

Here's a comprehensive guide to building a Django-based Course Management System with user authentication (login, logout, signup), CRUD operations for courses, and the ability to display lecture details with video content.

---

## □ Project Setup

### 1. Create a Django Project and App

```
django-admin startproject CoursePortal
cd CoursePortal
python manage.py startapp courses
```

### 2. Install Required Packages

```
pip install django
```

### 3. Add the App to `INSTALLED_APPS`

In `CoursePortal/settings.py`, add `'courses'` and `'django.contrib.auth'` to `INSTALLED_APPS`.

### 4. Create the Database

```
python manage.py migrate
```

## Models

In `courses/models.py`, define the models for `Course` and `Lecture`:

```
from django.db import models
from django.contrib.auth.models import User

class Course(models.Model):
    name = models.CharField(max_length=200)
    description = models.TextField()
    fees = models.DecimalField(max_digits=10, decimal_places=2)
    image = models.ImageField(upload_to='courses/images/')
    created_by = models.ForeignKey(User, on_delete=models.CASCADE)
```

```

def __str__(self):
    return self.name

class Lecture(models.Model):
    course = models.ForeignKey(Course, related_name='lectures',
on_delete=models.CASCADE)
    title = models.CharField(max_length=200)
    video_url = models.URLField()
    created_at = models.DateTimeField(auto_now_add=True)

def __str__(self):
    return self.title

```

## Forms

Create forms for user registration and course management in `courses/forms.py`:

```

from django import forms
from django.contrib.auth.forms import UserCreationForm
from .models import Course, Lecture

class SignUpForm(UserCreationForm):
    email = forms.EmailField()

    class Meta:
        model = User
        fields = ['username', 'email', 'password1', 'password2']

class CourseForm(forms.ModelForm):
    class Meta:
        model = Course
        fields = ['name', 'description', 'fees', 'image']

class LectureForm(forms.ModelForm):
    class Meta:
        model = Lecture
        fields = ['title', 'video_url']

```

courses/views.py file code :-

```
from django.shortcuts import render, redirect, get_object_or_404
from django.contrib.auth import login, authenticate, logout
from django.contrib.auth.forms import AuthenticationForm
from .forms import SignUpForm, CourseForm, LectureForm
from .models import Course, Lecture

def signup(request):
    if request.method == 'POST':
        form = SignUpForm(request.POST)
        if form.is_valid():
            user = form.save()
            login(request, user)
            return redirect('course_list')
        else:
            form = SignUpForm()
    return render(request, 'signup.html', {'form': form})

def login_view(request):
    if request.method == 'POST':
        form = AuthenticationForm(request, data=request.POST)
        if form.is_valid():
            user = form.get_user()
            login(request, user)
            return redirect('course_list')
        else:
            form = AuthenticationForm()
    return render(request, 'login.html', {'form': form})

def logout_view(request):
    logout(request)
    return redirect('login')

def course_list(request):
    courses = Course.objects.all()
    return render(request, 'course_list.html', {'courses': courses})

def course_detail(request, pk):
    course = get_object_or_404(Course, pk=pk)
    return render(request, 'course_detail.html', {'course': course})
```

```

def create_course(request):
    if request.method == 'POST':
        form = CourseForm(request.POST, request.FILES)
        if form.is_valid():
            course = form.save(commit=False)
            course.created_by = request.user
            course.save()
            return redirect('course_list')
    else:
        form = CourseForm()
    return render(request, 'create_course.html', {'form': form})

def update_course(request, pk):
    course = get_object_or_404(Course, pk=pk)
    if request.method == 'POST':
        form = CourseForm(request.POST, request.FILES, instance=course)
        if form.is_valid():
            form.save()
            return redirect('course_detail', pk=course.pk)
    else:
        form = CourseForm(instance=course)
    return render(request, 'edit_course.html', {'form': form, 'course': course})

def delete_course(request, pk):
    course = get_object_or_404(Course, pk=pk)
    if request.method == 'POST':
        course.delete()
        return redirect('course_list')
    return render(request, 'delete_course.html', {'course': course})

def create_lecture(request, course_id):
    course = get_object_or_404(Course, id=course_id)
    if request.method == 'POST':
        form = LectureForm(request.POST)
        if form.is_valid():
            lecture = form.save(commit=False)
            lecture.course = course
            lecture.save()
            return redirect('course_detail', pk=course.id)
    else:
        form = LectureForm()
    return render(request, 'create_lecture.html', {'form': form, 'course':
course})

def update_lecture(request, pk):

```

```

lecture = get_object_or_404(Lecture, pk=pk)
if request.method == 'POST':
    form = LectureForm(request.POST, instance=lecture)
    if form.is_valid():
        form.save()
        return redirect('course_detail', pk=lecture.course.pk)
    else:
        form = LectureForm(instance=lecture)
    return render(request, 'edit_lecture.html', {'form': form, 'lecture':
lecture})

def delete_lecture(request, pk):
    lecture = get_object_or_404(Lecture, pk=pk)
    course_id = lecture.course.id
    if request.method == 'POST':
        lecture.delete()
        return redirect('course_detail', pk=course_id)
    return render(request, 'delete_lecture.html', {'lecture': lecture})

```

## Add to `urls.py`

Update `courses/urls.py` to include the new views:

```

from django.urls import path
from . import views

urlpatterns = [
    path('signup/', views.signup, name='signup'),
    path('login/', views.login_view, name='login'),
    path('logout/', views.logout_view, name='logout'),

    path('courses/', views.course_list, name='course_list'),
    path('courses/create/', views.create_course, name='create_course'),
    path('courses/<int:pk>/', views.course_detail, name='course_detail'),
    path('courses/<int:pk>/edit/', views.update_course, name='update_course'),
    path('courses/<int:pk>/delete/', views.delete_course, name='delete_course'),

    path('courses/<int:course_id>/lectures/create/', views.create_lecture,
name='create_lecture'),
    path('lectures/<int:pk>/edit/', views.update_lecture, name='update_lecture'),
    path('lectures/<int:pk>/delete/', views.delete_lecture,
name='delete_lecture'),
]

```

Include these URLs in the main `urls.py`:

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('courses.urls')),
]
```

HTML templates for a Django project. You can enhance their appearance later using CSS frameworks like Bootstrap.

In templates folder of your app folder which is courses:-

#### `signup.html`

```
<!DOCTYPE html>
<html>
<head>
  <title>Sign Up</title>
</head>
<body>
  <h2>Sign Up</h2>
  <form method="post">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Sign Up</button>
  </form>
  <p>Already have an account? <a href="{% url 'login' %}">Login</a></p>
</body>
</html>
```

---

#### `login.html`

```
<!DOCTYPE html>
<html>
```

```

<head>
  <title>Login</title>
</head>
<body>
  <h2>Login</h2>
  <form method="post">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Login</button>
  </form>
  <p>Don't have an account? <a href="{% url 'signup' %}">Sign up</a></p>
</body>
</html>

```

---

### □ course\_list.html

```

<!DOCTYPE html>
<html>
<head>
  <title>Courses</title>
</head>
<body>
  <h1>Course List</h1>
  <a href="{% url 'create_course' %}">Add New Course</a> |
  <a href="{% url 'logout' %}">Logout</a>

  <ul>
    {% for course in courses %}
      <li>
        <a href="{% url 'course_detail' course.pk %}">{{ course.name
}}</a>
      </li>
    {% empty %}
      <li>No courses available.</li>
    {% endfor %}
  </ul>
</body>
</html>

```

---

### □ course\_detail.html

```

<!DOCTYPE html>
<html>
<head>
  <title>{{ course.name }}</title>
</head>
<body>
  <h1>{{ course.name }}</h1>
  
  <p>{{ course.description }}</p>
  <p>Fees: ₹{{ course.fees }}</p>

```

```

<a href="{% url 'update_course' course.pk %}">Edit</a> |
<a href="{% url 'delete_course' course.pk %}">Delete</a> |
<a href="{% url 'create_lecture' course.pk %}">Add Lecture</a> |
<a href="{% url 'course_list' %}">Back to Courses</a>

<h2>Lectures</h2>
{% for lecture in course.lectures.all %}
  <div>
    <h3>{{ lecture.title }}</h3>
    <video width="320" height="240" controls>
      <source src="{{ lecture.video_url }}" type="video/mp4">
      Your browser does not support the video tag.
    </video><br>
    <a href="{% url 'update_lecture' lecture.pk %}">Edit</a> |
    <a href="{% url 'delete_lecture' lecture.pk %}">Delete</a>
  </div>
{% empty %}
  <p>No lectures added.</p>
{% endfor %}
</body>
</html>

```

---

### □ create\_course.html

```

<!DOCTYPE html>
<html>
<head>
  <title>Create Course</title>
</head>
<body>
  <h2>Create Course</h2>
  <form method="post" enctype="multipart/form-data">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Save</button>
  </form>
  <a href="{% url 'course_list' %}">Back to Course List</a>
</body>
</html>

```

---

### □ create\_lecture.html

```

<!DOCTYPE html>
<html>
<head>
  <title>Add Lecture to {{ course.name }}</title>
</head>
<body>
  <h2>Add Lecture to {{ course.name }}</h2>
  <form method="post">

```

```
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Add Lecture</button>
    </form>
    <a href="{% url 'course_detail' course.pk %}">Back to Course</a>
</body>
</html>
```

## edit\_course.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Edit Course</title>
</head>
<body>
    <h2>Edit Course: {{ course.name }}</h2>
    <form method="post" enctype="multipart/form-data">
        {% csrf_token %}
        {{ form.as_p }}
        <button type="submit">Update</button>
    </form>
    <a href="{% url 'course_detail' course.pk %}">Back to Course</a>
</body>
</html>
```

---

## delete\_course.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Delete Course</title>
</head>
<body>
    <h2>Are you sure you want to delete the course "{{ course.name }}"?</h2>
    <form method="post">
        {% csrf_token %}
        <button type="submit">Yes, delete</button>
        <a href="{% url 'course_detail' course.pk %}">Cancel</a>
    </form>
</body>
</html>
```

---

## edit\_lecture.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Edit Lecture</title>
</head>
<body>
  <h2>Edit Lecture: {{ lecture.title }}</h2>
  <form method="post">
    {% csrf_token %}
    {{ form.as_p }}
    <button type="submit">Update Lecture</button>
  </form>
  <a href="{% url 'course_detail' lecture.course.pk %}">Back to Course</a>
</body>
</html>
```

---

#### □ `delete_lecture.html`

```
<!DOCTYPE html>
<html>
<head>
  <title>Delete Lecture</title>
</head>
<body>
  <h2>Are you sure you want to delete the lecture "{{ lecture.title
  }}"?</h2>
  <form method="post">
    {% csrf_token %}
    <button type="submit">Yes, delete</button>
    <a href="{% url 'course_detail' lecture.course.pk %}">Cancel</a>
  </form>
</body>
</html>
```