Software Test Plan

for the

Online Learning Platform

Version 4.0 (Approved)

Prepared by:

Antor Chandra Das

American International University-Bangladesh (AIUB)

June 24, 202

Table of Contents

TEST PLAN IDENTIFIER: Online Learning Platform

2.0 REFERENCES

2.1 Books and Journals:

Amland, S. (2000). Risk-based testing: Risk analysis fundamentals and metrics for software testing including a financial application case study. *Journal of Systems and Software*, 53(3), 287-295.

Anderson, T., & Dron, J. (2011). Three generations of distance education pedagogy. *International Review of Research in Open and Distance Learning*, 12(3), 80-97.

Clark, R. C., & Mayer, R. E. (2016). *E-learning and the science of instruction: Proven guidelines for consumers and designers of multimedia learning* (4th ed.). John Wiley & Sons.

Crispin, L., & Gregory, J. (2009). *Agile testing: A practical guide for testers and agile teams*. Addison-Wesley Professional.

Forsberg, K., & Mooz, H. (1991). The relationship of system engineering to the project cycle. *Engineering Management Journal*, 3(3), 36-43.

Holmes, A., & Kellogg, M. (2016). *Automating and testing a REST API: A case study in API testing using Java, REST Assured, Postman, Tracks, cURL and HTTP Proxies*. CreateSpace Independent Publishing Platform.

Myers, G. J., Sandler, C., & Badgett, T. (2011). *The art of software testing* (3rd ed.). John Wiley & Sons.

Ruiz-Calleja, A., Prieto, L. P., Ley, T., Rodríguez-Triana, M. J., & Sebastian, S. (2017). Learning analytics for professional and workplace learning: A literature review. *IEEE Transactions on Learning Technologies*, 10(3), 286-302.

2.2 Standards and Technical Documents:

Hardt, D. (Ed.). (2012). *The OAuth 2.0 authorization framework* (RFC 6749). Internet Engineering Task Force. https://tools.ietf.org/html/rfc6749

IEEE Computer Society. (2008). *IEEE standard for software and system test documentation* (IEEE Std 829- 2008). Institute of Electrical and Electronics Engineers.

ISO/IEC. (2011). *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models* (ISO/IEC 25010:2011). International Organization for Standardization.

2.3 Government and Institutional Report:

U.S. Department of Education. (2019). *Family Educational Rights and Privacy Act (FERPA) guidance for online education*. Office of Student Privacy.

UNESCO. (2021). *COVID-19 education: From disruption to recovery*. UNESCO Global Education Monitoring Report. https://en.unesco.org/covid19/educationresponse

World Economic Forum. (2020). *The future of jobs report 2020*. World Economic Forum Publications.

2.4 Web Resources and Documentation:

Mozilla Developer Network (2023).*Web security guidelines*. https://developer.mozilla.org/en-US/docs/Web/Security

OWASP Foundation. (2021). *OWASP top 10 web application security risks*. https://owasp.org/www-project- top-ten/

Selenium. (2023). *Selenium WebDriver documentation*. https://selenium-python.readthedocs.io/

Sommerville, I. (2015). *Software engineering* (10th ed.). Pearson Education.

Pressman, R., & Maxim, B. (2019). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.

2.5 Testing Methodologies:

Aggarwal, K. K., Singh, Y., & Chhabra, J. K. (2002). An integrated measure of software maintainability. *Proceedings of the Annual Reliability and Maintainability Symposium*, 235-241.

Graham, D., Van Veenendaal, E., Evans, I., & Black, R. (2012). *Foundations of software testing: ISTQB certification*. Cengage Learning.

Dustin, E., Rashka, J., & Paul, J. (1999). *Automated software testing: Introduction, management, and performance*. Addison-Wesley Professional.

2.6 Software Testing Resources:

Bachmann, A., & Diehl, S. (2007). *Software testing anti-patterns*. Proceedings of the 14th European Conference on Object-Oriented Programming, 431-455.

Beizer, B. (1995). *Black-box testing: Techniques for functional testing of software and systems*. John Wiley & Sons.

Kaner, C., Bach, J., & Pettichord, B. (2001). *Lessons learned in software testing*. John Wiley & Sons.

2.7 Educational Technology Research:

Ally, M. (2019). *Competency profile of the digital and online teacher in future education.*

International Review of Research in Open and Distributed Learning, 20(2), 302-318.

Garrison, D. R., & Kanuka, H. (2004). Blended learning: Uncovering its transformative potential in higher education. *The Internet and Higher Education*, 7(2), 95-105.

Siemens, G., & Long, P. (2011). Penetrating the fog: Analytics in learning and education. *EDUCAUSE Review*, 46(5), 30-32.

2.8 Software Engineering and Quality Assurance

Bourque, P., & Fairley, R. E. (Eds.). (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK Guide)* (Version 3.0). IEEE Computer Society.

3.0 INTRODUCTION

The rapid growth of online education has created a demand for comprehensive learning management systems (LMS) that can handle diverse educational needs **(Clark & Mayer, 2016)**. Recent studies indicate that over 70% of educational institutions have adopted some form of online learning platform since 2020 **(UNESCO, 2021).**

3.1 Background to the Problem

Educational technology has evolved significantly, with cloud-based platforms becoming the standard for remote learning delivery **(Anderson & Dron, 2011).**

Today, many students face problems when it comes to attending regular classes. Some live far from educational institutions, some have jobs, and others may have personal reasons that make it hard to go to class every day. Also, during emergencies like COVID- 19 or natural disasters, classes often get delayed or canceled because students and teachers can't meet in person. The COVID-19 pandemic accelerated the adoption of digital learning platforms by an estimated 5-10 years **(World Economic Forum, 2020).**

To deal with this, many institutions use tools like Zoom, email, or YouTube to continue classes. But using different tools separately can be messy. Students often get confused about where to find their lessons, and teachers struggle to manage everything in one place. There’s no proper system to track student progress, give quizzes, or share course materials in an organized way.

All these issues show that there’s a need for one easy, complete platform that can bring everything together and make learning possible anytime, from anywhere.

3.2 Solution to the Problem

To solve this, we plan to create an **Online Learning Platform** that brings all learning tools into one place. It will help both students and teachers by making learning easier, faster, and more organized.
This platform will let students:

- Register and log in easily
- Browse and join courses
- Watch video lectures and download class notes
- Take quizzes and submit assignments
- Track their learning progress
- Communicate with teachers and classmates Teachers and admins will be able to:
- Create and manage courses
- Upload lessons and materials
- Give and check quizzes or assignments
- See each student's performance
- Send updates or messages

This system will save time for everyone and make learning more flexible. Teachers can oversee classes without additional stress, and students can learn at their own pace. Overall, this platform is a smart and simple solution for modern education.

4.0 REQUEIREMNT SPECIFICATION

4.1 System Features:

4.1.1 User Login

Users (students/instructors/admins) can log in using their valid email and password.
If login is successful, they are redirected to their dashboard. If not, an error message is shown.
Priority Level: High
Precondition: The user needs to have a registered account with current login information

4.1.2 Course Enrollment

Students can enroll in free or paid courses from the course list.

Enrollment is immediate for free courses; paid courses require successful payment.

Priority Level: High

Precondition: Student must be logged in.

4.1.3 Watch Course Videos

Students can access and watch uploaded video lectures of enrolled courses.

Videos are arranged by modules and lessons.

Priority Level: High

Precondition: Student must be enrolled in the course.

4.1.4 Download Course Materials

Students can download notes, slides, or other files shared by instructors.

Files are available in each lesson.

Priority Level: Medium

Precondition: Student must be enrolled in the course.

4.1.5 Take Quizzes

Students can take quizzes after completing lessons.

Results are shown instantly or after instructor review.

Priority Level: High

Precondition: Quiz must be published by the instructor.

4.1.6 Submit Assignments

Students can upload assignments for evaluation.

The assignment tab displays instructions and deadlines.

Priority Level: High

Precondition: Assignment must be active.

4.1.7 View Progress Dashboard

Users can see course progress, quiz results, and completion percentage.

Visual indicators like progress bars or charts will be shown.

Priority Level: Medium

Precondition: User must be enrolled in at least one course.

4.1.8 Instructor Course Upload

Instructors can create new courses, upload videos, and add materials.

They can also set quizzes and assignments for each course.

Priority Level: High

Precondition: Instructor must have a verified instructor account.

4.1.9 Admin Course Approval

Admin reviews and approves courses submitted by instructors before they are published.

Can accept, reject, or request changes.

Priority Level: High

Precondition: Course must be submitted by instructor.

4.1.10 Messaging System

Messages about help or courses can be sent and received by users..

Instructors can reply to student queries.

Priority Level: Medium

Precondition: Both sender and receiver must be registered users.

4.1.11 Certificate Generation

After the course completion, students receive a downloadable certificate.

The course name, student name, and completion date are all listed on the certificate.

Priority Level: Medium

Precondition: Student must complete all course modules and pass required assessments.

4.1.12 Admin User Management

Admin can add, edit, or delete users (students or instructors).

Also manages user roles and permissions.

Priority Level: High

Precondition: Admin must be logged in.

4.1.13 User Logout

Any user can securely log out of their account.

After logout, they are redirected to the login page.

Priority Level: High

Precondition: User must be logged in.

4.2 System Quality Attributes

There is some software quality attributes as per ISO/ IEC 9126 that are very important to ensure the quality of software. The ISO/IEC 25010 quality model provides a comprehensive framework for evaluating software quality characteristics including functionality, reliability, and usability **(ISO/IEC, 2011).** Security in educational platforms must address data privacy concerns, particularly regarding student information protection under regulations like FERPA **(U.S. Department of Education, 2019).**

Functionality: Functionality ensures that only authenticated administrators ("valid Admins") can access all services and features after logging in. Anyone not recognized as a valid admin ("Invalid Admin") is denied access, enhancing system security and ensuring that only authorized personnel can operate within the administrative environment.

Security: System security should be sufficient to prevent unauthorized access to the system operations. This involves implementing strong authentication protocols, encrypting data transmissions, and possibly using intrusion detection systems to ensure that only authorized users can interact with the system and that their actions are securely managed. This all-encompassing security strategy aids in safeguarding private information and preserving the integrity of system functionality.

Reliability: All features will work as intended across a range of working environments or devices. This implies thorough testing and optimization to handle different hardware configurations, operating systems, and network conditions, guaranteeing stable and dependable user experience regardless of the platform used.

Usability: It is easy for everyone to understand the "Online Learning Platform" system. This means the interface is intuitive, instructions are clear, and users can efficiently complete tasks without confusion or the need for extensive training.

Efficiency: Our system size is small and efficient so that it can be handled by any device. This ensures quick response times and minimal resource consumption, making the system accessible and practical for a wide range of users.

Maintainability: If a bug or problem is found in the system, it will be solved as soon as possible. The system remains reliable and performs well over time, with minimal disruption to users. It reflects a commitment to regular updates and effective problem-solving to enhance user experience and system stability.

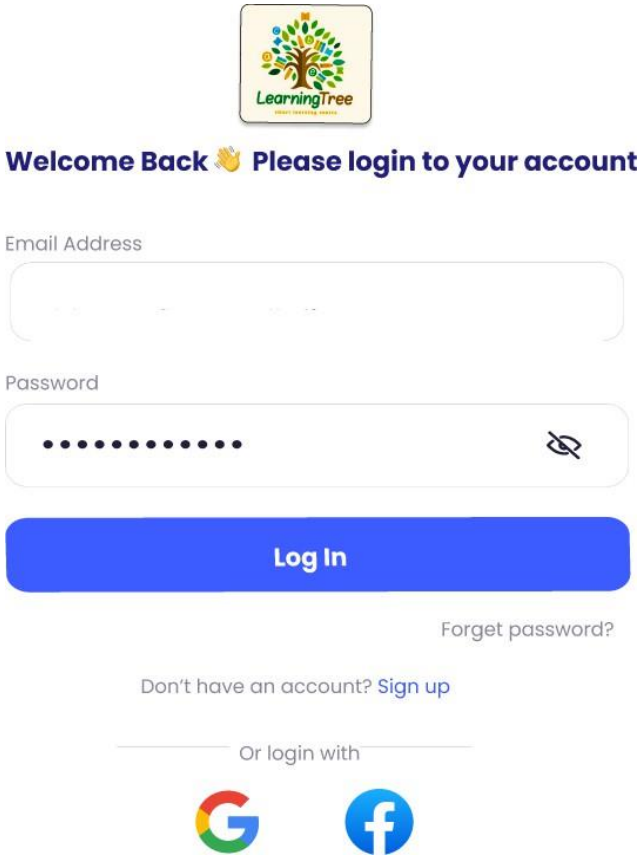
Portability: Switching the host or environment can be done in a short time. Reinstallation of the software can be done easily as well. This flexibility allows for quick adaptations to new systems, facilitating seamless transitions with minimal setup time, which is ideal for maintaining operations across diverse platforms

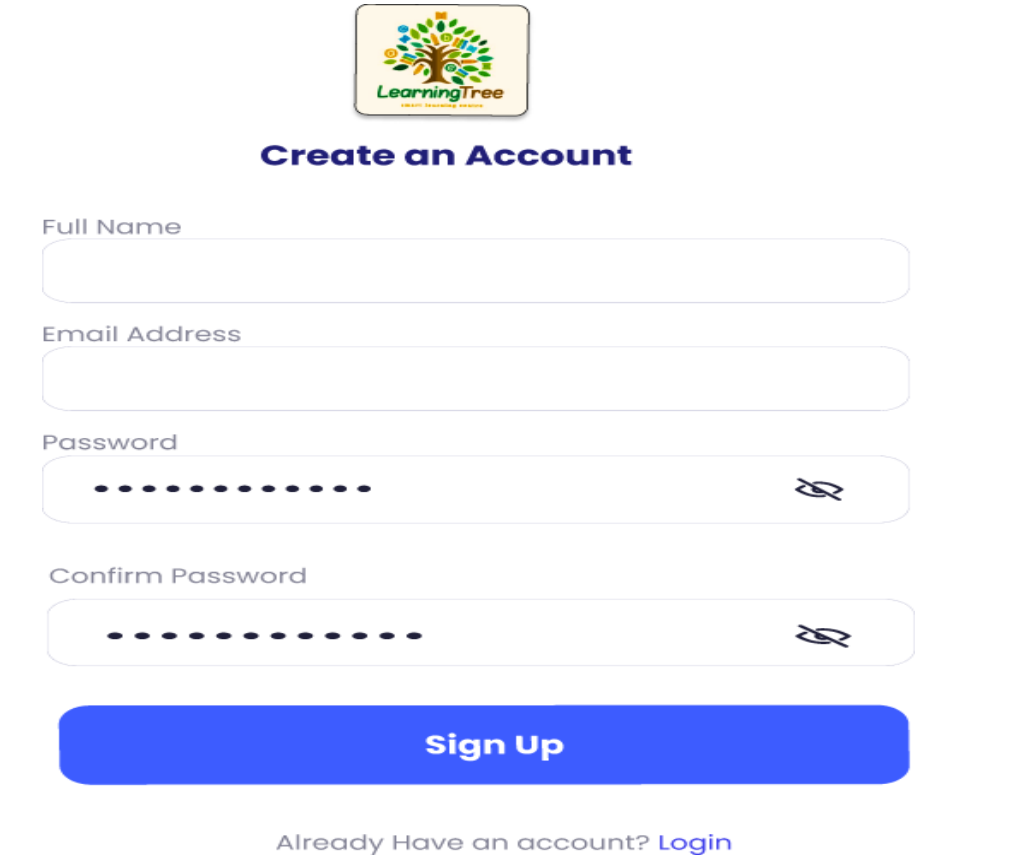
Accessibility: As it is web-based software, it can be accessed from anywhere through the Internet. Here users can reach and use the system conveniently, whether from home, work, or on the go, enhancing its usability and availability.

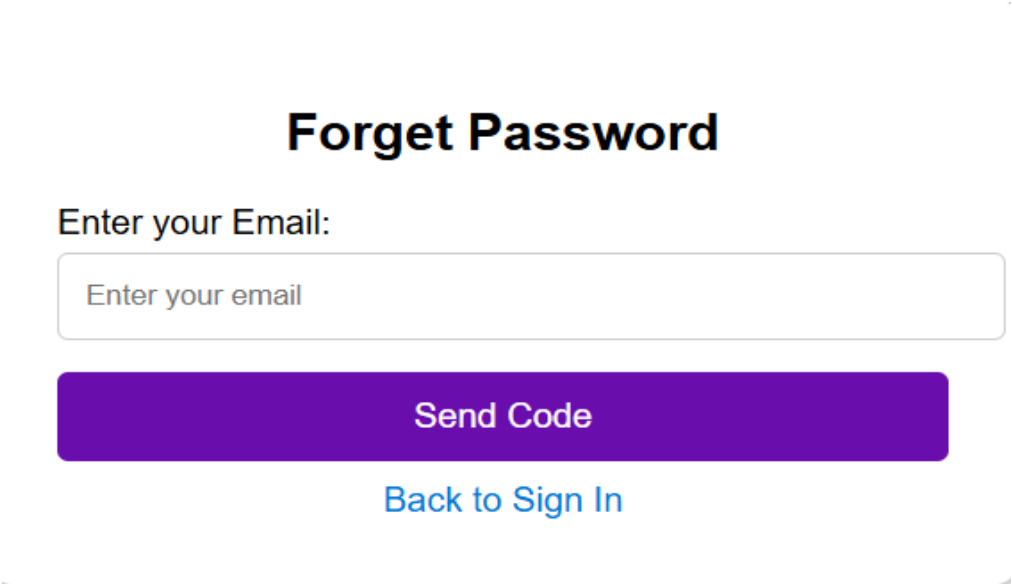
Installation: There won't be any time-consuming downloads or installations because it is web- based. It is based on web addresses. It is really simple for everyone to access

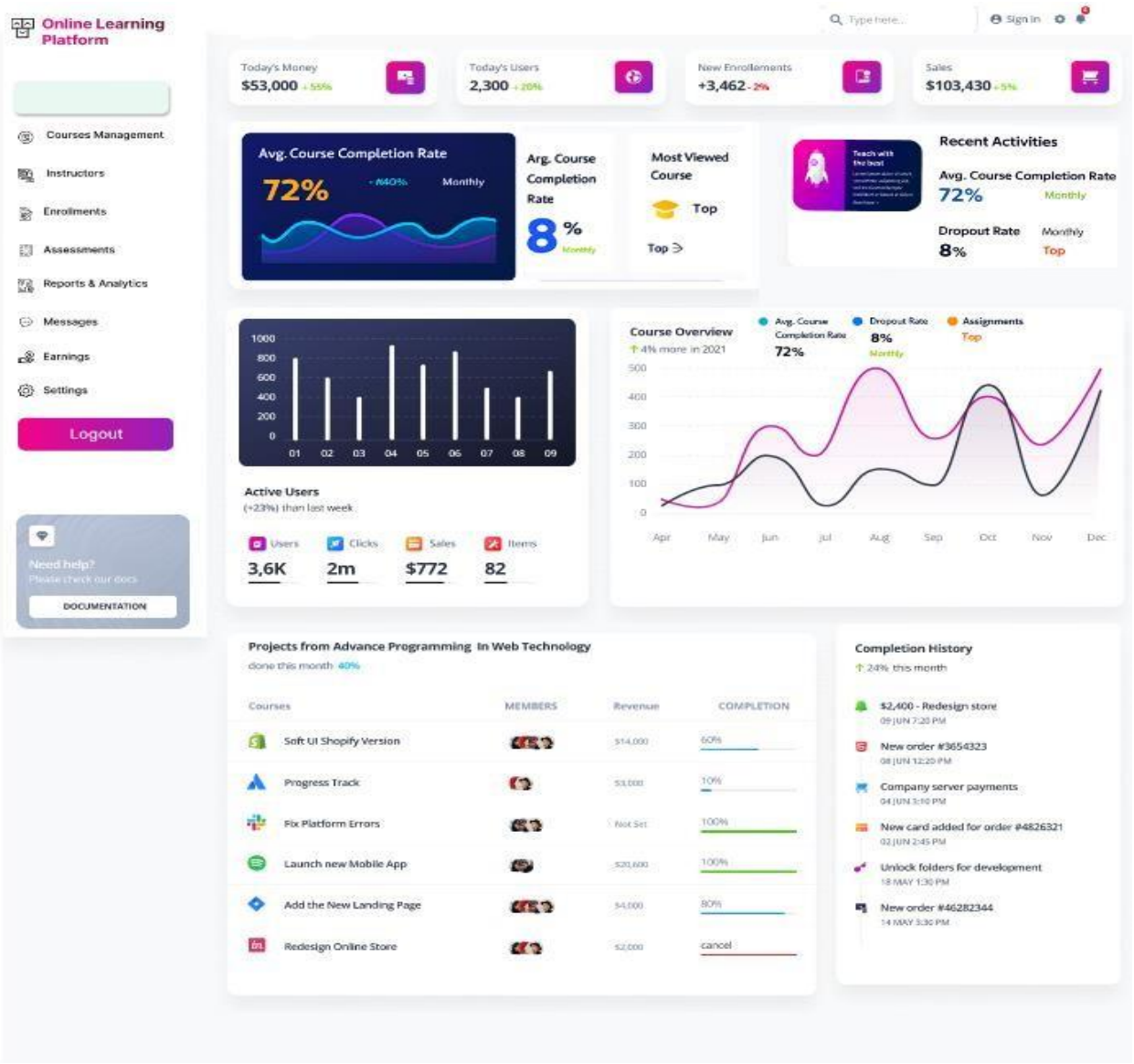
4.3 System Interface

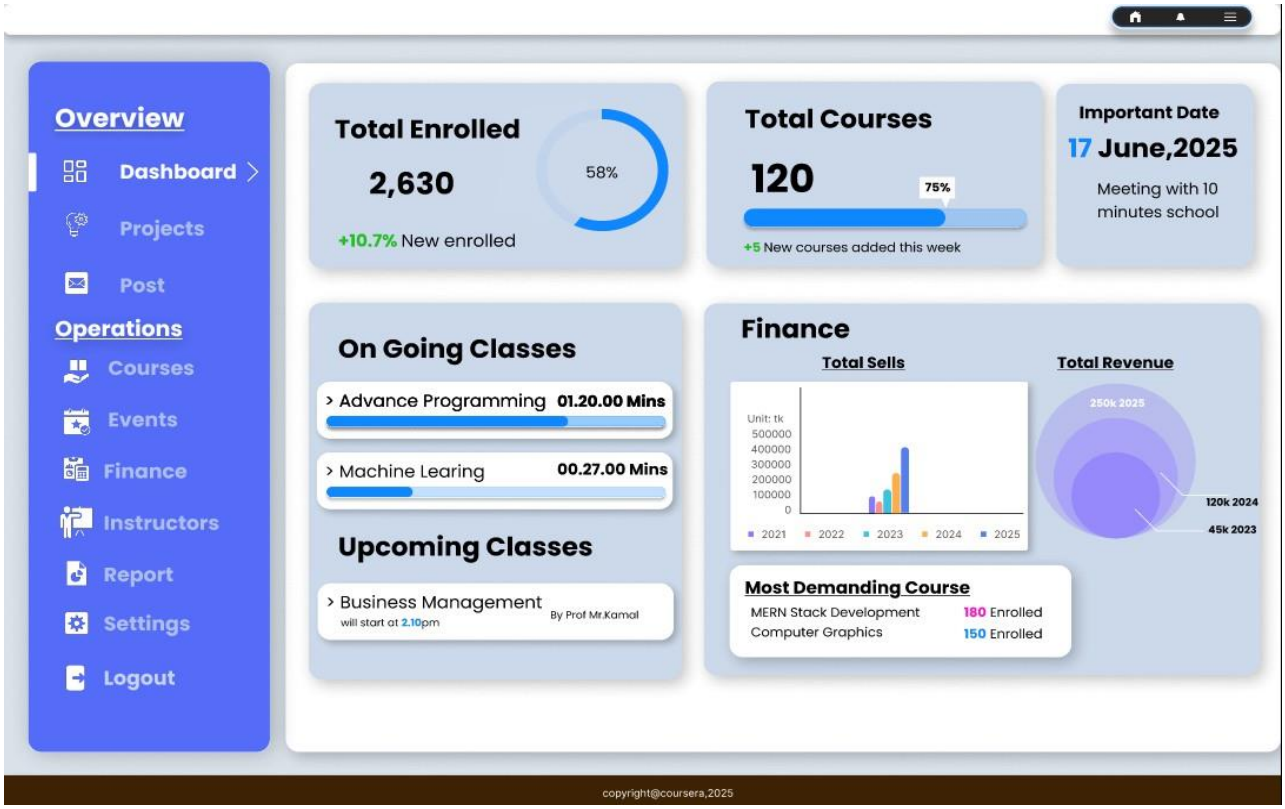
This is the home user system interface. All admins will first see this home page when they visit this Faculty-Course-Allocation Web application.

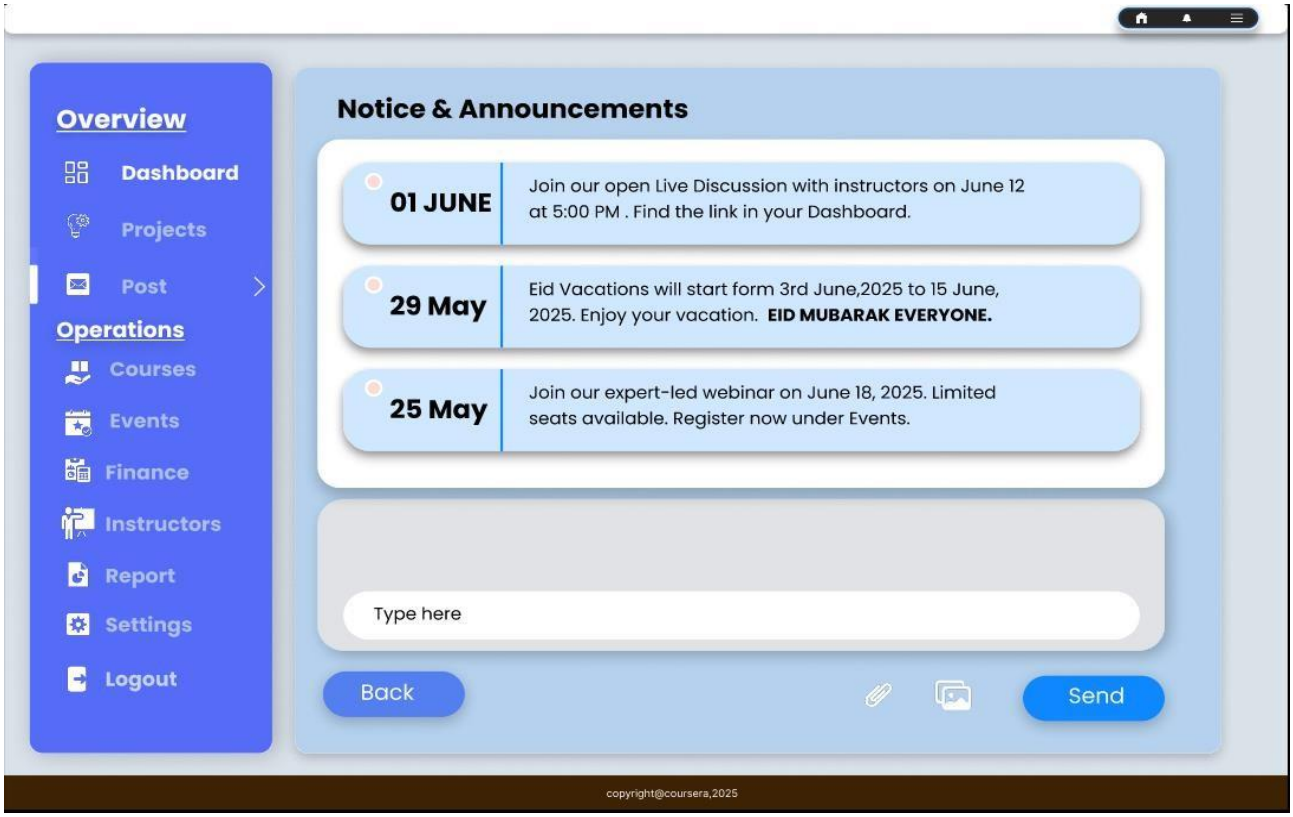


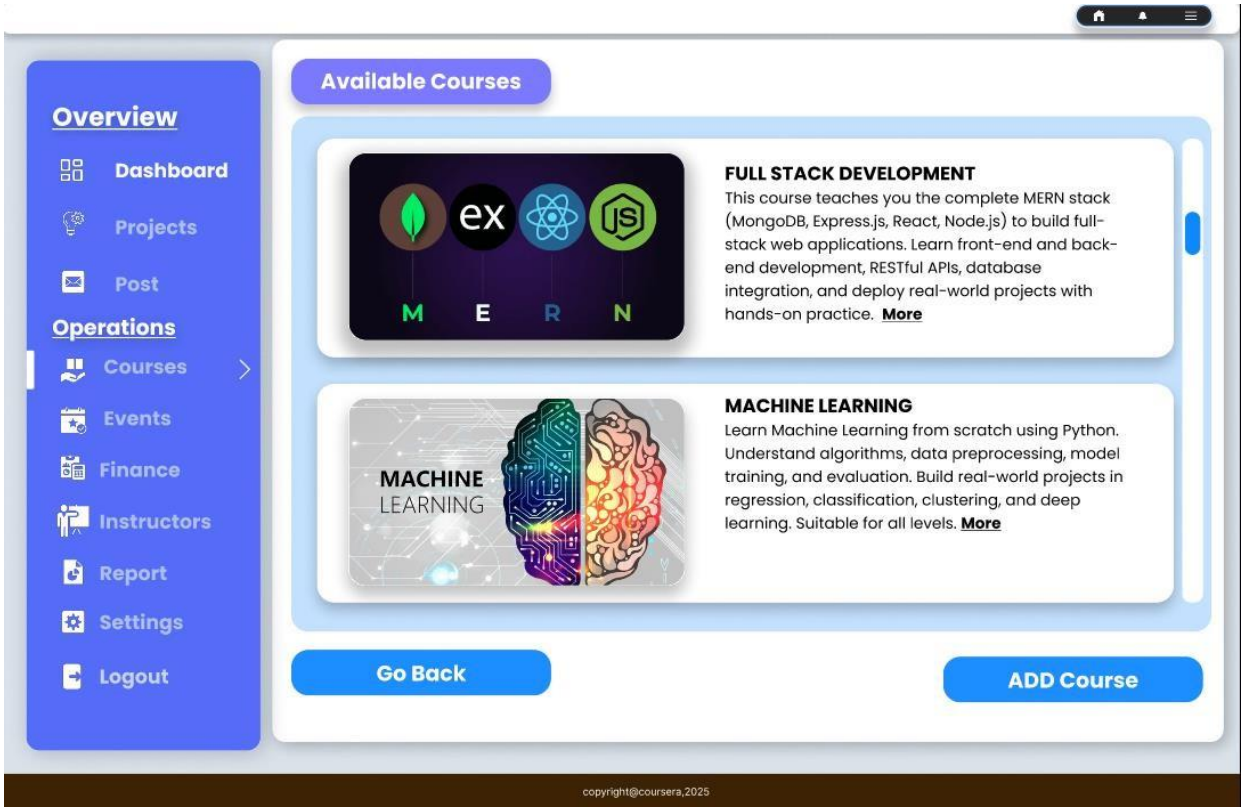

LearningTree
Create an Account
Full Name
Email Address
Password
Confirm Password
Sign Up
Already Have an account? Login

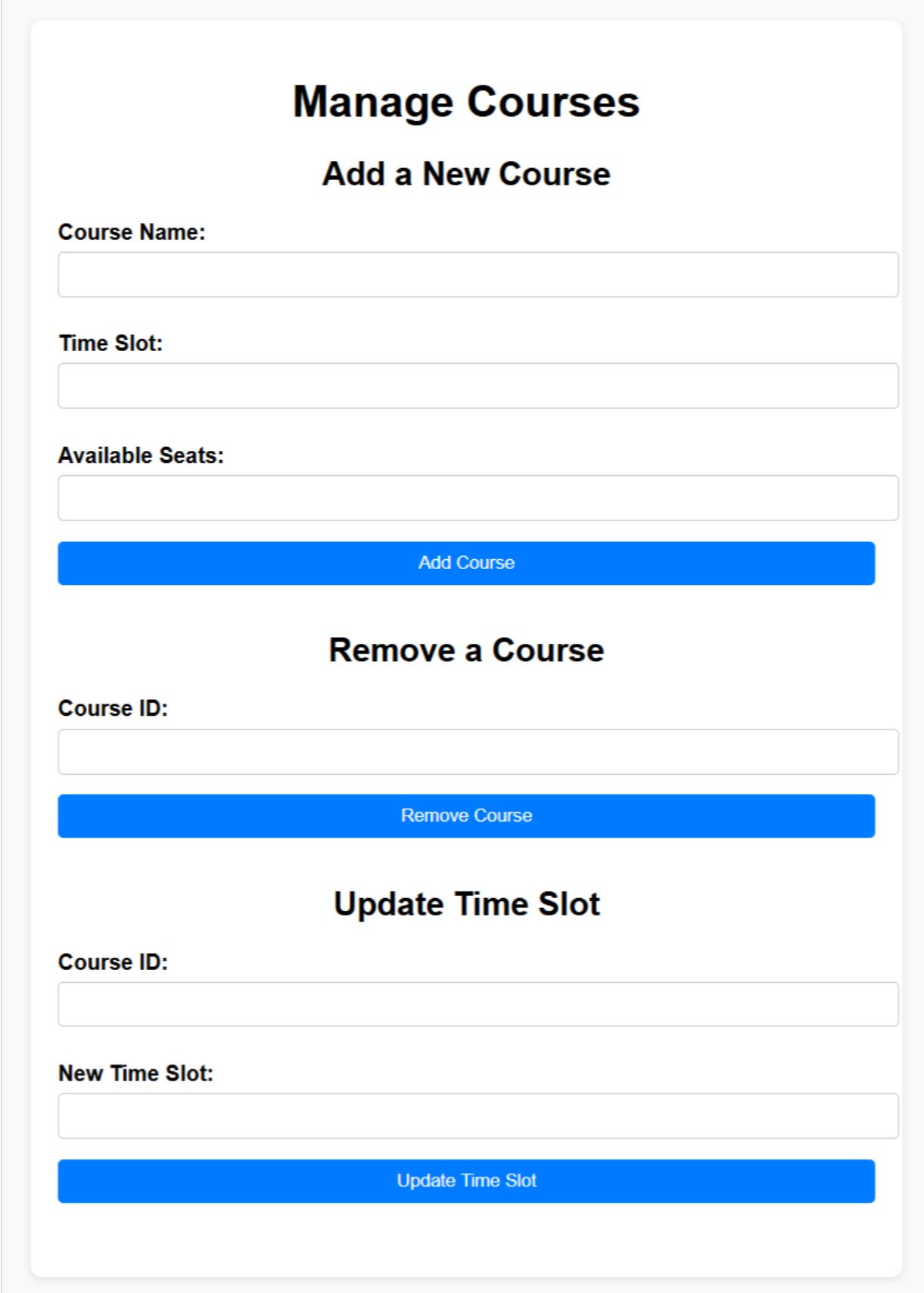

Forget Password
Enter your Email:
Enter your email
Send Code
Back to Sign In

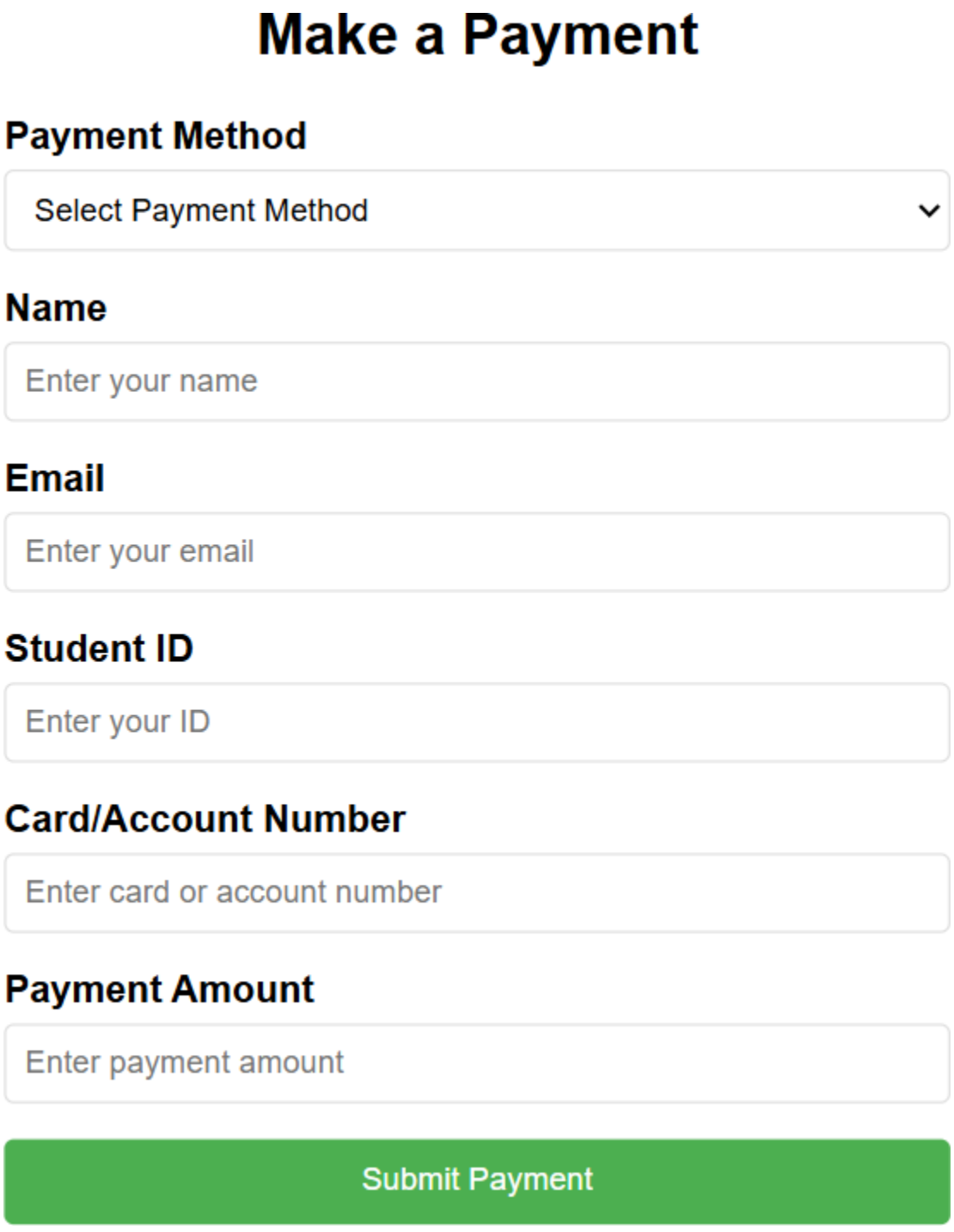

Online Learning Platform
Courses Management
Instructors
Enrollments
Assessments
Reports & Analytics
Messages
Earnings
Settings
Logout
Need help?
DOCUMENTATION
Today's Money
$53,000
Today's Users
2,300
New Enrollements
+3,462
Sales
$103,430
Avg. Course Completion Rate
72%
Monthly
Arg. Course Completion Rate
8%
Most Viewed Course
Top
Recent Activities
Avg. Course Completion Rate
72%
Dropout Rate
8%
Active Users
(+23%) than last week
Users
3,6K
Clicks
2m
Sales
$772
Items
82
Course Overview
Projects from Advance Programming In Web Technology
Courses
MEMBERS
Revenue
COMPLETION
Soft UI Shopify Version
Progress Track
Fix Platform Errors
Launch new Mobile App
Add the New Landing Page
Redesign Online Store
Completion History
$2,400 - Redesign store
New order #3654323
Company server payments
New card added for order #4826321
Unlock folders for development
New order #46282344

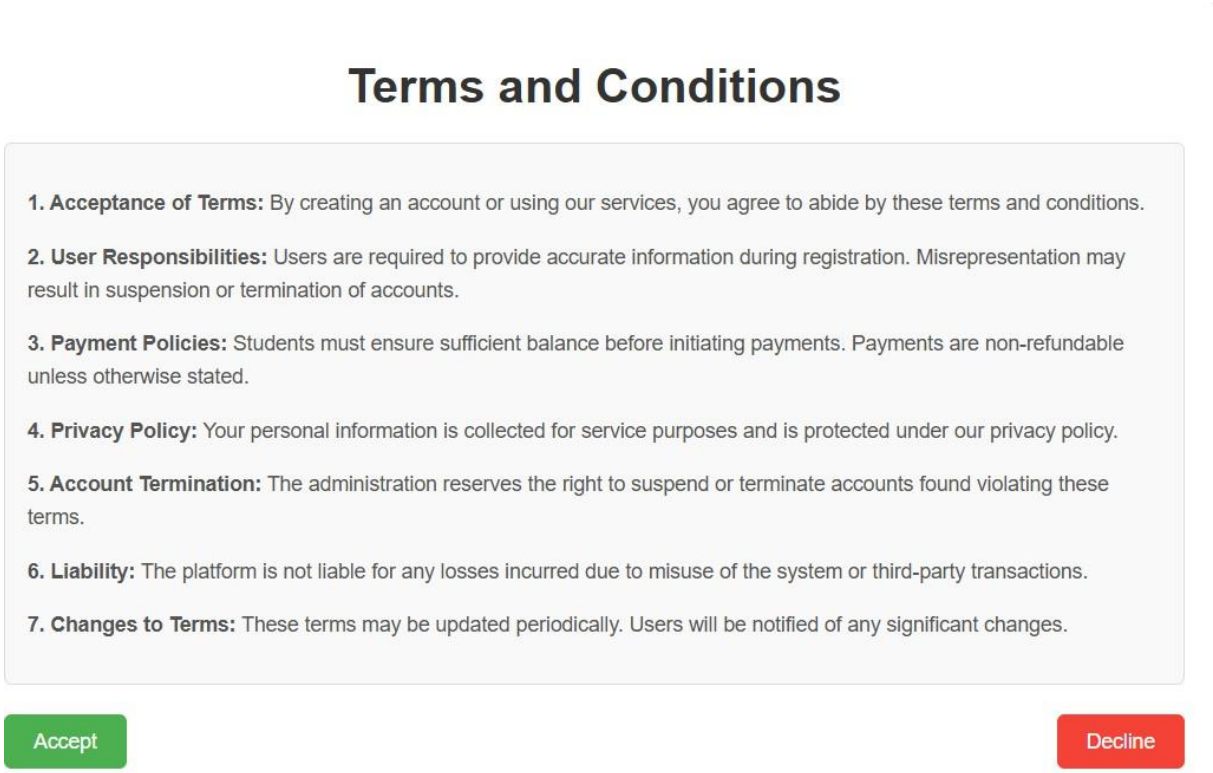

Overview
Dashboard
Projects
Post
Operations
Courses
Events
Finance
Instructors
Report
Settings
Logout
Total Enrolled
2,630
58%
+10.7% New enrolled
Total Courses
120
75%
+5 New courses added this week
Important Date
17 June,2025
Meeting with 10 minutes school
On Going Classes
Advance Programming 01.20.00 Mins
Machine Learing 00.27.00 Mins
Upcoming Classes
Business Management
will start at 2.10pm
By Prof Mr.Kamal
Finance
Total Sells
Total Revenue
Unit: tk
500000
400000
300000
200000
100000
0
2021
2022
2023
2024
2025
250k 2025
120k 2024
45k 2023
Most Demanding Course
MERN Stack Development 180 Enrolled
Computer Graphics 150 Enrolled
copyright@coursera,2025


Overview
Dashboard
Projects
Post
Operations
Courses
Events
Finance
Instructors
Report
Settings
Logout
Notice & Announcements
01 JUNE
Join our open Live Discussion with instructors on June 12 at 5:00 PM . Find the link in your Dashboard.
29 May
Eid Vacations will start form 3rd June,2025 to 15 June, 2025. Enjoy your vacation. EID MUBARAK EVERYONE.
25 May
Join our expert-led webinar on June 18, 2025. Limited seats available. Register now under Events.
Type here
Back
Send
copyright@coursera,2025


Overview
Dashboard
Projects
Post
Operations
Courses
Events
Finance
Instructors
Report
Settings
Logout
Available Courses
FULL STACK DEVELOPMENT
This course teaches you the complete MERN stack (MongoDB, Express.js, React, Node.js) to build full-stack web applications. Learn front-end and back-end development, RESTful APIs, database integration, and deploy real-world projects with hands-on practice. More
MACHINE LEARNING
Learn Machine Learning from scratch using Python. Understand algorithms, data preprocessing, model training, and evaluation. Build real-world projects in regression, classification, clustering, and deep learning. Suitable for all levels. More
Go Back
ADD Course
copyright@coursera,2025


Manage Courses
Add a New Course
Course Name:
Time Slot:
Available Seats:
Add Course
Remove a Course
Course ID:
Remove Course
Update Time Slot
Course ID:
New Time Slot:
Update Time Slot


Make a Payment
Payment Method
Select Payment Method
Name
Enter your name
Email
Enter your email
Student ID
Enter your ID
Card/Account Number
Enter card or account number
Payment Amount
Enter payment amount
Submit Payment



4.4 Project Requirements

- Time: This web-based application may take about 1.5 months to complete
- Budget: 4,50,000 BDT
- Size: The final size of this web-based application will not be more than 500-600 MB.
- .NET Core,Node.js, React/Next.js,tailwind.css,Typescript,ES6+ will be used to build this web-based application.

5.0 FEATURES NOT TO BE TESTED

There may be a number of good things about this project; you will want to test all of these features behind the scenes to ensure the project that comes to launch is what your users are looking for. Of which load test and performance test are found as important units. It's critical to be able to understand how your application performs under different workloads, so you can make confident predictions about how it will handle traffic without falling over. Performance testing evaluates the application's performance and responsiveness in both typical and extreme scenarios. By running thorough tests in these specific areas, we can spot potential bottlenecks, optimize the use of resources and ultimately the user experience. And most importantly prioritizing the testing of load and performance features ensures the system's reliability and scalability, and that our students have a software platform that is at the same high standard of who we are as a university technologically and educationally.

6.0 TESTING APPROACH

This online learning platform will be tested at four different levels: acceptability, system, integration, and unit.

Test-case design should involve the boundary value analysis and equivalence partitioning to achieve a full coverage testing (Myers et al., 2011). Acceptance testing measures should follow from and satisfy the business requirements and users' expectations (Crispin & Gregory, 2009) System and integration testing Under the current proposal, at least one full time independent tester is proposed for system and integration testing.

be present. However, because of the schedule and budget constraints, the test manager will have to do the bulk of the testing with assistance from the development team.

6.1 Testing Levels

Unit Testing:

It's the focus because the developer is doing it into unit testing, the first phase of testing. After a snippet of the software is developed, a programmer will test it for an expected outcome. It will be signed off by the dev team lead.Motivation The test person is informed about the present state of the software with the provided progress report for the unit testing.

Integration Testing:

Unit testing is done initially, followed by integration testing performed by a team of testers (whose only job is testing). Ironically, the smaller parts will come together at last. Also, we will look at the whole system to confirm that the new module ties in well with the

existing system, after creating a smaller part of the system.

SYSTEM TESTING:

Systems testing: Systems testing should be performed after the integration testing to guarantee that all its modules operate as a single unit and at the completion of integration. The test is a black box. A test case is written using the needs and product design so that the whole system is verified without necessarily having to know the behavior of each module.

ACCEPTANCE TESTING:

The final stage of software testing is acceptance testing. It's not the writers who actually do it. The software is made available to the general public in beta form. Users review the program after they use it, based on their responses.
Issues are resolved on the spot. Acceptance testing is a proof of the general quality of the program and validation of the work done by development and testing teams.

6.2 Test Tools

Selenium:

Please use only selenium WebDriver for testing. A nimble tester is more able to provide more sophisticated test cases by employing Selenium, which is used to automate duplicate test scripts for browser-based web applications.

6.3 Meetings

The test team will meet weekly to evaluate progress to date, quickly identify error tendencies and problems. The project manager will also meet with the head of the test team bi-weekly with development as a participant. We'll organize these two meetings for separate weeks. Additional meetings could be called in case of an emergency.

7.0 TEST CASES/TEST ITEMS

Test Case 1: User Login with Valid Credentials

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-01			Test Designed Date: 06/06/25	
Test Priority: High			Test Executed By: **Antor**	
Module name: Authentication			Test Execution Date:10/06/25	
Test Title: Successful Login with Valid Email/Password.				
Description: Verify user can log in with registered credentials.				
Precondition: User account exists (e.g., email: user@domain.com)				
Test Steps	Test Data	Expected Results	Actual Results	Status
1. Go to the login page 2. Enter email 3. Enter Password 4. Click the Login button	Email: user@domain.com Password: ValidPass1 23!	Redirect to Dashboard		Pass
Post Condition: Dashboard page loads with user-specific data				

Test Case 2: Password Recovery via Email

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-02			Test Designed Date: 06/06/25	
Test Priority: High			Test Executed : Antor	
Module name: Password Recovery			Test Execution Date: 06/06/25	
Test Title: Send Recovery Code to Registered Email				
Description: Verify system sends code to email for password reset.				
Precondition: Valid email exists in system				
Test Steps	Test Data	Expected Results	Actual Results	Status
Click "Forget password?" Enter email Click "Send Code"	Email: user@domain.com	"Send Code" button enabled Confirmation message		Pass
Post Condition: Code received in user's email inbox				

Test Case 3: Course Enrollment (Free Course)

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-03			Test Designed Date: 06/06/25	
Test Priority: Medium			Test Executed By: Antor	
Module name: Course Management			Test Execution Date: 10/06/25	
Test Title: Enroll in Free Course				
Description: Verify user can enroll in free course without payment.				
Precondition: User logged in; Course has "Free" tag				
Test Steps	Test Data	Expected Results	Actual Results	Status
Open course page Click "Enroll"	Course: "Web Development Bootcamp"	Immediate enrollment		Pass
Post Condition: Course appears in "My Courses" section				

Test Case 4: Payment Processing

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-04			Test Designed Date: 06/06/25	
Test Priority: High			Test Executed By: Antor	
Module name: Payment Gateway			Test Execution Date:10/06/25	
Test Title: Successful Course Payment				
Description: Verify payment submission with valid details.				
Precondition: User selected paid course				
Test Steps	Test Data	Expected Results	Actual Results	Status
Fill in the payment form Enter details Click "Submit Payment"	Card: 4111-1111-1111-1111 Expiry: 12/25, CVV: 123 Amount: $50	Payment success confirmatio n	Payment success confirmatio n	Pass
Post Condition: Course access granted; Receipt emailed				

Test Case 5: Dashboard Data Accuracy

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-05			Test Designed Date: 06/06/25	
Test Priority: Medium			Test Executed By: Antor	
Module name: Dashboard			Test Execution Date: 10/06/25	
Test Title: Verify Learning Progress Metrics				
Description: Confirm dashboard displays correct progress %.				
Precondition: User has completed 3/4 lessons in a course				
Test Steps	Test Data	Expected Results	Actual Results	Status
Log in Navigate to Dashboard		Progress = 75%		Pass
Post Condition: Redirected to Add Priority Course page				

Test Case 6: Course Creation (Admin)

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-06			Test Designed Date: 06/06/25	
Test Priority: High			Test Executed By: Antor	
Module name: Admin - Course Management			Test Execution Date: 10/06/25	
Test Title: Add New Course				
Description: Verify admin can create new courses				
Precondition: Admin logged in				
Test Steps	Test Data	Expected Results	Actual Results	Status
Go to "Manage Courses" Enter course details Set time/seats Click "Add Course"	Name: "Python Advanced" Time: "Mon/Wed 10AM", Seats: 30	"Course added" confirmation		Pass
Post Condition: New course appears in course catalog				

Test Case 7: Terms and Conditions Acceptance

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-07			Test Designed Date: 06/06/25	
Test Priority: Medium			Test Executed By: Antor	
Module name: Account Registration			Test Execution Date: 10/06/25	
Test Title: Mandatory ToS Acceptance				
Description: Verify user must accept Terms before payment.				
Precondition: User at payment step for paid course				
Test Steps	Test Data	Expected Results	Actual Results	Status
Attempt payment without ToS Check the “Accept" box		"Accept Terms" error. Submit button enabled	Edit Faculty Information successful	Pass
Post Condition: Payment proceeds after acceptance				

Test Case 8: Navigation Menu Functionality

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-08			Test Designed Date: 06/06/25	
Test Priority: Low			Test Executed By: Antor	
Module name: UI Navigation			Test Execution Date: 10/06/25	
Test Title: Menu Item Redirection				
Description: Verify all menu items lead to correct pages.				
Precondition: User logged in				
Test Steps	Test Data	Expected Results	Actual Results	Status
Click "Settings" Click "Reports" Click "Logout"		Settings page loads Analytics page loads Redirect to login screen		Pass
Post Condition: Correct pages render for each selection				

Test Case 9: Continue Learning Feature

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-09			Test Designed Date: 06/06/25	
Test Priority: Medium			Test Executed By: Antor	
Module name: Course Progress			Test Execution Date: 10/06/25	
Test Title: Resume Incomplete Lesson				
Description: Verify "Continue" button resumes last unfinished lesson.				
Precondition: User partially completed "JavaScript Essentials"				
Test Steps	Test Data	Expected Results	Actual Results	Status
Go to Dashboard Click "Continue" under JS course		Last unfinished lesson opens		Pass
Post Condition: Lesson resumes from previous progress point				

Test Case 10: Time Slot Update (Admin)

Project Name: Online Learning Platform			**Test Designed By: Antor**	
Test Case ID: OLP-10			Test Designed Date: 06/06/25	
Test Priority: High			Test Executed By: Antor	
Module name: Admin - Scheduling			Test Execution Date: 10/06/25	
Test Title: Modify Course Schedule				
Description: Verify admin can update course time slots				
Precondition: Course "Machine Learning" exists				
Test Steps	Test Data	Expected Results	Actual Results	Status
1. Go to "Update Time Slot" 2. Enter course ID Enter new slot Click "Update"	ID: ML-101 Slot: "Tue/Thu 2-4PM"	"Schedule updated" message		Pass
Post Condition: Course details reflect new schedule				

8.0 ITEM PASS/FAIL CRITERIA

The main objective of this section is to describe the PASS/FAIL criteria for the tests that are a part of this project. Any system or unit receiving a score of less than 90% will be subject to the failure criteria, and any component, unit, system, or integrated test item receiving a score of 90% to 95% will be considered to meet the pass criterion

9.0 TEST DELIVERABLES

Documents known as test deliverables are distributed to the stakeholders during the software development process. It includes a list of records, instruments, and other supplies that need to be made, made available, and kept up to date to facilitate testing activities inside a project.

Results and conclusions from unit testing shall be appropriately documented. A continuous progress report is necessary to stay on course.

Acceptance test participants are carefully chosen because the wrong people can produce inaccurate input and findings. It resembles an agreement for software delivery and release from the development team.

New modules are included into the system during the integration testing phase. It was also necessary to preserve these records for future verification.

A detailed report with the test findings will be provided whenever each testing phase is finished.

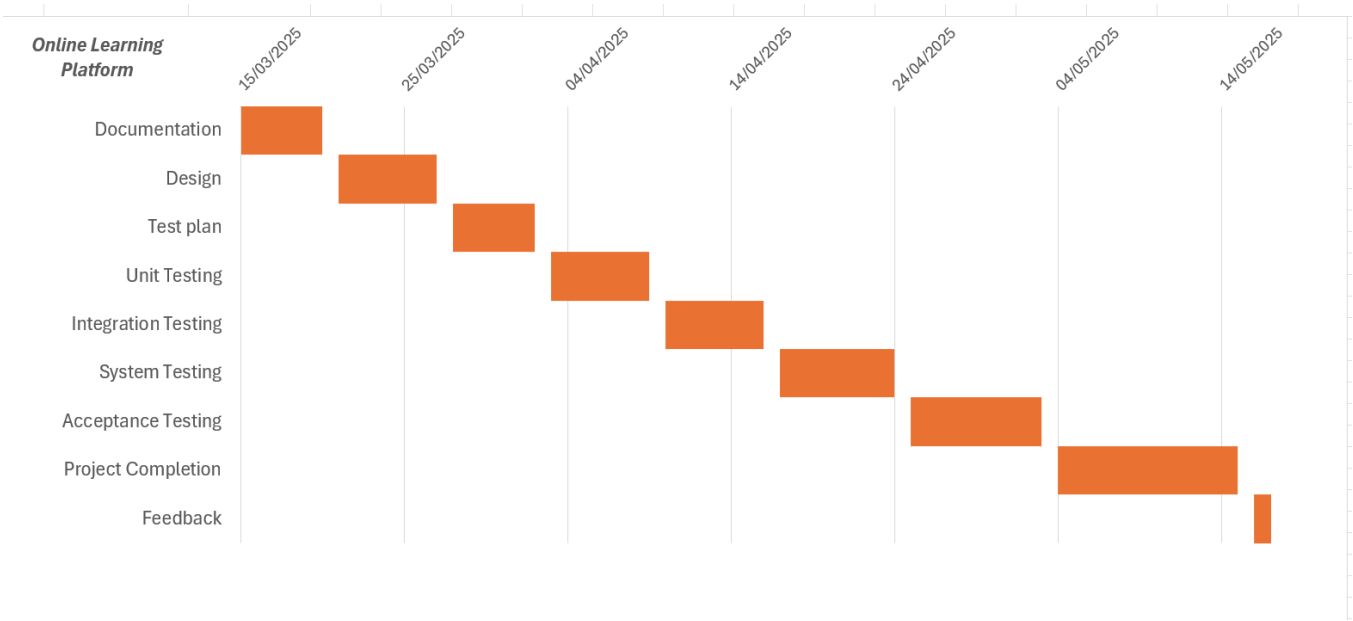
10.0 STAFFING AND TRAINING NEEDS

The purpose of this staffing approach is to increase the probability that the project will be assigned to enough qualified individuals to guarantee its effective completion. Employees must have the right staffing and training to foster creativity and boost productivity for the creation of products. Throughout the project's acceptance and system/integration testing stages, we require at least one full-time tester. For the first four months, a devoted tester will work on the project full-time. The test manager fills in when there isn't enough time for a specialized tester. Basic training on our project's user interface will be required for developers and testers. Before the project is approved, operations personnel must also complete extensive training in this project communication protocol. We must bring all the tools required to help the testing team because we will be using Selenium, and if necessary, training must also be given.

11.0 RESPONSIBILITIES

	TM	PM	Dev. Team	Test Team	Client
Acceptance test Documentation & Execution	X	X		X	X
System/Integration test Documentation & Exec.	X		X	X	
Unit test documentation & execution	X		X	X	
System Design Reviews	X	X	X	X	X
Detail Design Reviews	X	X	X	X	X
Test Procedures and rules	X	X	X	X	
Screen & Report Prototype reviews			X	X	X
Change control and Regression testing	X	X	X	X	X

12.0 TESTING SCHEDULE



13.0 PLANNING RISKS AND CONTINGENCIES

A project's ability to manage risk and handle emergencies well is critical to its success. It is used in a project to control the exception risk. Risk-based testing approaches help prioritize testing efforts based on potential impact and likelihood of failure **(Amland, 2000)**. The product created cannot accomplish its own objective if it is not in line with the service areas, ethics, and etiquette. In order to help the organization deal with unforeseen circumstances, there are also some norms and regulations. This kind of system must exist, and it must be properly maintained.

14.0 APROVALS

Project Sponsor	
Development Management	
EDI Project Manager	
RS Test Manager	
RS Development Team Manager	
Reassigned Sales	
Order Entity EDI Team Manager	