**A PROPOSED OFFERING OF A WEB-BASED TRAVEL BOOKING SYSTEM FOR RARE EXPLORER TRAVEL&TOURS AT DATAMEX COLLEGE OF SAINT ADELINE VALENZUELA BRANCH**

A Project Proposal Presented to the

Faculty of Datamex College of Saint Adeline, Inc.

In Partial Fulfillment of the Requirements for the

Degree of Bachelor of Science in Information Technology

By:

Alelojo,Red Hernie

Wong,Chrisdelyn S.

October 2025

**TESTING**

**DOCUMENTATION**

**INTRODUCTION**

**Overview**

The testing phase of the RareExplorer Travel and Tours Booking System aims to verify the accuracy, reliability, and performance of the system before its final deployment. This stage ensures that all system features function as intended and meet the requirements defined during the design and development stages. Testing is a crucial part of the software development life cycle (SDLC) because it identifies potential defects, verifies functionality, and ensures that the system provides a seamless and secure user experience.

**Purpose of the Testing Phase**

The primary purpose of the testing phase is to validate that the developed system performs according to its specified requirements. It ensures that all modules—such as user registration, booking management, and GCash payment integration—operate correctly and interact efficiently with the Firebase database. Testing also confirms that both functional and non-functional aspects of the system (e.g., usability, compatibility, and security) meet the expected standards.

**Objectives of the Testing Process**

* The objectives of the testing process are as follows:
* To verify that all modules and system components function as designed.
* To identify and correct defects or errors before system deployment.
* To ensure that user inputs, data storage, and payment transactions are processed accurately.
* To confirm that the system performs efficiently under different conditions and devices.
* To evaluate the overall system usability and responsiveness for both end-users and administrators.

**Scope of Testing**

The scope of testing covers all the core functionalities of the RareExplorer Travel and Tours Booking System, including user registration, authentication, travel package browsing, booking process, and payment transactions. The testing also evaluates data synchronization between the front-end and Firebase database, ensuring that all transactions and user records are stored correctly.

**TESTING ENVIRONMENT**

**Overview**

The testing environment refers to the combination of hardware, software, and data configurations used to execute various testing activities. Establishing a controlled testing environment ensures that all tests are consistent, reproducible, and reliable.

**Hardware Specifications**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Device Type | Processor | Memory (RAM) | Operating System | Remarks |
| Laptop (Primary) | Intel Core i5, 11th Gen | 8 GB | Windows 11 | Main development and testing device |
| Laptop (Secondary) | AMD Ryzen 5 | 16 GB | Windows 10 | Used for compatibility testing |
| Smartphone | Snapdragon 778G | 8 GB | Android 13 | Mobile responsiveness and UI testing |
| Desktop Computer | Intel Core i7, 10th Gen | 16 GB | Windows 11 | Performance testing |

*Table 1. List of Hardware*

**Software Requirements**

|  |  |  |
| --- | --- | --- |
| Software / Tool | Version | Purpose |
| Google Chrome | v.121.0 and above | Browser testing and web rendering |
| Mozilla Firefox | v.118.0 and above | Browser compatibility testing |
| Visual Studio Code | v.1.92 | Development and debugging environment |
| Firebase | Cloud Version (2025) | Database, authentication, and hosting |
| Node.js | v.20.11.0 | Backend environment and dependency management |
| GCash Sandbox API | 2025 Developer Version | Payment gateway simulation |
| GitHub | Latest | Version control and issue tracking |

*Table 2. List of Software*

**Test Data**

|  |  |  |
| --- | --- | --- |
| Data Type | Sample Entry | Purpose |
| User Accounts | user01@gmail.com / password123 | Test login and authentication |
| Destination | Boracay Island – PHP 5,000 | Validate destination display and price computation |
| Booking Record | 2 travelers, date: Dec 20, 2025 | Confirm total computation and record creation |
| Payment Data | Simulated GCash transaction (₱5,000) | Test payment API connection and status update |

*Table 3. List of Test Data*

**TESTING METHODOLOGY**

The testing methodology defines the overall approach used to ensure that the RareExplorer Travel and Tours Booking System meets both functional and non-functional requirements. Different testing strategies were employed to evaluate the system’s performance, reliability, and usability.

**Testing Approaches**

* Black-Box Testing – Focused on verifying system outputs without examining internal code logic.
* White-Box Testing – Used during development to test internal code flow and logic.
* User Acceptance Testing (UAT) – Conducted with selected users to evaluate usability and satisfaction.

**Testing Tools and Frameworks**

* Firebase Emulator Suite – Used to simulate database and authentication operations.
* GCash Sandbox API – Used to process simulated payments securely.
* Browser Developer Tools – Used for inspecting UI behavior, debugging, and responsiveness.
* GitHub Issues Tracker – Used to log and monitor bugs and fixes throughout testing.

**Test Cases and Criteria**

* The function must execute without errors or interruptions.
* The output must match the expected system behavior.
* The interface must remain responsive and consistent across devices.
* The system must correctly handle invalid or missing user inputs.

**Test Cases**

**Overview**

Test cases were developed to verify that each module of the RareExplorer Travel and Tours Booking System performs as intended. Each test case defines the scenario, steps, expected results, actual results, and corresponding status.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Test Description | Test Steps | Expected Output | Actual Output | Status | Remarks |
| TC001 | Login with valid credentials | 1. Enter username 2. Enter password 3. Click login | User is redirected to dashboard | User is redirected to dashboard | Pass | N/A |
| TC002 | Login with invalid password | 1. Enter username 2. Enter incorrect password 3. Click login | Error message appears | No error message appeared | Fail | Bug identified |

*Table 4. List of Test Cases*

**Bug Tracking & Issue Log**

During the testing phase, several issues were identified and logged for resolution. Each bug was categorized based on its severity and tracked according to its current status.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Bug ID | Description | Severity | Reported By | Status | Resolution |
| B001 | Login page crashes on incorrect password | High | Tester Name | Open | Pending fix |

*Table 5. List of Bug Tracking*

**User Acceptance Testing (UAT)**

**Overview**

User Acceptance Testing (UAT) was conducted with selected end-users to assess the overall usability, functionality, and satisfaction level of the system. The goal was to ensure that the RareExplorer Travel and Tours Booking System meets user expectations and performs smoothly in real-world scenarios.

**Test Scenarios**

* Users were able to register, log in, and book travel packages successfully.
* Payment transactions through GCash were completed and reflected accurately in the system database.
* Users reported that the interface was intuitive, organized, and visually appealing.
* Minor feedback included suggestions to improve text readability and mobile layout spacing.

**Feedback Summary**

Overall feedback from participants was positive. Users appreciated the simplicity and responsiveness of the interface. Minor concerns, such as button alignment and color contrast on mobile devices, were documented and scheduled for correction.

**CONCLUSION AND RECOMMENDATION**

The testing phase demonstrated that the RareExplorer Travel and Tours Booking System is stable, functional, and ready for deployment. Most of the major system functions, such as authentication, booking, and payment, performed successfully during tests.

**Key Observations**

* The system achieved a high success rate in functionality testing.
* Minor UI inconsistencies were identified but did not affect core operations.
* Payment module integration with GCash API worked as expected in the sandbox environment.

**Recommendations**

* Enhance system responsiveness for lower-end mobile devices.
* Implement automated backup routines for the Firebase database.
* Conduct periodic security audits for data protection and API authentication.
* Integrate analytics to monitor system performance and user behavior.