Check Exception Automation (CEA) System

Aashima Arora
Dakota State University

Follow this and additional works at: https://scholar.dsu.edu/theses
Part of the Databases and Information Systems Commons

Recommended Citation
https://scholar.dsu.edu/theses/324

This Thesis is brought to you for free and open access by Beadle Scholar. It has been accepted for inclusion in Masters Theses & Doctoral Dissertations by an authorized administrator of Beadle Scholar. For more information, please contact repository@dsu.edu.
Check Exception Automation (CEA) System

A graduate project submitted to Dakota State University in partial fulfillment of the requirements for the degree of

Master of Science

In

Information Systems

By

Aashima Arora

Project Committee:

Dr. Ronghua Shan
Dr. William Figg
Dr. Stephan Krebsbach
We certify that we have read this project and that, in our opinion, it is satisfactory in scope and quality as a project for the degree of Master of Science in Information Systems.

Student Name: Aashima Arora

Master’s Project Title: Check Exception Automation (CEA) System

Faculty supervisor: Raghuraman Date: 4/7/15

Committee member: Date: 4/7/15

Committee member: Date: 4/7/15
ACKNOWLEDGMENT

I express my sincere appreciation to Dr. Omar El-Gayar and Dr. William Figg who provided me the proper guidance and knowledge to understand the project management and project life cycle which helped me in understanding the overall picture of the information system from business point of view as well as from development point of view. It is my pleasure to say that Dr. Omar encouraged me and helped to implement the project models in real time. And Dr. William Figg provided me the knowledge how to manage the project timelines, team communications and relations.

I wish to express my sincere appreciation to Dr. Ronghua Shan and Dr. Stephan Krebsbach who put their valuable experienced time and knowledge to make me understand and implement the highly complex transactional database management system. Dr. Shan taught me and nourishes my skills to create and manage enterprise database management system using Oracle and Dr. Krebsbach took my knowledge to another level by providing me complex transactional database management skills.

I express my gratitude to Dr. Chris Olson, Dr. Richard Christoph, Dr. Ahmad Al-Omari and Dr. Jun Liu for their support and extreme depth of knowledge by which I could implement the necessary building blocks for my project. I am very thankful to Dr. Richard Christoph who provided me the depth knowledge of web application front end development.

I am very thankful to my project manager Kay E. Falli, my technical lead Scott W. Ayres and my whole team who gave me this opportunity to work on and supported me to complete my graduation project successfully.

I express my respect to my parents and my family who supported me mentally, financially and emotionally to complete my graduation and achieve my goals.

I express my worship to Almighty GOD for providing me such a good path and grace to build my carrier.

Aashima Arora
ABSTRACT

Check Exception Automation (CEA) System is a web application which provides the facility to create and resolve the check exception that occurs after the check is successfully deposited to the American Bank. Checks can be deposited via different channels such as ATM, Cash Vault (which is a kind of check dropping box) and via mobile. There are a number of exceptions that may occur after depositing the check such as missing signature, wrong amount, altered check, duplicate check and missing amount. The older check exception resolution process was manual process. The goal of this project is to automate the check resolution process with no or very little human involvement. This project is implemented using all the advanced technologies such as Java 1.7, Spring 3.1, Spring Security, Apache CXF web services and JQuery. This documentation describes the project plan which explains the requirements with timelines and the methodology used for project life cycle. It also briefly explains about the meeting plans and team building activities during the project development. Furthermore, it explains the problems existed in older system and the benefits expected from the new system. Moreover, this document also covers the basic technical as well as architectural aspects of the project. The project is implemented on agile methodology which is different than traditional waterfall model. This document also explains the interaction strategy and interaction flow among the development teams and business partners.
DECLARATION

I hereby certify that this project constitutes my own product, that where the language of others is set forth, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions or writings of another.

I declare that the project describes original work that has not previously been presented for the award of any other degree of any institution.

Signed,

Aashima Arora

Aashima Arora
TABLE OF CONTENTS

PROJECT APPROVAL FORM ..................................................................................................... ii
Acknowledgment ........................................................................................................................... iii
ABSTRACT ................................................................................................................................... iv
Declaration ...................................................................................................................................... v
TABLE OF CONTENTS ............................................................................................................... vi
INTRODUCTION .......................................................................................................................... 1
   Older System ........................................................................................................................... 1
   Background of the Problem .................................................................................................... 1
      High check resolution time ................................................................................................. 2
      Customer dissatisfaction ...................................................................................................... 2
      Expensive ............................................................................................................................ 2
   Check Resolution status ........................................................................................................ 3
   Statement of the Problem ........................................................................................................ 3
   New System ........................................................................................................................... 3
   Objective of the Project .......................................................................................................... 4
      Faster check resolution ....................................................................................................... 4
      Cutting the overhead cost ................................................................................................... 4
      Customer satisfaction ......................................................................................................... 4
   Project Name ........................................................................................................................... 5
   Organization Profile ................................................................................................................ 5
   Business Need ......................................................................................................................... 5
   Scope of the Project ................................................................................................................ 6
   Feasibility Study ...................................................................................................................... 6
      Technical Feasibility ............................................................................................................ 7
      Economic Feasibility .......................................................................................................... 7
      Operational Feasibility ........................................................................................................ 7
   Project Milestone List ............................................................................................................. 8
SYSTEM ANALYSIS .................................................................................................................... 10
   Technical Description of the Project ..................................................................................... 10
   Business Requirements .......................................................................................................... 10
      Priority ................................................................................................................................. 10
      Requirement Type .............................................................................................................. 11
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>11</td>
</tr>
<tr>
<td>Quality Management Plan</td>
<td>18</td>
</tr>
<tr>
<td>Introduction</td>
<td>18</td>
</tr>
<tr>
<td>Process Quality</td>
<td>19</td>
</tr>
<tr>
<td>Product Quality</td>
<td>19</td>
</tr>
<tr>
<td>Quality Tools</td>
<td>20</td>
</tr>
<tr>
<td>Jenkins Dashboard</td>
<td>21</td>
</tr>
<tr>
<td>Sonar Dashboard</td>
<td>22</td>
</tr>
<tr>
<td>Quality Planning</td>
<td>22</td>
</tr>
<tr>
<td>Quality Standards</td>
<td>23</td>
</tr>
<tr>
<td>Quality Assurance</td>
<td>24</td>
</tr>
<tr>
<td>Quality Control</td>
<td>25</td>
</tr>
<tr>
<td>Communication Management Plan</td>
<td>26</td>
</tr>
<tr>
<td>Introduction</td>
<td>26</td>
</tr>
<tr>
<td>Scrum Meeting</td>
<td>26</td>
</tr>
<tr>
<td>Retrospective Meeting</td>
<td>27</td>
</tr>
<tr>
<td>Scrum of Scrum Meeting</td>
<td>27</td>
</tr>
<tr>
<td>CEA All Hands Meeting</td>
<td>28</td>
</tr>
<tr>
<td>Team Lunch</td>
<td>28</td>
</tr>
<tr>
<td>Release Meeting</td>
<td>28</td>
</tr>
<tr>
<td>Acceptance Plan</td>
<td>28</td>
</tr>
<tr>
<td>Purpose</td>
<td>28</td>
</tr>
<tr>
<td>Scope</td>
<td>29</td>
</tr>
<tr>
<td>Output</td>
<td>29</td>
</tr>
<tr>
<td>Testing</td>
<td>30</td>
</tr>
<tr>
<td>General Testing Approach</td>
<td>30</td>
</tr>
<tr>
<td>Acceptance Testing Approach</td>
<td>30</td>
</tr>
<tr>
<td>Development Unit Testing</td>
<td>31</td>
</tr>
<tr>
<td>Functional Testing</td>
<td>33</td>
</tr>
<tr>
<td>System Testing</td>
<td>34</td>
</tr>
<tr>
<td>Acceptance Plan Responsibilities</td>
<td>35</td>
</tr>
<tr>
<td>Test Case Report</td>
<td>36</td>
</tr>
<tr>
<td>SYSTEM DESIGN</td>
<td>38</td>
</tr>
<tr>
<td>Project Methodology</td>
<td>38</td>
</tr>
</tbody>
</table>
List of Tables

Table 1 Project Milestones.............................................................................................................. 8
Table 2 Business Requirements 1 .................................................................................................... 11
Table 3 Business Requirements 2 .................................................................................................. 12
Table 4 Business Requirements 3 .................................................................................................. 13
Table 5 Business Requirements 4 .................................................................................................. 14
Table 6 Business Requirements 5 .................................................................................................. 14
Table 7 Business Requirements 6 .................................................................................................. 15
Table 8 Business Requirements 7 .................................................................................................. 15
Table 9 Business Requirements 8 .................................................................................................. 16
Table 10 Business Requirements 9 ................................................................................................. 16
Table 11 Business Requirements 10 .............................................................................................. 17
Table 12 Business Requirements 11 .............................................................................................. 17
Table 13 Quality Tools .................................................................................................................. 20
Table 14 Test Case Report.............................................................................................................. 36
Table 15 Project Implementation Technologies ............................................................................. 44
Table 16 CEA System Scope ......................................................................................................... 51
Table 17 Gantt chart...................................................................................................................... 118
List of Figures

Figure 1 Jenkins Dashboard .......................................................................................................... 21
Figure 2 Sonar Dashboard ............................................................................................................ 22
Figure 3 Agile Methodology Workflow ....................................................................................... 38
Figure 4 Agile Methodology Workflow ....................................................................................... 39
Figure 5 Agile Development ......................................................................................................... 40
Figure 6 MVC Architecture .......................................................................................................... 44
Figure 7 Spring Framework Architecture .................................................................................... 47
Figure 8 Spring Transaction Architecture .................................................................................. 49
Figure 9 CEA System Higher Level Design .............................................................................. 50
Figure 10 Use Case Diagram ...................................................................................................... 76
Figure 11 Log in Sequence Diagram .......................................................................................... 77
Figure 12 Login Collaboration Diagram ..................................................................................... 78
Figure 13 Login Class Diagram .................................................................................................. 79
Figure 14 Create Check Exception Sequence Diagram ............................................................... 79
Figure 15 Create Check Exception Collaboration Diagram ......................................................... 80
Figure 16 Create Check Exception Class Diagram ..................................................................... 81
Figure 17 View Check Exception Sequence Diagram ................................................................. 81
Figure 18 View Check Exception Collaboration Diagram ........................................................... 82
Figure 19 Maintain Check Exception Sequence Diagram ........................................................... 83
Figure 20 Maintain Check Exception Collaboration Diagram ....................................................... 84
Figure 21 Maintain Long Term Storage Location Sequence Diagram ......................................... 85
Figure 22 Maintain Long Term Storage Location Class Diagram ................................................ 85
Figure 23 Maintain Long Term Storage Location Collaboration Diagram ........................................ 86
Figure 24 Maintain BC Storage Location Sequence Diagram.................................................. 87
Figure 25 Maintain BC Storage Location Collaboration Diagram ........................................... 88
Figure 26 Maintain Check Exception Configuration Sequence Diagram .................................... 89
Figure 27 Maintain Check Exception Configuration Collaboration Diagram............................. 90
Figure 28 Maintain Check Exception Configuration Class Diagram ........................................ 90
Figure 29 CEA System User Management Sequence Diagram............................................... 91
Figure 30 CEA System User Management Collaboration Diagram ......................................... 92
Figure 31 CEA System User Management Class Diagram ..................................................... 92
Figure 32 Respond Check Exception Sequence Diagram ...................................................... 93
Figure 33 Respond Check Exception Collaboration Diagram ................................................ 94
Figure 34 Respond Check Exception Class Diagram ............................................................ 94
Figure 35 Adjust Customer Sequence Diagram ...................................................................... 95
Figure 36 Adjust Customer Class Diagram ........................................................................... 95
Figure 37 Adjust Customer Collaboration Diagram ................................................................ 96
Figure 41 Editing GSS Check Exception ................................................................................ 103
Figure 42 Check Exception Originator Landing Page ............................................................ 104
CHAPTER 1

INTRODUCTION

Older System

The American Bank did not have the online system to create or resolve the check exceptions. The whole process was manual process, in which when a commercial/ consumer customer deposits the check via any channel such as ATM, mobile, cash vault or at cashier’s office and later bank finds any problem such as missing signature, altered check, duplicate check or missing check amount then all the checks with the exceptions are mailed to the associated banking center (BC) site and a check resolver analyst manually analyzes the checks and resolves the check, after that all the checks are put into the bags and then all the bags are mailed to the bank long term paper storage location (BLTPS) site where all these bags are kept into the trays and checks are stored for next 45 days. Within 45 days, if any query occurs on any check then a check analyst at bank’s long term paper storage location (BLTPS) site searches for the tray, and then within the tray searches for the bag and then goes through each and every check in that bag and reads the check information to match the right check and finally tries to resolve the check. If necessary, analyst can mail the check back to the originating banking center for resolution. After 45 days, all the checks are discarded and all the empty bags are kept for the record.

Background of the Problem

The older system was fully manual to resolve the check exception. It was totally inefficient and very expensive process. To even resolve a single check at any banking center location, the complete process has to follow. The following are the major problems exists in older system.
High check resolution time

In older system check resolution time was quite high. The minimum time to resolve the check was 3 days. And resolution of after inquiry can take extra 2 days to 6 days. Check is a very crucial part of the business and for a customer if it get delays it can have direct impact on the customer. Suppose a customer has deposited his pay check and he urgently needs money to pay his bill or debt and it got exception, if check resolution takes 3 days to 6 days it can go very bad with that customer though there might not be customer’s fault at all.

Customer dissatisfaction

High resolution time in financial matters is certainly can cause the customer dissatisfaction. Every bank customer expects that he get the money in his account as soon as possible. If the customer cannot get the money on time, it can cause worry or sometimes it can cause serious problem in his life because customer might be dependent on the check money. The America Bank had high probability to lose customers because of the older system.

Expensive

Older system was quite expensive for the bank. Even if there was only one check with error at any ATM, it had to go through the whole process of resolution. First it had to be carried to the banking center (BC) site then to the bank longer term paper storage (BLTPS). Postal services charged the same amount because charges were based on bags and even if there was only one check it had to go into the bag. The American bank was paying around $50 million per year to the postal services. And check storage cost for next 45 days was approximately $20 million per year. So the older system was costing around at least $70 million per year only for checks resolutions.
Check Resolution status

As the older system was manual, there was no way to know the status of the check exception. Bank associate had to call to the banking center (BC) site or bank long term paper storage (BLTPS) site for checking the resolution status and problem. So if any customer called to check the status of the check resolution, it was time consuming process to just find out the status because there was no online system a bank associate can go and look to the check exception.

Statement of the Problem

Banking business totally depends on the customers. If bank loses the customer it will go bankrupt at some point of time. These days competition is very high in the market. Competitor banks are attracting customers with their good customer services and their interesting benefits such as 2% cashback, no account charges. The American bank has commercial customer also which includes bigger retail industries such as Walmart, Costco, Sam’s Club as well as bigger businesses such as stocks. Most of the transactions are check based transaction and with the older system there were so many problems a customer can face and because customers were not satisfied they stopped their business with American Bank.

New System

To resolve the issues with the older system, the new solution is to build a Check Exception Automation (CEA) web application which provides the functionality to create and resolve the check exception online. So when check exception occurs, a check analyst at the banking center can open a check exception online with this new automated system. This online system can directly access the check images from the America Bank’s central check image system. There is no need to keep the hard copy of check and no need to mail the check to bank long term paper storage site.
Any analyst can directly access the web application and can get the status as well as whole worked history on any check exception. He can adjust the customer account with the disputed or error amount directly with the help of CEA application online and after the check is resolved, a notification is automatically generated to the customer.

**Objective of the Project**

The objective of the project is to replace the manual check resolution process with the online Check Exception Automation System. This automated system provides the online capability to the process and thus helps in providing high values to the banking business as well as to the customer by resolving the checks quicker and reducing the overhead cost. The new automated system provides the following benefits:

**Faster check resolution**

Online check resolution process reduces the check resolution time from average 4 days to 1 day which is a big improvement.

**Cutting the overhead cost**

This project fully eliminates the need of mailing the checks by providing the analyst with the online check image system as an analyst has access to the check image online which saves bank approximately $70 million per year.

**Customer satisfaction**

Customer satisfaction is the success key of any business. This automated system provides faster check resolution. Also, customers can come to know the status of check resolution process very
quickly because an analyst can check the status online in seconds and give the status details to the customers.

**Project Name**

Business name of the project : Check Exception Automation (CEA) System  
Development name of the project : Exception Management

**Organization Profile**

The American Bank is one of the top notch US bank and is world’s leading wealth management company having history of more than 200 years for providing sales and services to clients in more than 40 countries in the world. The American bank serves individual consumer, financial institutions, government, small, middle market and large businesses. It also provides other risk management, asset management products and financial services. Company has approximately 49 million small businesses and consumer relationships. The American Bank has 16,200 ATMs and 5100 retail banking offices around the world. American Bank provides excellent online easy to use services, have approximately 15 million mobile users and 30 million active online users.

**Business Need**

The main goal for the initiation of the project is to eliminate the check resolution manual process completely and provide the easy to use online capabilities to create and resolve the checks with very less amount of time and to cut down the overhead mailing and storage costs.
**Scope of the Project**

Check Exception Automation System is completely new web application. Following is the broader scope of the project:

- Creating the check exception online
- Resolving the check exception online
- Access to the check images from Check Image system
- Adjusting the customer directly from CEA system by debiting or crediting the money
- Notifying the customer automatically after check resolution
- Access to the check exception creation and resolution work history
- Assigning the right status to the check exception
- Pulling the customer and customer’s account information and attach them with the exception.
- Posting the customer’s adjustment amount to America Bank financial system.

**Feasibility Study**

Feasibility study focuses on providing answers to the essential questions such as given project idea is worth to implement, should we proceed with the proposed idea etc. Assessing the feasibility of Check Exception Automation (CEA) System is very crucial step for American Bank Business partners as they have to determine the chances of success of the project. Operational, Economic and Technical feasibility analysis is used to determine this new project’s feasibility.
Technical Feasibility

Check Exception Automation (CEA) System is new system and will be using the existing bank web services to pull the data and will determine the required action. All web services are in place and bank has been using them for past many years. Proposed project will be using the latest well known technologies and bank already have the expertise in the technologies that are going to be used in this project. The process of setting up the technical environment, servers and getting the technical tools are already in place so technical project does not seem to have any sort of risk.

Economic Feasibility

This project is internal bank application and will be accessible by bank employees only and there might not be the need of expensive production servers or security setup. Almost all web services are in place, so there will not be any added cost of building them and last but not least, bank already has the necessary skills to build the project. So economically project is ranked a low cost project. Also, by this project bank is going to save $70 million per year thus project is very strong economically.

Operational Feasibility

The American Bank has very skilled employees who resolved the check exceptions on day to day basis and they are much trained employees. This new project will have very high concentration on user experience and user does not need to go for intense training because system will be using very minimal amount of user inputs and will have the availability of everything within three clicks. So at most, 3 days training by any of project team member such as technical analyst can trained the bank employees, so operationally also, this new project is feasible.
Project Milestone List

The below chart list the major milestones for Check Exception Automation (CEA) System. With these major milestones, there is highly likely chances project will have minor milestones during the development of the project. If in any case, project milestone is not hitting on time then project manager should be notified immediately so that right decision can be made on time and by knowing the cause, this kind of situation can be prevented in future. These milestones should be discussed with the team so that team is aware of deadlines and can plan their work accordingly.

\textit{Table 1 Project Milestones}

<table>
<thead>
<tr>
<th>Milestone</th>
<th>Description</th>
<th>Target Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Start</td>
<td>Project Start Date</td>
<td>Aug 18, 2014</td>
</tr>
<tr>
<td>Project Charter Approval</td>
<td>Project Charter Approved by Project Sponsor</td>
<td>Aug 28, 2014</td>
</tr>
<tr>
<td>Project Initiation Complete</td>
<td>Project Initiation Phase Complete</td>
<td>Aug 30, 2014</td>
</tr>
<tr>
<td>Development Plan Approval</td>
<td>Development plan approval by business</td>
<td>Oct 15, 2014</td>
</tr>
<tr>
<td>Test Plan Approval</td>
<td>Test plan approved by business</td>
<td>Oct 20, 2014</td>
</tr>
<tr>
<td>Maintenance Plan Approval</td>
<td>Maintenance plan approved by business</td>
<td>Oct 27, 2014</td>
</tr>
<tr>
<td>Acceptance Criteria Approval</td>
<td>Acceptance criteria approval by business</td>
<td>Oct 30, 2014</td>
</tr>
<tr>
<td>Architecture Design Complete and Approval</td>
<td>Hardware and software architecture design completion and approval by bank architectures</td>
<td>Nov 12, 2014</td>
</tr>
<tr>
<td>Product Design Complete and Approval</td>
<td>Product design completion and approval</td>
<td>Nov 30, 2014</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
<td>Date</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
</tr>
<tr>
<td>Test Plan Complete and Approval</td>
<td>Test plan completion and approval by Testing Manager</td>
<td>Dec 10, 2014</td>
</tr>
<tr>
<td>Requirement Grooming with Development Team</td>
<td>Discussing the business requirements with the team</td>
<td>Dec 30, 2014</td>
</tr>
<tr>
<td>Development Phase 1 Start</td>
<td>Development Phase 1 Start Date</td>
<td>Jan 1, 2015</td>
</tr>
<tr>
<td>Development Phase 1 End</td>
<td>Development Phase 1 Completed</td>
<td>Apr 30, 2015</td>
</tr>
<tr>
<td>Regression Testing End</td>
<td>QA Regression Testing End</td>
<td>May 12, 2015</td>
</tr>
<tr>
<td>Code Push To UAT</td>
<td>Pushing the Code to UAT environment</td>
<td>May 13, 2015</td>
</tr>
<tr>
<td>UAT Testing Start</td>
<td>User experience testing start</td>
<td>May 15, 2015</td>
</tr>
<tr>
<td>UAT Testing End</td>
<td>User experience testing end</td>
<td>May 27, 2015</td>
</tr>
<tr>
<td>Pre-Production Deployment</td>
<td>Deployment of application to pre-production</td>
<td>May 30, 2015</td>
</tr>
<tr>
<td>Pre-Production UAT Testing Start</td>
<td>UAT testing in pre-production environment start</td>
<td>Jun 2, 2015</td>
</tr>
<tr>
<td>Pre-Production UAT Testing Complete</td>
<td>UAT testing in pre-production environment complete</td>
<td>Jun 12, 2015</td>
</tr>
<tr>
<td>Release 1.0</td>
<td>Deployment of application to production</td>
<td>June 14, 2015</td>
</tr>
<tr>
<td>Project Phase 1 Complete</td>
<td>Project first major release complete</td>
<td>June 16, 2015</td>
</tr>
</tbody>
</table>
CHAPTER 2

SYSTEM ANALYSIS

Technical Description of the Project

Check Exception Automation (CEA) System is a web application which can be accessed by internal American Bank employees only. Bank employee at the post of check analyst can access the system via browser and can create a check exception online. While creating the check exception, the check analyst is able to pull the customer’s information online from the bank’s central database as well as the rejected check images from check image system. Check analyst can resolve the check exception by debiting or crediting the money from their bank accounts with the help of CEA system and then can post the transaction to the bank Financial Deposit System (FDS). While resolving the check exception, customer is notified automatically about the transaction on their accounts. After resolving the check exception, analyst can close the exception and update the status of the exception as “resolved”.

Business Requirements

Priority

Each functionality is given a priority based on its criticality as well as business requirements.

Critical: Functionality must go live as soon as possible or business requirements having core functionality.

Essential: Business requirements which serves a part of the critical functionality.

Nice to Have: Business requirements such as view changes, user experience
Requirement Type

Requirement type tells about if the given requirement is for security related, user interface related or relevant to interaction with other systems.

UI: User interface requirement
EI: Integration with external bank system
S: Bank security policy requirement
F: Functional requirement

Status

Status shows the progress of business requirements.

Not Started: Business requirement is not started yet
Ready for Development: Story is fully ready for development
In Progress: Business requirement is being developed or tested
Blocked: Business requirement cannot be worked on because of some internal or external issue.
Integration Testing: Functionality is under integration testing
Acceptance Testing: Functionality is under acceptance testing
Finished: Business requirement has finished all phases and completed.

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieve information for an ATM Check exception.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Business Requirement Description</th>
<th>As a check exception originator for ATM check exceptions, employee wants to enter minimal amount of information and rest of the information like check image, analyst details and customer details should get populated automatically.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acceptance Criteria</td>
<td>When analyst submits the check exception date, Check exception ATM ID and Sequence Number, system should retrieve the check details with check image.</td>
</tr>
</tbody>
</table>
Originator should be able to successfully submit the check exception request. If there is any error, error should be displayed on the check exception form and check exception request should be rejected.

**Validation**
- Check Exception Date: Mandatory with format mm/dd/yy.
- ATM ID: Mandatory and must be in database ATM table.
- Check Sequence Number: Mandatory and must be 11 digit.

**Priority**
- Critical

**Type**
- F

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Retrieve information for a Bank Center (BC) Check exception.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Requirement Description</td>
<td>As a check exception originator for BC check exceptions, employee wants to enter minimal amount of information and rest of the information such as check image, analyst details etc. should get populated automatically.</td>
</tr>
<tr>
<td>Acceptance Criteria</td>
<td>When analyst submits the check exception date and source Sequence Number, system should retrieve the check details with check image. Originator should be able to successfully submit the check exception request. If there is any error, error should be displayed on the check exception form and check exception request should be rejected.</td>
</tr>
<tr>
<td>Validation</td>
<td>Check Exception Date: Mandatory with format mm/dd/yy. Source Sequence Number: Mandatory and must be 11 digit.</td>
</tr>
<tr>
<td>Priority</td>
<td>Critical</td>
</tr>
<tr>
<td>Type</td>
<td>F</td>
</tr>
<tr>
<td>Status</td>
<td></td>
</tr>
<tr>
<td>Business Requirement</td>
<td>Selecting root-cause reason for check exception based on channel.</td>
</tr>
<tr>
<td>----------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Business Requirement Description</td>
<td>Check exception originator should be able to select the root-cause reason of the check exception based on all the channels.</td>
</tr>
<tr>
<td>Acceptance Criteria</td>
<td>When analyst select any channel like ATM or BC, analyst should be displayed with root cause reasons specific to that channel and analyst must select at least one of the root cause reasons.</td>
</tr>
</tbody>
</table>
| Validation | **ATM Root Cause Reason:**  
Altered Check  
Improper Endorsement  
Incomplete Check  
Missing Signature  
Payee not on Account  
Cash Out of Balance  
**Banking Center Root Cause Reason:**  
Altered Check  
Cash Out of Balance  
Improper Endorsement  
Incomplete Check  
Missing Signature  
Scanning Error  
Customer Error |
| Priority | Critical |
| Type | F |
| Status | |

*Table 4 Business Requirements 3*
### Table 5 Business Requirements 4

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Populating the check exception originator information at the time of check exception creation.</td>
<td></td>
</tr>
<tr>
<td>When check exception originator creates a check exception his/her details should be retrieved from bank’s employee central system and should populate in the check exception automatically. The following details should get populated: First Name, Last Name, Phone Number, Fax Number, Mail Code, Department Name, Street Address</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Acceptance Criteria</th>
<th>Acceptance Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>When check exception originator selects the create check exception, his details should get populated automatically and the details should be editable.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Validation</th>
<th>Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Name, Last Name: Maximum 255 characters, Not Null Phone/Fax Number: Maximum 15 digit, Format: 999-999-9999</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Priority</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Status</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 6 Business Requirements 5

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Populating the check information for ATM/ Banking Center check exception</td>
<td></td>
</tr>
<tr>
<td>Check details should be retrieved using web services and should be pre populated into the check exception form by providing only two inputs to that form- check capture date and check sequence number.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Acceptance Criteria</th>
<th>Acceptance Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>When as a check exception originator, I enter check capture date and check sequence number, inputs are added and check exception form gets populated.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Validation</th>
<th>Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>If external web services call fails, originator should be represented by appropriate message.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Priority</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Status</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Table 7 Business Requirements 6

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Business Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Assign check exception to the correct storage location</td>
</tr>
<tr>
<td></td>
<td>As originator, I want the check exception to be routed to the correct storage location so that request can be resolved.</td>
</tr>
<tr>
<td>Acceptance Criteria</td>
<td>ATM ID determines the storage location and the location is tied to the check exception.</td>
</tr>
<tr>
<td>Validation</td>
<td>ATM ID must be in ATM table otherwise location must be Unassigned.</td>
</tr>
<tr>
<td>Priority</td>
<td>Critical</td>
</tr>
<tr>
<td>Type</td>
<td>F</td>
</tr>
<tr>
<td>Status</td>
<td></td>
</tr>
</tbody>
</table>

### Table 8 Business Requirements 7

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Business Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Check exception responder view the check exception for resolution</td>
</tr>
<tr>
<td></td>
<td>As check exception responder, I want to view the check exception assigned to my storage location so that I can resolve them.</td>
</tr>
<tr>
<td>Acceptance Criteria</td>
<td>All check exception routed to check exception responder storage location should be on the responder landing page and all the exceptions should be sorted by date. Responder can view the individual check exception.</td>
</tr>
<tr>
<td>Validation</td>
<td>Check exception responder landing page should have check exceptions routed to his/ her storage location.</td>
</tr>
</tbody>
</table>
### Table 9 Business Requirements 8

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check Exception system can print the check exception</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Business Requirement Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>As a check exception responder I should be able to print the check exception assigned to my storage location</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Acceptance Criteria</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>I know I am done if I can print the check exception, check image and this activity should be logged.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Validation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check exception responder can print the check exception assigned to his/her storage location</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Priority</th>
<th>Critical</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>F</td>
</tr>
<tr>
<td>Status</td>
<td></td>
</tr>
</tbody>
</table>

### Table 10 Business Requirements 9

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check exception responder responds to the check exception</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Business Requirement Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check exception responder should be able to select the prepopulated response reasons and should be able to add comments to respond to the check exception</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Acceptance Criteria</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check responder can respond only one time</td>
<td></td>
</tr>
<tr>
<td>Responder can select only one response reason</td>
<td></td>
</tr>
<tr>
<td>Response Reasons:</td>
<td></td>
</tr>
<tr>
<td>Request processed</td>
<td></td>
</tr>
<tr>
<td>Request faxed</td>
<td></td>
</tr>
<tr>
<td>Request not located</td>
<td></td>
</tr>
<tr>
<td>Request not stored at this location</td>
<td></td>
</tr>
</tbody>
</table>
# Table 11 Business Requirements 10

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Requirement Description</th>
<th>Acceptance Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Check exception originator views all the check exception of his department mail code only.</td>
<td>Should be able to see all requests under working department mail code. Check exceptions list view should have sorting capabilities by each column.</td>
</tr>
</tbody>
</table>

**Validation**
- The default sorting should be by check exception creation date. One page can have maximum 50 check exception displayed. Check exception status should change to adjusted after resolution.

| Priority | Critical |

| Type | F |

| Status | |

# Table 12 Business Requirements 11

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Adjust alternate account and alternate delivery</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Business Requirement</th>
<th>Requirement Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Customer can be adjusted to his/her specified alternate account as well as can get notified to alternate delivery</td>
</tr>
</tbody>
</table>
### Acceptance Criteria

<table>
<thead>
<tr>
<th>Acceptance Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>If customer has specified alternate account for financial adjustment then alternate account should get adjusted.</td>
</tr>
<tr>
<td>If customer has specified alternate delivery then customer should be notified using alternate delivery such as email/ fax/ mail.</td>
</tr>
<tr>
<td>If customer has specified no delivery then customer should not get notified but notification should be saved in database.</td>
</tr>
</tbody>
</table>

### Validation

<table>
<thead>
<tr>
<th>Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alternate account must be a valid alternate account.</td>
</tr>
<tr>
<td>Alternate delivery must be a valid alternate delivery method.</td>
</tr>
</tbody>
</table>

### Priority

<table>
<thead>
<tr>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
</tr>
</tbody>
</table>

### Type

<table>
<thead>
<tr>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
</tr>
</tbody>
</table>

### Status

<table>
<thead>
<tr>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>

---

### Quality Management Plan

#### Introduction

Check Exception Automation (CEA) System project’s quality management plan is to manage the quality of the project from project planning to delivery and document the strategy and information which is essential for better manage and quality policies enforcement of the project. This document is created during the planning phase and it needs collaboration of project manager, sponsors and project team. Quality management plans states and documents the approach the CEA system team has been using throughout the project lifecycle from planning to product delivery. Quality management plan helps to manage the resources, cost and helps to reduce the waste, increase the project implementation time and it is the key component to build less error prone system which can be maintained and enhance easily. On high level, quality management plan for CEA system controls the quality for both process and product.
Process Quality

Process quality standards, requirements and tools are determined by CEA web application quality manager and project manager. There are restrictions to choose the tools to manage the process quality. Only American Bank licensed tools can be used. If there is critical need to use the tool which American Bank does not have the license than project manager needs to raise a request, need to describe the need for the tool and should wait until bank approves the request. Project can have its project specific process but they have to follow the bank standards, regulations and must get approved by the bank IT director. In order to measure process metrics, the iterative process is used throughout the project life cycle. It is the responsibility of quality manager to monitor the project process and to improve the process if there are problems in current processes.

Product Quality

The product quality requirements, standards and tools are determined by project manager with the co-operation of project technical leads and engineers. Product must follow the bank privacy policy, security and encoding standards to protect the user’s private data. Product can have its own product specific quality standards unless those standards do not breach the bank’s product quality standards. Product specific standards must be reviewed by bank’s IT quality department.

Quality management plan documents the following things to manage the project quality:

- Project’s quality policies
- Quality procedures
- Quality criteria for the CEA web application
- Team roles, responsibilities and authorities
## Quality Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mingle</td>
<td>To keep track of the business requirements as well as defects e.g. how much business functionality is done, how much work is pending and in what state.</td>
</tr>
<tr>
<td>Jenkins</td>
<td>Continuous integration tool to enforce the working development process within the development team.</td>
</tr>
<tr>
<td>Sonar</td>
<td>Code quality measurement tool, to maintain and enforce the code quality to minimize the risk of failure during the production.</td>
</tr>
<tr>
<td>Selenium</td>
<td>Automation web application testing tool to automate the functional integration testing to ensure the quality during the whole development life cycle of the CEA web application.</td>
</tr>
<tr>
<td>JIRA</td>
<td>To keep track the issues and resolution appears during the project implementation.</td>
</tr>
<tr>
<td>Quality Center</td>
<td>To keep track the CEA web application defects and defect status.</td>
</tr>
<tr>
<td>SharePoint</td>
<td>Central location to store all the project related documentation.</td>
</tr>
<tr>
<td>Clarity</td>
<td>Keep track the team resource hours, quality of work and their performance.</td>
</tr>
<tr>
<td>Crucible</td>
<td>Code review tool, enforces the code quality.</td>
</tr>
<tr>
<td>Check Style</td>
<td>To keep track of code formatting check style is run with every build and build is stopped if code does not met with the check style rules.</td>
</tr>
<tr>
<td>J-Meter</td>
<td>To check the application performance and response time J-Metter java tool is used at the end of each spring (2-weeks) to...</td>
</tr>
</tbody>
</table>
check the application performance. If application performance does not meet the criteria code must be refactored to improve the performance.

**Jenkins Dashboard**

![Jenkins Dashboard](image)

**Figure 1 Jenkins Dashboard**
Quality Planning

Quality planning documents the quality metrics, measures standards for project processes, product functionality, project deliverables, documentation, testing and how to satisfy these quality standards. Quality planning identifies the acceptance criteria for product performance, product quality and project deliverables.
Quality Standards

Business requirements quality standards

- Business requirements should not contain technical details and any solution recommendations. Requirement should be easy to understand and should be in plain text.
- Business analysts should expose the business requirements to the development team before proceeding to implementation, so that if there are any questions and concerns those can be identified and addressed early in the development phase.
- Business requirements must be prioritize, so that critical functionality can be built early. By this way, if project is running behind the schedule that can be handled easily without worrying too much as application has the critical functional behavior.

Development standards

- Unit testing code coverage is mandatory and minimum code coverage level should be 80%.
- Project should have the user interface integration testing for all user screens to make sure any new functionality does not break the existing functionality.
- Any new functionality must pass the code review, to identify any kind of bad code which can make application enhancement and maintenance difficult in future.
- Development code must not have any team member’s signature on it and all the code is the property of bank.
- Continuous integration build server should be active all the time, and should send auto generated emails to project team if build breaks.
- User experience (UX) reviews should be part of any functionality completion to make CEA web application user friendly and easy to access.
**Testing standards**

- For regression testing there should be a different server where testers can keep testing the existing and new functionalities without worrying about the current functionality which is in progress in development environment.
- Regression test must pass before pushing the code for user acceptance testing.
- For user acceptance testing there should be different server so that user acceptance testers keep testing existing and new functionalities without any dependencies.

**Performance standards**

- CEA web application must pass the bank performance standards e.g. web application server response time should not be more than 8 seconds.
- Application should pass the performance load with average number of users who currently are accessing the web application.

**Quality Assurance**

For Quality assurance of CEA web application, quality management of bank focuses on the performance, security, memory leak, privacy, application functional behavior, development and deployment process. Quality team processes quality check point every two weeks by gathering the statistics from the project teams and also collects the possible solutions from the project teams and chooses the best solution to improve the process. Product (CEA web Application) quality check point happens every month by bank quality team, security & privacy team and performance team. It the responsibility of performance team to make sure CEA web application is working well with existing user load and provides the response within the time bound. Security & privacy team is responsible to verify that CEA web application is following the bank standardized encryption
mechanism to encrypt sensitive data and fully met the bank privacy standards. Technical leads are responsible to do day to day code review and to detect the problems in the code at very early phase. Code review must pass before the code is pushed to integration server.

**Quality Control**

Quality control of CEA web application primarily focuses on application performance, code quality and application security.

- Performance team does hard check point every month with heavy test user load and measures the application’s performance and memory leakage. If performance team finds any performance issue than functionality pushes back to the technical leads to fix the issue. Application is not be pushed to next level until all performance issues get resolved.

- Unit testing code coverage must always be higher than the minimum threshold which is 80%. If code coverage gets down build breaks and it is the priority to fix the unit test code coverage. Test report code coverage is generated with every build and Sonar provides the statistics for code coverage to analyze.

- Integration tests and user acceptance tests must pass when any new functionality is added to the system. Integration and user acceptance testing happens iteratively with any new functionality in the system. Integration test engineers are responsible for integration automation. If integration tests fail, integration test engineers define the failure, priority level and raise the defect in quality center. Code does not migrate for user acceptance testing without passing the integration testing.

- Sonar analysis should run after every successful Jenkins’s build. If sonar analysis fails build also fails. There should always be zero major violation, zero critical violation and
zero cyclic dependencies. If any one of these sonar violation occurs, build fails and email is sent to the development team automatically. If there is any minor violation, build passes but it is the responsibility of the technical lead to ask the responsible person to fix those violations too.

- Business requirement cannot be changed at the time of development. Change in business requirements are tracked by Mingle.

**Communication Management Plan**

**Introduction**

Check Exception Automation project communication plan is to manage the communications among the on-shore team, off-shore team, business partners, quality team, security team, performance team, other external services team, development leads and managers. Communications among all the teams and within the team members is one of the key of project’s success. If team members and teams have better communication, it directly affects their performance and increase the product quality, as well as decrease the production time line. Poor communications can produce the confictions among the team members, and they might not work as whole team which will have very negative impact on project progress.

**Scrum Meeting**

Check Exception System have couple of on shore as well as off shore teams. Each team has their own stream of work. Each team have their own scrum meeting each day in which they discuss with the following things:

What you did yesterday?
What you will do today?

Is there any impediments?

By this way each team member gets very clear idea of the tasks progress and they get the chance to discuss if they have any road blocks or impediments.

**Retrospective Meeting**

Each team have their own retrospective meetings every two weeks. In this meeting each team member gets a chance to present and discuss his/her concerns to the business partners so that issues can be addressed. Each team member has the chance to discuss the following things:

What worked well?

What did not worked well?

What we can do to improve?

All team members can present their thoughts to above questions and team decides how to improve the process and eliminate the impediments.

**Scrum of Scrum Meeting**

Scrum of scrum meeting provides the chance to communicate across the team. Technical leads from each team attends the scrum of scrum and discusses about the cross team communications and supports. After the meeting, team leads discusses the annotated items with team members.
CEA All Hands Meeting

All hands meeting can be planned any time any day and can be set up just half an hour before. All team members require to join all hands. Usually project manager plans the all hands meeting to communicate about the critical things, team changes, and process changes.

Team Lunch

Every other week each team has team lunch and they get to gather at lunch. These lunch meetings helps to build the team relations and provides good coordination and team works as family.

Release Meeting

Release meeting are held after every release and business partners communicates to the development team about the product and user experience.

Acceptance Plan

Purpose

The acceptance plan document has the following purpose

- The acceptance plan document, documents the strategy for acceptance testing for CEA system and verifies that CEA web application meets all the business requirements as specified in the contract.
- To ensure that all the requirements for acceptance testing for CEA web application system are appropriately planned, reviewed and assessed within the overall project plan.
• And to demonstrate the testing processes to all stakeholders to make sure that all testing processes is being controlled and managed appropriately.

Scope

The scope of CEA Web Application system acceptance plan is to:

• Test all the views and verify the views with the user experience approved business artifact.
• Test the response time of each http request and verify that response time should not be greater than eight seconds.
• Test all the user screens on different mobile devices to verify that behavior of application is meeting the expectations.
• Test the functional behavior for each business requirement and verify that it meets the acceptance criteria mentioned in the business requirements.

Output

Acceptance plan should have the following outputs after completion of acceptance testing:

Defects which are related to the user view for example, position of a button or formatting on UI comes under category “Nice to have”.

Defects which are stopping user to complete his/ her work efficiently, but not blocking the user to complete the work comes under category “Essential”.

Defects where functionality does not meet the acceptance criteria and the functional behavior is blocking the user to complete the work comes under category “Critical”.
Testing

General Testing Approach

The general approach for Acceptance Testing for CEA Web Application is as follows:

- All tests should pass before pushing the application to next upper environment.
- Test cases and test scripts should be ready before functionality is ready to test.
- Test scripts should consider the business requirements as source of truth and thus should be written based on business requirements without any external assumption.

Acceptance Testing Approach

Acceptance testing is the responsibility of the acceptance test manager. The general approach for acceptance testing for CEA Web Application system is as follows:

- Acceptance testing is a continuous testing. New business stories is pushed to UAT server every two weeks. These stories’ test scripts is executed independently without integration testing.
- Every two weeks after the new stories acceptance testing and integration testing scripts are executed to test the integration of new functionality with existing functionality.
- Test data is provided for user acceptance testing before pushing the new functionality so that test scripts can be created beforehand.
- Acceptance testers can reach to internal development team directly in case of any test data related problems or to verify any functionality or defects.
- Acceptance tester can reach to web services team via team technical analyst to set the web services’ test data or to get the web services’ related test data.
• Defects are logged in the quality center and only acceptance testers can close the defects, after retesting and verifying the defect.

• Acceptance testing are allowed to do the manual testing also, in the cases where it is harder to catch all the testing scenarios from automated acceptance testing scripts.

• All views should passed through the acceptance testing and must match with the user’s experience design artifact.

• While retesting the defect, if acceptance tester finds any defect which is not fixed then that defect should be logged again in quality center, and it should be discussed personally with the development team to find the cause of failure.

• Two weeks before the release, acceptance testing environment is ceased and no new functionality is pushed.

• Release acceptance testing is undertaken against the Acceptance Test Plan which is prepared by CEA web application’s acceptance test manager.

• Acceptance is performed on user acceptance testing (UAT) environment

• At the time of code freeze, only defect’s fixes can be pushed and need technical lead’s approval.

• At the time of release maximum 3 cycles are possible, so all defect’s fixes should be pushed within first 2 cycles of testing.

**Development Unit Testing**

Development unit testing is responsibility of the project manager and technical lead. The general approach for development unit testing for CEA Web Application is as follows:
Development unit testing is undertaken by project developers. Developers establish the tests cases.

Test cases should include all possible happy path, negative path, errors and boundary condition for the module developers are developing.

Development team should follow the test driven development. Test should be written first, it should fail and then the functionality should be written, after that same tests should pass.

Project has a dedicated testing server called Continuous Integration (CI) server. CI box runs all test unit tests automatically whenever there is a commit in project repository.

Team can directly access the continuous integration build status from UI and can see the build status which are failed (red), unstable (yellow) and pass (green). Team can see the changes are committed in a particular build and can see the stack trace of the build.

Continuous build runs all the tests. In case of any test failure build fails and automatic notification is sent to the development team with build failure information.

In case of build failure, it must be claimed by the responsible team member if the team member is not present then technical lead should take the initiative. It is the responsibility and priority of whole team to fix the build and broken tests.

Unit test coverage is tracked by Sonar. Unit test coverage must always be more than 90%. If unit test coverage goes less than 90% build becomes unstable, and if unit test coverage goes less than 80% build breaks.

Development operations team is responsible for any CI server issues and actively work to maintain and keep CI server up and running.

In case of any defect, first unit test should be created to replicate the defect and then defect fix should be build and after defect fix test should pass.
Functional Testing

Functional testing is responsibility of the project manager and functional test manager. Functional testing is project functionality centered, and tests the end to end functional behavior. The general approach for functional testing for CEA Web Application is as follows:

- Functional unit testing is undertaken by functional test engineers. Functional test engineers establish the functional tests cases.
- Functional test cases include all possible happy path, negative path, errors and boundary conditions for the module developers are developing.
- Selenium automation tool is used for functional testing.
- Project has a dedicated functional testing server called Dev2 server. Dev2 server runs all functional tests twice a day.
- Functional testing defects are logged and tracked into Mingle, new business functionality must pass the functional testing for completeness. Only functional engineer can close the defect after retesting and verifying the defects.
- Development operation team is responsible to migrate the new code, database changes, files, tables and reference data to Dev2 server.
- Functional testing is a continuous testing throughout the development.
- Development operations team is responsible for any Dev2 server issues and actively works to maintain and keep Dev2 server up and running.
- Functional test completion is recorded and documented, and documentation is retained in project share point repository.
**System Testing**

System testing is responsibility of the project manager and a delegated system test manager. System testing is whole system centered, which does the integration testing as a whole system scope. The general approach for system testing for CEA Web Application is as follows:

- System testing is undertaken by system test engineers. System test engineers establish the system integration test scripts.
- Primary purpose of system testing is to ensure that the system works as expected as a whole.
- System test plan and scripts is prepared by system test engineers, and is executed by system the test engineers.
- Project has a dedicated system testing server called SIT server. System test engineers perform system testing on SIT, they have the SIT database access also so that they can directly insert data for testing purpose.
- System testing defects are logged and tracked into Quality Center, only system test engineer can close the defects after retesting and verifying the defects at SIT server.
- Development operation team is responsible to migrate the code, database changes, files, tables and reference data to SIT server.
- System testing is performed against live data. It is the responsibility of business and technical analyst to provide the live test data to system test engineers.
- System testing is continuous testing throughout the development.
- System operations team is responsible for any SIT server issues and actively works to maintain the keep SIT server up.
• System test completion is recorded, documented and documentation is retained in project share point repository.

• System testing includes high data volume testing with a high number of records and transactions to test the system behavior at high volume of data.

Acceptance Plan Responsibilities

This sections describes the roles and responsibilities of stakeholders and other project team members required for the acceptance plan process.

Acceptance Test Manager

Acceptance test manager has the following responsibility:

• Preparing the Acceptance Test Plan, Acceptance Test process and procedures. And arrange the necessary Acceptance Test resources and tools.

• Coordinating the Acceptance Testing activities with acceptance testing team and manages the Acceptance Tests.

• Establishing the Acceptance Test team’s training plan, if training is required.

• Managing and authorizing any modification, documentation, test data and Acceptance Test’s environment.

• Managing the communication with the stakeholders regarding any test related data, change requests and test problems.

• Reviewing Acceptance Test’s scripts, Test Cases to make sure scripts and test plans are met the business requirements as detailed in the contract.

• Reviewing the test results and to ensure that all test plans run on schedule.
• Administering the test related problems and raise the remedy ticket with high priority to resolve the problem if it is blocking the test scripts to run.

Operation Team

Operation team has the following responsibility with respect to Acceptance Test:

• Maintains the UAT server, if there is any problem in the environment it is the responsibility of development operation team to resolve the problem and keep running the UAT server.
• Migrates the application’s new functionalities, database tables, and test data to the UAT server and UAT database server and ensures the data security and data integrity.
• Maintains the database’s backup, provides database support services in case it is required.

Test Case Report

Test case report documents the test activities.

Table 14 Test Case Report

<table>
<thead>
<tr>
<th>Report Heading</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acceptance Test Case ID</td>
<td>Unique identifier of the acceptance test case.</td>
</tr>
<tr>
<td>Acceptance Testing Official initials</td>
<td>Initials of the acceptance tester who is running the test script.</td>
</tr>
<tr>
<td>Test Date</td>
<td>Test date when it was conducted</td>
</tr>
<tr>
<td>Test Case Description</td>
<td>Brief description of the test being conducted which explains what is the purpose of this test case and what are the expected results.</td>
</tr>
<tr>
<td>Data Used</td>
<td>Test data which is used to run this particular test script.</td>
</tr>
<tr>
<td><strong>Expected Result</strong></td>
<td>States what is the expected result for this test case.</td>
</tr>
<tr>
<td>---------------------</td>
<td>------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Actual Result</strong></td>
<td>Actual result which came up after executing the test case.</td>
</tr>
<tr>
<td><strong>Test Result Code</strong></td>
<td>Accept/ Not Tested/ Fail/ Partial Failure/ Can Not be Tested/ Test Data Unavailable.</td>
</tr>
<tr>
<td><strong>Test Review Number</strong></td>
<td>Test review number which is issued by Test Manager to review the test.</td>
</tr>
</tbody>
</table>
CHAPTER 3

SYSTEM DESIGN

Project Methodology

Agile Methodology

Check Exception Automation system is a web application which is used by American Bank’s employees to resolve the check exceptions. This work on CEA web application is totally new, and methodology like waterfall is not best fit for this web application, because user experience is one of the critical aspect of web application as well as there are highly likely chances of changing the requirements as project progresses. CEA web application is built based on good user experience and is capable to adapt with changing requirements. To achieve this goal, agile methodology is best fit for CEA web application. Development teams can deliver the product within 3 months, and bank employees can start using the product, and by their experience product can be enhanced at very early stage which produces a better quality and user friendly product.

![Figure 3 Agile Methodology Workflow](image-url)
Figure 4 Agile Methodology Workflow
Figure 5 Agile Development
Advantages of Agile Software Engineering Process

*Test Driven Development*

Agile uses the test driven development which means test should be created before the actual code. All the positive, negative and exception tests should be created before the actual functionality and then actual functionality should be implemented, and tests should pass after that. This helps to prevent any kind of bug in the code. If any developer changes anything in one piece of code and it breaks some other functionality, then at that moment executing tests helps developer to know what is being broken.

*Better Communication*

In Agile methodology, all team members usually sits very closely. Scrum master, business analyst, technical analyst, developers, user experience and testers all sits together. So if any team member has any question they can just ask by calling the other team members. Agile increases the communication among the team and nobody has to wait for the response because all team members are easily accessible.

*Better Project Progress Tracking*

In Agile, each team has scrum meeting every day, in which each team member discusses things he or she is working on, which provides very keen track of each task and project overall progress. If there is any impediments that can be resolved very early.

*Pair Programming*

Agile has terminology called pair programming. If there is any difficult task or any new team member joins the team then team can pair program, which helps to resolve a complex piece of
problem very quickly and a new team member can learn about the project very easily and quickly duration. This helps new team members to contribute in the project development from very first week of their joining.

**Quick Releases**

Agile methodology produces the full piece of functionality, which provides the value to end user very quickly and product can be release to production very quickly, as quickly as three months and further releases can be set every month. It provides immense benefit to the end user to put their thoughts about the product as well as business partners which help to improve the application as well as fix the bugs and missing requirements.

**Architectural Technology**

**MVC Architectural Technology**

Model View Controller (MVC) design pattern is best suits for CEA web application. MVC design patterns decouples the model, view and controller. Developers can work in parallel without waiting for functionality to be completed. MCV design pattern divides the functionality and can be divided into the expertise level groups. Spring MVC framework provides the dispatcher servlet which handles all the web requests and handles the request to appropriate controller by finding the mapping from handler mapper. After finishing the work controller hands out the model and view object to the dispatcher servlet and dispatcher servlet renders the specified view with populated model into that.
Model

Model layer of MVC architecture has all the data access related functionality which directly interacts with database. As CEA application is using Hibernate ORM framework, all hibernate related functionality is implemented under model layer, so hibernate experts can only concentrate on model layer.

View

View layer contains the all user interface related functionality. It contains all the java script pages (JSPs), java scripts (JS), JQuery and Cascading Style Sheet (CSS).

Controller

Controller is the communicator between the view and business service layer. Controller handles all the web requests and calls appropriate business logic to serve the request and redirects the view populated with model to the user.
Project Implementation Technologies

Check Exception Automation system is a web application which is built from scratch using all the latest technologies, which made the development faster and high quality product. The following are the technologies used to build CEA system.

Table 15 Project Implementation Technologies

<table>
<thead>
<tr>
<th>Technology Name</th>
<th>Technology Category</th>
<th>Technology Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JAVA 1.7</td>
<td>Backend Programming Language</td>
<td>JAVA is most widely used back end programming language and it is open source which increase its usability as well as quick bug tracking and fixing. CEA system</td>
</tr>
</tbody>
</table>
uses JAVA as programming language and whole code is written on JAVA.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring Framework</td>
<td>Spring is a java component and has immense amount of latest code inbuilt which provides faster development. Spring dependency injection (DI) provides very high performance to the application.</td>
</tr>
<tr>
<td>Oracle 12c</td>
<td>Oracle is one of the most powerful database server in the world. CEA application is highly transactional and heavy database interaction bank application. Oracle 12c has all the latest features such as views, locking and it can handle very heavy amount of data.</td>
</tr>
<tr>
<td>Hibernate</td>
<td>Hibernate is Object Relational Modeling framework. Hibernate provides a way to decouple the application with the database server and makes the interaction with database very easy. Hibernate provides the inbuilt caching mechanism which provides the higher performance.</td>
</tr>
<tr>
<td>J Boss</td>
<td>CEA application is deployed on J Boss application server. J Boss is open source server and has the capability to manage the container level highly complex transactional behavior.</td>
</tr>
<tr>
<td>SLF4J</td>
<td>SLF4J is logging interface which decouples the application with the underlying logging frameworks. If in future there is need to change the logging framework it can be changed without any concern because SLF4J façade takes care of that.</td>
</tr>
<tr>
<td>LOG4J</td>
<td>Logging Framework Implementation</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Spring Security</td>
<td>Web Application Security Framework</td>
</tr>
<tr>
<td>CXF Soap Web Services</td>
<td>Integration with Other Bank System</td>
</tr>
<tr>
<td>Spring Restful Web Services</td>
<td>Integration with CEA System with Other Bank Systems</td>
</tr>
<tr>
<td>JAXP</td>
<td>XML parser</td>
</tr>
<tr>
<td>JAXB</td>
<td>XML Binding</td>
</tr>
<tr>
<td>JSON</td>
<td>UI Communication</td>
</tr>
<tr>
<td>J Unit</td>
<td>Testing Framework</td>
</tr>
</tbody>
</table>
Easy Mock is a testing framework which provides a way to mock the dependencies, which helps to concentrate on a single piece of testing without worrying about outside dependencies.

Spring Framework Architecture

Spring framework has all the component inbuilt which is required to build a complete enterprise web application. CEA system is built on spring framework which reduces the implementation time and provides the inbuilt secure, high performance code ready to use.

Figure 7 Spring Framework Architecture
**Spring Transaction**

CEA system has very high transactional behavior as it is a bank application, transaction is very crucial and critical aspect of the CEA application. Spring provides very easy way to make any functional behavior transactional just annotating the method with transactional annotation.

**What does spring transaction provides**

- Opening the DB connection
- Closing the DB connection
- Opening the transaction
- Closing transaction
- Rolling back or committing the transaction

**Spring transaction workflow**

When `@Transaction` annotation applies on a method implementation (`@Transaction` annotation never applies on interface method) then spring creates a proxy for that class and any call to that method goes via proxy. So when call is already in the class via passing the proxy means call have access to the method.
Figure 8 Spring Transaction Architecture

CEA System Higher Level Design

CEA system workflow starts from user browser. User requests a resource via URL then request goes to the application deployment descriptor which hands out the request to spring security which determines the access and authorization of the user and resource. After successful authentication and authorization spring security hands out the request to spring dispatcher servlet which finds the correct controller method for the given URL and invokes the controller. Controller invokes the business logic under service layer and service layer calls the data access layer. Data access layer makes the connection with database server and retrieves the data and hands out the result to service. Service invokes the business logic if there is needed and hands out the result to controller and then controller renders the success page or error page based on the response. Higher layer does know about its lower layer and lower layer does not have any idea about its higher layer for example...
data access layer does not have any idea about service layer and service layer cannot be called by data access layer. Controller layer cannot be called by service layer. It decouples the application layer from each other.

Figure 9 CEA System Higher Level Design
Use Cases Description

Check Exception Automation (CEA) System Scope

Table 16 CEA System Scope

<table>
<thead>
<tr>
<th>Title</th>
<th>In</th>
<th>Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log In</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Create Check Exception</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>View Check Exception</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Maintain Check Exception</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Route Check Exception</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Maintain LTPS Storage Location</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Maintain Banking Center Storage Location</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Maintain Check Exception Configuration</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>CEA System User Management</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Respond Check Exception</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Adjust Customer</td>
<td>In</td>
<td></td>
</tr>
<tr>
<td>Print Check Exception</td>
<td>In</td>
<td></td>
</tr>
</tbody>
</table>

USE CASE 1 Log In

CHARACTERISTIC INFORMATION

Brief description

This use case verifies bank employee, bank manager, bank supervisor all denoted as CEA user, have a valid unique corporation directory ID and password to access the CEA system.

Primary Actor

CEA user

Secondary Actor

Unique User Management (UUM) System
Stakeholders
CEA user
CEA Manager
CEA Employee
CEA Supervisor

Trigger
A CEA user access the CEA system to log on to CEA system.

Preconditions
None

Guarantees

Success End Condition
The CEA user is successfully logged into the CEA system and allows access to restricted parts of the CEA system.

Failed End Condition
CEA system notifies user with login failure message

MAIN SUCCESS SCENARIO
1. CEA user provides the credential information bank unique ID and password to CEA system.
2a. CEA system authenticates and authorizes the CEA user.
3a. CEA system retrieves user details from UUM system
4. CEA system logs successful entry, read user's role and permissions.
5. CEA system reads the check exceptions based on role and displays the default landing page based on user's role with welcome message to CEA user.

EXTENSIONS
*a. CEA user decides to quit:
   *a1. The CEA system returns to login screen [fail].

2b. Invalid bank unique ID or invalid password:
   2b1. CEA system notifies CEA user with incorrect credentials warning message and requests valid bank unique ID and password [Repeat at 1].

2c. Valid bank unique ID, but invalid user's password (4 times):
   2c1. CEA system disables access to CEA user (temporarily) [fail].

3b. User not found:
   3b1. CEA system displays error message “Access Denied” [fail].
3c. UUM not available:
   3c1. CEA system displays warning message with error event id [step 4].

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bank Unique ID must be alphanumeric and must be exactly of 7 digits.</td>
</tr>
<tr>
<td>2</td>
<td>Password must be alphanumeric and between 6 to 12 characters.</td>
</tr>
<tr>
<td>3</td>
<td>ID and password must not contain any special characters.</td>
</tr>
<tr>
<td>4</td>
<td>CEA system user must have one of the role associated: Exception Originator, System Manager, Exception Supervisor, Exception Responder, Exception Analyst</td>
</tr>
<tr>
<td>5</td>
<td>User can have maximum 4 login attempts in case of valid user ID.</td>
</tr>
</tbody>
</table>

**USE CASE 2 Create Check Exception**

**CHARACTERISTIC INFORMATION**

**Brief description**

This use case allows the Exception Originator to create the ATM/Cash Vault/Banking Center check exception.

**Primary Actor**

Exception Originator

**Secondary Actor**

Check Info Central System

**Stakeholders**

Exception Originator

**Trigger**

Exception Originator receives the rejected check.

**Preconditions**

CEA Exception Originator is logged on

**Guarantees**

*Success End Condition*
CEA Exception Originator successfully create the check exception and CEA system successfully saves the changes.

**Failed End Condition**

Nothing happens

**MAIN SUCCESS SCENARIO**

1. The CEA system presents the ‘Create Check Exception’ menu tab.

2a. Exception Originator chooses ‘Create ATM Check Exception’.

3. CEA system displays the ‘Create ATM Check Exception Modal Window’ with editable form.

4. Exception Originator submits the ‘ATM ID’, Sequence Start and ‘Capture Date’.

5. CEA system validates the submitted form.

6a. CEA system presents the editable ‘Create Exception’ form with populated Exception Originator details and Check Exception Details.

7. Exception Originator fills the check exception information and submits the ‘Create Check Exception’ form.

8. CEA system validates the submitted form with business rules.

9a. System successfully saves the check exception and displays the success message with unique check exception number.

**EXTENSIONS**

*a. Exception Originator decides to quit:

   *a1. CEA system discards any intermediate data and exits [fail].

2b. Exception Originator chooses ‘Create BCRC’ Check Exception’:

   2b1. CEA system displays the ‘Create BCRC Check Exception Modal Window’ with editable form.

   2b2. Exception Originator submits the ‘SMI sequence Number’ and ‘Capture Date’ [step 5].

2c. Exception Originator chooses ‘Create Image Mark’ Check Exception’:

   2c1. CEA system displays the ‘Create Image Mark Check Exception Modal Window’ with editable form.

   2c2. Exception Originator submits the ‘SMI sequence Number’ and ‘Capture Date’ [step 5].
6b. Invalid data:

   6b1. CEA system displays the validation error with error message.
   6b2. Exception Originator submits the corrected information [repeat at 5].

6c. ATM ID/ SMI sequence does not found:

   6c1. CEA system displays the error message ‘Check information does not found’.
   6c2. Exception Originator provides the other check information [repeat at 5].

6d. External system failure:

   6d1. CEA system displays the network down error message with error event id
   6d2. Exception Originator submits the form again [repeat at 5].

9b. Invalid information:

   9b1. CEA system presents invalid error fields with error messages to Exception Originator.
   9b2. Exception Originator submits the corrected information [repeat at 8].

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Only CEA Exception Originator can create the check exception.</td>
</tr>
<tr>
<td>2</td>
<td>There must be ATM ID, SMI Sequence Number existed into the external system to proceed to the ‘Create Exception’ form from ‘Create Exception’ modal.</td>
</tr>
<tr>
<td>3</td>
<td>ATM ID must be alphanumeric and must be 12 digits.</td>
</tr>
<tr>
<td>4</td>
<td>SMI sequence number must be alphanumeric and must be of 11 digits.</td>
</tr>
<tr>
<td>5</td>
<td>Capture date must be in format MM/DD/YY</td>
</tr>
<tr>
<td>6</td>
<td>Capture date must not be the future date.</td>
</tr>
</tbody>
</table>
| 7      | Allowed Root Cause Reasons are:  
               | Altered Check  
               | Cash Out of Balance  
               | Check Appeared Blank  
               | Customer Error  
               | Improper Endorsement |
USE CASE 3 View check exception

CHARACTERISTIC INFORMATION

Brief description
This use case allows the Exception Originator, Exception Responder, System Manager, Exception Supervisor and Exception Analyst all denoted as CEA user to view the check exception.

Primary Actor
CEA User
Stakeholders
Exception Originator
Exception Responder
Exception Supervisor
Exception Analyst
System Manager

Trigger
CEA user clicks on Check Exception/ search for check exception.

Preconditions
CEA user is logged on

Guarantees

Success End Condition
CEA user successfully view the check exception details with exception history and comments.

Failed End Condition
Nothing happens
MAIN SUCCESS SCENARIO

1a. The CEA system presents the paginated check exceptions to the CEA user with search box.
2a. CEA user clicks on the desired check exception.
3a. CEA system displays the exception details with check image.

EXTENSIONS
*a. CEA user decides to quit:
   *a1. CEA system discards any intermediate data and exits [fail].

1b. If Exception Supervisor then CEA system displays the paginated check exceptions with check exception metrics:
2b. CEA user enters the ‘Check Exception ID’ to the search box:
   2b1a. CEA system searches for the requested ‘Check Exception ID’ [step 3].
   2b1b. Check Exception not found:
      CEA system displays the warning message and asks the CEA user to refine the search [repeat 2].
3b. CEA user selects the ‘View History’ tab:
   3b1. CEA system displays the all history event details of the check exception in Chronological order.
3b. CEA user selects the ‘View Comment’ tab:
   3b1. CEA system displays the all check exception comments with timestamp in Chronological order.

Business Rules

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Check Exception ID must be numeric</td>
</tr>
<tr>
<td>2</td>
<td>Default tab will be the Check Exception Detail tab</td>
</tr>
<tr>
<td>3</td>
<td>If no check image is available then ‘No Check Image’ must be displayed.</td>
</tr>
<tr>
<td>4</td>
<td>Check exception history must be in Chronological order.</td>
</tr>
<tr>
<td>5</td>
<td>Check exception comments must have the timestamp with it and must be in Chronological order.</td>
</tr>
</tbody>
</table>
USE CASE 4 Maintain check exception

CHARACTERISTIC INFORMATION

Brief description
This use case allows the Exception Originator to edit, close and resubmit the check exception.

Primary Actor
Exception Originator

Stakeholders
Exception Originator
Customer

Trigger
CEA Exception Originator clicks on check exception detail.

Preconditions
CEA Exception Originator is logged on

Guarantees

Success End Condition
CEA Exception Originator successfully edit/ close/ resubmit the check exception.

Failed End Condition
Nothing happens

MAIN SUCCESS SCENARIO

1. CEA Exception Originator view the check exception details.
2. System displays the conditionally edit/ resubmit and close options.
3a. CEA Exception Originator chooses ‘Edit’ option.
4. CEA system presents the check exception editable form.
5. CEA Exception Originator submits the filled from.
6. CEA system validates the submitted form with business rules.

7a. CEA system successfully saves the updated check exception, update the check exception history and presents the success message.

**EXTENSIONS**

*a. CEA Exception Originator decides to quit:

*a1. CEA system discards any intermediate data and exits [fail].

3b. CEA Exception Originator clicks on ‘Resubmit’ option:

3b1. [step 4]

3c. CEA Exception Originator clicks on ‘Close’ option:

3c1. CEA system change the status to close and presents the user with success message.

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Only the Check Exception Originator can edit/ resubmit and close the check exception who actually created the check exception.</td>
</tr>
<tr>
<td>2</td>
<td>Only Check Exception with ‘Responded’ status can be resubmitted.</td>
</tr>
<tr>
<td>3</td>
<td>Only Check Exception with ‘Responded’ status and ‘Item Sent’ disposition reason can be closed.</td>
</tr>
<tr>
<td>4</td>
<td>All Check Exception changes must be saved in Check Exception history.</td>
</tr>
</tbody>
</table>

**USE CASE 5 Route check exception**

**CHARACTERISTIC INFORMATION**

**Brief description**

This use case allows the check exception to route to a particular storage location so that check exception Responders can further work on check exception to resolve it.

**Primary Actor**

Check Exception Originator

**Stakeholders**
Check Exception Originator
Check Exception Responder
Check Exception Supervisor
Check Exception Manager

**Trigger**

CEA Exception Originator submits the ATM ID/ BCRC/ Image Mark SMI Sequence number and Capture Date on Create Check Exception Modal.

**Preconditions**

CEA Exception Originator is logged on

**Guarantees**

**Success End Condition**

Check exception gets routed to a specific location.

**Failed End Condition**

Nothing happens.

**MAIN SUCCESS SCENARIO**

1a. CEA Exception Originator submits the create ATM check exception modal.
2. CEA system validates the submitted data.
3a. CEA system do storage location lookup based on ATM ID.
4. CEA system assigns the storage location to check exception and presents the ‘Create Check Exception’ form with populated storage location.

**EXTENSIONS**

*a. CEA Exception Originator decides to quit:

  *a1. CEA system discards any intermediate data and exits [fail].

1b. CEA Exception Originator submits the create BCRC/ Image mark check exception modal:

  1b1. CEA system validates the submitted data.
  1b2. CEA system do storage location look based on SMI Sequence number [step 4].

3b. Invalid data:

  3b1. CEA system presents the error fields with error message [repeat 1].
3c. ATM ID does not exist:

3c1. CEA system presents the error message ‘ATM ID not found’ and asks to provide the valid ATM ID

[Repeat 1].

3b. No storage location found:

3b1. CEA system assigns the ‘Invalid Storage Location’ to check exception [step 5].

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>To route AMT check exception ATM ID must present in Check Exception Automation database.</td>
</tr>
<tr>
<td>2</td>
<td>To route BCRC/ Image mark check exception SMI Sequence number must exist in central check exception system</td>
</tr>
<tr>
<td>3</td>
<td>ATM ID must be alphanumeric and must be 12 digits.</td>
</tr>
<tr>
<td>4</td>
<td>SMI sequence number must be alphanumeric and must be of 11 digits.</td>
</tr>
</tbody>
</table>

**USE CASE 6 Maintain LTPS Storage Location**

**CHARACTERISTIC INFORMATION**

**Brief description**

This use case allows the CEA Manager to edit, delete and add the LTPS storage location.

**Primary Actor**

Exception Manager

**Stakeholders**

CEA Manager
CEA Responder
CEA Supervisor

**Trigger**

CEA Exception Manager chooses LTPS storage location ‘Configuration’.
Preconditions

CEA Exception Manager is logged on

Guarantees

Success End Condition

CEA Manager successfully performs CRUD (Create/ delete/ edit) operations on LTPS storage location.

Failed End Condition

Nothing happens

MAIN SUCCESS SCENARIO

1a. CEA Manager chooses ‘Add New LTPS’ location.
2. System presents the editable ‘Add LTPS’ location form to CEA Manager
3. CEA Manager submits the filled form to the CEA system.
4. CEA system validates the submitted data from business rules.
5a. CEA system successfully saves the new LTPS location to the system and presents the CEA Manager Success message.

EXTENSIONS

*a. CEA Manager decides to quit:
   *a1. CEA system discards any intermediate data and exits [fail].

1b. CEA Manager chooses ‘Edit LTPS’ location:
   1b1. CEA system presents the LTPS storage location editable form with data to the CEA Manager [step 3].
1c. CEA Manager chooses ‘Delete LTPS’ location:
   3c1. CEA system asks CEA Manager to confirm the delete operation.
   3c2a. CEA Manager confirms delete operation:
      CEA system validates the delete operation with business rules.
      CEA system deletes the LTPS storage location and presents the CEA Manager Success message.
      CEA system denies the delete operation and presents the CEA Manager Failure message [failure].
3c2b. CEA system cancel the delete operation:

CEA system discards any changes [failure].

5b. Invalid data:

5b1. CEA system presents the invalid field with error message.

5b2. CEA Manager corrects the error fields and submits the form [repeat 4].

5c. Duplicate LTPS storage location:

5c1. CEA system presents the error message ‘Duplicate entry’ to the CEA Manager [failure].

Business Rules

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Duplicate LTPS storage location can be added.</td>
</tr>
<tr>
<td>2</td>
<td>LTPS storage location company, cost center, mail code are mandatory.</td>
</tr>
<tr>
<td>3</td>
<td>Company can contain only numeric values and must have 5 digits.</td>
</tr>
<tr>
<td>4</td>
<td>Cost center can contain only numeric values and must be 7 digits.</td>
</tr>
<tr>
<td>5</td>
<td>At least one channel is mandatory.</td>
</tr>
</tbody>
</table>

USE CASE 7 Maintain Banking center Storage Location

CHARACTERISTIC INFORMATION

Brief description

This use case allows the CEA Manager to edit, delete and add the Banking Center storage location.

Primary Actor

Exception Manager

Stakeholders

CEA Manager
CEA Responder
CEA Supervisor
Trigger
CEA Exception Manager chooses Banking Center storage location ‘Configuration’.

Preconditions
CEA Exception Manager is logged on

Guarantees

Success End Condition
CEA Manager successfully performs CRUD (Create/ delete/ edit) operations on Banking Center storage location.

Failed End Condition
Nothing happens

MAIN SUCCESS SCENARIO

1a. CEA Manager chooses ‘Add New Banking Center’ storage location.
2. System presents the editable ‘Add Banking Center’ location form to CEA Manager
3. CEA Manager submits the filled form to the CEA system.
4. CEA system validates the submitted data from business rules.
5a. CEA system successfully saves the new Banking Center location to the system and presents the CEA Manager Success message.

EXTENSIONS

*a. CEA Manager decides to quit:

   *a1. CEA system discards any intermediate data and exits [fail].

1b. CEA Manager chooses ‘Edit Banking Center’ location:

   1b1. CEA system presents the Banking Center storage location editable form with data to the CEA Manager [step 3].

1c. CEA Manager chooses ‘Delete Banking Center’ location:

   3c1. CEA system asks CEA Manager to confirm the delete operation.

   3c2a. CEA Manager confirms delete operation:

       CEA system validates the delete operation with business rules.
CEA system deletes the Banking Center storage location and presents the CEA Manager Success message.

CEA system denies the delete operation and presents the CEA Manager Failure message [failure].

3c2b. CEA system cancel the delete operation:

CEA system discards any changes [failure].

5b. Invalid data:

   5b1. CEA system presents the invalid field with error message.

   5b2. CEA Manager corrects the error fields and submits the form [repeat 4].

5c. Duplicate Banking Center storage location:

   5c1. CEA system presents the error message ‘Duplicate entry’ to the CEA Manager [failure].

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Duplicate Banking Center storage location can be added.</td>
</tr>
<tr>
<td>2</td>
<td>Banking Center storage location company, cost center, mail code and banking center name are mandatory.</td>
</tr>
<tr>
<td>3</td>
<td>Company can contain only numeric values and must have 5 digits.</td>
</tr>
<tr>
<td>4</td>
<td>Cost center can contain only numeric values and must be 7 digits.</td>
</tr>
<tr>
<td>5</td>
<td>Banking center can be associated with banking center or one of the LTPS storage location.</td>
</tr>
</tbody>
</table>

**USE CASE 8** Maintain check exception configuration

**CHARACTERISTIC INFORMATION**

**Brief description**

This use case allows the CEA Manager to edit, delete and add the Business Address, Root Cause Reason, Request Type and Originator Edit Reasons.

**Primary Actor**
Exception Manager

**Stakeholders**

CEA Manager
CEA Responder
CEA Supervisor
CEA Originator

**Trigger**

CEA Exception Manager chooses maintain check exception ‘Configuration’.

**Preconditions**

CEA Exception Manager is logged on

**Guarantees**

* **Success End Condition**

CEA Manager successfully performs CRUD (Create/ delete/ edit) operations on check exception configuration.

* **Failed End Condition**

Nothing happens

**MAIN SUCCESS SCENARIO**

1a. CEA Manager chooses maintain one of the options Business Address/ Root Cause Reason/ Request Type/ Originator Edit Reason’ configuration.

2a. CEA Manager chooses ‘Add configuration’.

3. CEA system presents the editable form to the CEA Manager.

4. CEA Manager submits the filled form to the CEA system.

5. CEA system validates the submitted data with the business rules.

6a. CEA system successfully saves the new configuration to the system and presents the CEA Manager Success message.

**EXTENSIONS**

*a. CEA Manager decides to quit:

   *a1. CEA system discards any intermediate data and exits [fail].
2b. CEA Manager chooses ‘Edit Configuration’:

   1b1. CEA system presents the configuration editable form with data to the CEA Manager [step 4].

2c. CEA Manager chooses ‘Delete Configuration’:

   3c1. CEA system asks CEA Manager to confirm the delete operation.

   3c2a. CEA Manager confirms delete operation:

       CEA system validates the delete operation with business rules.

       CEA system deletes the configuration and presents the CEA Manager Success message.

       CEA system denies the delete operation and presents the CEA Manager Failure message [failure].

   3c2b. CEA system cancel the delete operation:

       CEA system discards any changes [failure].

5b. Invalid data:

   5b1. CEA system presents the invalid field with error message.

   5b2. CEA Manager corrects the error fields and submits the form [repeat 5].

5c. Duplicate Configuration:

   5c1. CEA system presents the error message ‘Duplicate entry’ to the CEA Manager [failure].

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Duplicate Configuration can be added.</td>
</tr>
<tr>
<td>2</td>
<td>Root Cause Reason must contain only alphabets, not special characters are allowed, and maximum allowed length is 255 characters.</td>
</tr>
<tr>
<td>3</td>
<td>Request Type must contain only alphabets, not special characters are allowed, and maximum allowed length is 255 characters.</td>
</tr>
<tr>
<td>4</td>
<td>Business Address name must contain only alphabets, not special characters are allowed, and maximum allowed length is 255 characters, Address line 1, City, State and Postal Code are mandatory fields.</td>
</tr>
</tbody>
</table>
USE CASE 9 CEA System User management

CHARACTERISTIC INFORMATION

Brief description
This use case allows the CEA Manager to edit, delete, add and upload CEA system user.

Primary Actor
Exception Manager

Stakeholders
CEA Manager

Trigger
CEA Exception Manager chooses maintain CEA system ‘User Management’.

 Preconditions
CEA Exception Manager is logged on

Guarantees

Success End Condition
CEA Manager successfully performs CRUD (Create/ delete/ edit/ upload) operations on CEA system user.

Failed End Condition
Nothing happens

MAIN SUCCESS SCENARIO
1. CEA Manager chooses ‘User Management’ from CEA system.
2. CEA system presents the CEA user maintenance options.
3a. CEA Manager chooses ‘Add User’ from CEA system.
4. CEA system presents the ‘Add User’ editable form to CEA Manager.
5. CEA Manager submits the filled ‘Add User’ form to the CEA system.
6. CEA system validates the submitted data with CEA system business rules.
7a. CEA system saves the new user to the CEA system database and presents the success message to the CEA Manager.
EXTENSIONS

*a. CEA Manager decides to quit:

*a1. CEA system discards any intermediate data and exits [fail].

3b. CEA Manager chooses ‘Edit User:

3b1. CEA Manager Search for the user by unique bank ID.

3b2a. User Found:

CEA system presents the user details with editable form to the CEA Manager.

CEA Manager submits the modified form to the CEA system [step 6].

3b2b. User Not Found:

CEA system presents warning message and asks CEA Manager to refine the search [repeat at 3b1].

3c. CEA Manager chooses ‘Delete User:

3c1. CEA Manager Search for the user by unique bank ID.

3c2a. User Found:

CEA system presents the user details to the CEA Manager and asks CEA Manager to confirm the delete operation.

CEA Manager confirms delete operation:

CEA system validates the delete operation with business rules.

CEA system deletes the user and presents the CEA Manager Success message.

CEA system denies the delete operation and presents the CEA Manager Failure message [failure].

CEA system cancel the delete operation:

CEA system discards any changes [failure].

3c2b. User Not Found:

CEA system presents warning message and asks CEA Manager to refine the search [repeat at 3b1].

3d. CEA Manager chooses ‘Upload User:

3d1. CEA System asks for CSV format file to upload the users.
3d2. CEA Manager provide the CSV file to the CEA system.

3d3. CEA system validates the file format.

3d4a. CEA system uploads the file to the CEA system database.

3d4b. Invalid File:

CEA system presents the error message to the CEA Manager and asks to provide the valid file.

7b. Invalid data:

5b1. CEA system presents the invalid field with error message.

5b2. CEA Manager corrects the error fields and submits the form [repeat 5].

7c. Duplicate User:

7c1. CEA system presents the error message ‘Duplicate entry’ to the CEA Manager [failure].

**Business Rules**

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bank Unique ID is mandatory and must be alphanumeric and must be exactly of 7 digits.</td>
</tr>
<tr>
<td>2</td>
<td>Role is mandatory for editing/ adding/ uploading user.</td>
</tr>
<tr>
<td>3</td>
<td>Duplicate user cannot be added.</td>
</tr>
<tr>
<td>4</td>
<td>Upload file must be CSV format file.</td>
</tr>
</tbody>
</table>

**USE CASE 9 Respond check exception**

**CHARACTERISTIC INFORMATION**

**Brief description**

This use case allows the CEA Responder to respond to the check exception for check exception resolution.

**Primary Actor**

CEA Responder
Stakeholders
CEA Responder

Trigger
CEA responder chooses ‘Respond’ check exception.

 Preconditions
CEA Responder is logged on

 Guarantees

Success End Condition
CEA Responder successfully responds to the check exception.

Failed End Condition
Nothing happens

MAIN SUCCESS SCENARIO
1. CEA Responder chooses ‘View Exception Details.’
2. CEA system validates the check exception and presents the check exception details with conditional ‘Respond’ option.
3. CEA Responder chooses ‘Respond’ check exception.
4. CEA system presents the ‘Respond Modal’ to the CEA Responder.
5. CEA Responder submits the ‘Respond’ check exception form to the CEA system.
6. CEA system validates the submitted data with business rules.
7a. CEA system changes the check exception status to ‘Responded’ or ‘Closed’, saves into the CEA database and presents the success message to the CEA Responder.

EXTENSIONS
*a. CEA Responder decides to quit:
   *a1. CEA system discards any intermediate data and exits [fail].
7b. Invalid data:
   5b1. CEA system presents the invalid field with error message to CEA Responder.
   5b2. CEA Responder corrects the error fields and submits the form [repeat 6].
### Business Rules

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Only check exception with status ‘Ready For Work’ or ‘In Process’ can be responded.</td>
</tr>
<tr>
<td>2</td>
<td>Responder can respond to only those check exception having storage location mail code prefix same as responder mail code prefix e.g. SD1-067-1234 and SD1-987-4567 have same prefix SD1.</td>
</tr>
<tr>
<td>3</td>
<td>Disposition reason is mandatory and must be chosen from available disposition reasons list</td>
</tr>
<tr>
<td>4</td>
<td>Respond comments are optional, maximum allowed length of comment are 255 and no special characters are allowed in comment.</td>
</tr>
<tr>
<td>5</td>
<td>Item can be responded only one time.</td>
</tr>
</tbody>
</table>

### USE CASE 10 Adjust customer

#### CHARACTERISTIC INFORMATION

**Brief description**

This use case allows the CEA Analyst to adjust the customer to debit or credit the disputed amount to the customer bank account, and send the customer automatic notification.

**Primary Actor**

CEA Analyst

**Stakeholders**

CEA Analyst

Customer

**Trigger**

CEA Analyst chooses ‘Adjust Customer’ from check exception details.

**Preconditions**

CEA Analyst is logged on

**Guarantees**
**Success End Condition**

CEA Analyst successfully adjust the customer and generate the customer notification letter.

**Failed End Condition**

Nothing happens

**MAIN SUCCESS SCENARIO**

1. CEA Analyst chooses ‘View Exception Details.’

2. CEA system communicates with the external Bank Account systems and receives the customer as well as customer account details.

3a. CEA system displays the check exception, customer and account details with ‘Adjust Customer’ option to the CEA Analyst.

4. CEA Analyst chooses the ‘Adjust Customer’ from CEA system.

5. CEA system presents the customer adjust details modal popup and asks to confirm the customer adjustment.

6a. CEA Analyst confirms the customer adjustment.

7. CEA system communicates with the external bank system to post the customer adjustment.

8a. CEA system generates the customer notification and saves to the CEA system.

9. CEA system changes the check exception status from Non-Monetary to Monetary, check exception status from ‘Ready for Work’ to ‘In Process’ and presents to the CEA Analyst.

**EXTENSIONS**

*a. CEA Analyst decides to quit:

   *a1. CEA system discards any intermediate data and exits [fail].

3b. Remote System Exception:

   3b1. CEA system presents the check exception details with warning message having unique error event id.

   3b2. CEA system does not display the ‘Adjust Customer’ option [failure].

6b. Cancel the Adjustment:

   CEA system hides the ‘Adjust Customer’ modal on check exception details.

8b. Remote System Exception:

   CEA system displays the error message to the CEA Analyst [failure].
Business Rules

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>After successful customer adjustment financial status must change from ‘Non-Monetary’ to ‘Monetary’.</td>
</tr>
<tr>
<td>2</td>
<td>After successful customer adjustment check exception status must change to ‘In Process’</td>
</tr>
<tr>
<td>3</td>
<td>Debit check exception amount must be credited to the customer and Credit check exception amount must be debited to the customer.</td>
</tr>
</tbody>
</table>
| 4      | Credited Adjustment must have the Transaction Code 001.  
|        | Debited adjustment must have the Transaction Code 002. |
| 5      | Customer and account details must be present on check exception details before making adjustment, if customer and account details are not present customer adjustment should fail and no customer notification must be generated. |

USE CASE 11 Print check exception

CHARACTERISTIC INFORMATION

Brief description

This use case allows the CEA Originator/ CEA Responder/ CEA Supervisor/ CEA Manager all denoted as CEA user to print the check exception.

Primary Actor

CEA user

Stakeholders

CEA user

Trigger

CEA user chooses ‘Print Check Exception’.

Preconditions

CEA user is logged on
Guarantees

Success End Condition
CEA user successfully prints the check exception.

Failed End Condition
Nothing happens

MAIN SUCCESS SCENARIO
1a. CEA user selects multiple check exceptions.
2. CEA system enables the ‘Print Selection’ button.
3. CEA user selects the ‘Print Selection’ button from CEA system.
4. CEA system opens the operating system print widow.
5a. CEA user selects ‘Print’ from print window.
6a. CEA system prints the check exceptions.

EXTENSIONS
*a. CEA user decides to quit:
   *a1. CEA system discards any intermediate data and exits [fail].
1b. CEA user select check exception details:
   1b1. CEA system presents the check exception details with ‘Print Check Exception’ button.
   1b2. CEA user selects ‘Print Check Exception’ from CEA system [step 4].
5b. CEA user chooses ‘Cancel’:
   CEA system discards any changes and close the print window [failure].
6b. Printer Unavailable:
   CEA system displays printer unavailable warning message [failure].

Business Rules

<table>
<thead>
<tr>
<th>Number</th>
<th>Business Rules Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Print selection button should be enabled after at least one selection.</td>
</tr>
</tbody>
</table>
Use Case Diagram

Figure 10 Use Case Diagram
PRU System Behavioral Modeling

Use Case 1 Log in

Sequence Diagram

Figure 11 Log in Sequence Diagram
Collaboration Diagram

Figure 12 Login Collaboration Diagram
Figure 13 Login Class Diagram

Use Case 2 Create Check Exception

Sequence diagram

Figure 14 Create Check Exception Sequence Diagram
Collaboration diagram

Figure 15 Create Check Exception Collaboration Diagram
Figure 16 Create Check Exception Class Diagram

Use Case 3 View Check Exception

Sequence diagram

Figure 17 View Check Exception Sequence Diagram
Collaboration diagram

Figure 18 View Check Exception Collaboration Diagram
Use Case 4 Maintain Check Exception

Sequence diagram

Figure 19 Maintain Check Exception Sequence Diagram
Collaboration diagram

**Figure 20** Maintain Check Exception Collaboration Diagram
Use Case 6 Maintain Long Term Storage Location

Sequence diagram

Figure 21 Maintain Long Term Storage Location Sequence Diagram

Figure 22 Maintain Long Term Storage Location Class Diagram
Collaboration diagram

Figure 23 Maintain Long Term Storage Location Collaboration Diagram
Use Case 7 Maintain Banking Center Storage Location

Sequence diagram

Figure 24 Maintain BC Storage Location Sequence Diagram
Collaboration diagram

Figure 25 Maintain BC Storage Location Collaboration Diagram
Use Case 8 Maintain Check Exception Configuration

Sequence diagram

Figure 26 Maintain Check Exception Configuration Sequence Diagram
Collaboration diagram

Figure 27 Maintain Check Exception Configuration Collaboration Diagram

Figure 28 Maintain Check Exception Configuration Class Diagram
Use Case 9 CEA System User Management

Sequence diagram

Figure 29 CEA System User Management Sequence Diagram
Collaboration diagram

Figure 30 CEA System User Management Collaboration Diagram

Figure 31 CEA System User Management Class Diagram
Use Case 10 Respond check exception

Sequence diagram

Figure 32 Respond Check Exception Sequence Diagram
Figure 33 Respond Check Exception Collaboration Diagram

Figure 34 Respond Check Exception Class Diagram
Use Case 11 Adjust Customer

Figure 35 Adjust Customer Sequence Diagram

Figure 36 Adjust Customer Class Diagram
Figure 37 Adjust Customer Collaboration Diagram
CHAPTER 4

CASE STUDY

Web Pages

Creating Banking Center Check Exception

Figure 38 shows the new check exception creation form. When a bank employee called requestor which is already logged in and visit the check exception system with “Originator” role and choose the “Create Exception” option then application presents the check exception creation form. On this page originator can perform the following actions:

- Search any check exception by entering the exception ID using search box.
- Can create a new check exception of given kinds such as ATA, ATM, BCRC and Image Mark. Requestor information is prepopulated by the application. To populate the information application makes a call to the central employee system via web service call and receives the information. This information is editable. But there are some information which are prepopulated by the given check number and exception date on previous screen, that information is locked and cannot be edited. After filling the entire check exception form, requestor can submits the form or can cancel the form. If requestor cancels the request then application takes the control to the requestor’s default landing page. If requestor chooses the “Submit Request” option then application creates the new check exception and routes it to the appropriate resolution location.
Figure 38 Creating Banking Center Check Exception
Validating Banking Center Check Exception

Figure 39 shows the validation mechanism of the application when requestor submits the new check exception form. Application validates the data entered by the requestor by business validation rules and if any rules fail an appropriate error message with failure reason is present under the error fields and error field is marked as red. Like in figure 39, there are few fields have invalid values. Capture Company Code is not in the right format, State Exception Occurred is not two character code and Exception Total is missing which is a mandatory field. Errors are shown in red color and warnings are shown in orange color. Requestor will be unable to submit the check exception request until all the invalid fields are fixed.
Figure 39 Validating Banking Center Check Exception
View GSS Check Exception with Check Image

Figure 40, shows the check exception details with check exception image. User can reach to this page via searching the exception on search box or via clicking the exception on landing page. This page shows the full details of the check exception. The check exception detail contains the status of the exception, check exception total, check exception balance which is equal to the check exception total initially but can be different as check exception amount gets paid, reason of the check exception which is in this case is “Wrong Amount”, customer details whose check is failed, bank general ledger account details from where application can withdraw and deposit the disputed amount, check posting details with check number and check image which got failed to deposit. Few of the information are editable.
Figure 40 Viewing GSS Check Exception with Check Image
Editing GSS Check Exception

Figure 41, shows the editable Transaction section where requestor can update the information.
Check Exception Originator Landing Page

Figure 42 shows the requestor having role “Originator” default landing page, when a requestor logs in and try to access check exception application then application does the authorization check and loads the user role. If user role is “Originator” then application presents the originator landing page which contains the list of check exceptions created by originators of given mail code location sorted by date. Requestor can view, edit and print the check exceptions from this view.

![Check Exception Originator Landing Page](image-url)
Filtering the List of Check Exceptions

User can filter the list of check exceptions using the filter. User can select type of exception, root cause of exception, request type, status and disposition reason. By applying the selected filter application shows only those exception which matches the filter criteria.

<table>
<thead>
<tr>
<th>Status</th>
<th>Exception ID</th>
<th>Amount</th>
<th>Request Date</th>
<th>Channel</th>
<th>Request Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ready For Work</td>
<td>123</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>116</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>115</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>138</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>127</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>124</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>114</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>139</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>12122</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
<tr>
<td>Ready For Work</td>
<td>113</td>
<td>100.00</td>
<td>2021-01-01</td>
<td>BCRC</td>
<td>Customer Tape Envelope</td>
</tr>
</tbody>
</table>

Figure 43 Filtering the List of Check Exceptions
Creating Check Exception Modal

Figure 44, figure 45 and figure 46 show the functionality when requestor selects the create check exception option. Application presents the popup box where requestor has to enter the Item Sequence Number and Capture Date which is the uniqueness of the rejected check. By this information application makes a web service call to central bank system which has all the information of rejected checks and pulls the information and prepopulate the check creation form with some locked fields. Figure 50, shows the user friendly spinner to make user aware that application is processing the data while application is calling the external systems to gather the data.

![Creating Check Exception Modal](image)

*Figure 44 Creating Check Exception Modal*
Figure 45 Creating Check Exception Modal with Entered Values

Processing User Request Waiting Spinner

Figure 46 Waiting Spinner
Check Exception Creation Success Message and Routing Failure

Figure 47, shows the success message with check exception unique ID after creating the check exception successfully, this screen also shows the warning message because it is currently unassigned, when a responder opens the exception then the status will be changed to “Ready for Work”.

<table>
<thead>
<tr>
<th>Item Requests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status: Unassigned</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
<tr>
<td>Ready For Work</td>
</tr>
</tbody>
</table>

**Figure 47 Check Exception Success Message**
Check Exception Detail View

Figure 48, shows the BCRC check exception detail view with all of the information. Few information can be edited by selecting edit button and check exception can be printed as well by choosing print option.

Figure 48 Check Exception Detail View
Check Exception History View

Figure 49, shows the check exception life time history. By request history user can see what changes have been made, who made the changes, at what day and time, what values have been changed with older values and newer values. Exception history is kind of audit for check exception and plays an important role to keep track any changes made on exception. This figure shows a check exception has been created with requestor details.

Figure 49 Check Exception History View
Printing Check Exception View

Figure 50, shows the print view of the check exception. User can select the exceptions from exception list or can choose the select all if user wants to print all the check exceptions and can give print command.

![Figure 50 Printing Check Exception View](image)
Check Exception with Respond Option View

Figure 51, shows the view where a responder can respond the check exception. Responder is the check resolver and resolves the check exception and responds by clicking on respond button and provides the reason and comments of the responded action.

![Figure 51 Check Exception Responder View](image-url)
Supervisor Landing Page

Figure 52, shows the supervisor landing page. Supervisor landing page is tied to the supervisor mail code location. Supervisor landing page has all the details of all the check exceptions being worked under supervisor’s mail code location. By this view supervisor can see how many check exceptions are open, how many are new, how many have been closed and number of exceptions with age.

Figure 52 Supervisor Landing Page
Responding Check Exception

Figure 53 shows the popup box when responder click respond button for a check exception. Responder needs to enter the disposition reason and comments to track the check exception resolution.

Figure 53 Responding Check Exception
CHAPTER 5

CONCLUSION

Check Exception Automation system is a good example of how IT can make a business better and can impact the business process. This project is not only become beneficial for organization but also providing high quality quick check resolution services to the customers. Now audit check can be resolved in average one day, which is a huge benefit to the customers as they do not need to wait for their own money. This system also benefits the bank in liquid money by saving the mail services and check maintenance cost. System also reduces the human error which were usual before because of the complexity and number of checks under one employee to resolve. Overall, Check Exception Automation is a big win for business as well as for customers. Business is very happy with this project and want to add more modules to this project. Currently this project handles only few kinds of check location such as ATM, ATA etc. but bank also deals with bigger check systems such as big organization like Walmart, Costco etc. also which accept checks from customers and deposit in bank in a big bundle which is another level of complexity. So business wants to automate all kinds of check exception with this project. There is lot of work and complexity this project needs to handle in future.
REFERENCES


http://docs.oracle.com/database/121/TDPJD/introduction.htm#TDPJD101


https://docs.jboss.org/author/display/AS7/JPA+Reference+Guide?_sscc=t


APPENDICES

APPENDIX A: USERS’ MANUAL

Gantt chart

Below table shows the project tasks with estimated timelines.

Table 17 Gantt chart

<table>
<thead>
<tr>
<th>Task Mode</th>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Total Project</td>
<td>12.75 mons</td>
<td>Tue 6/10/14</td>
<td>Sat 5/30/15</td>
</tr>
<tr>
<td>2</td>
<td>1 Planning Phase</td>
<td>60 days</td>
<td>Tue 6/10/14</td>
<td>Mon 9/1/14</td>
</tr>
<tr>
<td>3</td>
<td>2. Analysis Phase</td>
<td>44 days</td>
<td>Mon 9/1/14</td>
<td>Thu 10/30/14</td>
</tr>
<tr>
<td>4</td>
<td>2.1 Examine Requirements</td>
<td>22 days</td>
<td>Mon 9/1/14</td>
<td>Tue 9/30/14</td>
</tr>
<tr>
<td>5</td>
<td>2.2 Develop and Analyze Test Plan</td>
<td>11 days</td>
<td>Wed 10/1/14</td>
<td>Wed 10/15/14</td>
</tr>
<tr>
<td>6</td>
<td>2.3 Design and Analyze Maintenance Plan</td>
<td>9 days</td>
<td>Thu 10/16/14</td>
<td>Tue 10/28/14</td>
</tr>
<tr>
<td>7</td>
<td>2.4 Design and Approve Acceptance Plan</td>
<td>3 days</td>
<td>Tue 10/28/14</td>
<td>Thu 10/30/14</td>
</tr>
<tr>
<td>8</td>
<td>3 Design Phase</td>
<td>44 days</td>
<td>Sat 11/1/14</td>
<td>Wed 12/31/14</td>
</tr>
<tr>
<td>9</td>
<td>3.1 Architecture Design</td>
<td>9 days</td>
<td>Sat 11/1/14</td>
<td>Wed 11/12/14</td>
</tr>
<tr>
<td>10</td>
<td>3.2 Database Design</td>
<td>9 days</td>
<td>Thu 11/13/14</td>
<td>Tue 11/25/14</td>
</tr>
<tr>
<td>11</td>
<td>3.3 Application Design</td>
<td>14 days</td>
<td>Wed 11/26/14</td>
<td>Mon 12/15/14</td>
</tr>
<tr>
<td>12</td>
<td>3.4 Requirement Grooming With Development Team</td>
<td>12 days</td>
<td>Tue 12/16/14</td>
<td>Wed 12/31/14</td>
</tr>
<tr>
<td>13</td>
<td>4. Implementation Phase</td>
<td>108 days</td>
<td>Thu 1/1/15</td>
<td>Sat 5/30/15</td>
</tr>
<tr>
<td>14</td>
<td>4.1 Development Phase 1</td>
<td>43 days</td>
<td>Thu 1/1/15</td>
<td>Sat 2/28/15</td>
</tr>
<tr>
<td>15</td>
<td>4.2 Dark Release And Support</td>
<td>8 days</td>
<td>Sat 2/28/15</td>
<td>Tue 3/10/15</td>
</tr>
<tr>
<td>16</td>
<td>4.3 Development Phase 2</td>
<td>44 days</td>
<td>Wed 3/11/15</td>
<td>Sun 5/10/15</td>
</tr>
<tr>
<td>17</td>
<td>4.4 Production Release</td>
<td>16 days</td>
<td>Mon 5/11/15</td>
<td>Sat 5/30/15</td>
</tr>
</tbody>
</table>
Project Use Case Diagram

Check Exception Automation (CEA) System

- CEA Originator
- CEA Manager
- CEA Responder
- CEA Analyst
- Banking Center System
- Image Mark System
- Unique Bank ID System
- Bank Account System
- Log In
- CEA System User Management
- Respond Check Exception
- Adjust Customer
- Maintain UIPS Storage Location
- Maintain Banking Center Storage Location
- Maintain Check Exception Configuration
- Print Check Exception
- View Check Exception
- Route Check Exception
- Maintain Check Exception
- Create Check Exception
The above use case diagram shows high level overview of the check exception system. It shows what kind of role can access what kind of functionality and how check exception system is linked with bank central system.

**CEA Database Design**
### CEA System Data Dictionary

#### ROLES

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROLE_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>ROLE_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
</tbody>
</table>

#### PERMISSIONS

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERMISSION_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>PERMISSION_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
</tbody>
</table>

#### ROLES_PERMISSIONS

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROLE_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>PERMISSION_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
</tbody>
</table>
### USERS Table

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>STANDARD_ID</td>
<td>VARCHAR2(7 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>NUMBER(1,0)</td>
<td>No</td>
<td>1</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>5</td>
<td>(null)</td>
</tr>
<tr>
<td>PHONE</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>6</td>
<td>(null)</td>
</tr>
<tr>
<td>EMAIL</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>7</td>
<td>(null)</td>
</tr>
<tr>
<td>DEPARTMENT</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>8</td>
<td>(null)</td>
</tr>
<tr>
<td>FAX</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>9</td>
<td>(null)</td>
</tr>
<tr>
<td>ADDRESS_1</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>10</td>
<td>(null)</td>
</tr>
<tr>
<td>ADDRESS_2</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>11</td>
<td>(null)</td>
</tr>
<tr>
<td>LAST_LOGGED_IN_DATE</td>
<td>TIMESTAMP(6)</td>
<td>Yes</td>
<td>(null)</td>
<td>12</td>
<td>(null)</td>
</tr>
<tr>
<td>CITY</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>13</td>
<td>(null)</td>
</tr>
<tr>
<td>STATE</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>14</td>
<td>(null)</td>
</tr>
<tr>
<td>ZIP</td>
<td>VARCHAR2(10 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>15</td>
<td>(null)</td>
</tr>
<tr>
<td>ROLE_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>16</td>
<td>(null)</td>
</tr>
</tbody>
</table>

### CHECK_EXCEPTION Table

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXCEPTION_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>PHONE</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
<tr>
<td>FINANCIAL_STATUS</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>5</td>
<td>(null)</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>NUMBER(1,0)</td>
<td>No</td>
<td>1</td>
<td>6</td>
<td>(null)</td>
</tr>
<tr>
<td>FAX</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>7</td>
<td>(null)</td>
</tr>
<tr>
<td>EMAIL</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>8</td>
<td>(null)</td>
</tr>
<tr>
<td>ADDRESS</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>9</td>
<td>(null)</td>
</tr>
<tr>
<td>CITY</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>10</td>
<td>(null)</td>
</tr>
<tr>
<td>STATE</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>11</td>
<td>(null)</td>
</tr>
<tr>
<td>ZIP</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>12</td>
<td>(null)</td>
</tr>
<tr>
<td>REQUEST_TIMESTAMP</td>
<td>TIMESTAMP(6)</td>
<td>No</td>
<td>(null)</td>
<td>13</td>
<td>(null)</td>
</tr>
<tr>
<td>CLOSED_TIMESTAMP</td>
<td>TIMESTAMP(6)</td>
<td>Yes</td>
<td>(null)</td>
<td>14</td>
<td>(null)</td>
</tr>
<tr>
<td>COMMENTS</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>15</td>
<td>(null)</td>
</tr>
<tr>
<td>EXCEPTION_TOTAL</td>
<td>NUMBER(10,2)</td>
<td>No</td>
<td>(null)</td>
<td>16</td>
<td>(null)</td>
</tr>
<tr>
<td>COMPANY_CODE</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>17</td>
<td>(null)</td>
</tr>
</tbody>
</table>
### CHECK_EXCEPTION_BASE

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXCEPTION_ID</td>
<td>NUMBER(25, 0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>REFERENCE_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>PROD_TYPE_CODE</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>PORTAL_ID</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
<tr>
<td>BALANCE_AMOUNT</td>
<td>NUMBER(20, 2)</td>
<td>Yes</td>
<td>(null)</td>
<td>5</td>
<td>(null)</td>
</tr>
<tr>
<td>CORRECT_AMOUNT</td>
<td>NUMBER(20, 2)</td>
<td>Yes</td>
<td>(null)</td>
<td>6</td>
<td>(null)</td>
</tr>
<tr>
<td>DISPUTED_AMOUNT</td>
<td>NUMBER(20, 2)</td>
<td>Yes</td>
<td>(null)</td>
<td>7</td>
<td>(null)</td>
</tr>
<tr>
<td>BAG_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>8</td>
<td>(null)</td>
</tr>
<tr>
<td>CHECK_IMAGE_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>9</td>
<td>(null)</td>
</tr>
<tr>
<td>CAPTURE_DATE</td>
<td>TIMESTAMP(6)</td>
<td>No</td>
<td>(null)</td>
<td>10</td>
<td>(null)</td>
</tr>
<tr>
<td>CHECK_SEQUENCE_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>11</td>
<td>(null)</td>
</tr>
<tr>
<td>RESOLUTION_RSN</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>12</td>
<td>(null)</td>
</tr>
<tr>
<td>BANK_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>13</td>
<td>(null)</td>
</tr>
<tr>
<td>ACCOUNT_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>14</td>
<td>(null)</td>
</tr>
<tr>
<td>ADDL_INFO_REQUEST</td>
<td>VARCHAR2(4000 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>15</td>
<td>(null)</td>
</tr>
<tr>
<td>RESOLUTION_COMMENT</td>
<td>VARCHAR2(4000 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>16</td>
<td>(null)</td>
</tr>
<tr>
<td>STORE_NUMBER</td>
<td>VARCHAR2(90 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>17</td>
<td>(null)</td>
</tr>
<tr>
<td>TOTAL_CURR_MISSING_AM</td>
<td>NUMBER(10,2)</td>
<td>Yes</td>
<td>(null)</td>
<td>18</td>
<td>(null)</td>
</tr>
<tr>
<td>TOTAL_COINS_MISSING_AM</td>
<td>NUMBER(10,2)</td>
<td>Yes</td>
<td>(null)</td>
<td>19</td>
<td>(null)</td>
</tr>
<tr>
<td>IMAGE_META_DATA_ID</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>20</td>
<td>(null)</td>
</tr>
</tbody>
</table>

### CHECK_EXCEPTION_COMMENT

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMENT_ID</td>
<td>NUMBER(20, 0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>COMMENT_TEXT</td>
<td>CLOB</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>EXCEPTION_ID</td>
<td>NUMBER(20, 0)</td>
<td>No</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
</tbody>
</table>

### CUSTOMER

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUSTOMER_ID</td>
<td>NUMBER(20, 0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>FIRST_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>ACCOUNT_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
</tbody>
</table>
### Customer Address Table

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDRESS_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>CITY</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>STATE</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>COUNTRY</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
</tbody>
</table>

### Historical Events Table

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVENT_ID</td>
<td>NUMBER(10,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>EVENT_TYPE_ID</td>
<td>NUMBER(10,0)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>ENTITY_ID</td>
<td>NUMBER(10,0)</td>
<td>No</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>EVENT_DESCRIPTION</td>
<td>VARCHAR2(255 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
<tr>
<td>BEFORE_VALUE</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>5</td>
<td>(null)</td>
</tr>
<tr>
<td>AFTER_VALUE</td>
<td>VARCHAR2(255 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>6</td>
<td>(null)</td>
</tr>
<tr>
<td>USER_ID</td>
<td>VARCHAR2(10 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>7</td>
<td>(null)</td>
</tr>
</tbody>
</table>

### Item Request Channel Table

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHANNEL_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>NAME</td>
<td>VARCHAR2(50 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>VARCHAR2(100 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
</tbody>
</table>

### Item Request Type Table

<table>
<thead>
<tr>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>NULLABLE</th>
<th>DATA_DEFAULT</th>
<th>COLUMN_ID</th>
<th>COMMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQUEST_TYPE_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>NAME</td>
<td>VARCHAR2(30 BYTE)</td>
<td>No</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>VARCHAR2(100 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>COLUMN_NAME</td>
<td>DATA_TYPE</td>
<td>NULLABLE</td>
<td>DATA_DEFAULT</td>
<td>COLUMN_ID</td>
<td>COMMENTS</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------</td>
<td>----------</td>
<td>--------------</td>
<td>-----------</td>
<td>----------</td>
</tr>
<tr>
<td>INSTRUCTION_ID</td>
<td>NUMBER(20,0)</td>
<td>No</td>
<td>(null)</td>
<td>1</td>
<td>(null)</td>
</tr>
<tr>
<td>BANK_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>2</td>
<td>(null)</td>
</tr>
<tr>
<td>ACCOUNT_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>3</td>
<td>(null)</td>
</tr>
<tr>
<td>ALTI_DELIVERY_ID</td>
<td>NUMBER(20,0)</td>
<td>Yes</td>
<td>(null)</td>
<td>4</td>
<td>(null)</td>
</tr>
<tr>
<td>ALTI_BANK_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>5</td>
<td>(null)</td>
</tr>
<tr>
<td>ALTI_ACCOUNT_NUM</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>6</td>
<td>(null)</td>
</tr>
<tr>
<td>CREATED_DT</td>
<td>TIMESTAMP(6)</td>
<td>Yes</td>
<td>(null)</td>
<td>7</td>
<td>(null)</td>
</tr>
<tr>
<td>ACTIVE_INDICATOR</td>
<td>NUMBER(1,0)</td>
<td>No</td>
<td>(null)</td>
<td>8</td>
<td>(null)</td>
</tr>
<tr>
<td>COMPANY_NUMBER</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>9</td>
<td>(null)</td>
</tr>
<tr>
<td>CUSTOMER_NAME</td>
<td>VARCHAR2(50 BYTE)</td>
<td>Yes</td>
<td>(null)</td>
<td>10</td>
<td>(null)</td>
</tr>
<tr>
<td>ALLOW_AUTO_SETTLEMENT</td>
<td>NUMBER(1,0)</td>
<td>Yes</td>
<td>(null)</td>
<td>11</td>
<td>(null)</td>
</tr>
</tbody>
</table>
APPENDIX B: SYSTEM TECHNICAL DOCUMENTATION

CEA System Architecture

Figure 54 CEA System Architecture
CEA System Flow Diagrams

Check Exception Flow Diagram

Figure 55 Check Exception Flow Diagram
Check Exception Routing Flow Diagrams

Figure 56 ATM Routing
Figure 56 Banking Center Routing