**SOFTWARE TEST PLAN**

****

**eBay website**

**Team Members**

Najeeb Jabareen

**TABLE OF CONTENT**

1. **INTRODUCTION**
   1. Scope **3**
   2. Glossary **3**
2. **TEST TREE**
   1. Functional tests 4
   2. Features to be tested 6
   3. Features will not be tested 8
3. **TEST TYPE** 8
4. **RISK AND ISSUES** 9
5. **TEST LOGISTICS**
   1. Who will test10
   2. When will the test occur10
6. **TEST OBJECTIVE** 10
7. **TEST CRITERIA**
   1. Suspension Criteria 10
   2. Exit Criteria10
8. **RESOURCE PLANNING**
   1. System resource11
   2. Human Resource12
9. **TEST ENVIRONMENT** 13
10. **SCHEDULE & ESTIMATION**
    1. All project task and estimation15
    2. Schedule to complete these tasks15
11. **TEST DELIVERABLES**
    1. Before testing phase15
    2. During the testing15
    3. After the testing cycles is over16
12. **INTRODUCTION**

**1.1.scope**

The project not covers the entire ebay.com website, but it's include the primary goals:

-User Account Creation, Management, and Authentication Processes.

-Product Search, Listing, and Bidding Functionalities.

-Buy Now and Shopping Cart Features.

-User Interface (UI) and User Experience (UX) Across Desktop and Mobile Platforms.

-Payment Processing.

-Integration with External Service Providers

-Auction and Fixed-Price Sales Mechanisms

-Feedback and Dispute Resolution Systems

**1.2. Glossary**

Auction:A sales mechanism on eBay where buyers place bids on items over a specified period, with the item being sold to the highest bidder at the end of the auction.

Bid: An offer made by a buyer to purchase an item on auction on eBay. The bid represents the price the buyer is willing to pay for the item.

Buy Now: A feature that allows buyers to purchase items immediately at a fixed price without going through the auction process.

eBay Managed Payments: The payment processing system used by eBay to handle transactions between buyers and sellers, supporting various payment methods.

Listing: An item posted for sale on eBay, including details such as description, price, photos, and shipping information.

1. **TEST TREE:**

Functional tests:

1. Login page:
2. Sign in with email

* Login with existing username and password.
* Login with correct username and wrong password.
* Login with not exist username.

1. Third party login

I. login with facebook account

1. Login with signing in facebook account
2. Login with not signing in facebook account

Ii. login with google.

Iii. login with apple

2. Search product

1. Valid product search

1.with account

* Search valid product with login account
* Search valid product without login account

2. By filtering

* Choosing a product and continue search by filter
* Choosing category and search products by sorting

1. Invalid product search
2. Search product by categories

* Checking if valid product is appear

1. continue search by filter

* Checking if invalid product is exist

3. Product detail page

1. Product information and images

* Checking the correctness of products
* Checking for compatible filters

1. **Buy product**

* Logging in to the account
* Not Logging in to the account

1. Add to watching

* Logging in to the account
* Not Logging in to the account

1. Add to Cart

4. Shopping Cart

1. Cart persistence

I. Not Logging in to account

* Check if Items remain in the Cart after exit from the website

Ii.Logging in to account

* Check if Items remain in the Cart after exit from the website

1. Updating Quantity

* The price is change after update the quantity
* the updating quantity is saved after Logging out and login again

1. Remove item from Cart

I.if there is 1 item

* The Cart is “empty “ and the subtotal is 0 and navigate to “start shopping page.”

Ii. if there is more than 1 item

* Removing some of the items and the subtotal is updating, and the page still Exists.(Cart persistence).

5. Checkout

1. Logging in to account

I. if the item support location

* Navigate to pay page

Ii. if the item doesn’t support location

* Navigate ” review shipping” to update address

1. Not Logging in to account

* Ask to continue as a guest or to log in
* fill the form to buy

6. Shop by category

* Choosing a category or subcategory
* Shop by category (in the sidebar at left)

7. Home page

* Choose product or category from “Explore Popular Categories”
* Choose category “Today's Deals – All With Free Shipping”

8. Watchlist

1. With login account

* Add item to watchlist

1. Without login account

* Ask to login

1. If there are more than 1 item

I. if the item from the same category :

* All the items display in the page

Ii. if the item from different categories

* Choose the category to see the items

9. Purchase history

1. Logging to account

* Show purchased items by category.
* Show order items.
* Filtering orders by year.

1. Without Logging to account

* Can not see the history.

10 . Category Barside

* After choosing category , choose “Most popular categories” and “more categories”

2.2 Features to be tested

All the features of the eBay website which were defined in software requirement specs :

|  |  |  |
| --- | --- | --- |
| Module name | Applicable Roles | Description |
| Login | User | * Confirm that the log in success when login with correct user name and correct password. * Confirm that the login not success when login with user name is correct and password incorrect. * Confirm that the login not success when login with an incorrect user name. * Login using a connected account. |
| payment | User | * Confirm that the payment is processed successfully. * The user receives a payment confirmation by email. * Verify that the booking is a success. |
| Search | User | * Show relevant items to the description in the search. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

2.3. Features will not be tested

These feature are not be tested because they are not included in the software requirement specs

-Software Interfaces

-Database logical

-Communications Interfaces

-Website Security

3. **TEST TYPE**

To ensure the strength and reliability of shopping and selling website, a comprehensive testing strategy has been devised, through various types of testing to address both functional and non-functional aspects.

Functional tests:

1. Integration Testing:

* Verify the interaction between different modules.
* Ensure seamless data flow and functionality.

1. API Testing:

* Validate APIs handling data requests and responses.
* Confirm accurate data exchange between the website and external services.

1. Regression Testing:

* Ensure that new updates or changes do not adversely affect existing functionalities.
* Conduct regression testing after each update to validate the overall system stability.

Non-functional tests:

1. Performance Testing:

* Assess responsiveness and load handling capabilities.
* Conduct stress testing to evaluate performance under high user loads.

1. Security Testing:

* Identify and address potential vulnerabilities.
* Test for secure data transmission and storage practices.

1. Compatibility Testing:

* Ensure compatibility across different browsers and devices.
* Validate functionality on various operating systems.

1. Usability Testing:

* Evaluate the overall user experience during the booking process.
* Check for intuitive navigation and user-friendly design.

1. Load Testing:

* Assess the website's ability to handle specific loads.

1. Volume Testing:

* Evaluate system performance with a large volume of data.

1. Stress Testing:

* Test system robustness under extreme conditions.

1. Recovery Testing:

* Verify the system's ability to recover from failures.

1. Localization Testing:

* Test the website for different languages and regions if applicable.
* Ensure that the content and features of the website adjust well to different languages and cultures.

4. **RISK AND ISSUES**

|  |  |
| --- | --- |
| **Risk** | **Mitigation** |
| **Delays in project timelines.** | Implement agile project management practices to allow for flexible planning and adjustments. Holding regular stand-ups to address and mitigate delays promptly. |
| **Poor communication among team members and stakeholders.** | Establish clear communication channels and regular update meetings. Using project management tools to keep everyone informed and engaged. |
| **Changes in project requirements.** | Implement a change management process to evaluate the impact of changes on budget, timeline, and quality. Ensuring stakeholder alignment on changes before implementation. |
| **Inadequate testing coverage leading to quality issues.** | Using risk-based testing to prioritize testing areas. Implement automated testing for repetitive tasks to increase coverage. Regularly reviewing and adjusting test plans as the project evolves. |
| **The project schedule is too tight; it's hard to complete the whole project on time.** | Setting Test Priority for each of the test activities. |

5. **TEST LOGISTICS**

5.1. Who will test ?

The project will be tested by Beyond Dev QA Automation Student.

5.2. When will the test occur?

Having the tests is a crucial part of the planning process.

The timing of tests can be broadly categorized into several phases of the software development lifecycle and based on specific criteria.

-During requirements analysis.

-After Design Phase.

-During Development.

-Before and After Deployment.

6 **TEST OBJECTIVE**

The primary objective of our QA automation testing for the eBay website is to validate the functionality, performance, usability, and security of key features and processes. Through systematic, automated testing, it's aim to :

Ensuring Functional Accuracy Verifying that all features of eBay, including user registration, listing creation, search functionality, bidding processes, and payment transactions, perform as expected. This involves checking the correctness of operations and data integrity across different user scenarios.By achieving these objectives, the QA automation project aims to contribute significantly to the overall quality assurance of the eBay platform, enhancing user satisfaction, operational efficiency, and the platform's reputation in the marketplace.

7 **TEST CRITERIA**

7.1 Suspension Criteria

If the team members report that there are 40% of test cases failed, suspend testing until the development team fixes all the failed cases.

The tests are tests that constitute a better and higher quality user experience.

7.2 Exit Criteria

Specifies the criteria that denote a successful completion of a test phase

-Run rate is mandatory to be 100% unless a clear reason is given.

-Pass rate is 80%, achieving the pass rate is mandatory.

8 **RESOURCE PLANNING**

8.1 System Resource

|  |  |  |
| --- | --- | --- |
| **No.** | **Resources** | **Descriptions** |
| 1 | **Test Automation Server** | A dedicated server for running automated tests, including scheduling and executing test suites. |
| 2 | **Version Control System** | Tools like Git for source code management, allowing for version tracking and collaboration. |
| 3 | **Continuous Integration (CI) Tools** | CI platform (e.g., Jenkins, CircleCI) for automating the integration of code changes. |
| 4 | **Test Management Tool** | Software to manage test cases, plans, runs, and reporting (e.g., TestRail, Zephyr). |
| 5 | **Issue Tracking System** | Tool for tracking bugs and issues (e.g., JIRA, Bugzilla). |
| 6 | **Automated Testing Frameworks** | Frameworks and libraries for writing and executing tests (e.g., Selenium, Appium). |
| 7 | **Browser and Device Lab** | A collection of physical or virtual devices and browsers for testing website compatibility. |
| 8 | **Performance Testing Tools** | Tools for load and stress testing (e.g., JMeter, LoadRunner). |
| 9 | **Security Testing Tools** | Software for identifying vulnerabilities in the web application (e.g., OWASP ZAP, Burp Suite). |
| 10 | **Database Servers** | Servers hosting databases for test environments, mirroring production setups. |
| 11 | **API Testing Tools** | Tools specifically designed for testing APIs (e.g., Postman, SoapUI). |
| 12 | **Monitoring and Logging Tools** | Tools for monitoring system performance and logging during test execution (e.g., ELK stack). |

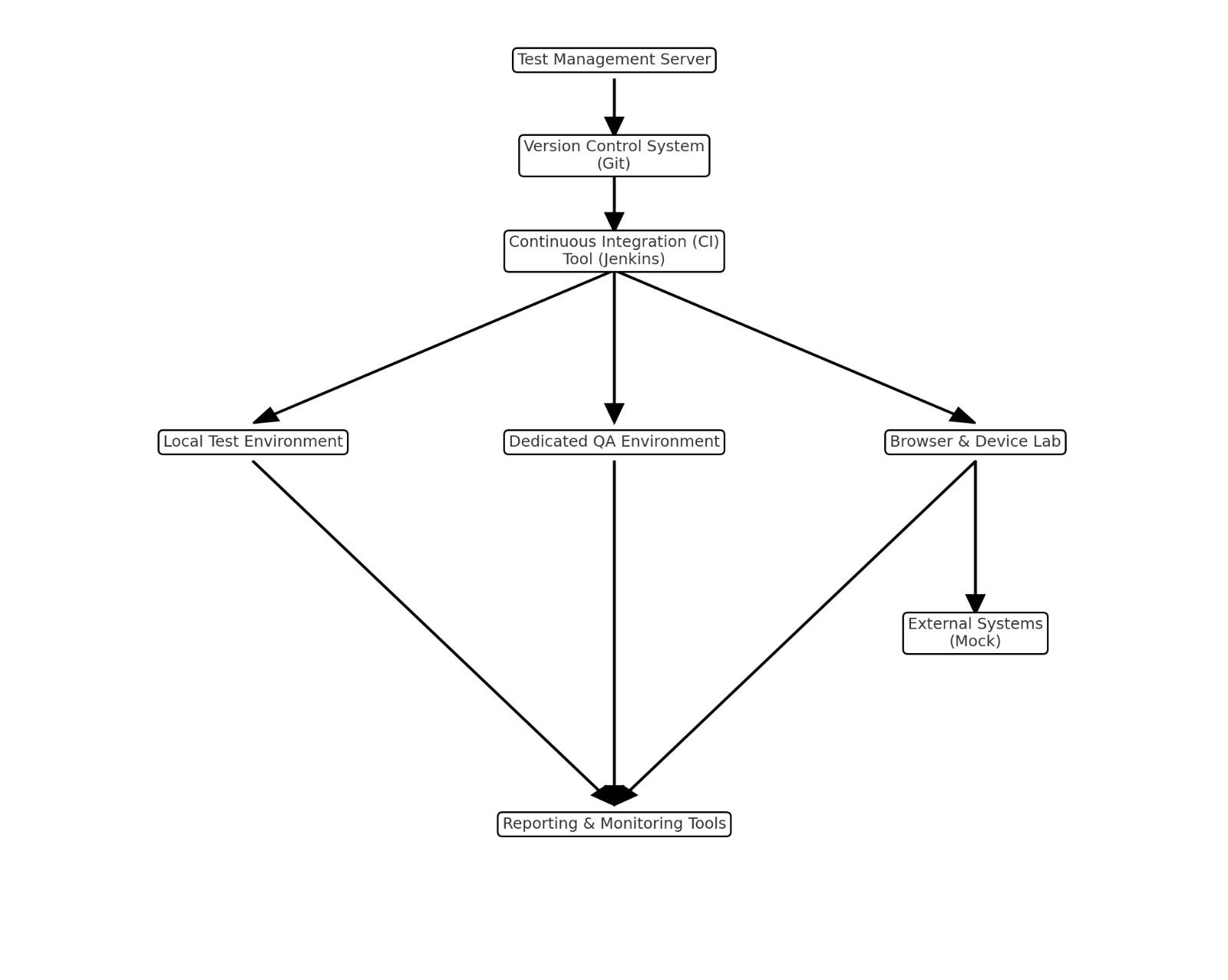
8.2 Human Resource

|  |  |  |
| --- | --- | --- |
| **No.** | **Members** | **Tasks** |
| 1 | Test Lead | Overseeing the test strategy, planning, and ensuring quality standards are met. |
| 2 | Automation Engineer(s) | Designing, developing, and maintaining test automation scripts. |
| 3 | Manual Tester(s) | Executing manual test cases that are not automated and assisting in exploratory testing. |
| 4 | Performance Tester | Specializing in performance testing, analyzing results, and identifying bottlenecks. |
| 5 | Security Tester | Conducting security assessments and vulnerability testing to ensure application security. |
| 6 | DevOps Engineer | Managing CI/CD pipelines, environments, and supporting automation infrastructure. |
| 7 | QA Analyst | Analyzing test results, reporting bugs, and verifying fixes. |
| 8 | Project Manager | Coordinating between teams, managing timelines, and ensuring project goals are met. |
| 9 | Business Analyst | Ensuring the test cases align with business requirements and user stories. |
| 10 | Stakeholder(s) | Providing feedback, prioritizing features, and ensuring the project aligns with business goals. |

9 **TEST ENVIRONMENT**

The Test Environment should be setup as figure below.

test environment for ebay.com that include user, database and web server.



This visual represents the main components and their interactions, including the Test Management Server, Version Control System (Git), Continuous Integration (CI) Tool (Jenkins), various test environments (Local, Dedicated QA, and Browser & Device Lab), External Systems (Mocks), and Reporting & Monitoring Tools. Arrows indicate the flow of information and control between these components.

10 **SCHEDULE & ESTIMATION**

10.1 All project task and estimation

|  |  |  |
| --- | --- | --- |
| **Task** | **Members** | **Estimate effort** |
| **Create the test specification** | Test Administrator | 10 man-hour |
| **Perform Test Execution** | Tester, Test Administrator | 7 man-hour |
| **Test Report** | Tester | 10 man-hour |
| **Test Delivery** | Tester | 8 man-hour |
| **Testing Code** | Tester | 30 man-hour |
| **Total** |  | **65 man-hour** |

10.2 Schedule to complete these tasks

11 **TEST DELIVERABLES**

refer to the tangible items or documents produced as a result of the testing process.

Test deliverables are provided as below

1. Test cases: scenarios that we are going to test is a case of a person who is using the product and is going to make an order when he registers for the Guinness program and he will make an order and receive an order confirmation by email

2. Test scripts: manual scripts and automation with unit test cases and run the tests to see the performance against the expected scenarios.

3. Test data: We need user login details and internet connection and the website address.

11.1 Before testing phase

Test plans document.

Test cases documents.

11.2 During the testing

during the test

test tool

simulators.

test data

Email is open to receive notifications

User details for connection and testing

11.3 After the testing cycles is over

Test Results/reports

Defect Report

Installation/ Test procedures guidelines