Online Medieval Weapon and Armour Store

Supplementary Specification

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 21/03/2017 | 1.0 | Initial version of supplementary specification | Nicolae-Florian Onica |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

Supplementary Specification

# Introduction

This document presents the non-functional requirements(availability, performance, security, testability, usability) which the application meets as well as design constraints imposed for developing it(tools and environment).

# Non-functional Requirements

## Availability

Availability is the ratio of time a system or component is functional to the total time it is required or expected to function.

Source of stimulus: user

Stimulus: login

Environment: normal environment

Artifact: user interface

Response: access to user page

Response measure: functionality of feature

## Performance

Performance represents the software response time.

Source of stimulus: user

Stimulus: item creation

Environment: normal environment

Artifact: database access and user interface

Response: placement of item in cart

Response measure: speed of response

## Security

Security represents protection against harm from external sources.

Source of stimulus: user

Stimulus: logging in

Environment: normal environment

Artifact: whole system

Response: user page access

Response measure:

Tactics: Using validation for the login data:

## Testability

Testability represents to which software supports testing.

Source of stimulus: developer

Stimulus: trying to access data from the database

Environment: normally running

Artifact: database access layer

Response: the result of the test

Response measure: the accuracy of the result

Tactics: using JUnit test

## Usability

Usability represents the degree to which software can be used by consumers to achieve objectives efficiently.

Source of stimulus: user

Stimulus: cart checkout

Environment: normal environment

Artifact: whole system

Response: checkout receipt

Response measure: the quantifiable indication of the response

# Design Constraints

This application will be developed using Java and Spring in a Windows environment using Eclipse and MySQL.