|  |  |
| --- | --- |
| **Project Title** | **Application Integration - API** |
| **Qualification Name (NICF)** |  |
| **Product Name** |  |
| **Module Name (NICF)** | **Application Integration (API using Spring Boot & React JS)** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Student name** | | **Assessor name** | |
| Syukur Sidiq Nur Alam | |  | |
| **Date issued** | **Completion date** | | **Submitted on** |
|  |  | | 09/08/2023 |
|  | |  | |
| **Project title** | **Application Integration - API** | | |

|  |
| --- |
| **Learner declaration** |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature: Date: 09-08-2023 |

1. Project Background

The Know-Your-Neighborhood application was developed using Spring Boot..

This assignment gives an opportunity to demonstrate your capabilities in the following

areas:

• Be able to design and develop a backend using Spring Boot and JPA Framework.

• Be able to develop API using Restful Web Services.

• Be able to develop frontend application using React JS.

• Be able to identify existing APIs and its uses in already developed application

1. Project Objective

* - The goal of this project was to analyze various existing APIs, assess their suitability, identify security vulnerabilities, and integrate the connectivity functionality of the selected APIs into existing web pages. The analysis involves evaluating documentation, stability, support, and compatibility. Security tests will also be conducted. Once a suitable API is selected, integration with the website is done through the development process and necessary configuration. The end goal is to enhance the functionality of the website by using the latest APIs and ensuring adequate acceptance and security.
* Tools and Platform

1. Windows 10 Pro
2. Visual Studio Code
3. React JS
4. Facebook API
5. Spring Suite Tools
6. Postman
7. MySQL Workbench
8. Axure RP
9. Google Chrome
10. Microsoft Word
11. Project Requirement Specification

The application can use existing APIs to log in and retrieve basic information such as name and email from the API. Users can also register/login manually.

The Know Your Neighborhood website consists of the following main pages:

1. Home Page
2. Registration Page
3. Login Pages with API link
4. Contact us Page
5. About us Page
6. Terms and Conditions Page
7. Task 1

**APIs and Type of APIs**

1. Explain what API is, its role and need for API and research existing APIs.
2. Examine the relationship between API and SDK.
3. Identify types of API and its uses.
4. Identify the potential security issues with API and critically evaluate the suitable API for given scenario or your selected application.

**Solution:**

1. **Explain what API is, its role and need for API and research existing APIs.**
2. What is API?

Application programming interfaces are called APIs. One of the interfaces that allows you to connect one program to another is the API itself, in other words, the API acts as a bridge between different apps running on the same or different platforms. Additionally, the API itself can be used to communicate with several programming languages. Certainly easier for developers. In fact, developers can use APIs to get the data they need from other platforms, so they don't have to provide all the information themselves.

An example of an API that can be used is a restaurant waiter API. A waiter's job in a restaurant is to introduce the chef to the guests. So customers just order food from the menu list and the waiter informs the chef. After a while, the waiter returns to the customer with the prepared food.

1. Its Role

The purpose of APIs is to provide applications with a standardized way to request resources and services from other software components without having to understand the intricate details of how those other software components work internally. That's it. It acts as an intermediary that enables the exchange of information and the execution of actions between different software systems, facilitating integration.

1. Need for API

Various software components can use APIs to communicate and transfer data. It allows developers to bridge the gaps between small discrete pieces of code to create robust, robust and secure applications that meet the needs of their users. APIs improve collaboration by facilitating interoperability, enabling modular development, promoting specialization, facilitating third-party integration, accelerating development, and supporting service composition. By leveraging APIs, teams can collaborate effectively, seamlessly integrate systems, and combine the strengths of multiple parties to create robust software solutions.

1. Example APIs for mobile, desktop, Web APIs.

Here are some examples of APIs for mobile, desktop and web APIs:

* Mobile APIs:
  + Google Maps API
  + Spotify Web API
  + GIPHY API
* Desktop APIs:
  + PayPal API
  + Windows API
* Web APIs:
  + Google Maps API
  + Google Calendar API
  + YouTube Data API
  + Facebook API
  + Twitter API

1. **Examine the relationship between API and SDK.**

A Software Development Kit (SDK) is a collection of software development tools provided by a platform or company. SDKs often consist of many parts, such as APIs, libraries, documentation, and other tools that help programmers build applications for specific platforms and environments. The SDK makes it easy for programmers to use functionality provided by the platform or company.

APIs define interfaces and protocols for interacting with software systems. At the same time, an SDK is a comprehensive package containing APIs to simplify development for a particular platform or system, as well as additional tools, libraries, and resources. The API is the core component of the SDK and provides a way to access functionality. SDKs, on the other hand, extend the API by providing a complete set of development tools and resources.

An SDK can optionally contain one or more platform-specific APIs. Developers use her SDKs and APIs to connect apps to existing platforms or systems, shortening application development cycles. SDKs typically include one or more APIs that developers use to access and take advantage of functionality provided by the platform, developers, and companies. This is the relationship between API and SDK.

1. **Identify types of API and its uses.**
2. Examine different APIs
   * 1. Public API

Public API is often referred to as Open API. As the name implies, Public API is an API that anyone can use across platforms.

Also, this type of API is the easiest to use. You can sign up or directly take advantage of it in your application. For example, the API for Google Maps.

* + 1. Private API

Unlike the public API, Private API is an API not open for public use. Usually, this type of API is created for internal purposes in developing certain applications.

For example, the API from the back end is used to access the front end of a website. Or applications for mobile application development.

* + 1. API Partners

API partners can be used for public purposes, but only those with permission can use them. As with public APIs, you must register with the API provider first. Then, use it only in certain applications according to the agreement. For example, the Pinterest API.

* + 1. Composite API

Composite API is an API that stores data from various servers or hosts in one place. Of course, this really saves time for the user. That's because users can get various data types in just one access.

1. Examine the uses of APIs for a particular type

• Web APIs:

Web APIs are used for many purposes, including:

- Integrate third-party services (e.g. social media platforms, mapping services) into web applications.

- Retrieve data from external sources, such as weather information or stock market data.

- Grant access to certain web application features, such as user authentication or file downloads.

• REST APIs:

The REST API finds applications in:

- Create mobile applications that use data from server-side resources.

- Develop client applications that interact with cloud services and APIs provided by service providers. Enables machine-to-machine communication in Internet of Things (IoT) scenarios.

• SOAP API:

SOAP APIs are commonly used in enterprise-grade integration to:

Interact with complex systems such as Customer Relationship Management (CRM) platforms or enterprise resource planning (ERP) systems.

- Perform operations that require advanced security features, such as authentication, encryption, and digital signatures.

• Database API:

The database API is used to:

- Create custom applications that interact with the database to store and retrieve data.

Perform database operations in web applications, including querying, updating, and managing data. • Payment Gateway API:

Payment Gateway APIs are essential for:

- Enable secure online payments for e-commerce sites and applications.

- Integration with various payment methods, such as credit cards, digital wallets or bank transfers.

- Manage transaction-related tasks such as processing payments, managing subscriptions or generating reports.

1. **Identify the potential security issues with API and critically evaluate the suitable API for given scenario or your selected application.**
2. Identify potential security issues with API

* Injection Attack: APIs that do not properly validate and sanitize user input can be vulnerable to injection attacks, where malicious code or commands are injected into API requests or responses.
* DDoS Attacks: Distributed Denial of Service (DDoS) attacks, in which numerous requests overload an API's resources and disrupt service, can target poorly protected APIs.
* Broken Authentication: APIs that have weak or flawed authentication mechanisms can be exploited by attackers to gain unauthorized access to sensitive data or perform unauthorized actions.
* Sensitive Data Exposure: APIs that transmit or store sensitive data without proper encryption or protection mechanisms are at risk of exposing that data to unauthorized parties.
* Broken Access Control: APIs that do not enforce proper access controls and authorization checks can allow unauthorized users to access or manipulate sensitive data or perform actions they shouldn't have access to.
* Parameter Tampering: APIs that do not validate and enforce integrity checks on input parameters are susceptible to parameter tampering, where attackers modify parameters to manipulate or bypass business logic or gain unauthorized access.
* Man-in-the-Middle Attack (MITM): MITM attacks, in which an attacker intercepts and alters the data sent between the API and client, can be used against APIs that transport data over insecure networks or do not employ sufficient encryption and certificate validation.

1. Evaluate potential security issues in a suitable API of Know-Your-Neighborhood

In the scenario of the "Know-Your-Neighborhood" application, which provides information about neighborhoods, local amenities, and safety ratings, the following security issues may be relevant:

1. Injection:

The API should ensure that any user-generated input is properly validated, sanitized, and parameterized to prevent SQL, command, or code injection attacks. This is crucial to protect against attempts to manipulate or compromise the underlying database or system.

1. Broken User Authentication:

The API must have robust authentication mechanisms to prevent unauthorized access to user accounts or sensitive information. It should implement secure password storage, strong authentication protocols (e.g., multi-factor authentication), and protection against brute-force attacks.

1. Broken Access Control/Authorization:

The API should enforce proper access controls to ensure that users can only access the data and perform actions that they are authorized for. It should validate user permissions and implement role-based access control (RBAC) or other suitable authorization mechanisms.

1. Suitable API for the given scenario :

The Public API, because we will use the RESTful API and Facebook API

1. Task 2

**Apply the knowledge of API research to design an application.**

1) Analyze the given scenario, identify the requirements and select the appropriate API.

2) Develop relevant wireframes for using the API for specific purposes.

3) Identify your scope and target platform.

4) Evaluate and justify your choice of APIs for your application. (Shows security for the selected API.)

1. **Solution: .**
2. Identify what are the requirements in “Know-Your-Neighborhood” project.

• Be able to design and develop a backend using Spring Boot and JPA Framework.

• Be able to develop API using Restful Web Services.

• Be able to develop frontend application using React JS.

• Be able to identify existing APIs and its uses in already developed application

1. Select a suitable login API among 3 different login APIs in research in the project scenario.

* Facebook Login API

Users can log in using their Facebook credentials. With the proliferation of Facebook accounts among users, this allows for an efficient login process. Facebook became popular ahead of other programs, so it's easy to create an account using the Facebook API. Therefore, we believe that there are many Facebook users and that this API can be used to make a good amount of money.

* Google OAuth2

Google OAuth2 allows users to authenticate with their Google accounts. Provides secure login options. Since Google is already growing rapidly and has so many users around the world, we believe that using Google Login will make the process of logging into our system easier for our users.

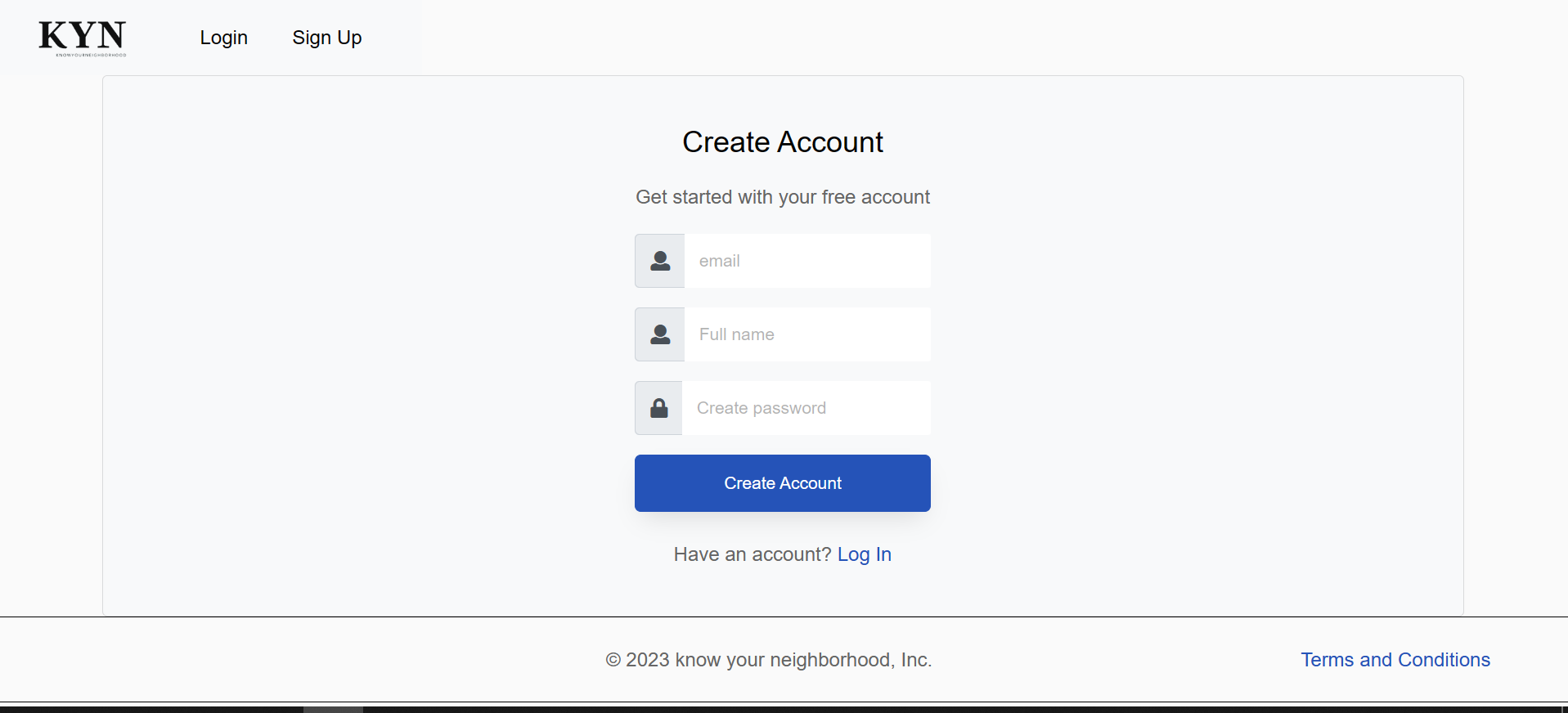
* LinkedIn

Users can authenticate with their LinkedIn account via LinkedIn's login API. This API is suitable if your project aims to reach professional and business-minded people who are likely to have his LinkedIn account. This gives you access to job profile information that is useful for specific project functions..

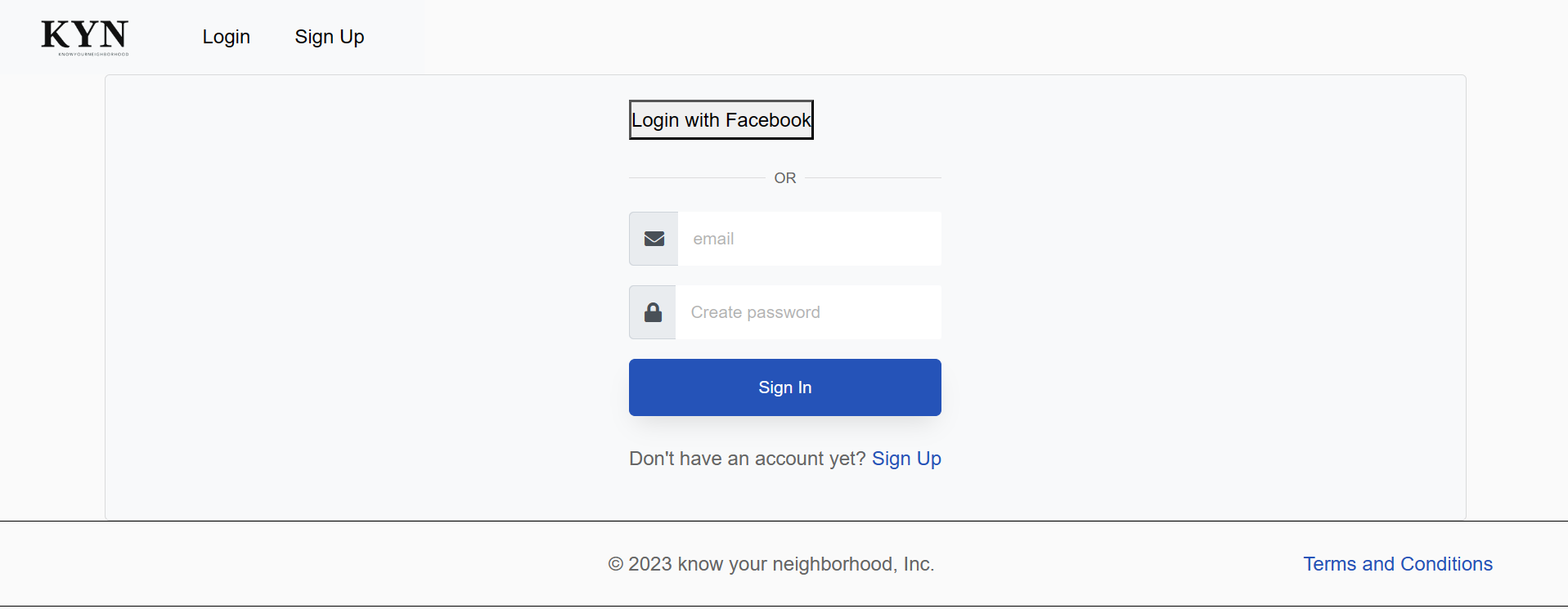
**Suitable Login API for KYN**

Using Facebook API Login for KYN Project involves integrating Facebook's authentication system, allowing users to log in with their Facebook credentials. This simplifies registration and login, as users do not need to remember any additional details. With user consent, your project will receive an access token from Facebook, allowing the project to request and use specific user information from the Facebook API, improving the user experience in the app. your use. This approach leverages Facebook's established security and user trust, but careful data management and privacy considerations are important.

1. **Develop the relevant wireframes to utilize the API for given purpose.**
2. Register Page showing Button and Form Input Register Wireframe

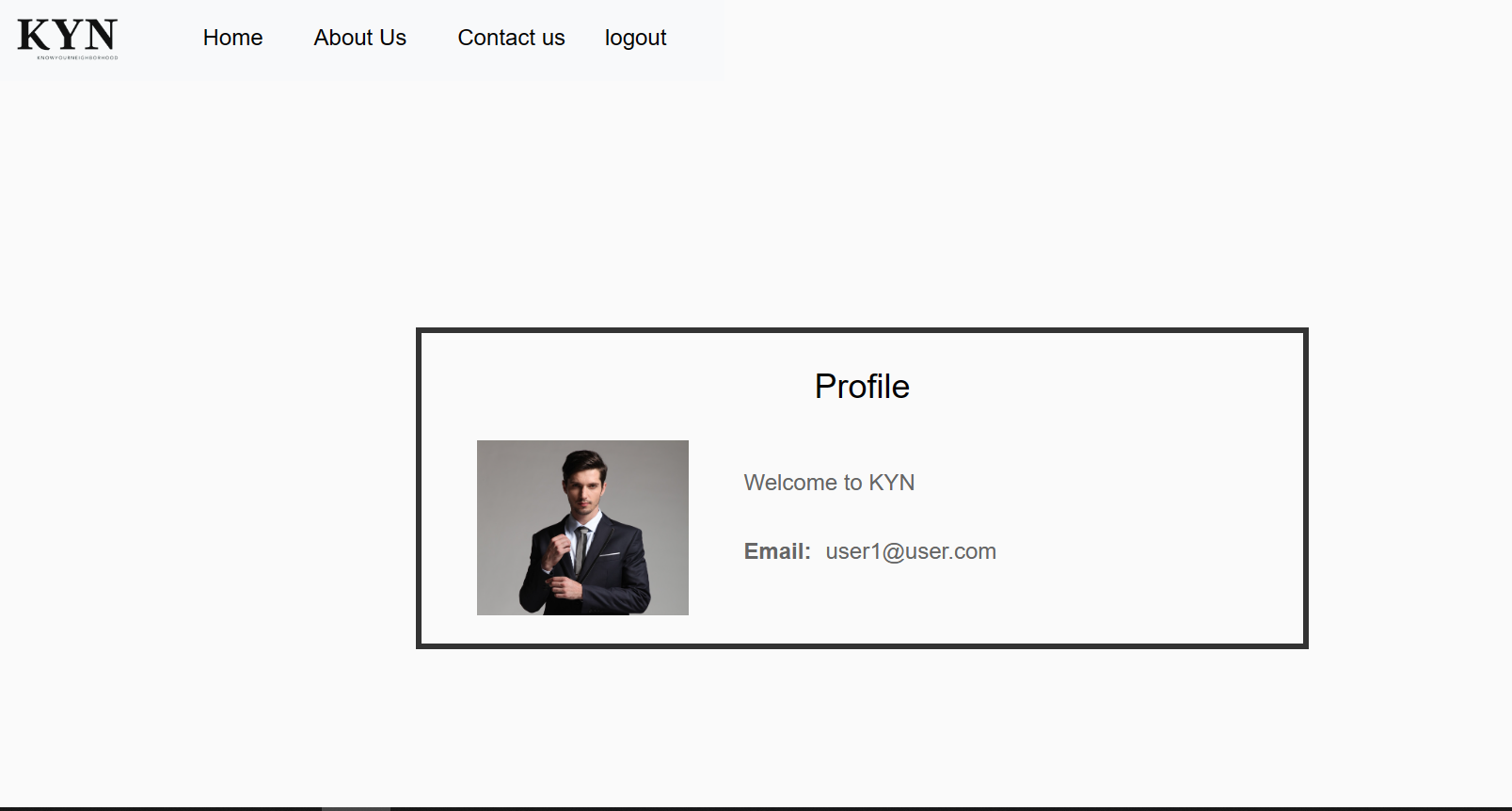


1. Login Page showing Facebook Button and Form Input Login Wireframe



After user signing using Facebook account with default scope, KYN website can access some user information like user’s facebook ID, name, and profile URL. With this information KYN website access the user to sign in to the website. After choose the account user will go to Dashboard page

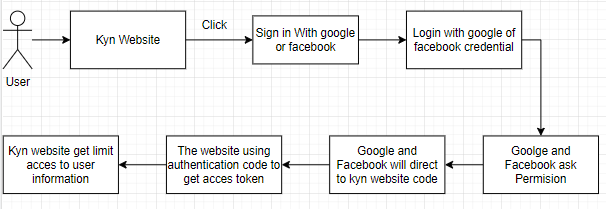
1. Your KYN dashboard Wireframe after Facebook login API

z

1. **Identify the scope and target platforms.**

Because the Facebook APIs use different languages ​​for building applications, these two APIs are now available on most devices and run on most operating systems. For example, these APIs can be used in Windows and Mac environments as well as desktop and mobile devices.

1. **Evaluate and justify the selection of chosen APIs for the application. (Show security of selected APIs.)**

****

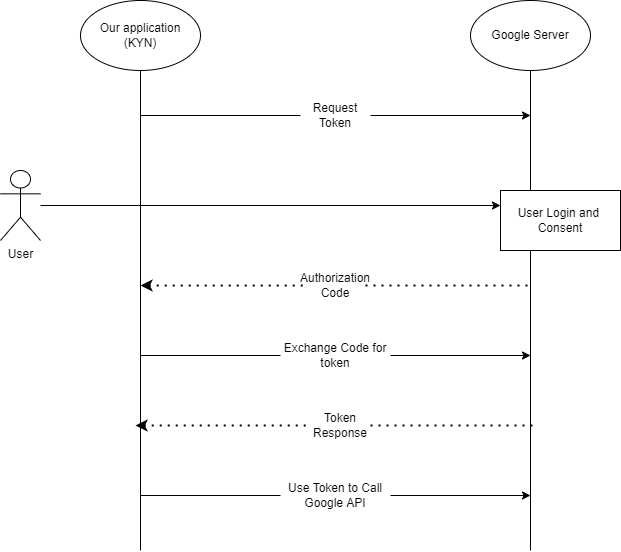
The user needs to log in/register on the KYN website. First, a user clicks the login/register with Facebook button. then the user will be directed facebook.com, where a login status check is done.

**Security of selected API for the application**

**Server-side authentication**

We may use server-side authentication to access and retain sensitive user data and features. Server-side authentication, commonly known as the OAuth authorization token flow, is designed for use by programs running on a web server. Google offers this option for most use cases as it is the most secure

way to implement OAuth.



**6. Task 3**

**Application Implementation**

1. Introduce three different types of backend, frontend, and API implementation process
2. Discuss a range of suitable development environments for front-end and back-end to develop an application
3. Develop a backend and Web service using selected development environment for given scenario
4. Develop an application that utilizes an API.
5. Construct the application which implements the selected API in Task 2.

**Solution:**

1. **Introduce three different types of backend, frontend, and API implementation process**
2. Three different types of Backend

* Express JS

A popular lightweight web application framework for Node.js is called Express.js. It provides an easy way to build APIs and web applications. The simplicity and customization of Express.js allows developers to effectively manage routes, middleware, and HTTP requests. Suitable for building scalable and fast server-side applications.

* Spring Boot

A Java-based framework called Spring Boot makes it easy to create enterprise-grade standalone Spring apps. It provides a convention-less-configuration approach that allows developers to easily set up and configure their applications with little boilerplate code. Spring Boot is known not only for providing a complete environment, but also for its reliability, extensibility, and extensive community support.

* Laravel
* The Model-View-Controller (MVC) architectural paradigm is used by the PHP-based web application framework Laravel. With features like routing, ORM, authentication, and caching, it is simple to construct online applications. It also offers an elegant and expressive syntax. Laravel is a well-liked option for PHP developers since it places an emphasis on developer productivity, maintainability, and simple code.

1. Three different of frontend

* React JS

A JavaScript package called React.js is used to create user interfaces. It allows developers to make reusable UI components and adheres to a component-based architecture. React.js effectively renders and updates components based on data changes, resulting in a fluid and engaging user experience. Mobile and single-page applications (SPAs) are frequently built using it.

* Angular JS

Google maintains the feature-rich JavaScript framework known as Angular.js. It offers a fully functional development environment for creating web apps. Angular.js uses declarative templates, two-way data binding, and dependency injection to build scalable and maintainable systems. It can provide comprehensive testing and features like routing and form validation.

* Vue JS

A progressive JavaScript framework for creating user interfaces is called vue.js. It is renowned for being straightforward, adaptable, and simple to integrate into current projects. Vue.js is appropriate for small and large-scale projects since it allows developers to absorb its features gradually. It features a component-based architecture and a reactive data-binding framework.

1. Three different of API

* REST API

A RESTful API interface allows two computer systems to communicate securely over the Internet. Most business applications need to connect to other internal and external applications to perform various activities. For example, to automate billing and connect to an internal time tracking application, your internal accounting system needs to exchange information with your customer's banking system to generate monthly payslips. RESTful APIs support this information exchange by adhering to software communication standards that are secure, reliable, and effective.

* SOAP API

SOAP, or Simple Object Access Protocol, is a standard web communications protocol developed by Microsoft in 1998 and is an important part of Service Oriented Architecture (SOA). SOAP is used to expose web services and can integrate various communication protocols using the HTTP protocol. SOAP uses Extensible Markup Language (XML) to send messages and is a complete and official protocol with built-in security features and strict usage rules. While SOAP has clear advantages of security and standards, it is also a complex protocol and requires more bandwidth than some alternatives.

* RPC API

Remote Procedure Call (RPC) builds distributed client-server based systems. Also called function call, subprogram call. The called process does not have to be in the exact same address space as the calling procedure, as it is based on traditional local procedure calls. RPC is ideal for client-server interactions where control flow exists between the two. Threads of execution are staggered on both the client and server rather than simultaneously.

1. **Discuss a range of suitable development environments for front-end and back-end to develop an application**

Front-end development environments:

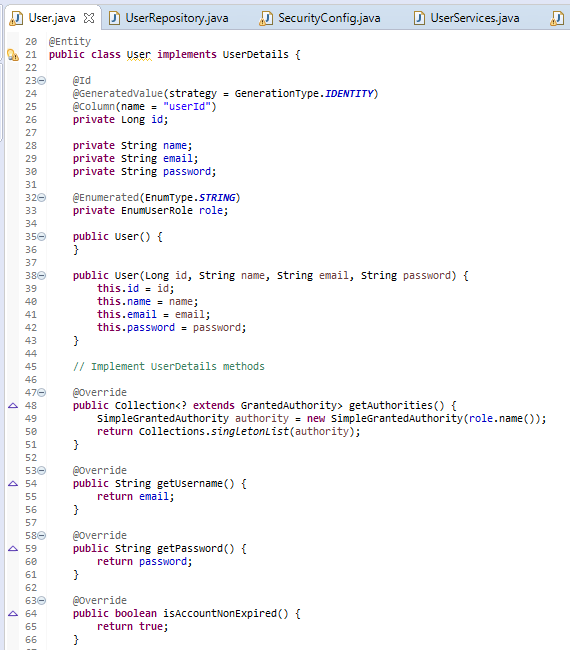
* Visual Studio Code: a popular open-source code editor with built-in debugging that supports Git and several programming languages.
* Sublime Text: a powerful, portable text editor with an easy-to-use interface, a wide selection of plugins, and customizability possibilities.
* WebStorm: an effective integrated development environment (IDE) for front-end languages like TypeScript and JavaScript.

Back-end development environments:

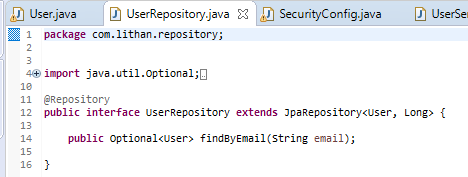
* Visual Studio: Microsoft developed an IDE that is well-known, packed with features, and supports a wide range of programming languages and frameworks.
* Eclipse: is an open-source, widely used IDE supporting numerous programming languages and frameworks.
* IntelliJ IDEA: an established integrated development environment (IDE) for Java development that fully features and supports a wide range of additional programming languages and frameworks.

1. **Develop a backend and Web service using selected development environment for given scenario**
2. **Back-end login/register**

**a) User.java**



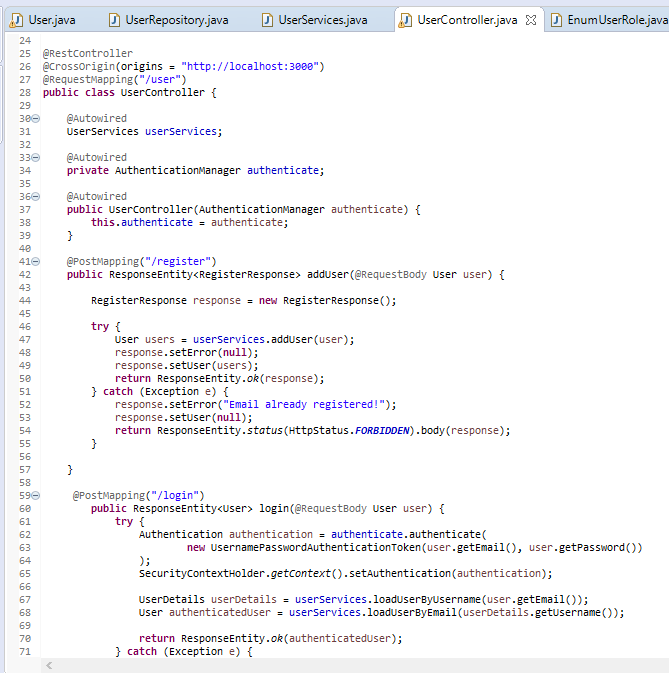
**b) UserRepository.java**



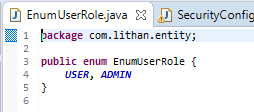
1. **UserServices** .java



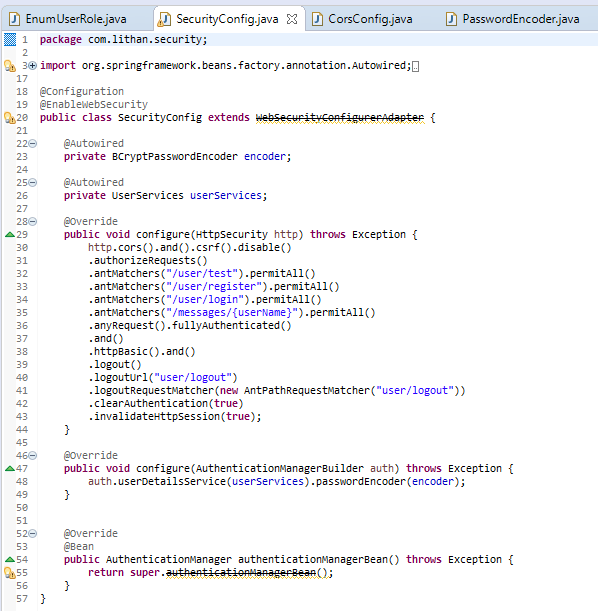
**d) UserControler.java**



**e) EnumUserRole.java**



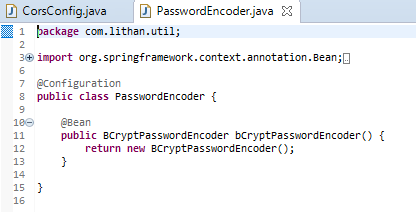
**F ) SecurityConfig.java**



**g) CorsConfig.java**



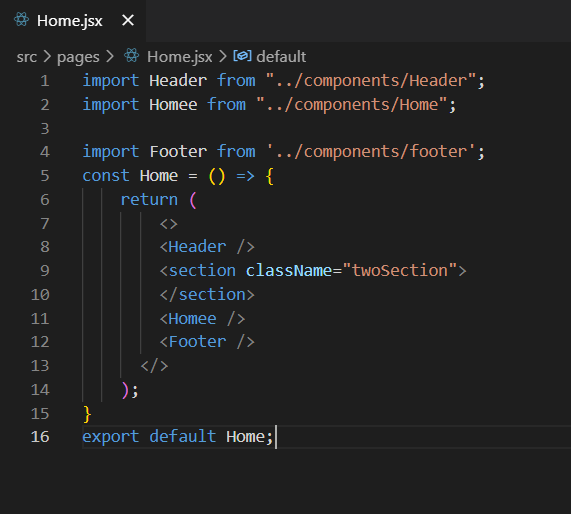
**h) PasswordEncoder.java**



**4) Develop an application that utilizes an API.**

**Develop front-end using react**

* **Home.jsx**



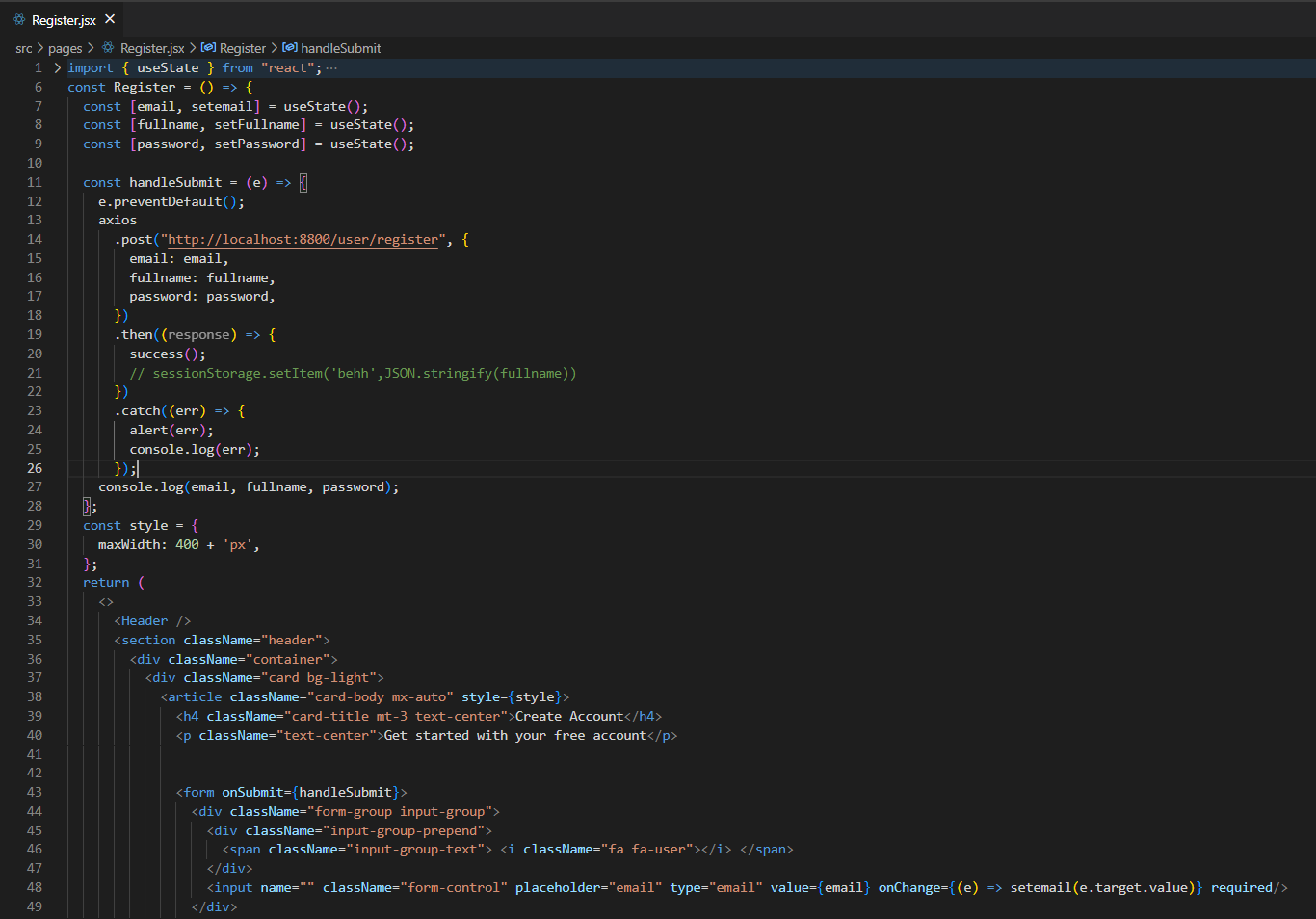
* **Login.jsx**

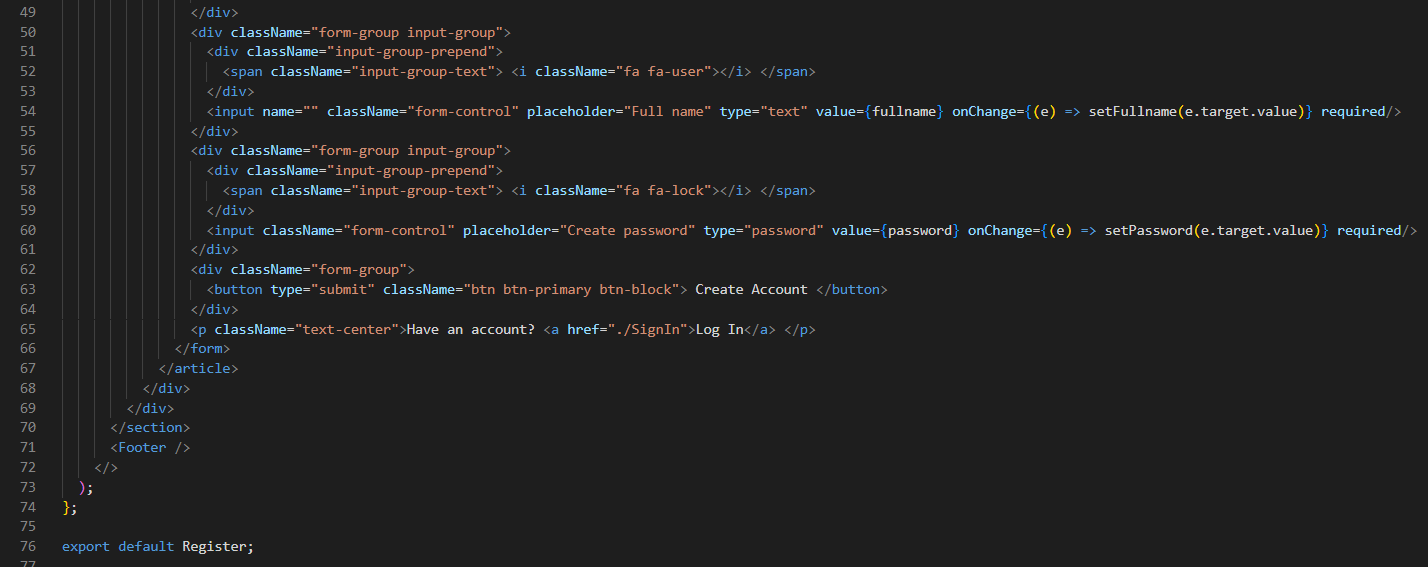






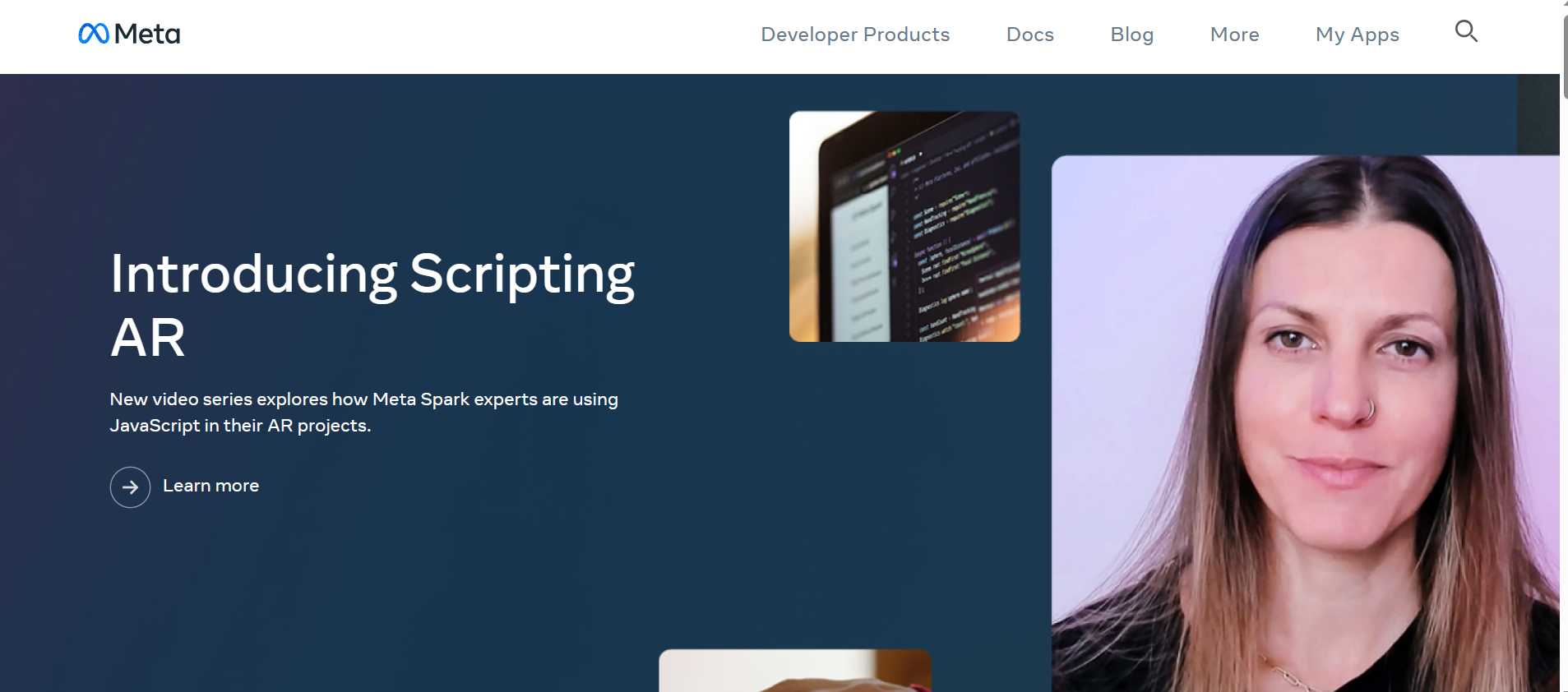
* **Register.jsx**



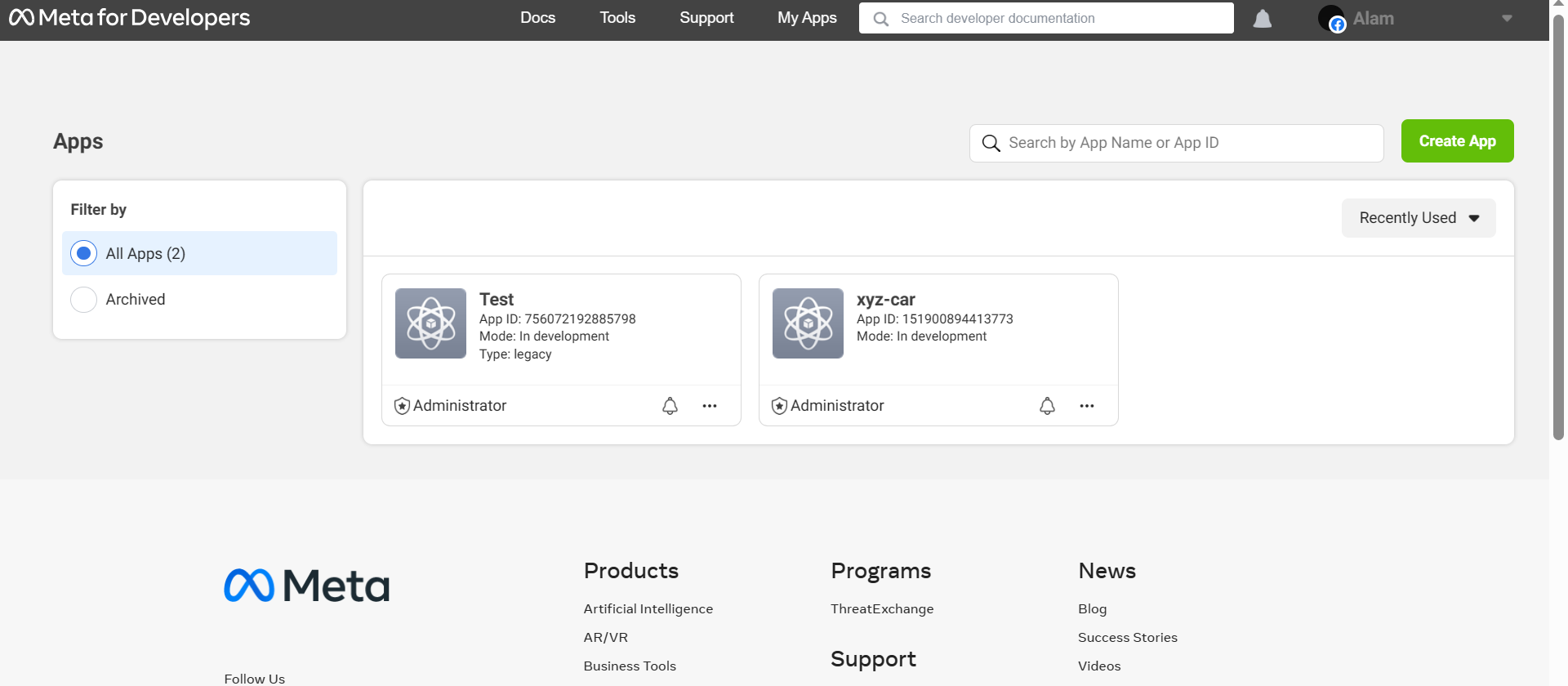


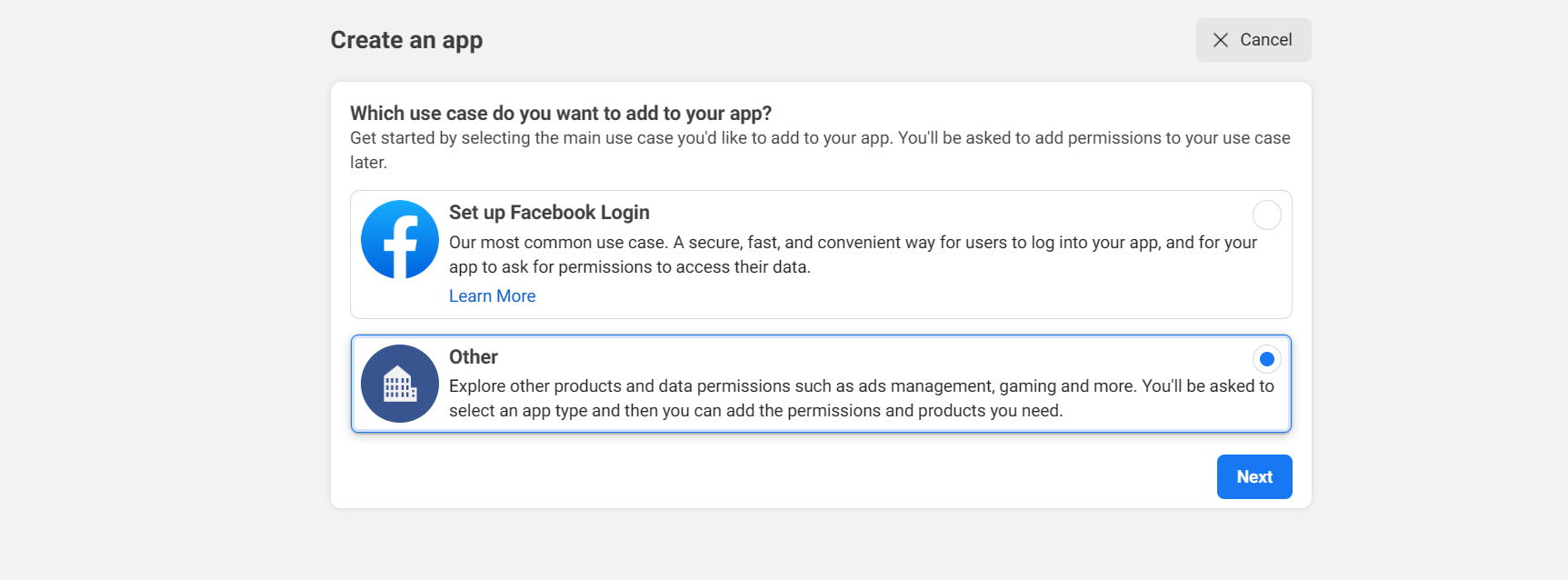
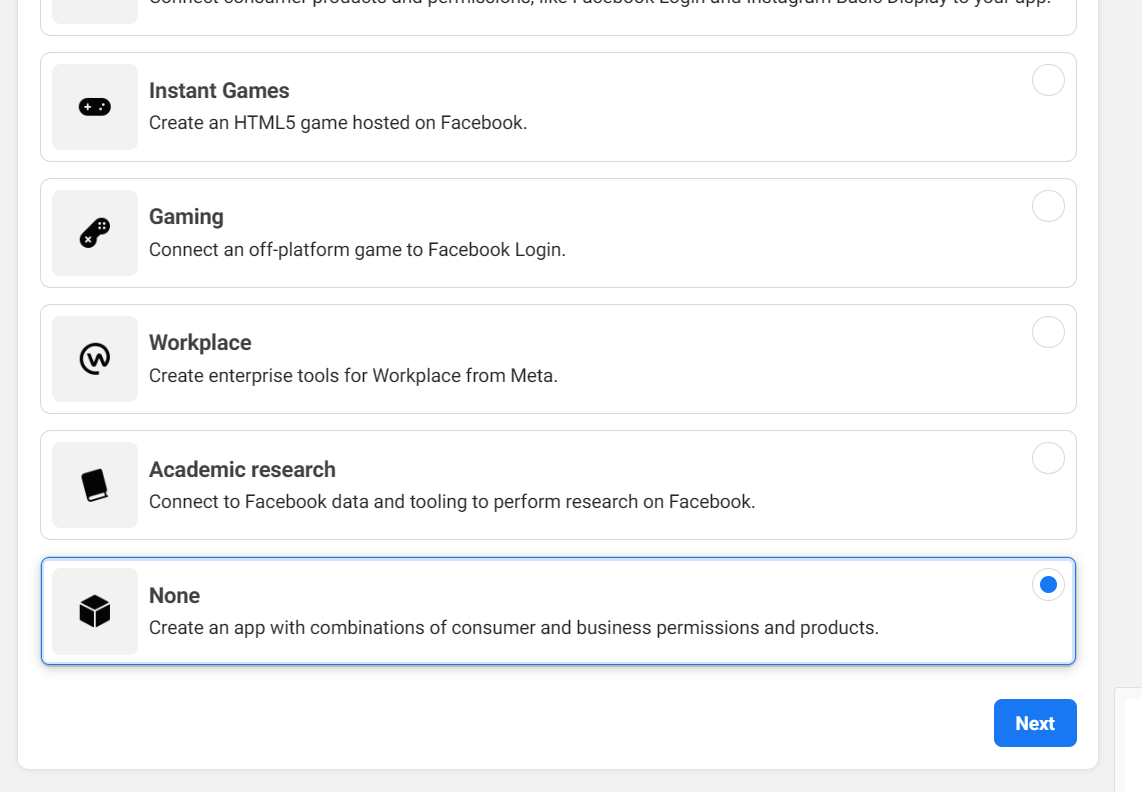
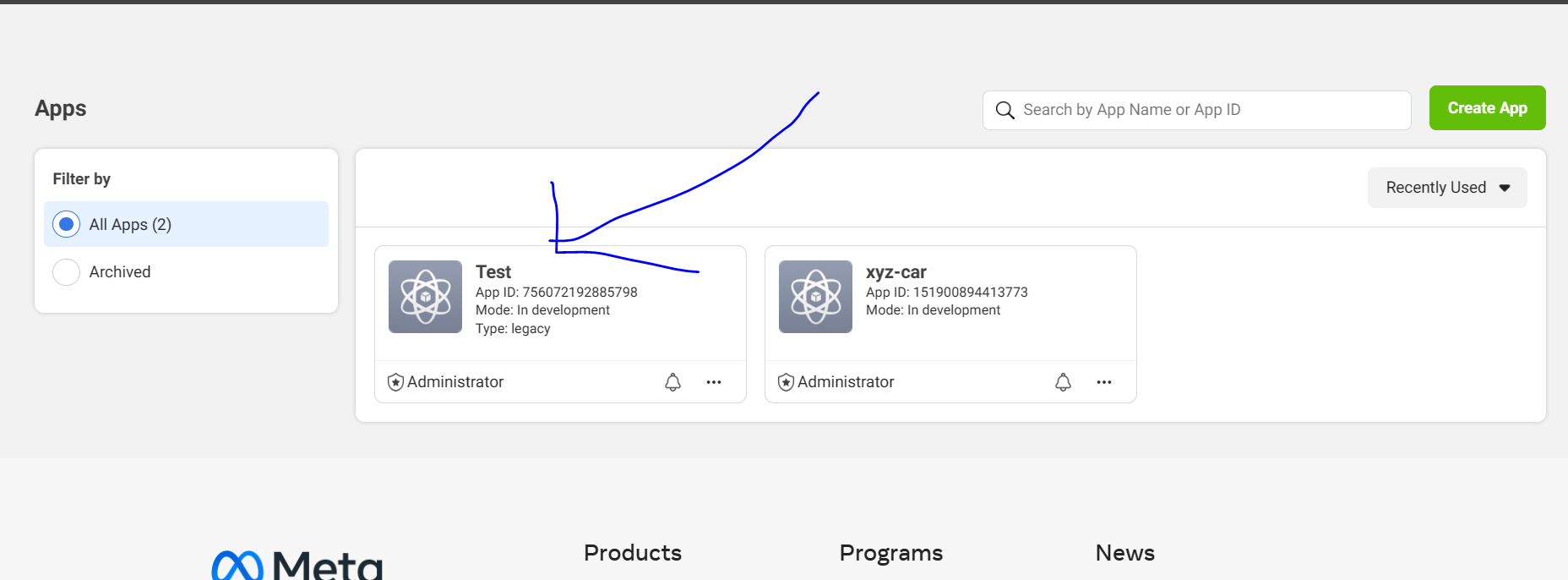
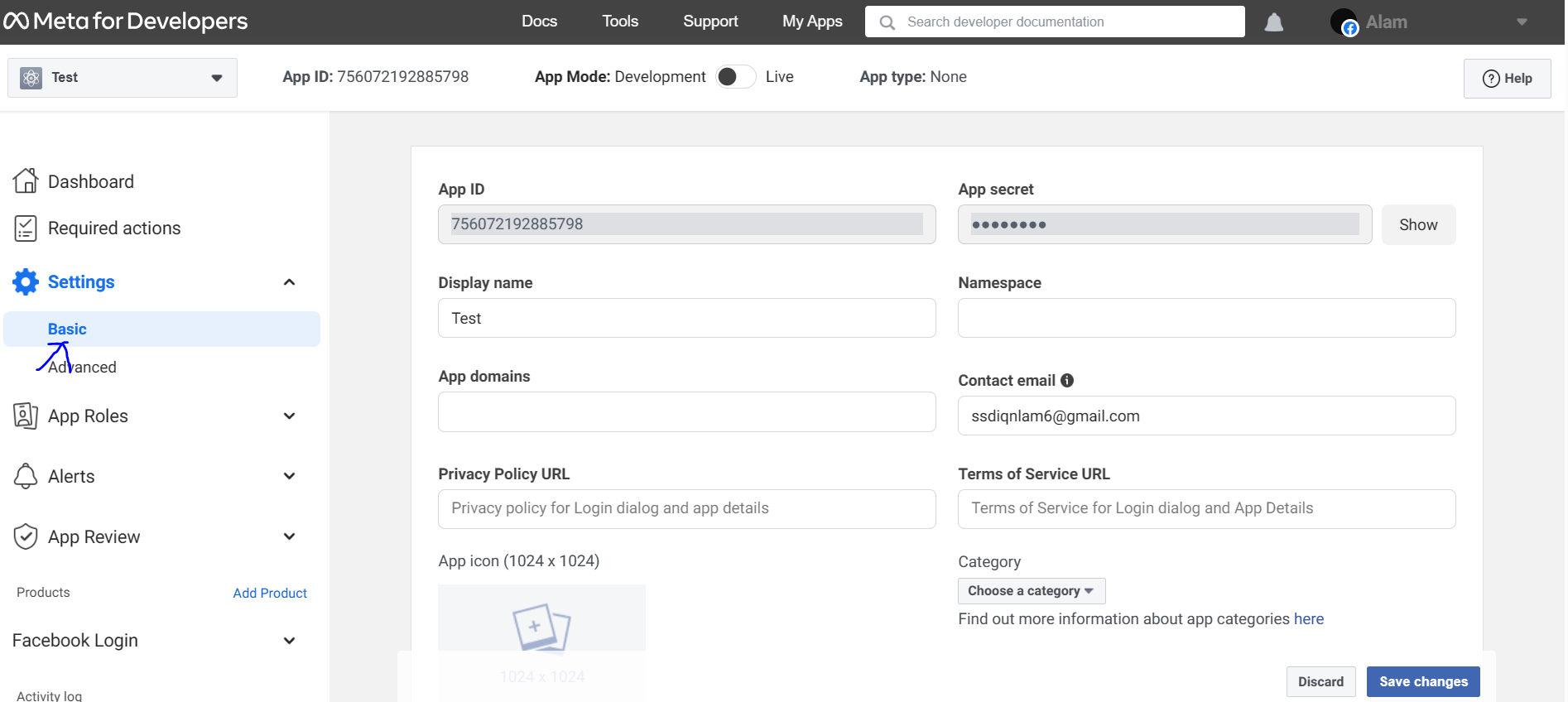
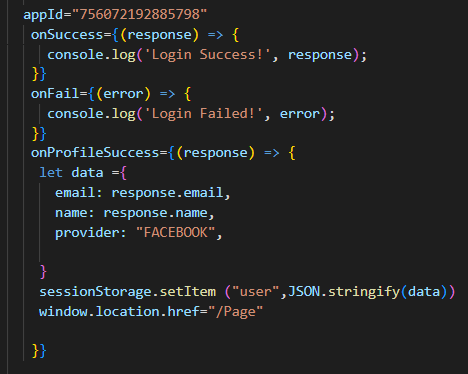
**5) Construct the application which implements the selected API in Task 2.**

1. **How to Login/Register with Facebook API**
   * Firstly, open meta for developer in Facebook then login into your account



* + create an app by clicking “create app” below



* + Choose Facebook Login  
    
  + Choice None  
    
  + Name the app  
    
  + Open Your App  
    
  + Go to setting and basic, and you can see your APP Id  
    
  + Then copy and paste “App ID” and to the code editor in your facebook login file  
    

7. Task 4

**Create the following items under “Application Testing” in Project Report**

1. Implement white Box testing for the developed API of your Application
2. Conduct Black Box testing (UAT testing) of your developed application and show the evidence for each test case.
3. Once the testing done check failed test cases and the reason to fail the same and implement your application accordingly.

**Solution:**

1. **Implement white Box testing for the developed API of you Application**

* **What is white Box testing?**

White box testing is a software testing technique that examines the internal structure and implementation details of a system. Code-savvy testers design test cases to ensure full coverage and detect errors. This improves code quality, detects problems earlier, and optimizes performance. However, this requires programming knowledge, may not fully capture the user's perspective, and may not include all scenarios or external dependencies. Overall, white box testing ensures internal correctness and improves software quality.

* **Why we using white box testing?**

White box testing is used for comprehensive testing and verification of the internal structure of the software system and code implementation. This helps ensure thorough testing, early error detection, and improved code quality. Code-savvy testers can design test cases to cover all code paths, identify coding errors, and optimize performance. By examining the internal structure, white box testing improves the reliability and efficiency of the software system.

**White box testing is used to:**

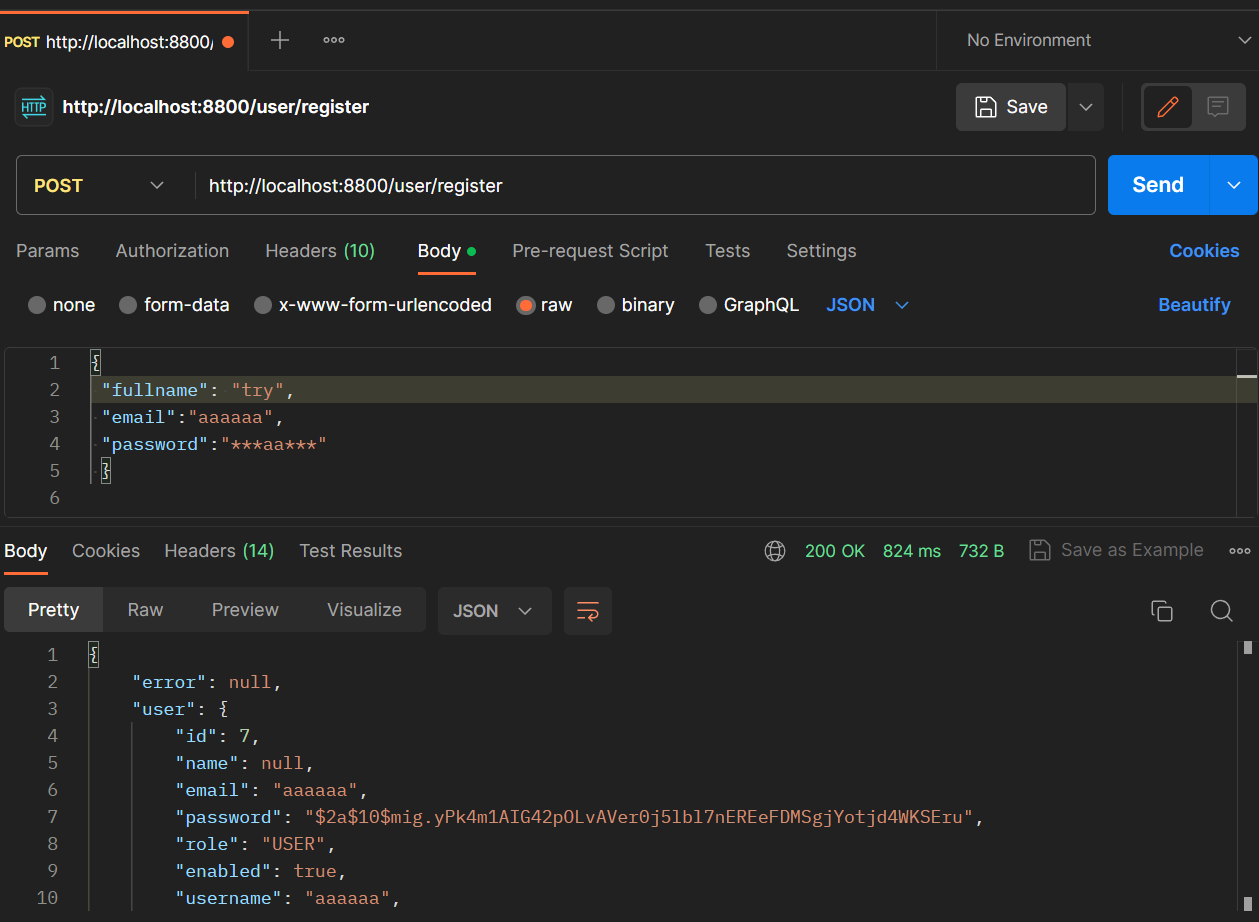
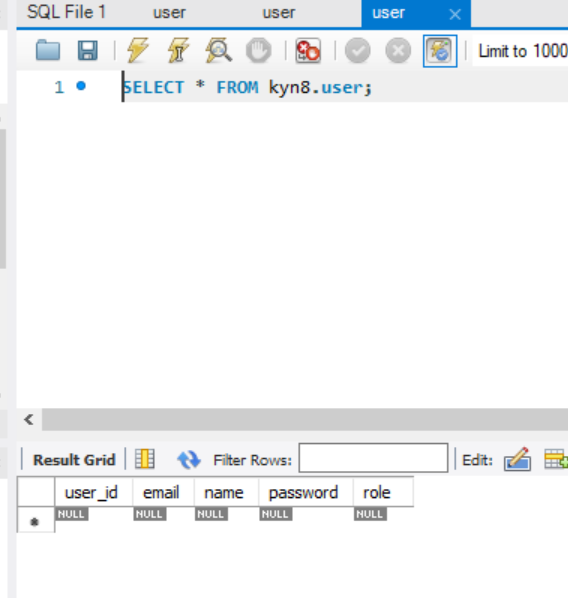
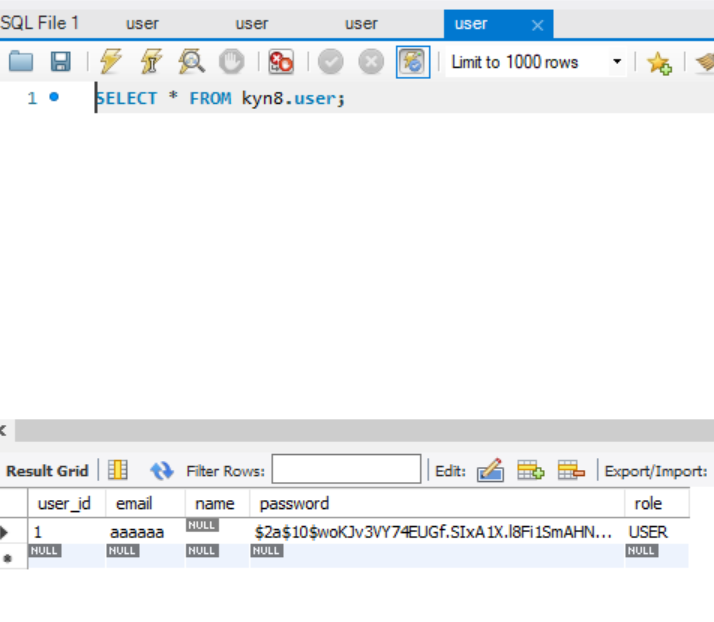
• Ensure that each independent path within a module is executed at least once.

• Discover logic or design errors that may occur during the actual implementation.

• Identify security holes or vulnerabilities in your code.

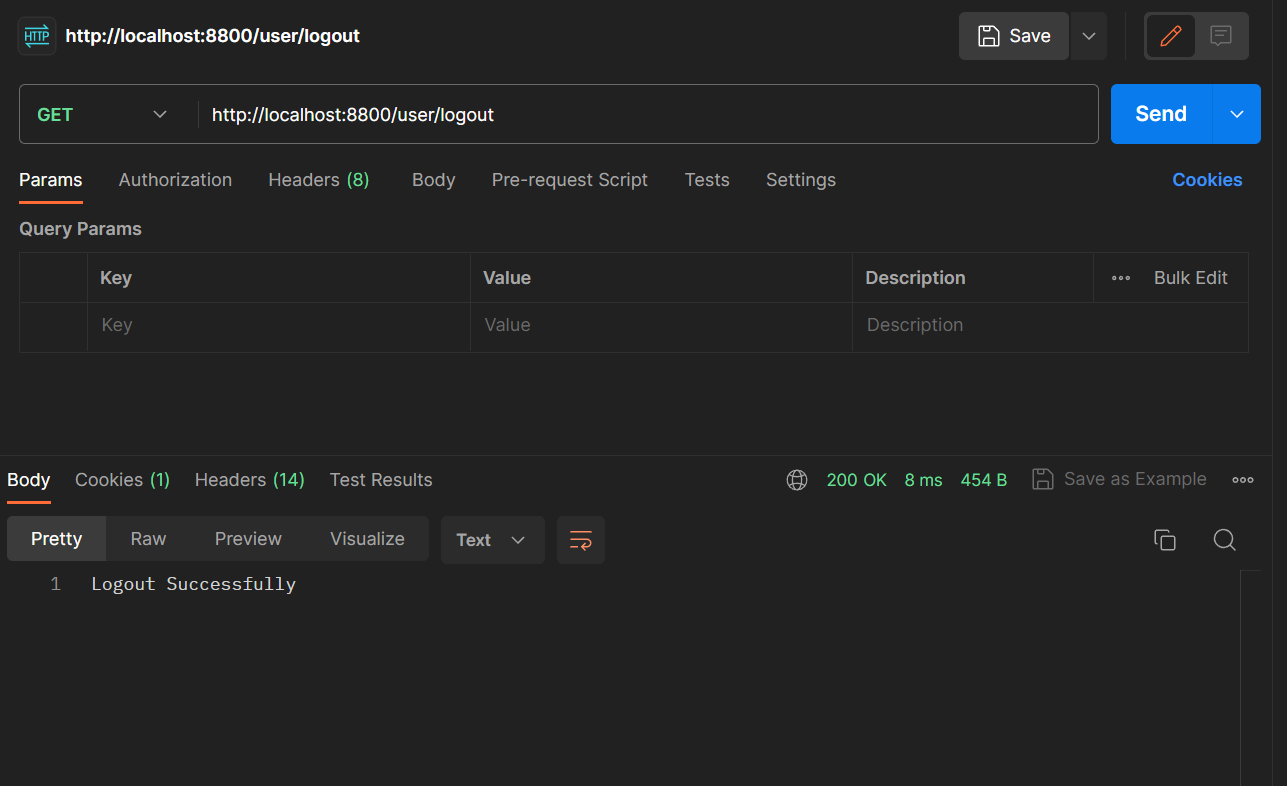
• Check for typos and grammatical errors.

* **Test all controller mappings using POSTMAN tool and then record your result**
* Register

  
Database Before Register  
  
Database After Register  


* Login



* Logout  
  

1. **Conduct Black Box testing (UAT testing) of your developed application and show the evidence for each test case.**

* What is black box testing?

Black box testing, also known as Behavioral Testing, is a test carried out to observe the input and output results of the software without knowing the code structure of the software. This test is carried out at the end of making the software to find out whether the software can function properly.

* Benefit black box testing?

a. Testers do not have to know a programming language.

b. Testing is carried out based on the user's point of view. This is done to find inconsistencies in the software.

c. Developers and testers depend on each other.

D. Testers don't need to check the code.

e. Allows testers and developers to work independently without interfering with each other's work processes.

* UAT testing

|  |  |
| --- | --- |
| **Test Scenario** | Registration |
| **Test Cases** | Registration in KYN Website |
| TC001 | User input forms and not fill one forms and it should show danger alert |
| TC002 | User input data into forms and it should show success alert |

* Test Data Table

|  |  |  |
| --- | --- | --- |
| **Test Scenario ID** | **Test Case ID** | **Test Data** |
| TS001 | TC001 | fullname = Syukur Sidiq  Email = syukur@email.com  Password = (not inputed) |
| TC002 | Name = Syukur Sidiq  Email = syukur@email.com  Password = mypassword |

* Test Evidence

|  |  |  |
| --- | --- | --- |
| **Test Scenario ID** | **Test Case ID** | **Test Evidences** |
| TS001 | TC001 |  |
| TC002 |  |

|  |  |
| --- | --- |
| **Test Scenario** | Login |
| TS001 |
| **Test Cases** | Login in KYN Website |
| TC001 | Login with account that have registered |
| TC002 | Login with facebook |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Case ID** | **Test Priority** | **Pre-conditions** | **Expected Result** | **Actual Result** | **Final Result** |
| TS001 | TC001 | High | - User visit KYN Website  -User open login page | User input email and password to login into website, user should be redirect to profile page | As expected | Pass |
| TC002 | High | User click login with facebook button into website, user should be redirect to profile page | As expected | Pass |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | **Test Case ID** | **Test Priority** | **Pre-conditions** | **Expected Result** | **Actual Result** | **Final Result** |
| TS001 | TC001 | High | - User visit KYN Website  -User open login page | User input email and password to login into website, user should be redirect to profile page | As expected | Pass |
| TC002 | High | User click login with facebook button into website, user should be redirect to profile page | As expected | Pass |

|  |  |  |
| --- | --- | --- |
| **Test Scenario ID** | **Test Case ID** | **Test Evidences** |
| TS001 | TC001 |  |
| TC002 |  |

1. Task 5

**Create the following items under “Review and Reflect on the APIs Used” in Project Report**

1. Review your developed API, identify the strength and weaknesses of API.
2. Provide data security report of your developed application.

**Solution:**

1. **Review your developed API, identify the strength and weaknesses of API.**

* The application development process
* Strengths and Weakness selected API, features
* Strengths REST API

1. It can be used by various programming languages, including the various platforms it uses.
2. Simpler, especially when compared to using SOAP
3. Easier to learn
4. Like the web, which always uses HTTP in every part of it.
5. Android applications that use the Rest API are much faster than web view-based Android applications.

* Weakness REST API

1. Usually longer access times compared to native libraries
2. More vulnerable to security attacks because they have to pass the HTTP protocol
3. **Provide data security report of your developed application.**

In developed applications, data security is very important to maintain the confidentiality and integrity of user information. Therefore, we have implemented Spring Security as a security mechanism in the backend of KYN applications.

Our Spring Security Configuration defines several steps to ensure effective security. First, we disabled CSRF to prevent potentially damaging multisite request forgery attacks. In addition, we allow access to certain URLs that do not require authentication, such as "/user/test", "/user/register", "/user/login" and "/messages/ {userName}". This allows users to perform actions such as registration, login, and mail access without having to authenticate first.

However, to protect more sensitive data, we require users to fully authenticate all other URLs. We use basic HTTP authentication to ensure reliable authentication security. In addition, we have configured a URL for the logout process so that users can safely exit the application and delete their authentication session.

To secure user passwords, we use BCryptPasswordEncoder provided by Spring Security. This allows us to encrypt passwords before storing them in the database, maintaining security and preventing unauthorized access.

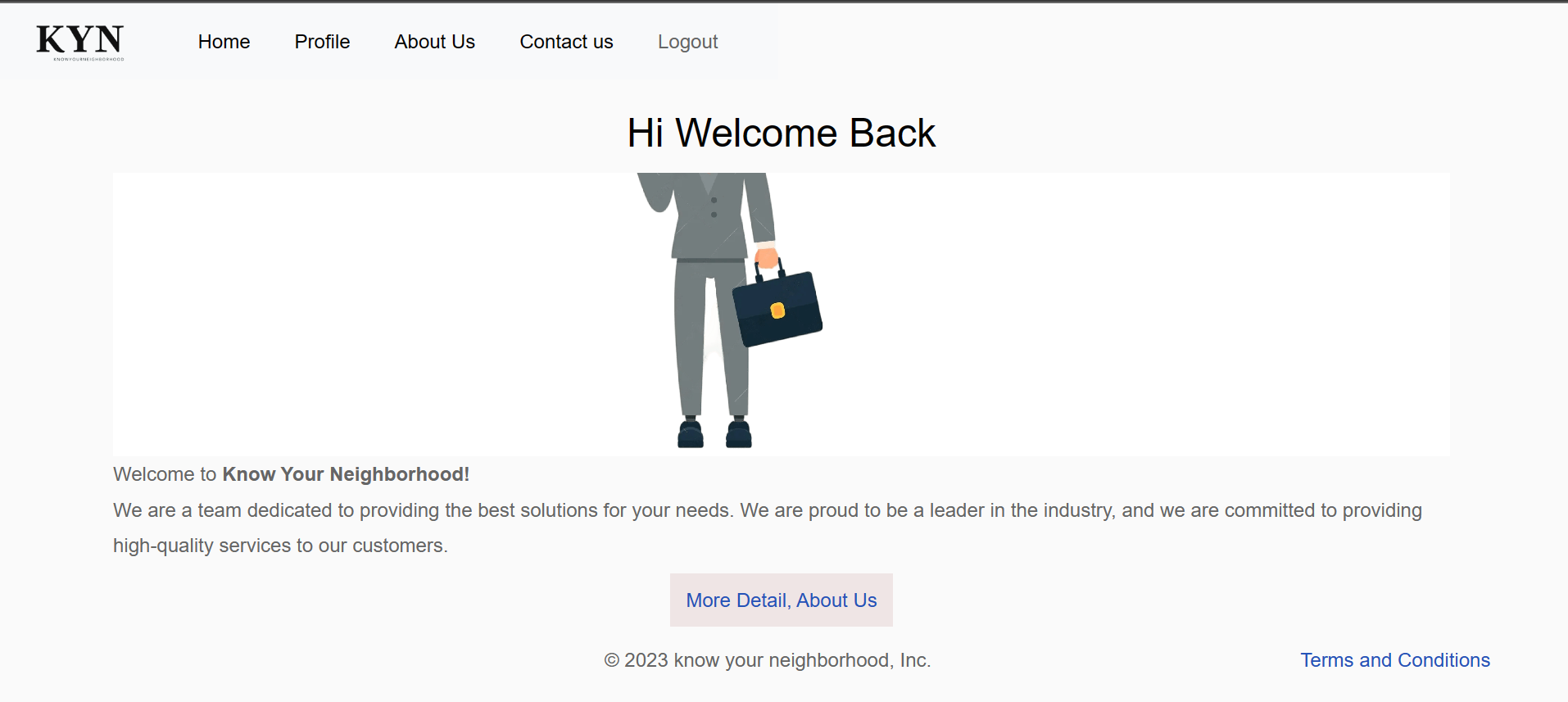
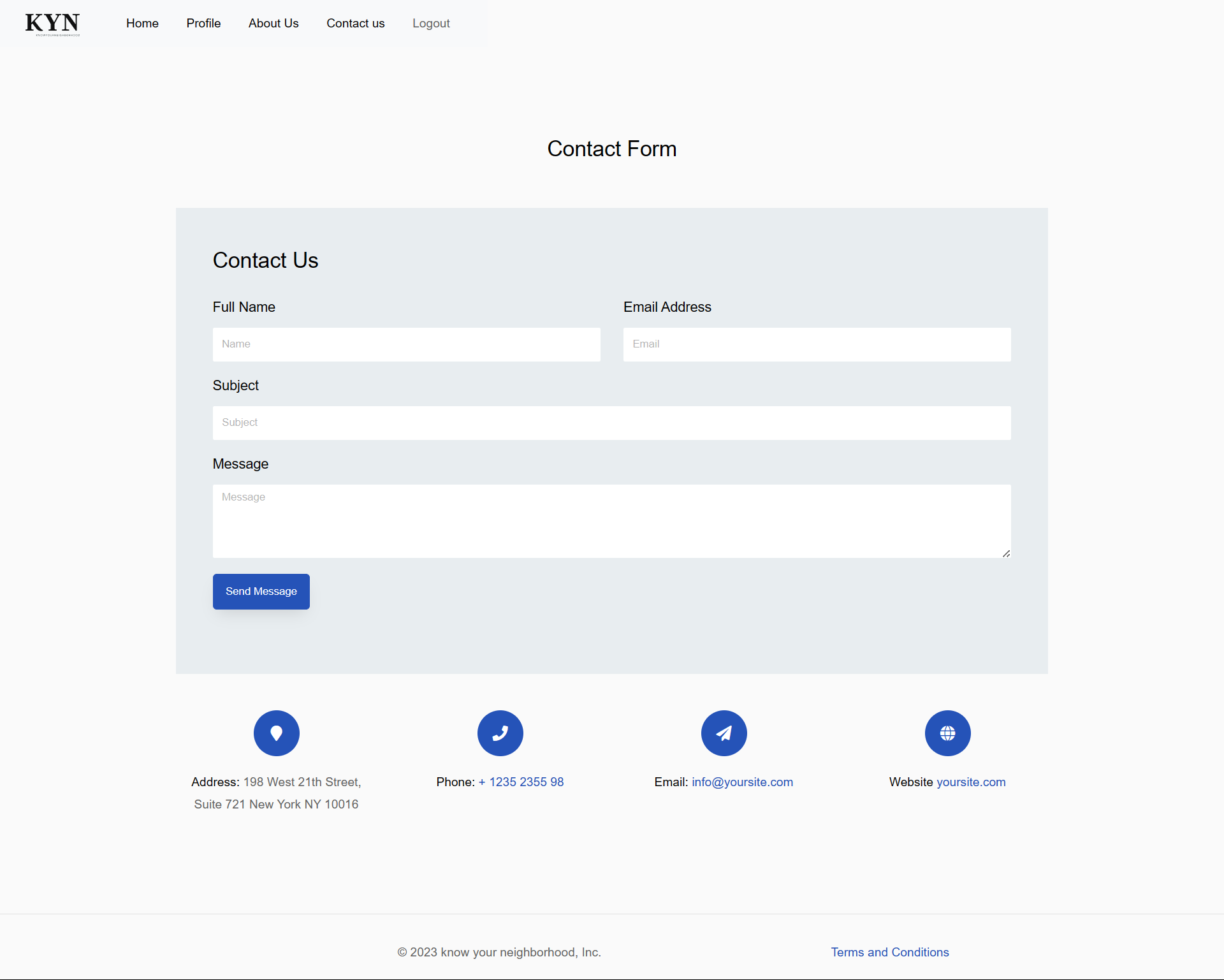
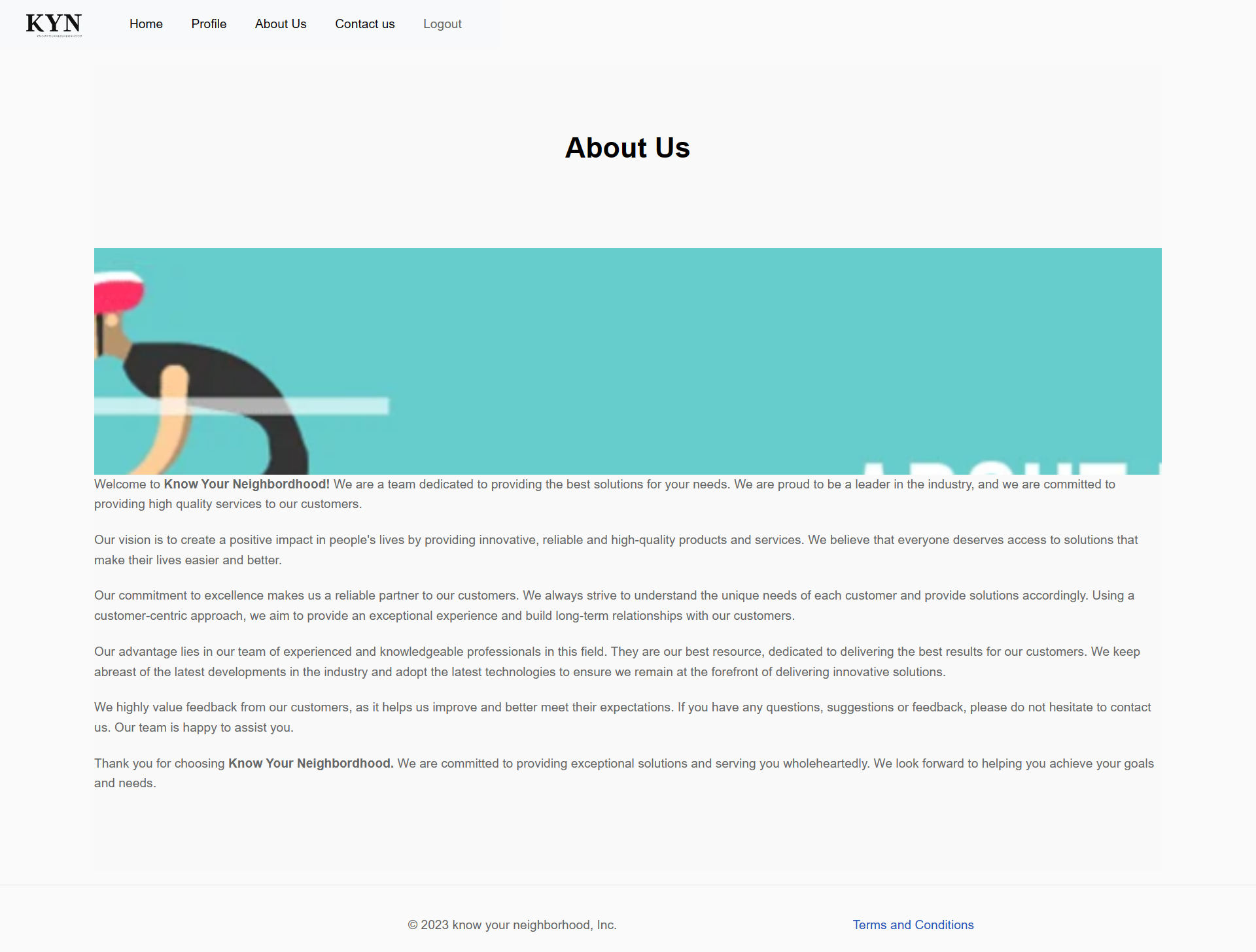
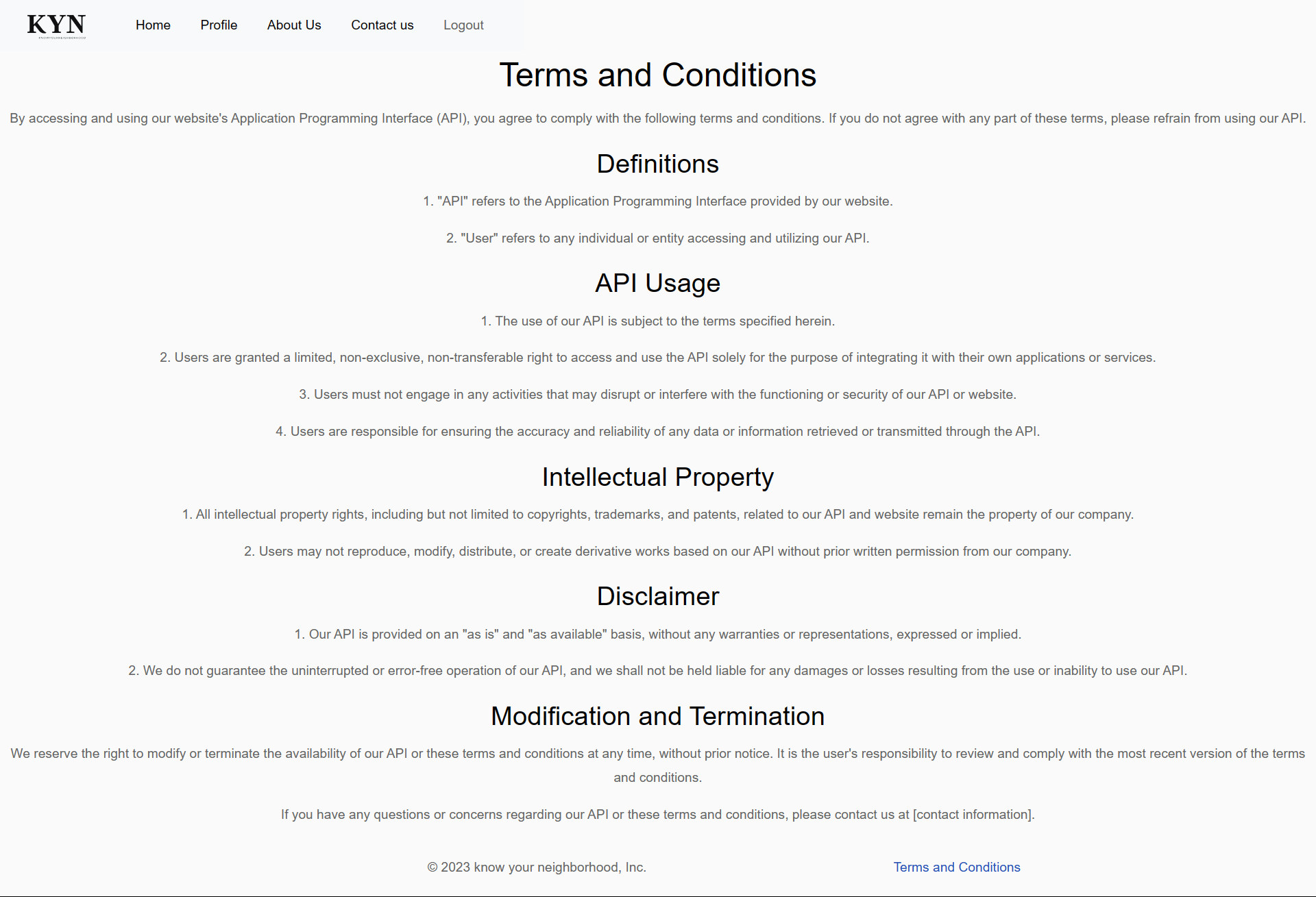
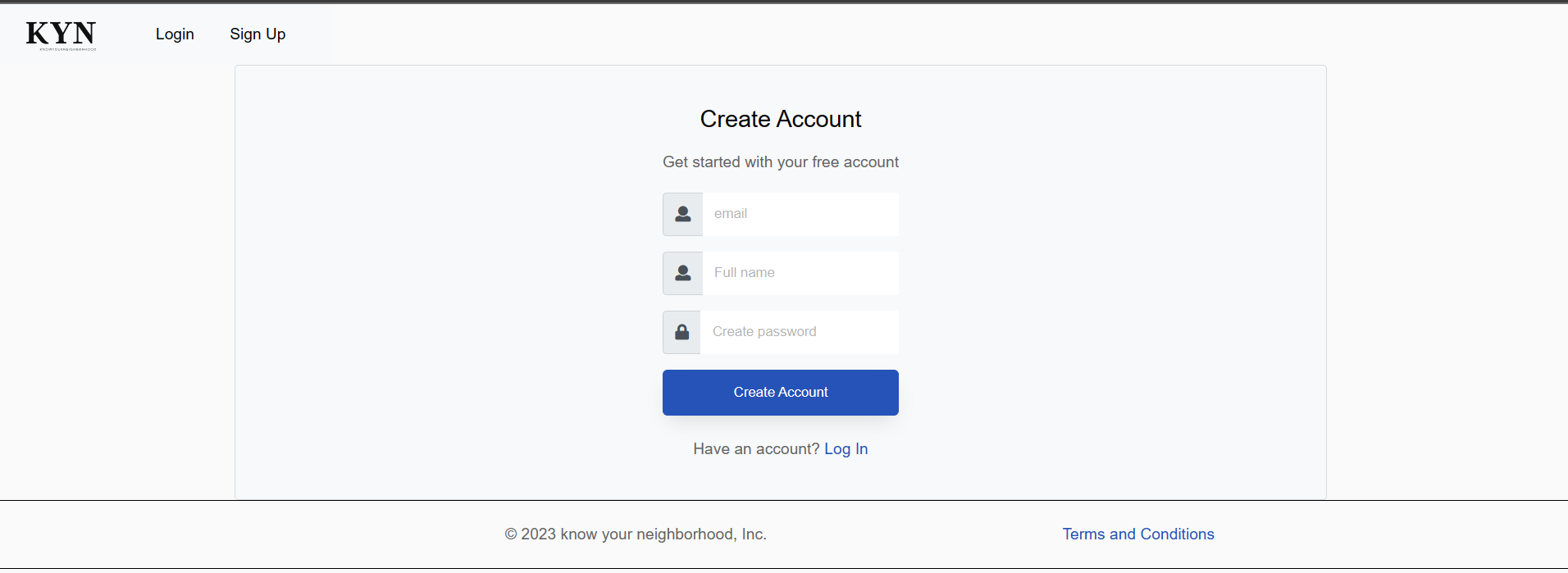
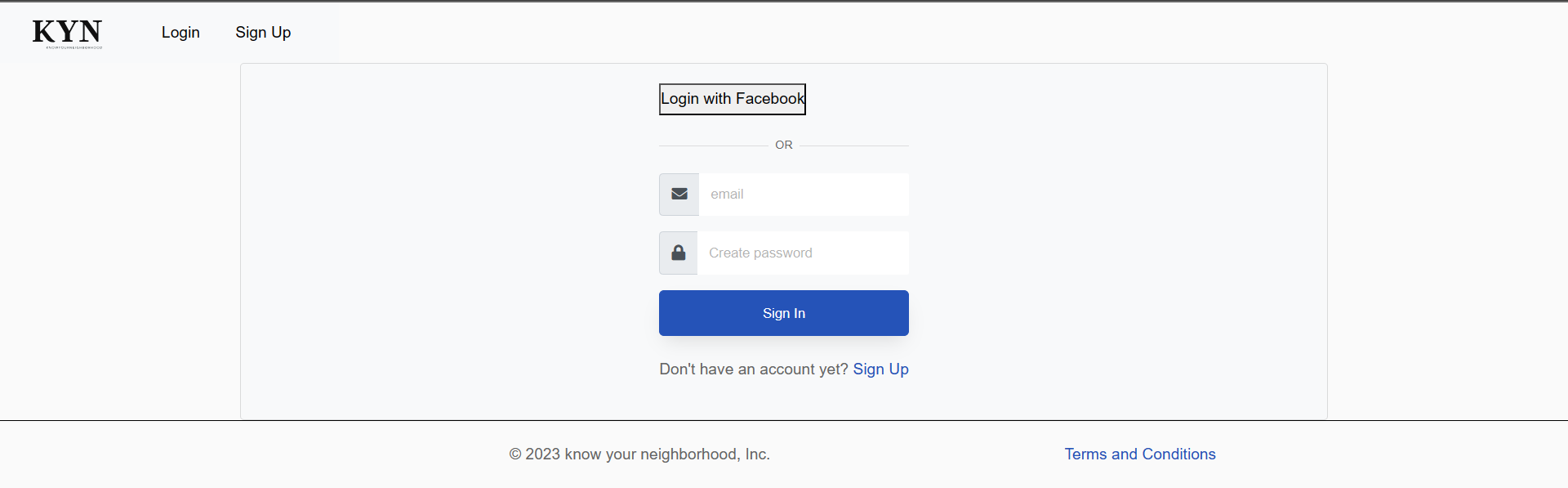
In addition, we have implemented UserDetailsService which is used by Spring Security to retrieve user information from the database. As a result, we can efficiently and securely manage and verify user credentials.

In addition to backend security, we manage CORS (Cross-Original Resource Sharing) policies using CorsConfig configuration. We only restrict access from predefined origin i.e. "http:

//local server:

3000". In addition, we only allow certain HTTP methods, such as GET, POST, PUT, and DELETE. This helps prevent potentially harmful cross-domain attacks and maintains data integrity.

By implementing Spring Security, Authentication and proper CORS configuration, we have ensured that KYN applications are secure. The intelligent and scalable authentication and authorization features provided by Spring Security effectively protect user data and provide a secure user experience.

1. Task 6
   1. Home Page  
      
   2. Contact Page  
      
   3. About Page  
      
   4. Terms and Condition Page  
      
   5. Register Page  
      
   6. Login Page  
      

Profile Page  
