

Qs

You are required to build a simple event management backend where users can come and signup, and create events on the platform. The backend will be created using Django Framework and will consist of two models (User) and (Event). The user model will be extended from the AbstractUser and we will be adding additional property to it (mobile_number). The event will have its usual set of properties (title, description, date, location) and a foreign key to user as owner (one to one) and attendees (many to many).

The following endpoints are required in the API:

- User Authentication
- User Signup
- CRUD Event
- Attend a particular event

Objectives:

- ****Setup your codebase using Django Cookiecutter****
- Setup Git, make the initial commit
- Push to Github (create a public repo). For rest of the tasks, we will branch from master and work on a feature branch.
- Create feature branch, checkout.
- Configure user model and authentication. Commit. Bonus: Use semantic commits
- Configure event model and its CRUD APIs. Commit.
- GET events should be available public. Rest of CRUD operations should require login. Commit.
- Update and Delete (an event) should only be allowed by its owner. Commit.
- GET events should return complete User objects. Keep performance in mind. Commit
- Configure "attend" endpoint. Commit.
- Configure testing for all endpoints.
- Bonus: document all endpoints using Postman.
- Bonus: Configure TypeHinting in your code

Points to note:

- Please ensure you make an initial git commit and then commit your work after each objective!
- Make sure you are using Pythonic code and best practices.
- Please use the Docker setup that comes with Cookie Cutter

Qs. Write a Django migration for a table with 5 million records (sample of a few hundred attached) which adds a new column to the table in Postgres for created_at and sets default time to 2 days ago. The migration should be completed in under 10 mins.