Test Plan For: TAKA
Prepared By: Promiti Dasgupta
Rushmia Ahmed

Version History

Version	Date	Change	Author	Approved by
0.1	2023-09-15	Initial Creation	Rushmia	Enisha Ashrefa
0.2	2023-09-29	Phase-2 Testable Feature and schedule added.	Promiti	Enisha Ashefa
0.3	2023-10-13	Phase-3 Testable Feature and schedule added.	Rushmia, Promiti	

Table of contents

1. Introduction

The project is a mobile financial service system in Bangladesh that allows users to perform a wide range of financial transactions using their mobile devices.

The project involves the development of this project, built entirely from scratch. Client-initiated project to develop various features using Laravel framework as the backend language, including registration, cash in, cashout, pay bill, mobile recharge,personal attributes,integration with E platforms and sending money. Progress is aligned with the schedule and approved phases, with the next step being to prepare the installer.

These Test Plans lay out how we will test the features of the mobile financial service system, "TAKA."The main goal of this document is to make sure everything is well-tested, including the quality plan,testing steps, schedule, criteria for approval, and reporting.

2. Reference

The following documents are used as sources of information for this test plan:

Ref. No	Document Title
1.0	Client Provided Figma Forms https://www.figma.com/file/%BF%EF%BC%89?node-id=530%3A0
2.0	Project plan Click here to view
3.00	http://redmine.companyname.com/redmine/documents/0000

3. High Level Test Objective

Our main goals are to examine this product for making features better, adding new ones, solving issues, and supporting ongoing maintenance tasks. We've outlined the following test objectives:

1. Confirm that the application's functionality aligns with specified requirements.

2. Verify that the product's output meets the defined requirements.

3. Proactively identify and address defects, ensuring prompt issue resolution.

4. Validate the completeness and accuracy of software requirements.

5. Develop comprehensive test plans for thorough testing coverage.

6. Provide transparent quality information to gain trust in the product's quality.

7. Formulate and document precise test scenarios and test cases.

8. Check how the app collects, processes, and responds to user feedback or error reports.

9. Evaluate how well the app behaves when it's running in the background, minimizing resource consumption.

4. Test Strategy

To ensure the utmost quality in the TAKA project, we will undertake a meticulous testing approach encompassing the following strategies:

UI/UX Testing Strategies:

We'll start testing by closely examining the app's design and user experience as soon as it's with the QA team. This helps us make sure the app's look matches what the client wants, and users can use it easily.

Pixel Perfection through Developer Tools:

We will closely examine the app's visual elements using mobile emulators or real devices. By making use of browser Developer Tools, we will confirm that each pixel meets the accuracy specified in the Figma design.

Integration Testing Strategies:

Test interactions between different features and components within the app.

Security Testing Strategies:

- Conduct penetration testing to identify potential vulnerabilities and security risks.
- Check that data is kept safe while being sent and stored, so user details are protected.

Performance Testing Strategies:

- Assess the app's responsiveness, load times, and overall performance under various user loads.
- Validate the app's stability and responsiveness during peak usage periods.
- Evaluate resource consumption and memory usage to ensure optimal performance.

Stress Testing strategies:

Stress testing is conducted in the TAKA mobile app to evaluate its performance under extreme user loads. By simulating high user activity, we can identify how the app responds, uncover potential performance bottlenecks, and ensure it can handle increased demand.

Compatibility Testing Strategies:

- Test the app on various devices, operating systems, and screen sizes to ensure compatibility.
- Verify that the app functions correctly across different versions of Android and iOS.
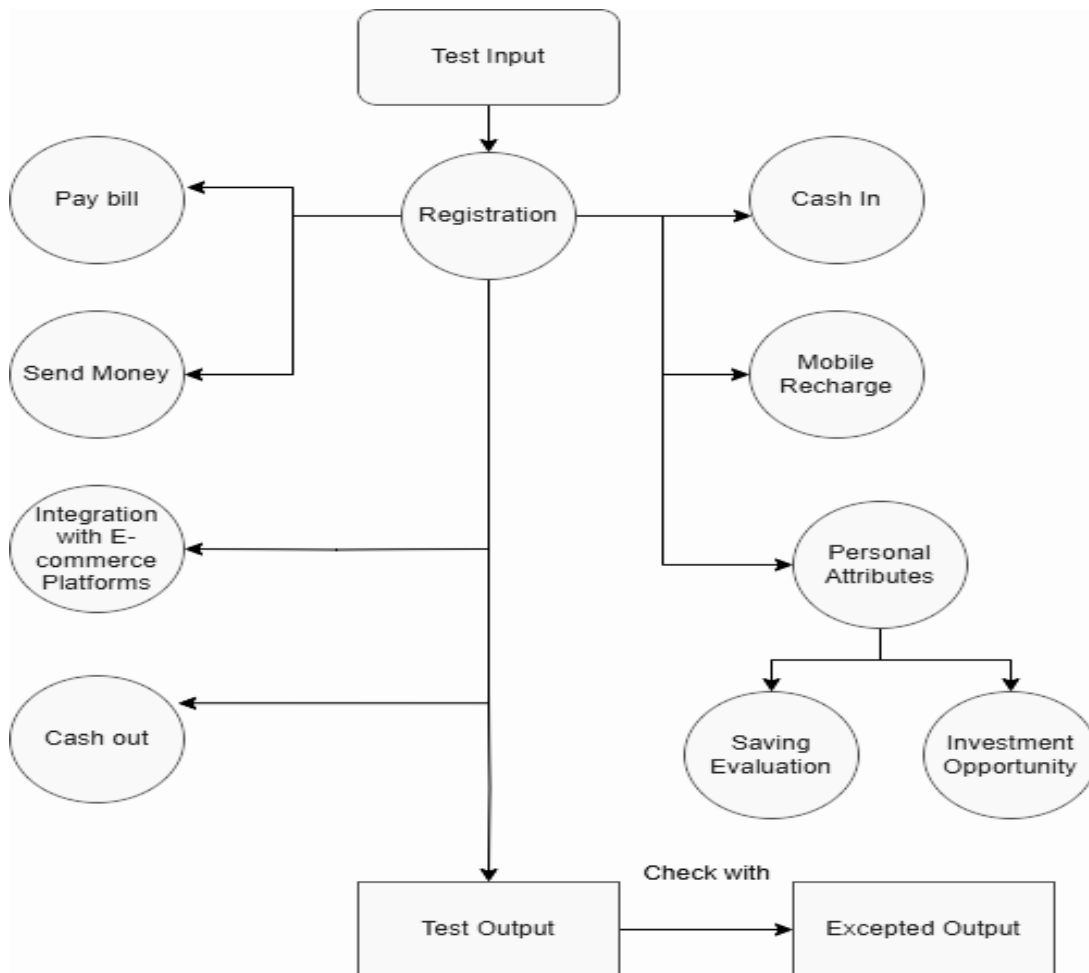
Fig 1: Test Strategy

5. Scope

The scope of the project includes testing the following features of 'TAKA' mobile application.

Inclusions:

1.Registration

       a. Phone Number Verification:
       b. Personal Information Entry
       c. Secure Password Creation
       d. Two-Factor Authentication
       e. Picture Upload for Profile Picture
       f. Account Type Selection
       g. Terms and Conditions Acceptance
       h. Confirmation and Activation
       i. Captcha Integration
       j. Terms and Conditions Acceptance

       m. Bank Account Verification

2.Cash in

- Bank Account Linking
- Credit/Debit Card Cash-In
- Cash-In Limits and Notifications
- Instant Updates
- Cash-In Reminder
- Cash-In via QR Code

3.Cash out

- Agent Cash-Out
- ATM Cash-Out
- Bank Transfer
- Utility Bill Payment
- Cash-Out Limits and Notifications
- Transaction History

4.Pay Bill

- Bill Selection
- Bill Amount Entry
- Payment Method Selection
- Bill Details Verification
- Payment Confirmation
- Payment Receipt
- Bill Due Date Reminder
- Bill Splitting
- Bill Comparison

5.Mobile Recharge

6.Send Money

7.Personal attributes

1.Spending history

- Category analysis
- Interactive Visualizations

2.Investment Opportunity

- Investment Tracking
- Education Hub

8.Integration with E-commerce Platforms

- Buy Now, Pay Later

From our understanding, we believe the above functional areas need to be tested.

Exclusions:

- All the features except that are mentioned under 'Inclusions'
- Test Automation

6. Features Dropped

- **International Money Transfer**

The "International Money Transfer" function will not be present in the TAKA app, as was our decision. According to our user surveys, the majority of users concentrate on local transactions because handling international transfers would result in more regulations and security issues.

- **Paying using a QR Code**

Despite the convenience that QR code payments provide, user feedback revealed that our target market prefers more conventional payment methods. We decided to give user preferences and needs-based features priority because QR code integration would call for more resources and user education.

7. Types of testing will be performed for the scoped items:

- **Functionality Testing**

Verify that each feature's functions work as intended, including user registration, cash transactions, bill payments, mobile recharges, and money transfers, etc. Confirm the app's ability to execute specific tasks and operations smoothly.

- **Integration Testing**

Validate interactions between different features and components. Confirm that data flows smoothly and accurately between various parts of the app.

- **Usability Testing**

Assess the user-friendliness of the app's interface. Evaluate the ease of navigation, layout, and overall user experience. Ensure that users can interact with the app intuitively.

- **GUI Testing**

We can confirm that buttons, menus, forms, and other interface elements are correctly displayed, aligned, and functional on various devices and screen sizes by performing GUI testing

- **Security Testing**

This app must undergo security testing to protect user information, financial transactions, and sensitive data, providing users with a safe and reliable platform to conduct various financial operations without jeopardizing their privacy or security.

- **Compatibility Testing**

Test the app on various mobile devices, operating systems, and versions. Confirm that the app functions correctly across a range of device configurations.

- **Installation Testing**

Installation Validation: Test software installation, reinstallation, and uninstallation processes.

Functional Integrity: Confirm that the application functions as expected after installation.

- **Performance Testing**

By conducting performance testing, this mobile app can manage a large number of users and transactions without slowing down or crashing, offering a smooth and dependable experience even during high usage periods.

- **Cookies Testing**

The TAKA app has to test its cookies frequently to ensure that user information, login sessions, and preferences are correctly stored in cookies.

- **Beta Testing**

Before the full launch, we conduct beta testing on the TAKA app to identify and address issues. Real users assist us in identifying problems, improving the app, and ensuring that it functions properly for everyone once it is formally released.

- **Retest**

Retesting will be conducted to verify the reported bugs are fixed or not. The relevant test cases will be re-executed to ensure the fix.

- **Regression Testing**

By running regression tests, we can catch any unexpected issues that might arise due to code changes, ensuring that the app remains stable and functional even after updates.

8. Types of testing not to be performed for the scoped items:

Testing Types	Here are the reasons for not conducting that testing at this time:
Legal and Compliance Testing	Making sure the app follows app store rules, but we're not doing deep testing for legal and compliance matters right now.
Beta Testing	While beta testing is valuable, it's not a top priority right now. We'll focus on establishing and testing core features internally first.
Network Disruption Testing	We're not testing network disruptions because it takes special tools and time, and we want to focus on more important testing areas. Also, since we have fewer users at the moment, the effects

	of network issues might not be major. As our user numbers increase, this testing could become more important.

9. Test Estimation

The effort required for testing the taka app can vary based on factors like:

- Quality of the testing foundation
- Size of the app
- How complex the app's functions are
- Need for documentation
- Time constraints
- Number of defects and rework needed
- How much regression testing is required?

10. Release Procedure

Below procedures will be followed for Release:

Step-1:Understand what the app needs.
Step-2:Build and test it within the team.
Step-3: Execute test cases and report issues if any.
Step-4:Fix the reported issues.
Step-5:Retest to confirm that no issue got left.
Step-6:Prepare the final version for this phase.
Step-7:Make sure the previous bugs are fixed.
Step-8:Test everything again and note any more issues.
Step-9:TAKA will go release to the production environment if all severe bug's status is low.
Step-10:Fix any leftover issues in the next phase.

Step-11:The QA team conducts post-release validation to ensure the deployed code functions correctly in the production environment.

11. Test Suspension Criteria

Suspension in the software testing process in which the testing team will suspend the testing activities based on some criteria.

- Test resources are not available when needed.
- Instability in the app's testing environment.
- Security vulnerability is identified that poses an immediate risk to the user data of TAKA app.
- Prioritization of higher-priority testing activities.

- Delays due to pending approvals or necessary changes.
- Resolution of complex technical challenges.

After the testing process gets suspended, the testing team should immediately provide the criteria of suspending the flow of testing to the project manager.

To restart the process which is invoked after the suspension criteria are met, following conditions have to be met:

- Issue due to which suspension occurs gets resolved.
- After the security vulnerability is validated.
- Through retesting, the highly impacting defect is resolved.

12. Test Acceptance Criteria

In order to make sure that the user story is finished in accordance with the specifications, a list of requirements known as the acceptance criteria is prepared.

- The system should have availability of no less than 90%.
- The six integrated modules should successfully communicate and integrate without serious issues.
- Deliverable documentation should be complete and correct.
- There are no significant security flaws left in the software.
- The system should categorize each user as trusted or not trusted.

13. QA Task List and Testing Process

- Examine the specifications
- Arrange kick-off meeting

- Form clear testing goals
- Define the scope of testing
- Choose the right tools and technologies
- Identify the types of testing
- Select the right team
- Creating a test plan
- Perform thorough testing
- Document the testing process
- Track defects discovered during testing
- Report and communicate through daily morning meeting and specification grooming meeting
- Continuous testing of TAKA at frequent intervals throughout the development process
- Release readiness assessment

14. Test Environment

Hardware Requirement	**Mobile Devices**:Android And IOS Emulators
Software Requirement	**Operating Systems**: Android, iOS. **Test Case Management**: Google Sheets. **Document Management**: Google Drive. **Design Tool:** Figma. **Emulator**
Network Requirement	Internet connectivity to the devices.
Device Oriented Testing	We'll test the app on various platforms using different devices and screen resolutions: **Platform:** Android **Devices**: Android smartphones **Screen Resolutions**: Varied sizes based on devices (e.g., 1080 x 1920, 720 x 1280) Platform: iOS **Devices**: iOS android Emulator Screen Resolutions: Adjusted for device

	specifications (e.g., 768 x 1024)

15. Test Schedule:

Feature Name	No of Test Items	Test Case Design	Internal QA Deployment	Final Release
KT + TC Writing Execution Retesting for Registration	10	80	08.09.23-15.09.23	TBA
KT + TC Writing Execution Retesting for Cash In and Cash Out	10	90	15.09.23-22.09.23	TBA
KT + TC Writing Execution Retesting for pay bill	10	70	22.09.23-29.09.23	TBA
KT + TC Writing Execution Retesting for Mobile Recharge And Send Money	12	60	29.09.23-06.10.23	TBA
System testing +issue reporting +Retesting	12	40	09.10.23-13.10.23	TBA
Personal Attribute	18	30	13.10.23-30.10.23	TBA
Integration with E Platforms	6	30	30.10.23-12.11.23	TBA

Total	78	400		

16. QA Summary Report

Phase	URL
1st Release (V-1.10)	http://redmine.com/redmine/issues/000000
2nd Release (V-1.20)	http://redmine.com/redmine/issues/000000
3rd Release (V-1.30)	http://redmine.com/redmine/issues/000000

17. Roles and Responsibilities

Resource Name	Roles and Responsibilities
Arifa Akter	General Manager
Enisha Ashrefa	Project Manager
Takia Maliha	BE
Nigah Hossain	Software Architect
Rana Tabassum	

Rushmia Ahmed	Software Engineer
Sehrish Zeba	
Promiti Dasgupta	QA

18. Risk and Contingencies

Schedule

- If SRS are not Cleared as per schedule, then we may not be able to meet the Testing deadline.

- If Releases for Testing are not provided as per schedule, then we may not be able to meet the deadline.

- Any changes to the requirements/scope could affect the test schedule.

Testing

- Insufficient Test Environment Stability: Frequent environment changes may lead to inconsistencies in testing results and hinder progress.

Application Risk

Regulatory Non-compliance Risk

Operating TAKA without comprehensive legal and compliance testing might lead to regulatory violations and potential legal actions, loss of user trust and reputation damage.

- **Beta Testing Delayed**

Launching TAKA without comprehensive beta testing might result in undiscovered issues, poor user experience and increased post-launch bug fixing.

- **Network Disruption Testing Skipped**

Not testing TAKA's behavior during network disruptions could lead to poor user experience or potential security vulnerabilities and negative app reviews.

19. Test Entry Criteria

Test entry criteria define the conditions that must be met before testing activities can begin. These criteria ensure that the testing environment and the application are ready for testing:

- Complete or partially testable code is available.
- Requirements are defined and approved.
- The test plan, detailing the testing strategy, scope, objectives, and resources, should be approved.
- Test cases are developed and ready.
- Test environment has been set-up and all other necessary resources such as tools and devices are available.

20. Test Exit Criteria

Test exit criteria define the conditions that must be met before testing activities can be concluded. These criteria help ensure that the TAKA application has been adequately tested and is ready for release:

- Deadlines meet or budget depleted.
- Execution and updating of all test cases.
- Desired and sufficient coverage of the requirements and functionalities under the test.
- All the identified defects are corrected and closed
- No high priority or severity or critical bug has been left out.

An exit review meeting should be conducted to assess if the exit criteria have been met and determine the readiness for release.

21. Bug Status Explanation

We maintain following status of the **Bug** in our **Test Execution** report:

New: When any defect or bug is found and validated, the status of bug or defect is logged in as To-do.

In-Progress: When assigned developer working on the respective ticket.

Done: This status means that the issue is fixed, Now tested can retest.

Verified: The tester re-tests the bug after it got fixed by the developer. If there is no bug detected in the software, then the bug is fixed, and the status assigned is "verified."

Reopened: If the bug persists even after the developer has fixed the bug, the tester changes the status to "reopened". Once again, the bug goes through the life cycle.

Closed: If the bug no longer exists then the tester assigns the status "Closed."

Duplicate: If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to "duplicate."

Rejected: If the developer feels the defect is not a genuine defect then it changes the defect to "rejected."

Deferred: If the present bug is not of a prime priority and if it is expected to get fixed in the next release, then status "Deferred" is assigned to such bugs

Not a bug: If it does not affect the functionality of the application then the status assigned to a bug is "Not a bug".

22. Test Deliverables

Test Deliverables is a list of all the documents, tools and other components that have to be developed and maintained in support of the testing effort.

Followings are the deliverables from QA for TAKA project:

Test Deliverables before Testing

- Test Specification Document
- Test Design Document
- Test Plan Document
- Functional Specification Documentation
- Test Management Plan Document

Test Deliverables during Testing

- Test Scripts
- Test Data
- Test Traceability Matrix
- Error logs and execution logs

Test Deliverables after Testing

- Test execution report
- Defect report
- Installation/Test procedures guidelines
- Release Notes
- Acceptance Test Report

- Coverage Analysis Report

23. Test Plan Approvals

Name	Role	Signature	Date
Enisha Ashrefa	Project Manager		2023-09-08
Enisha Ashrefa	Project Manager		2023-09-15
Enisha Ashrefa	Project Manager		2023-10-13