Test Plan

|  |  |
| --- | --- |
| Project Name | Academic ERP System with Payment Gateway Integration |
| Version | 2.0 |
| Date | 12/01/2020 |
| Prepared by | Sameer Rege |
| Submitted to | Manager/Lead |

**1. Introduction**

**Objective**

The objective of this test plan is to outline the testing process for the Academic ERP system, which includes a payment gateway for online payment facilities. This plan will ensure that all functionalities are tested thoroughly and meet the quality standards required by stakeholders.

**Scope**

The scope of testing includes functional, non-functional, integration, system, and regression testing of the Academic ERP system, focusing on the payment gateway integration. The scope also includes testing the ERP-based dashboard for its performance and security aspects.

**Assumptions and Dependencies**

* Assumption: All necessary requirements and design documents are available.
* Dependency: Payment gateway API documentation and sandbox environment.

**2. Test Items**

**Features to be Tested**

* User registration and authentication
* Course enrollment and management
* Payment gateway integration
* Dashboard functionalities
* Notification system

**Features not to be Tested**

* Third-party payment gateway internals

**3. Test Strategy**

**Test Levels**

* Unit Testing
* Integration Testing
* System Testing
* Acceptance Testing

**Test Types**

* Functional Testing
* Non-functional Testing (Performance, Security)
* Usability Testing
* Regression Testing

**Entry and Exit Criteria**

* **Entry Criteria:**
  + Requirements are finalized and approved.
  + Test environment is set up.
  + Test data is prepared.
* **Exit Criteria:**
  + All planned test cases executed.
  + No high-severity defects open.
  + Test completion report reviewed and approved.

**4. Test Environment**

**Hardware**

* Servers: Application server, Database server
* Client Machines: PCs, Mobile devices

**Software**

* Operating Systems: Windows, Linux, iOS, Android
* Browsers: Chrome, Firefox, Safari, Edge
* Payment Gateway: [Name of Payment Gateway]

**Network Configuration**

* LAN/WAN setup
* Secure VPN for remote testing

**5. Test Schedule**

**Milestones**

* Test Plan Review: [Date]
* Test Case Development: [Date Range]
* Test Execution: [Date Range]
* Test Closure: [Date]

**Testing Tasks**

* Test planning
* Test case development
* Test environment setup
* Test execution
* Defect logging and tracking
* Test reporting

**Dependencies**

* Availability of development and testing resources
* Timely delivery of test environments

**6. Test Deliverables**

**Deliverables List**

* Test Plan
* Test Cases
* Test Scripts (for automated tests)
* Test Reports
* Defect Logs

**Status Reports**

* Daily status reports
* Weekly summary reports

**Test Logs**

* Execution logs
* Defect logs

**7. Roles and Responsibilities**

**QA Team**

* Test planning and execution
* Defect reporting
* Test case development

**Developers**

* Fixing defects
* Assisting in test environment setup

**Business Analysts**

* Requirement clarification
* Assisting in acceptance testing

**Stakeholders**

* Providing requirements
* Reviewing test results

**8. Test Cases and Scenarios**

**Detailed Test Cases**

* **Test Case 1:** User Registration
  + **Precondition:** User navigates to the registration page.
  + **Steps:** Enter user details, submit the form.
  + **Expected Result:** User is registered successfully.
* **Test Case 2:** Payment Processing
  + **Precondition:** User has items in the cart.
  + **Steps:** Select payment method, enter payment details, confirm payment.
  + **Expected Result:** Payment is processed successfully, confirmation message displayed.

**Scenarios**

* **Scenario 1:** A student enrols in a course and pays the fee online.
* **Scenario 2:** An admin views the payment reports on the dashboard.

**9. Defect Management**

**Defect Logging**

* Log defects in [Defect Tracking Tool]
* Include steps to reproduce, severity, and screenshots

**Defect Tracking**

* Regular review of defect status
* Prioritization of defect fixing

**10. Risk Management**

**Identified Risks**

* Delays in receiving API documentation
* Issues with payment gateway integration

**Mitigation Plans**

* Frequent communication with the payment gateway provider
* Backup plans for critical functionalities

**11. Tools and Automation**

**Tools Used**

* Test Management: JIRA, TestRail
* Automation: Selenium, JMeter
* Defect Tracking: JIRA
* Programming Language: Core-java

**Automation Strategy**

* Identify critical test cases for automation
* Develop and execute automated scripts
* Regularly update scripts for new features

**12. Quality Assurance Sign-off**

**Acceptance Criteria**

* All critical test cases passed
* No high-severity defects open
* Acceptance test results approved by stakeholders

**Sign-off Process**

* Review of test results by QA Lead
* Approval from Product Owner
* Final sign-off from Project Manager

Addition Information:

Attachment:

Common Credentials:

Mail Id’s: