



# HireHive

Web-site to find jobs





# How we covered the project requirements: Front end

- All needed interfaces / classes for APIs from back-end

Company, User, JobPost interfaces and service classes.

- All needed services for getting data from API

Company-service, user-service, job-board-service.

- Minimum 4 (onclick) events, which hits API

Submit, CreatePost of Login, Registration, Job pages etc.

- Minimum 4 [(ngModel)] - job-post, job-post-list

- Minimum CSS styling - everywhere.

- Configured Routing module

Configured through app.module.ts

- Usage of directives: ngFor, ngIf

For looping getCompany(), getApplicants() result.

- JWT based authentication - jwt(restframework\_simplejwt)

- interceptor - used for authorization

- login - working

- logout - working



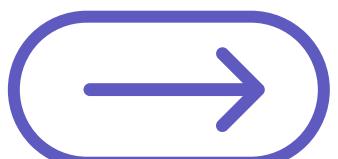
# Back End



1. **Minimum 4 models** - Company, JobPost, Applicant, User.
2. **Minimum 1 model Manager (optional)** - UserManager
3. **Minimum 2 relations between models (ForeignKey) - 4**
4. **Serializer**
  - a. **at least 2 from serializer.Serializer** - User(VerifyUser, CreateToken, GetUsers).
  - b. **at least 2 serializer from serializer.ModelSerializer** - Job\_Board, Applicant, Company, User.
5. **Views**
  - a. **at least 2 FBV drf view** - Applicants, Users
  - b. **at least 2 CBV APIView** - Job\_Board, Company, Users
6. **Token based authentication**
  - a. **login**
  - b. **logout**
7. **At least for 1 model CRUD operations**
  - a. CRUD job posts(ModelSerializer, ModelViewSet, permission\_classes)
  - b. CRUD companies(ModelSerializer, ModelViewSet, permission\_classes)
  - c. Applicants(ModelSerializer, ModelViewSet)
8. **Postman requests with all implemented methods** - 
- 9.



# What was done and features



- While registering users get verification code with templated mail.
- Custom Manager for creating Users and superUsers.
- CustomUserAdmin to manipulate with user permissions and groups.
- Used caching of user session and code with Redis.
- Used Swagger for documentation.
- Used FilterClasses for searching.