**Sprint 1 Product Review & Retrospective** **Online Course Registration System**

### Team Members:

* Mohamed Elnafe
* Andrew Slobodnik
* Mohamed Salman
* Robert Freeman

## Sprint 1 Product Review

#### **1. Sprint Goal and Accomplishments (10 Points)**

**Sprint Goal:** To establish a secure and functional foundation for the online course registration system by implementing essential user registration, authentication, and course browsing features.

**Accomplishments:**

* **User Registration and Authentication:**
  + Registration page UI implemented with form validation using JavaScript.
  + Registration form successfully integrated with backend API.
  + Secure login system is complete; sessions are managed correctly.
* **Course Browsing and Search:**
  + Course listing page designed and implemented.
  + Users can filter courses by category and department.
  + Backend API successfully integrated to fetch course data dynamically.

These accomplishments align with our product vision to deliver a scalable, user-centric registration platform.

#### **2. Software Demonstration (25 Points)**

**Technologies Used:**

* **Frontend:** React.js (hosted on AWS S3)
* **Backend:** Node.js + Express (hosted on EC2)
* **Data Handling:** Local schema structure implemented via MySQL Workbench
* **Security:** VPC, security groups, HTTPS

**Demonstrated Features:**

* **Student Registration & Login:**
  + Secure signup with real-time validation.
  + Form submission integrated with backend API.
  + Secure login/logout flow fully implemented.
* **Course Browsing by Category:**
  + Dynamic filtering of courses.
  + Responsive UI for smooth navigation.
  + API integration shows static course listings.

#### **3. Challenges and Resolutions (5 Points)**

| **Challenge** | **Resolution** |
| --- | --- |
| Form flow issues | Refactored form logic, added clearer success/failure messaging, and aligned API request handling with backend |
| Backend-to-frontend sync issues | Created shared API documentation and added weekly integration meetings |
| AWS budget and service limitations | Used Free Tier services strategically to stay within budget constraints |

### **Sprint 1 Retrospective**

#### **1. What Went Well (2 Points)**

* **Defined Team Roles:** Clear division of responsibilities improved productivity.
* **Effective GitHub Collaboration:** Use of branches and pull requests reduced conflicts.
* **AWS Deployment Success:** Deployed a cost-effective and scalable architecture on AWS.

#### **2. What Could Be Improved (2 Points)**

* **Automated Testing Coverage:**
  + Suggestion: Implement Jest and Cypress tests for key components.
* **Task Breakdown and Estimation:**
  + Suggestion: Break user stories into smaller tasks for more accurate tracking and progress.

#### **3. Challenges Faced (2 Points)**

* **Time Constraints During Midterms:**
  + Adjusted daily meetings to asynchronous check-ins to maintain communication.
* **Frontend/Backend Mismatches:**
  + Began using Swagger and Google Docs to align API expectations and endpoints.

#### **4. Action Items (4 Points)**

* Introduce automated testing with Jest and Cypress in Sprint 2.
* Add logout functionality and handle edge-case errors in registration and login.
* Improve documentation for API routes using Swagger.
* Schedule integration checkpoints mid-sprint for smoother teamwork.

### **Contribution Summary**

| **Team Member** | **Role(s)** | **Key Contributions** |
| --- | --- | --- |
| Mohamed Elnafe | Application Designer | UI/UX wireframes, React frontend, API integration |
| Mohamed Salman | Cloud Architect | AWS EC2, S3, network/security setup |
| Robert Freeman | Network Engineer & QA Analyst | VPC setup, IAM roles, testing via Postman/Cypress |
| Andrew Slobodnik | Database Architect | Designed normalized MySQL schema, maintained form flow, integrated front/back end |