## Test Plan for Bamboo Warriors Repository



1. BACKGROUND
2. INTRODUCTION
3. ASSUMPTIONS
4. TEST ITEMS
5. FEATURES TO BE TESTED
6. FEATURES NOT TO BE TESTED
7. APPROACH
8. ITEM PASS/FAIL CRITERIA Blanket statement
9. SUSPENSION/RESUMPTION CRITERIA
10. TEST DELIVERABLES
11. TESTING TASKS Functional tasks (e.g., equipment set up)
12. ENVIRONMENTAL NEEDS
13. RESPONSIBILITIES
14. STAFFING & TRAINING
15. SCHEDULE
16. RESOURCES
17. RISKS & CONTINGENCIES
18. APPROVALS

**BACKGROUND**

The "Bamboo Warriors Repository of Knowledge" project is an educational platform dedicated to promoting awareness and collaboration around bamboo in the Philippines. This website aims to serve as a centralized hub of knowledge, bringing together information on bamboo’s cultural, ecological, and economic significance. By providing an organized repository of articles, multimedia resources, and an interactive forum, the Bamboo Warriors project seeks to inform a wide range of users, including students, researchers, farmers, policymakers, and environmental advocates. The platform will not only offer access to reliable and diverse content about bamboo but also foster community engagement and collaboration, encouraging knowledge sharing and innovation in sustainable bamboo-based industries. Ultimately, the Bamboo Warriors Repository strives to highlight bamboo as a versatile and valuable resource, advocating for its potential to support sustainable development within the Philippines.

**INTRODUCTION**

This Test Plan describes the testing strategy, objectives, resources, schedule, and responsibilities for verifying the functionality, performance, and security of the Bamboo Warriors Repository of Knowledge. The primary goal is to confirm the system's reliability, security, and usability align with the expectations outlined in the SRS.

**ASSUMPTIONS**

This Test Plan assumes that the "Bamboo Warriors Repository of Knowledge" website will be deployed and accessible within the GitHub Codespaces environment, which will serve as the primary platform for all testing activities. A comprehensive mock dataset, including sample articles, multimedia content, and user accounts, will be available to facilitate thorough testing of content management and user interaction functionalities. Testers will be provided with both member and admin access credentials, allowing them to assess role-based features, permissions, and interactions in accordance with the requirements specified in the SRS.

**TEST ITEMS**

|  |  |  |
| --- | --- | --- |
| **Test Item** | **Description** | **Test Type** |
| Check user registration with an existing username | Attempt to register using an existing username | Functional |
| Check user login with invalid credentials | Attempt to login using a wrong username and password | Functional |
| As a guest user check “Members Only” button in side bar | Check if “Members Only” button ask the user to sign in | UI/UX |
| As a guest user check "Forum" button in side bar | Check if "Forum" button ask the user to sign in | UI/UX |
| Check “Add Article” form has appropriate fields | Checking form functionality and if the corresponding fields have appropriate entry types (date, dropdown, text, etc.) | Functional |
| Check “Add Book” form has appropriate fields | Checking form functionality and if the corresponding fields have appropriate entry types (date, dropdown, text, etc.) | Functional |
| Check “Add Research Paper” form has appropriate fields | Checking form functionality and if the corresponding fields have appropriate entry types (date, dropdown, text, etc.) | Functional |
| Check “Add Video” form has appropriate fields | Checking form functionality and if the corresponding fields have appropriate entry types (date, dropdown, text, etc.) | Functional |
| Validate role-based content access | Ensure only authorized roles (member, admin) can access exclusive content | Functional |
| Validate multimedia upload formats | Attempt to upload multimedia files (JPEG, PNG, MP4) to ensure only supported formats are accepted | Functional |
| Check content creation for admins only | Ensure that only users with admin rights can access the content creation interface | Functional |
| Check the “Create new post” button on Forum | Confirm the "Create new post" button on the forum redirects to the post creation form | UI/UX |

**FEATURES TO BE TESTED**

**Content Management:**

* Create, edit, delete, and categorize articles, books, videos, and forum posts.
* Admin functionality for managing content visibility (public or member-only).

**Search and Filtering:**

* Validate the keyword search for accuracy and relevance, with sorting options (e.g., by date or popularity).
* Check filter options by content type (articles, videos, forum posts) and category (e.g., bamboo species, conservation techniques).

**User Authentication and Role-Based Access Control:**

* Verify account creation, login, and role-based permissions for guests, members, and admins.

**Multimedia Handling:**

* Test the upload, display, and management of images, videos, and audio files, ensuring compatibility with supported formats (e.g., JPEG, PNG, MP4).

**FEATURES NOT TO BE TESTED**

Listed below are features that should not be part of the scope of our project. Its functionalities should remain to be outside the scope to maintain its originality:

**Third Party Software** – This feature is not included in the project itself and will not be tested in any way.

**APP Creation** – The project is a website platform and there will be no other apps that existed. The project has only existed on mobile and Pc devices.

**APPROACH**

**System Testing -** The entire Bamboo Warriors Repository application will be tested as a whole to simulate interactions between various users (guests, members, admins) and the system. This includes testing user login, browsing and interacting with exclusive content, managing multimedia resources, and verifying user role-specific permissions. System testing will ensure that all components work together seamlessly and that the system meets specified functional and non-functional requirements.

**Acceptance Testing -** Acceptance criteria are set in alignment with requirements from both the client and project team. Each feature—such as user authentication, role-based content access, and content creation—must meet these pre-defined requirements to pass acceptance testing. The goal is to validate that the application performs as expected and meets end-user needs as per the Software Requirements Specification (SRS)

**ITEM PASS/FAIL CRITERIA**

In this section is the item on each test case of the website and its expected output

**Functions of the Website**

|  |  |  |
| --- | --- | --- |
| **Test Item** | **Pass** | **Fail** |
| Create a User Account | User can create a membership account | Upon creating the account there are multiple users with the same account |
| Continue as Guest | Allows user to browse without needing to login | Guest enters a restricted access |
| Dashboard | Displays all the articles and latest events | Articles and events failed to load up |
| Accessibility of Catalogue | Fully functional; users can view and access the contents | Some items don’t display the contents and causes errors |
| Lock Members Only (Restricted Access) | Successfully restricts access to non-members and requires members to log-in | Nonmembers can bypass the login |
| Night/Day mode for Screen | Toggles for smooth transition | Toggle doesn’t work |
| Ai Chat (Placeholder) | Placeholder displays and its function is coming soon | Placeholder doesn’t load or even not inputed. |
| Last Recommendation Articles | Displays the Recommended Articles | Doesn't show any recommended articlkes |

**Members (Logged in)**

|  |  |  |
| --- | --- | --- |
| **Test Item** | **Pass** | **Fail** |
| Create a User Account | User can create a membership account | Upon creating the account there are multiple users with the same account |
| Access to Exclusive Content | Allows user to browse without needing to login | Guest enters a restricted access |

Admin (Logged-In)

|  |  |  |
| --- | --- | --- |
| **Test Item** | **Pass** | **Fail** |
| Admin Tools | Tools are in the Navigation Panel | Tools are inaccessible by the admin or even causes an error |
| Book List | Displays all books and can add, delete and update | Books are missing and the tools to edit doesn’t work |
| Add Book | Allows the Admin to add books including the title, author, description | Upload fails |
| Add Research Papers | Successfully upload the research files | Upload Fails |
| Add Video | Video can be added | Video failed to add |
| Video List | Video are displayed properly | Video isn’t loading |
| List of Articles | List of uploaded articles that can be edited, added and updated | Some articles don’t properly load |
| Add Article | Allows user to add | Editor tool isn’t working |

**SUSPENSION/RESUMPTION CRITERIA**

**Suspension Criteria**

Testing may be suspended under the following conditions:

* System Outage: Any unexpected downtime or instability in the GitHub Codespaces environment.
* Critical Defects: Discovery of critical bugs that prevent further testing of key functionalities, such as user authentication, role-based access, or content management.
* Dependency Failures: Issues with external integrations or APIs, such as social media or SEO tools, that are essential for testing specific requirements.

**Resumption Criteria**

Testing will resume when:

* System Stability Restored: The GitHub Codespaces environment or other dependencies are restored.
* Defect Resolution: Critical issues have been resolved, and fixes have been verified in the test environment.
* Data or API Access Restored: Necessary data and external integrations become available again.

**TEST DELIVERABLES**

This section shows the list of the deliverables which are required documentations aside from the software:

**Software Requirements Specifications (SRS)** – A detailed document outlining the project’s functional and non-functional requirements, serving as the primary reference for test case design and validation.

**Test Plan Documentation** – A comprehensive guide describing the testing strategy, objectives, scope, schedule, and responsibilities for the testing process.

**Test Case Document** – A collection of individual test cases specifying input, execution conditions, and expected outcomes to ensure each feature and requirement is validated.

**Requirement Traceability Matrix (RTM)** – A matrix linking each requirement in the SRS to corresponding test cases, ensuring full coverage and traceability throughout the testing lifecycle.

**TESTING TASKS**

1. **Functional Tasks**

1.1 Login/Creation/Guest - verify login functionality with the correct and incorrect credentials.

User Login:

* Description: Test the login with the valid credentials if it's a member, admin and guest.
* Expected Outcome: The expected outcome is that the credentials should recognize who is the person authenticated to view the website.

1.2 Dashboard – show that the information loads and displays the right information.

View Dashboard:

* Description: Test if the articles, videos and other researchers are updated and loading properly
* Expected Outcome: It should load and view the latest articles.

1.3 Forum Posting – it is a discussion thread to all the community

View Forum:

* Description: Test posting, editing, and deleting posts for logged-in users.
* Expected Outcome: Logged-in users should be able to post, edit, and delete posts without errors. Guests should not have posting permissions.

1.4 Catalogue – it shows the items in the side bar of the navigation panel, that shows books, articles, research papers and videos.

Cataloguge Browsing:

* Description: Test the ability to view and filter catalogue items (Books, Videos, Research Papers, Articles).
* Expected Outcome: All items should display correctly, and filter based on selected criteria (type, date, popularity).

1.5 Lock Members Only – this from the catalogue where members and admin only can access certain articles, research papers and videos.

Restircted Access:

* Description: Test if the restricted access is being accessed by guests.
* Expected Outcome: when clicked by the guest it should redirect them to the login page where members and admin can access.

1.6 Night/Day Mode – it can adjust the user’s preference on its display of screen on the website.

Mode Toggle:

* **Description**: Test toggling between Night and Day modes, ensuring readability.
* **Expected Outcome**: Mode should toggle smoothly, saving user preference, and color schemes should be consistent across pages.

1.7 AI Chat – it is a feature that is a placeholder and soon to come

AI Chat Placeholder:

* Description: Ensure that there is placeholder however it should be a coming soon
* Expected Outcome: It shouldn’t work since it is a coming soon feature.

1. **Membership Feature**

2.1 Exclusive Content – when members login on the website they have access to exclusive articles, research papers, videos and updated content.

Access on Contents:

* Description: Test if the members can see available content
* Expected Outcome: When members have access, they see contents that are available on the website that guest seeing these available contents.

1. **Admin Feature**

3.1 Admin Catalogue:

* **Description**: Verify that admin users have access to all tools (book list, add book, etc.) from the dashboard.
* **Expected Outcome**: Admin tools should be accessible without errors, with permission restricted to admins.

3.2 Book, Articles, Video and Research Paper Management – admin has the authority to add, update and delete books within the website itself.

* Description: Test if the admin has the authority to add, update and delete necessities within the website.
* Expected Outcome: Admin should see his/her added files, updated descriptions and deleted documents.

**ENVIRONMENTAL NEEDS**

The testing environment requires specific configurations and tools to accurately simulate user interactions and validate system functionality:

**Testing Platform -** GitHub Codespaces will be used to host the website and perform all testing activities, providing a controlled and cloud-based environment.

**Hardware Requirements -** Desktop or laptop computers for testers, equipped with reliable internet access and standard web browsers, such as Chrome, Firefox, Safari, and Edge, to ensure compatibility testing.

**Software Tools -** Access to automated testing tools and documentation tools to record test case results, defect logs, and other testing documents.

Security Protocols: Secure connections for testing security features like SSL/TLS encryption and authentication mechanisms, ensuring sensitive tests are conducted in a safe environment.

**RESPONSIBILITIES**

|  |  |  |
| --- | --- | --- |
| **Role** | **Name** | **Description** |
| SOFTDEV Adviser | Manuel Sebastian S. Sanchez | Provides guidance on software development practices and ensures the project aligns with academic and industry standards. The adviser supports the team in meeting project requirements and technical objectives. |
| Client | Bamboo Warriors Philippines | Defining project goals and requirements with their interests and requirements. They also provide feedback and approves key deliverables to ensure the project meets expectations. |
| Consultant | Jose Eugenio L. Quesada | Offers expertise and strategic advice on project direction, helping the team address challenges and make informed decisions. The adviser may also assist in refining project goals and scope. |
| Product Owner | Carl Von Nicanor | Acts as the main point of contact between the client and development team, ensuring that the team understands and prioritizes client needs. The Product Owner defines the project’s vision, manages the product backlog, and approves completed features. |
| Scrum Master | Johanne Marlowe Sanchez | Facilitates Scrum meetings, removes obstacles, and ensures the development team follows agile processes. The Scrum Master supports team collaboration, helping to maintain an efficient and productive workflow. |
| Developers | Carl Von Nicanor  Henzon Ivan Valguna | Responsible for coding, implementing features, and addressing bugs. The developers work closely with the QA team to resolve defects and ensure that all functionalities align with project specifications. |
| Documentation Team | Gabrielle Ann Yogawin  Henzon Ivan Valguna | Responsible for creating, organizing, and maintaining all project documentation, including requirements, test plans, and other deliverables. The Documentation Team ensures that all documents are clear, up-to-date, and accessible, facilitating knowledge transfer and supporting project continuity. |
| QA Testers | Aldrin Dave Alia | Execute test cases to identify defects and ensure the software functions as intended. They document test results, report issues, and verify fixes, contributing to the overall quality of the project. |

**STAFF AND TRAINING**

**Staffing:**

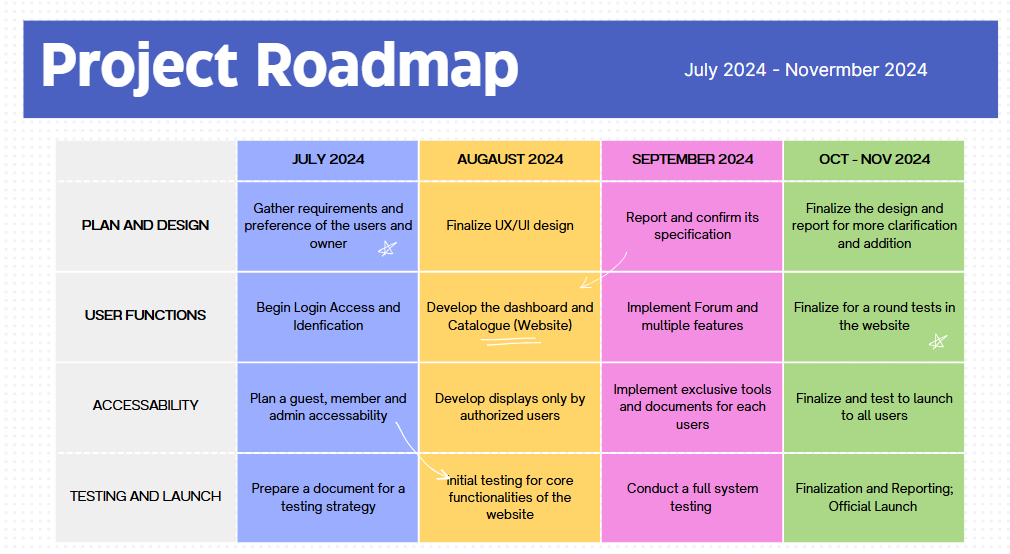
The team composition for the **Bamboo Warriors Philippines** project consists of Trainers, a Product Owner, Developers, QA Testers, and a Documentation Team. These roles are tailored to meet the project’s requirements for creating a comprehensive knowledge repository. With each team member doing their respective roles, the group collaborates to handle their responsibilities effectively and ensure the timely delivery of project outcomes.

**Training:**

The project adviser and consultant are the ones who provide guidance on development practices, agile methodologies, and project-specific requirements. They conduct meeting sessions to equip team members with the necessary skills for their roles, supporting both technical and strategic project alignment.

**SCHEDULE**

The schedule below shows the roadmap of the whole project development bringing key deliverables:

Figure 1: Project Roadmap

**RESOURCES**

The testing phase for the "Bamboo Warriors Repository of Knowledge" website requires a range of resources to support accurate and efficient testing. These resources include human, hardware, and software, all essential to creating a reliable testing environment that mirrors the expected production setup.

**1. Hardware Resources**

* Desktop/Laptop Computers: Each QA tester will require a dedicated workstation (desktop or laptop) with up-to-date specifications to ensure smooth operation of testing tools and the GitHub Codespaces environment.
* Internet Connectivity: Reliable, high-speed internet access is essential for maintaining consistent connections to the cloud-based GitHub Codespaces and for accessing external APIs used in the project.

**2. Software Resources**

* GitHub Codespaces: This is the primary development and testing environment, where the Bamboo Warriors website will be deployed for testing. Codespaces provides a collaborative, version-controlled workspace for managing and running test cases in real time.
* Testing and Documentation Tools: Tools such as MS Office and MS SharePoint are needed to manage and track test cases, defects, and documentation. These tools help organize the testing process and ensure that all cases are recorded and reviewed.
* Communication and Collaboration Tools: Microsoft Teams are needed for real-time communication among testers, developers, and QA leads, ensuring seamless collaboration and prompt issue resolution.

**3. Data and Storage Resources**

* Test Data Repository: A centralized storage location within GitHub Codespaces or an alternative secure cloud location will house test data, ensuring that all necessary sample data (e.g., articles, multimedia files, and user profiles) is accessible and backed up.
* Backups and Versioning: Version control within GitHub will track changes and allow rollback if data loss or corruption occurs. This ensures the consistency and integrity of test data throughout the testing cycle.

**4. Human Resources**

* QA Analysts: QA analysts will review test cases, track issues, and validate that resolved issues meet project standards. They will work closely with Test Engineers to oversee test documentation and reporting.
* Developers: Developers will provide essential support by addressing reported defects and implementing fixes. Their role includes ensuring that the testing environment accurately reflects any changes made to the codebase.

**5. External Services and API Access**

Access to Third-Party APIs: Testing will require access to any external APIs used in the project (e.g., for social media or SEO integrations). Temporary credentials for these services may be necessary for testing data retrieval, content sharing, and integration functionality.

**RISKS AND CONTINGENCIES**

This section will discuss the Risks and Contingencies that help identify potential challenges that could impact the successful development of the Bamboo Warriors Philippines project.

**Risks**

* Technical Downtime - Downtime or instability in key tools or environments, such as the development or testing platforms, may disrupt the testing process.
* Delayed Issue Resolution - If critical issues or defects are not resolved promptly, this may delay testing schedules and affect project timelines.
* External Dependency Unavailability - Unavailability of external services, APIs, or other dependencies could impact the ability to test certain integrations or functionalities.

**Contingencies**

* Backup Testing Environment - Establish a backup environment to continue testing in case of technical downtime. Schedule critical testing activities to avoid peak times for potential disruptions.
* Prioritization of Defects - Implement system to prioritize issues based on their impact on functionality, ensuring that critical defects are addressed first to minimize testing delays.
* Mock or Simulated Dependencies - Use mock data or simulated responses to proceed with tests if external dependencies are unavailable. Retest once dependencies are restored.

**APPROVALS**

Approved by Project Sponsors:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Manuel Sebastian S. Sanchez Jett Banoan Reyes**

Software Development Adviser Bamboo Warriors Philippines Client