

## **1. Title: Edu Consultancy System**



## **2. Project Statement:**

They provide guidance to administrators and other stakeholders in developing or improving training programs. Consultants may assist with training on the use of education programs, tools, and technologies. They may also be involved in the sales of education products and services and expected to meet sales quotas.

### **Outcomes:**

A Consultant who helps parents/students and organizations with educational planning.

To communicate with Student/parents.

Student/parents can Login/Signup to application.

Student/Parent should be able to take appointment with consultants.

Student/Parent should be able view appointment status.

## **3. Modules to be implemented**

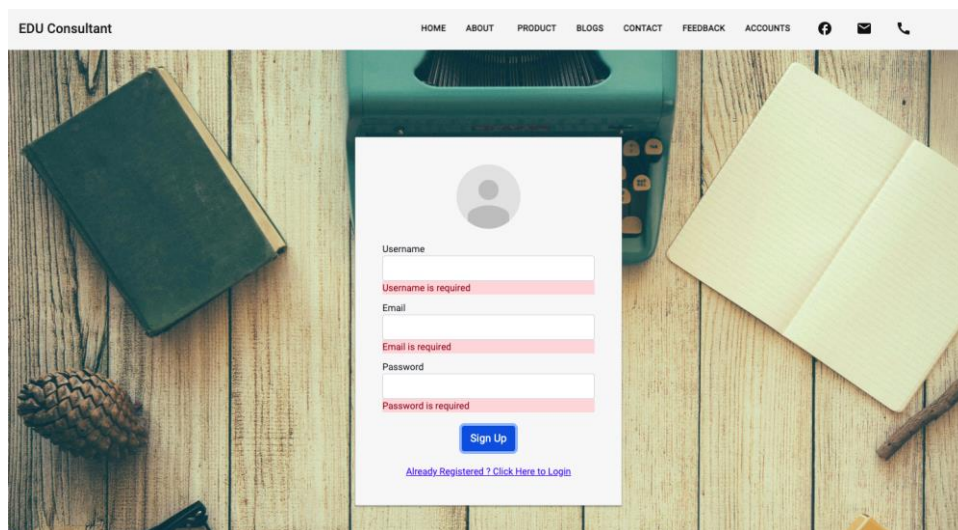
1. Home (Dashboard)
2. User Authentication and Registration includes admin, student, parent forms.
3. The product
4. About US
5. Contact US
6. Feedback

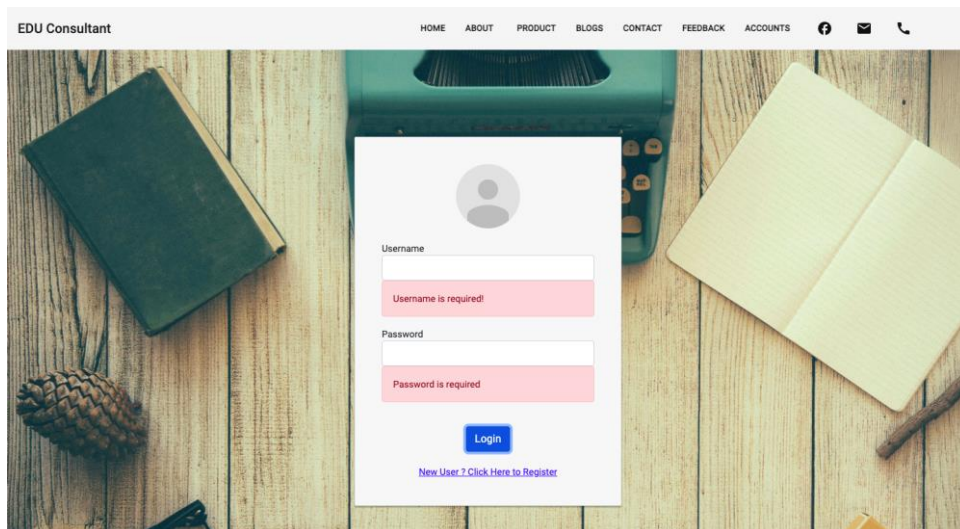
#### 4. Week-wise module implementation and high-level requirements with output screenshots

##### Milestone 1: Weeks 1-3

##### Module 1: User Authentication and Registration

- Implement user registration functionality.
- Develop user login mechanism.
- Spring security mechanism for redirecting to the respective dashboard. Ex: admin, student, and parent dashboards separately.
- After login successfully display the profile section with picture.
- Integrate email verification and password reset features.
- Output screenshot:





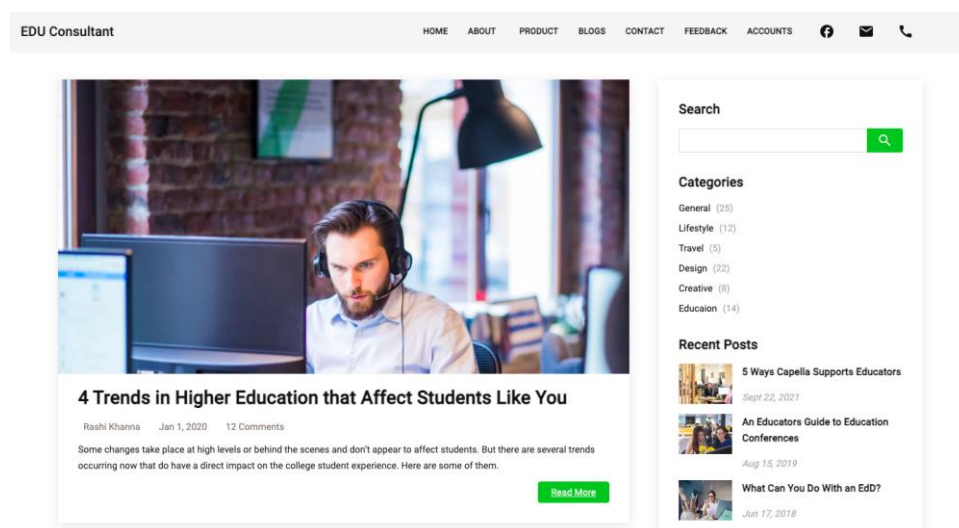
## Module 2: Contact and Role Management

- Design database schema for storing User information.
- Develop an interface for users to set their details.
- Implement backend functionality to manage and update user information.
- Write test cases using JUnit and Mockito for all the Services with code coverage.
- Implement respective module frontend using Angular/React JS and write test cases for frontend logic.
- Form validations must be implemented for all the services.
- Output screenshot

## Milestone 2: Weeks 4-5

## Module 3: Product and Blog Interface

- Design and develop the interface for blogs to display.
- Implement functionality for users to select products, read blogs, and find reasons for contact.
- Integrate AI tools with Spring boot to generate content for blogs and post daily blogs to the application.
- After selecting the product(course)it will go to the payment gateway accept all type of payments (Card, UPI, Net banking) and single and EMI options.
- Once student/parent subscribed successfully display the start date and end date in profile section.
- Integrate the scheduling interface with the backend for storing contact details.
- In this module, we need to provide a dashboard for respective roles: Ex-admin, student, and parent role with respective dashboards.
- In admin dashboard displays all services accessible to do the CRUD operations and e can search for specific candidates to check status of the student, and admin can reply with remarks to the specific student.
- Write test cases using JUnit and Mockito for all the Services with code coverage.
- In dashboard display student status report in graphical representation for learning and exam completion status with percentage.
- Implement respective module frontend using Angular/React JS and write test cases for frontend logic.
- Form validations must be implemented for all the services.
- Output screenshot



#### DESIGN & DEVELOPMENT



Java, Microservices, Cloud, Dot Net, Php, Node JS, Angular JS, React JS UI/UX is our strength and our Services, Experience the magic at your touch. [read more](#)

#### LEARNING & DEVELOPMENT



learning and development strategy outlines how an organisation develops its workforce's capabilities, skills and competencies to remain successful. [read more](#)

#### BULK SMS & DIGITAL MARKETING



Improve Marketing Performance with Predictive Analytics and Multi-touch Attribution. Unified data, seamless campaigns, deep customer and more. [read more](#)

### Milestone 3: Weeks 6-7

#### Module 4: Payment Management

- Design database schema for storing payment details.
- Implement backend functionality for validating the payments.
- Develop an interface for users and admin to view and manage subscriptions.
- For subscription renewal automatically send alerts to respective mail or notifications in profile section with alert symbol.
- Write test cases using JUnit and Mockito for all the Services with code coverage.
- Implement respective module frontend using Angular/React JS and write test cases for frontend logic.
- Form validations must be implemented for all the services.
- Output Screenshot:

EDU Consultant

HOME ABOUT PRODUCT BLOGS CONTACT FEEDBACK ACCOUNTS

### Product Payment Form

Product Name:

Name:

Email:

Phone:

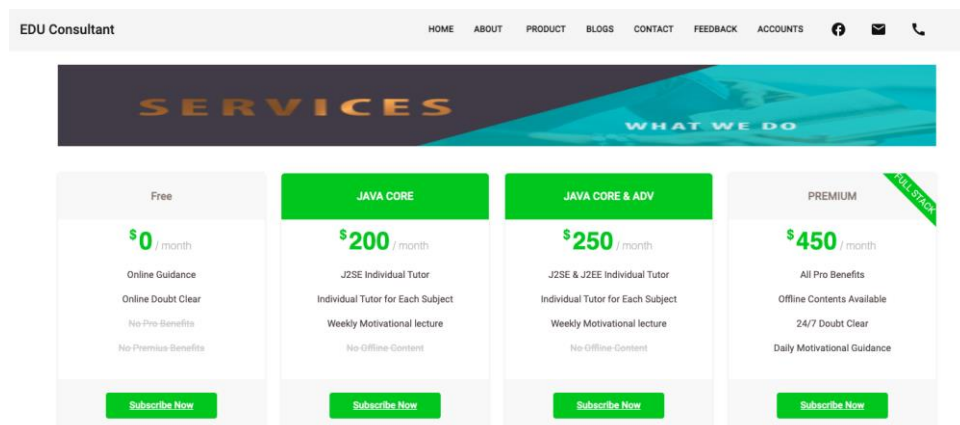
Amount:

[Confirm](#)

## Milestone 4: Weeks 8-10

### Module 5: Product Subscription Management

- Design database schema for storing payment information securely.
- Develop functionality for users to provide and update their subscription information.
- Implement backend logic for accessing and managing subscription information securely.
- User can manage their subscriptions either renewal or cancel the subscription.
- Admin can only manage subscription notifications in this module.
- Admin can remove User data once subscription period is over.
- The subscribed users can watch recorded videos without interruption up to their subscription time.
- Write test cases using JUnit and Mockito for all the Services with code coverage.
- Implement respective module frontend using Angular/React JS and write test cases for frontend logic.
- Form validations must be implemented for all the services.
- Output Screenshot:



### 7. Module 6: Notification System and Feedback

- Integrate email and/or SMS notification services.
- Develop triggers and notifications for subscription reminders, new subscription offers, and cancellations.
- Implement a system for handling and tracking notifications.

- Integrate AI tools with Spring boot to generate automatically message content and send to the respective candidates on their subscription services.
- Implement validations for email and mobile to avoid duplicate entries.
- Write test cases using JUnit and Mockito for all the Services with code coverage.
- Implement respective module frontend using Angular/React JS and write test cases for frontend logic.
- Form validations must be implemented for all the services.
- Output screenshot:

Student Status Updates...


Dear Student,

Successfully subscribed to the requested course FSD Java and

Start Date xx-xx-xx and end Date xx-xx-xx.

Thank you Happy learning...

EDU Consultant
HOME
ABOUT
PRODUCT
BLOGS
CONTACT
FEEDBACK
ACCOUNTS
f
✉
☎



Your Name
 Your Email

Phone Number

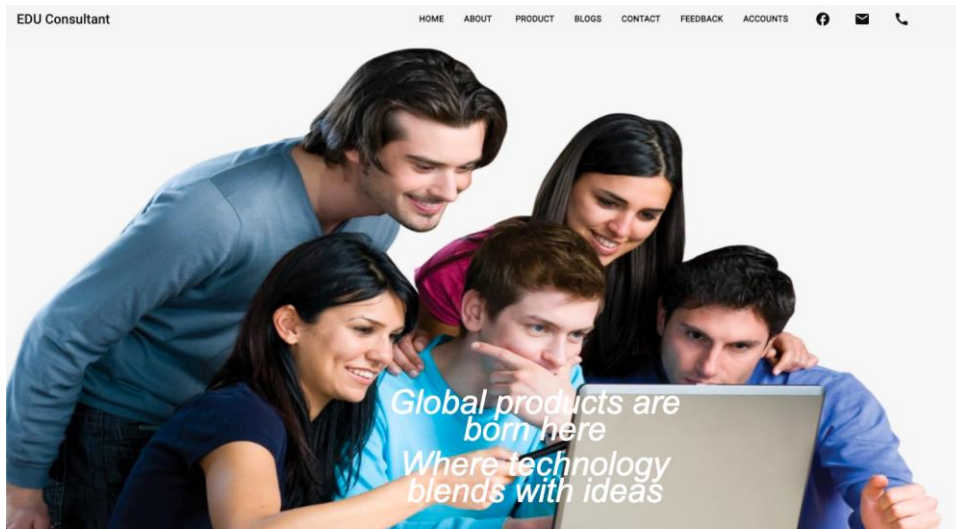
FeedBack

## Module 7: Admin Dashboard

- Design the layout and functionality of the admin dashboard.
- Implement user management features for admins to add, edit, and delete users.
- Develop modules for managing payments, subscriptions, user, and parent information.



- Write test cases using JUnit and Mockito for all the Services with code coverage.
- Implement respective module frontend using Angular/React JS and write test cases for frontend logic.
- Form validations must be implemented for all the services.
- Output Screenshot:



#### **Evaluation Criteria:**

##### **Milestone 1 Evaluation (Week 3):**

- Completion of user authentication and registration.
- Successful implementation of contact functionalities.

##### **Milestone 2 Evaluation (Week 5):**

- The product and blog interface is fully developed and integrated with the backend.

##### **Milestone 3 Evaluation (Week 7):**

- Payment management system is operational, allowing for subscribing and viewing product services or course details.

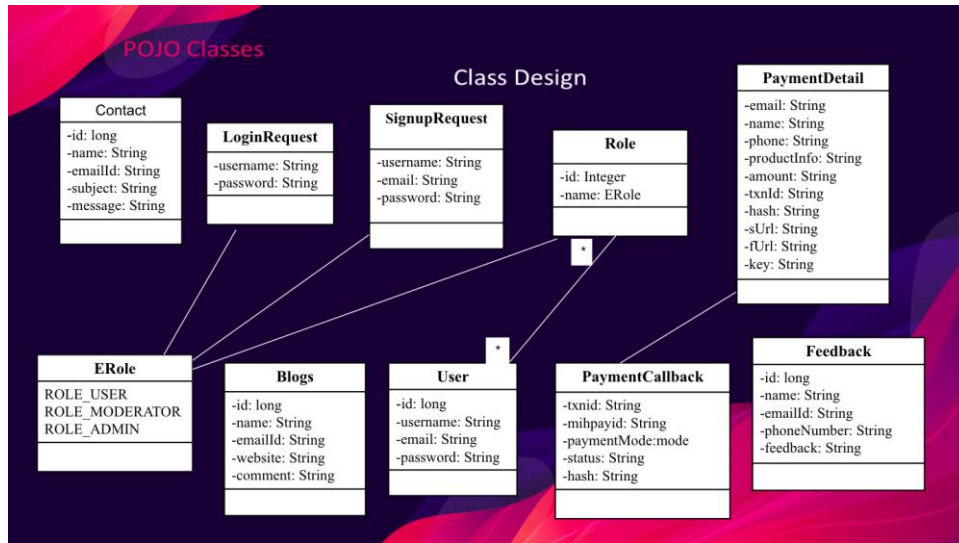
##### **Milestone 4 Evaluation (Week 10):**

- Subscription management, notification system, feedback and admin dashboard are fully implemented and functional.

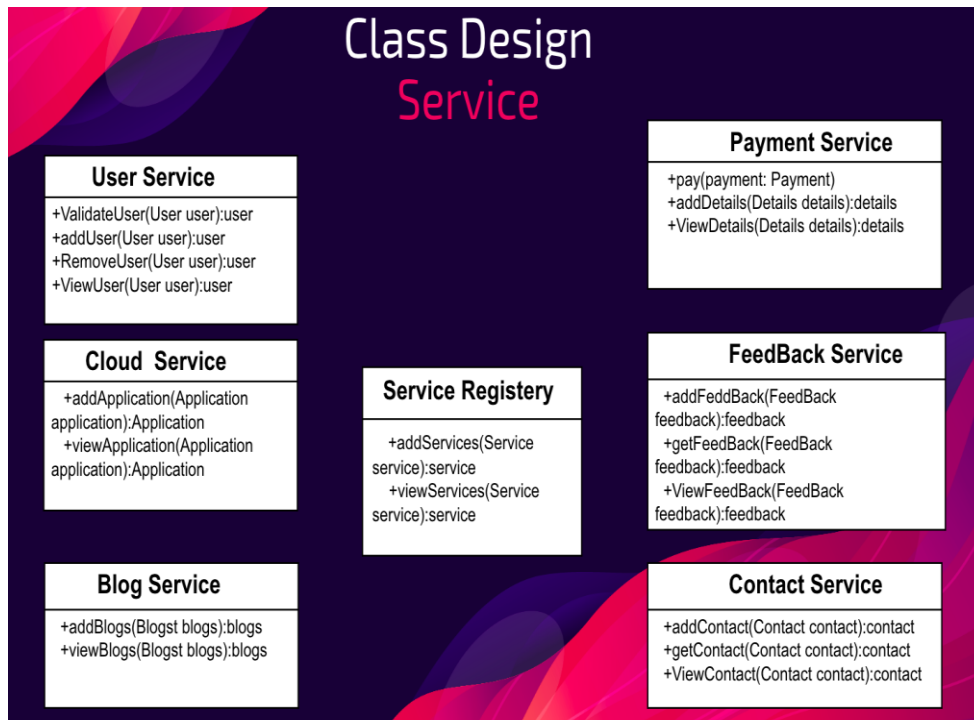


## 5. Design diagrams

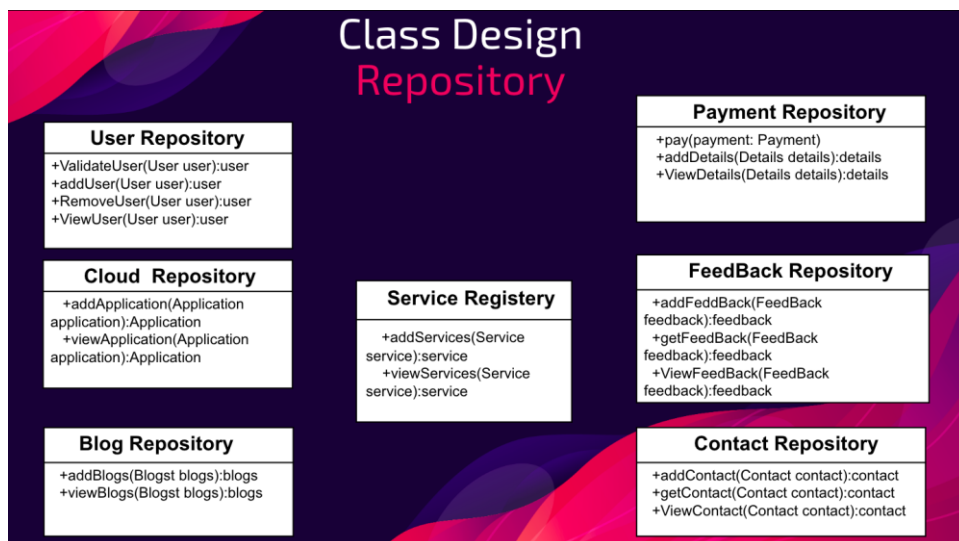
### Class Diagram: Entity classes



### Class Diagram: Service Interface

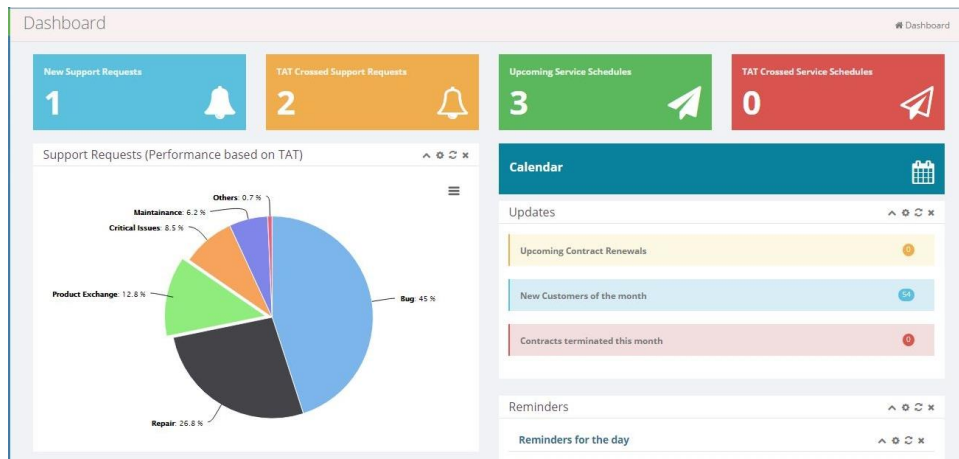


Class Diagram: Repository



### Output Screenshots:

1. Dashboard User:



## 2. Admin Dashboard

