

Full-stack developer technical test

Django + React

General information

- Please upload the solution to a public or private (with given access) GIT when you finish the test!
- By working on it, please register your time on the different tasks and upload this time log to the same repository!
- After receiving this test, you have one week to send back the link of the repository with your solution!
- Use English in the code, comments, and variable names!

Django

Create a Django 5.2 application with multiple apps. The application must support the following data structure:

Please use PostgreSQL for the database.

At our company, we want to schedule our internal meetings.

- Every appointment must have the following:
 - a. Start date-time
 - b. End date-time
 - c. Title
 - d. Description
 - e. Employee
- Every employee must have the following:
 - a. Name
 - b. E-mail address
 - c. Position (employee, manager)
- Every employee must work in a department.
- Every department must have the following:
 - a. Name
 - b. Manager
 - c. Description
- Next to these fields, you can create other columns and models you feel necessary.

API

We would like to see an API to access the data. You could use Django-rest-framework, Django-Ninja, or any other third-party dependency.

Tasks:

- Create the necessary endpoints to the DB for CRUD.
- Create an endpoint that gives back every employee of a department. This endpoint will be used for a department drop-down on the front end.
- Create an endpoint that gives back employees with filtering on the e-mail address.
 - a. **Bonus:** this filter must allow filtering on the name as well.
- Set the API to be able to handle calls from **localhost:5000** (CORS)
- Implement Custom Exception Handling. Thrown exceptions should be handled by the front end!
- Write Unit/Integration tests whenever you see fit.

Front-end

As a user, I want to see appointments in a calendar component that I can use as a sidebar in the application.

I also wish to have a popup showing additional data for appointments.

You are not allowed to use third-party components for the calendar, but you can use it for smaller parts, like a date picker or button. For the development, use React and Typescript!

Description of the calendar:

- It must have a date selector.
- It must show only one day simultaneously with hourly blocks.
- It must load the closest date, which contains at least one appointment.
- It must represent the hours on the Y-axis.
- The maximum width of the calendar is 500px.

Description of the appointments:

- It must be placed by the start and end times.
- Show the item with 100% width if there is only one appointment.
- If more than one meeting occurs at the same time or overlaps, make the visible appointment slightly smaller and add a counter next to it showing the number of hidden sessions.

