8/13/2024

Local Authority Council

Website

Contents

[Introduction 2](#_Toc173681460)

[Purpose 2](#_Toc173681461)

[Requirement Analysis 3](#_Toc173681462)

[Functional Requirements 3](#_Toc173681463)

[Non-Functional Requirements 4](#_Toc173681464)

[Scope 4](#_Toc173681465)

[System Architecture and Design 5](#_Toc173681466)

[Front-End Architecture 5](#_Toc173681467)

[Back-End Architecture 5](#_Toc173681468)

[Deployment Architecture 6](#_Toc173681469)

[UML Representation 6](#_Toc173681470)

[UML DIAGRAM 6](#_Toc173681471)

[Use Case Diagram: 6](#_Toc173681472)

[Class Diagram: 7](#_Toc173681473)

[Sequence Diagram: 8](#_Toc173681474)

[Activity Diagram: 8](#_Toc173681475)

[Configuration, Deployment and Testing on localhost environment: 10](#_Toc173681476)

[Install XAMPP 10](#_Toc173681477)

[Install Composer 10](#_Toc173681478)

[Install Visual Studio Code (VS Code) 11](#_Toc173681479)

[Install Laravel 11](#_Toc173681480)

[**Your environment is ready now to import and run project.** 11](#_Toc173681481)

[Functional and Non-Functional Test Cases for Local Council Authority Website 17](#_Toc173681482)

[Functional Test Cases 17](#_Toc173681483)

[User Registration 17](#_Toc173681484)

[User Login 17](#_Toc173681485)

[File and Pay Council Taxes 17](#_Toc173681486)

[Apply for Benefits 17](#_Toc173681487)

[Search for Property Listings 18](#_Toc173681488)

[Submit a Complaint 18](#_Toc173681489)

[Non-Functional Test Cases 18](#_Toc173681490)

[Performance Testing (Unable to test as Website is locally hosted) 18](#_Toc173681491)

[Security Testing Unable to test as Website is locally hosted 18](#_Toc173681492)

[Accessibility Testing Unable to test as Website is locally hosted 19](#_Toc173681493)

[Usability Testing 19](#_Toc173681494)

[Compatibility Testing Unable to test on tablet and phone as website is currently host on localhost server 19](#_Toc173681495)

[Load Testing Unable to load test as Website is locally hosted 19](#_Toc173681496)

[Conclusion 20](#_Toc173681497)

[Use of Advanced Technologies 20](#_Toc173681498)

[Laravel Framework: 20](#_Toc173681499)

[HTML, CSS: 20](#_Toc173681500)

[Bootstrap Framework: 20](#_Toc173681501)

[Composer: 20](#_Toc173681502)

[XAMPP: 20](#_Toc173681503)

[VS Code (Visual Studio Code): 20](#_Toc173681504)

[Security Measures: 21](#_Toc173681505)

[Achievements 21](#_Toc173681506)

[User-Centric Design: 21](#_Toc173681507)

[Comprehensive Service Suite: 21](#_Toc173681508)

[Transparent and Accountable Governance: 21](#_Toc173681509)

[REFERENCES 21](#_Toc173681510)

# Introduction

Local council websites are essential for fostering resident engagement and providing access to crucial public services. This project aims to evaluate and enhance the accessibility of a council website, ensuring that information and services are readily available to all citizens.

The goal is to develop a dynamic website for the local council authority that offers a comprehensive range of public services, such as applying for benefits, filing council taxes, and accessing various other services. This platform will serve as a reliable and inclusive resource for the entire community, enabling efficient and effective interaction with the council's services.

# Purpose

Local council websites are vital in fostering resident engagement and providing access to essential public services. This project aims to develop a dynamic website for the local council authority, focusing on evaluating and improving accessibility. The goal is to ensure that information and services are readily available to all citizens, enabling them to apply for benefits, file council taxes, and access various other services seamlessly. By creating an inclusive and user-friendly online platform, we seek to enhance the overall interaction between the council and the community, ensuring that all residents, regardless of their circumstances, can efficiently access the services they need.

# Requirement Analysis

## Functional Requirements

**Home Page:** Serves as the central hub for accessing all council services and information.

Features: Navigation links to key sections (About, Council Members, Pay Taxes, Properties, Complaints, Contact).

**About Page:**  Provides comprehensive information about the council.

Features: Background and history of the Local Council Authority. Details about the council's mission, values, and long-term vision. Information about the organizational structure and key personnel.

**Council Members Page:** Profiles of elected officials.

Features: Names, images, and contact details of council members. Background information and areas of specialization.

**Pay Taxes Page:** Secure online system for filing and paying council taxes.

Features: Secure login for residents. Forms for filing taxes. Integration with payment gateways. Display of payment history and receipts.

**Properties Page:** Listings of properties for sale or rent within the council’s jurisdiction.

Features: Search and filter options. Detailed property information (dimensions, location, cost, facilities). Contact information for property inquiries.

**Property Details Page:** In-depth view of individual properties.

Features: High-definition images. Comprehensive list of features and amenities. Inquiry form for potential buyers or renters.

**Complaint Page:** Platform for residents to submit complaints and feedback.

Features: Online form for submitting complaints. Categories for different types of issues. Status tracking for submitted complaints.

**Contact Page:**  Provides multiple communication channels for residents.

Features: General inquiry form. Phone numbers and email addresses. Physical address and map of council offices.

## Non-Functional Requirements

**User Interface (UI):** The website must have a clean, modern design with an intuitive interface.

Details: Consistent branding across all pages. Easy navigation with a clear menu structure. User-friendly layout for all devices.

**Performance:** The website must load quickly and handle high traffic efficiently.

Details: Optimize images and other media for fast loading times. Implement caching strategies to reduce server load. Ensure the website can handle peak traffic periods without performance degradation.

**Security:** The website must protect user data and ensure secure transactions.

Details: Use HTTPS for all communications. Implement strong encryption for sensitive data. Regularly update security measures to prevent vulnerabilities.

**Accessibility:** The website must be accessible to all users, including those with disabilities.

Details: Comply with WCAG 2.1 guidelines. Provide alternative text for images. Ensure keyboard navigation is possible for all interactive elements.

**Compatibility:** The website must work seamlessly across different browsers and devices.

Details: Support for major browsers (Chrome, Firefox, Safari, Edge). Responsive design to ensure usability on desktops, tablets, and smartphones.

# Scope

The Local Council Authority website is designed to provide a comprehensive, dynamic platform for residents to access a range of essential public services and information. The scope of this project includes the development of the following key pages and features:

**Home Page:** The central hub for navigating the website, featuring links to various sections, recent updates, and calls to action for key services.

**About Page:** Detailed information about the council’s history, mission, values, structure, and key personnel.

**Council Members Page:** Profiles of elected officials, including their contact details and areas of specialization.

**Pay Taxes Page:** A secure, user-friendly system for residents to file and pay their council taxes online.

**Properties Page**: Listings of properties available for sale or rent, with detailed information and contact options.

**Property Details Page:** Comprehensive details about individual properties, including high-definition images and inquiry forms.

**Complaint Page:** An online platform for residents to submit complaints and provide feedback, with status tracking for submissions.

**Contact Page:** Multiple communication channels, including a general inquiry form, phone numbers, email addresses, and a map of council offices.

The project will adhere to high standards of user interface design, performance, security, accessibility, and compatibility. The aim is to create an inclusive, reliable, and efficient online resource that meets the diverse needs of the community, ensuring that all residents can easily access and utilize the council's services.

# System Architecture and Design

## Front-End Architecture

**Technologies**: HTML, CSS, Bootstrap, jQuery

**Features**: Responsive design for all devices. Interactive elements for improved user engagement. The site's content is laid out using HTML, this is the most fundamental part, while the styles used to give it an appealing aspect are CSS. By using jQuery and AJAX, website's interfaces become responsive dynamic and contents are changed dynamically just without page reloads. To make sure the website is responsive and looks well on all devices, Bootstrap is used. Modularity and structured architectural advantage of front-end development, it is easily modifiable and maintainable. In order to make sure that this website is accessible to individuals with disabilities it is also ensuring that the relevant standards for web accessibility are applied.

## Back-End Architecture

**Technologies**: PHP (Laravel framework), Laravel's ORM

**Features**: Server-side logics of the website are tasking Laravel. This entails processing form submissions, authenticating users, carrying out database operations, and generating responses. The connection between the front-end and the back-end is accomplished with the use of PHP which is a popular server-side scripting language for its simplicity and functionality. Server-side code has an eye on maintaining the effectiveness so that the website could process a big number of users as well as work with their requests. Furthermore, it has built-in error correction mechanisms to prevent the website from crashing when something goes wrong.

Data storage and retrieval are handled on the website with the help of this relational database management system. The database keeps position of the council, locals, properties, tax rates, and complaints. Given the right restraints and guides, it is the purpose of this system to ensure the data confidentiality, integrity and availability adhering to the CIA trait.

Eloquent, Laravel's ORM that replaces the obviate need of writing numerous queries is used to trawl through the database. This enables "builder" apps to manage and retrieve data efficiently simply by clicking buttons, which eliminates the need to create complicated SQL queries. Moreover, the ORM is a safeguard against SQL injection attacks which ensures the security of database operations.

## Deployment Architecture

**Technologies**: Laravel installed on localhost for development. XAMPP is an open-source software package that provides a local web server environment for testing and development. It helps you test web applications locally before deployment, ensuring they function correctly on a live server. Composer is also installed as a package management tool in PHP. Its main purpose is to make it easier to manage the libraries in this project by installing and updating them automatically. I am also using VS-Code. Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. I am utilising mostly the open-source technologies which will helps me to test web applications locally before deployment, ensuring they function correctly on a live server.

**Deployment and Publish:** Once development and testing have been completed application can be hosted on any major hosting providers including AWS/Azure. CI/CD pipelines and GitHub will be integrated.

**Features**: Automated deployment processes. Version control and file management. Regular backups and disaster recovery plans.

## UML Representation

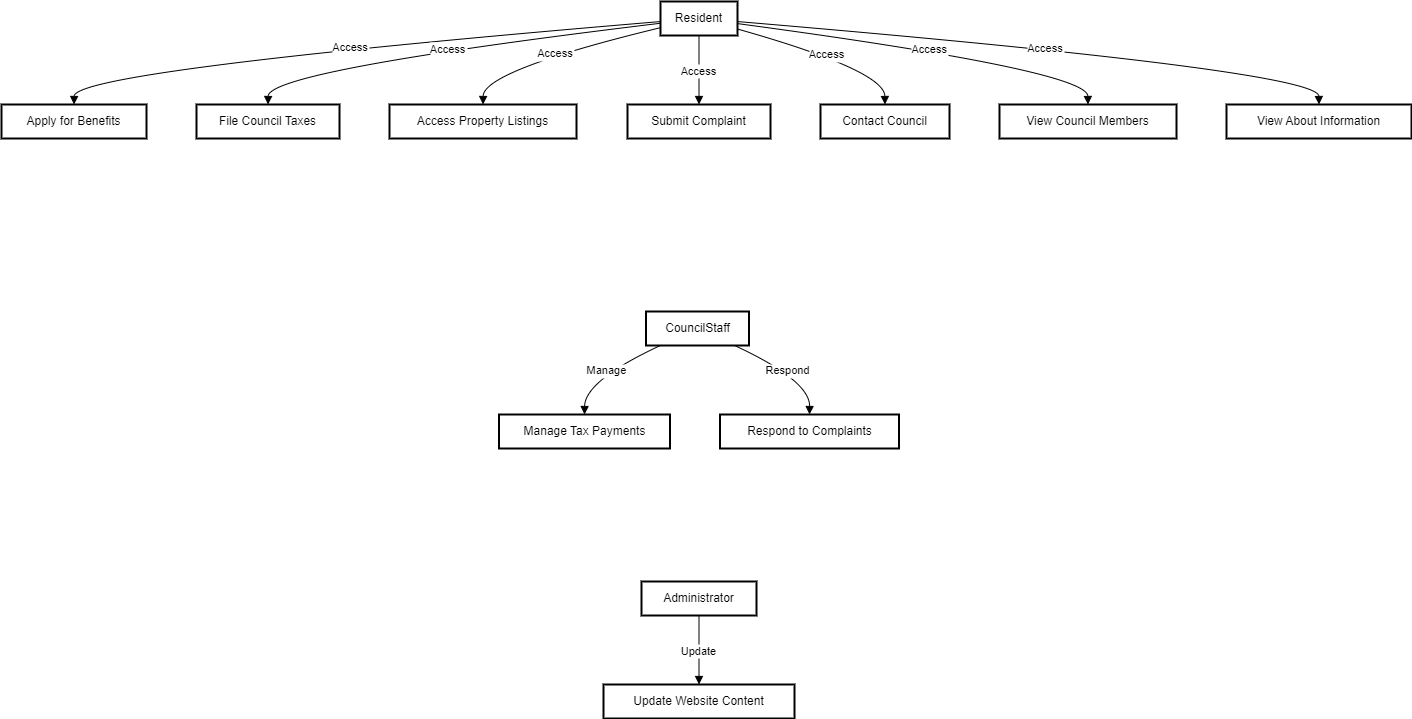
**Technology:** draw.io SaaS application

**Feature:** draw.io is a technology stack for building diagramming applications, and the world's most widely used browser-based end-user diagramming software. draw.io is a registered trademark of JGraph Ltd and draw.io AG. Unified Modeling Language (UML) diagrams are essential tools in software engineering for visualizing, specifying, constructing, and documenting the artifacts of a software system. They provide a standardized way to understand the design and architecture of the system, enabling better communication and collaboration among stakeholders.

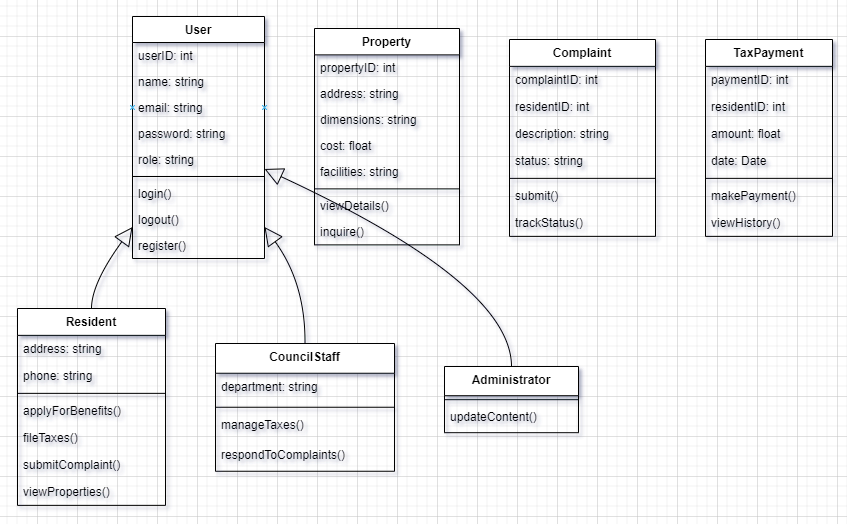
# UML DIAGRAM

In this section, we present a series of UML diagrams that illustrate the functional and structural aspects of the Local Council Authority website.

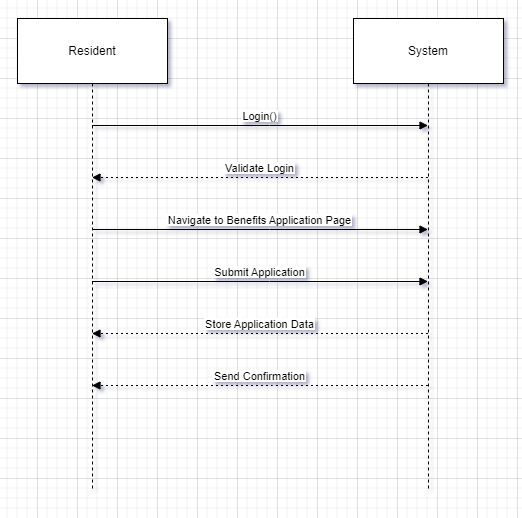
Use Case Diagram: Illustrates the interactions between users (actors) and the system, highlighting the different functionalities available to residents, council staff, and administrators.



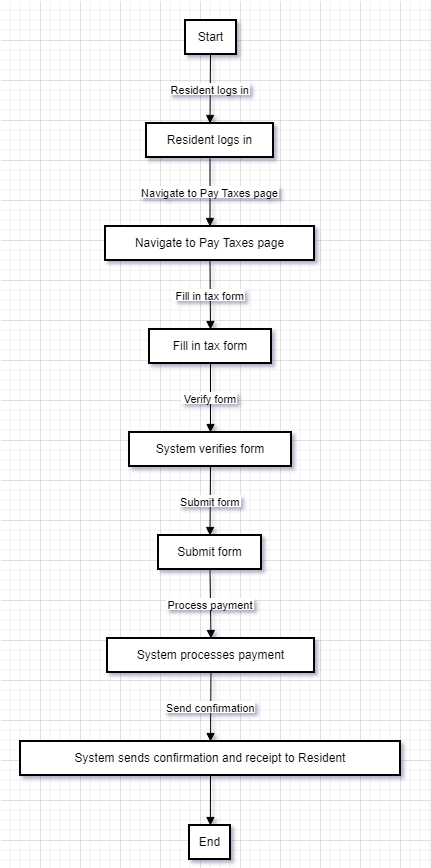
Class Diagram: Describes the static structure of the system, showing the system's classes, attributes, methods, and the relationships between objects.



Sequence Diagram: Depicts the sequence of interactions between objects in a specific scenario, detailing the flow of operations for a use case.



Activity Diagram: Visualizes the dynamic aspects of the system, illustrating the workflow and the sequence of activities involved in specific processes.



These diagrams serve as blueprints for understanding the system's architecture, ensuring that all stakeholders have a clear and consistent view of the system's functionality and design. They also facilitate the development process by providing detailed specifications that guide the implementation and testing phases.

# Configuration, Deployment and Testing on localhost environment:

Below are step-by-step instructions for installing Laravel, XAMPP, Composer, and Visual Studio Code (VS Code) on localhost system to configure, deploy and test Local Authority website.

### Install XAMPP

Step-by-Step Instructions:

1. Download XAMPP:
   * Go to the XAMPP download page.
   * Choose the appropriate version for your operating system (Windows, Linux, macOS).
2. Install XAMPP:
   * Run the downloaded installer.
   * Follow the installation wizard steps.
   * Choose the components you need (Apache, MySQL, PHP, etc. are recommended).
   * Select the installation directory (default is usually fine).
   * Click 'Next' and then 'Finish' to complete the installation.

Start XAMPP:

* Open the XAMPP Control Panel.
* Start the Apache and MySQL modules by clicking the 'Start' buttons next to each.

### Install Composer

Composer is a dependency manager for PHP, to manage project dependencies easily.

**Step-by-Step Instructions:**

1. **Download Composer:**
   * Go to the Composer download page.
   * Follow the instructions to download the Composer-Setup.exe file for Windows.
2. **Install Composer:**
   * Run the Composer-Setup.exe file.
   * Follow the installation wizard steps.
   * Ensure that the PHP executable path is correctly set (usually C:\xampp\php\php.exe if XAMPP is installed in the default directory).
   * Complete the installation and verify by opening a command prompt and typing composer -V to check the version.

### Install Visual Studio Code (VS Code)

VS Code is a free source-code editor made by Microsoft for Windows, Linux, and macOS.

**Step-by-Step Instructions:**

1. **Download VS Code:**
   * Go to the VS Code download page.
   * Choose the appropriate version for your operating system.
2. **Install VS Code:**
   * Run the downloaded installer.
   * Follow the installation wizard steps.
   * Select additional tasks (like adding to PATH) as needed.
   * Complete the installation and launch VS Code.

### Install Laravel

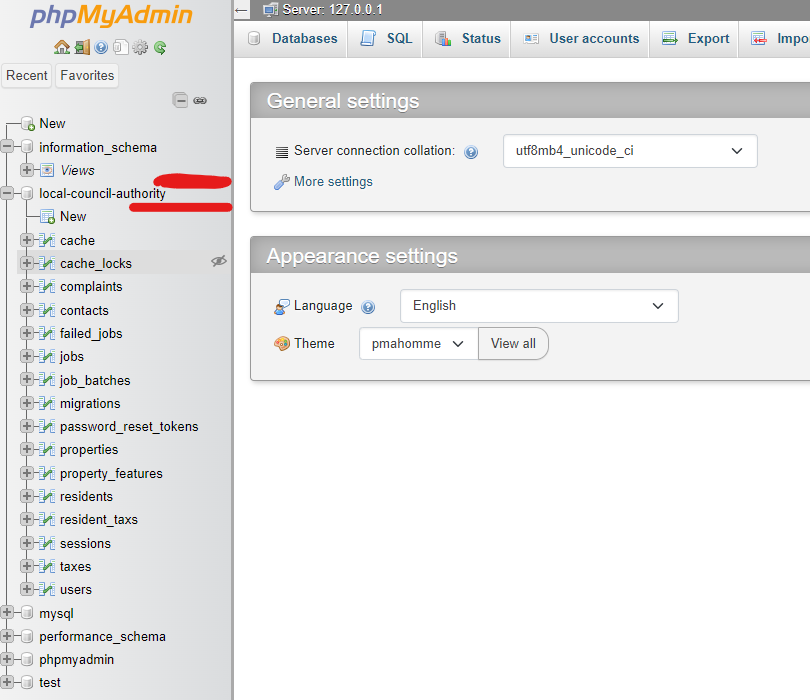
Laravel is a PHP framework for web artisans, designed for building modern web applications.

**Step-by-Step Instructions:**

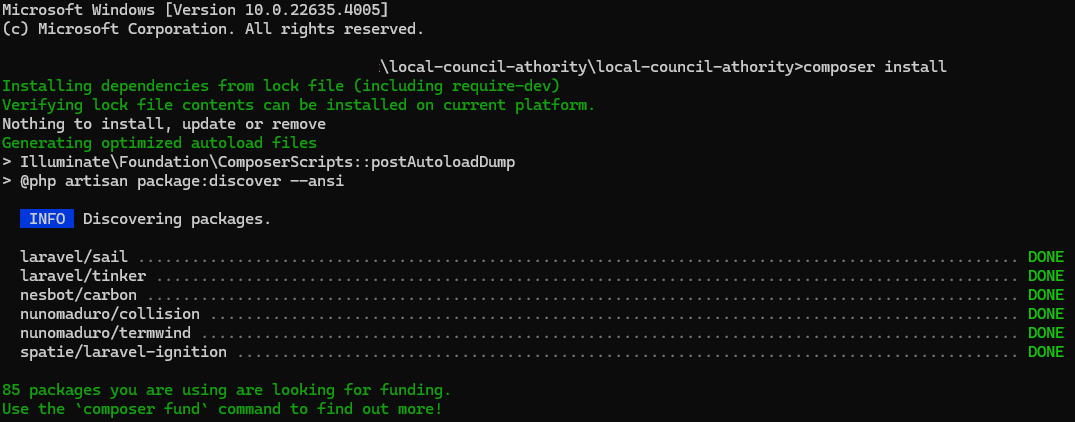
1. **Open Command Prompt:**
   * Open a command prompt or terminal window.
2. **Install Laravel via Composer:**
   * Navigate to the directory where you want to install Laravel.
   * Run the following command to create a new Laravel project

### **Your environment is ready now to import and run project.**

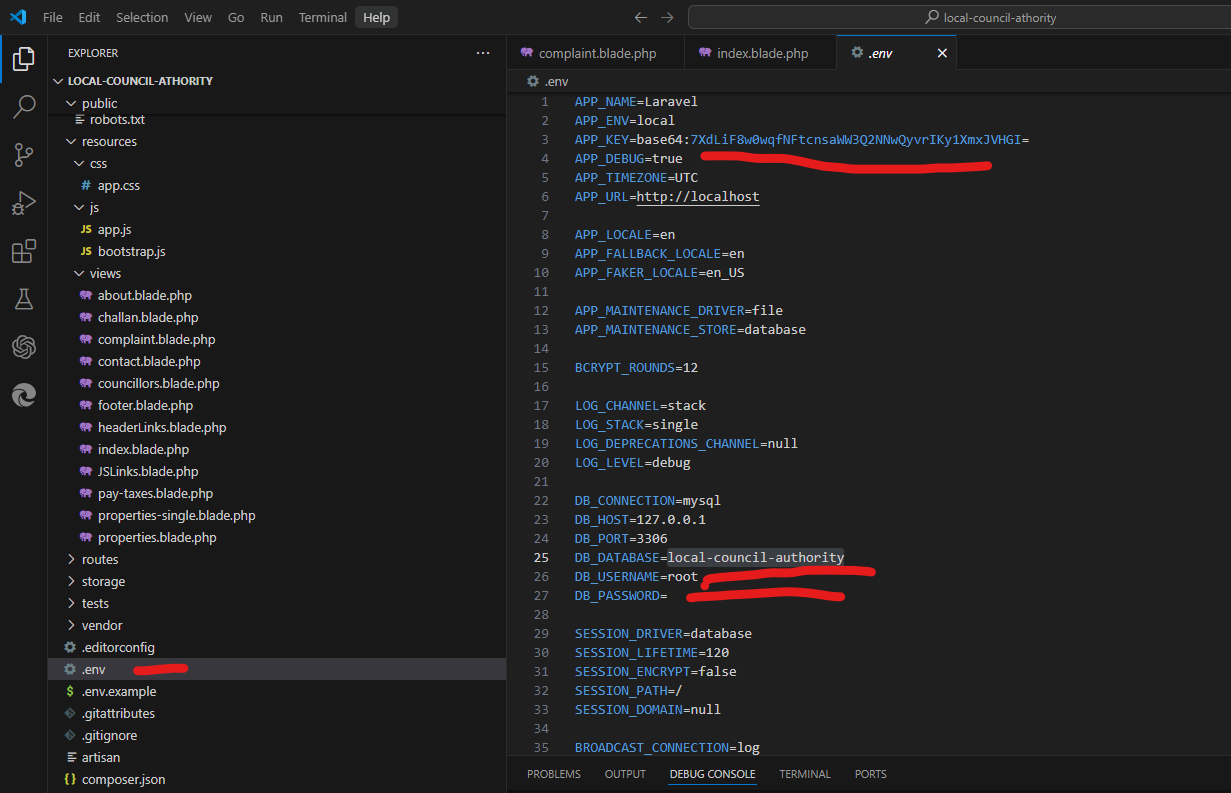
* Download source code zip file of local council authority website and extract the content to a suitable location on your Device or Server.
* Browser to PHP admin portal <http://localhost/phpmyadmin/> and create a new database named **local-council-authority**



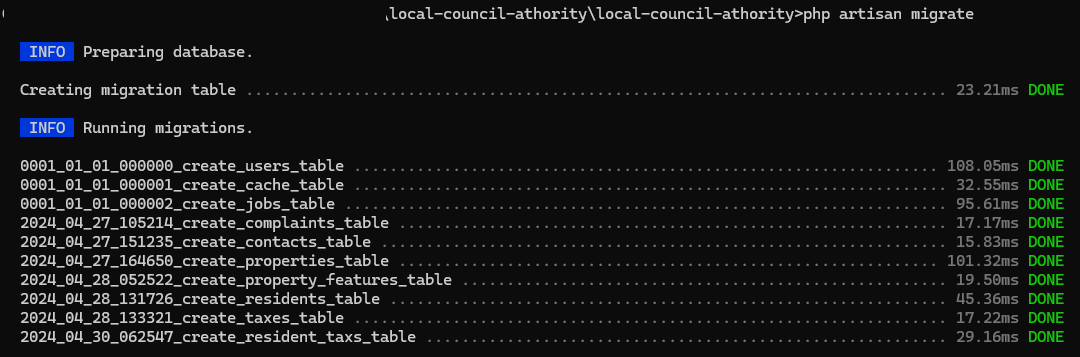
* Open cmd.exe and run command ***>composer install*** this will install required packages for our project. Ensure that you are in the local-council-authority folder.



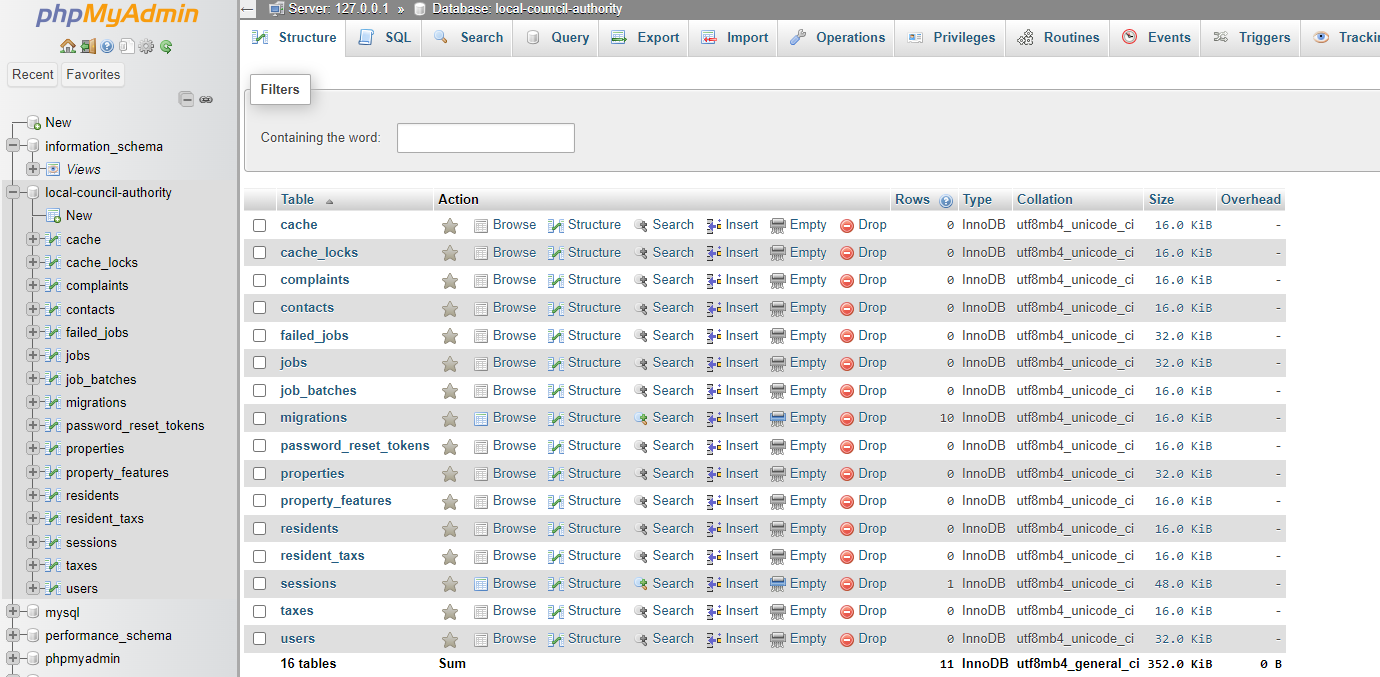
* Now open VS-Code and open the project folder in VS-Code folder tab
* On left side panel open .evn file and configure correct database name, password and APP Key. You can generate a new APP KEY if not exist. To generate App key uses ***>php artisan key:geneate*** Copy and past this new key in .env file as below.



* At this point database has been created and configured in .env config file and ready to migrate database table into new database you previously created in PHP admin panel. To migrate datebase from the project folder you need run below command ***>php artisan migrate***



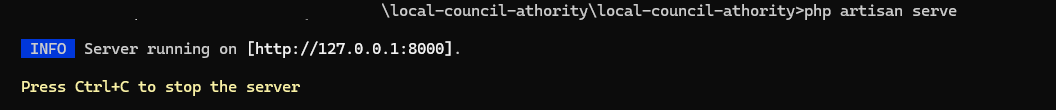
This command will migrate all tables into **phpMyAdmin** installation



* Next step is to link public folder to the storage folder in our project folder. Run ***>php artisan storage: link command***

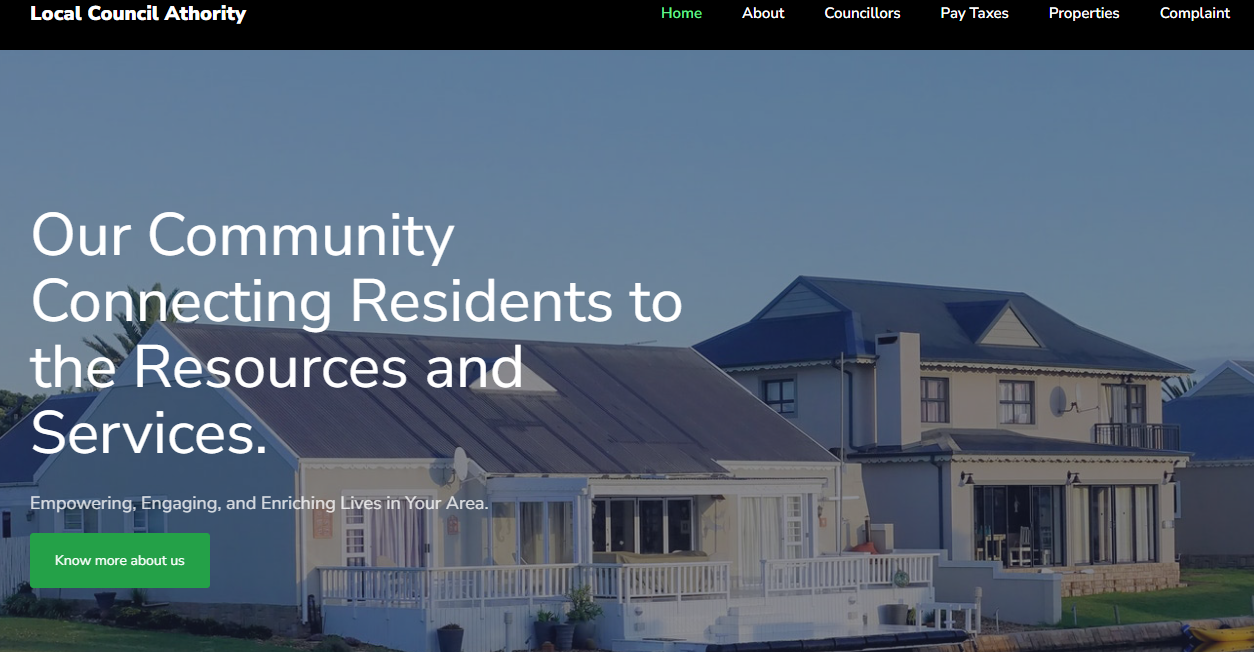
Now environment is fully configured with database migration, database links folder application folders.

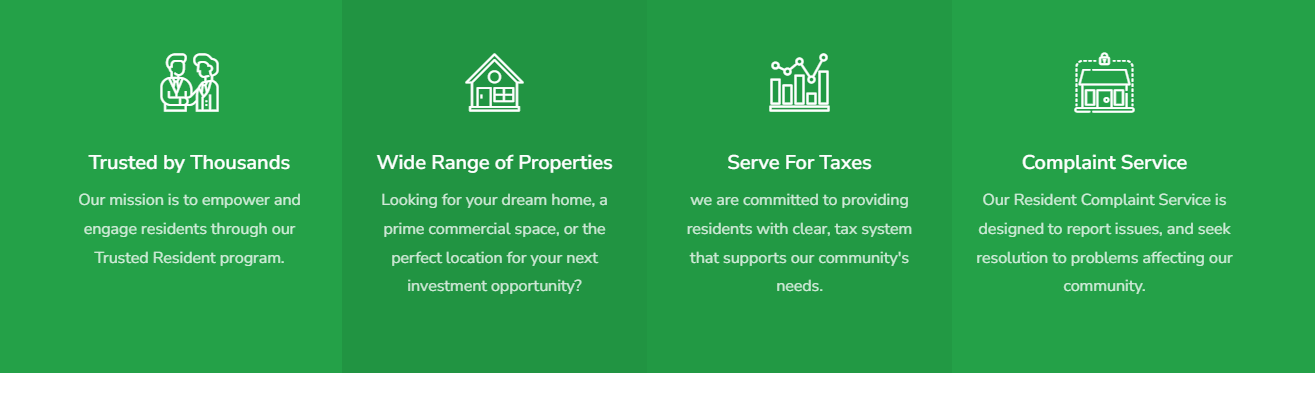
* To execute the deployment open command prompt cmd.exe and navigate to the project folder and run below command. ***>php artisan serve***

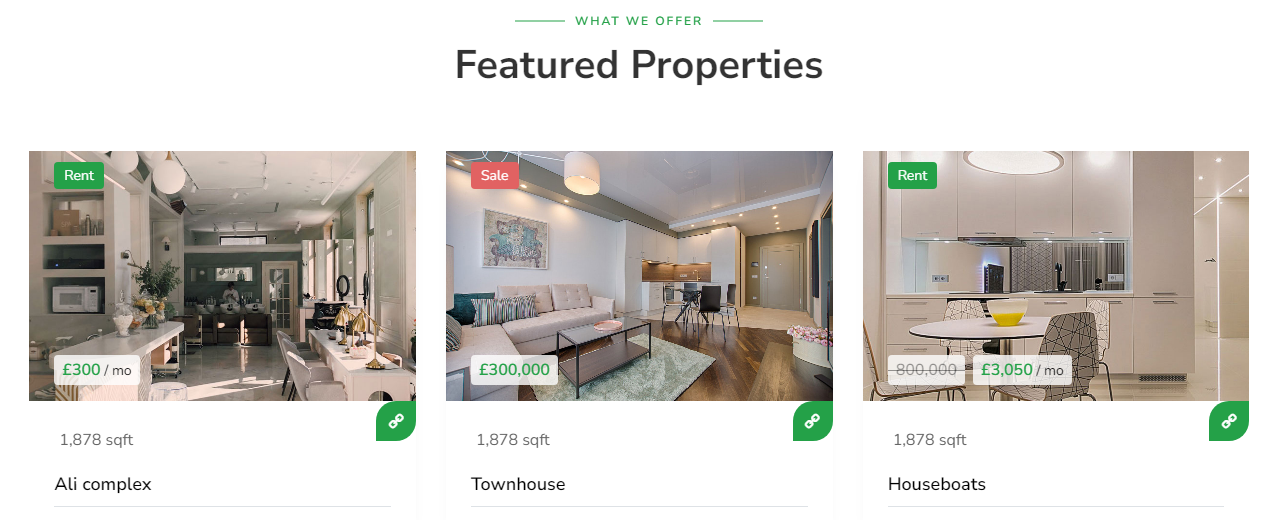


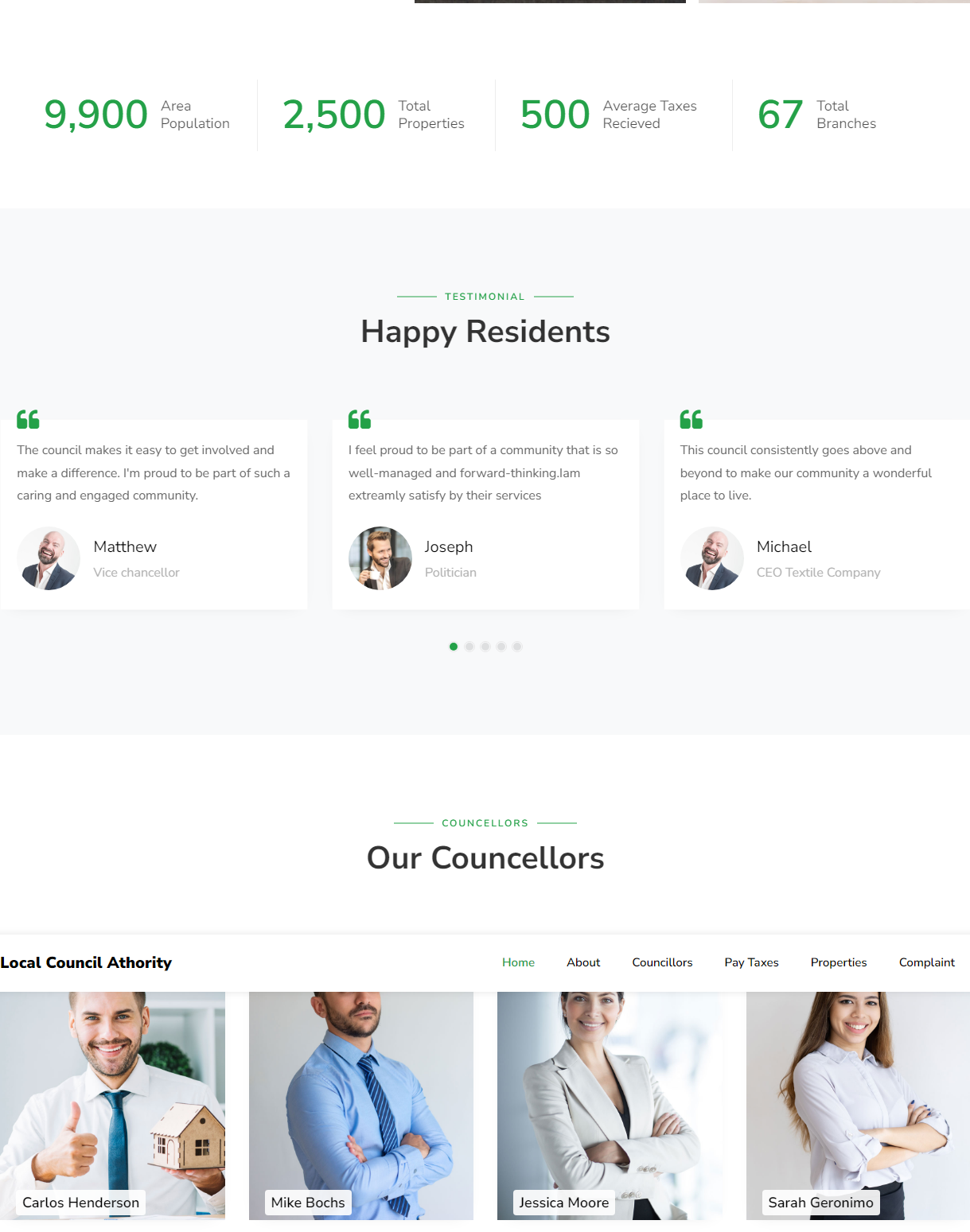
This command will start Laravel development server. Click on <http://127.0.0.1:8000> to run our website. This will provide us our ***index.php*** launching point and will open our website in a web browser.

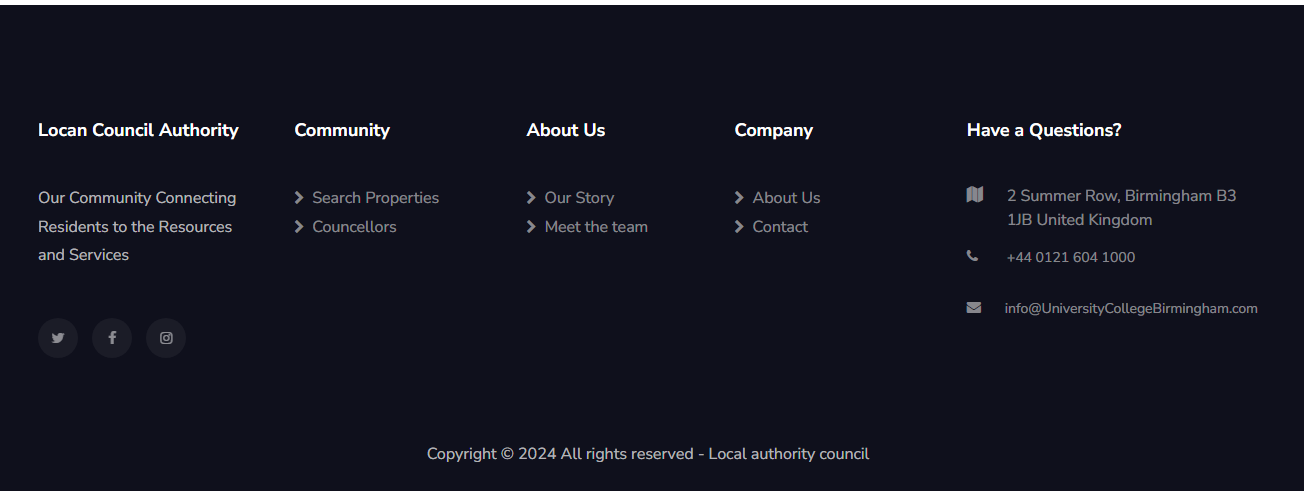
Below are some User Experience screenshots

~~~~

~~~~

~~~~

~~~~

~~~~

# Functional and Non-Functional Test Cases for Local Council Authority Website

## Functional Test Cases

### User Registration

* + Objective: Verify that a new user can register successfully.
  + Preconditions: None

Steps:

* + 1. Navigate to the registration page.
    2. Enter valid details (name, email, password, address, phone number).
    3. Click on the "Register" button.
    4. Verify the confirmation message and email received.
  + ExpectedResult: The user is successfully registered, and a confirmation email is received.

### User Login

* + Objective: Verify that a registered user can log in successfully.
  + Preconditions: User must be registered.

Steps:

* + Navigate to the login page.
  + Enter valid email and password.
  + Click on the "Login" button.
  + Verify successful login and redirection to the home page.

Expected Result: The user logs in successfully and is redirected to the home page.

### File and Pay Council Taxes

* **Objective**: Verify that a user can file and pay council taxes online.
* **Preconditions**: User must be logged in.
* **Steps**:
  1. Navigate to the "Pay Taxes" page.
  2. Enter necessary tax information.
  3. Proceed to the payment section.
  4. Enter valid payment details and submit.
  5. Verify the payment confirmation and receipt generation.
* **Expected Result**: The tax filing and payment are processed successfully, and a receipt is generated.

### Apply for Benefits

* **Objective**: Verify that a user can apply for benefits online.
* **Preconditions**: User must be logged in.
* **Steps**:
  1. Navigate to the "Apply for Benefits" page.
  2. Fill in the benefits application form with valid details.
  3. Submit the application.
  4. Verify the submission confirmation message.
* **Expected Result**: The benefits application is submitted successfully, and a confirmation message is displayed.

### Search for Property Listings

* **Objective**: Verify that users can search for properties and view details.
* **Preconditions**: None
* **Steps**:
  1. Navigate to the "Properties" page.
  2. Use the search filters (e.g., location, price range, property type) to find properties.
  3. Click on a property listing to view its details.
  4. Verify the property details and images are displayed correctly.
* **Expected Result**: The properties can be searched, and details are displayed correctly.

### Submit a Complaint

* **Objective**: Verify that a user can submit a complaint and track its status.
* **Preconditions**: User must be logged in.
* **Steps**:
  1. Navigate to the "Complaint" page.
  2. Fill in the complaint form with valid details.
  3. Submit the complaint.
  4. Verify the submission confirmation and status tracking functionality.
* **Expected Result**: The complaint is submitted successfully, and the status can be tracked.

### Non-Functional Test Cases

### Performance Testing (Unable to test as Website is locally hosted)

* + **Objective**: Ensure the website loads quickly and handles high traffic efficiently.
  + **Preconditions**: Website is fully deployed.
  + **Steps**:
    1. Tool like JMeter can be used to simulate multiple users accessing the website simultaneously.
    2. Monitor the website's response times and load times.
    3. Identify any performance bottlenecks or slow pages.
  + **Expected Result**: (Unable to test as Website is locally hosted)

### Security Testing Unable to test as Website is locally hosted

* **Objective**: Verify that the website is secure from common vulnerabilities.
* **Preconditions**: Website is fully deployed.
* **Steps**:
  1. Perform a security audit using any suitable tool.
  2. Test for common vulnerabilities such as SQL injection, XSS, CSRF, and insecure authentication.
  3. Verify that all sensitive data is encrypted.
* **Expected Result**: Unable to test as Website is locally hosted

### Accessibility Testing Unable to test as Website is locally hosted

* **Objective**: Ensure the website is accessible to users with disabilities.
* **Preconditions**: Website is fully deployed.
* **Steps**:
  1. Tools like WAVE or Axe can evaluate the website's compliance with WCAG 2.1 guidelines.
  2. Manually test keyboard navigation for all interactive elements.
  3. Check that all images have alternative text.
* **Expected Result**: Unable to test as Website is locally hosted

### Usability Testing

* **Objective**: Ensure the website is user-friendly and intuitive.
* **Preconditions**: Website is fully deployed.
* **Steps**:
  1. Conduct usability testing sessions with real users.
  2. Observe users as they complete common tasks such as registration, login, and service access.
  3. Gather feedback on the ease of use and any difficulties encountered.
* **Expected Result**: Users can navigate and use the website easily, with minimal guidance needed.

### Compatibility Testing Unable to test on tablet and phone as website is currently host on localhost server

* **Objective**: Ensure the website works seamlessly across different browsers and devices.
* **Preconditions**: Website is fully deployed.
* **Steps**:
  1. Test the website on various browsers (Chrome, Firefox, Safari, Edge).
  2. Test the website on different devices (desktop, tablet, mobile).
  3. Verify that the website displays correctly and functions properly on all tested platforms.
* **Expected Result**: The website is compatible with all major browsers and devices, providing a consistent user experience.

### Load Testing Unable to load test as Website is locally hosted

* **Objective**: Ensure the website can handle peak loads without performance degradation.
* **Preconditions**: Website is fully deployed.
* **Steps**:
  1. Tools like LoadRunner can be used to simulate high traffic conditions.
  2. Gradually increase the number of simulated users until the system's limits are reached.
  3. Monitor the website's performance and resource usage.
* **Expected Result**: The website handles peak loads efficiently, with no significant performance degradation or crashes. Unable to load test as Website is locally hosted

# Conclusion

The design and implementation of the Local Council Authority's website represent a significant advancement in making public services more accessible, efficient, and transparent. By integrating state-of-the-art technologies and adhering to best practices in web development, I have created a robust platform tailored to meet the diverse needs of the community.

## Use of Advanced Technologies

### Laravel Framework:

Laravel was chosen for its elegant syntax and powerful features, such as built-in authentication, routing, and database management. This PHP framework allowed us to create a scalable and maintainable back-end structure, ensuring robust server-side logic and seamless integration with the front-end.

### HTML, CSS:

The front-end of the website was developed using HTML, CSS, These technologies ensured a responsive and interactive user interface, offering an intuitive navigation experience across all devices. CSS was used to create a visually appealing design, while scripting added interactivity and dynamic content updates.

### Bootstrap Framework:

Bootstrap was employed to streamline the development process and ensure the website is fully responsive. Its grid system and pre-designed components allowed us to create a mobile-first design, ensuring the website looks and functions well on smartphones, tablets, and desktops.

### Composer:

Composer was used to manage PHP dependencies, ensuring that all libraries and packages required by Laravel were up-to-date and compatible. This facilitated a smooth development workflow and simplified the management of third-party packages.

### XAMPP:

XAMPP provided a local development environment with Apache, MySQL, and PHP, enabling us to develop and test the website locally before deployment. This setup ensured that the development environment closely mirrored the production server, reducing potential issues during deployment.

### VS Code (Visual Studio Code):

VS Code was the primary development tool, offering a rich set of features such as IntelliSense, debugging, and Git integration. Its extensions for Laravel and PHP development enhanced productivity and code quality.

### Security Measures:

HTTPS was implemented to secure all communications between the users and the server. Additionally, Laravel’s built-in security features, such as CSRF protection and data sanitization, were utilized to protect against common vulnerabilities.

## Achievements

### User-Centric Design:

By focusing on a user-centric design, we ensured that the website is intuitive, accessible, and responsive. Every feature and page was crafted to enhance user engagement and simplify access to services.

### Comprehensive Service Suite:

The website provides a wide range of services, including detailed council information, member profiles, secure tax payments, and a streamlined complaint submission process. Each feature promotes greater community involvement and trust in local government operations.

### Transparent and Accountable Governance:

The platform fosters transparency and accountability by making detailed council information and operations readily available to the public. This allows residents to stay informed and actively participate in local governance.

In conclusion, the Local Council Authority's website not only modernizes public service delivery but also sets a foundation for continuous improvements and future enhancements. By bridging the gap between traditional service delivery and modern digital solutions, we are empowering the community with greater access, convenience, and transparency. This project marks the beginning of a new era of public service, where every resident can engage with their local government effectively and meaningfully.

# REFERENCES

<https://elementor.com/blog/website-header-design/>

<https://www.design19.org/blog/practical-guide-how-to-use-headings/>

<https://www.w3schools.com/html/html_headings.asp>

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/Heading_Elements>

<https://searchatlas.com/blog/mastering-website-headings-a-step-by-step-guide/>

<https://usability.yale.edu/web-accessibility/articles/headings>

https://calistocode.com/html-headings/