Dental Clinic Appointment System

Supplementary Specification

Supplementary Specification

[The introduction of the **Supplementary Specification** provides an overview of the entire document.

The **Supplementary Specification** captures the system requirements that are not readily captured in the use cases of the use-case model. Such requirements include:

Legal and regulatory requirements, including application standards.

Quality attributes of the system to be built, including usability, reliability, performance, and supportability requirements.

Other requirements such as operating systems and environments, compatibility requirements, and design constraints.]

# Non-functional Requirements

Non-functional requirements include the characteristics of a system. Most of the non-functional requirements will have direct implications to a project in the form of cost, performance, further development, availability, etc.

## Availability

Availability of a system indicates when a system is operational as well as how reliable it is during operational periods.

Hours of operation- the app is available all the time

Locations of operation- available from anywhere but requires internet connection

## Performance

Response times

Processing times

Query and Reporting times

## Security

It requires the user to make an account and login

Access permissions for application data may only be changed by the system’s data admin

Password requirements

Security requirements ensure that the software is protected from unauthorized access to the system and its stored data. It considers different levels of authorization and authentication across different users roles. For instance, data privacy is a security characteristic that describes who can create, see, copy, change, or delete information. Security also includes protection against viruses and malware attacks.

## Testability

## Usability

The application is really easy to use for any type of user that knows how to use a laptop and to surf the internet.

# Design Constraints

[This section needs to indicate any design constraints on the system being built. Design constraints represent design decisions that have been mandated and must be adhered to. Examples include software languages, software process requirements, prescribed use of developmental tools, architectural and design constraints, purchased components, class libraries, and so on.]