TRADECORE

Python/Django test

Object of this task is to create a simple REST API based social network in Django, and create a bot which will demonstrate functionalities of the API according to defined rules.

Social Network

Basic models:

- User
- Post (always made by a user)

Basic features:

- user signup
- user login
- post creation
- post like
- post unlike

For User and Post objects, candidate is free to define attributes as they see fit.

Requirements:

- use emailhunter.co for verifying email existence on signup
- use clearbit.com/enrichment for getting additional data for the user on signup
- use JWT for user authentication
- use Django with any other apps, databases etc.

Automated bot

To reiterate -- the purpose of the bot is to demonstrate the usage of the previously created **REST API**. This bot should read rules from a config file (in any format chosen by the candidate), but should have following fields (all integers, candidate can rename as they see fit):

- number_of_users
- max_posts_per_user
- max_likes_per_user

Bot should read the configuration and create this activity:

- signup users (number provided in config)
- each user creates random number of posts with any content (up to max_posts_per_user)

After creating the signup and posting activity, posts should be liked using following rules:

- next user to perform a like is the user who has most posts and has not reached max likes
- user performs “like” activity until he reaches max likes
- user can only like random posts from users who have at least one post with 0 likes
- if there is no posts with 0 likes, bot stops
- users cannot like their own posts
- posts can be liked multiple times, but one user can like a certain post only once

Notes

- emailhunter and clearbit have either free pricing plans or free trial, the candidate can use their own account
- visual aspect of the project is not important, the candidate can create a frontend for viewing the result, but it is not necessary. Clean and usable REST API is important
- the project is not defined in detail, the candidate should use their best judgment for every non-specified requirements (including chosen tech, third party apps, etc), however every decision must be explained and backed by arguments in the interview
- **Result should be sent by providing a Git url. This is a mandatory requirement.**