

Educational Platform for Courses, Subjects, and Topics

This project is an educational platform that allows users to navigate through courses, subjects, and topics. It implements both the Structure Guided Browsing and Hypertext Model architectures to enhance user experience and information retrieval.

Table of Contents

- [Overview](#)
- [Features](#)
- [Installation](#)
- [Usage](#)
- [Code Explanation](#)
- [Structure Diagram](#)
- [Contributing](#)
- [License](#)

Overview

This platform is designed to help users navigate through educational content in a structured and intuitive manner. It uses a hierarchical structure to organize courses, subjects, and topics, and provides hyperlinks to related content for easy navigation.

Features

- Structure Guided Browsing: Hierarchical navigation through courses, subjects, and topics.
- Hypertext Model: Hyperlinks between related content.
- Responsive design for various screen sizes.

Installation

1. Clone the repository:

```
git https://github.com/KhadimHussainDev/structure_browsing-and-hypertext-model
```

2. Navigate to the project directory:

```
cd structure_browsing-and-hypertext-model
```

3. Install the required dependencies:

```
pip install -r requirements.txt
```

4. Run the Django development server:

```
python manage.py runserver
```

Usage

1. Open your web browser and navigate to <http://127.0.0.1:8000/>.
2. Browse through the available courses, subjects, and topics.
3. Click on hyperlinks to navigate between related content.

Code Explanation

Models

The models define the structure of the database and the relationships between courses, subjects, and topics.

```
from django.db import models

class Course(models.Model):
    name = models.CharField(max_length=200)
    description = models.TextField()

    def __str__(self):
        return self.name

class Subject(models.Model):
    course = models.ForeignKey(Course, on_delete=models.CASCADE,
related_name="subjects")
    name = models.CharField(max_length=200)
    description = models.TextField()

    def __str__(self):
        return self.name

class Topic(models.Model):
    subject = models.ForeignKey(Subject, on_delete=models.CASCADE,
related_name="topics")
    name = models.CharField(max_length=200)
    content = models.TextField()

    def __str__(self):
        return self.name
```

Templates

The templates define the HTML structure and layout of the web pages. They use Django template tags to dynamically generate content.

course_list.html

Displays a list of available courses.

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Courses</title>
  <link rel="stylesheet" href="{% static 'css/styles.css' %}">
</head>
<body>
  <header>
    <h1>Available Courses</h1>
  </header>
  <main>
    <ul>
      {% for course in courses %}
      <li>
        <a href="{% url 'course_detail' course.id %}">{{ course.name }}
      </a>
        <p>{{ course.description }}</p>
      </li>
      {% endfor %}
    </ul>
  </main>
</body>
</html>
```

course_detail.html

Displays the details of a selected course and its subjects.

```
{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>{{ course.name }}</title>
  <link rel="stylesheet" href="{% static 'css/styles.css' %}">
</head>
<body>
  <header>
    <h1>{{ course.name }}</h1>
    <p>{{ course.description }}</p>
  </header>
  <main>
```

```

        <button onclick="history.back()">Go Back</button>
        <h2>Subjects</h2>
        <ul>
            {% for subject in course.subjects.all %}
            <li>
                <a href="{% url 'subject_detail' subject.id %}">{{ subject.name }}
            </a>
                <p>{{ subject.description }}</p>
            </li>
            {% endfor %}
        </ul>
    </main>
    <footer>
        <a href="{% url 'course_list' %}">Back to Courses</a>
    </footer>
</body>
</html>

```

subject_detail.html

Displays the details of a selected subject and its topics.

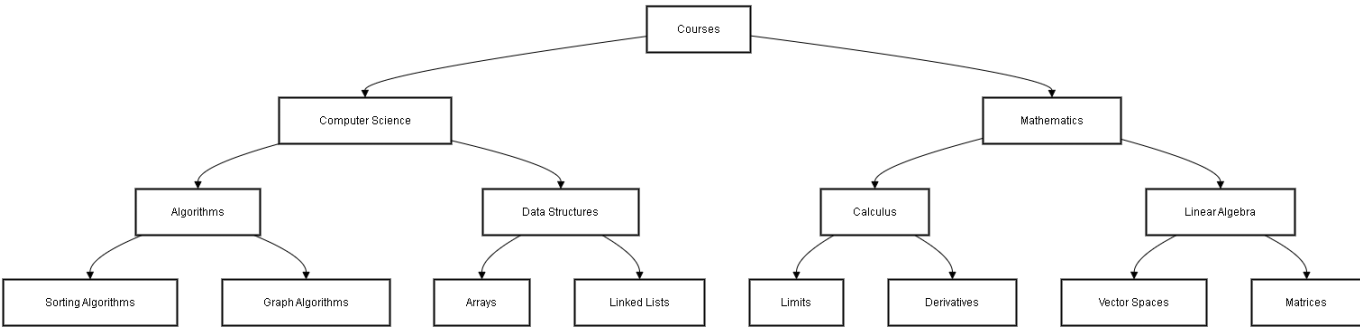
```

{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>{{ subject.name }}</title>
    <link rel="stylesheet" href="{% static 'css/styles.css' %}">
</head>
<body>
    <header>
        <h1>{{ subject.name }}</h1>
        <p>{{ subject.description }}</p>
    </header>
    <main>
        <button onclick="history.back()">Go Back</button>
        <h2>Topics</h2>
        <ul>
            {% for topic in subject.topics.all %}
            <li>
                <h3>{{ topic.name }}</h3>
                <p>{{ topic.content }}</p>
            </li>
            {% endfor %}
        </ul>
    </main>
    <footer>
        <a href="{% url 'course_detail' subject.course.id %}">Back to {{
subject.course.name }}</a>
    </footer>

```

```
</footer>
</body>
</html>
```

Structure Diagram



Contributing

Contributions are welcome! Please open an issue or submit a pull request for any improvements or bug fixes.

License

This project is licensed under the MIT License. See the [LICENSE](#) file for details.