Test Plan for Bikroy.com

Prepared by: Jakia Sultana

Table of contents

Introduction

The approach, goals, standards, setting, materials, timetable, and approximation of testing tasks are described in the test plan for the Bikroy.com web application. In addition to addressing the risks related to the test plan and outlining the mitigation strategies for each risk mentioned, it also specifies the testing scopes, including features that must be tested and as well as the types of testing, tools, environments, and individuals for testing.

Overview of Bikroy E-commerce Platform

In Bangladesh, Bikroy.com is a well-known online marketplace that makes it easier to purchase and sell a variety of goods and services. An outline of its features, navigation, and user flow is provided below:

1.Easy browsing with filter-based search functionality.
2. Detailed product listings that include pictures and text.
3. Listings and correspondence are managed using seller accounts.
4. Use of messaging apps for communication in case of questions.
6. User profiles include evaluations
7. Security protocols for safe and secure transactions.
8. Making websites and applications mobile-friendly with responsive design

Navigation:
   The website features a user-friendly navigation system with a top bar for major categories like Jobs, Electronics, Vehicles, Real Estate, Education, and Services along with chat, ad and account. Users may quickly browse through various product or service categories by using drop-down menus to reach subcategories.

User flow:

Purchasing Procedure:

1. The user accesses the homepage and browses the categories.
2. The user peruses listings or adds filters to enhance search results.
3. To see further information, such as pictures and descriptions, the user clicks on a listing.
4. The user chats with the vendor if they are interested in negotiating a price or with queries.
5.After an agreement is reached, the user makes arrangements with the supplier for delivery and payment

Selling Procedure:

1.The seller either registers a Bikroy.com account or connects in to an existing one.
2.The seller enters specific information about the goods or service they want to list.
3.The seller includes high-quality photographs and sets a competitive price for the item.
4.The seller posts the listing, making it visible to potential purchasers.
5.Seller replies to buyer inquiries, gives extra information, and negotiates prices as needed.

6.Once a customer is located, the vendor makes arrangements for payment and delivery of the item/service.

Test Objectives

Overall, Bikroy.com offers a user-friendly platform for shopping and selling a wide range of goods and services in Bangladesh. Since we will be evaluating the application's quality and fulfillment, these are the primary goals of the test plan:

-Validate the application's functionality to ensure alignment with requirements.
-Proactively identify and rectify defects while emphasizing issue resolution.

-Gain customer confidence by delivering a product of exceptional quality.

-Ensure that the software is user friendly.

-Evaluate the application's performance under varying load conditions to ensure it meets performance benchmarks and provides a responsive user experience.

Test Scenarios

The website's easy navigation, full search capabilities, and robust features ensure a flawless experience for both buyers and sellers.Here,we create a list of test scenarios for each of the application's features, such as :
- Login and registration of users
- Search and selection of items
- Add or Remove post
- Chat option
- Payment processing
- User profile management

Test Cases

Create extensive test cases for each scenario, including processes, expected results, and acceptance criteria.

Test Strategy

To ensure the quality of the "Bikroy" project's applications, testing will be implemented using the following strategies:

- **UI/UX Testing Approach**: The initial testing phase involves conducting UI/UX testing within the QA team. This ensures that the application's UI aligns with client requirements and provides an intuitive user experience as intended.

- **Pixel Precision**: We will employ the browser's developer tools to achieve pixel-perfect alignment with the Figma design, ensuring accurate visual representation.

- **Functional Testing Approach**: Functional testing is performed subsequent to the creation of test cases for each functional aspect, verifying that each component operates as expected.

- **Integration Testing Approach:** This level of testing evaluates interactions and interface among integrated components within the project's scope, ensuring seamless integration as developed by our company.

- **API Testing Approach**: If the project incorporates APIs, dedicated testing will concentrate on API functionality, security, and performance. This testing guarantees smooth communication between distinct software elements.

- **Cross-Browser Testing Approach**: The application undergoes testing on multiple web browsers to confirm consistent performance and appearance across diverse browser platforms.

- **Performance Testing Approach**: This approach focuses on assessing responsiveness, speed, scalability, and stability under varied load conditions. Its purpose is to ensure optimal application performance, even during periods of high user activity.

- **Usability Testing Approach:** Usability testing evaluates the application's user-friendliness and navigational ease. It identifies design and usability issues that could impact the overall user experience.

- **Retesting Strategy**: After functional testing reveals issues, a retesting strategy is employed. Once the issues are rectified, the application undergoes retesting to confirm the resolution of identified problems.

- **Regression Testing**: With the addition of new elements, regression testing is executed on pre-existing functionalities to ensure that new inclusions have not adversely affected the existing components.

Test Types

We will conduct a variety of testing methods to ensure the quality of the application:

- **Functional Testing**: Functional tests will be executed to cover both positive and negative scenarios. The objective is to ensure that the application aligns with its Scope Statement and performs as intended.

- **GUI Testing:** This testing phase will comprehensively evaluate the graphical user interface of the application. The aim is to ensure the flawless implementation of the specifications and to test the interaction between the application and the user. This includes assessing how the application handles user input and displays various graphical elements such as text, images, buttons, menus, dialog boxes, icons, and toolbars.

- **Pixel Perfection:** Pixel perfection testing will encompass the following aspects: A. Evaluating responsiveness. B. Verifying alignment, which includes margins, padding, spacing between images and other content, positioning, and placement. C. Ensuring image ratios are accurate on desired screens, icons, and image resolutions. D. Checking font attributes like size, color, style, line spacing, and character spacing. E. Scrutinizing icon and button design for meticulous craftsmanship, proper sizing, and spacing.

- **Performance Testing**: This evaluation will test the application's performance under varying conditions, including different user loads and network speeds. It encompasses load testing to assess performance under heavy user loads, stress testing to push the application beyond its capacity, and responsiveness testing to measure response times.

- **Security Testing**: Security testing will focus on pinpointing vulnerabilities and weaknesses in the application's security mechanisms. This involves techniques such as penetration testing to simulate hacking attempts and verifying the integrity of secure authentication and authorization processes.

- **Usability Testing:** Real users will engage with the application to evaluate its user friendliness and intuitiveness. Feedback will be collected on any challenges encountered during task completion, aiding in the enhancement of the overall user experience.

- **Compatibility Testing:** We will verify that the application functions correctly across various devices,operating systems, browsers and screen resolutions. Ensuring a consistent user experience across different environments is of utmost importance.

- **Retest:** We will perform retesting to confirm the resolution of reported bugs. Relevant test cases will be re-executed to ensure that the fixes have effectively addressed the identified issues.

- **Accessibility Testing:** Accessibility testing guarantees that the application is accessible to individuals with disabilities. This includes verifying compatibility with assistive technologies and adherence to accessibility guidelines.

- **Integration Testing:** We will conduct integration testing to verify the smooth interaction of different components or modules within the application. It ensures accurate data exchange and identifies potential integration issues between various parts of the app.

- **Regression Testing**: This form of testing involves reevaluating the entire application whenever new features are added or changes are implemented. The objective is to ensure that these modifications haven't introduced new defects or negatively impacted existing functionalities.

- **Exploratory testing**: Testers also perform exploratory testing to uncover unexpected issues and edge cases that may not be covered by predefined test cases.

- **Automated testing**: By automating registration,search,ad management features bikroy can streamline testing processes, reduce manual effort, and ensure consistent quality across its platform.

Features to be Tested

In "Bikroy", these features are to be tested.

Phase 1	Phase 2	Phase 3
UI/UX of Start up page Registration page Login page.	Add/Remove post Item/Services Side menu bar	Profile settings Make Payment Chat

Test Estimation

Factors Affecting Test Estimation:

- Product Scope and Size
- Problem Domain Complexity
- Documentation Requirements
- Time Constraints
- Defect Count and Rework

Release Procedure

To Release the product, we will follow below procedures:

Step 1: Requirement Analysis

Analyzing project requirements to establish a clear understanding of scope and objectives.

Step 2: Development and Internal Release for QA

Beginning development based on the analyzed requirements. Create an internal release for Quality Assurance (QA) testing.

Step 3: QA Testing and Bug Reporting

The QA team rigorously will test the internal release. Identify and report any issues or bugs encountered during testing.

Step 4: Bug Fixing by Developers

Development team will address and fix reported bugs.

Step 5: Final Release for the Current Phase

We will prepare and release the final version of the app for the current development phase.

Step 6: QA Verification of Bug Fixes

QA will validate that the reported bugs from the internal release have been successfully fixed in the final release.

Step 7: Comprehensive Phase Testing and Bug Recording

The QA team will perform comprehensive testing of the entire phase, recording any remaining bugs or issues.

Step 8: Release Evaluation

QA team will assess the status of the project: if there are no blocking issues and low-severity bugs remain, proceed with the release. Otherwise, the release will be postponed until issues are resolved.

Step 9: Future Phase Bug Resolution

Any outstanding bugs that were not addressed for the current phase will be prioritized for resolution in the next development phase.

Test Suspension Criteria

The Quality Assurance (QA) team will halt testing and decline to approve the receivables if any of these below events get encountered:

- Any showstopper bug found while testing.

- Bug has not been fixed or addressed in the respective release.

- The release has been made without the provision of release notes to provide essential information to users and stakeholders.

Test Acceptance Criteria

These are the test acceptance criteria for "Bikroy" project.

Application UI Alignment with Figma UI Specification:
The application's user interface (UI) should precisely match the design outlined in the Figma UI Specification provided at the link.

No High-Priority Defects:
The application should not have any high-priority defects that would hinder its usability, security or performance.

Absence of Blocking Issues:
The application must not have any critical or blocking issues that hinder its core functionality or user experience.

Test Case Coverage of 90%:
The application should be tested extensively, with test cases covering at least 90% of the identified scenarios and functionalities.

Performance Standards:
The application should fulfill predetermined performance criteria for response times and loading speeds. Data retrieval and user interactions should happen in reasonable amounts of time.

Browser Compatibility:
The application should be compatible with the following supported browsers:Mac,Windows chrome etc.

QA Task List and Testing Process

QA Team will be conducting the following tasks:

● Requirement analysis and knowledge transfer
● Identify Test areas
● Test Case writing on identified test areas
● Prepare Test environment
● Test Cases Execution
● Bug reporting/retest
● Deliver Test report
● Perform Test closure activity
● Daily morning meeting

Test Environment

To ready the test for the Taka project, the following prerequisites are necessary:

Hardware Requirement:

PC (Widows)
    Android,
    Iphone
    iPad

Software Requirement:

    Operating System:
    AndroidOS11orafter,iOS8orafter Microsoft
    Office

Network Requirement:
Internet connectivity for mobile phones.

Tools to be used:

Test Case management: Microsoft Excel/Google Drive
Document management: Google Drive
Project management: Redmine/Jira
Design Tool: Figma, Balsamiq
UI/Pixel Perfections Test Tool: Browsers' Development Tolls.

Schedule

Within feature release per plan, schedule may get varied / updated:

Phase	Feature Name	Req. Analysis	No of Test Items	Test Case Design	Internal QA Release	Final Release
1	UI/UX of T Start up page, registration page and login page.		3	26	2023-07-10	Test case
2	Complete UI/UX for Startup page, Add/Remove post		3	26	2023-08-05	Test case
3	Product Cart Side menu bar		4	26	2023-09-10	Test case
4	Profile settings Make Payment Chat		8	26	2023-09-30	Test case

QA Summary Report

The timetable will be revised in accordance with the roll out of each phase's features:

Phase	URL
1st Release(V-1.0.0)	QA Report
2nd Release(V-1.0.1)	QA Report
3rd Release(V-1.0.2)	QA Report
4th Release(V-1.0.3)	QA Report

Resource Allocation:

Name	Roles	Responsibilities
XYZ	Project Manager	Project planning, Resource allocation, Time management
XYZ	QA Lead	Planning strategy , Team management
XYZ	Internal Tester	Integration Testing ,Load Testing,Regression Testing
XYZ	Internal Tester	System Testing, Performance Testing ,Security Testing

Risk and Contingencies

Identify potential risks that could impact the testing and overall project.

Data Security Breach

- The application deals with sensitive financial data. A security breach could lead to data compromise and financial losses.

Performance Degradation under Load

- High user load might lead to app slowdowns or crashes during peak usage times.

Lack of Usability

- Poor user experience due to complex user interface or confusing navigation.

Schedule:

- Failure to clear SRS as scheduled might hinder us from meeting the testing deadline.
- In case Releases for Testing aren't provided per the schedule, it could jeopardize meeting the deadline.
- Modifications to requirements/scope may potentially impact the test schedule.

Testing:

- The introduction of new features and function enhancements could inadvertently lead to side effects and disrupt the existing functionality due to inadequate testing time.

- To mitigate this, we need to establish a well-defined testing scope and allocate sufficient testing time.

- **Application Risk:**

  The application might exhibit abnormal behavior, and key features might not function as intended when accessed from unsupported devices or interfaces. Additionally, the application might not behave as expected on the latest browsers if it hasn't been adapted .

Contingency Plans

Define specific actions to be taken if risks materialize.

Data Security Breach Contingency:

- Isolate affected systems and servers.
- Notify relevant stakeholders and users about the breach.
- Engage cybersecurity experts to investigate the breach and implement remediation measures.

Performance Degradation Contingency:

- Scale up server resources to handle the increased load.
- Notify users of potential performance issues and expected resolution time.
- Communicate with the development team to identify and address bottlenecks.

Lack of Usability Contingency:

- Prioritize and address critical usability issues immediately.
- Conduct emergency usability testing to identify pain points and possible quick fixes.
- Communicate with UX designers to implement changes that enhance usability.

Test Exit Criteria

Testing process of the **Bikroy** application will be ended if following criteria are met:

- Define the conditions that must be met for testing to be considered complete.
- Include criteria related to test case coverage, defect closure, and stability.
- All specified functions are functioning properly.
- Major bugs are identified, resolved and retested.
- All test cases are executed and passed.

Status Explanation

New: The status of any defect or bug is logged in as To-do when it is discovered and verified.When a developer who has been assigned to the ticket is currently at work.

Done: This status denotes that the problem has been resolved, and further testing is permitted.

Verified: The bug is retested by the tester after the developer fixes it. If no software flaw is found, the flaw has been addressed, and the status has been changed to "verified."

Reopened: The tester updates the state to "reopened" if the bug still exists after the developer has fixed it. The bug goes through its life cycle once more.

Closed: When a bug is declared to be fixed, the tester marks it as "Closed."

Duplicate: The status is changed to "duplicate" if the defect occurs twice or relates to the same notion as the bug.

Rejected: When a developer determines that a flaw is not actually present, the defect is changed to "rejected."

Deferred: A bug is given the classification "Deferred" if it is not a top priority at the moment and is anticipated to be fixed in the upcoming version.

Not an error: The status of a problem is "Not a bug" if it has no impact on the application's operation.

Test Deliverables

Followings are the deliverables from QA for **Bikroy** project:

Test Deliverables before Testing

- Test Plan
- Test Cases
- Test Scripts

Test Deliverables after Testing

- Test execution report

Test Plan Approvals

Name	Role	Signature	Date
XYZ	Project Manager		2023-07-11
XYZ	Project Manager		2023-08-07
XYZ	Project Manager		2023-09-20
XYZ	Project Manager		2023-09-30

References

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

- System Requirement Specification: https://bikroy.com