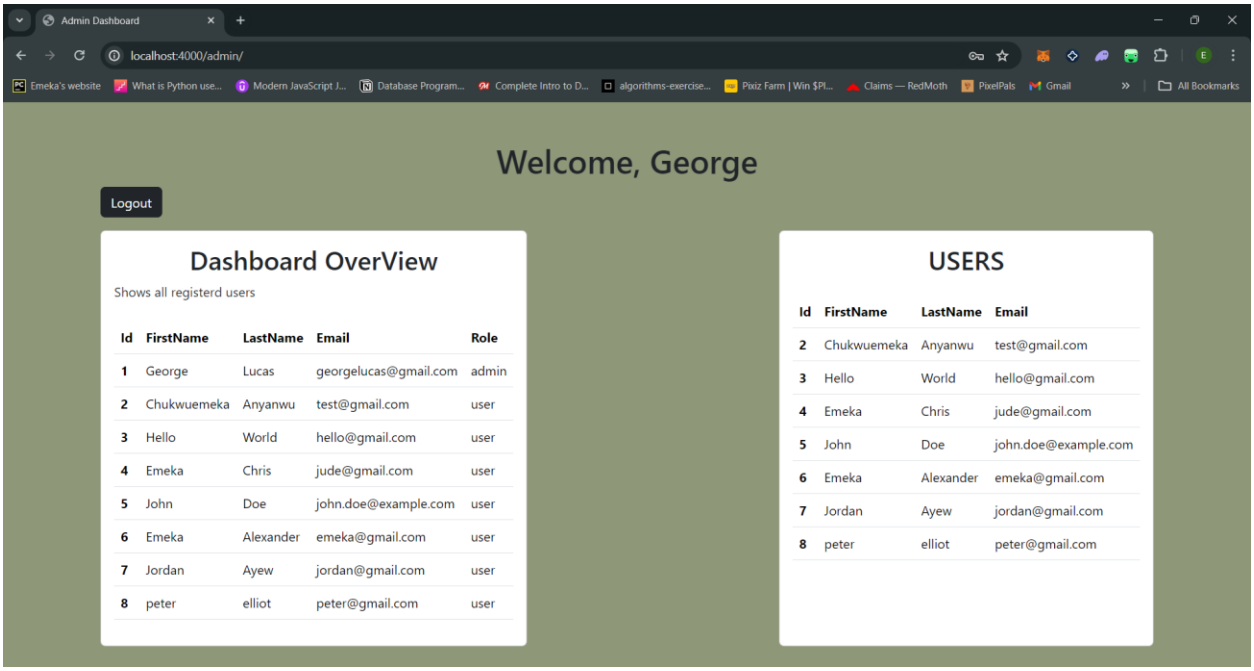
Picture 4.7.1.2 Nexus Academia SignUp page



Picture 4.7.1.3 Nexus Academia Login page



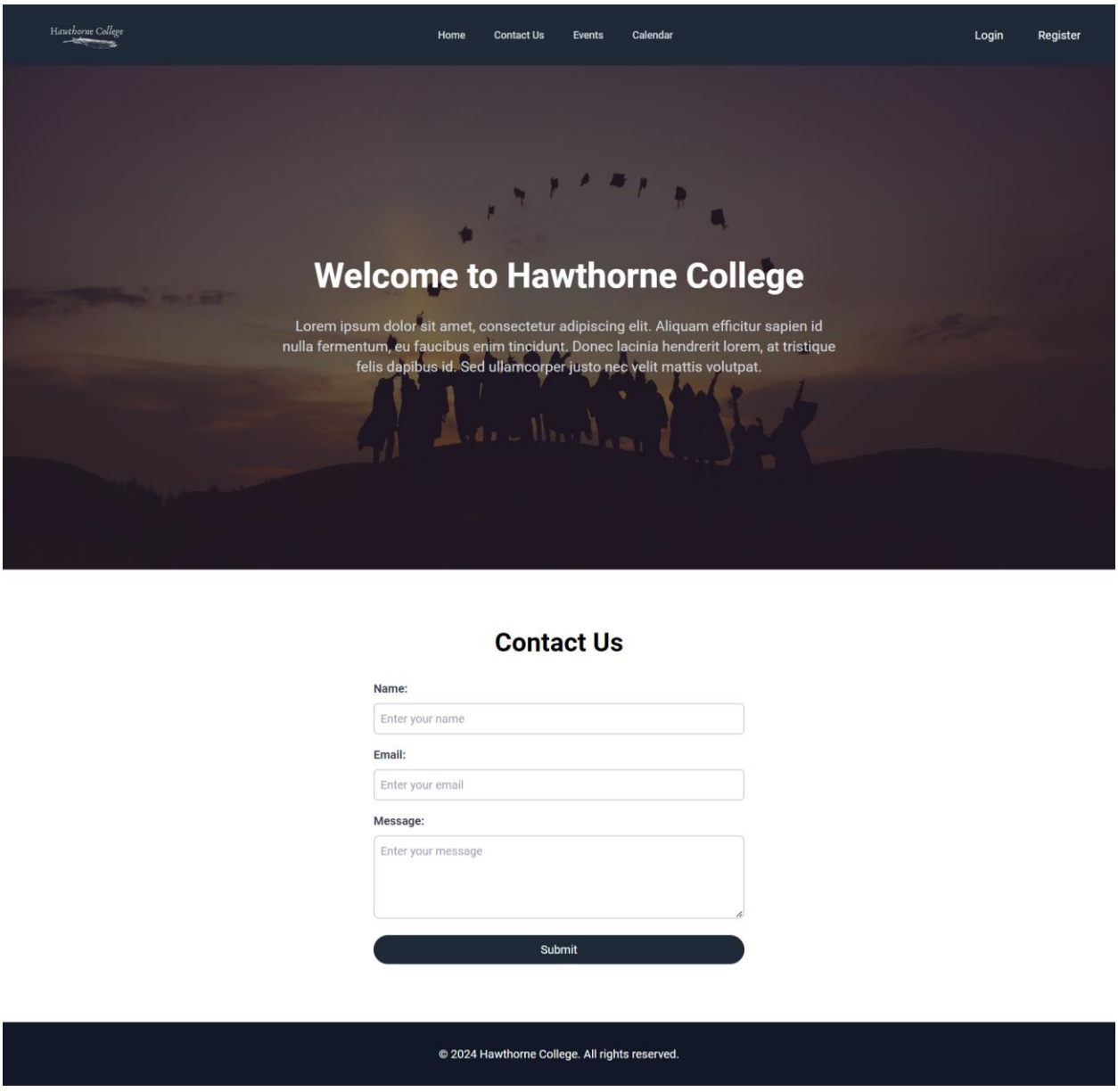
Picture 4.7.1.4 an already Signed In User dashboard



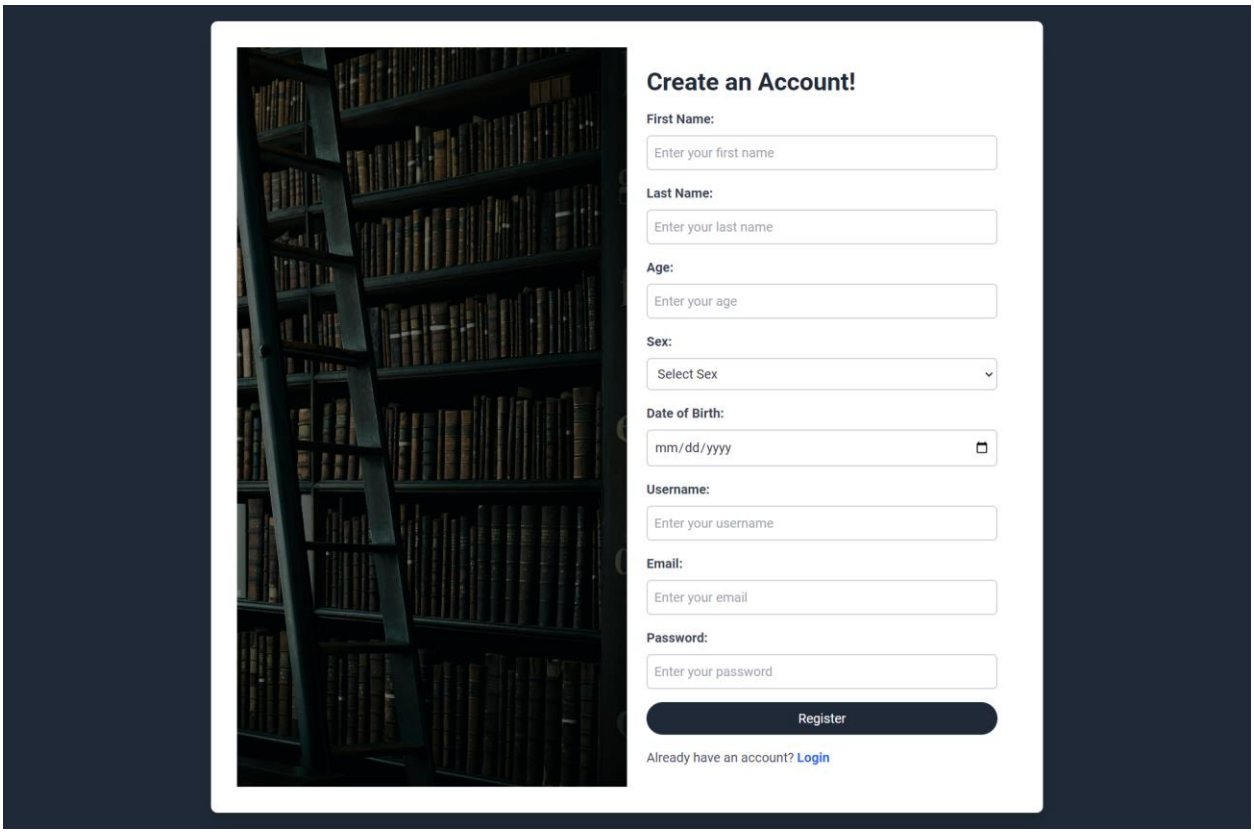
Picture 4.7.1.5 Nexus Academia Administrator Page

4.7.2 **Hawthorne College Website**

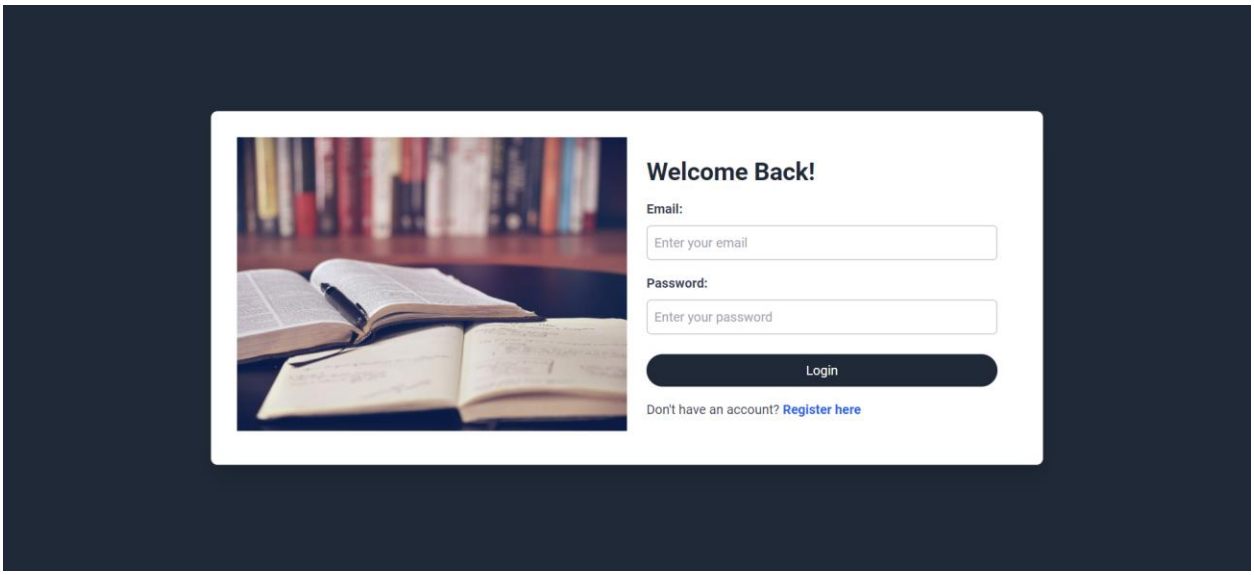
This is another school website that is more robust. It excludes a functional student dashboard that allows a for student registration and printing of student result based on semester. It also boost a robust administrator dashboard that allows for CRUD (Create, Read, Update, Delete) operations on student, lecturer, faculty and department. Built using PHP with MVC architecture implemented and MySQL for the database



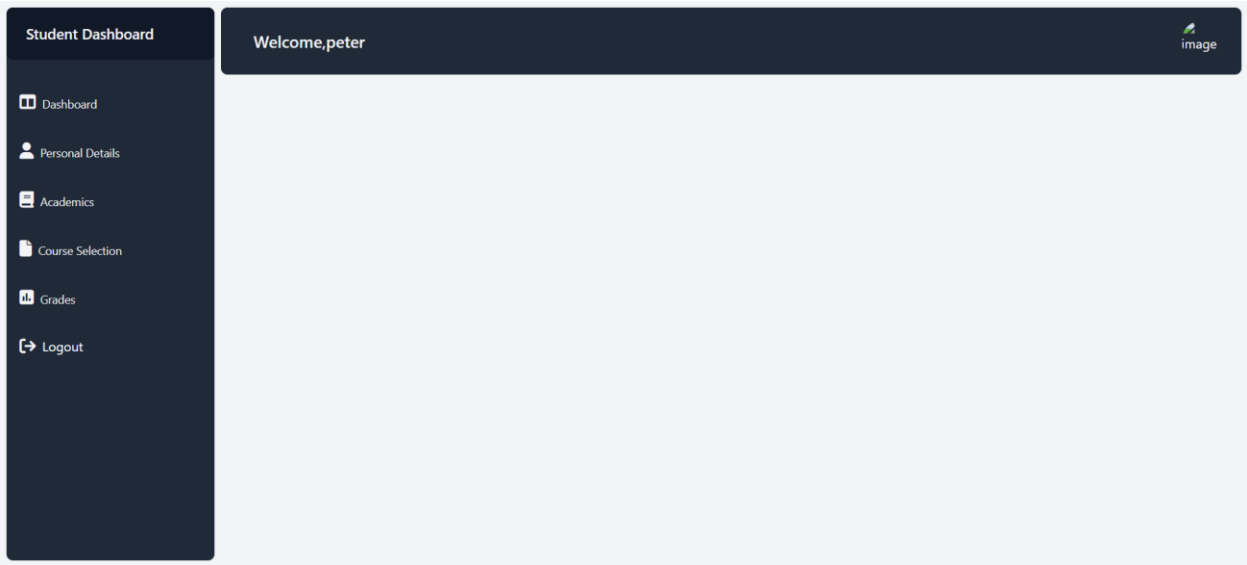
Picture 4.7.2.1 Hawthorne College home dashboard



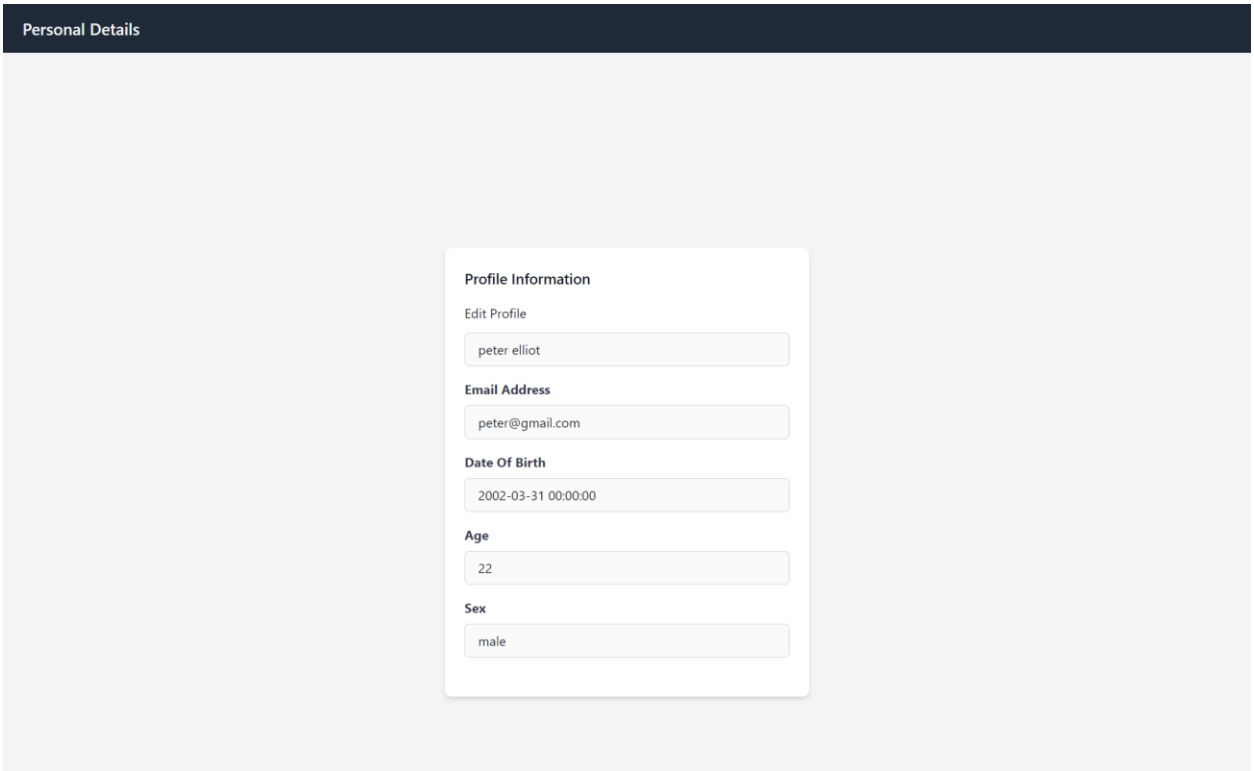
Picture 4.7.2.2 Hawthorne College Registration Page



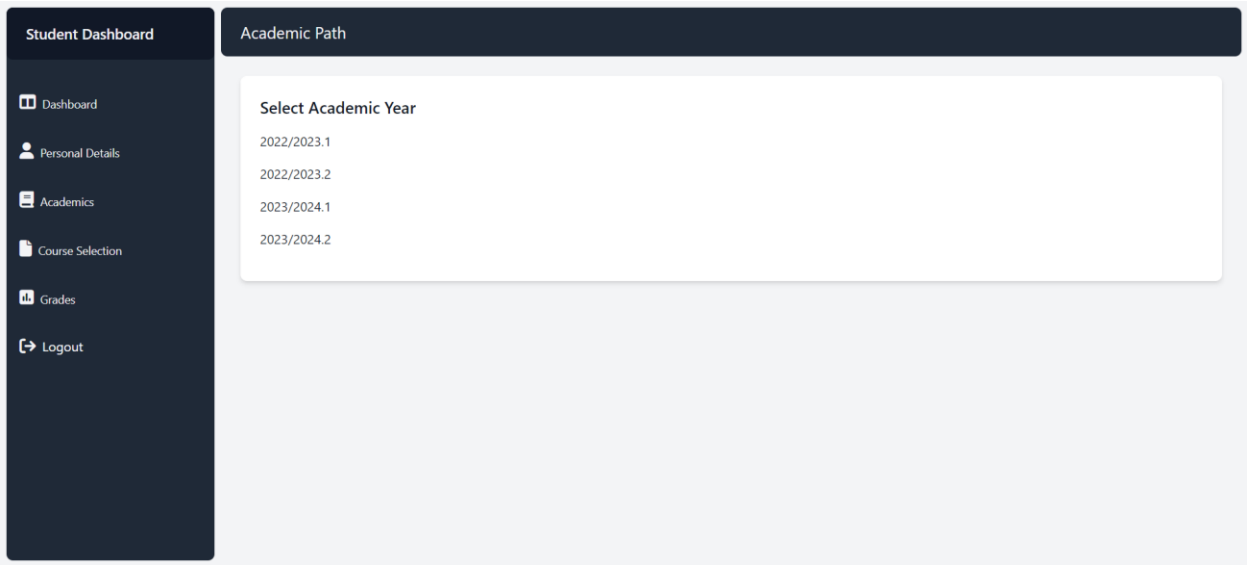
Picture 4.7.2.3 Hawthorne College Login Page



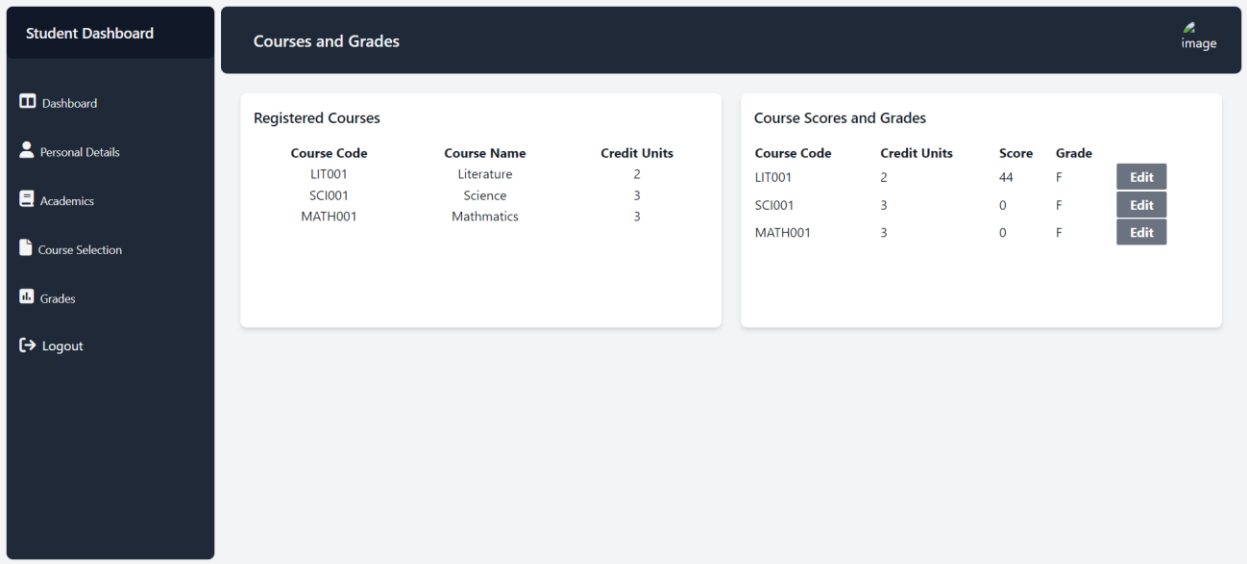
Picture 4.7.2.4 Hawthorne College Student Dashboard



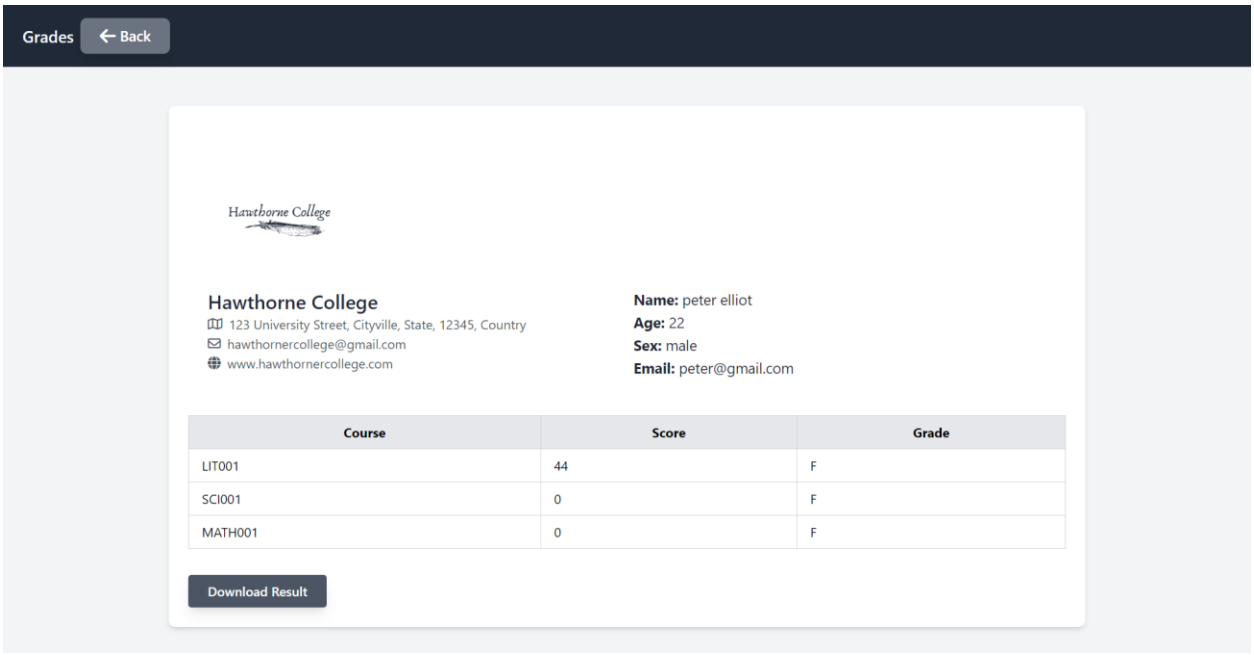
Picture 4.7.2.5 Hawthorne College Student Profile Page



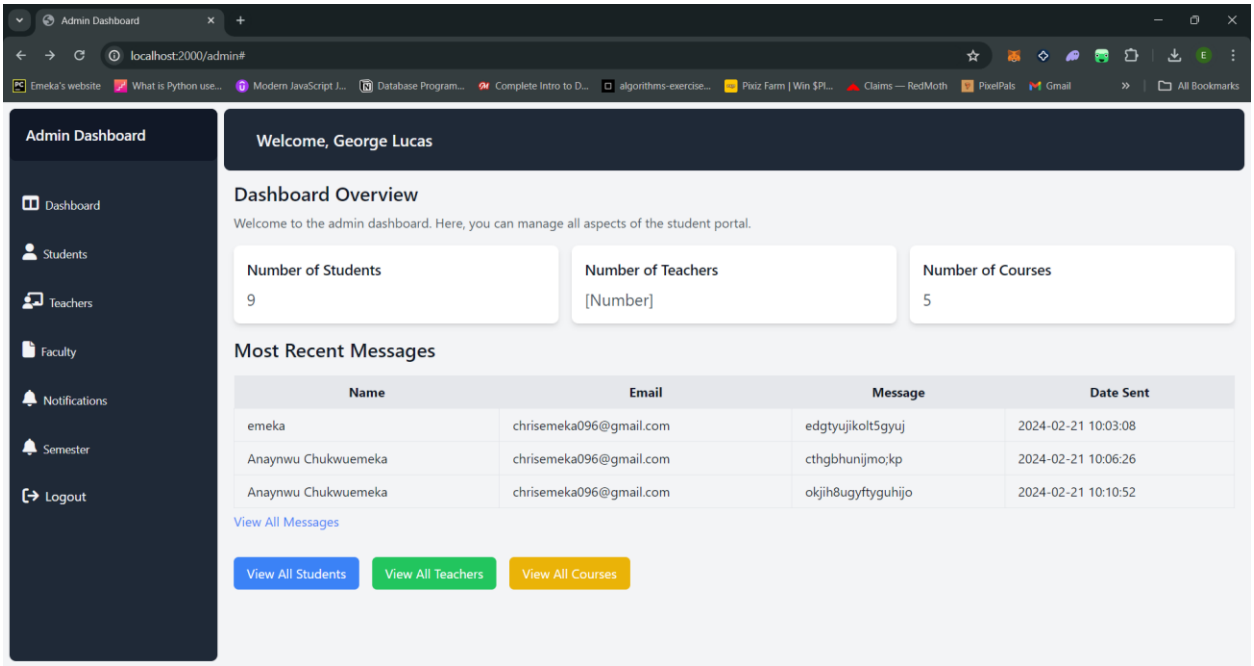
Picture 4.7.2.6 Hawthorne College Academic Year Page



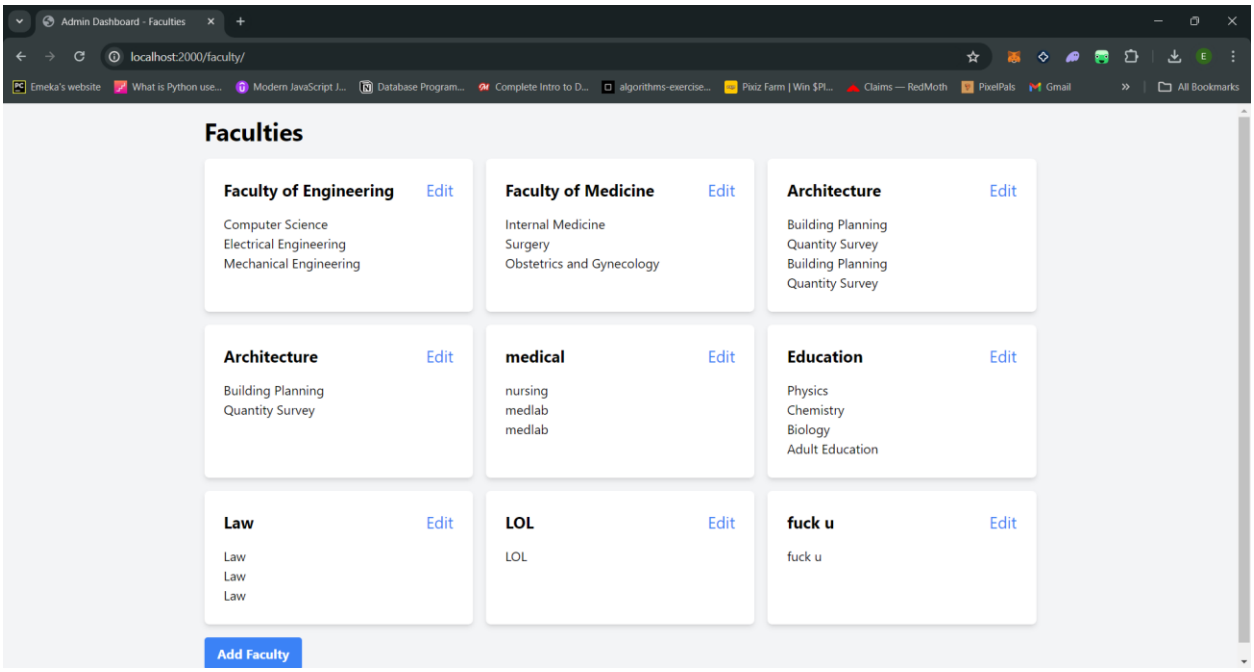
Picture 4.7.2.7 Hawthorne College Course and Grade Page



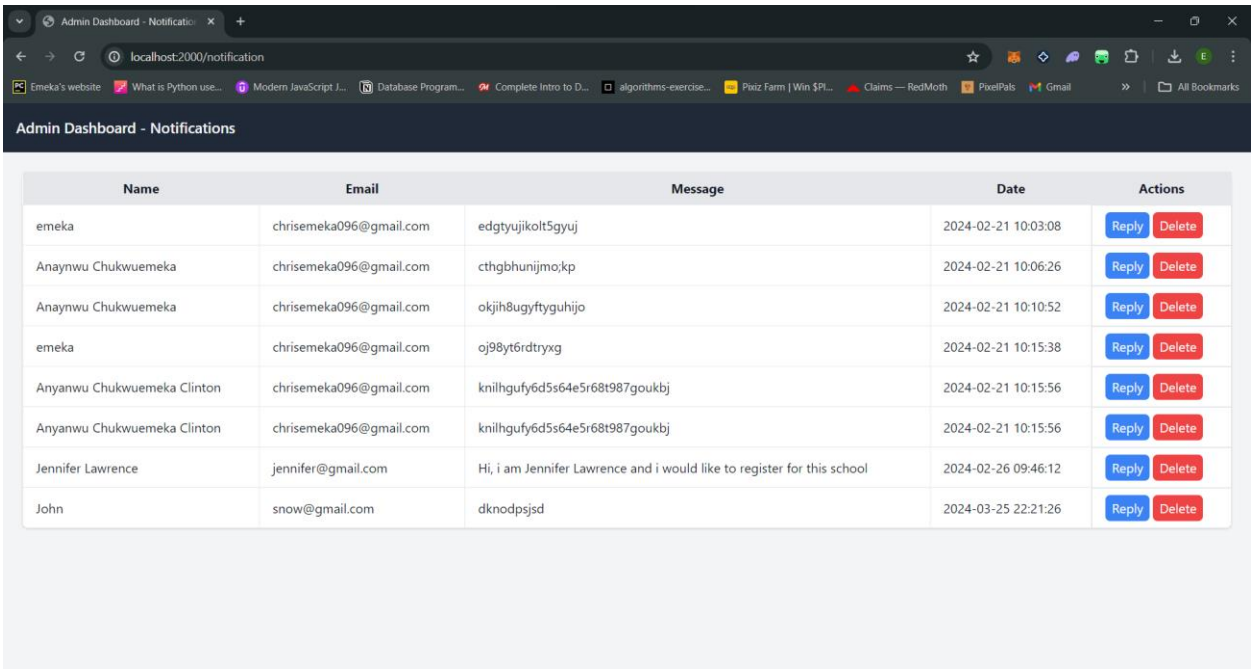
Picture 4.7.2.8 Hawthorne College Result Page



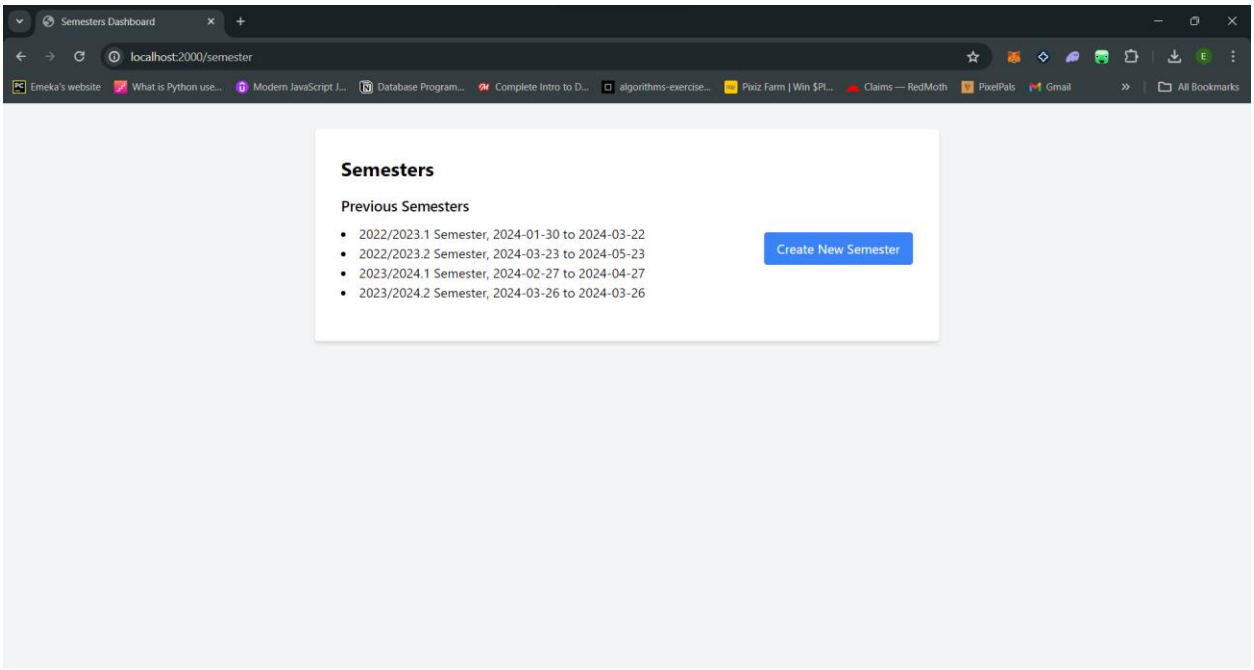
Picture 4.7.2.9 Hawthorne College Administrator Page



Picture 4.7.2.10 Hawthorne College Administrator Faculty Page



Picture 4.7.2.11 Hawthorne College Administrator Notification Page

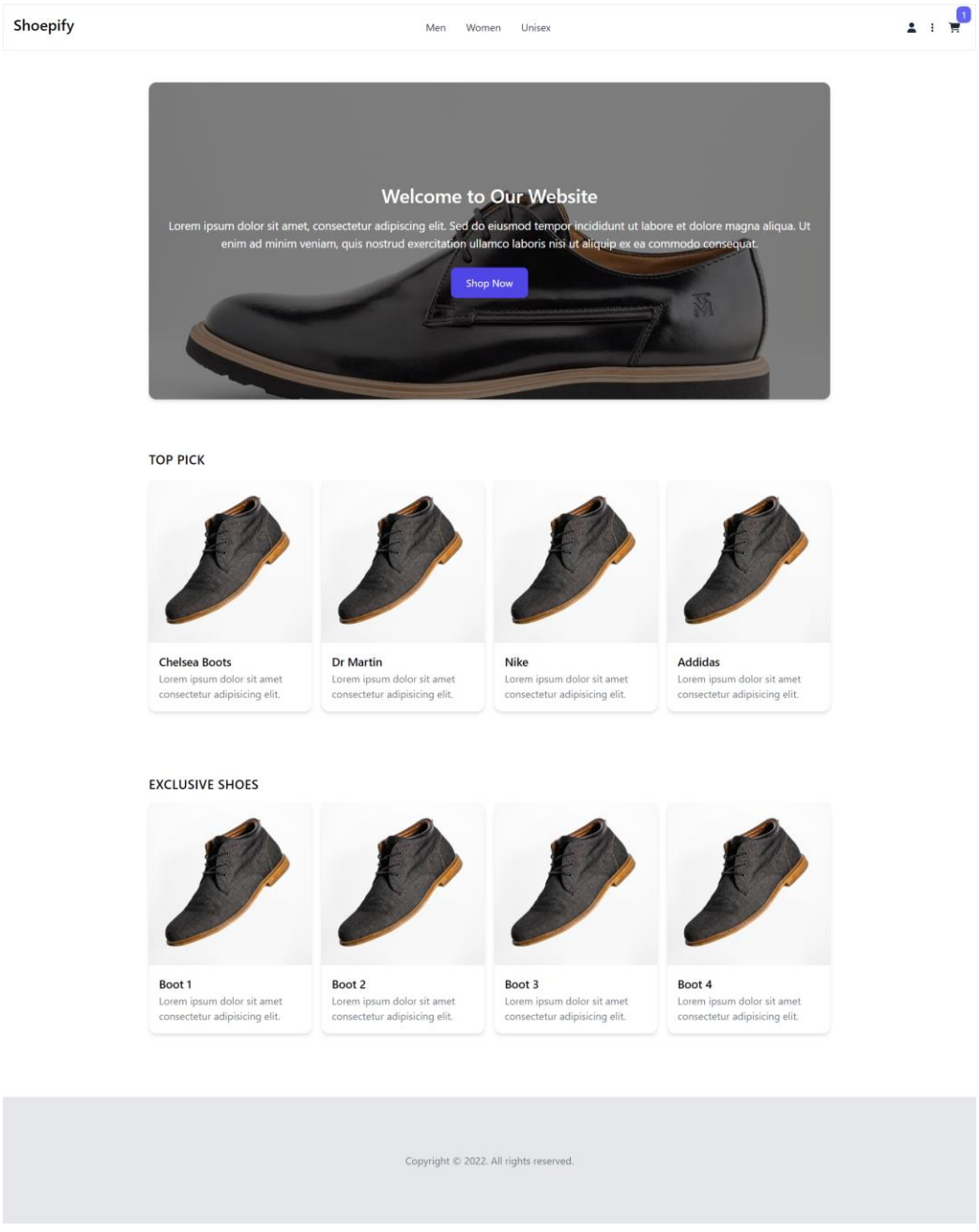


Picture 4.7.2.12 Hawthorne College Semester Page

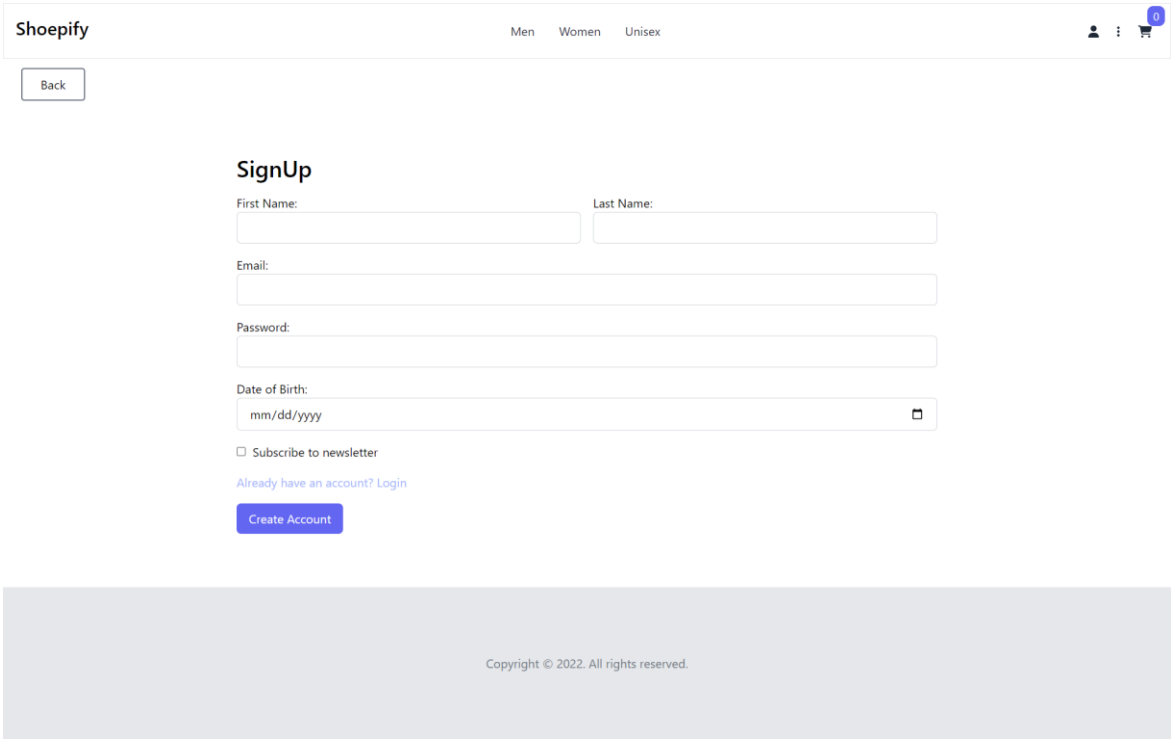
4.7.3 **Shoepify Website**

This website is an attempt of me to build an E-commerce store similar to Jumia but for only shoes.

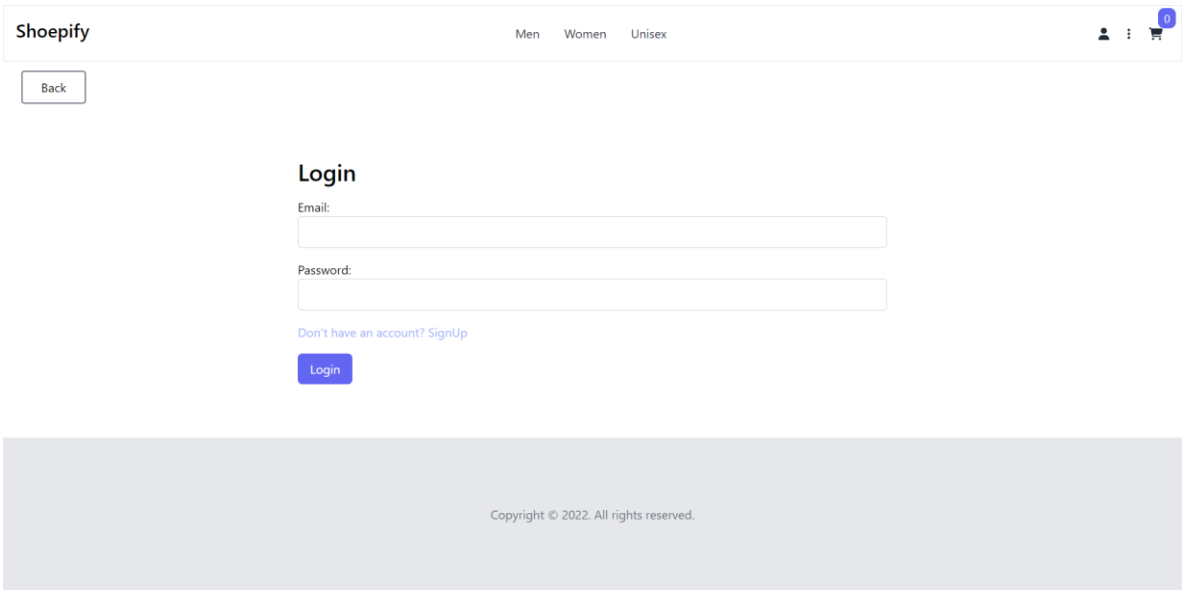
It was built using PHP with MVC architecture implemented.



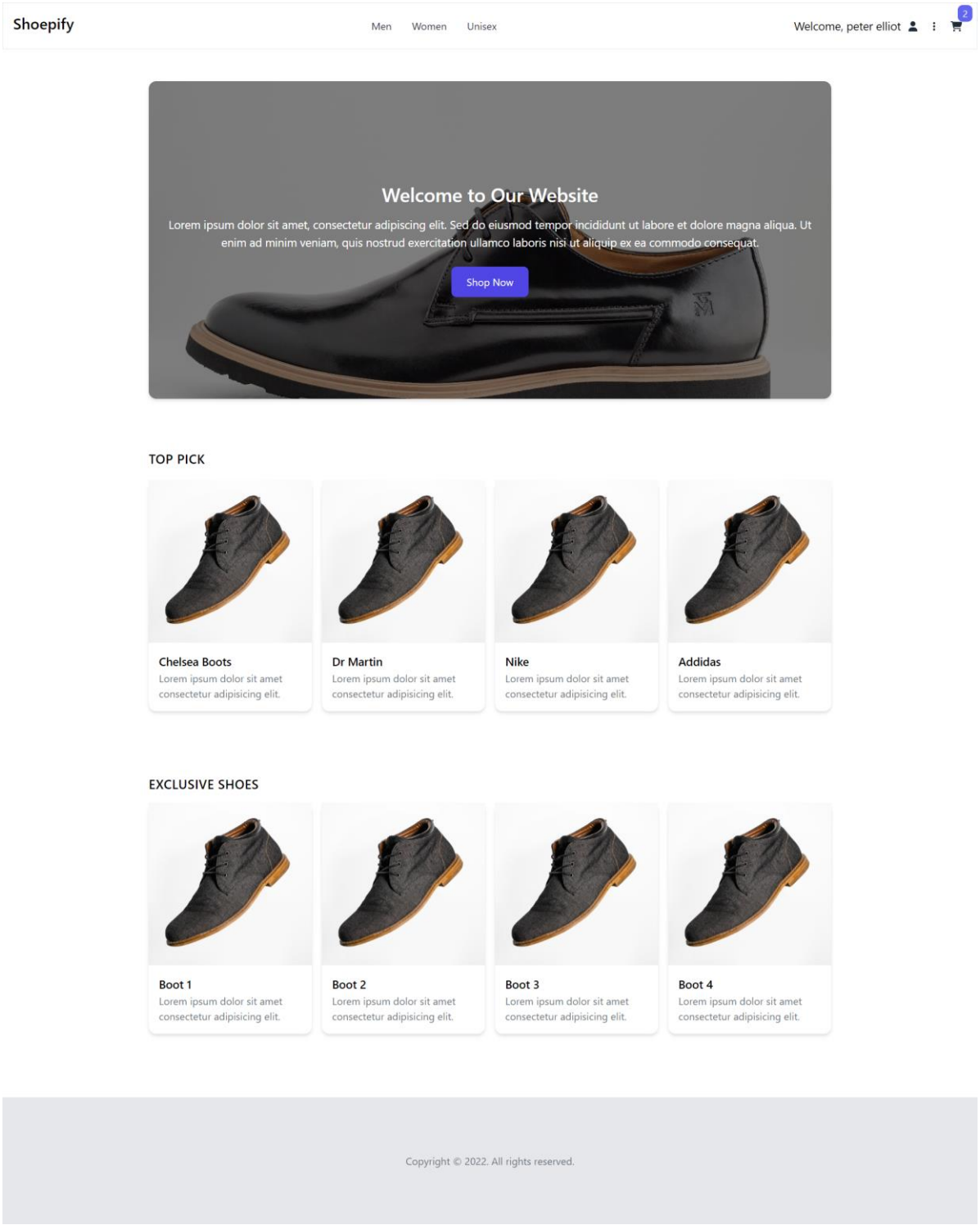
Picture 4.7.3.1 Shoepify Home Dashboard



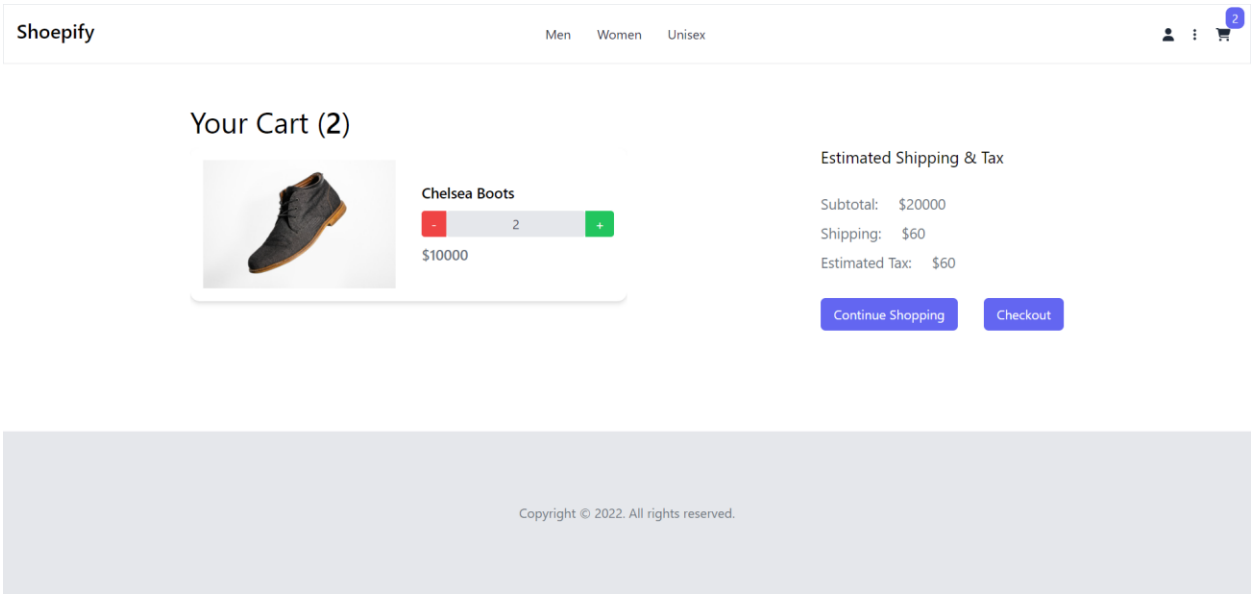
Picture 4.7.3.2 Shoepify SignUp Page



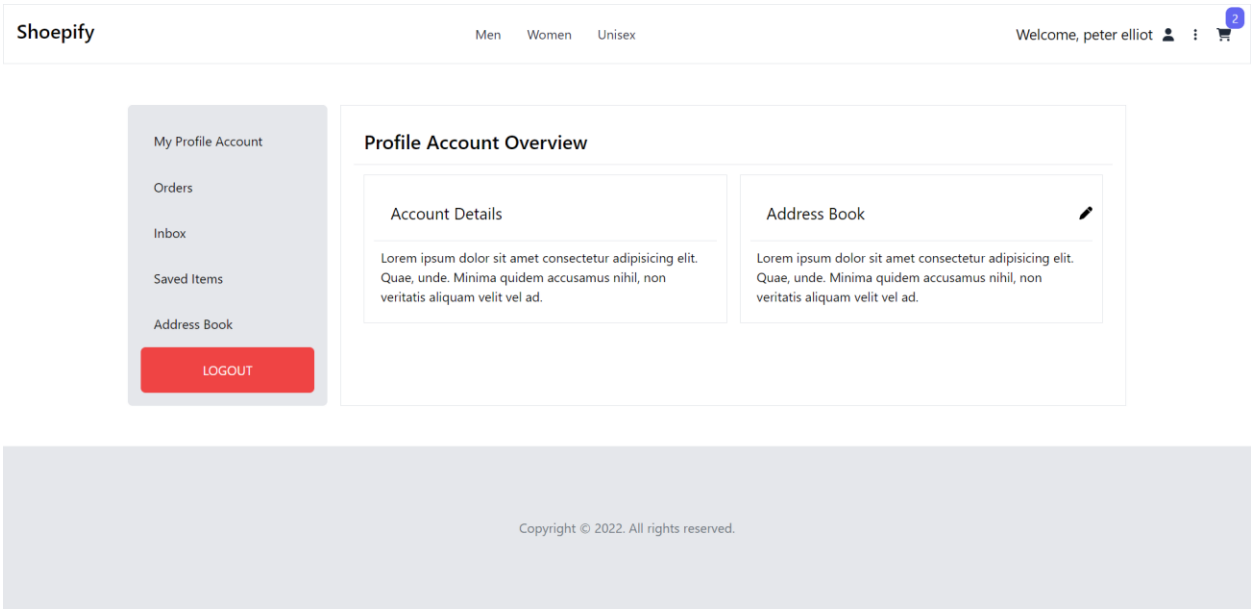
Picture 4.7.3.3 Shoepify Login Page



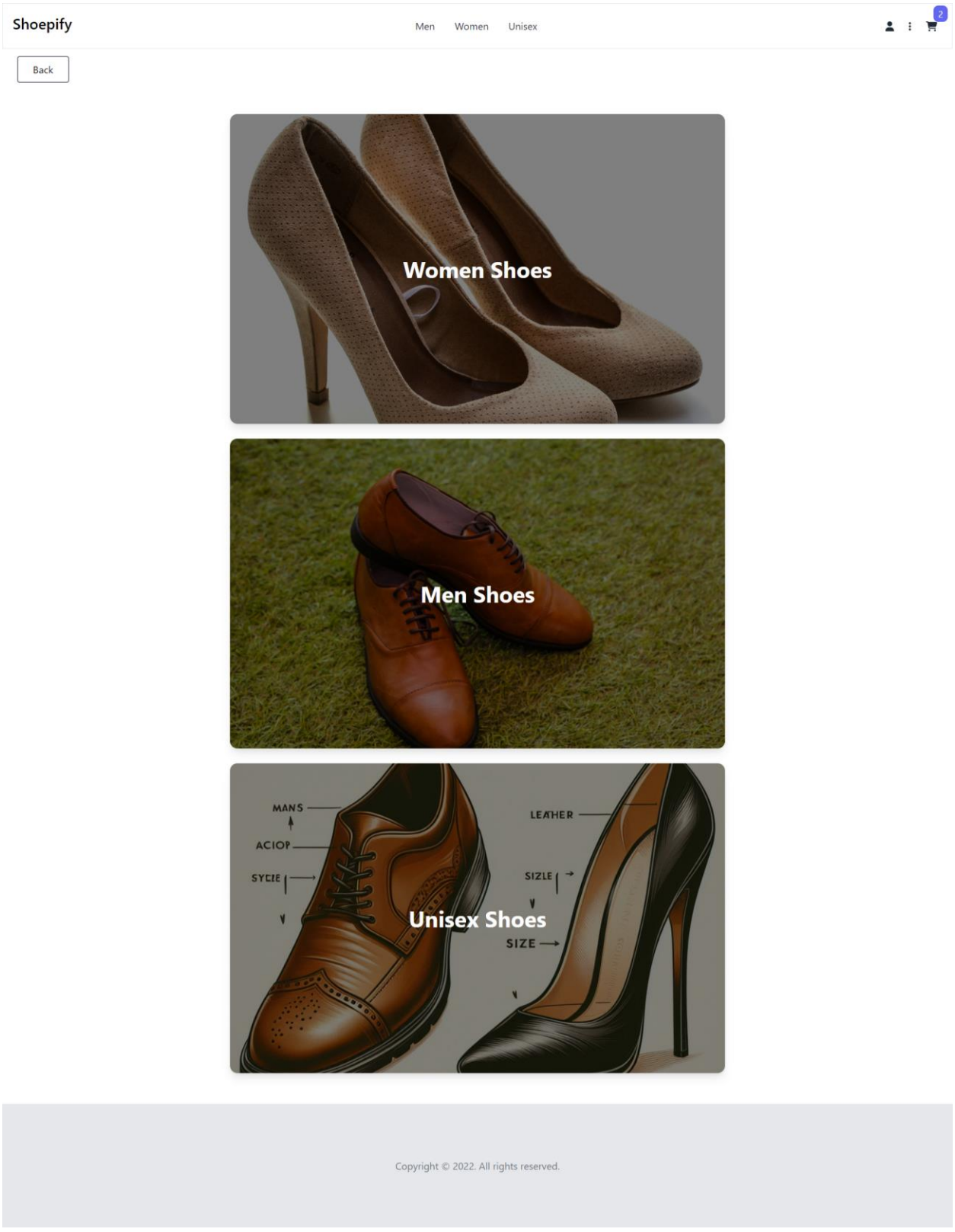
Picture 4.7.3.4 Shoepify Signed In User Home Dashboard



Picture 4.7.3.5 Shoepify Checkout Page



Picture 4.7.3.6 Shoepify User Profile Page



Picture 4.7.3.7 Shoepify Collection Page

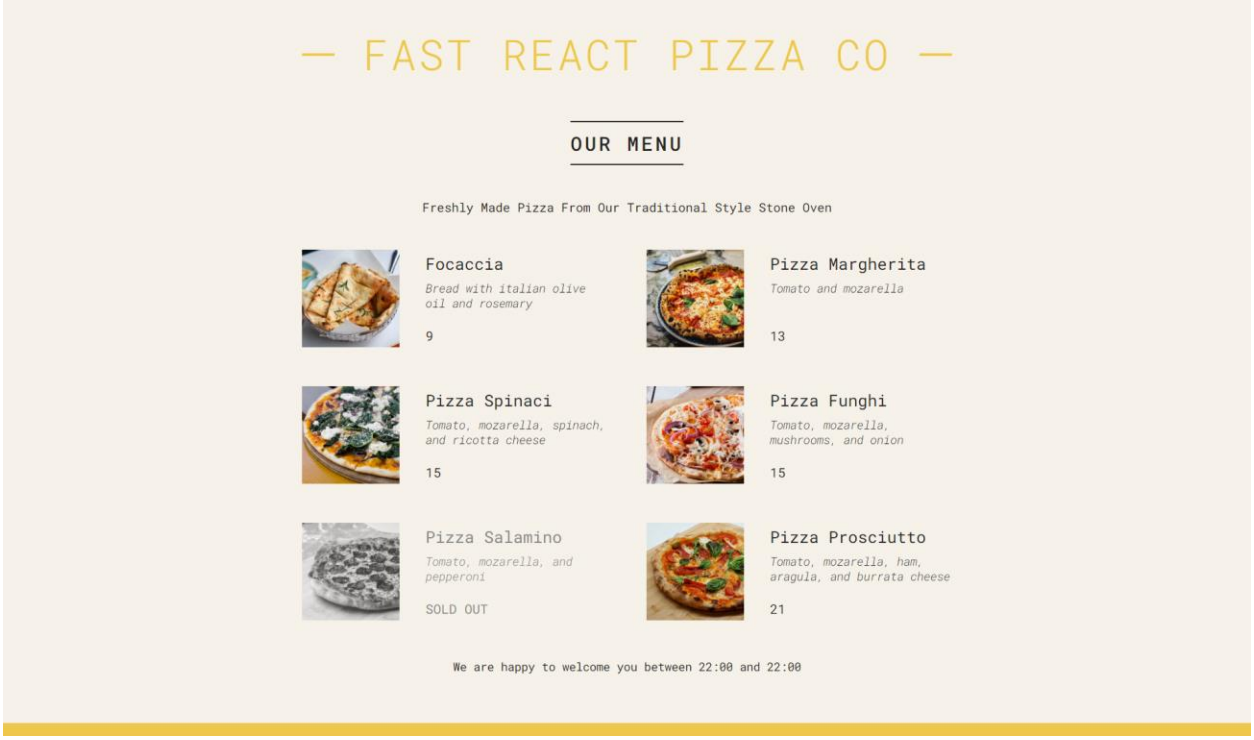
4.7.4 **Tip Calculator WebApp**

This is a React WebApp built using Create React App. It is a simple tip calculator app that calculates the tip a user wants to give to a waiter based on the user satisfaction with the meal.

Picture 4.7.4.1 Tip Calculator WebApp

4.7.5 **Pizza Menu WebApp**

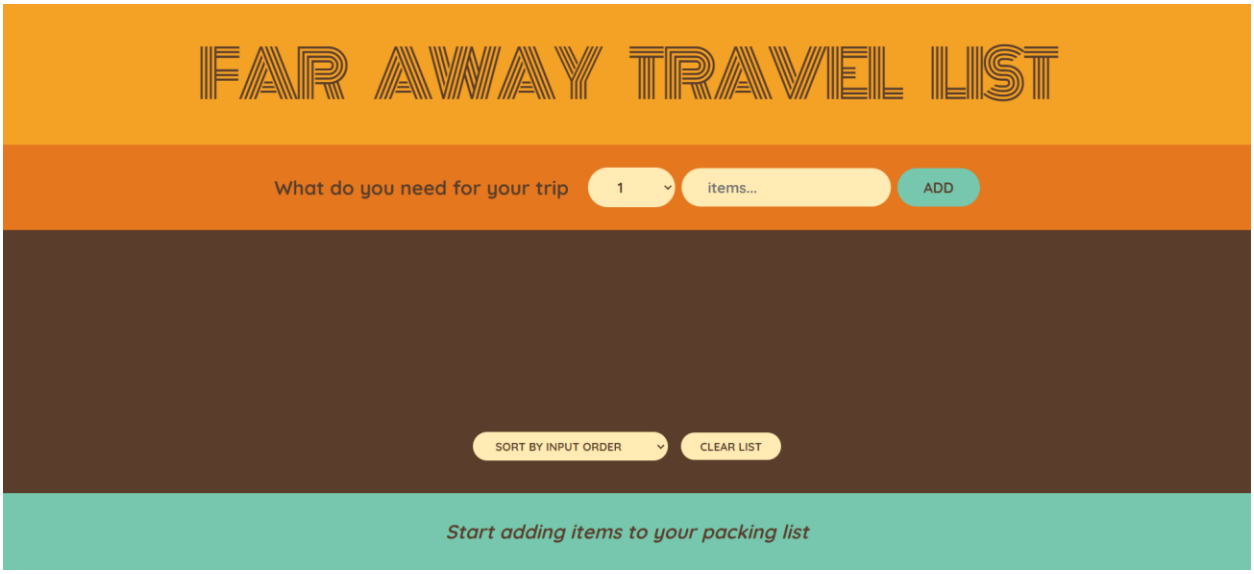
This is a React WebApp built using Create React App. It is a Pizza Menu App that shows the Pizzas available for order by a customer visiting website.



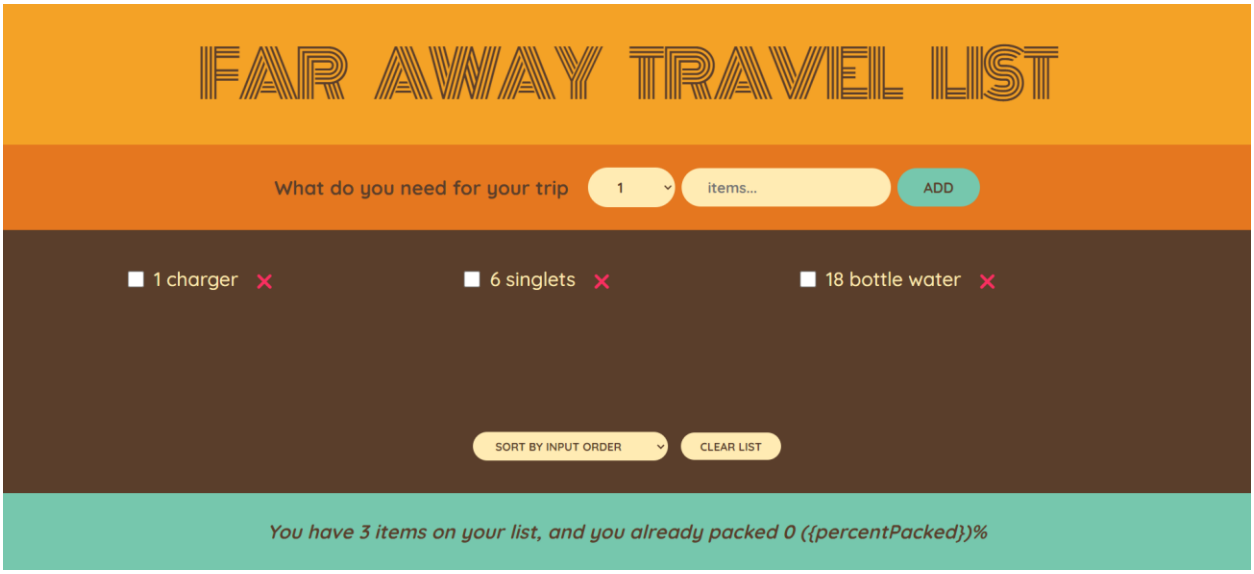
Picture 4.7.5.1 Pizza Menu WebApp

4.7.6 **TravelList_Webapp**

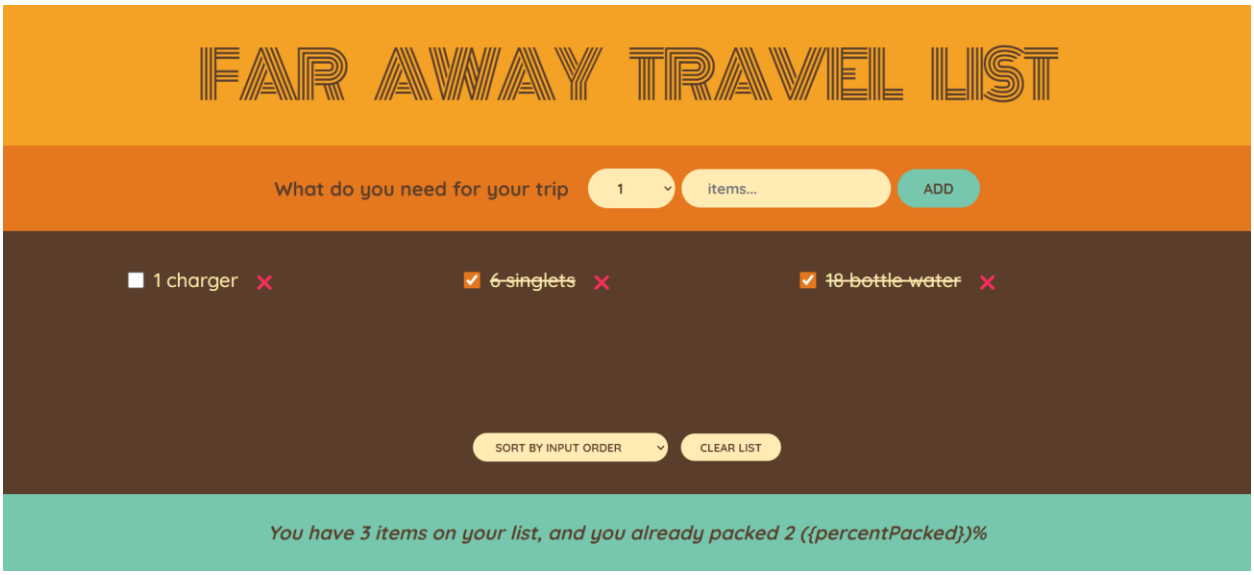
This is a React WebApp built using Create React App. It is a todo webapp but for travelling, it acts as a checklist for the items needed to be packed or gotten before going on a trip. User can add new items, cross out already gotten items, sort lists based different criteria and clear the entire list.



Picture 4.7.6.1 Travel List WebApp



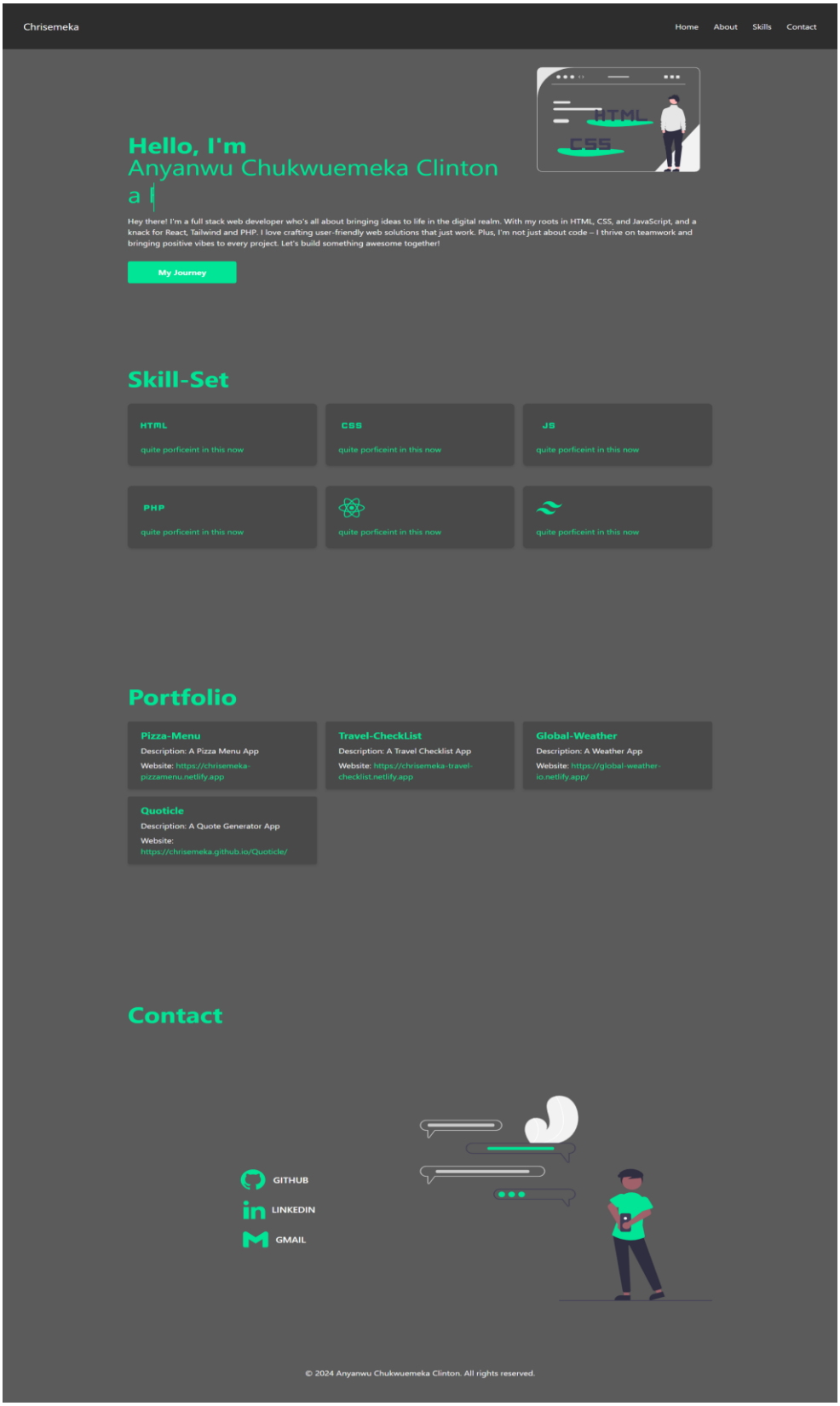
Picture 4.7.6.2 Items added to the list



Picture 4.7.6.3 Crossed out items

4.7.7 **Personal Porfolio WebApp**

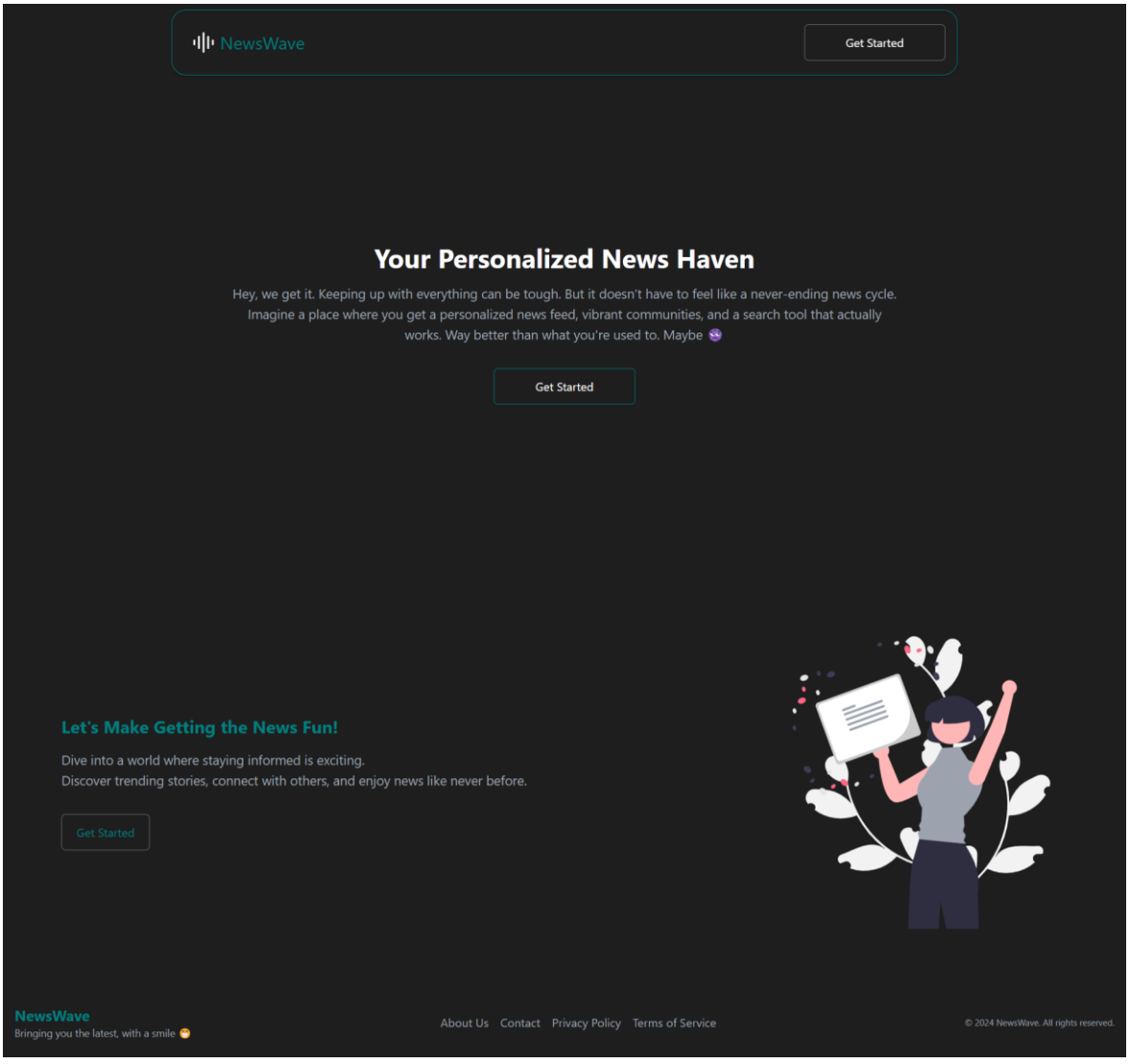
This is my updated personal portfolio website updated using React from the plain HTML, CSS and JAVASCRIPT. I added some animations to practice the little animations skills I learned



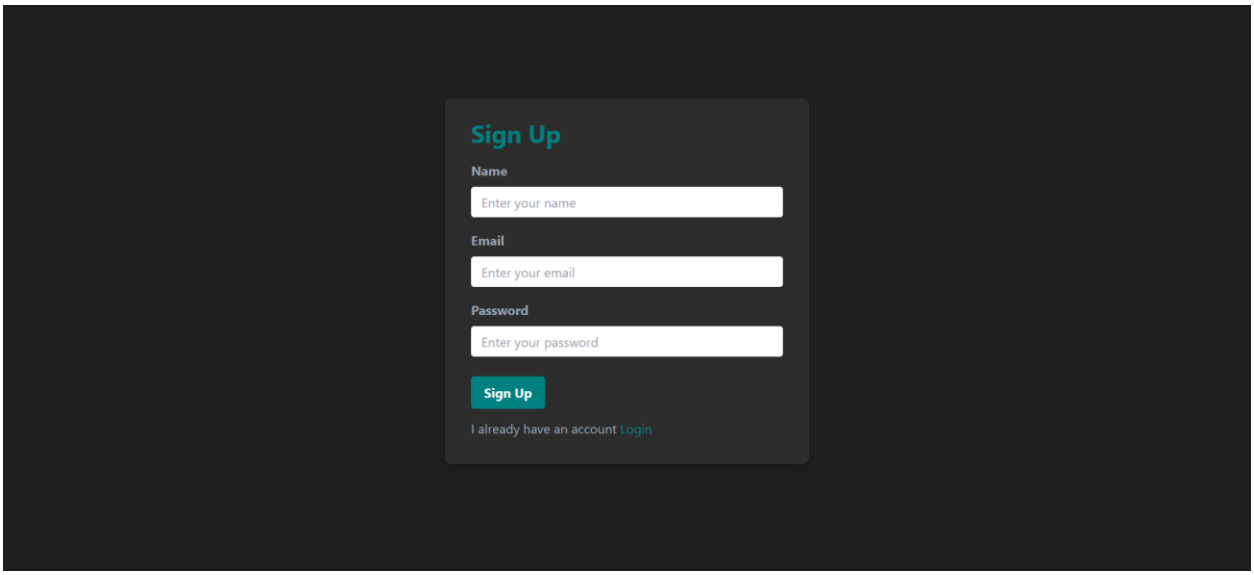
Picture 4.7.7.1 My Personal Portfolio

4.7.8 **NewsWave WebApp**

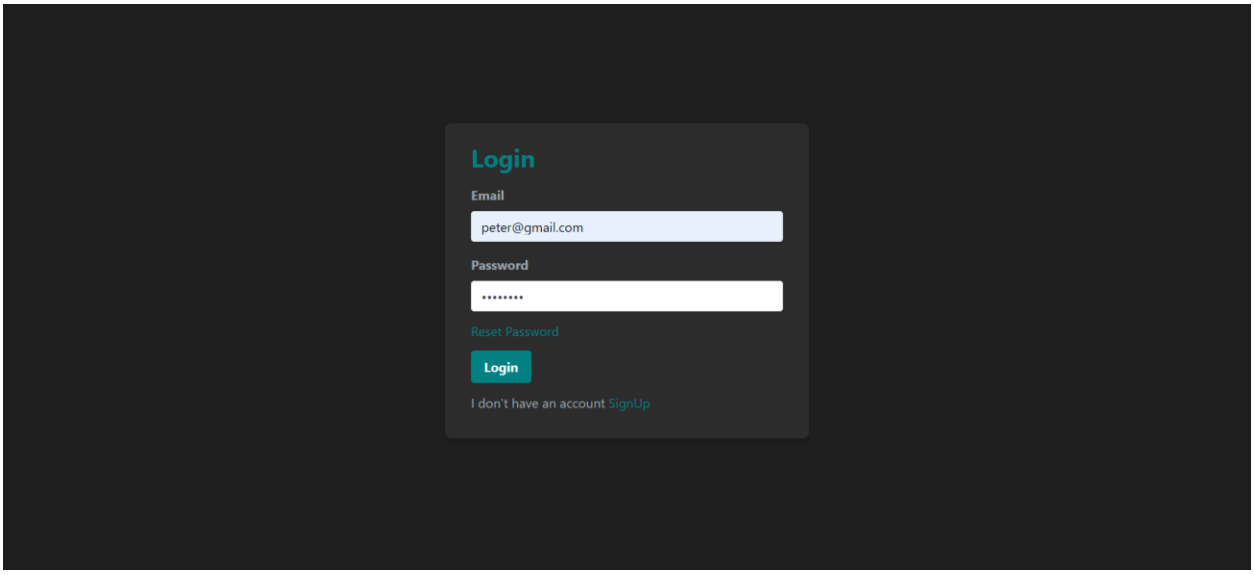
This webapp is a accumulation of all I have learned in PHP and React. NewsWave is a news feed webapp used to get news from different countries based on different topics. The news are gotten from Newsapi.org, using their API I am able to get news from all around the world. The frontend was built using React and Tailwind for styling. The backend was built using PHP and custom APIs are used for communication between Frontend and Backend. MVC architecture was implemented in the Backend and MySQL was used to store registered sign up users.



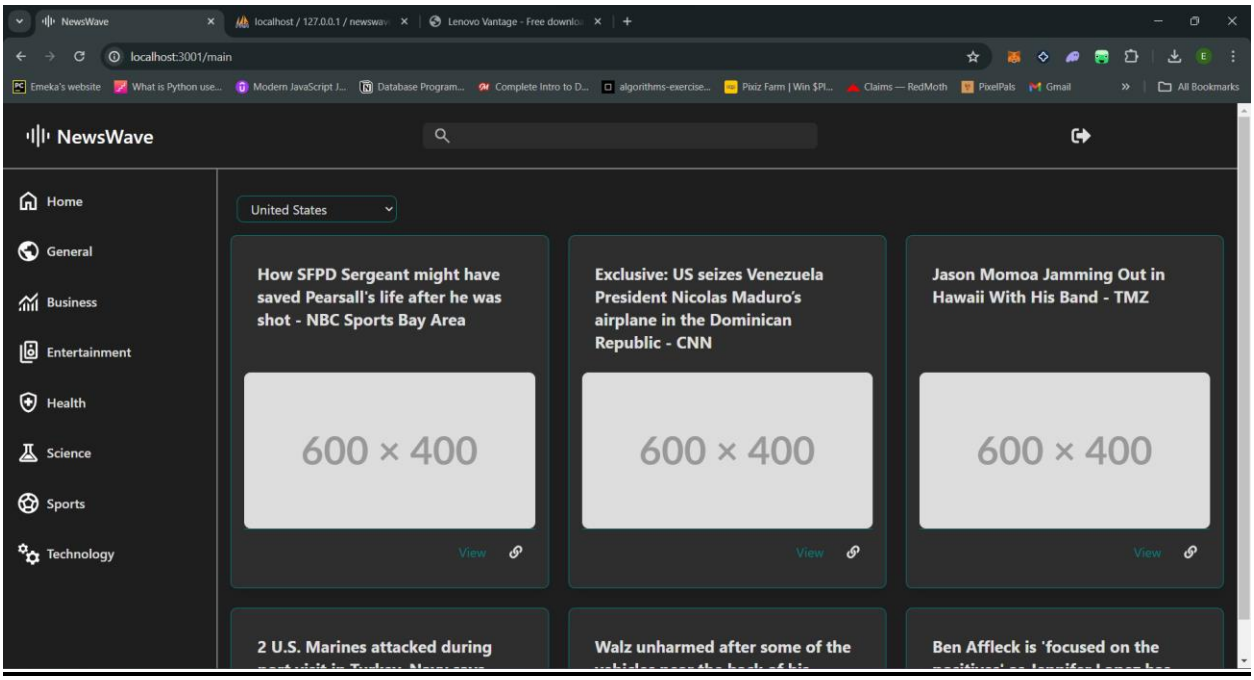
Picture 4.7.8.1 NewsWave Home Dashboard



Picture 4.7.8.1 NewsWave SignUp Page



Picture 4.7.8.1 NewsWave Login Page



Picture 4.7.8.1 NewsWave Logged In User Dashboard

4.8 **Challenges faced during the SIWES program**

Thankfully, I didn't face any major challenge during my SIWES program. The challenge I faced was adjusting to the hustle and bustle of a 8am to 5pm work lifestyle, where I had to wake up early to beat traffic and get transportation to the office. The only displeasing thing about my department was the fact that not much work or roles was given to interns as most works are sensitive works for the company which non-staff aren't allowed to work on. Other the these I had a wonderful experience and a good work environment.

CHAPTER FIVE: SUMMARY AND CONCLUSION

5.1 Summary

This report encompasses the activities carried out during my six-month industrial training experience at Vas2Net Technologies Limited in partial fulfilment of the requirements for the award of the degree of Bachelor of Science (B.Sc. Hons.) in Software Engineering, Babcock University. The Student Industrial Work Experience Scheme (SIWES) is a curricular provision of the Universities in Nigeria, where students of most technological-based courses are given the privilege to work with a company, industry, firm, or Ministry, that is related to their course of study, in order to get acquainted with the Methodologies, Instrumentation, Procedures, and mode of Processing acquired data and also to get them prepared for the life after School. This scheme has been a real benefit to me as most of the things I now know, I had no knowledge of prior.

I was able to further build my Frontend Development and also gain skills in Backend Development thereby fostering my transition to Full stack Development which has been a huge development in career. The SIWES program taught me to push myself and not just start in my comfort zone as a Software Engineer.

Picture 5.1.1 me in front of my company building

Picture 5.1.2 me and the intern i supervised

5.2 Conclusion

In conclusion I would like to once again appreciate God for his mercy during this 6 months SIWES and also appreciate Babcock University and Vas2Net Technologies Limited for the opportunity to be opposed to the real world of Software Engineering.

I accept that SIWES ought to be made required for all students who are learning at tertiary level regardless of their course of study as it also provides experience in areas that are only loosely related to the subject matter of our studies, such as timeliness, discipline, resource management, interactivity, efficiency, and many more.