<Travel Agency Manager>

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 22/03/2023 | <1> |  | Deac Denisa Bianca |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 Travel Agency Manager is a client-server application that allows employees of a tourism agency to manage the agency's activity. The application enables employees to add, modify, and delete vacations, manage client information, and reserve vacations for clients. The application is designed to store and manage information about clients and vacations in a database that is updated periodically based on the information provided by operators who collaborate with the agency.

The purpose of this Supplementary Specification document is to provide additional requirements that are not captured in the use-case model. These requirements include legal and regulatory requirements, quality attributes of the system, and other design constraints.

This document will detail the non-functional requirements, design constraints, and other requirements that will guide the development of the Travel Agency Manager application. The requirements in this document will ensure that the application is reliable, secure, easy to use, and meets the needs of the agency and its employees.

# Non-functional Requirements

## Availability

* Quality attribute definition: The Travel Agency Manager application should be available for use 24/7.
* Source of stimulus: Agency employees or clients attempting to access the application.
* Stimulus: The application is accessed.
* Environment: Normal operational conditions.
* Artifact: The entire system.
* Response: The system should respond by being always available for use.
* Response measure: The application should have an uptime of 99.99%.
* Tactics: The system should be designed to handle high availability, including redundancy, fault tolerance, and load balancing.

## Performance

* Quality attribute definition: The Travel Agency Manager application should perform well under heavy load.
* Source of stimulus: Agency employees or clients performing tasks within the application.
* Stimulus: The system is under heavy load.
* Environment: Normal operational conditions.
* Artifact: The entire system.
* Response: The system should respond to user requests within an acceptable time frame.
* Response measure: The application should have a response time of less than 2 seconds for user requests.
* Tactics: The system should be designed to handle scalability, caching, and efficient algorithms.

## Security

* Quality attribute definition: The Travel Agency Manager application should be secure, protecting against unauthorized access and data breaches.
* Source of stimulus: Hackers or unauthorized users attempting to gain access to the application or its data.
* Stimulus: An attempt to access or modify the application or its data.
* Environment: Normal operational conditions.
* Artifact: The entire system, including the database and network.
* Response: The system should prevent unauthorized access and protect against data breaches.
* Response measure: The application should meet industry-standard security practices and pass security audits.
* Tactics: The system should implement secure user authentication, encryption, access controls, and monitoring.

## Testability

* Quality attribute definition: The Travel Agency Manager application should be easily testable.
* Source of stimulus: Developers or testers attempting to test the application.
* Stimulus: The system is being tested.
* Environment: Testing environment.
* Artifact: The entire system.
* Response: The system should allow for easy testing and debugging.
* Response measure: The application should have a high test coverage percentage and a low number of bugs reported.
* Tactics: The system should implement automated testing and debugging tools, logging, and error reporting.

## Usability

* Quality attribute definition: The Travel Agency Manager application should be easy to use and intuitive.
* Source of stimulus: Agency employees or clients using the application.
* Stimulus: The system is being used.
* Environment: Normal operational conditions.
* Artifact: The entire system.
* Response: The system should be easy to use and require minimal training.
* Response measure: The application should have a high usability score based on user feedback.
* Tactics: The system should implement a user-friendly interface, clear navigation, and contextual help.

# Design Constraints

Design constraints are restrictions or requirements that must be taken into account during the system's design and development. They can be of a technical, administrative, or legal character. Examples of design limitations for the Travel Agency Manager application are as follows:

Java programming must be used to create the system.

The architecture of the system must be client-server.

To store and handle data, the system has to employ a relational database management system (RDBMS).

Standards set by the industry for data security and privacy must be followed by the system.

The system must work with the most recent iterations of widely used web browsers. (e.g., Chrome, Firefox, Safari).

To handle future increases in the number of customers and personnel, the system must be expandable.

The system needs to be made so that it is simple to upgrade and maintain.

For non-technical users, the system must be created to be user-friendly and intuitive.

To make sure that the system complies with the relevant requirements and limitations, these design constraints should be taken into account during the project's design and development phases.