**Task 1: Course Creation**

**Objective**: Implement a feature that allows instructors to create courses.

**Requirements**:

1. Create a form to input course details like title, description, and duration.
2. Use Django's built-in admin to manage courses.

**Task 2: User Roles and Authentication**

**Objective**: Implement user roles (Student, Instructor, Admin) and authentication.

**Requirements**:

1. Implement user registration, login, and logout.
2. Differentiate between student, instructor, and admin roles.

**Task 3: Course Scheduling**

**Objective**: Allow instructors to schedule courses.

**Requirements**:

1. Create a calendar interface.
2. Allow instructors to add course sessions to the calendar.

**\*\*\*\*\***

**Set up a Basic Django Project (you can utilize the one you created above)**

1. Create a new Django project and a new app within it.
2. Set up a basic model for courses with fields like title, description, and syllabus.

**Integrate a Large Language Model API**

1. Choose an API for a large language model, such as OpenAI's GPT-3 or GPT-4. You can use mock data or a free tier of the API for development.
2. Write a Python function to call this API and generate text based on a prompt.

**Automated Content Generation Feature**

1. Add a feature in the Django admin panel where instructors can choose to auto-generate content for a course.
2. When selected, the feature should populate the description and syllabus fields of the course model using the large language model.

**User Interface**

1. Create a simple user interface where instructors can trigger this auto-generation feature and see the generated content.
2. Implement a "Regenerate" button to allow instructors to regenerate content if they are not satisfied with the initial output.