S.A. Bank

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

Supplementary Specification

# 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

## Availability

High availability software is software used to ensure that systems are running and available most of the time. The application is a web app, which can be accessed anytime.

## Performance

Performance engineering encompasses the techniques applied during a systems development life cycle to ensure the non-functional requirements for performance (such as throughput, latency, or memory usage) will be met.

At the current state of the application, it fits fine in the resources that we have

## Security

Security, as part of the software development process, is an ongoing process involving people and practices, and ensures application confidentiality, integrity, and availability.

We assure security in more ways: a log in system, the passwords are stored encrypted in the database, we kill the session after a specific amount of inactivity.

## Testability

Software 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.

To see more about the test scenarios of the app, please check design and analysis document.

## Usability

In software engineering, usability is the degree to which a software can be used by specified consumers to achieve quantified objectives with effectiveness, efficiency, and satisfaction in a quantified context of use.

The app offers high usability, not only thanks to its user friendly interface, but also te support of the machine learning

# Design Constraints

We are going to use C# as the backend language, ASP.NET Core MVC as the framework. MVC design pattern. LinQ for querying the database. ML.NET is going to be used for training the machine learning models