TEST PLAN

CCST Project – Testing of ePariksha Platform

---

1. Introduction

This Test Plan document defines the scope, strategy, approach, resources, and schedule for testing the **ePariksha Platform**, a web-based MCQ examination system developed and hosted by **CDAC – ACTS (ASD Software Division)**.
The objective of testing is to ensure that the application functions correctly, reliably, and securely for all defined user roles.

---

2. Project Overview

- **Project Name:** CCST – Testing of ePariksha Platform
- **Organization:** CDAC – ACTS (ASD Software Division)
- **Application Type:** Web-based MCQ Examination System
- **Application URL:** http://ccst-testvm.pune.cdac.in:8080/epAdmin/index.jsp
- **Project Duration:** 3 Weeks

User Roles Under Test

- Admin (Super User)
- Examiner (Course Coordinator)
- Examinee (Candidate)

---

3. Testing Scope

3.1 In-Scope

- Manual functional testing for all user roles
- Automation testing using Selenium WebDriver with Java
- Smoke testing
- Regression testing
- Role-based access and authorization validation
- End-to-end examination workflow testing
- Cross-browser testing on Chrome, Firefox, and Edge
- Cloud-based cross-browser testing using BrowserStack
- Continuous Integration execution using Jenkins

3.2 Out-of-Scope

- Performance and load testing
- Security testing
- API testing

---

4. Test Strategy

4.1 Manual Testing Strategy

Manual testing will be conducted to validate:

- Functional correctness of Admin, Examiner, and Examinee workflows
- Positive, negative, and boundary value scenarios
- Excel-based bulk upload functionality
- End-to-end examination lifecycle
- System constraints and validation rules

4.2 Automation Testing Strategy

Automation testing will be implemented selectively to ensure stability and repeatability.

- **Automation Tool:** Selenium WebDriver
- **Programming Language:** Java
- **Test Framework:** TestNG
- **Design Pattern:** Page Object Model (POM)
- **IDE:** Eclipse IDE

Automation will cover:

- Smoke test suite
- Regression test suite
- Authentication workflows
- Core examiner functionalities
- Candidate examination happy-path scenarios

---

5. Test Types

- Smoke Testing
- Functional Testing
- Regression Testing
- UI Validation Testing
- Cross-Browser Testing

---

6. Test Environment & Execution

Execution Details

- Test execution will be performed from local test machines
- The application under test is hosted on CDAC servers and accessed via web browsers
- Cross-browser and cross-platform testing will be conducted using BrowserStack
- Continuous Integration execution will be configured using Jenkins

Supported Browsers

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

Operating Systems

- Windows (Local Execution)
- BrowserStack-supported cloud environments

---

7. Tools & Technologies

Category	Tool
Automation Tool	Selenium WebDriver
Programming Language	Java
Test Framework	TestNG
IDE	Eclipse IDE
CI Tool	Jenkins
Cloud Testing	BrowserStack
Test Case Management	Excel
Defect Tracking	Jira
Version Control	GitHub

---

8. Team Roles & Responsibilities

Name	Project Role	Responsibility
Manoj Ambhore	Admin (Super User)	User, course, examiner, and report management

Vishvjeet Singh Chauhan	Examinee (Candidate)	Candidate-side exam execution and validation
Vivek Tiwari	Automation Tester	Automation framework design, regression testing, and coordination
Tilak Shivaji Shelke	Automation Tester	Automation execution, CI (Jenkins), and BrowserStack testing
Geeta	Project Guide	Review, guidance, and approval

---

9. Entry & Exit Criteria

Entry Criteria

- Application accessible on CDAC server
- Test environment configured
- Test data prepared
- Test cases reviewed and approved

Exit Criteria

- All planned test cases executed
- Smoke and regression testing completed
- No open critical defects
- Test Summary Report approved

---

10. Defect Management

- **Defect Tracking Tool:** Jira
- **Defect Lifecycle:**
  New → Assigned → Fixed → Retest → Closed
- Defects will be classified based on severity and priority.

---

11. Test Data Management

- Dummy courses, modules, and exams will be created for testing
- Test PRNs will follow system constraints (minimum 9 digits)
- Separate test data will be maintained for each user role
- Test data will be reused where applicable without impacting live data

---

12. Automation Constraints

- Real-time exam timers are not fully automatable
- Exam submission requires manual verification
- Result visibility depends on examiner configuration
- Automation is limited to stable and low-risk application areas

---

13. Requirement Traceability

All test cases are traceable to the requirements defined in the **System Understanding Document** provided by CDAC, ensuring complete coverage.

---

14. Risks & Mitigation

Risk	Mitigation
Dynamic exam behavior	Manual validation
Limited automation feasibility	Selective automation strategy
CI execution issues	Backup local execution

---

15. Test Closure

Testing activities will be formally closed after:

- Completion of all planned test executions
- Resolution of critical defects
- Approval of the Test Summary Report by the Project Guide

---

16. Approval

Name	Role
Geeta	Project Guide