

AWS-hosted Virtual Classroom and Learning Platform

Prepared For

Smart-Internz
Cloud Practitioner

By

Shubham Dhanaji Patil

D Y Patil Agriculture and Technical University, Talsande

On

26 June 2025

Project Title:

AWS-hosted Virtual Classroom and Learning Platform

Category:

Cloud Deployment | Web Application | AWS Cloud Practitioner

Skills Utilized:

- Core Python
- Flask Framework
- AWS EC2, S3, RDS
- MySQL
- HTML, CSS, JavaScript
- Git & GitHub

1. Project Overview:

In the modern era of digital learning, building a secure, scalable, and flexible virtual classroom is crucial. This project demonstrates the integration of Flask and multiple AWS services to develop a cloud-native educational platform.

Deployed on AWS EC2, the application leverages Amazon S3 for content storage and Amazon RDS (MySQL) for user and application data. Users can register, log in, and access course materials hosted on the cloud — making the platform both effective and expandable for future needs.

2. Key Features:

- Scalable Infrastructure using EC2
- Secure File Storage using S3
- User Management via RDS (MySQL)
- Responsive Web UI for students and instructors
- Seamless Cloud Integration with GitHub

3. Architecture:

```
project/
├─ app.py
├─ templates/
│   ├─ home.html
│   ├─ register.html
│   ├─ login.html
│   └─ content.html
└─ static/ # If needed for CSS, JS, or images
```

4. Final Project Flow:

4.1 Create an AWS Account:

- Sign up and verify your account.
- Explore the AWS Management Console.

4.2 Create an S3 Bucket and Upload Data:

- Create a bucket (e.g., aws-classroom-content).
- Upload files (PDFs, videos).
- Set proper permissions (public or signed URLs).

4.3 Create an RDS Instance (MySQL):

- Launch RDS with MySQL engine.
- Configure DB instance and create a database.
- Connect using MySQL Workbench to create tables.

4.4 Launch and Configure EC2 Instance:

- Launch instance with Amazon Linux 2 or Ubuntu.
- Set security groups and SSH key pair.
- Install Python, Flask, MySQL client.

4.5 Develop Flask App:

- Build routes for register, login, content.
- Create templates: `home.html`, `register.html`, `login.html`, `content.html`.
- Use Bootstrap for styling.
- Connect app to AWS S3 (using `boto3`) and RDS.

4.6 Deploy Flask App on EC2:

- SSH into EC2.
- Clone GitHub repository.
- Install dependencies: `pip install -r requirements.txt`.
- Run app using Gunicorn + Nginx (optional).

4.7 Upload Code to GitHub:

- Create repository.
- Push project files with commits and documentation.

4.8 Test Scenarios:

- **Scenario 1: Student Registration and Login**
- **Scenario 2: Admin Upload of Course Materials**

- **Scenario 3: Downloading Course Content**

5. User Scenarios:

Scenario 1: Student Registration and Course Access

- **User:** Alice Johnson
- **Process:** Registers via form, logs in, and accesses course content from S3.

Scenario 2: Admin Uploads Course Materials

- **User:** System Admin
- **Process:** Uploads PDFs; content is stored in S3 and metadata is updated in RDS.

Scenario 3: Student Downloading Course Content

- **User:** Bob Patel
- **Process:** Selects a file, clicks a link, and downloads directly from S3.

6. Challenges Faced:

- Learning AWS services and IAM policies
- Managing AWS credentials securely
- Flask and AWS integration (using `boto3`)
- RDS connection issues and MySQL Workbench setup
- Debugging EC2 deployment issues

7. Output Pages:

- Landing page with navigation

The landing page features a green header with the site name 'Virtual Classroom' and navigation links: Home, Courses, Login, and Register. The main content area has a light green background with the heading 'Welcome to Virtual Classroom' and the tagline 'Transform your learning experience with our interactive platform'. A green 'Get Started' button is centered below the text. Below this, there are three white boxes with green titles: 'Live Classes' (Join interactive live sessions with expert instructors), 'Rich Content' (Access high-quality learning materials and resources), and 'Community' (Connect with peers and collaborate on projects). The footer is dark gray with the copyright notice '© 2025 Virtual Classroom. All rights reserved.'

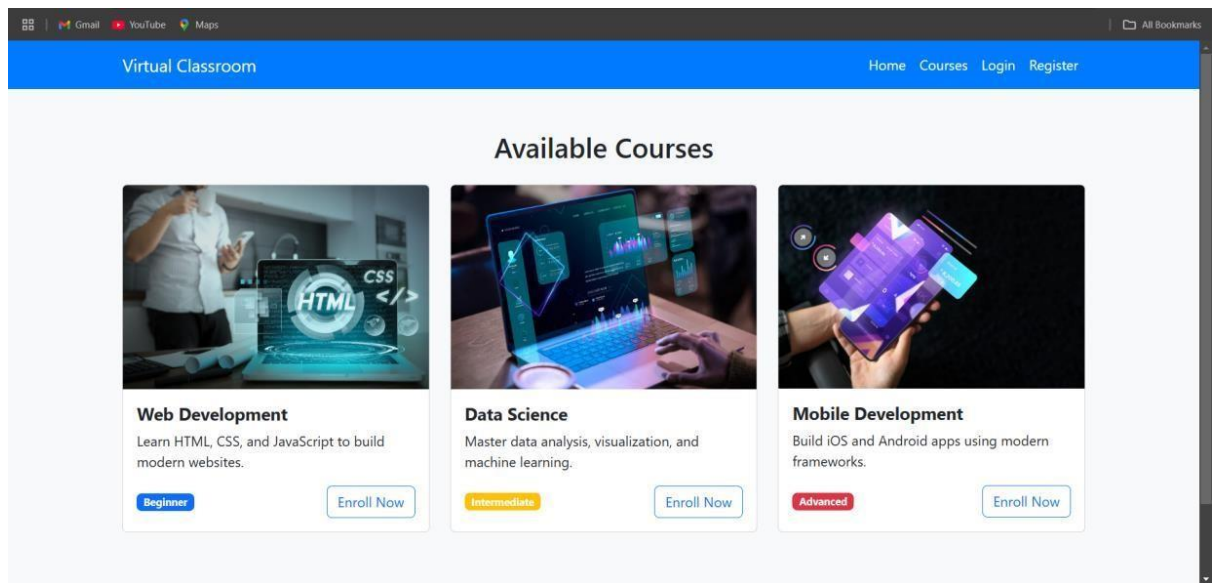
- Registration form

The registration form is titled 'Create Your Account' and is set against a light gray background. It includes input fields for 'First Name' (filled with 'jhon'), 'Last Name' (filled with 'doe'), 'Email Address' (filled with 'name@qwe.com'), 'Password' (masked with dots), and 'Confirm Password' (masked with dots). A checkbox labeled 'I agree to the Terms and Conditions' is checked. A green 'Register' button is at the bottom of the form, with a link 'Already have an account? Login here' below it. The header and footer are consistent with the landing page.

- Login page

The login page is titled 'Login to Your Account' and features a white form box on a light gray background. It contains input fields for 'Email Address' (filled with 'jhon@2003.com') and 'Password' (masked with dots). A 'Remember me' checkbox is checked. A green 'Login' button is at the bottom of the form, with a link 'Don't have an account? Register here' below it. The header and footer are consistent with the other pages.

- Course materials page with download links from S3



8. Conclusion:

This project highlights the integration of cloud computing with web development to build a fully operational virtual classroom. Leveraging AWS's scalability and Flask's simplicity, the platform achieves reliable user access, secure data handling, and an overall smooth educational experience.

9. GitHub And Demo Link:

Demo Link :

[
<https://drive.google.com/file/d/1SKSIBe1ydjG1anNSPYwPDEzxtx-LTWpP/view?usp=drivesdk>
]

GitHub Link:

[
<https://github.com/marvel008/AWS-hosted-Virtual-Classroom-and-Learning-Platform-main.git>
]