SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

Website Development & Software Testing Project Planning

Institution Details

College Name: Silver Oak College of Computer Application
Department: Computer Application
Program: Bachelor of Computer Application (BCA)
Semester: 3
Academic Year: 2025–26
Course: Software Testing
Course Code: 4040003281
Assignment Type: Innovative Assignment

1. Project Title

FOR REFFRENCE :“Smart E-Learning Web Portal with Peer-Based Testing Integration”

(Students may change the title based on their application idea, e.g., “Online Bus Booking System”, “Digital Portfolio Site”, etc.)

2. Objective of the Project

- To simulate an **industry-standard software development and testing process**.
- To give students a **hands-on understanding of the Software Testing Life Cycle (STLC)**.
- To build a web application that meets defined functional and non-functional requirements.
- To train students in **designing, executing, and reporting software tests**.
- To demonstrate proficiency in **Manual Testing** and optionally **Automated Testing**.
- To develop professional-grade documentation: **Test Cases, Bug Reports, Summary Reports**.
- To promote teamwork, cross-reviewing, and **peer collaboration in software validation**.
- To apply **black-box**, **white-box**, and **UI testing techniques** effectively.

3. Project Description

The proposed project is a **Web-Based E-Learning Portal** that provides students with access to study materials, online quizzes, video lectures, assignment submissions, and real-time announcements.

Once the development is completed by Group A, it will be handed over to **Group B for testing**. Similarly, Group A will test Group B’s project.

The application will go through a comprehensive testing process by the testing group which includes:

- Test Plan preparation
- Test case writing and execution
- UI/UX testing
- Functional and boundary value testing
- Bug identification and documentation
- Optionally automated testing via Selenium or Postman

4. Target Users

- College Students
- Faculty/Teachers
- Admin Panel Users

5. Key Modules/Features

- **User Authentication** (Login/Signup, Forgot Password)
- **Student Dashboard** (Quizzes, Materials, Progress Reports)
- **Faculty Panel** (Upload Notes/Assignments, Manage Quizzes)
- **Admin Panel** (User Management, Content Management)
- **Quiz Module** (MCQ-based tests with auto evaluation)
- **File Upload & Download System**
- **Email Notification Integration**
- **Responsive Web Design for Mobile/Tablet**

6. Technology Stack

Layer	Tools / Technologies
Frontend	HTML5, CSS3, JavaScript, Bootstrap 5, ReactJS (optional)
Backend	PHP (Laravel) / Python (Flask/Django) / Node.js
Database	MySQL / PostgreSQL / MongoDB
Hosting	GitHub Pages / Firebase Hosting / Localhost (XAMPP)
Testing Tools	Manual Testing, Selenium WebDriver (for UI), Postman (for API), JUnit (optional)
IDE & Others	VS Code, GitHub, Draw.io (for diagrams), Excel/Word for reports

7. Software Testing Techniques to be Applied

- **Black Box Testing:** Tests functionality without knowing internal code — *e.g., login with wrong credentials should show an error*.
- **White Box Testing:** Tests internal code logic and paths — *e.g., testing all branches in a grade calculation function*.
- **Unit Testing:** Tests individual functions or units — *e.g., validating quiz score calculation logic*.
- **Boundary Value Analysis (BVA):** Tests input at edge limits — *e.g., testing age field with 17, 18, 30, 31*.
- **Equivalence Partitioning:** Tests representative values from valid/invalid input groups — *e.g., valid grades A–F, invalid inputs like "X" or "123"*.
- **Functional Testing:** Verifies each feature works as per requirement — *e.g., file upload allows only PDFs*.
- **Integration Testing:** Checks data flow between connected modules — *e.g., quiz submission reflects on user dashboard*.
- **UI/UX Testing:** Ensures user interface is friendly and usable — *e.g., buttons are labeled and mobile-responsive*.
- **Regression Testing:** Re-tests old features after updates — *e.g., after fixing quiz bug, recheck assignment module*.
  - **Exploratory Testing:** Unscripted testing to discover hidden bugs — *e.g., randomly navigating app to find crashes*.

- **Bug Life Cycle:** Tracks the status of a bug from discovery to closure — *e.g., New → Assigned → Fixed → Retested → Closed.*
- Optional Automation Testing using Selenium/Postman

8. Documentation to be Submitted

✅ From Development Group:

- Source Code (GitHub Link or Zip)
- Deployment Link or Screenshots (if hosted)
- Project Guide/User Manual
- Video Demo (Optional)

✅ From Testing Group:

- Test Plan Document
- Test Case Sheet (Excel or Word)
- Bug Report (with severity & priority levels)
- Test Summary Report
- Optional:
  - Selenium Scripts (in Python/Java)
  - Postman Collections (for API testing)
  - Screenshots of Testing Results

9. Execution Plan / Timeline

Phase	Activity	Timeline
Phase 1	Group Formation & Project Selection	July - August
Phase 2	Application / Website / Software Development	August
Phase 3	Project Swapping & Test Planning	September
Phase 4	Test Case Execution & Reporting	September - October
Phase 5	Final Report Compilation & Viva	October

10. Expected Outcome

- Real-world understanding of **software development-testing interaction**.
- Improved knowledge of **test documentation standards and bug lifecycle**.
- Proficiency in **defect detection**, severity/priority classification, and **issue tracking**.
- Ability to convert **requirement documents into test cases and reports**.
- Experience with tools like **Selenium**, **Postman**, and **manual test planning**.
- Preparedness for **industry internships or junior QA roles**.
- Collaborative learning through **peer evaluation** and **cross-testing projects**.
- Confidence in participating in **Software Quality Assurance processes**.

11. Team Details

Sr. No.	Student Name	Enrollment No.	Email	Role (Dev/Tester)
1				
2				
3				
4				
5				

Group No.: ____

Cross-Tested By Group No.: ____

12. Execution Phases

Phase	Activity	Description	Date	Phase Sign
Phase 1	Group Formation and Project Topic Selection	Students form groups (2–5 members) and select topics		
Phase 2	Development of Application (Application / Website / Software)	Develop Web/Mobile application		
Phase 3	Group Swapping for Testing	Exchange projects with other groups for testing		
Phase 4	Manual/Automated Testing, Report Generation	Perform testing, write test cases, generate bug report		
Phase 5	Submission and Viva/Presentation	Submit all documents and present project & testing		

12. Faculty Guide Remarks

____

____

____

____