American International University-Bangladesh (AIUB)

Department of Computer Science

Faculty of Science & Technology (FST)

OnePay

A Software Quality and Testing Project Submitted
By

Semester: Fall_23_24			Section:	Group No:
SN	Student Name	Student ID	Individual Contribution (in %)	Total Marks: 50 Earned Marks:
23	Aiyoun Khan Anady	20-42921-1	28.3%	
40	Rafid Hassan Risun	21-44878-2	28.3%	
3	Moshiur Rahman Nahin	17-35959-3	28.3%	
6	Gazi MD. Jubayar Hossain	19-40016-1	15%	

The project will be Evaluated for the following Course Outcomes

EVALUATION CRITERIA	Total Marks (50)	
Revision History, Test Plan Identifier, Reference Materials, Problem Background, Solutions	[10 Marks]	
Requirements Specification (System feature, Quality Attributes, System Interface, Project Requirements)	[10 Marks]	
Item Not to be tested, Testing approach (Testing levels, tools, meetings), Test cases	[10 Marks]	
Item pass/fail criteria, Test deliverables, Staffing and Training, Responsibilities, Scheduling, Risk	[10 Marks]	
Approval, Format, Submission, and Defense	[10 Marks]	

Software Test Plan

for

OnePay

Version 1.0 approved

Prepared by Rafid Hassan Risun

sqtDevs

24/12/2023

Table of Contents

Revision History

Revision	Date	Updated by	Update Comments
1	2023.11.04	RAFID HASSAN RISUN	Fixed the responsibilities
2	2023.11.05	AIYOUN KHAN ANADY	Changed the quantifiers of the quality attributes
3	2023.11.06	AIYOUN KHAN ANADY	Changed the project size to Semi Detached to COCOMO
4	2023.11.12	RAFID HASSAN RISUN	

5	2023.11.20	MOSHIUR RAHMAN NAHIN	Fixed Test Cases/Test Items
6	2023.11.25	Gazi MD. Jubayar Hossain	Added instant notification in the pass fail criteria
7	2023.11.05	AIYOUN KHAN ANADY	Added the cross references in the functional requirements
8	2023.12.10	RAFID HASSAN RISUN	Fixed the Testing Approaches
9	2023.12.11	AIYOUN KHAN ANADY	User interface fixes
10	2023.12.13	RAFID HASSAN RISUN	Specified the tools that will be used for the testing

1. TEST PLAN IDENTIFIER: TP-001

2. REFERENCE MATERIALS

Figma 101: Introduction to Figma (no date) Designlab. Available at: https://designlab.com/figma-101-course/introduction-to-figma/ (Accessed: 24 December 2023).

Lars Vogel (c) 2007 - 2023 vogella GmbH (no date) Get more..., vogella.com. Available at: https://www.vogella.com/tutorials/JUnit/article.html (Accessed: 24 December 2023).

Selenium webdriver tutorial in java with examples (2023) BrowserStack. Available at: https://www.browserstack.com/guide/selenium-webdriver-tutorial (Accessed: 24 December 2023).

3. INTRODUCTION

3.1 Background to the Problem

When conducting cross-border financial transactions, people in an increasingly interconnected global economy encounter obstacles. When a user wants to make an international transaction, traditional banking procedures, which include complicated documentation requirements, rigorous verification processes, and high transaction costs, frequently need to be altered. This issue is especially noticeable in some areas where lengthy processes and bureaucratic red tape make it difficult to conduct business internationally. The constraints of Bangladesh's current financial infrastructure prevent people from easily engaging in international

transactions. Users are discouraged from engaging in international commerce due to friction caused by complex verification standards, lengthy processing delays, and expensive transaction prices. The problem is made worse by the requirement for more user-friendly platforms, which deprives potential users of a practical and effective way to transact financial transactions internationally.

Traditional financial institutions' antiquated and onerous policies, which leave them unprepared to handle the needs of an increasingly digitized and globalized world, are the primary source of this issue. These organizations frequently put strict security protocols and legal compliance ahead of user experience, which makes the system difficult and time-consuming for users. The increased demand for people to transact internationally for personal, professional, and lifestyle requirements emphasizes how critical it is to address this issue. There is a clear demand for a financial platform that enables consumers to easily perform cross-border transactions as globalization continues to change the current world. The removal of obstacles caused by conventional banking procedures empowers people to handle their finances globally, promoting economic engagement and cross-border cooperation. Given these difficulties, the suggested web application aims to completely transform how consumers deal with their money by offering a simplified method for managing accounts and making transactions—and most importantly, by making international transactions easier. With the provision of a safe, effective, and user-friendly platform for worldwide financial transactions, this initiative seeks to close the gap between the constraints of traditional banking and the changing demands of consumers in a globally interconnected world.

3.2 Solution to the Problem

A user-centric web application is the suggested remedy, and it was created to address the issues with traditional banking for cross-border transactions. The platform prioritizes user-friendliness, guaranteeing secure login and effortless navigation. A dependable payment channel is included to enable safe domestic transfers for routine transactions. In addition, the program ensures compliance with international financial regulations by

streamlining overseas transactions through relationships with reliable international payment providers. An effective and reliable financial platform benefits from real-time exchange rates and clear fee information. The user experience is improved with a feature that provides a detailed transaction history and in-app assistance for foreign transactions. Legal compliance is of utmost importance, encompassing compliance with data protection laws and international financial rules. Performance monitoring, security improvements, and the addition of new features are all accomplished through the use of ongoing monitoring tools and frequent upgrades. The expected results encompass improved customer experience, heightened economic involvement, and a dedication to efficiency and openness in financial operations worldwide.

3.3. Existing studies of the problem areas.

Popular international payment apps like Google Pay, Apple Pay, and Paypal are not available in Bangladesh, which creates a big gap in the local population's ability to conduct cross-border financial transactions. For foreign transactions, people in Bangladesh typically use traditional banking systems, which are frequently burdened by intricate procedures. Notably, popular mobile banking apps like Upay, Bkash, and Nagad are only available for domestic transactions; they do not offer international service. Due to this, there is now a significant void in the market for a specialized application made specifically for Bangladeshi consumers that would offer a smooth and effective way to handle foreign financial transactions. By providing a user-centric platform tailored especially for the special difficulties and needs of cross-border transactions in Bangladesh, the suggested web application seeks to close this gap.

4. REQUIREMENT SPECIFICATION

4.1. System Features

4.1.1. User Registration

1. The system shall prompt users to provide a valid email address, username, and password during registration.
2. Users must verify their email addresses through a secure confirmation link sent to their registered email.
3. The system shall enforce unique usernames and perform real-time validation during registration.

Priority: High
Preconditions: None
Cross reference: 7.1, 7.2.1

4.1.2. User Login:

1. The system shall provide a login interface requiring users to input their validated username and password.
2. After three consecutive failed login attempts, the system shall generate a one-time-use random verification code for additional security.
3. The system shall implement CAPTCHA functionality after five consecutive failed login attempts to mitigate potential brute-force attacks.

Priority: High
Preconditions: The user has a valid registered account.
Cross reference: 7.1, 7.2.1, 7.2.2

4.1.3. Account Locking

1. If the number of consecutive failed login attempts exceeds a predefined limit (e.g., 5 times), the system may optionally lock the user account for a specified duration (e.g., one hour).

Priority: Medium (optional feature)
Preconditions: The user account is not already locked.

4.1.4. Password Management

1. Users shall have the option to reset their password through a secure, multi-step verification process.
2. The system shall enforce password complexity with a minimum length, alphanumeric requirements, and inclusion of special characters.

Priority: High
Preconditions: The user is logged in or has access to the registered email address.
Cross reference: 7.1, 7.2.1, 7.2.2

4.1.5. Domestic Transactions

1. Users shall initiate domestic transactions by selecting the recipient, entering the amount, and confirming the transaction.
2. The system shall perform real-time balance checks to ensure sufficient funds for the transaction.
3. In case of insufficient funds, the system shall notify the user and prevent the transaction.

Priority: High
Preconditions: The user is logged in and has sufficient funds.
Cross reference: 7.4

4.1.6. Transaction History

1. The application shall maintain a comprehensive transaction history for each user, displaying details such as transaction amounts, dates, and recipients.

Priority: High
Preconditions: The user is logged in.

4.1.7. Foreign Transactions

1. The system shall provide a dedicated interface for users to initiate foreign transactions.

Priority: High
Preconditions: The user is logged in and has access to foreign transaction features.
Cross reference: 7.5

4.1.8. Currency Exchange

1. Users shall select the desired foreign currency and enter the amount in the local currency.
2. The system shall display real-time exchange rates, including any applicable fees, before confirming the transaction.
3. Users shall receive a summary of the transaction, including the converted amount and exchange rate, before finalizing the transaction.

Priority: High
Preconditions: The user is logged in, and foreign transaction features are accessible.
Cross reference: 7.6

4.1.9. Compliance Measures

1. The system shall perform Know Your Customer (KYC) checks for users engaging in foreign transactions.

2. Compliance with international financial regulations, including Anti-Money Laundering (AML) laws, shall be enforced.

Priority: High
Preconditions: The user is logged in, and KYC information is up-to-date.
Cross reference: 7.7

4.1.10. Secure Transactions

1. The system shall implement secure communication protocols (HTTPS) to protect user data during transactions.
2. Regular security audits shall be conducted to identify and address potential vulnerabilities.

Priority: High
Preconditions: None
Cross reference: 7.8

4.1.11. Transaction Verification for High Amounts:

1. For foreign transactions exceeding a specified threshold, the system shall require additional verification steps.

Priority: High
Preconditions: The user is logged in and has a history of high-value transactions.
Cross reference: 7.9

4.1.12. Monitor Logging

1. The application will track the user's logging status and also where the user is currently logged in.
2. The application shall include monitoring tools to track performance, detect anomalies, and log critical events for review.

Priority: High
Preconditions: None
Cross reference: None

4.1.13. Device Management

1. Users shall have the ability to view and manage devices associated with their accounts, enhancing security.

Priority: Medium
Preconditions: The user is logged in.
Cross reference: None

4.1.14. In-app Notifications

1. Users shall receive in-app notifications to stay informed about system updates, new features, and promotions.

Priority: High
Preconditions: The user is logged in.
Cross reference: 7.12

4.1.15. GDPR Compliance

1. The system shall adhere to the General Data Protection Regulation (GDPR), including user data protection and privacy rights.

Priority: High
Preconditions: None
Cross reference: 7.13

4.1.16. Data Portability

1. The system shall allow users to export their account data in a standard format, ensuring compliance with data portability regulations.

Priority: Medium
Preconditions: The user is logged in.
Cross reference: 7.14

4.1.17. Social Media Integration

1. Users shall have the option to link their social media accounts for personalized sharing or social transactions.

Priority: Low
Preconditions: The user has linked social media accounts (optional).
Cross reference: 7.15

4.1.18. Flexible Transaction Management

1. The system shall enable users to split a single transaction into multiple categories or recipients, providing flexibility in expense tracking and allocation.
2. Users shall have the capability to set up recurring domestic transactions for periodic payments, facilitating the automation of routine financial activities.
3. The system shall allow users to assign custom tags to transactions, enabling personalized categorization and efficient filtering for budgeting purposes.

Priority: High
Preconditions: The user is logged in.
Cross reference: 7.16.1, 7.16.2

4.1.19. Instant Notifications

1. The system shall send instant notifications to users upon the successful completion of domestic and foreign transactions, ensuring timely awareness.
2. Notifications shall dynamically display the impact of the transaction on the user's budget, allowing for immediate awareness of the financial consequences.
3. Users can set transaction amount thresholds to receive instant notifications only for transactions that exceed a specified amount, allowing for personalized alert management.

Priority: High
Preconditions: The user is logged in.
Cross reference: 7.17

4.1.20. Multi-Currency Wallet Management

1. Users have access to a multi-currency wallet, allowing them to hold and manage various foreign currencies seamlessly.
2. Users can set rules for automated currency allocation, specifying preferred currencies for specific types of transactions or expense categories.
3. Before initiating a foreign transaction, users shall be able to preview the impact on their multi-currency wallet, including potential currency conversion fees and resulting balances.

Priority: High
Preconditions: The user is logged in.
Cross reference: None

4.1.21. Real-Time Exchange Rate Alerts

1. Users receive real-time alerts on exchange rate fluctuations before confirming foreign transactions, facilitating informed decision-making.

Priority: High
Preconditions: The user is logged in.
Cross reference: 7.18

4.1.22. Clear Cross-Border Fee Disclosure

1. The system transparently discloses any cross-border fees associated with foreign transactions, ensuring users are fully informed about potential charges.

Priority: High
Preconditions: The user is logged in.

Cross reference: 7.19

4.1.23. Transaction Status Tracking and History:

1. Users have access to comprehensive transaction status tracking, including processing stages and transaction completion, providing transparency.

Priority: High
Preconditions: The user is logged in.
Cross reference: 7.20

4.2. System Quality Attributes

QA1 - Usability:

1. Users should be able to navigate through the application and perform common tasks without any confusion, achieving an average task completion time of under 1 minute.
2. The system should provide a responsive and intuitive user interface, ensuring a positive user experience.

QA2 - Performance:

1. The system response time for standard transactions should be less than two seconds to maintain user engagement.
2. Concurrent user support: The system should handle a minimum of 1000 simultaneous users without a significant degradation in performance.

QA3 - Reliability:

1. The system should have an uptime of 99.9%, ensuring minimal downtime and consistent availability.
2. In the event of a failure, the system should recover within five minutes to minimize service disruption.

QA4 - Security:

1. User data should be encrypted during transmission and storage to ensure confidentiality.
2. The system should implement multi-factor authentication to enhance user account security.

QA5 - Scalability:

1. The system should scale horizontally to accommodate a 20% growth in user base over the next year.
2. Database scalability: The system should handle a 30% increase in data volume without performance degradation.

QA6 - Maintainability:

1. Code changes and updates should be deployable with less than 5 minutes of downtime or impact on ongoing operations.

2. The system should have comprehensive documentation to facilitate maintenance and updates.

QA7 - Compatibility:

1. The application should be compatible with major web browsers (Chrome, Firefox, Safari, Edge) to support a diverse user base.
2. Mobile responsiveness: The system should provide a seamless experience on both desktop and mobile devices.

QA8 - Testability:

1. The system should have an automated testing suite covering at least 80% of the codebase to ensure robust and efficient testing.
2. Test data generation: The testing environment should allow for the creation of realistic and diverse test scenarios.

QA9 - Interoperability:

1. The system should integrate seamlessly with third-party payment gateways, ensuring an average transaction processing time of less than 5 seconds for secure and efficient transactions.
2. API compatibility: The application should expose well-documented APIs for integration with external systems.

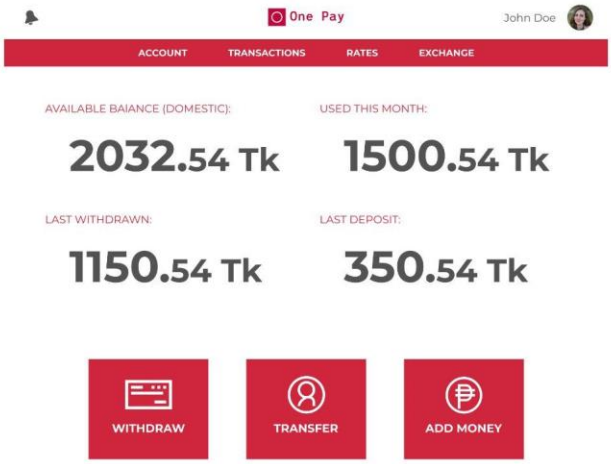
QA10 - Compliance:

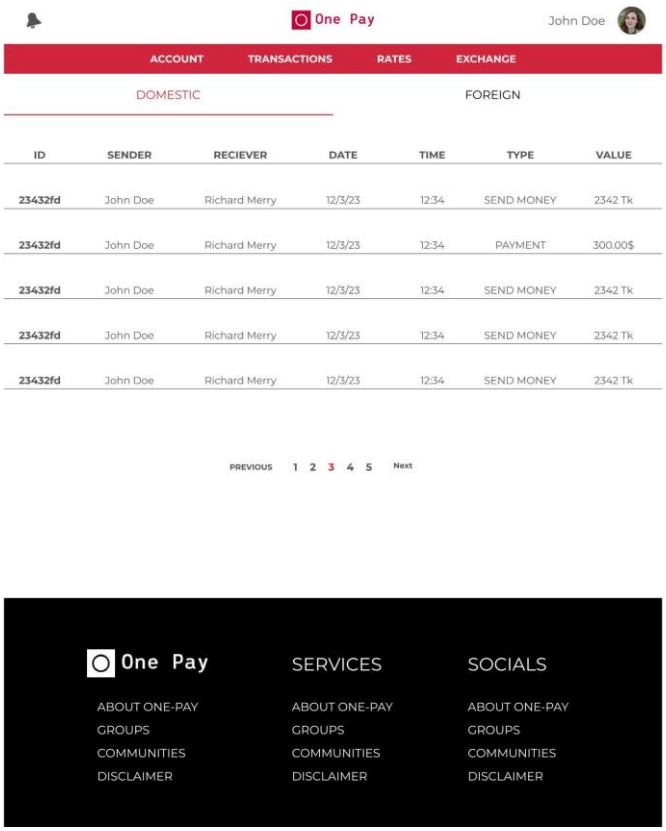
1. The system should comply with relevant data protection regulations (e.g., GDPR) to ensure user privacy.
2. Accessibility compliance: The application should adhere to WCAG guidelines to support users with disabilities.

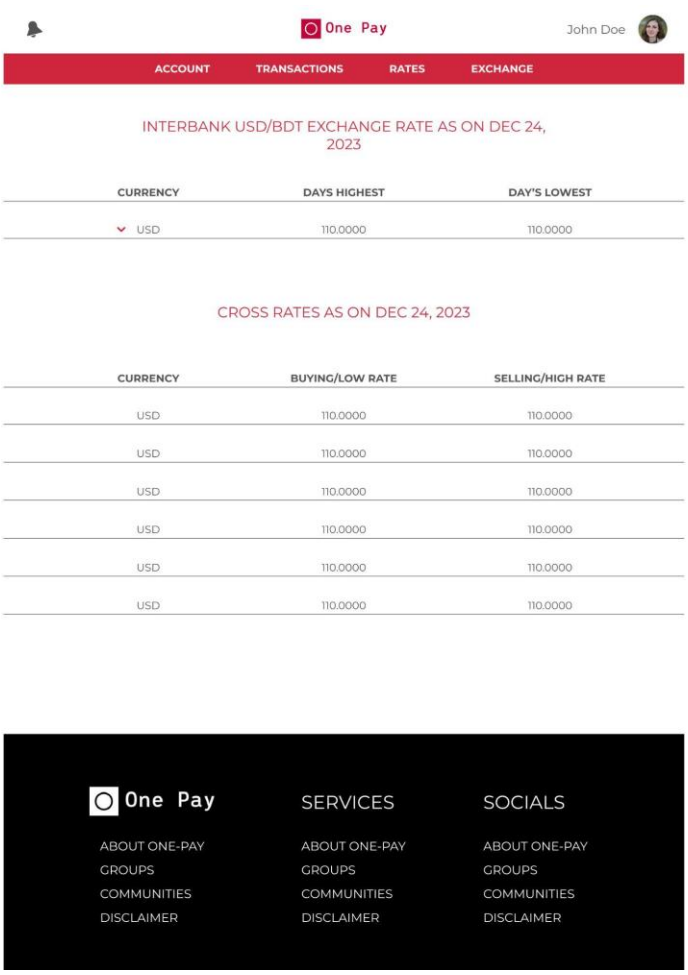
QA11 - Resilience:

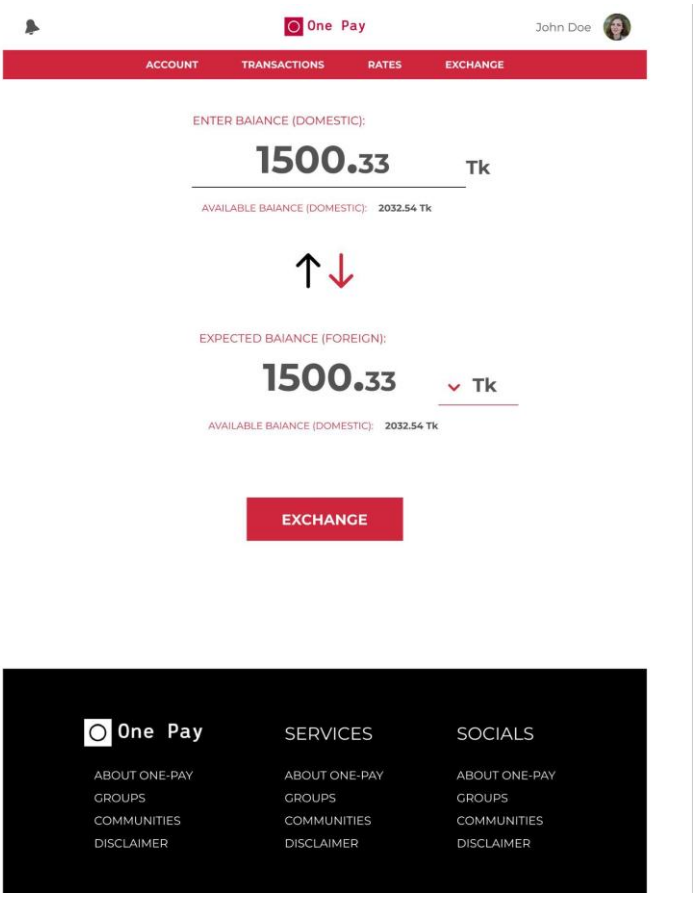
1. The system should be resilient to sudden traffic spikes, automatically scaling resources to handle increased loads without service degradation.
2. In the event of a server failure, the system should seamlessly switch to alternative servers within 30 seconds to maintain continuous service.

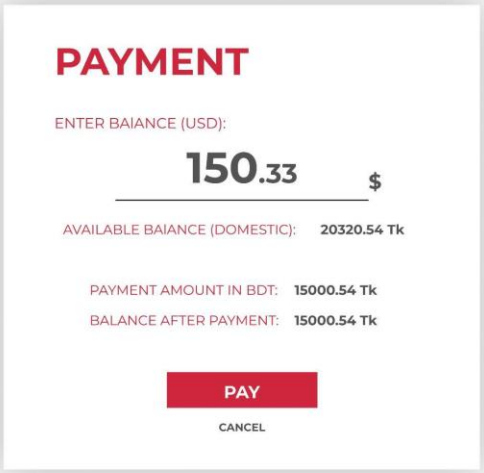
4.3 System Interface

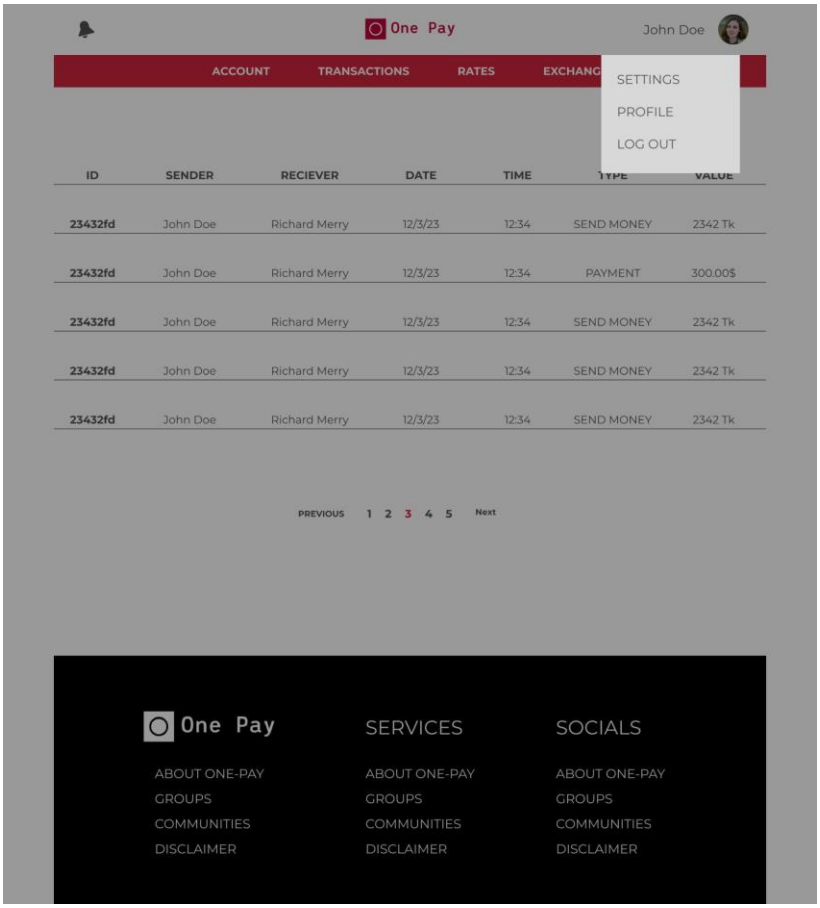


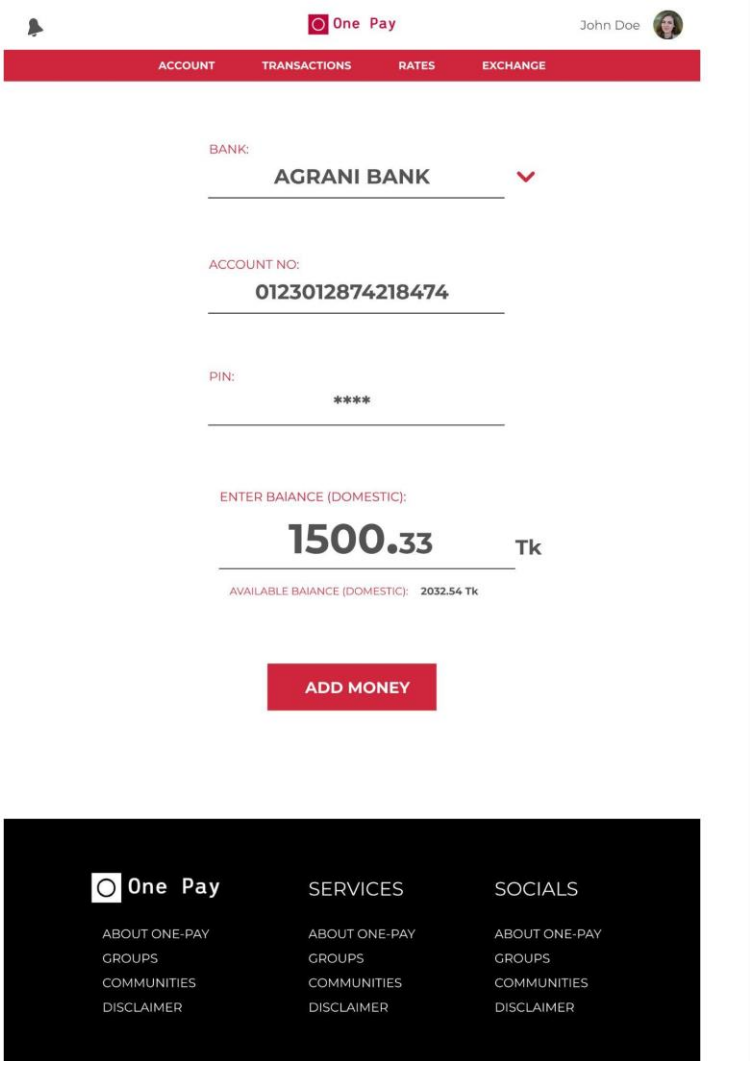

One Pay
John Doe
ACCOUNT
TRANSACTIONS
RATES
EXCHANGE
DOMESTIC
FOREIGN
ID SENDER RECIEVER DATE TIME TYPE VALUE
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
23432fd John Doe Richard Merry 12/3/23 12:34 PAYMENT 300.00$
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
PREVIOUS 1 2 3 4 5 Next
One Pay
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SERVICES
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SOCIALS
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER


One Pay
John Doe
ACCOUNT
TRANSACTIONS
RATES
EXCHANGE
INTERBANK USD/BDT EXCHANGE RATE AS ON DEC 24, 2023
CURRENCY
DAYS HIGHEST
DAY'S LOWEST
USD
110.0000
110.0000
CROSS RATES AS ON DEC 24, 2023
CURRENCY
BUYING/LOW RATE
SELLING/HIGH RATE
USD 110.0000 110.0000
USD 110.0000 110.0000
USD 110.0000 110.0000
USD 110.0000 110.0000
USD 110.0000 110.0000
USD 110.0000 110.0000
One Pay
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SERVICES
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SOCIALS
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER


One Pay
John Doe
ACCOUNT
TRANSACTIONS
RATES
EXCHANGE
ENTER BAIANCE (DOMESTIC):
1500.33
Tk
AVAILABLE BAIANCE (DOMESTIC): 2032.54 Tk
EXPECTED BAIANCE (FOREIGN):
1500.33
Tk
AVAILABLE BAIANCE (DOMESTIC): 2032.54 Tk
EXCHANGE
One Pay
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SERVICES
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SOCIALS
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER


PAYMENT
ENTER BAIANCE (USD):
150.33
$
AVAILABLE BAIANCE (DOMESTIC): 20320.54 Tk
PAYMENT AMOUNT IN BDT: 15000.54 Tk
BALANCE AFTER PAYMENT: 15000.54 Tk
PAY
CANCEL


One Pay
John Doe
ACCOUNT
TRANSACTIONS
RATES
EXCHANG
SETTINGS
PROFILE
LOG OUT
ID
SENDER
RECIEVER
DATE
TIME
TYPE
VALUE
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
23432fd John Doe Richard Merry 12/3/23 12:34 PAYMENT 300.00$
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
23432fd John Doe Richard Merry 12/3/23 12:34 SEND MONEY 2342 Tk
PREVIOUS 1 2 3 4 5 Next
One Pay
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SERVICES
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER
SOCIALS
ABOUT ONE-PAY
GROUPS
COMMUNITIES
DISCLAIMER



4.4. Project Requirements

4.4.1. Time

The project should be completed within a specified timeframe, with a target deadline of 8 to 12 months.

4.4.2. Budget

The project budget is limited to 700,000 BDT to 1,000,000 BDT, including all development, testing, and deployment costs.

4.4.3. Resources

1. Development Machines:
   High-performance computers or laptops for developers with the necessary software development tools.

2. Version Control System:
   A version control system like Git to manage and track changes in the source code.
3. Development Environment:
   Integrated Development Environments (IDEs) such as Visual Studio Code, IntelliJ, or Eclipse.
4. Database Server:
   Servers or cloud services to host the database, depending on your chosen database technology (e.g., MySQL, PostgreSQL, MongoDB).
5. Web Server:
   Servers or cloud services to host the web application, possibly using technologies like Apache, Nginx, or cloud platforms like AWS, Azure, or Google Cloud.
6. Continuous Integration/Continuous Deployment (CI/CD) Tools:
   Tools like Jenkins, Travis CI, or GitLab CI to automate the build, test, and deployment processes.
7. Testing Tools:
   Tools for automated testing, such as Selenium for browser automation, JUnit for Java, or pytest for Python.
8. Monitoring Tools:
   Tools to monitor application performance and detect issues, such as Prometheus, Grafana, or New Relic.
9. Containerization and Orchestration:
   Tools like Docker for containerization and Kubernetes for container orchestration.
10. Collaboration Tools:
    Communication and collaboration tools like Slack, Microsoft Teams, or communication channels within development platforms.
11. Documentation Tools:
    Tools for creating and managing documentation, such as Confluence, Markdown editors, or Wikis.
12. Security Tools:
    Tools to ensure the security of the application, such as vulnerability scanners, security auditing tools, and firewalls.
13. Backup and Recovery Tools:
    Tools for regular backups and recovery procedures, ensuring data integrity and availability.
14. Task and Project Management Tools:
    Tools like Jira, Trello, or Asana for managing tasks, tracking progress, and ensuring project organization.

15. Versioning and Release Tools:
    Tools for versioning and releasing software, such as Semantic Versioning, GitHub Releases, or similar platforms.
16. Collaborative Coding Platforms:
    Platforms like GitHub, GitLab, or Bitbucket for hosting code repositories, collaboration, and version control.

4.4.4. Environment

4.4.5. Documentation Standards:

1. Comprehensive documentation should be maintained throughout the project, covering design specifications, code documentation, and user manuals.
2. Documentation should follow industry best practices and be accessible to relevant stakeholders.

4.4.6. Quality Assurance:

1. The project should adhere to defined quality standards, ensuring that deliverables meet or exceed customer expectations.
2. Quality assurance processes, including testing and validation, should be integrated into each phase of the project.
3. Acceptance criteria for each project deliverable should be clearly defined in collaboration with stakeholders.

4.4.7. Project Estimation (COCOMO)

As the estimated lines of codes are 80,000 to 120,000, the project will be a Semi-Detached project. It is not possible to calculate with a range of values, so finding out a mean of the ranged value will be appropriate for this situation and that will be (120,000+80,000)=100,000. The calculation will be done based on this value.

Effort estimation:

$$E = 3.0(KLOC)^{1.12} \text{ PM}$$

$$E = 3.0(100)^{1.12} \text{ PM}$$

$$E = 521.34 \text{ person-months}$$

Duration estimation:

$$Tdev = 2.5(Effort)^{0.35} \text{ months}$$

$$Tdev = 2.5(521.34)^{0.35} \text{ months}$$

$$Tdev = 22.33 \text{ months} \approx 23 \text{ Months}$$

Average staff size:

$$SS = \frac{E}{D} \text{ persons}$$

$$SS = \frac{521.34}{22.33} \text{ persons}$$

$$SS = 23.34 \text{ persons} \approx 24 \text{ persons}$$

5. FEATURES NOT TO BE TESTED

1. Email Service Functionality: Since email is typically provided by a third party, testing the email service itself—that is, sending and receiving emails—might not be required. Making sure the system delivers the required emails as planned and successfully manages email-related issues or notifications should be the main priority.

2. External Currency Exchange Rates: Since the system displays the external currency rates, it may not be possible to verify their veracity directly. These rates were obtained from a third-party source. It is crucial to confirm that the system handles errors or discrepancies and displays the received rates accurately.

3. Real Banking Transactions: It's possible that no direct testing with actual money transactions is done. It is preferable to test transaction process simulations or emulations in a testing environment without real money transactions.

4. External Compliance Regulations Enforcement: The actual enforcement or verification of compliance with external rules may not be explicitly tested, even though the system is supposed to comply with KYC and AML laws. Rather, the emphasis would be on guaranteeing that the implemented checks and processes of the system comply with these standards.

5. Third-Party API Performance: It's possible that the functionality of external APIs offering currency rates or other services hasn't been tested. That being said, it is essential to confirm that the system manages answers, timeouts, and errors from these APIs.

6. Regulatory Documentation Compliance: Even though the system is made to comply with GDPR standards, it's possible that compliance with legal frameworks or actual legal documentation hasn't been tested. Instead, it is crucial to make sure that the elements of the system that have been put into place comply with the GDPR and that the way that user data is handled complies with the guidelines.

7. Social Media Platform Functionality: It's possible that testing hasn't been done to confirm the performance or functionality of third-party social networking sites such sharing or linking. Rather, it is crucial to make sure that the system interacts and integrates with these platforms in a way that is appropriate and secures data transfer or error management.

8. Live Communication with Banking Systems: It may not be possible to do direct testing with operational banking systems, particularly in a production setting. Rather, system behaviors ought to be validated in a testing environment that mimics banking interactions.

9. Physical Device Security: Software testing may not include verifying the physical security of the computers or mobile phones that users use to access the system. Securing user accounts and data within the system itself would be the main focus instead.

10.Impact of Foreign Exchange Rates on Actual Transactions: It's possible that the true financial impact of exchange rates on transactions has not been tested. It is imperative to confirm, though, that the system computes and displays the converted amounts to consumers appropriately.

6. TESTING APPROACHES

6.1 Testing Levels:

Unit Testing:

For Unit testing developers will write tests alongside code development to ensure individual components like functions or methods work as intended. These tests will be conducted using JUnit frameworks and will focus on validating isolated units of code. Obtaining high test coverage which includes both positive and negative scenarios while introducing tests into the Continuous Integration process for regular, automated validation is the goal.

Integration Testing:

For Integration testing, developers will aim to validate interactions and data flow between different modules or components of the system. Test scenarios covering several integration points will be conducted using the Selenium tool, with a focus on data transfers, API endpoints, and service interactions. Integration problems can be quickly resolved with the support of developers and constant validation during the development process.

System Testing:

For System testing the team will do end-to-end validation of the entire system's functionality. This involves using Selenium WebDriver for UI testing across a range of user situations. Thorough testing of functional and non-functional elements like usability, security, and performance is the aim. At the conclusion of every sprint, iterative testing verifies that the system is complete and takes input for future improvements into account.

Acceptance Testing:

Close coordination with stakeholders will be maintained during acceptance testing. This will involve developing test scenarios based on user stories and specifying acceptance criteria in line with business objectives. To make sure the system satisfies business objectives and user expectations, manual or automated testing will replicate real-world user scenarios. Constant stakeholder participation will guarantee alignment with changing business requirements, with acceptance criteria incorporated into user stories for every sprint.

6.2 TEST TOOLS:

1. Unit Testing Tools: JUnit
2. Integration Testing Tools: Selenium
3. System Testing Tools: Selenium WebDriver
4. Acceptance Testing Tools: Manual test scripts

6.3 MEETINGS:

Daily Stand-up Meetings (Daily):

Purpose: Discuss progress, challenges, and plan tasks for the day.

Duration: 15-20 minutes.

Attendees: Entire testing team (Testers, QA Leads, Test Managers).

Test Planning Meetings (Weekly):

Purpose: Define test objectives, scope, and strategy for the upcoming week.

Duration: 1 hour.

Attendees: Test Managers, QA Leads, Product Managers, Developers.

Test Case Review Meetings (Bi-Weekly):

Purpose: Review and validate test cases, ensuring coverage and accuracy.

Duration: 1 hour.

Attendees: Testers, QA Leads, Developers, Business Analysts.

Defect Triage Meetings (Weekly):

Purpose: Prioritize identified defects, assign ownership, and discuss resolution timelines.

Duration: 30-45 minutes.

Attendees: Testers, QA Leads, Developers, Product Managers.

Regression Testing Meetings (As Required):

Purpose: Coordinate regression testing efforts after significant system changes.

Duration: Varies based on regression scope.

Attendees: Testers, QA Leads, Developers.

Post-Implementation Review Meetings (Monthly):

Purpose: Assess testing strategies' success post-implementation, discuss lessons learned, and plan improvements.

Duration: 1-1.5 hours.

Attendees: Test Managers, QA Leads, Product Managers, Developers.

7. TEST CASES/TEST ITEMS

Table 1: Test Case for **Account Locking**

Project Name: OnePay	Test Designed by: Moshiur Rahman Nahin
Test Case ID: FR_1	Test Designed date: 27/11/2023
Test Priority (Low, Medium, High): Medium	Test Executed by:
Module Name: Login Session	Test Execution date:
Test Title: verify account locking after consecutive failed login attempts	
Description: Test account locking	
Precondition (If any): User must have valid username	

Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Enter username 3. Enter a wrong password 4. Click submit	Username: Nahin Password: 321 Password: 321 Password: 321 Password: 321 Password: 321	User should not login into the application and see a warning message including "Account Locked please try again with valid credentials"		
Post Condition: Users username must be registered and should be validated with the database.				

Project Name: OnePay				Test Designed by: Moshiur Rahman Nahin
Test Case ID: FR_2				Test Designed date:27/11/2023
Test Priority (Low, Medium, High): High				Test Executed by:
Module Name: Password Management				Test Execution date:
Test Title: Verifying password reset feature				
Description: Test password resetting				
Precondition (If any): The user is logged in or has access to the registered email address.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Click reset/forget password 3. Enter associated username/email 4. Click submit	Username: Nahin /email: smanhin1@gmail.com	User should get an email containing password reset link		
Post Condition: Users username and email must be registered and should be validated with the database.				

Table 2.1: Test Case for **Password Management**

Table 2.2: Test Case for **Password Management**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_2.2			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Password Management			Test Execution date:	
Test Title: Verifying password reset feature				
Description: Test password resetting				
Precondition (If any): The user is logged in or has access to the registered email address.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the email 2. Click password reset link. 3. Enter new password twice satisfying all conditions. 4. Click submit	Password: N@hin Confirm password: N@hin	User should see a message containing "Password reset successful"		
Post Condition: Users username and email must be registered and should be validated with the database.				

Table 3: Test Case for **Domestic Transactions**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_3			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Login Session			Test Execution date:	
Test Title: verify transaction using ATM card information				
Description: Test transaction page				
Precondition (If any): The user is logged in and has sufficient funds.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to home page 2. Go to transactions 3. Select recipient 4. Enter amount 5. Click confirm	Amount: 50	User should see a message containing "Transaction successful"		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 4: Test Case for **Transaction History**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_4			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Transaction History			Test Execution date:	
Test Title: verify Transaction details are showing or not				
Description: Test transactions history page.				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to home page 2. Select transaction 3. Click history		User should be able to see all his transactions he made previously.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 5: Test Case for **Foreign Transactions**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_5			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Foreign transactions			Test Execution date:	
Test Title: verify foreign transactions interface				
Description: Test foreign transactions page				
Precondition (If any): The user is logged in and has access to foreign transaction features.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to home page 2. Select transaction 3. Click foreign transaction		User should see a dedicated foreign transaction interface		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 6: Test Case for **Currency Exchange**

Project Name: OnePay	Test Designed by: Moshiur Rahman Nahin
Test Case ID: FR_6	Test Designed date:27/11/2023
Test Priority (Low, Medium, High): High	Test Executed by:
Module Name: Currency exchange	Test Execution date:
Test Title: Verify currency exchange in real time	
Description: Test currency exchange page	

Precondition (If any): The user is logged in, and foreign transaction features are accessible.

Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to homepage 2. Select transaction 3. Select foreign transaction 4. Choose local currency USD 5. Choose foreign currency BDT 6. Enter amount 7. Click confirm transaction	Local currency: 9.19 USD	Foreign currency: 1000 BDT Exchange rate: 108 taka Total cost: 10 USD		

Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.

Table 7: Test Case for **Compliance Measures**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_7			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Compliance Measures			Test Execution date:	
Test Title: verify compliance measures				
Description: Test foreign transactions page				
Precondition (If any): The user is logged in, and KYC information is up-to-date.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to homepage 2. Select transaction 3. Select foreign transaction 4. Give receivers details		After giving receivers details the system will proceed to payment confirmation		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 8: Test Case for **Secure Transactions**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_8			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Secure transactions			Test Execution date:	
Test Title: verify transaction process				
Description: Test website transaction page				
Precondition (If any): none				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website home page 2. Go to transactions 3. Do a transaction		User should see HTTPS protocol is used on the webpage link		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 9: Test Case for **Transaction Verification for High Amounts**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_9			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High):High			Test Executed by:	
Module Name: Transaction Verification for High Amounts			Test Execution date:	
Test Title: Verify high volume transaction verification				
Description: Test website transaction page				
Precondition (If any): The user is logged in and has a history of high-value transactions.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transactions 3. Select foreign transaction 4. Give receivers details & transfer amount	Transfer amount:301 USD Transfer amount:209USD	User should see a warning containing “high amount detected. Additional verification is required”		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 10: Test Case for **Monitor Logging**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_10			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Monitor Logging			Test Execution date:	
Test Title: verify logging monitoring				
Description: Test website monitoring system				
Precondition (If any): System admin is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Login		Admin should be able to see which user is logged in including time, date, OS and browser info. During the login how's the system performance.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 11: Test Case for **Device Management**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_11			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by:	
Module Name: Device Management			Test Execution date:	
Test Title: verify device management feature				
Description: Test website device management page				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to settings 3. Click privacy 4. Click device management		User should be able to see all the devices he used to login and can remove any device.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 12: Test Case for **In-app Notifications**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_12			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High):High			Test Executed by:	
Module Name: In-app Notifications			Test Execution date:	
Test Title: verify notification system				
Description: Test website notification				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to settings 3. Click notifications 4. Click turn on		User should get notifications from the system from now.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 13: Test Case for **GDPR Compliance**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_13			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: GDPR Compliance			Test Execution date:	
Test Title: verify GDPR Compliance				
Description: Test website GDPR Compliance				
Precondition (If any): None				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Click on privacy policy.		User should be able to see the system is adhere to the GDPR and GDPR is explained briefly on the page.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 14: Test Case for **Data Portability**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_14			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by:	
Module Name: Data Portability			Test Execution date:	
Test Title: verify data portability				
Description: Test website data portability				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Click on profile 3. Select share my profile.		User should be able to see his profile data using the link using any OS and browser in standard format.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 15: Test Case for **Social Media Integration**

Project Name: OnePay			Test Designed by:Moshiur Rahman Nahin	
Test Case ID: FR_15			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): Low			Test Executed by:	
Module Name: Social Media Integration			Test Execution date:	
Test Title: verify social media integration feature				
Description: Test website social media integration				
Precondition (If any): The user has linked social media accounts (optional).				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Click on profile 3. Click on add social media		User should see a social media login page asking for permission to add. After giving permission user should be able to do social transactions.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 16.1: Test Case for **Flexible Transaction Management**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_16.1			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Flexible Transaction Management			Test Execution date:	
Test Title: verify flexible transaction management feature				
Description: Test website transaction page				
Precondition (If any): The user is logged in and did a transaction.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transitions 3. Click on manage transaction 4. Select edit 5. Give input in category	Category: monthly internet bill	User should be able to see payment category is showing "Monthly internet bill"		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 16.2: Test Case for **Flexible Transaction Management**

Project Name: OnePay			Test Designed by:Moshiur Rahman Nahin	
Test Case ID: FR_16.2			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Flexible Transaction Management			Test Execution date:	
Test Title: verify flexible transaction management feature				
Description: Test website transaction page				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transitions 3. Click on manage transaction 4. Select autopay 5. Give information. 6. Click OK.	Autopay: Enabled Merchant id: 123 Amount: 1 USD	User should see a message containing "Autopay is enabled and merchant id:123 will get amount:1 USD monthly"		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 17: Test Case for **Instant Notifications**

Project Name: OnePay			Test Designed by:Moshiur Rahman Nahin	
Test Case ID: FR_17			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Instant notifications			Test Execution date:	
Test Title: verify instant notifications feature				
Description: Test website instant notifications				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transaction 3. Do a transaction		User should get a message in email and phone containing information about the transaction.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 18: Test Case for **Real-Time Exchange Rate Alerts**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_18			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Real-Time Exchange Rate Alerts			Test Execution date:	
Test Title: verify real-time exchange rate alerts feature				
Description: Test website transaction page				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transactions 3. Select foreign transaction 4. Give amount	Amount:1 USD Exchange rate:123 taka	User should see exchange rate is:123 taka		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 19: Test Case for **Clear Cross-Border Fee Disclosure**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_19			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Clear Cross-Border Fee Disclosure			Test Execution date:	
Test Title: verify cross-border fee disclosure feature				
Description: Test website foreign transaction page				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transactions 3. Select foreign transaction 4. Give amount	Amount:10 USD + Cross-Border Fee:1.2% Total:10.12 USD	User should see the total fee is:10.12 USD		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 20: Test Case for **Transaction Status Tracking & History**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_20			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Transaction Status Tracking and History			Test Execution date:	
Test Title: verify transaction status tracking and history feature				
Description: Test website transaction page				
Precondition (If any): The user is logged in.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transaction 3. Do a transaction 4. Click transaction history		The user should be able to see all the transactions with the details they provided.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 21: Test Case for **Usability**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_21			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by:	
Module Name: Usability			Test Execution date:	
Test Title: Verify functionality use of ease				
Description: Test website usefulness				
Precondition (If any): User must have valid username and password				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website 2. Enter username 3. Enter password 4. Click submit 5. Go to transaction 6. Click transfer fund 7. Give receivers details and id 8. Click proceed 9. Go back to homepage 10. Go to transactions 11. Go back to homepage 12. Go to transaction 13. Click pay bill 14. Give billing info 15. Click proceed 16. Click transaction history 17. Click logout		For every task user should not take more than 6 second. Users should be able to navigate through the application and perform common tasks without any confusion. The system should provide a responsive and intuitive user interface, ensuring a positive user experience.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

4.2 System Quality Attributes

Table 22: Test Case for **Performance**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_22			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by:	
Module Name: Performance			Test Execution date:	
Test Title: verify website performance				
Description: Test website responsiveness				
Precondition (If any): Users must be logged in				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Go to the website homepage 2. Go to transaction 3. Make any kind of transaction 4. Simultaneously initiate the login process for 1000 users and do transactions.		The system response time for standard transactions should be less than 2 seconds. The system should handle a minimum of 1000 simultaneous users without a significant degradation in performance.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 23: Test Case for **Reliability**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_23			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Reliability			Test Execution date:	
Test Title: Uptime and Availability Testing				
Description: To verify that the system achieves and maintains an uptime ensuring minimal downtime and consistent availability				
Precondition (If any): 1. The system is deployed in a production environment. 2. Monitoring tools are set up to track system uptime.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Monitor the system continuously over a defined period 30 days. 2. Simulate a server failure. 3. Simulate a network outage. 4. Simulate a database failure.		uptime =99.9% recovery time=5 minutes		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 24: Test Case for **Security**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_24			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Security			Test Execution date:	
Test Title: Security Features Testing				
Description: To verify that user data is appropriately encrypted during transmission				
Precondition (If any): The system is deployed in a secure environment. Encryption keys and multi-factor authentication settings are properly configured.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Log in as a user and initiate data transmission 2. Use a network packet analyzer tool to intercept and analyze the transmitted data		Data transmitted should be encrypted, and sensitive information should not be readable in plain text.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 25: Test Case for **Scalability**

Project Name: OnePay	Test Designed by: Moshiur Rahman Nahin
Test Case ID: FR_25	Test Designed date:27/11/2023
Test Priority (Low, Medium, High): Medium	Test Executed by:
Module Name: Scalability	Test Execution date:
Test Title: verify Horizontal Scalability Testing for User Base Growth	
Description: Horizontal Scalability Testing for User Base Growth	

Precondition (If any): 1. The system is deployed in a scalable and distributed architecture.

2. Monitoring tools are in place to measure system performance and resource utilization.

Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Gradually increase the number of simulated users to simulate a 20% growth over the existing user base. 2. Monitor system performance metrics during the user load increase.		The system should handle the increased user load		

Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.

Table 26: Test Case for **Maintainability**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_26			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Maintainability			Test Execution date:	
Test Title: Verify that code changes and updates can be deployed with minimal downtime or impact				
Description: Code changes and updates can be deployed with minimal downtime or impact				
Precondition (If any): The system is deployed in a production-like environment. The code repository and deployment pipelines are properly configured. Documentation is available and up-to-date				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Implement a minor code change and deploy it to the production environment. 2. Monitor the deployment process and system behavior during and after the deployment.		Down time=1 hour		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 27: Test Case for **Compatibility**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_27			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): High			Test Executed by:	
Module Name: Compatibility			Test Execution date:	
Test Title: verify website compatibility in browsers				
Description: Test website running in browsers				
Precondition (If any):				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Run the website in Chrome, Firefox, Safari and edge from Desktop and Mobile		The website is compatible with browsers.		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

Table 28: Test Case for **Testability**

Project Name: OnePay			Test Designed by: Moshiur Rahman Nahin	
Test Case ID: FR_28			Test Designed date:27/11/2023	
Test Priority (Low, Medium, High): Medium			Test Executed by:	
Module Name: Testability			Test Execution date:	
Test Title: Automated testing suite coverage				
Description: verify that the system has an automated testing suite covering				
Precondition (If any): The automated testing suite is set up and configured. Code coverage tools are integrated into the testing environment.				
Test Steps	Test Data	Expected Results	Actual Results	Status (/Fail)
1. Verify that the automated testing suite is properly configured. 2. Execute the entire automated testing suite. 3. Use code coverage tools to measure the percentage of code covered by the tests.		The testing suite should be set up without errors. code coverage=80%		
Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database.				

8. PASS FAIL CRITERIA

1. Account Locking

Pass: After giving invalid password consecutively 5 times the account locking should take immediate effect.

Fail: After giving invalid password consecutively 5 times the account is not locked.

2. Password Management

Pass: User should get a password reset link in the e-mail.

Fail: If user don’t get a password reset link in his/her e-mail. After going to password reset link

3. Domestic Transactions

Pass: If the payment is successful.

Fail: If the payment is not successful.

4. Transaction History

Pass: If the user able to see the previous transactions and details.

Fail: If the user is not able to see the previous transactions and details or anyone of them.

5. Foreign Transactions

Pass: If the user see the foreign transaction interface properly.

Fail: If the user can’t see the foreign transaction interface properly.

6. Currency Exchange

Pass: If the foreign currency shows accurate exchange rate, converted amount and total cost properly against the given local currency.

Fail: If any of the output value shows wrong a number.

7. Compliance Measures

Pass: After giving right information payment gets proceeded.

Fail: after giving wrong information payment gest proceeded or giving right information but payment not getting proceeded.

8. Secure Transactions

Pass: After going to the payment window the link contains HTTPS protocol.

Fail: After going to the payment window the link doesn't contains HTTPS or contains other protocol.

9. Transaction Verification for High Amounts

Pass: If the transfer amount crosses the minimum normal value=300 asking for additional verification/ if doesn't cross the minimum value additional verification is not asked by the system.

Fail: If any one of the condition is not met.

10. Monitor Logging

Pass: If the system keeps the track of user's login information.

Fail: If the system doesn't keep the track of the user's information.

11. Device Management

Pass: If the user able to see all the devices he used to login to the system.

Fail: If the user is unable to see all the devices he used to login to the system / if the user is unable to see any one of the devices he used to login to the system.

12. In-app Notifications

Pass: After enabling notifications user is getting notifications.

Fail: without enabling user getting notifications/ notifications is enabled but not getting notifications.

13. Data Portability

Pass: Profile data is showing in proper format.

Fail: Profile data is not showing/not is proper format.

14.Social Media Integration

Pass: User successfully login to the social media and able to do transactions.

Fail: User successfully login to the social media but unable to do transactions.

15.Flexible Transaction Management

Pass: Category name is showing what user has given.

Fail: Category name not is showing what user has given.

16.Instant Notifications

Pass: User getting transaction notifications in phone and email.

Fail: User is not getting transaction notifications in phone and email/ any one of them.

17.Real-Time Exchange Rate Alerts

Pass: Current exchange rate is showing in the interface.

Fail: Current exchange rate is not showing in the interface.

18.Clear Cross-Border Fee Disclosure

Pass: User can see total fee in details.

Fail: User can’t see total fee in details.

19.Transaction Status Tracking & History

Pass: User can see all the transactions and details.

Fail: User can’t see all the transactions and details.

20.Usability

Pass: Each task is completed within 6 second, Interface is user friendly,

Fail: Any task excesses 6 second to perform and interface is not user friendly.

21.Performance

Pass: System response time is within 2 second and handles 1000 simultaneous users without significant degradation.

Fail: System response time is more than 2 second and can't handle 1000 simultaneous users without significant degradation.

22.Reliability

Pass: minimum Uptime 99.9%.

Fail: minimum Uptime is less than 99.9%.

23.Security

Pass: Data transmitted in encrypted format.

Fail: Data is not transmitted in encrypted format/information is readable in plain format.

24.Scalability

Pass: The system handled the increased user load.

Fail: The system is failed to handle the increased user load.

25.Maintainability

Pass: Down time 1 hour.

Fail: Down time exceeds 1 hour.

26.Compatibility Testability

Pass: Successfully ran in Chrome, Firefox, Safari, and Edge from both desktop and mobile platform.

Fail: Failed to run in any one of the mentioned browsers from both platform.

27.Testability

Pass: Testing suite is set upped without errors and code coverage is minimum 80%.

Fail: Testing suite is set upped with errors / code coverage is less than 80%.

9. TEST DELIVERABLES

1. Test Plan
2. Test Data

3. Traceability Matrix
4. Defect Reports
5. Test Execution Logs
6. Test Summary Report
7. Performance Test Reports
8. Security Test Reports
9. User Manuals/Documentation
10. Test Closure Report
11. Training Materials
12. Compliance Reports
13. Metrics and KPIs

10. STAFFING AND TRAINING NEEDS

10.1 Recruitment:

As Horizontal Staffing, individuals with diverse skill sets will be hired to cover a broad range of testing types such as functional, usability and security.

As Vertical Staffing, experts on specific areas such as automation, performance and security testing will be hired.

Various channels like job boards, social media, professional networks and industry-specific forums will be utilized to attract suitable candidates. Partnering with recruiting agencies or attending job fairs to find potential candidates will also be considered.

Comprehensive interviews to assess technical skills, problem-solving abilities, teamwork, and communication skills will be conducted. The candidates will be evaluated through practical tests or simulations to gauge their testing capabilities.

1.2 Training:

An extensive orientation will be conducted to help the new members be familiar with the project's goals, testing methods, tools, and internal processes.

Ongoing training sessions or workshops will be offered to keep the team updated on the lates testing trends, tools, and methods. Participation in industry conference, webinars, and certification programs to enhance skill sets will be encouraged.

The team will be ensured to be proficient in using testing tools specific to the project's needs. Hands-on training sessions will be conducted or online courses will be provided to improve tool expertise.

Mentorship programs will be encouraged where experienced team members will mentor new hires. Also a culture of knowledge sharing through regular team meetings, presentations, or internal knowledge sharing sessions will be fostered.

11. RESPONSIBILITIES

The responsibilities of this project typically involve various stakeholders, each with distinct roles and duties throughout different phases of development, testing, deployment, and maintenance. Here's a breakdown of typical responsibilities for this project:

1. Project Manager:

- Overseeing the entire project lifecycle.
- Planning, scheduling, and coordinating project activities.
- Budget management and resource allocation.
- Communication facilitation among teams and stakeholders.

2. Development Team:

- Designing and developing the application based on specifications.
- Writing clean and efficient code that meets requirements.
- Conducting unit testing and ensuring code quality.

3. Testing/QA Team:

- Creating comprehensive test plans and strategies.
- Conducting various testing levels (unit, integration, system, acceptance).
- Reporting and tracking defects, ensuring their resolution.
- Verifying adherence to quality standards and requirements.

4. UX/UI Designers:
- Designing an intuitive and user-friendly interface.
- Conducting user research and creating wireframes/mockups.
- Ensuring consistency in the application's visual and interactive elements.

5. Security Analysts:
- Identifying potential security vulnerabilities.
- Conducting security assessments and penetration testing.
- Implementing security measures and best practices.

6. Compliance Officers:
- Ensuring the system complies with relevant regulations (GDPR, AML, etc.).
- Conducting audits and checks to verify regulatory compliance.

7. Database Administrators:
- Designing and managing the database architecture.
- Ensuring data integrity, security, and scalability.

8. Network Administrators:
- Setting up and maintaining the network infrastructure.

- Ensuring reliable system connectivity and performance.

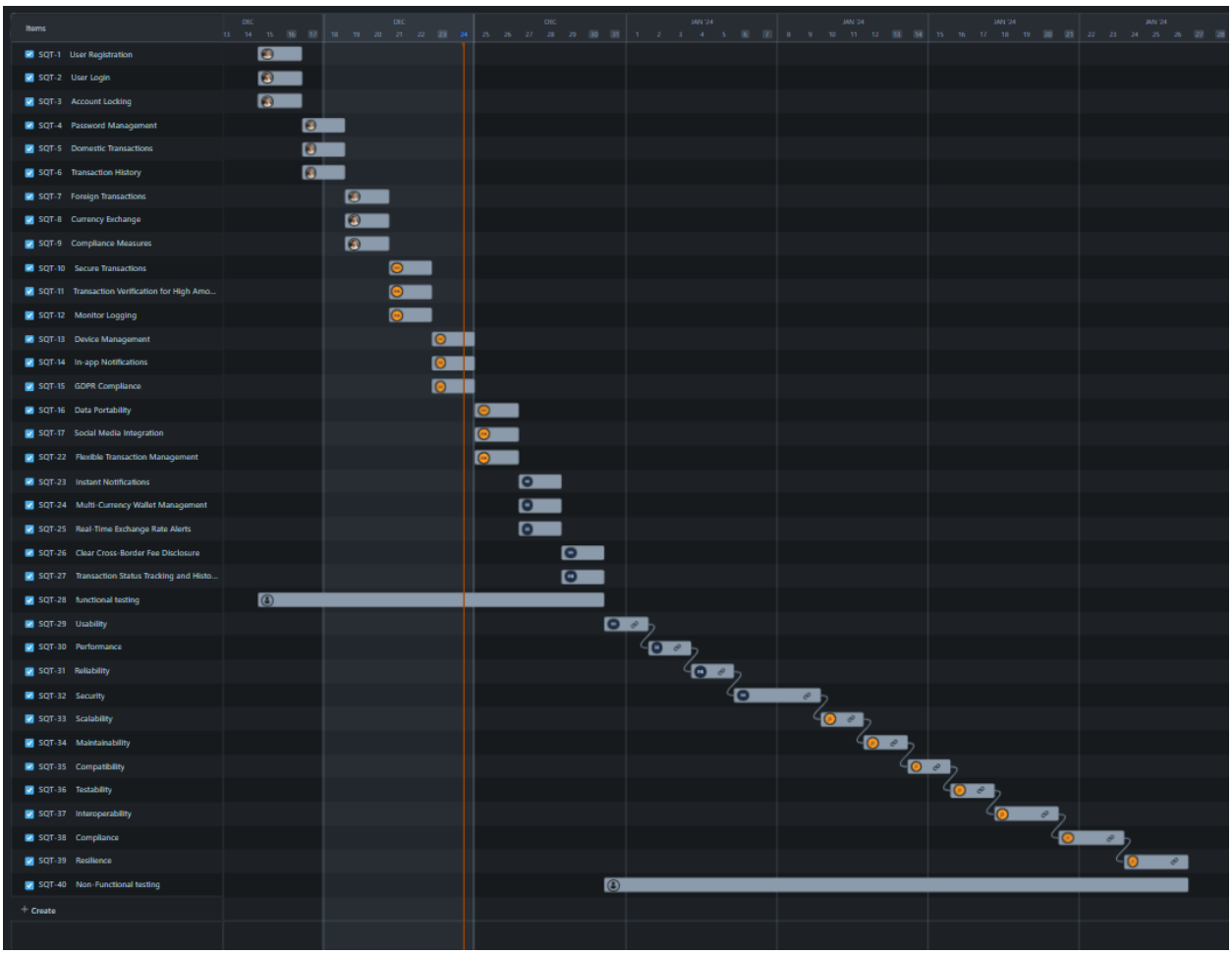
9. Customer Support/Helpdesk:
   - Providing technical assistance and support to end-users.
   - Addressing user queries, issues, and escalations.

10. Documentation Team:
    - Creating and maintaining technical documentation, user manuals, and guides.
    - Ensuring clear and comprehensive documentation for system components and processes.

11. Deployment/DevOps Engineers:
    - Managing deployment processes and infrastructure setup.
    - Implementing continuous integration and deployment pipelines.
    - Monitoring system performance and ensuring scalability and reliability.

12. Business Analysts/Product Owners:
    - Eliciting and documenting business requirements.
    - Prioritizing features and functionalities based on business value.
    - Acting as a liaison between stakeholders and the development team.

12. TESTING SCHEDULE



13. PLANNING RISKS AND CONTINGENCIES

Risk Mitigation Plan for testing

S/N	Risk Description	Probability	Impact	Mitigation Plan
1	Exchange Rate Fluctuations	25%	Financial loss due to rate changes	Keep an eye on currency fluctuations, think about hedging techniques, and, if at all feasible, set up contracts with fixed exchange rates.
2	Payment Delays	20%	Cash flow disruption	Clearly define the terms of payment, make use of dependable payment methods, and investigate the financial soundness of potential trading partners.
3	Regulatory Compliance Issues	15%	Legal and financial penalties	Maintain current knowledge of laws governing international trade, get legal counsel, and put in place reliable compliance procedures.
4	Political Instability in Foreign Markets	20%	Business disruption	Maintain awareness of geopolitical risks, diversify your market holdings, and prepare backup plans in case the market becomes unstable.
5	Cultural and Language Differences	15%	Miscommunication and errors	Ensure efficient communication routes, employ multilingual documentation, and make cultural training investments.

6	Technology Failures or Cybersecurity Breaches	18%	Data loss, financial theft	Put strong cybersecurity safeguards in place, upgrade IT infrastructure frequently, and carry out security assessments.
7	Supply Chain Disruptions	22%	Product delivery delays	Maintain buffer stocks, diversify your providers, and have open lines of communication with your vendors.
8	Inadequate Risk Assessment and Due Diligence	12%	Entering risky partnerships	Make careful risk assessments, investigate partners thoroughly, and, if needed, seek outside assistance.

14. APPROVALS

Position	Name	Signature
Project Sponsor	Jubayar, Anady, Nahin & Rafid	
Development Management		
Project Manager		
Test Manager		
Development Team Manager		