PLAGIARISM SCAN REPORT

0% Plagiarised	100% Unique	**Date**	2024-11-03
		Words	1000
		Characters	7080

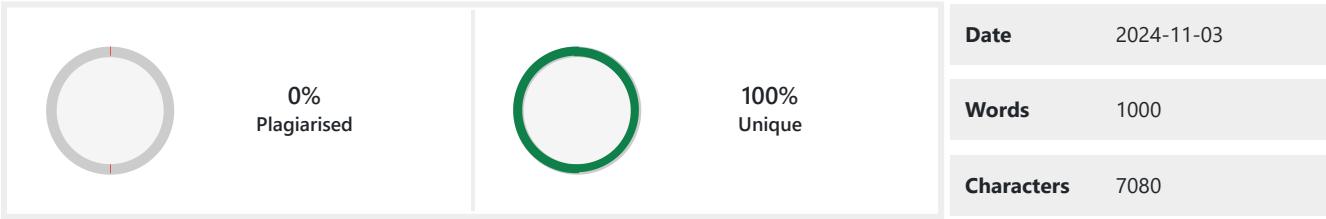**Content Checked For Plagiarism**

2.   Overall Description

2.1   Product Perspective

The City Council platform is meant to be a key part of how the city manages its services. It will be an online portal that helps city officials and citizens communicate more easily, making it simpler for people to report issues, request services, and give feedback. This platform will work together with the city's existing systems, like those used for managing public services and other local government websites. The goal is to make city services more accessible and responsive to citizens' needs through a user-friendly interface.

Technically, the City Council platform will connect with various backend systems to gather and process information. It will be built using common web technologies like HTML, CSS, and JavaScript, along with a backend that supports databases and user login. The platform will need to support different types of users, including citizens, city employees, and administrators, each with their own set of permissions and features. The design will focus on making the system scalable, secure, and easy to use, ensuring that it can handle more users and requests as needed while keeping all data safe. Overall, this platform aims to modernize how the city serves its residents.

2.2   Product Functionality

The City Council platform will provide the following major functions for all user roles:

•   User Registration and Login: Users can create accounts and log in securely based on their roles (citizens, city employees, administrators).

•   Service Request Submission: Citizens can submit requests for city services, report issues, or request public works.

•   Request Tracking and Status Updates: Citizens can track the status of their submitted requests. City employees can update the status of requests (Pending, In Progress, Completed).

•   Feedback and Communication: Citizens can provide feedback on services and communicate with city officials through a chat feature. City employees can respond to citizen queries and feedback.

•   Public Information Access: Citizens can access information about local services, schools, universities, and health facilities. All users can search a database of frequently asked questions (FAQs).

•   Booking Public Halls: Citizens can book public function halls and other city-owned facilities for events.

•   Administrative Management: Administrators can create, modify, and delete employee accounts. Administrators can view service requests, and monitor payments.

•   Data Security and Privacy: The system implements security measures to protect user data and ensure compliance with data protection regulations. Administrators manage security protocols and ensure that all user data is securely handled.

2.3   Users and Characteristics

Citizens: Use the platform to report issues, access public services, book halls, manage tasks, request certificates, and provide feedback.

Require easy navigation, efficient service access, tracking of requests, and simple feedback mechanisms.

Employees: Manage and update service requests, view and respond to customer messages.

Need efficient request management tools, communication features with citizens, and task tracking capabilities.

Admins: Create, modify, and delete employee accounts, manage service requests, oversee payment and booking records, monitor request completion times, and review customer feedback.

Require comprehensive administrative tools, detailed reporting and monitoring features, and control over user accounts and service management.

2.4   Operating Environment

The City Council system operates within a specific technological and software environment to ensure optimal performance and accessibility:

1.   Operating System: The project is designed to be compatible with all major operating systems, including Windows, macOS, and Linux for desktop users, ensuring widespread accessibility across various devices.

2.   Processor: While the system can run efficiently on dual-core processors, a quad-core processor is recommended for optimal performance, particularly when processing multiple user requests simultaneously or handling complex administrative tasks.

3.   RAM: A minimum of 2 GB RAM is required, with higher RAM capacities (4 GB or more) recommended for better overall performance, especially when handling large volumes of data, such as customer requests and feedback.

4.   Storage: The application itself has a lightweight footprint, requiring less than 200 MB of storage space for installation on user devices. Server-side, it is recommended to allocate at least 10 GB of free storage space to accommodate the growing database of user accounts, requests, and feedback.

5.   Graphics: The City Council system is primarily a text-based interface with minimal graphical requirements. However, to enhance the user experience, the system supports high-resolution displays and requires a basic GPU capable of rendering responsive design elements smoothly.

The City Council system is designed to be accessible on a range of devices and operating systems while providing an optimal user experience for both citizens and administrative staff.

2.5   Design and Implementation Constraints

1.   Hardware Limitations: The system must run on a variety of devices, including older phones with limited memory and processing power. This means the developers have to ensure the system works smoothly, even on devices with as little as 2 GB of RAM. The system should also respond quickly, so developers need to optimize it to work efficiently on slower devices.

2.   Integration with Other Systems: The City Council system may need to connect with other existing city services and databases. This might force developers to use specific methods or tools that work with these systems, which could limit their choices.

3.   Required Tools and Technologies: The developers might be required to use certain technologies like HTML for the front end and specific frameworks for the backend. They might also be restricted to using certain databases like MySQL. These requirements limit the developers' options to try other tools or technologies.

4.   Handling Multiple Operations: The system may need to handle many tasks at once, like processing several user requests simultaneously. This requires careful planning to ensure everything runs smoothly, which can limit how the system is designed.

5.   Programming Language Restrictions: The programming languages used might be limited to what the team is familiar with or what the City Council's existing systems support. This means developers might not be able to use newer or more efficient languages.

6.   Communication Requirements: The system needs to use certain communication methods, like secure HTTPS, to protect user data. These required methods could limit other ways the system communicates with users other system.

Matched Source

No plagiarism found