

### Experiment-10

For students enrolment developed in Module 2, create a generic class view which displays list of students and detail view that displays student details for any selected student in the list.

**Step-1: First, let's create models for Student and Course if you haven't already:**

#### Models.py

class Student(models.Model):

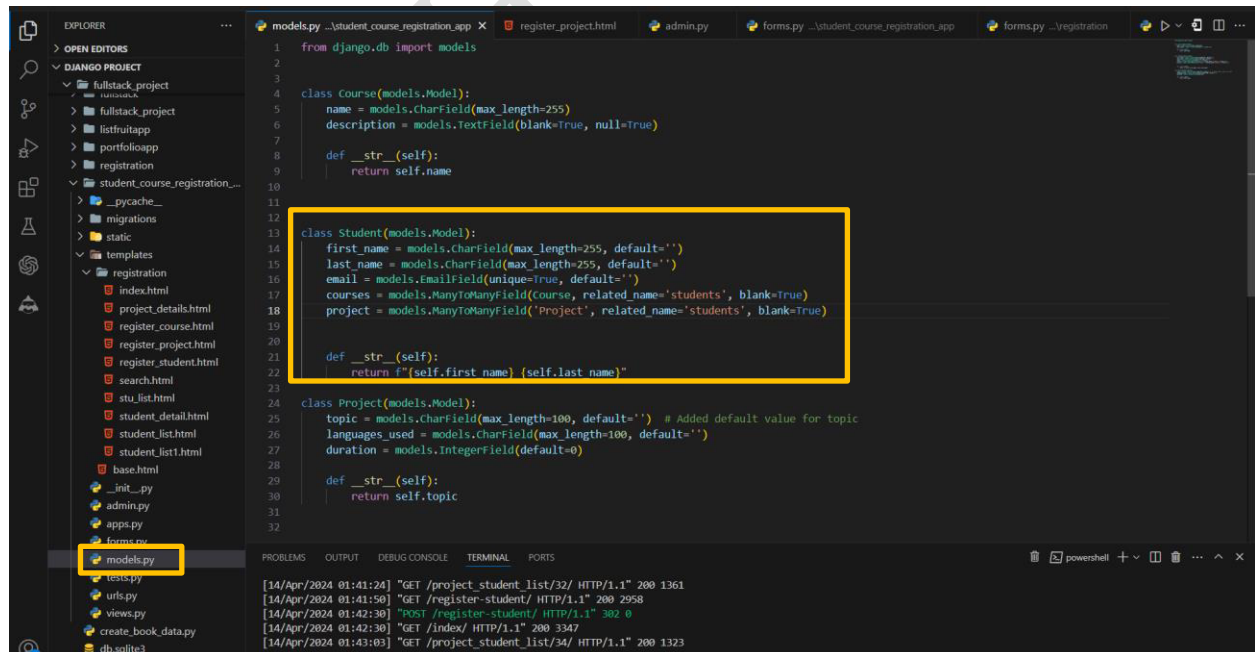
first\_name = models.CharField(max\_length=255, default="")

last\_name = models.CharField(max\_length=255, default="")

email = models.EmailField(unique=True, default="")

courses = models.ManyToManyField(Course, related\_name='students', blank=True)

project = models.ManyToManyField('Project', related\_name='students', blank=True)



Step-2: Next, create views for the list and detail views and import necessary packages:

```
from django.views.generic import ListView, DetailView
```

```
class StudentListView(ListView):
```

```
    model = Student
```

```
    template_name = 'registration/stu_list.html'
```

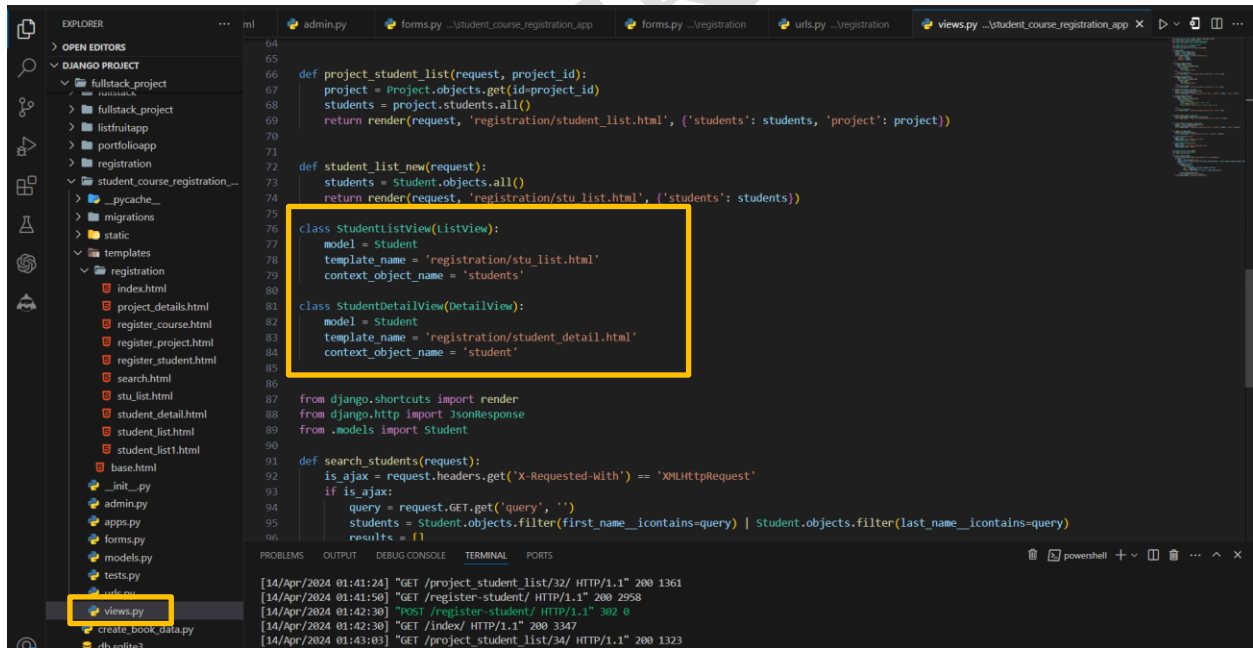
```
    context_object_name = 'students'
```

```
class StudentDetailView(DetailView):
```

```
    model = Student
```

```
    template_name = 'registration/student_detail.html'
```

```
    context_object_name = 'student'
```



- In the StudentListView, we've overridden the get\_queryset method to filter the students based on the course\_id parameter in the URL. If the course\_id is

provided, it will return the students associated with that course; otherwise, it will return all students.

**Step-3: Now, let's create the templates:**

**stu\_list.html**

```
{% extends 'base.html' %}

{% block content %}

<h1>Students</h1>

<ul>

    {% for student in students %}

        <li>

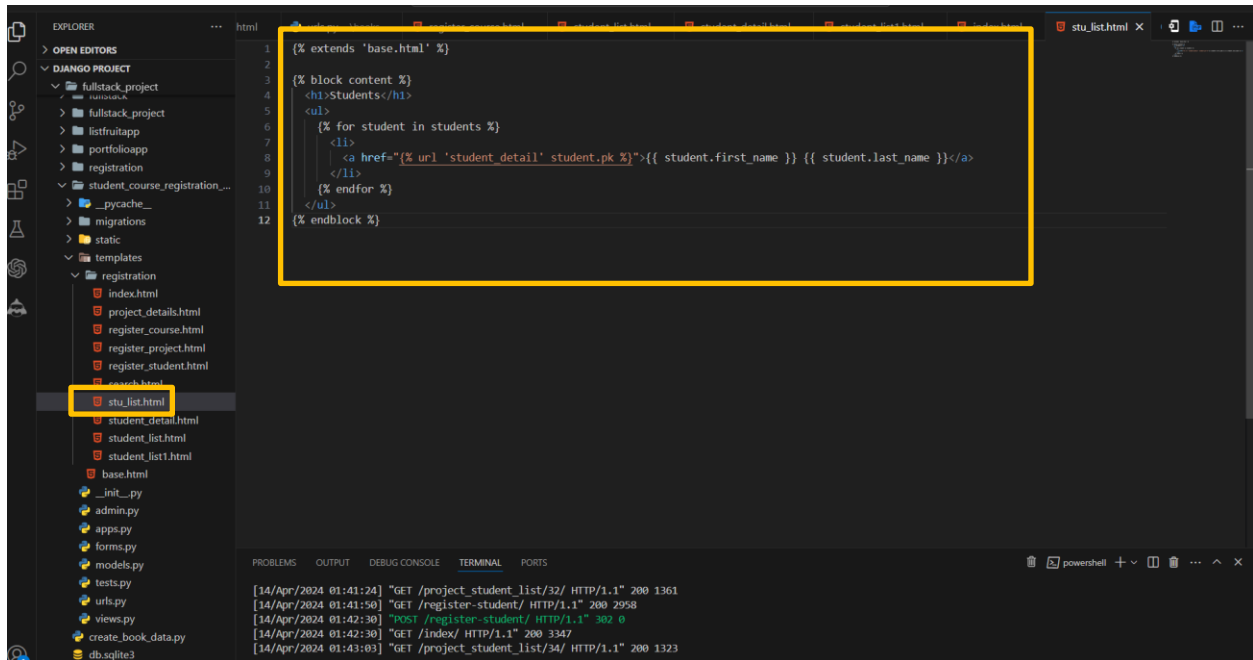
            <a href="{% url 'student_detail' student.pk %}">{{ student.first_name }} {{
student.last_name }}</a>

        </li>

    {% endfor %}

</ul>

{% endblock %}
```



### student\_detail.html

```
{% extends 'base.html' %}

{% block content %}

<div class="container my-5">

  <div class="row">

    <div class="col-md-6 offset-md-3">

      <div class="card">

        <div class="card-body">

          <h1 class="card-title"> Name: {{ student.first_name }} {{ student.last_name }}</h1>

          <h1 class="card-title">Email: {{ student.email }}</h1>

        </div>

      </div>

    </div>

  </div>

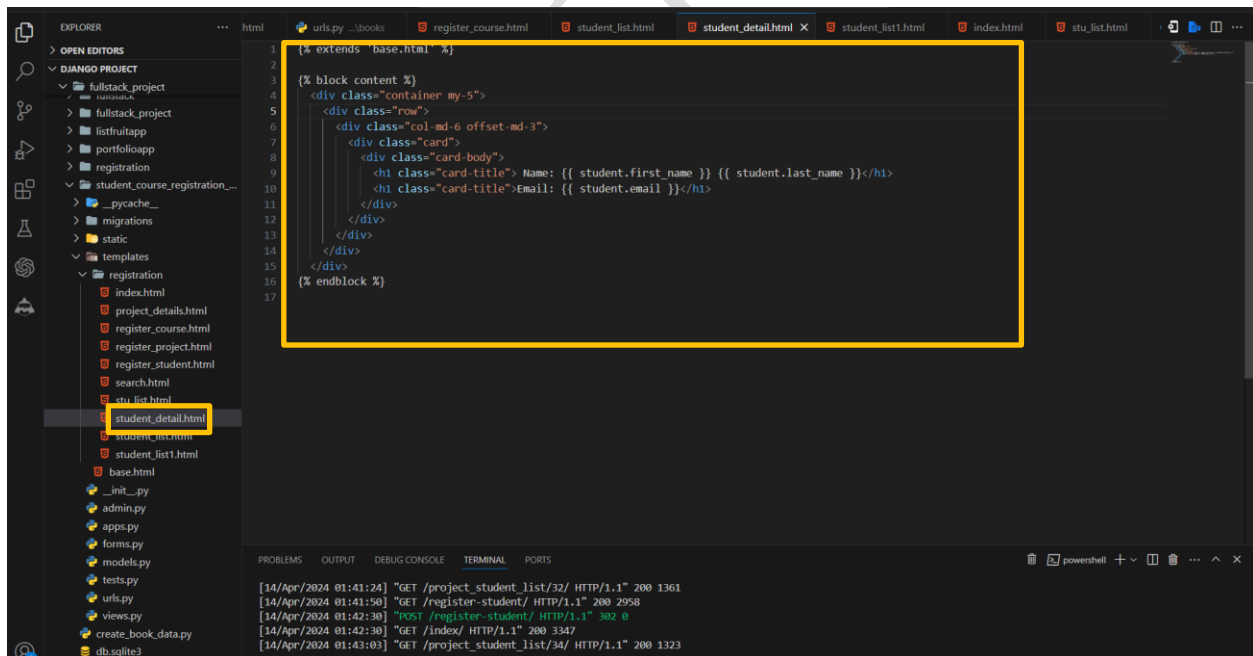
</div>
```

</div>

</div>

</div>

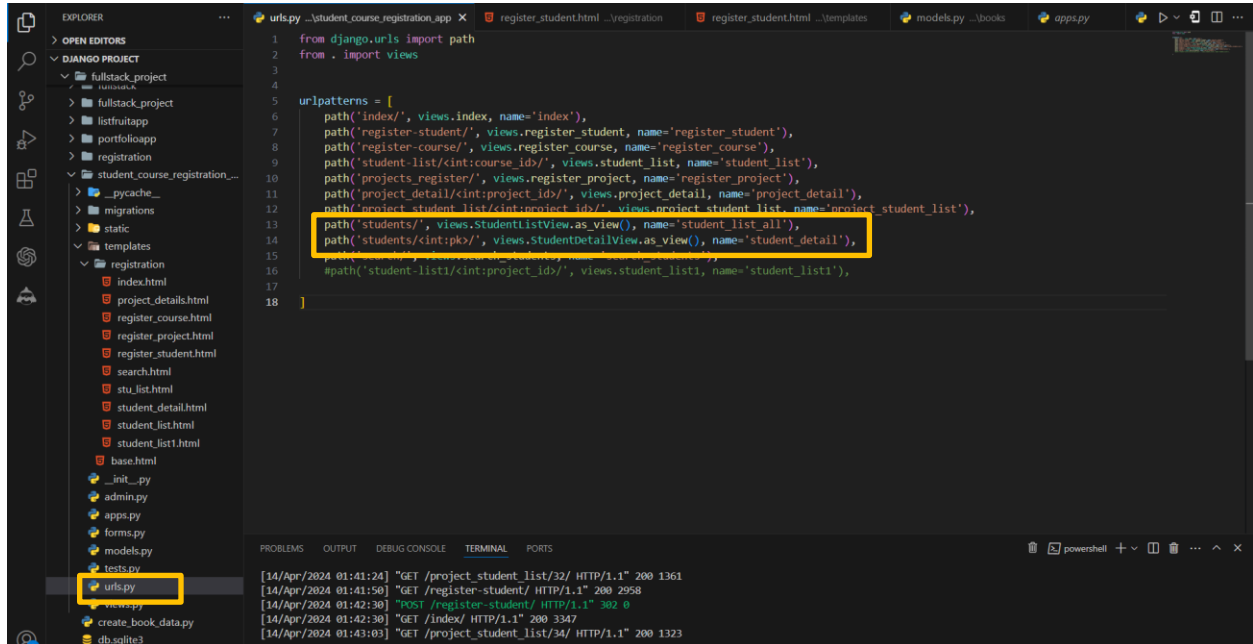
{% endblock %}



### Step-4: add the URL patterns:

```
path('students/', views.StudentListView.as_view(), name='student_list_all'),
```

```
path('students/<int:pk>/', views.StudentDetailView.as_view(), name='student_detail'),
```



### Step-5: add Index.html For showing the Students In Main Page:

#### Index.html

```
<h1 class="mt-5 my-5">Students</h1>
```

```
<ul class="list-group mt-3">
```

```
    {% for student in students %}
```

```
    <li class="list-group-item">
```

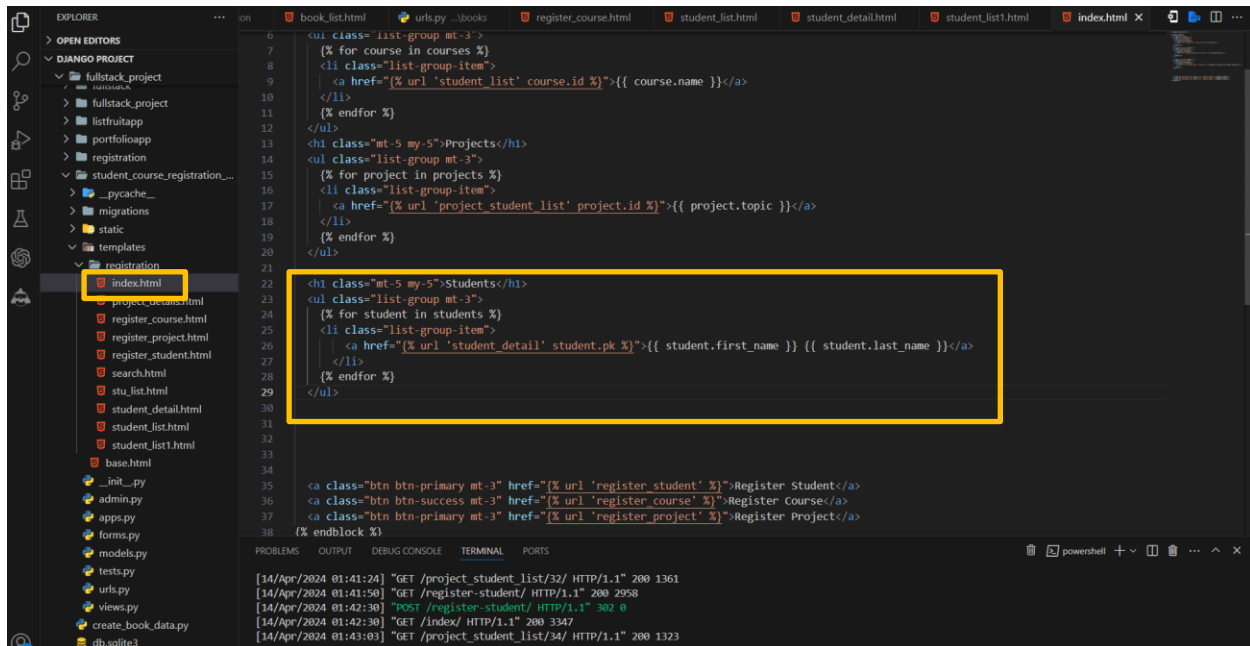
```
        <a href="{% url 'student_detail' student.pk %}">{{ student.first_name }} {{
```

```
student.last_name }}</a>
```

```
    </li>
```

```
    {% endfor %}
```

</ul>



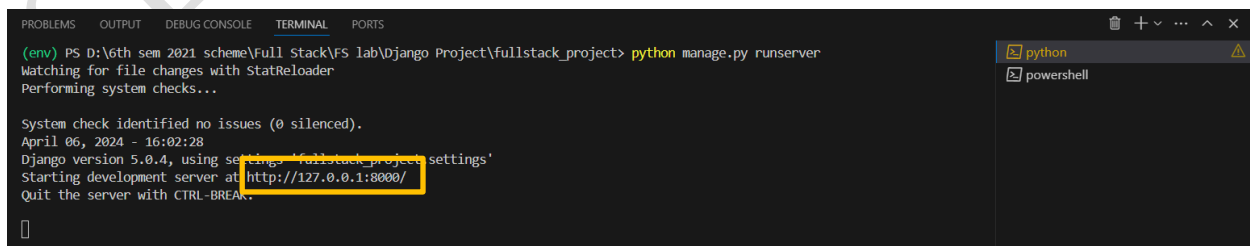
```
6 <ul class="list-group mt-3">
7   {% for course in courses %}
8   <li class="list-group-item">
9     <a href="{% url 'student_list' course.id %}">{{ course.name }}</a>
10  </li>
11  {% endfor %}
12 </ul>
13 <h1 class="mt-5 my-5">Projects</h1>
14 <ul class="list-group mt-3">
15   {% for project in projects %}
16   <li class="list-group-item">
17     <a href="{% url 'project_student_list' project.id %}">{{ project.topic }}</a>
18   </li>
19   {% endfor %}
20 </ul>
21
22 <h1 class="mt-5 my-5">Students</h1>
23 <ul class="list-group mt-3">
24   {% for student in students %}
25   <li class="list-group-item">
26     <a href="{% url 'student_detail' student.pk %}">{{ student.first_name }} {{ student.last_name }}</a>
27   </li>
28   {% endfor %}
29 </ul>
30
31
32
33
34
35 <a class="btn btn-primary mt-3" href="{% url 'register_student' %}">Register Student</a>
36 <a class="btn btn-success mt-3" href="{% url 'register_course' %}">Register Course</a>
37 <a class="btn btn-primary mt-3" href="{% url 'register_project' %}">Register Project</a>
38 {% endblock %}
```

## Step-6: Run the development server

- In the VS Code terminal, navigate to your Django project's directory (the one containing manage.py).
- Run the development server

## python manage.py runserver

- Open your web browser and visit <http://127.0.0.1:8000/>.

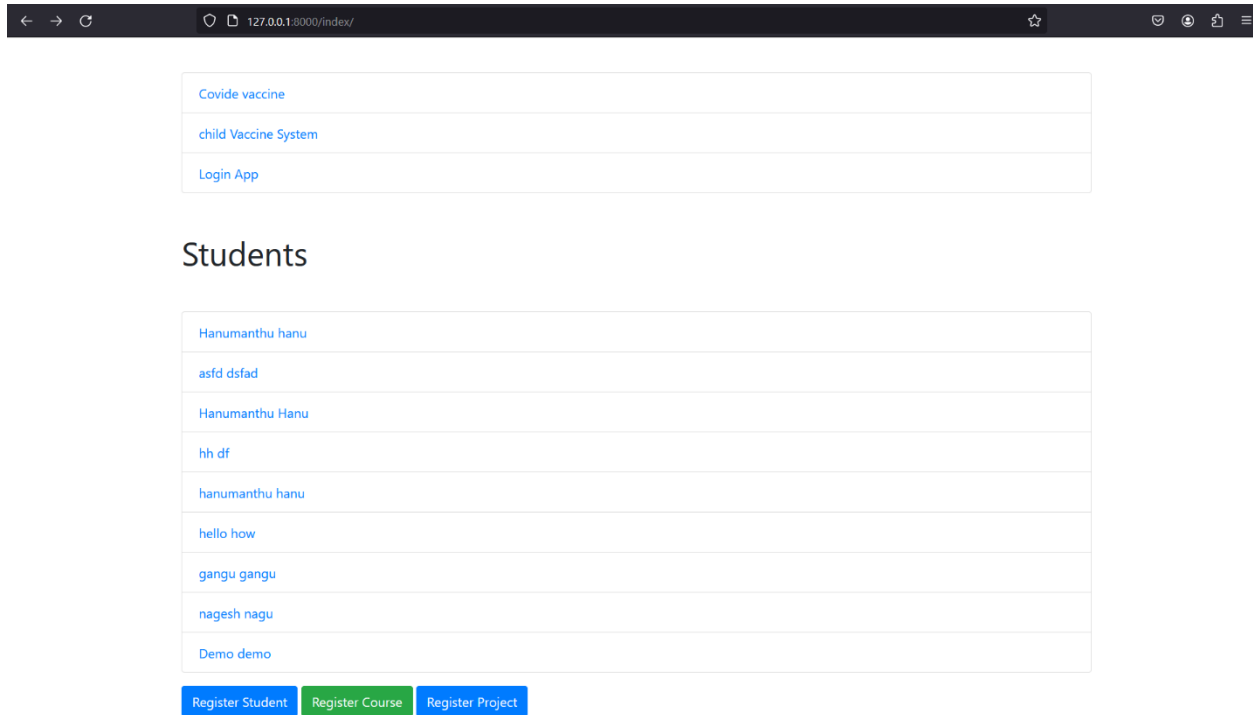


```
(env) PS D:\6th sem 2021 scheme\Full Stack\FS lab\Django Project\fullstack_project> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

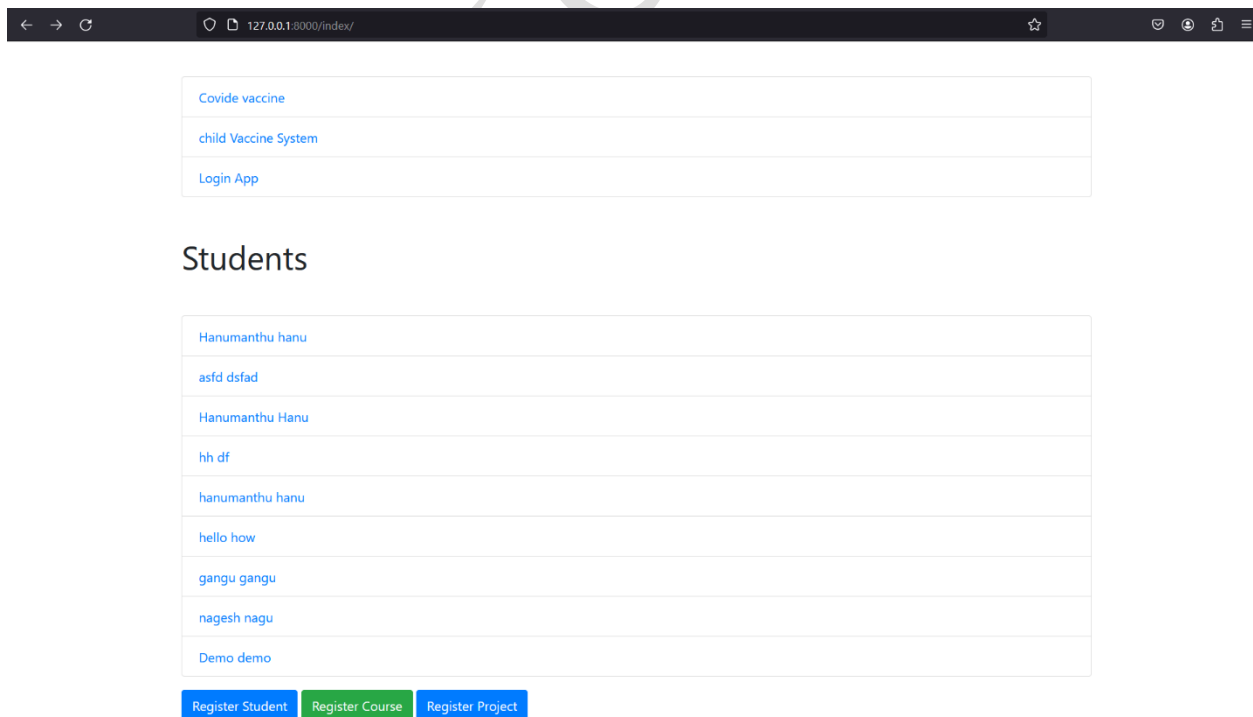
System check identified no issues (0 silenced).
April 06, 2024 - 16:02:28
Django version 5.0.4, using settings 'fullstack_project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

- Type or copy this <http://127.0.0.1:8000/index/>

## Final Output By Clicking Register Project and Register Student



**Fig: Main Screen**



**Fig: List View of Students Screen**





**Fig: Detailed View of Student Screen**