**Testing Document and Specification**

Test Plan

Deft Developers Team 1

CS 451, SP2016

|  |  |  |
| --- | --- | --- |
| Evan Bells | Steven Evans | Muying Cui |
|  | Kien Chiem |  |

# Introduction

This document outlines the test plan for the University of Missouri-Kansas City Computer Science department for our project Ach Case Tracking with Commerce Bank. As outlined in the project Requirements Document, this system will allow automation of some file processing, add automatic auditing and make cases visible. This product will make the case management process more effective and result in saved costs for Commerce Bank. The testing activities discussed in this document will verify that the software for the ACH Case Tracking meets the needs of the customer by verifying that the requirements for this system, as outlined in the Requirements Document, are met.

The testing routine will test the ACH Case Tracking system’s response to the function on the menu, which is Dashboard, audit log, cases, my cases, import/export, and create cases, also logging in to front-end/back-end websites and posting it to the front-end website. The results of this testing procedure will enable the creators of this system to gauge project success as outlined in the Macro Project Plan.

# Terminology

Throughout this document the terms ACH ops, Call center, provider, system, site and username/password will be used frequently therefore, formal definitions will be given.

ACH Ops-ACH Operators who have the permission to create cases, edit cases, and view cases.

Call­Center ­- Answer phone calls and then create cases in the system. People from Call­Center also have permission to view cases.

DNE ­ - Death notification entry

NACHA ­ - National Automated Clearing House Association

DCN ­ - Document Control Number

CRF ­ - Credit Research Foundation

SLA ­ - Service level  agreement

POA ­  - Power of attorney

DDA ­ - Direct Deposit Account

# Items Tested

Items that will be tested during the testing phase as laid out by the Macro Project Plan will be but are not limited to:

Ability for an individual to log in via username and password and succeed the authentication

Test Case: 2.0

Ability for ACH/call center to login via username and password and view/choose a case

Test Case: 2.1

Ability for an admin to login via username and password and delete a case Test Case: 2.2

Ability for ACH ops and call center to login via username and password to the back-end system and create/view case from the front-end site.

Test Case: 2.3

Ability to search (by keyword) through search bar to view desired cases.

Test Case: 2.4

Ability to add/remove users of the system (back-end) as needed

Test case: 2.5

Ability to import NACHA file and read it.

Test Case: 2.6

# Items Not Tested

There are features that will not be included in the current testing procedure. This does not mean that these features will not be implemented, but that they have not been implemented and are not available for testing. Those features include but are not limited to:

* Process Nacha file function
* Querying the Database
* Administrative Features that include editing a case, duplicate amount to another check, edit cases.

# Approach

The overall method to this testing procedure is manual system testing. Each test case created will have a direct link to the requirements as laid out in the Requirements Document. Test cases that include similar Feature methods will be tested together. Examples of these features include logging in to view case, logging in to create a case and/or logging in to the back-end website. Each test case will test the security feature with invalid and valid data (usernames and passwords) to ensure that user requirement of valid users for these features is met and be able to view/create cases.

The features that specify going through the search bar searching by keyword will be tested together but in separate test cases. Searching by keyword through search bar will verify the system’s ability to produce results that an individual can validate visually.

Manual system testing will continue throughout the second and third iteration of the project. For each iteration, both old and newly implemented features will be tested. Adding new features or functionality can sometimes interfere with the functionality of old features and to ensure product/project success, all features implemented should function as intended throughout the life of the software.

# Item Pass/Fail Criteria

The minimum requirements for this software system were laid out in the Requirements Document and the Macro Project Plan outlined what the creators of the software considered project success.

Implemented features that meet the requirements as determined by the customer, meaning the feature does what the user wants it to do with very little difficulty, passes the testing procedure. Difficulty, as used here, is determined by user comprehension and user ability to use the feature with little to no training.

Features that contain major defects will fail the testing procedure and will be documented via an incident report and turned over to the developer for investigation and revision.

# Test Deliverables

In addition to the Test Plan, other test deliverables include the Test Specification which outlines the specific test cases and expected results of each test, and Test reports which is comprised of Incidents, Defects and Changes.

# Testing Tasks

The following list the testing deliverables and the activities required to produce the deliverable.

|  |  |
| --- | --- |
| **Deliverables** | **Activities** |
| **Test Plan** | * Analyze Requirements for System Features * Determine Testable/Non-Testable Features * Develop Approach/Method for testing * Determine Task and Estimate Efforts * Create a Case for Testing |
| **Test Specifications** | * Analyze Requirements * Define Test Cases for Testable Features as Outlined by the Test Plan |
| **Test reports** | * Implement Test Cases as Outlined by the Test Specifications * Document Incidents and Defects * Determine Severity of Incidents and Defects * Determine Changes that Need to be Made to System * Document and Submit Change Request to Developer |