Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <dd/mmm/yy> | <x.x> | <details> | <name> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

2. Non-functional Requirements 4

2.1 Availability 4

2.2 Performance 4

2.3 Security 4

2.4 Testability 4

2.5 Usability 4

3. Design Constraints 4

# Introduction

[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

*[Define system quality attributes in terms of scenarios according to the following template:*

* *Quality attribute definition*
* *Source of stimulus: the entity (human or another system) that generated the stimulus or event*
* *Stimulus: a condition that determines a reaction of the system*
* *Environment: the current condition of the system when the stimulus arrives*
* *Artifact: is a component that reacts to the stimulus. It may be the whole system or some pieces of it*
* *Response: the activity determined by the arrival of the stimulus*
* *Response measure: the quantifiable indication of the response*
* *Tactics*

*]*

## Availability

-the system should meet the the agreed availability target(service time and hours , providing the services it specified) as defined in the operational level agreement.

-availability will be provided if there is an internet connection and server is available.

## Performance

-the performance non-functional requirement refers to how well does the system perform its tasks

-performance characteristics:

* any number of users should be able to access at a time
* all the details of the users should be stored securely
* the response of the system for the request of the user should be fast and accurate
* the system interacts efficiently with the user

## Security

-this non-functional requirement refers to the protection of your system and its data

-for this application to provide security ,the database in which all the information is stored needs to be secured and only the administrator should be given access to make modifications in the database.

## Testability

-this requirement refers to the degree to which a software artifact supports testing in a given context.

-different modules of the application will be tested in order to find an d prevent error that might occur at the moment of use

## Usability

-this requirement refers to the degree to which a software product can be used by specified consumers in order to achieve quantified objectives with effectiveness

# Design Constraints

-the application must respect the layered architectural pattern

[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.]