



TECH SCREENING JAVA

CASUS

TECH SCREENING JAVA

| | |
|----------------------------|---|
| 1 INTRODUCTIE | 3 |
| 2 FUNCTIONELE REQUIREMENTS | 4 |
| 3 TECHNISCHE REQUIREMENTS | 5 |
| 4 EXTRA REQUIREMENTS | 5 |
| 5 OPDRACHT | 6 |

INTRODUCTIE

Het leven van een IT Rockstar gaat natuurlijk niet over rozen. Zowel zakelijk als privé hebben wij dagelijks te maken met de meest uiteenlopende irritaties. Als Rockstar in hart en nieren doe je er natuurlijk alles aan om die ongemakken weg te automatiseren en als de wereld daar ook beter van wordt is dat uiteraard mooi meegenomen.

Daarom bieden we jou graag een technische challenge aan! De uitdaging is als volgt:

Zoals onze naam doet vermoeden hebben we een bijzondere link met muziek, en dan vooral keiharde Rockmuziek. Dat genre is enorm uiteenlopend en heeft een prachtige historie. Vanwege al die variatie zou het handig zijn om met een paar clicks de juiste band voor het moment te vinden! Van onze IT Rockstars wordt verwacht dat ze zo'n app snel en kundig in elkaar zetten, dan maak je altijd een top indruk bij je collega's en vrienden! Aan jou dus om de volgende specs te vertalen naar de nieuwe app™

Good luck and may the force be with you.

FUNCTIONELE REQUIREMENTS

Backend

De backend moet JSON bestanden met bands en nummers uit kunnen lezen en opslaan. Op de opgeslagen data moeten CRUD (Create/Read/Update/Delete) operaties uitgevoerd kunnen worden

Onder CRUD functionaliteit verstaan we:

- › (C): Het kunnen toevoegen van nieuwe band(s) of nummers met alle details die ook in de JSON staan waarbij de naam van de band natuurlijk niet dubbel voor mag komen;
- › (R): Het kunnen zoeken naar een band op naam of nummers op genre (voor wanneer je echt zin hebt in Finse Death Metal);
- › (U): Het kunnen updaten van alle informatie gerelateerd aan bands en nummers op basis van de bijbehorende id's;
- › (D): Het kunnen verwijderen van een band of nummer op basis van de id.

We zijn bij het inlezen van de data alleen geïnteresseerd in:

- › alle bands waarbij het genre "Metal" bevat
- › alle nummers die zijn uitgebracht voor 2016

De JSON bestanden kunnen uitgeserveerd worden via de Web API zelf, als een statisch bestand.

De benodigde bestanden vind je hier:

<https://www.teamrockstars.nl/sites/default/files/artists.json>

<https://www.teamrockstars.nl/sites/default/files/songs.json>

Frontend

Bij deze opdracht wordt er geen frontend implementatie verwacht en moet het mogelijk zijn de web API te testen met bijvoorbeeld Postman of cURL.

TECHNISCHE REQUIREMENTS

Quality assurance

We zien nu al aankomen dat dit een super populaire web app wordt. En bij een veelgebruikte app horen goede unit tests en goed leesbare code. Immers gaat deze applicatie nog lang onderhouden moeten worden door mede-developers. We verwachten dan ook unit tests voor de backend en duidelijke naamgeving en documentatie waar nodig.

EXTRA REQUIREMENTS

Om jou als potential IT Rockstar extra te challengen, willen we jou vragen om uit onderstaande lijstje nog een aantal extra requirements te selecteren die je toevoegt in jouw applicatie. De keuze voor de requirements is vrij, het aantal niet. Zonder te veel in hokjes te denken...hieronder een indicatie voor de aantallen. Meer mag altijd!

| | |
|---------|-------------------------------|
| Junior: | minimaal 1 extra requirement |
| Medior: | minimaal 3 extra requirements |
| Senior: | minimaal 4 extra requirements |

Waar moet je dan uit kiezen? Zie hieronder:

Opties:

- › **Database;** Kies een database(geen in-memory) en zorg dat je alle queries zelf schrijft om te laten zien dat je de database beheerst.
- › **Design Patterns;** Kies 4 design patterns en implementeer deze in de code en ligt ze toe.
- › **Docker;** Draai de applicatie in docker (en de database apart als je deze gekozen hebt).
- › **Tests;** Zorg voor optimale coverage. Gebruik Unit tests, Integration tests en Behavior Driven Development.
- › **Cache;** Implementeer een cache systeem dat responses cached voor als de database down is.
- › **API documentation;** Maak een API documentatie visualisatie beschikbaar.
- › **API versioning & content type negotiation;** Implementeer meerdere versies van je API en maak content type negotiation mogelijk.
- › **API security;** Beveilig de API in de manieren die jij kunt bedenken.
- › **MicroProfile;** Bouw de applicatie volledig gebruik makend van MicroProfile.
- › **RESTful API;** Houd rekening met de RESTful principes en bouw de API zodat deze eraan voldoet.

OPDRACHT

Focus

De focus van deze hele exercitie ligt op de functionaliteit. Daarnaast ontvangen we graag ook vooraf toelichting over bepaalde shortcuts die genomen zijn en een aanpak/ architectuur keuze/ strategie die je gebruikt hebt.

Verder wordt alle code beoordeeld op de volgende aspecten:

- › Security
- › Performance
- › Code style

Rules

Ondanks dat wij niet van regels en verplichtingen houden, zijn er toch een aantal zaken waar je rekening mee moet houden:

- › Je mag maximaal 4 uur van jouw kostbare tijd aan de casus spenderen. Op deze wijze krijgen wij het meest realistische beeld van jouw skills, vaardigheden en snelheid.
- › Het is overigens geen schande als je het binnen deze tijd niet 'af' krijgt!
- › Kwaliteit is in deze beter dan kwantiteit, dus zorg dat hetgene dat af is ook goed is.
- › De applicatie moet met simpele commando's te runnen zijn en voorzien zijn van duidelijke instructies hoe dit te doen.
- › Je levert de solution minimaal 48 uur van te voren aan bij iemand van de techboard. De gegevens van de techboarder worden verstrekt door de Agent die jouw kwalificatietraject begeleidt.
- › Na het gesprek volgt er een definitieve GO / NO GO vanuit de Rockstar! Bij een GO ontvang je binnen 48 uur een contractvoorstel vanuit jouw Agent!



WWW.TEAMROCKSTARS.NL