

استک آندرفلو

کوئرا تصمیم گرفته است برای راحتی شرکت کنندگان بوتکمپ، یک سامانه طراحی کند تا مانند سامانه‌ی استک اورفلو دانشجویان بتوانند سوالات خود را در آنجا مطرح کرده و سپس باقی دانشجویان به سوالات پاسخ دهند و با هم تعامل کنند. کوئرا از شما خواسته تا در پیاده‌سازی این سامانه به او کمک کنید.

پروژه اولیه

پروژه اولیه را از این [لینک](#) دانلود کنید. ساختار فایل‌های این پروژه به صورت زیر است:

```
StackUnderflow
├── config
│   ├── __init__.py
│   ├── asgi.py
│   ├── settings.py
│   ├── urls.py
│   └── wsgi.py
├── manage.py
├── questions
│   ├── __init__.py
│   ├── admin.py
│   ├── apps.py
│   ├── forms.py
│   ├── migrations
│   │   ├── 0001_initial.py
│   │   ├── 0002_initial.py
│   │   └── __init__.py
│   ├── models.py
│   ├── templates
│   │   ├── question
│   │   │   ├── question_detail_view.html
│   │   │   ├── question_form.html
│   │   │   ├── question_list_view.html
│   │   │   └── question_search.html
│   │   └── tag
│   │       ├── create_tag_form.html
│   │       └── tag_list_view.html
│   ├── tests.py
│   └── urls.py
```

```

|   └─ views.py
└─ requirements.txt
└─ templates
    |   └─ base.html
    |   └─ home.html
    |   └─ message.html
    |   └─ navbar.html
└─ users
    |   └─ __init__.py
    |   └─ admin.py
    |   └─ apps.py
    |   └─ forms.py
    |   └─ migrations
    |       └─ 0001_initial.py
    |       └─ __init__.py
    |   └─ models.py
    |   └─ templates
    |       └─ user
    |           └─ user_edit.html
    |           └─ user_login_form.html
    |           └─ user_profile_view.html
    |           └─ user_register_form.html
    |   └─ tests.py
    |   └─ urls.py
    └─ views.py

```

مدل‌ها

در این پروژه انتظار می‌رود که مدل‌های زیر را با توجه به خواسته‌های زیر پیاده‌سازی کنید.

۱. مدل User

این مدل از `AbstractUser` ارث‌بری می‌کند. و فیلدهای `phone_number` و `national_id` را اضافه دارد که شماره تلفن دارای ۱۱ کارکتر است و کد ملی نیز دارای ۱۰ کارکتر است.

۲. مدل Question

- `title` : عنوان سوال با محدودیت ۱۰۰ کاراکتر.

- `body` : توضیح سوال

- created_at : زمان ایجاد سوال
- updated_at : زمان آخرین ویرایش سوال
- user : کاربر ایجاد کننده سوال
- upvoters : افراد upvote کننده سوال
- downvoters : افراد downvote کننده سوال
- tags : تگ‌های سوال
- votes : یک property که تفاضل تعداد آرا مثبت و منفی را نشان می‌دهد.

۳. مدل Tag

- name : نام تگ با محدودیت ۵۰ کاراکتر

۴. مدل Answer

- body : توضیح پاسخ
- created_at : زمان ایجاد پاسخ
- updated_at : زمان آخرین ویرایش پاسخ
- user : کاربر ایجاد کننده پاسخ
- question : سوالی که این پاسخ به آن تعلق دارد
- upvoters : افراد upvote کننده پاسخ
- downvoters : افراد downvote کننده پاسخ
- votes : یک property که تفاضل تعداد آرا مثبت و منفی را نشان می‌دهد.

فرم‌ها

۱. فرم‌های users

در اپ user سه فرم باید پیاده سازی شود. اولین فرم، فرم ثبت نام کاربر است که در آن username و email و password و confirm_password و phone_number و national_id از کاربر گرفته می‌شود و کاربر ثبت نام می‌شود. در این فرم در صورت تکراری بودن ایمیل خطایی با متن:

This email is already exist!

و در صورت تکراری بودن نام کاربری خطایی با متن:

This username is already exist!

و همچنین برابر نبودن پسورد و confirm آن باید خطایی با متن:

Password must match

و در صورت خالی بودن پسورد یا confirm آن باید خطایی با متن:

Enter the both of password

پرتاب شود. در فرم بعدی ویرایش کاربر انجام می‌شود که به غیر از پسورد بقیه‌ی فیلدها می‌تواند تغییر کند. خطاهای این فرم دقیقا مانند خطاهای فرم قبلی است. در فرم آخر نیز یک username و یک password از کاربر گرفته می‌شود که این فرم لاگین است.

۲. فرم‌های questions

در این فایل چهار فرم برای question و answer و tag و search است. که در فرم question سه فیلد title و body و tag که تگ به صورت CheckboxSelectMultiple است. در فرم جواب هم تنها یک body است که پنج خط فضا دارد. در فرم سرچ یک query است که از جنس charfield است. در فرم تگ هم فقط نام آن می‌باشد.

تمپلیت‌ها

**** دقت کنید که تمامی تمپلیت‌ها از base ارث بری می‌کنند.****

۱. user_login_form

در این تمپلیت فرم login قرار دارد و یک دکمه‌ی submit نیز برای آن وجود دارد. همچنین csrf token نیز در آن وجود دارد.

۲. user_register_form

در این تمپلیت فرم register قرار دارد و یک دکمه‌ی submit نیز برای آن وجود دارد. همچنین csrf token نیز در آن وجود دارد.

۳. user_edit

در این تمپلیت فرم edit قرار دارد و یک دکمه‌ی submit نیز برای آن وجود دارد.

۴. user_profile_view

در این تمپلیت در چند پاراگراف به ترتیب فیلدهای زیر آمده است:

- username
- email
- date_joined
- phone_number
- national_id
- is_active

۵. question_form

در این تمپلیت فرم question قرار دارد و یک دکمه‌ی submit نیز برای آن وجود دارد. همچنین csrf token نیز در آن وجود دارد.

۶. question_view

در ابتدا سوال با موارد زیر می‌آید:

- اسم
- متن
- تاریخ ایجاد
- تاریخ آخرین تغییر
- تعداد آرا
- نام کاربر ایجاد کننده سوال
- تگ‌های سوال

- دو دکمه‌ی upvote و downvote
- دکمه‌ی ویرایش سوال
- هر کاربر همواره حق زدن یکی از این دکمه‌ها را دارد و چند بار نمی‌تواند این دکمه‌ها را بزند.

سپس جواب‌ها به ترتیب تاریخ ایجاد آن زیر هستند:

- متن پاسخ
- تاریخ ایجاد
- تاریخ آخرین تغییر
- تعداد آرا
- نام کاربر ایجاد کننده پاسخ
- دو دکمه‌ی upvote و downvote
- دکمه‌ی ویرایش پاسخ در پایان نیز یک فرم answer وجود دارد که کاربر می‌تواند با استفاده از آن یک پاسخ به سوال مورد نظر بدهد.

۷. question_list_view

در این تمپلیت لیستی از:

- عناوین سوالات
- نام کاربری کاربر پرسنده‌ی سوال
- تاریخ ایجاد سوال

۸. create_tag

در این تمپلیت فرم tag قرار دارد و یک دکمه‌ی submit نیز برای آن وجود دارد. همچنین csrf token نیز در آن وجود دارد.

۹. question_search

در این تمپلیت یک نوار سرچ مطابق فرم است و می‌توان با آن یک سوال را بر اساس عنوانش سرچ کرد.

۱۰. tag_list_view

در این تمپلیت لیستی از تگ‌ها نمایش داده می‌شوند.

ویووها و endpoint ها

تمامی url ها به شما در پروژه داده شده و شما باید view مورد نظر را طراحی کنید.

1. register

در این view شما باید یک کاربر را ثبت نام کنید. پس از اینکه ثبت نام انجام شد یک message موفقیت آمیز داده شود و سپس login انجام شود و به صفحه‌ی home ری‌دایرکت شود در غیر این صورت در همان فرم بماند و message ناموفق داده شود. (در سطح warning) محتوای message ها به شرح زیر است:

- پیام موفقیت آمیز:

you registered successfully

- پیام هشدار:

register unsuccessfully

2. login

در صورت درست بودن اطلاعات کاربر، یک مسیج موفقیت آمیز داده شود و کاربر به صفحه home ری‌دایرکت شود و در صورت ناموفق بودن آن کاربر یک مسیج در سطح warning بگیرد. محتوای message ها به شرح زیر است:

- پیام موفقیت آمیز:

login successful

- پیام هشدار:

login unsuccessful

3. logout

کاربر از حساب کاربری خود خارج شود.

۴. profile

کاربر اطلاعات حسابی که id آن را می‌دهد، مشاهده کند.

۵. profile_edit

کاربر اطلاعات حساب خود را ویرایش می‌کند.

۶. question_create

در این view با استفاده از تمپلیت یک سوال ایجاد می‌شود.

۷. question_detail

در این view یک سوال مطابق تمپلیت گفته شده نشان داده می‌شود.

۸. question_update

در این view سوال را می‌توان ویرایش نمود.

۹. question_delete

این view یک سوال را حذف می‌کند و به لیست سوالات redirect می‌شود.

۱۰. question_upvote

این view یک سوال را upvote می‌کند. (در صورتی که از قبل به این سوال رای نداده باشد).

۱۱. question_downvote

این view یک سوال را downvote می‌کند. (در صورتی که از قبل به این سوال رای نداده باشد).

۱۲. answer_create

در این view مطابق تمپلیت گفته شده یک جواب تولید می‌شود و به صفحه جزئیات سوال می‌رود.

۱۳. answer_update

در این view مطابق تمپلیت گفته شده یک جواب ویرایش می‌شود و به صفحه جزییات سوال می‌رود.

۱۴. `answer_delete`

در این view یک جواب حذف می‌شود و به صفحه سوال `redirect` می‌شود.

۱۵. `answer_upvote`

در این view مانند سوال یک جواب `upvote` می‌گردد.

۱۶. `answer_downvote`

در این view مانند سوال یک جواب `downvote` می‌گردد.

۱۷. `question_list`

در این view لیست سؤالاتی که کاربر پرسیده به آن نمایش داده می‌شود.

۱۸. `question_search`

در این view یک سوال بر اساس عنوانش جستجو می‌شود و سپس لیست عنوان‌های مرتبط نمایش داده می‌شود.

۱۹. `tags_list`

در این view لیست تگ‌ها نشان داده می‌شوند.

۲۰. `tag_create`

در این view طبق تمپلیت یک تگ تولید می‌شود.

****** احراز هویت در جلسات آتی تدریس می‌شود اما هدف این تمرین این است که شما درباره‌ی آن مطالعه کنید و در صورت لزوم در پروژه پیاده‌سازی کنید. ******