

Models .py

```
from django.db import models
from django.contrib.auth import get_user_model
from django.utils.text import Truncator

# Create your models here.
User=get_user_model()

class Tags(models.Model):

    name=models.CharField(max_length=100)
    description=models.CharField(max_length=100)

    def __str__(self):
        return self.name

class Question(models.Model):
    title=models.CharField(max_length=100)
    description=models.CharField(max_length=200)
    asked_by=models.ForeignKey(User,on_delete=models.DO_NOTHING)
    asked_at=models.DateTimeField(auto_now_add=True)
    tags=models.ManyToManyField(Tags)
    image1=models.ImageField(upload_to='questions', null=True)
    image2=models.ImageField(upload_to='questions', null=True)

    def __str__(self):
        return self.title

class Solutions(models.Model):
    question=models.ForeignKey(Question,on_delete=models.CASCADE)
    solution=models.CharField(max_length=200)
    upvote=models.IntegerField()
    answerd_at=models.DateTimeField(auto_now_add=True)
    answered_by=models.ForeignKey(User,on_delete=models.CASCADE)
    image1=models.ImageField(upload_to='solutions', null=True)
    image2=models.ImageField(upload_to='solutions', null=True)

    def __str__(self):
        x=Truncator(self.solution).chars(15)
        return x
```

```
class User_details(models.Model):
    email = models.CharField(max_length=100)
    profile_image = models.ImageField(upload_to='profile', null=True)
    first_name = models.CharField(max_length=200)
    last_name = models.CharField(max_length=200)
    created_at = models.DateTimeField(auto_now_add=True)
    watched_tags=models.ManyToManyField(Tags)
    user=models.ForeignKey(User,on_delete=models.CASCADE)

    def __str__(self):
        return self.first_name

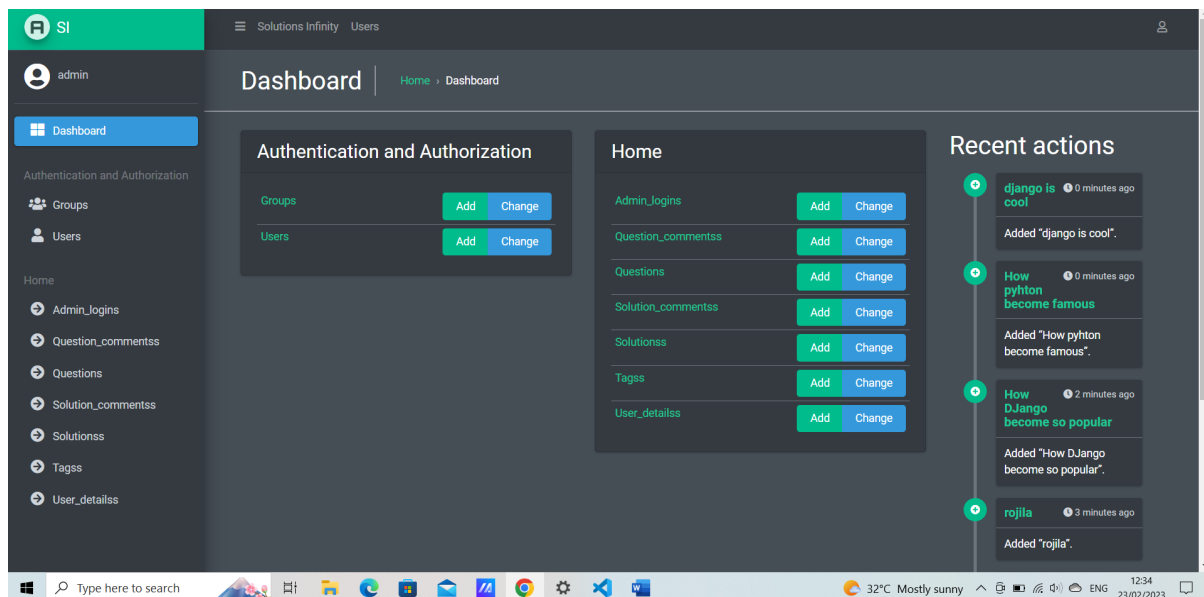
class Solution_comments(models.Model):
    comment=models.CharField(max_length=1000)
    comment_by=models.ForeignKey(User,on_delete=models.DO_NOTHING)
    solution=models.ForeignKey(Solutions,on_delete=models.CASCADE)
    commented_at=models.DateTimeField(auto_now_add=True)
    def __str__(self):
        return self.comment_by.get_username + ":" + self.commented_at

class Question_comments(models.Model):
    comment=models.CharField(max_length=1000)
    comment_by=models.ForeignKey(User,on_delete=models.DO_NOTHING)
    question=models.ForeignKey(Question,on_delete=models.CASCADE)
    commented_at=models.DateTimeField(auto_now_add=True)
    def __str__(self):
        return self.comment_by.get_username + ":" + self.commented_at

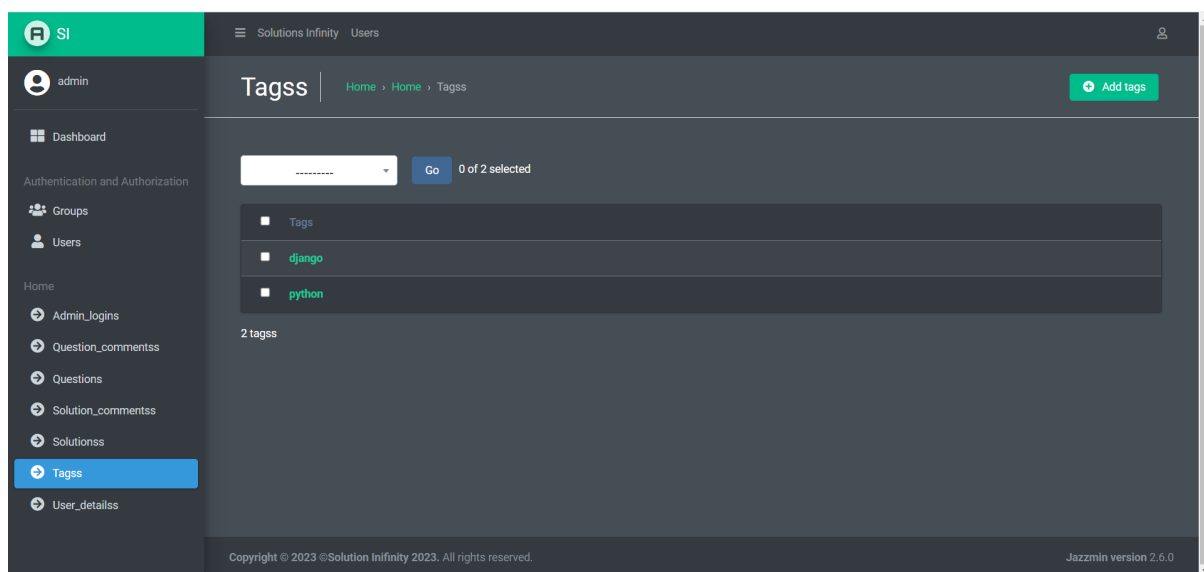
class Admin_login(models.Model):
    username=models.CharField(max_length=100)
    password=models.CharField(max_length=100)
    def __str__(self):
        return self.username
```

Screenshot of Admin Interface

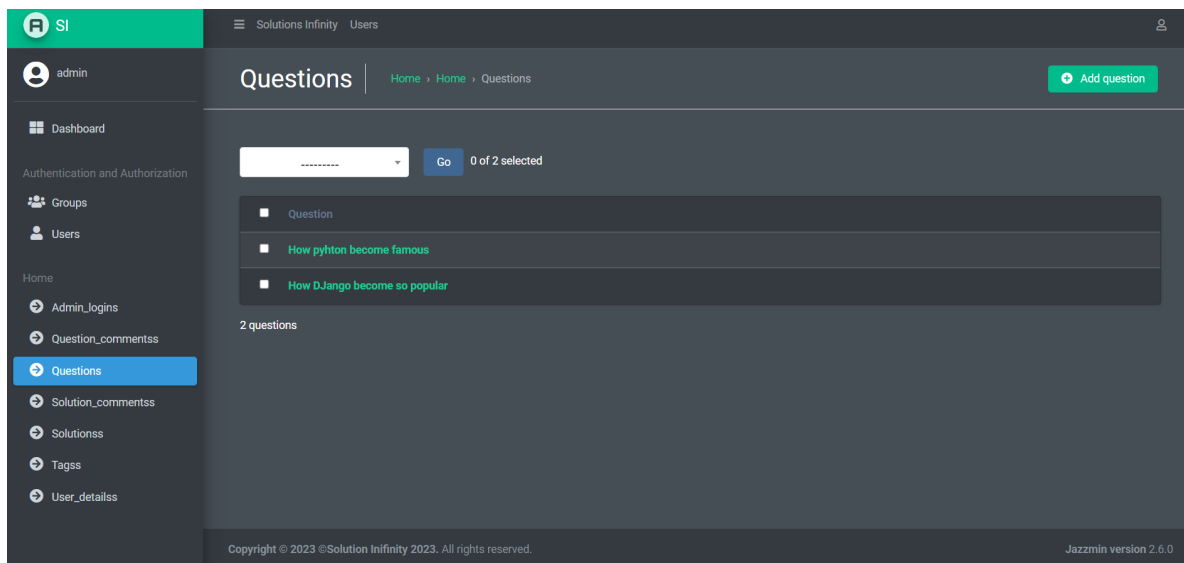
1)Dashboard



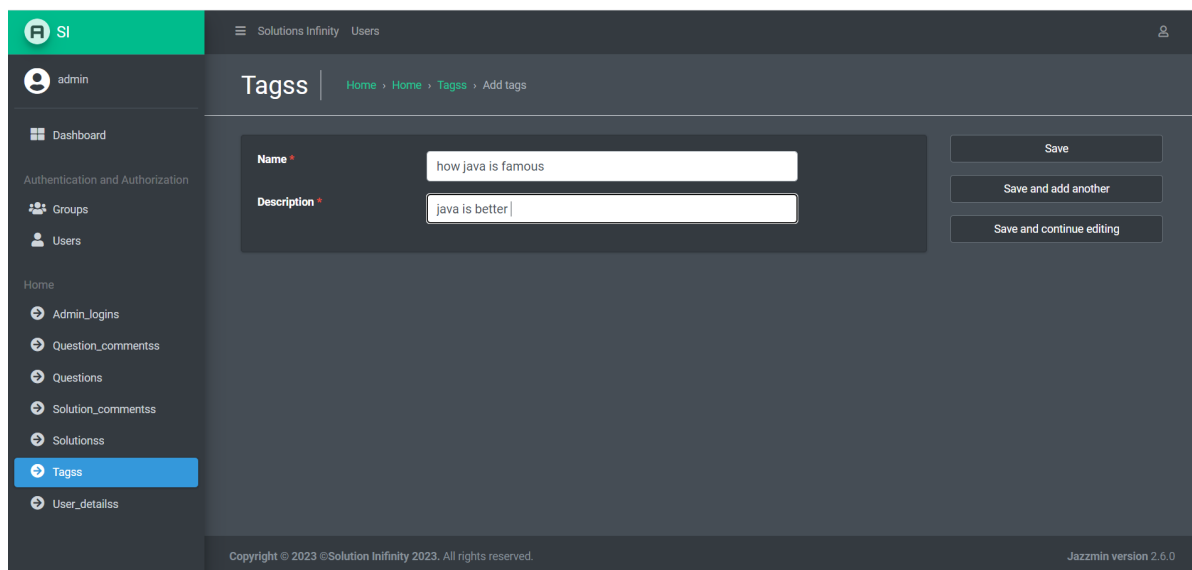
2)All Tags



3)All Questions



4)Add Tags



ER Diagram

