Back End Java Developer

Seniority: Senior
Team: Project 2 (Charter Communications project)
Jobvite:

Role Description

Designs, develops, modifies, and implements web applications and/or services designed to support our Charter Portals Platforms based web applications. The selected candidate will be a member of the team that is responsible for all aspects of the software development lifecycle. The selected candidate will also be expected to be on top of industry tools and trends in web development and/or web API development.

Tech stack: ReactJS, ES6, Webpack, Babel, Immutable, Java 8, Spring Boot, Netflix OSS, Mongo.

Responsibilities

- Develop software based on Portals Platform standards and patterns
- Work as a part of a development team
- Participate in peer and code review activities
- Mentor less experienced developers

Must Have

- Strong working knowledge and experience with Spring & Spring MVC
- Solid understanding of OOP
- Experience with Unit Testing frameworks and continuous integration tools
- Strong communication skills

Nice to Have

- Spring Boot
- Solid understanding of RESTful API design
- Good understanding of git version control system
- Experience with Java 8
- Experience with Maven

Homework Problem

All candidates who are a fit for this role will be required to do a homework problem before any interviews. This will be the basis for the first client interview. Your solution must be coded in Java and use Spring Boot.

A retailer offers a rewards program to its customers, awarding points based on each recorded purchase.

A customer receives 2 points for every dollar spent over $100 in each transaction, plus 1 point for every dollar spent over $50 in each transaction
(e.g. a $120 purchase = 2x$20 + 1x$50 = 90 points).

Given a record of every transaction during a three month period, calculate the reward points earned for each customer per month and total.

Technical Notes and Architecture:

- *Code your solution in Java and Spring*
- *Make up a data set to best demonstrate your solution*
- *Use OOP concepts as much as possible when designing classes.*
- *Swagger/OpenAPI (Optional) - helps communicate the contract in a better way*
- *Document all response codes expected. The REST API should be sending the appropriate code and not 200 always.*
- *Functional Programming constructs of Java 8 as appropriate.*
- *Generic exception handler for un-anticipated exceptions.*
- *Assume that the client will look at your commit history, so it should reflect a good project progression.*

Additional Requirements:

- *Follow standard best practices for structuring the code.*
- *Prepare and provide Test Data along with the code.*
- *Both positive and negative unit test cases for all operations – to be run as part of the build process.*
- *Implement RestAPI's for all CRUD operations – in this case – creating/updating transactions, calculating and providing reward information for a User. (Java Backend/Full Stack)*
- *Consistent error handling and reporting of all failures including unexpected error conditions*
- *Use of appropriate logging levels, framework*
- *Reward calculation logic should be accurate*
- *Readme file is mandatory – must contain the steps required to build and run/test the code*
- *The solution must be checked into Github (provide a public github url)*

How to Submit:

- *Check solution into GitHub*
- *Send email to Overactive with the public Github URL*