**System Documentation**

**Team 9**

April 25, 2020

**Team Members**

Michael Cu

Joshua Daniels

Mao Zheng

Toni Tan

# Contents

Contents 2

1 Introduction 4

2 Setup Proxy Server 4

3 Access iOS Application Code 4

4 Install on Simulator or Device 4

5 System Maintenance 5

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Name** | **Description** |
| 1 | 4/25/2020 | Toni Tan | Initial Document |
| 2 | 4/26/2020 | Michael Cu | 3.2, 4.1, 4.2 |
| 3 | 4/26/2020 | Mao Zheng | Revisions |

# Introduction

The Commerce Bank Web Application allows Commerce Bank customers to check their balance and track their spent base on the alert system. The alert system is set up by the users based on their personal needs. This document will provide instruction for accessing the source code for the application and documentation or instructions for maintaining or deploying code.

# Access Application Code

Internet connection is required to access the code in GitHub repository.

All code for the application is found in the Github.com repository located at: <https://github.com/umkc-cs-451-2020-spring/semester-project-group-9>

This source code can be forked to another GitHub repository, cloned, or downloaded to a personal computer.

# Setup Local Environment

## Required Components

Any operating system that supports Visual studio should be able to edit, maintain, and run this application

Download Visual Studio: <https://visualstudio.microsoft.com/downloads/>

We used .Net framework and SQL Server Database

Download .Net: <https://visualstudio.microsoft.com/vs/features/net-development/>

## Install Code

1. Launch Visual Studio Code 2019
2. Clone Repository into Workspace Environment
   1. Click Clone or Check out Code
   2. Copy and Paste the following link into Repository Location:

* + 1. <https://github.com/umkc-cs-451-2020-spring/semester-project-group-9>
  1. Choose an appropriate location for the local path.
  2. Click Clone

1. Database Setup
   1. Click Server Explorer
   2. Right-Click Data Connections
   3. Click Create New SQL Server Database
   4. Select Server Name
   5. Input Database Name as CommerceBank
   6. Right-Click CommerceBank.mdf and Select New Query
   7. Paste the Query provided in the github into the New Query Textbox.
   8. Update Query
2. Debug --> Start Debugging in VisualStudio Code.
3. Web Application is now running and usable.

# System Maintenance

* 1. .NET Framework

The user interface consists of four tabs for viewing information, plus the login screen. Each of these webpages implement aspx.net. A webpage will have an html view and underlying C# functions for the page’s functionalities where the users could set/edit notification rules, pull the account information, export spreadsheet and etc.

On the first run, the website will display the login page. Once the user is logged in, they are sent to their transaction summary. The navigation bar, which is at the top of the webpage, allows users to navigate between tabs and contains all four tabs for each type of information.

The underlying C# will request data from the SQL server and pull appropriate data, format it, and display it in the correct fields.

* 1. The server uses Microsoft SQL that connects to aspx.net. The webpage will log into the server using the user’s credentials provided in the html. Afterwards, the page will redirect to a unique URL to request the user’s account balance and account history. We will then be able to access the information and data stored in the database through the webpage.