<Patient Guided Assessment System >

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.]

Given that the “Patient Guided Assessment System” is intended to serve all (or at least the majority) of the people entering a hospital , it should have some well-defined application standards. It is clear that the system should be a reliable one (given that its uses are in the spectrum of health care ) and also a performant one( we need a way to process the data pretty fast and output the right result ). In terms of supportability , the application is not intended to be used from personal computers . It is mainly built for special devices that have to be placed in hospitals, clinics etc.( otherwise the idea of having the medical personnel checking the results will not be possible).

The operating system in which the application is run is Windows . The application is developed using the layered architecture , for a better testability of the components .

# 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

**Availability** is the probability that an item will be in an operable and committable state at the start of a mission when the mission is called for at a random time, and is generally defined as uptime divided by total time (uptime plus downtime).

Availability Tactics are: Fault Detection , Fault Recovery , Fault Prevention .

Source of stimulus : internal, external.

Stimulus: Fault(timing, response).

Environment : Normal operation , Degraded mode.

Artifact: processors, communication.

Response: detect failure and either record it, continue to operate- normal or degraded.

## Performance

**Performance** is the amount of useful work accomplished by a computer system. Outside of specific contexts, computer performance is estimated in terms of accuracy, efficiency and speed of executing [computer program](https://en.wikipedia.org/wiki/Computer_program) instructions.

Source of stimulus: one of a number of independent sources,external or internal.

Stimulus: arrival of periodic events /sporadic events.

Artifact : System

Environment: normal mode.

Response: processes stimuli , changes level of service.

Response measure: latency , deadline, miss rate, data loss.

## Security

Security is the protection of [computer systems](https://en.wikipedia.org/wiki/Computer_system) and [networks](https://en.wikipedia.org/wiki/Computer_network) from the theft of or damage to their [hardware](https://en.wikipedia.org/wiki/Computer_hardware), [software](https://en.wikipedia.org/wiki/Software), or [electronic data](https://en.wikipedia.org/wiki/Data_(computing)), as well as from the [disruption](https://en.wikipedia.org/wiki/Denial-of-service_attack) or [misdirection](https://en.wikipedia.org/wiki/Botnet) of the services they provide.

Source of stimulus: individual or system , internal-external, authorized, access to limited resources / vast resources.

Stimulus: tries to display data / change data / access system sevices.

Artifact: system services, data in system.

Evironment : online, connected.

Response: authenticates the user / hides identity of the user.

## Testability

**Testability** is the degree to which a software artifact (i.e. a software system, software module, requirements- or design document) supports testing in a given test context. If the testability of the software artifact is high, then finding faults in the system (if it has any) by means of testing is easier.

Source of stimulus : unit developer, increment integrator, system verifier.

Stimulus: analysis / design / architecture / class / system delivered.

Artifact: design , code.

Environment: design time , compile time, development time.

Response : provides access to state values , prepares test environment.

Response measure: time to perform test, probability of failure if fault exists.

## Usability

Usability is the ease of use and learnability of a human-made object such as a tool or device.

Source of stimulus: end user.

Stimulus: wants to learn system features / use system efficiently.

Artifact: system.

Environment: runtime / configure time.

Response: to support learning, to support efficient use.

Response measure: user satisfaction.

# Design Constraints

*The application will be hosted on the Apache Tomcat server. Regarding the architecture, the system is developed using a layered architecture . The logic of the application is done in JSP( Java) , while the front end is built with the use of HTML,CSS & Javascript. The database implementation is made using MySQL , with the information being modified/retrieved by SQL statements directly from the application.*