Muresan Alex
<Company Name>

ClothingForAll
Supplementary Specification

Version <1.0>

Revision History

Date	Version	Description	Author
11/03/19	<1.0>	Initial Requirements Statement	Muresan Alex

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

1. Introduction

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.

2. Non-functional Requirements

2.1 Availability

The system is not expected to be used in urgent scenarios so we can afford a SLA[1] of 99.5%. This translates into a yearly downtime of roughly 1 day and 19 hours, or a monthly downtime of 3 hours and 39 minutes. This time can be used to perform software updates, data compression and garbage collection.

2.2 Performance

Performance is needed for our system. In the situation of many orders at one time the system must be capable of manage them and keep track of each of this order. For this reason we can allow a response time of up to 10 seconds for request submissions in the worst case scenario. The average response time, depending on the load of the system, should be less than 1 second.

2.3 Security

The system will be secured using https encrypted connections. Also we will demand user authentication and will not keep passwords in plain text. Also the bill information will be encrypted too to respect the confidentiality contract between us and our customers.

2.4 Testability

The system will be tested independently at each step using Java Unit Testing. We aim to have over 90% test coverage, through unit and integration tests. With respect to manual testing, the system will log all information that is not displayed in the user interface, so that the system is fully observable and testable.

2.5 Usability

The user should be able to reach the desired goal in under 40 mouse clicks depending on the complexity of their chosen item. Also since the selling department works under stress the system must offer clear information with clear confirmation messages for each order.

3. Design Constraints

The system is constrained to use Java 8 as implementation language. The software development process will be the Rational Unified Process (RUP), tailored to fit the team and the project. The conceptual architecture of the system will be a client server one. The required development tools are either Eclipse IDE or IntelliJ IDEA. In terms of libraries we will use: JavaFX, Hibernate, JDBC and GSON.

[1] SLA = Service Level Agreement = Availability