



Java Coding Challenge

Task

Build a backend service that handles a (very simple) recruiting process. The process requires two types of objects: job offers and applications from candidates. minimum required fields for the objects are:

Offer:

- jobTitle (unique)
- startDate
- numberOfApplications

Application:

- related offer
- candidate email (unique per Offer)
- resume text
- applicationStatus (APPLIED, INVITED, REJECTED, HIRED)

Not all of the fields have to be persisted. You may use ad hoc calculation, event sourcing, or whatever you see fit. These are the fields that must be returned by the API. You may add fields where necessary.

Use cases

- user has to be able to create a job offer and read a single and list all offers.
- candidate has to be able to apply for an offer.
- user has to be able to read one and list all applications per offer.
- user has to be able to progress the status of an application.
- user has to be able to track the number of applications.
- status change triggers a notification (*)

(*) a log output will suffice as a notification here, but you should design it as if each status change triggers a completely different business case.

Technical requirements

Use SpringBoot to build this service. The service must run standalone and must not require any third party software to be installed. The service must communicate Json over http

(REST). Return proper status codes for the most common problems. The data does not have to be stored permanently, it may be handled in-memory during runtime. Use Hibernate to handle database operations.

Things we are looking for

- a description on how to build and use the service
- clean code
- use of the spring framework and spring best practices
- structure of the project
- Unit tests
- how you test your code

We do not require you to build a frontend, only endpoints are relevant. And we can't stress enough: we are looking for clean, structured code. If the task takes you too long, you can hard code a shortcut, mark it with a comment and it will be fine. But make sure everything you hand in is well laid out and tidy!

Please zip your code and any instructions the reviewer might need to run your application. Send the zip file to hr@heavenhr.com. Include any instructions to build and run in a README file.