2.1 Overview

A 3-tier architecture is a type of software architecture which is composed of three "tiers" or "layers" of logical computing which we use it for designing the architecture of our application. They are using in applications as a specific type of client-server system. 3-tier architectures provide many benefits for production and development environments by modularizing the user interface, business logic, and data storage layers. Doing so gives greater flexibility to development teams by allowing them to update a specific part of an application independently of the other parts. That is why we choose this method

• **Presentation Tier:** The presentation tier is the front-end layer in the 3-tier system and consists of the user interface. This user interface is often a graphical one accessible through a web browser or web-based application and which displays content and information useful to an end user. This tier is often built on web technologies such as HTML5, JavaScript, CSS, or through other popular web development frameworks, and communicates with others layers through API calls.

• **Application Tier:** The application tier contains the functional business logic which drives an application’s core capabilities. It’s often written in Java, .NET, C#, Python, C++, etc.

**• Data Tier:** The data tier comprises of the database/data storage system and data access layer. Examples of such systems are MySQL, Oracle, PostgreSQL, Microsoft SQL Server, MongoDB, etc. Data is accessed by the application layer via API calls.

**2.2 High level architecture**

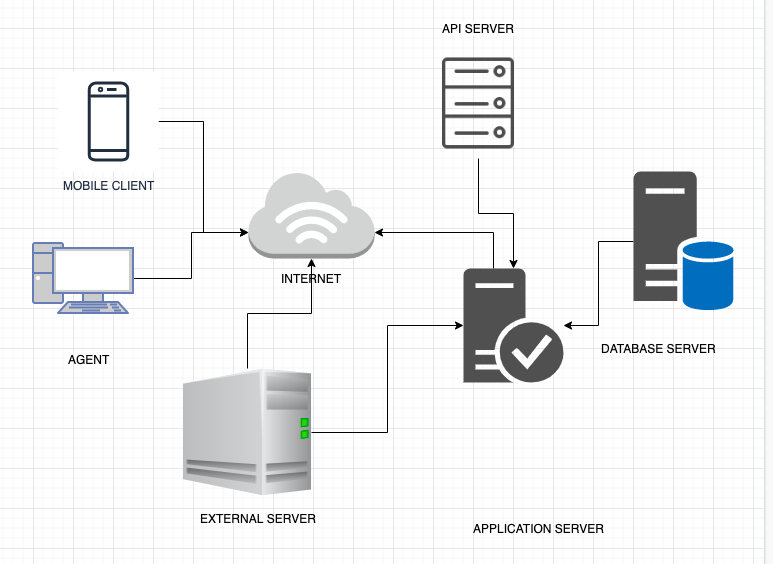


Figure 1: High Level Architecture Diagram.

The figure 1 shows the general architecture of the system. Both Registered User and municipal agent nodes are connected to the Internet in order to communicate with the External Server and Application Server nodes. Registered Userand municipal agent are registered to the application in order to communicate with the application server. Application Server is the part of the program that encodes the business logic that determine how data can be created, stored, and changed. On the other way it is the logical part of the system. Application server uses the services which External Server provides. Application Server is connected to the Database Server to have access to the data. In the Database server all the data, information, pictures and reports which used in the system will be stored, such as the personal information of Registered User.

2.3 Component view

As shown in figure 2, overall architecture of components has been carefully constructed accoring to the requirements. Safestreet is the core component of the system. From the requirements, municipalrity System will be developed on top of Safestreet and it will use for monitoring functionality. Report component will be a sub component of safestreet.

Figure 2: High Level Components Diagram.