TEAM I

QUEST 1

Software Requirements Specification v1.0

October 13, 2023

Sponsored by MonkeyBiz Limited Co.

Revision History

Date	Description	Author	Comments
October 6, 2023	Identified the initial functional requirements.	Kajal Tomar	Add the reason of each functionality.
October 6, 2023	Identified the system's eventual users and described their general characteristics.	Richard Marinas Jr.	
October 9, 2023	Identified more functional requirements and sorted the requirements into categories.	Kajal Tomar	
October 9, 2023	Identified and expanded on non-functional requirements.	Logan Doran	
October 11, 2023	Numbered Section 3 subsections and added them to the Table of Contents.	Richard Marinas Jr.	
October 11, 2023	Created title page.	Richard Marinas Jr.	
October 10, 2023	Completed section 1.	Casandra Hayward	
October 12, 2023	Completed section 3.3 - 3.5	Rishavjot Singh	

Table of Contents

1. Introduction

The Introduction section of the Software Requirement Specification (SRS) document outlines the contents of the rest of the document. It will discuss the purpose, definitions, acronyms, abbreviations, and references. The SRS will outline the overall design, requirements, assumptions, and direction of the Online Virtual Phone System that our team is developing for MonkeyBiz Limited Co. Our SRS will start by reviewing our target audience as well as any assumptions and dependencies. Then it will define our system's functional requirements such that it meets the client's expectations. After we have defined the systems functional requirements, we will define the non-functional requirements including the necessary hardware, software, performance, reliability, security, and availability to ensure user satisfaction as well as client satisfaction. Lastly, we will cover any Legal notices, other requirements, and other additional documentation.

1.1 Purpose

The purpose of the Software Requirement Specification (SRS) document is to collect our ideas on the system design, function, use case, target audience, and requirements. This document will be a roadmap used to guide the rest of the project while still giving us room to further refine aspects of the project or the flexibility to change directions. This SRS will be shared with the development team, QA team, project manager, and our client to make sure that all the relevant parties are on the same page about where this project is headed.

1.2 Definitions, Acronyms, and Abbreviations

Acronym/Definitions	-	Meaning of the acronym / definition
SRS	-	Software Requirement Specification
MonkeyBiz	-	This is the client’s company name
RTO	-	Recovery Time Objective
GUI	-	Graphical User Interface

1.3 References

Document Name and Version	Description	Location
Online Virtual Phone System Publish Year: 2023	This is the project overview provided by MonkeyBiz Limited Co. It describes the projects, desired features, and constraints.	https://umanitoba-my.sharepoint.com/:w:/g/personal/shaowei_wang_umanitoba_ca/EX6PKNTuX1hGomL4NyNYSIcBHg5ImCFl2BPrVJnFbBWIs Q?e=prDfNs
Project Charter Publish Year: 2023	This is the project charter that our team developed for the Online Virtual Phone System as proposed by MonkeyBiz Limited Co.	In UM Learn for COMP-4050 A01 – Project Management: Assessments > Assignments > Team 1: [Group Assignment] Assignment 1

2. General Description

This section describes the general factors that affect the Online Virtual Phone System and its requirements.

2.1 User Characteristics

This subsection describes the typical characteristics of the Online Virtual Phone System's eventual users, which will affect the system's requirements.

2.1.1 Customers

Customers include everyone from adults to seniors to people with disabilities. Their proficiency with technology varies. They will primarily use the system to make and receive calls.

2.1.2 Employees

Employees are trained to use the system for basic tasks like sending, receiving, holding, and forwarding calls. They communicate with customers and colleagues daily. Their role is central to the operations of MonkeyBiz.

2.1.3 Administrators

Administrators receive more training than Employees and have a higher level of technical expertise. They utilize the System Console to manage and monitor the network of phones, phone numbers, user accounts, and IP addresses.

2.1.4 Technicians

Technicians have the highest level of technical expertise. They interact with the system when issues arise to ensure that the system's stability and performance are in peak condition.

2.2 Assumptions and Dependencies

This subsection lists the external factors that affect the requirements in this document.

2.2.1 Assumptions

- Individuals with low technical expertise will use the system.
- The system will be used on devices that meet the minimum system requirements.
- The system will be used on a desktop computer and a mobile phone.
- Some calls may be conducted on an unstable internet connection.
- Calls can be long-distance and international.

2.2.2 Dependencies

- The system depends on the presence of audio input (microphone) and output (speakers or headphones) devices.

- The system depends on the availability and optimal performance of the database server that stores user info, call logs, and billing records.
- The system depends on a reliable network infrastructure, including servers and routers.

3. Specific Requirements

This section contains the requirements that the Online Virtual Phone System are expected to meet to be considered complete.

3.1 Functional Requirements

This subsection contains the functional requirements that our Online Virtual Phone System is expected to meet.

3.1.1 User Interface and Access Control

- As a MonkeyBiz customer, I want to be able to see my contacts order by most recent calls so I can make important calls withing 5 seconds.
- As a MonkeyBiz administrator, I want a specific user interface so I can monitor and control a network of phones.
- The System Console shall have a login feature so that only approved personnel may gain access.
- The system shall display all the available features of the System Console to the Administrators so they can administer the system.

3.1.2 Call Management

- As a MonkeyBiz customer, I want to make and receive audio calls through the internet, so I do not need to have a traditional phone service.
- The system will provide a record of every established call so that users can maintain a history of their communication and for administrative purposes.

3.1.3 System Management

- As a MonkeyBiz administrator, I want the system to allow me to associate a given phone number with a given IP address so I can assign them to customers.
- The system shall provide a phone number search so that Administrators can find unassigned phone numbers within 10 seconds.
- The system shall show available IP addresses separately so MonkeyBiz Administrators can find them conveniently within 10 seconds.

3.1.4 Service and Subscription Plan Management

- As a MonkeyBiz administrator, I want the ability to suspend or cancel a user's telephone service if the user fails to pay his or her bill so that we ensure that our services are used by customers that make timely payments.

- As a MonkeyBiz administrator, I want to be able to change which plan a customer subscribes to from the following billing plans that I can accommodate changing customer needs and preferences:
    - the regular charge rate for calls;
    - one or more periods, i.e., days of the week and times of the day; and
    - the discount rate for each of these periods.

3.1.5 Data Management and Security

- As a MonkeyBiz customer, I want to make sure that all my calls are secure so that I can feel comfortable with my privacy when making calls.
- As a MonkeyBiz customer, I want to be able to access my records, so I do not have to keep track of everything myself.

3.1.6 Billing and Discount Management

- As a MonkeyBiz customer, I want to be able to view my bill for any billing period, so I do not have to keep my own records.
- As a MonkeyBiz customer, I want a warning on the next bill if I fail to make a payment so that I do not miss another one.
- As a MonkeyBiz administrator, I want the ability to record the customer's bill payments so I can accurately maintain financial records and ensure that all payments are accounted for.
- As a MonkeyBiz administrator, I want the ability to change the amount charged for calls by adding new billing plans or editing existing ones so that I can adapt our pricing to meet the changing needs of our customers.
- The System shall automatically issue the bill to each user at the end of every billing period so that customers are billed promptly and accurately for the services they have used.
- The System will immediately issue the user a bill if the user's service is canceled so that any outstanding charges can be settled promptly upon service termination.
- The system shall check that the dates and times of the discount periods are valid (they do not overlap) so that billing accuracy and consistency are maintained.

3.2 Non-Functional Requirements

This subsection contains the non-functional requirements that our Online Virtual Phone System is expected to meet.

3.2.1 Performance

- The system shall process dialed numbers and initiate calls within 1 second of user input.
- The system shall support at least 10000 simultaneous users without loss of call quality or performance.
- The product shall be a desktop application and does not require web server hosting.

3.2.2 Reliability

- The system shall regularly back up user account data and call records to prevent data loss and corruption.
- The system shall provide storage of all databases on redundant computers with automatic switchover.
- The system shall allow for replication of databases to off-site storage locations.
- In the event of a system failure, the system shall have an RTO of no more than 30 minutes to restore full functionality.

3.2.3 Availability

- The system shall be available for use 24/7, allowing users to make and receive calls at any time.
- The system shall automatically scale up or down based on demand to maintain consistent performance.
- The system shall perform regular state checkpoints every 15 minutes to minimize data loss in case of a failure.
- In the event of a system failure, the system shall be capable of recovering to its previous state, including active calls and user data, within 5 minutes.

3.2.4 Security

3.2.4.1 Data transfer

- The system shall automatically log out all customers after a period of inactivity.
- The system shall use end-to-end encryption for all call data and user communication to ensure that data is secure during transmission.

3.2.4.2 Data storage

- The system shall retain customer records indefinitely so that historical data is preserved.
- Access to sensitive data in the database shall be restricted to authorized personnel only.
- The application shall never display a customer's password. It shall always be echoed with special characters representing typed characters.
- The system's back-end databases shall be encrypted.

3.2.5 Interfaces

3.2.5.1 User Interfaces

- The system shall support the creation of user profiles with different settings and configurations for different users.
- The system shall support a clean and intuitive graphical user interface (GUI).
- The user interface shall follow accessibility standards to accommodate users with disabilities including features such as text-to-speech and keyboard navigation.

3.2.5.2 Hardware Interfaces

- The system shall support a range of audio input and output devices.

3.2.5.3 Software Interfaces

- The system shall communicate with a database management system to store and retrieve user information and call records.
- The system shall communicate with a web server for hosting web-based components.
- The system shall communicate with the billing system to generate invoices for users. The main system for billing and payment processing of users.
- The system shall communicate with LDAP directories for user authentication and directory look up.

3.2.5.4 Communications Interfaces

- The system shall support common local network protocols, including TCP/IP.
- The system shall implement Quality of Service (QoS) technologies.

3.3 Design Constraints

3.3.1 Structural Model

The system must be built based on a client-server model, where the server should be able to support one or more processes and possibly a database. The communication between server and client must occur over sockets as the client is connected to the system through an IP network.

3.3.2 Standard Development Tools

The system should be built using standard tools that support socket communication effectively and can handle a robust client-server network.

3.3.3 User Friendly Interface

The interface for the system should be intuitive and easy to use, following Microsoft's GUI standards.

3.3.4 Multiple Devices

There should be apps that allow users to make and receive calls and that can be installed on various end user devices like laptops, smartphones, and desktop computers.

3.3.5 End-to-End Encryption

The calls should be end-to-end encrypted, protecting the information being shared over the call.

3.3.6 Load Balancing

The system should be robust, meaning that it should be able to handle 10,000 calls concurrently, without any calls being dropped. At any time, there should not be a latency of more than 2 seconds for any call.

3.3.7 Response Time

Response time for loading the system on any device should not take more than 2 minutes.

3.3.8 Knowledge Requirements

A general knowledge of using devices like laptops, smartphones, and desktop computers is required to use the system.

3.4 Legal, Copyright, and Other Notices

1. The GUI of the system should display the copyright, word mark, and trademark of MonkeyBiz Limited Co.
2. All the code for the system should be held private. It should not be made open sourced at any point.
3. All the information, technical and non-technical, about the system is confidential. No information should be made public.

3.5 Other Requirements

The system should be built in a way that, if needed, there should be room to add new features such as: text messaging, video calls, voicemail, playback, and group calling.

The system's structure should be of the form that new features can be added without altering the existing structure, which means system should be open to take on the new features on the top of the existing system.