SOFTWARE REQUIREMENTS SPECIFICATION

for

Quicker Queue

Version 2.0

Prepared by Holden Hall, Luong Dinh, Garret Gulker

Shockwave Tech

April 30, 2019

Contents

Revision History

Version	Name	Date	Changes
Revision 1.0	Holden Hall	2/21/19	Original Document
Revision 2.0	Holden Hall	4/30/19	Added Team members to title page, Added Revision History Page, Added Appendix

1 Introduction

1.1 Purpose

The purpose of Quicker Queue is to assist different businesses manage the workflow of customers. Quicker Queue eases both the business, and customer side of priority queue systems by using instant email notifications. Customers will be able to make and edit appointments on the system.

1.2 Project Scope

Quicker Queue is for designed for businesses who need a priority queue system, or want to update from their old one. Quicker Queue has the benefit of being able to be customized for specific business needs.The system will use it's secured database to collect user information and notify them with stored email addresses. Personal information will be encrypted using AES encryption.

1.3 Acronyms, Abbreviations, Definitions

AES - Advanced Encryption Standard
Namespace - Container that holds premade classes allowing faster and easier programming.
Priority Queue System - System similar to a waiting list or line. Allows businesses to keep track of customers based on business' set standards.
Encryption - Algorithm designed to obfuscate important or private information.
Decryption - Algorithm designed to take encrypted text and return the original text.

1.4 References

http://www.mimekit.net/ - possible library for notifications
docs.microsoft.com/en-us/dotnet/csharp/ - C# and .Netframework documentation
https://github.com/jpeisenbarth/SRS-Tex - SRS IEEE LaTex template used

1.5 Outline

Section 2 will go over the general design of the product and how it works. Section 3 will go into the specifics and looks of the Interfaces, system requirements, functional requirements, and limitations.

2 Overall Description

2.1 Context of Product

This system is designed to improve on current priority systems. One way Quicker Queue beats other priority queue systems is the instant notifications. The users are notified via email when their appointment is ready. Quicker Queue will allow customers to search for different businesses using the app, ensuring higher application usage.

2.2 Product Functions

Product Functions include: Notifying a user when they're next in the queue by email. Storing encrypted user data in a database. Allowing a customer or business to register/login to the system. Allowing customers to edit their appointment or check the queue. Allowing the business to manage the queue. Allowing the business to customize the app.

2.3 User Classes and Characteristics

There are two user classes that inherit from the superclass User, Customer and Business. The customer class has the ability to make and edit appointments, while being able to check the queue. Business manages the queue and has the ability to notify, delete, or add an appointment.

2.4 Design and Implementation Constraints

The User Interface of Quicker Queue is constrained to C# and the .Netframework. The application is constrained to scheduled iterations, where the product needs a certain amount of features each iteration. With April 18, 23, 25 being the final scheduled date. The product is constrained to work on the windows operating system.

2.5 Assumptions and Dependencies

Planned features such as notifications and encryption are dependent on 2 .Netframework namespaces(System.Security.Cryptography and System.Net.Mail) However it is possible that instead of using System.Net.Mail we use mimekit. We are assuming that the client's computer has the up to date .Netframework to run the C# code.

3 Specific Requirements

3.1 Functional Requirements

3.1.1 Functional Requirement 1

System must let customers to see all available businesses using our platform and their information.
This will allow the customer to have an higher ease of access when using the application. The customer will use the application more frequently if they need the system for more than one business. The system will allow customers to view and edit their information.

3.1.2 Functional Requirement 2

Customers must be able to search available businesses.
Similar to functional requirement 1, customer needs to be able to search for different businesses to be able to make appointments. Customer should be able to do different searches to find different businesses.

3.1.3 Functional Requirement 3

Businesses must be able to see all customers queued for their business and information necessary for their specific service.
A business needs to be able to view and manage their queue. The point of the system is to help manage customer flow. This allows the business to manage mistakes and appointments.

3.1.4 Functional Requirement 4

Businesses must be able to notify any queued customer through the system.
One of the major requirements of Quicker Queue is a fast notification system to put it above other Queue systems. Information needed to notify customers is stored in the azure database.

3.1.5 Functional Requirement 5

Customers and Businesses must be able to edit appointments.
If a customer makes a mistake or has a change of plans they must be able to edit or delete their appointment. The business also must be able to edit appointments in case there is a mistake or a customer don't show up.

3.2 External Interface Requirements

3.2.1 User Interfaces

The interface will include a simple, yet stylish design to for easy usage. Information about the application like "help" or "about" will be able on a menu bar at the top of the application. The Startup screen will be a Quicker Queue Logo above a register and login button. Standard Error Messages will be thrown when unexpected things happen, If the error is fixable, it can be sent back to Shockwave Tech to be fixed in maintenance.

3.2.2 Hardware Interfaces

Quicker Queue runs on Windows computers, and currently has no other hardware componets.

3.2.3 Software Interfaces

The product stores encrypted user data in a azure database. User data includes: names, usernames, email usernames, passwords. Business data will include: name, phone number, hours, Address, City, State. The system must run on windows most likely windows 7. We have yet to test on older versions of Windows. The application will use at least one .net namespace, possibly two. This will depend on how well the notification system works. If the mail namespace does not work we will use the mimekit library instead. We are working on NETFramework,Version v4.6.1.

3.2.4 Communications Interfaces

The system will send emails through a SMTP client. The system will also send and retrieve data through a database, so an internet connection is required. Personal information will be encrypted using AES encryption.

3.3 Performance Requirements

The product shall be able to run on a windows 7 or higher machine. The machine will require an internet connection to notify customers and access the database.

3.4 Design Constraints

3.4.1 Standards Compliance

Shockwave Tech follows the standards set by Doctor Eduardo Colmenares, and the Midwestern State University Computer Science Department.

3.4.2 Hardware Limitations

Client will need a windows 7 or higher machine with a working internet connection.

3.5 Attributes

3.5.1 Availability

The goal of the product is to be available to all businesses to ease their customer flow.

3.5.2 Security

Quicker Queue was designed with security in mind, we will secure user data using AES encryption. In case of a data breach, criminals will not have access to private user information.

3.5.3 Maintainability

Shockwave Tech will maintain the system and fix any problems businesses come across. We also have the ability to update the software to satisfy the needs of more specific businesses if the current customization options are not fit for a client.

3.6 Other Requirements

3.6.1 Database

Quicker Queue requires an Azure database

4 Appendix

Most businesses have a system that forces you to wait in line, and leaves the possibility for employes to forget people. Even If they do remember you, you might not hear them call your name. That's why Quicker Queue was designed to make the overall queueing process a more pleasurable experience, for both costumer and business. Quicker Queue allows you to make an account that works for any business using Quicker Queue. Any user can use their account to make appointments, check business queues, search for businesses, add favorites, and edit their account. A business can easily manage their business throught the Business homepage. Which will allow them to add,delete or notify costumers. A costumer is notified by email, when a business clicks the notify button. When a customer or business wants to edit their account they can change information like their password, email, username or delete their account.