

Loyalty Program Backend Challenge –

NodeJs Backend Engineer

As a Backend Software Engineer for the trivago Business Studio project, you'll actively shape the project's future by establishing a quality-driven engineering culture as well as contributing to the integration, refinement and planning of new and existing services and their architecture.

trivago Business Studio is a Platform as a Service (PaaS) targeted at our B2B customers. Our goal is to develop the technical infrastructure that will power the hotels of the future. With an intuitive UX and well-crafted APIs, our marketplace will integrate cutting-edge applications developed by trivago and partners to offer easy access to a customer base of over 500 000 hotel owners!

This assignment is designed to give you a glimpse of some of the challenges you will be facing in this role. Please be aware there are no perfect solutions - for us, it's more important to see how you find solutions, process your ideas, structure your thoughts and how you make decisions.

Be creative but realistic about what's possible. We are thrilled to get to know a bit more about the way you solve tasks.

Good luck!

SUBMISSION DEADLINE: As indicated in the email.

HOW TO SUBMIT: Make sure to create a single compressed zip folder which contains all relevant files. (Make sure the .git folder is included and that the node_modules, on the other hand, is not part of the compressed zip.) If you would like to add any comments feel free to add them to your README.md file.

Product Overview

trivago Business Studio

trivago Business Studio [\[1\]](#) is an App Platform that enables independent hoteliers to improve their online marketing performance and manage their property operations efficiently with free and paid apps. Hoteliers can select the right apps that fit their business needs. With personalized app recommendations within the platform, hoteliers can easily find the apps that are most relevant to their business. It works very simply; a hotelier receives notifications about a recommended app, then he/she can read the description and click the ADD button to install the app. If the product is a paid app, there is another step where the hotelier has to enter their payment details in order to start the yearly subscription. ¹

THE CHALLENGE

Introduction

The goal of this case study is to get insights into your approaches of implementing modern, testable and maintainable services using NodeJS. The exercise represents some of the daily work you'll do within this software engineering position. You are recommended to set yourself a time limit of 5 to 8 hours and submit your latest state, even if some tasks are not finished 100%.

Please note that we do not allow the publication of this case study and would like you to not publish your results/code as well such that applicants have a comparable starting point on which an evaluation can take place being fair for everyone.

Task

For one of your next big projects, the team has decided to go with microservices. There are multiple experienced engineers with strong knowledge in NodeJS and while other programming languages were on the table, the team decided to proceed with Node only. It is your task to develop a containerized prototype for a small and simplified part of the system using docker-compose or Kubernetes.

This part should enable frequent travellers to bypass the regular checkout flow and reserve a hotel room using available "bonus points" when they have an account in the system. There are no given API or payload specs and you are asked to come up with your own. In addition, you decide how the communication between services should work.

Scope

- Create at least two or more services using NodeJS
- Only one service or gateway is accessible publicly, all other communication happens internally
- Some, determined by you, critical aspects are covered by tests

¹ Business Studio: A platform for independent hoteliers provided by trivago to manage their online marketing channels. Reference: <https://studio.trivago.com>

- Create a configuration file for a CI/CD pipeline using an automation server of your choice (example: jenkins, circle-ci, ...)

Preparation

- Setup a new local Git repository within an empty folder (When using services like Github, please keep the repositories private)
- Create a README.md file which holds a short description of your setup and how to run tests and application later

Note: Full requirement details are included in additional files.