TY B.Tech

Advanced Database System Lab.

Assignment No. 4

Batch: T6

Roll No: 2019BTECS00033

Title:

Design and implement a web-enabled student MIS (Management Information System).

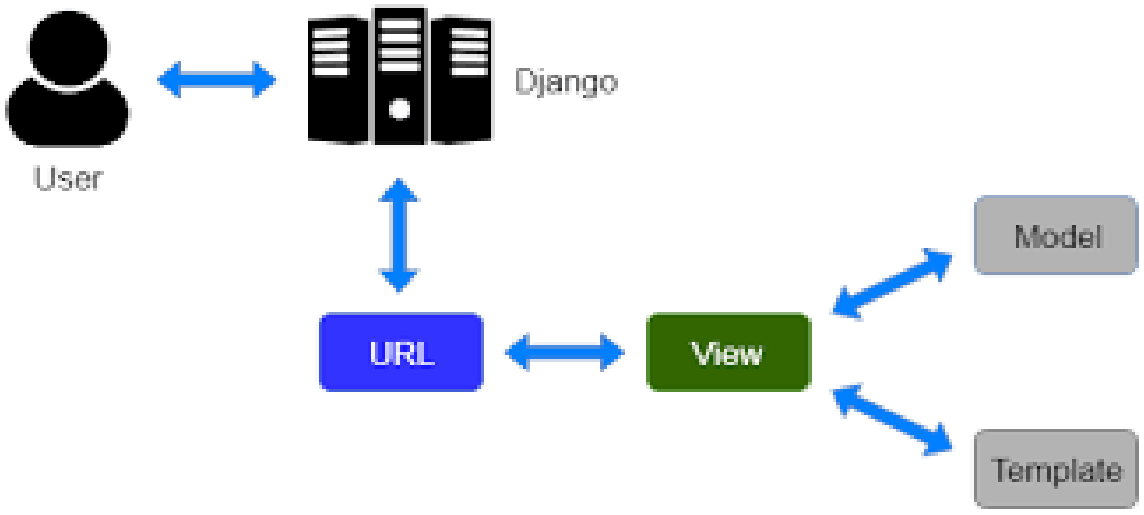
Objective: Use Django/ MySQL / Bootstrap v5.0.

Introduction:

Django:

Django is a python library which is popular for its entire development .

This is how Django MVT model works .



MySQL:

MySQL : Here Django provides inbuilt db.SQLite for its database .which has admin panel to visualize entire db in more efficient way also we have create a superuser for that .

Bootstrap v5.0:

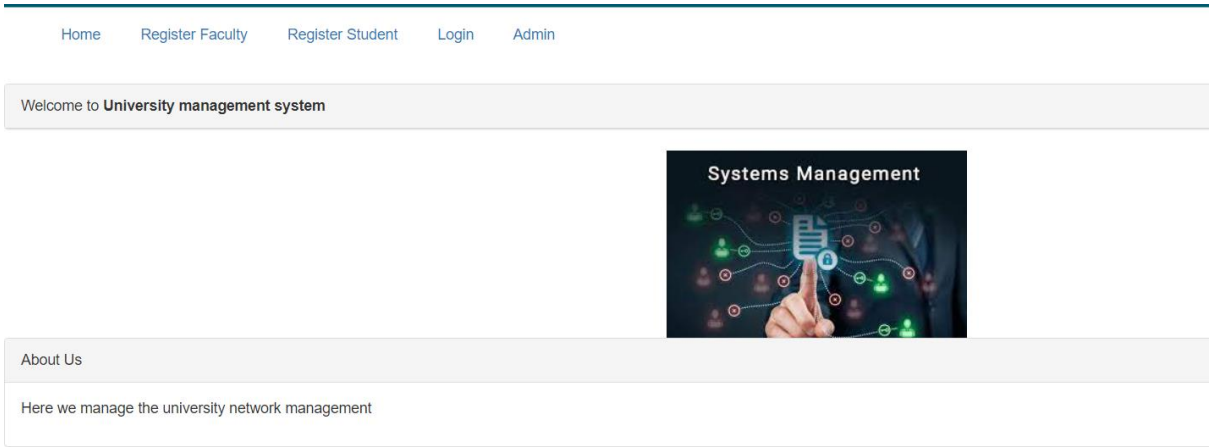
Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and (optionally) JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Adding bootstrap to application:

```
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO" crossorigin="anonymous">
```

Above script is added in <head> </head> of the HTML page.

Home page:



Entity Boards :

Home Register Faculty Register Student Login Admin

Register Faculty

Name

Enter Your Name

Username

Enter Your Username

Email

Enter Your Email

Password

Enter Your Password

Submit

Home Register Faculty Register Student Login Admin

Register Student

Name

Enter Your Name

Username

Enter Your Username

Email

Enter Your Email

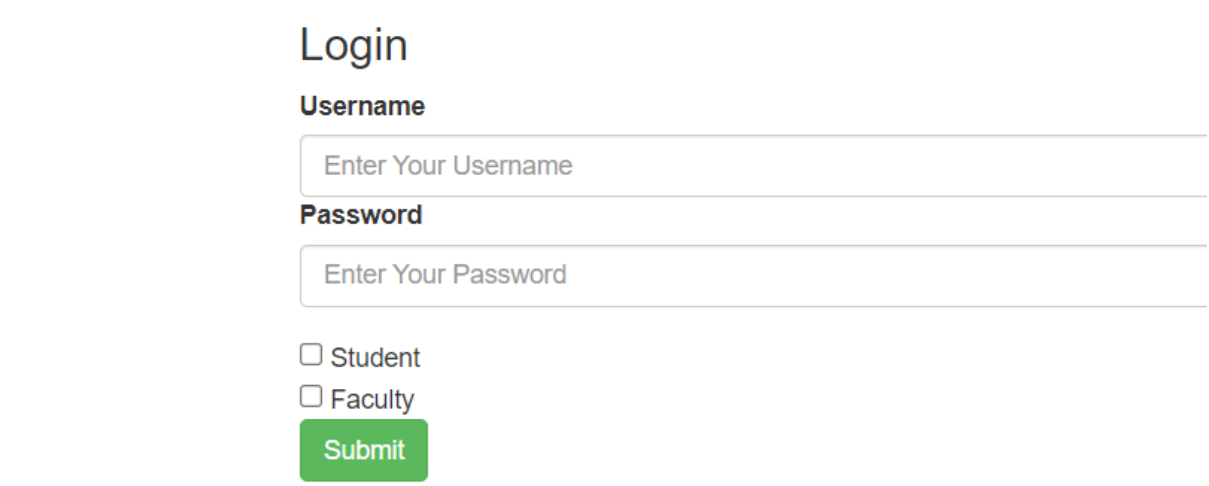
Password

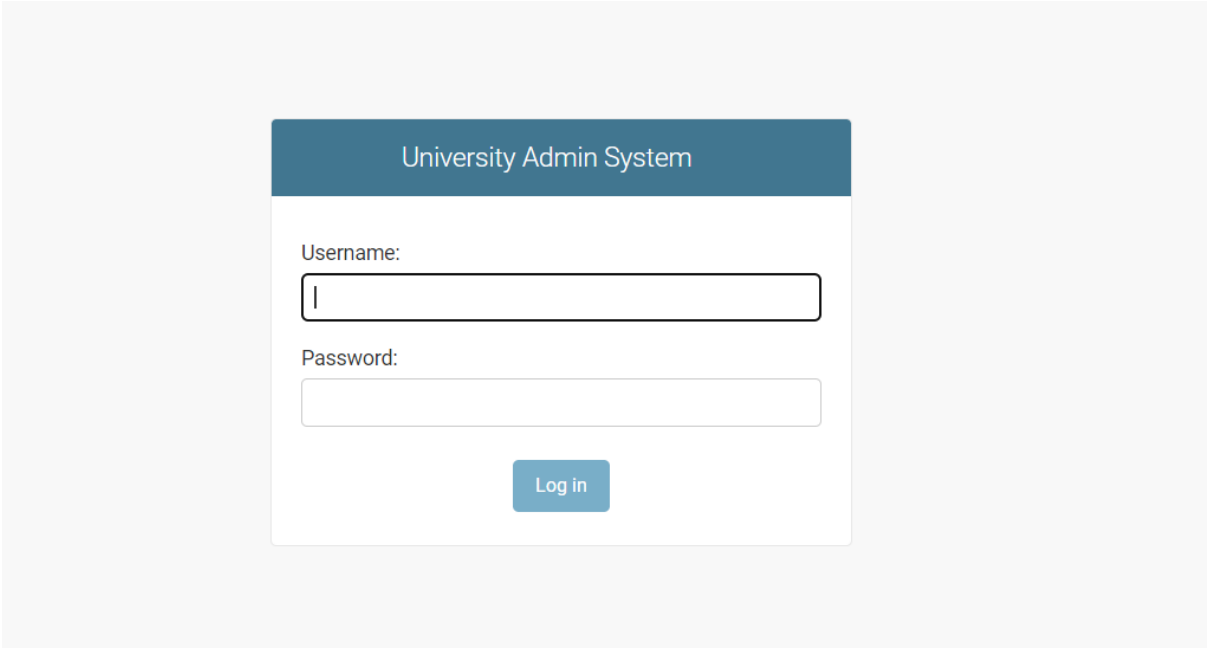
Enter Your Password

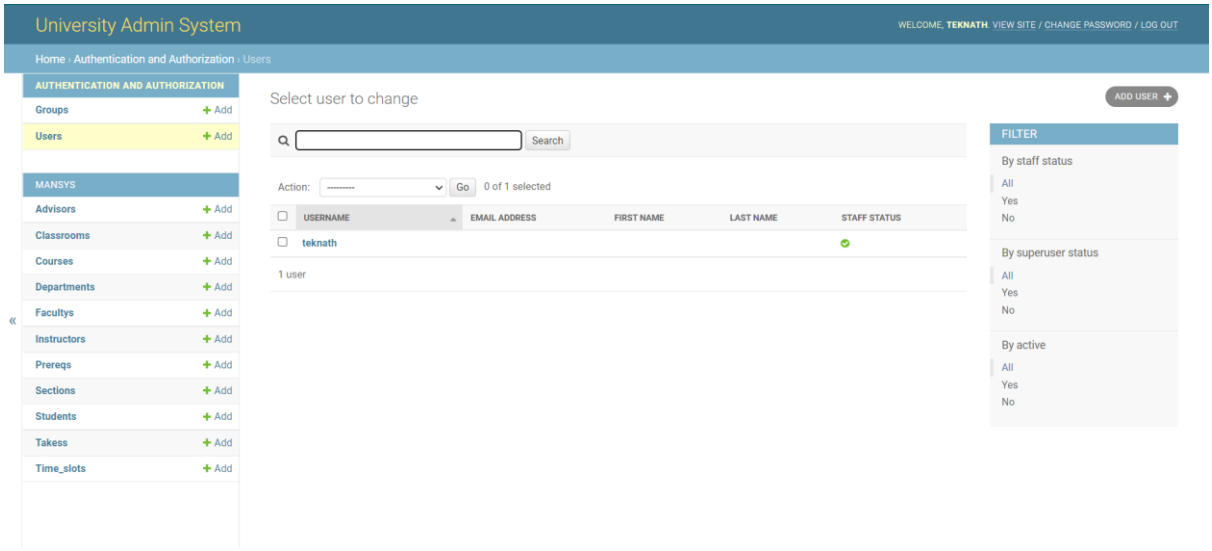
Roll No

Enter Your Roll No.

Submit


Login
Username
Enter Your Username
Password
Enter Your Password
Student
Faculty
Submit

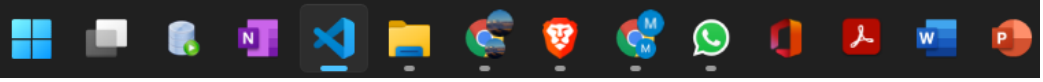

University Admin System
Username:
Password:
Log in




```python
# Create your models here.
class Faculty(models.Model):

    name=models.CharField(max_length=255,default='')
    # id=models.CharField(max_length=255,default='')
    # course_id=models.CharField(max_length=255,default='')
    # sec_id=models.CharField(max_length=255,default='')
    # semester=models.CharField(max_length=255,default='')
    # year=models.CharField(max_length=255,default='')
    password=models.CharField(max_length=255,default='')
    email=models.CharField(max_length=255,default='')
    username=models.CharField(max_length=255,default='')

    def __str__(self):
        return f'{self.name}'



class Student(models.Model):

    name=models.CharField(max_length=255,default='')
    email=models.CharField(max_length=255,default='')
    username=models.CharField(max_length=255,default='')
    password=models.CharField(max_length=255,default='')
    rollNo=models.CharField(max_length=50,default='')


    def __str__(self):
        return f'{self.name} {self.rollNo}'
```

```python
class Classroom(models.Model):

    building=models.CharField(max_length=255,default='')
    room_number=models.CharField(max_length=255,default='')
    capacity=models.IntegerField()


    def __str__(self):
        return f'{self.building} {self.room_number}'

class Department(models.Model):

    dept_name=models.CharField(max_length=255,default='')
    room_number=models.CharField(max_length=255,default='')
    capacity=models.IntegerField()

    def __str__(self):
        return f'{self.dept_name} {self.room_number}'


class Course(models.Model):

    course_id=models.CharField(max_length=255,default='')
    title=models.CharField(max_length=255,default='')
    dept_name=models.CharField(max_length=255,default='')
    credits=models.IntegerField()
```

```python
class Instructor(models.Model):

    ins_id=models.CharField(max_length=255,default='')
    name=models.CharField(max_length=255,default='')
    dept_name=models.CharField(max_length=255,default='')
    salary=models.CharField(max_length=255,default='')

    def __str__(self):
        return f'{self.name} {self.rollNo}'

class Section(models.Model):

    course_id=models.CharField(max_length=255,default='')
    sec_id=models.CharField(max_length=255,default='')
    semester=models.CharField(max_length=255,default='')
    year=models.CharField(max_length=255,default='')
    building=models.CharField(max_length=255,default='')
    room_number=models.CharField(max_length=255,default='')
    time_slot_id=models.CharField(max_length=255,default='')


    def __str__(self):
        return f'{self.course_id} {self.sec_id}'

class Takes(models.Model):

    takes_id=models.CharField(max_length=255,default='')
    course_id=models.CharField(max_length=255,default='')
    sec_id=models.CharField(max_length=255,default='')
    semester=models.CharField(max_length=255,default='')
    year=models.CharField(max_length=255,default='')
    grade=models.CharField(max_length=255,default='')
```



```python
class Advisor(models.Model):

    s_id=models.CharField(max_length=255,default='')
    i_id=models.CharField(max_length=255,default='')

    def __str__(self):
        return f'{self.s_id} {self.i_id}'

class Time_slot(models.Model):

    time_slot_id=models.CharField(max_length=255,default='')
    day=models.CharField(max_length=255,default='')
    start_time=models.CharField(max_length=255,default='')
    year=models.CharField(max_length=255,default='')
    grade=models.CharField(max_length=255,default='')


    def __str__(self):
        return f'{self.time_slot_id} {self.day}'

class Prereq(models.Model):

    course_id=models.CharField(max_length=255,default='')
    prereq_id=models.CharField(max_length=255,default='')

    def __str__(self):
        return f'{self.course_id} {self.prereq_id}'
```

Conclusion:

Django application was developed for the Student Information Management System with the help of Bootstrap5 as a front-end framework and SQLITE as backend database.