JOB HUNT-SPRING BOOT BACKEND APPLICATION

[PROBLEM STATEMENT](#_oy2qch6ysckj)

[PROJECT STRUCTURE:](#_ckhvzgtoblma)

[PROJECT FLOW:](#_4cgle8ki8key)

[SOURCE CODE:](#_tdxvhczask75)

# **PROBLEM STATEMENT**

The JOBHUNT allows applicants to search for job opportunities based on job or companies or location and allows users to apply for the same.The users will be able to view the jobs for which they have applied.In addition to this the users can upload the interview process that they have undergone or if they wish to share the experience with the particular company.The employees can add rating and review for that company where the average of all the rating will be displayed along with company details.If the user is not satisfied with the present job vacancies,they can specify their requirements and this will be saved separately.When any job vacancy arises that satisfies user requirement specification,a notification through email is generated to them.

A company Admin can have access to only one company at a time.He can publish new vacancies,edit published vacancies and view the users who have applied for the vacancies in that company.The admin can mark a particular user as contacted for their future references.

# PROJECT STRUCTURE:

* Create a **Spring boot** project in eclipse.
* Have packages such as:
* com.zilker.onlinejobsearch.beans,config,constants,controller,customException,dao,delegate,utils.
* In CustomException define all the custom exceptions for the project.All this custom exception must extend another custom exception called ApplicationException which extends Exception.
* In This backend project all the requests from front end are handled and required data is fetched and returned to front end in JSON Format.
* For wrapping and sending data from backend to frontEnd ,reponse Wrapping is used.

# PROJECT FLOW:

* Define three controllers, for companyController, User Controller , JobController and define methods with different URLpatterns.
* Call respective methods in delegate,from controller.
* Do all business logic in delegate.
* Call methods in DAO to perform database operations.
* And return the result in either object or arraylist to delegate , then to controller.
* Controller wraps the result in ReponseEntity object and sends it to the frontend app.
* Define error code and error message for each exception and set it to the response object sent in case of any exception.
* Create a separate ReponseGeneratorClass in utils, for sending response from backend to frontend.

# SOURCE CODE:

<https://github.com/sruthiviswanathan/Zterns-19-Sruthi/tree/master/JobHunt-Backend-SpringBOOT>