

## Desafio Técnico – Desenvolvedor(a)

Este desafio tem como objetivo avaliar sua capacidade de desenvolver uma aplicação completa, contemplando backend (API) e frontend, utilizando tecnologias atuais, boas práticas de arquitetura e organização de código.

A seguir, você encontrará todas as instruções necessárias para realizar o desafio, incluindo descrição do sistema, regras de negócio, tecnologias obrigatórias e detalhes sobre a entrega.

---

### Objetivo do Desafio

Desenvolver um sistema completo para **gestão de reservas de veículos**, dividido em dois projetos independentes:

#### 1. API (Backend)

- Node.js
- TypeScript
- MongoDB

#### 2. Frontend

- Angular 18
- Interface desenvolvida com base no protótipo oficial no Figma:
  - <https://www.figma.com/design/qnyLkPP0Xe2mJMgZuLnAD0/Teste---App-AI-uguel-de-carros?node-id=2-2&t=P0mczIHcxfsKFBiY-1>

Os assets de design necessários estão disponíveis em:

<https://compartilhamento.happmobi.com.br/TesteFrontend.zip>

---

### Importante – Escopo por Área do Processo Seletivo

- **Candidatos a Backend:** desenvolver **somente a API**, considerando apenas o escopo do backend.
- **Candidatos a Frontend:** desenvolver **somente o Frontend**, considerando apenas o escopo da interface.
- **Candidatos a Full Stack:** desenvolver **API e Frontend**, entregando todo o escopo completo.

---

### Descrição do Sistema

O sistema deve permitir que usuários se cadastrem, visualizem veículos disponíveis e realizem **reservas**.

É importante destacar que se trata **exclusivamente de um sistema de reservas**, não havendo cálculos de cobrança, tarifas ou pagamento.

---

### Funcionalidades Obrigatórias

## Autenticação

- Login do usuário (única rota pública).

## Usuários

- Cadastro de usuário
- Edição de usuário
- Remoção de usuário

## Veículos

- Cadastro de veículo
- Edição de veículo
- Remoção de veículo
- Listagem de veículos cadastrados

## Reservas

- Reserva de um veículo
- Liberação (cancelamento/finalização) da reserva
- Listagem dos veículos reservados por um usuário



## Regras de Negócio

- Todas as rotas, **exceto login**, devem ser protegidas por **token** (JWT ou equivalente).
- Um veículo **não pode** ser reservado caso já esteja reservado.
- Um usuário **não pode** ter mais de um veículo reservado simultaneamente.



## Entrega

A entrega deve ser feita através de um repositório Git, contendo:

- Projeto do **Frontend** (Angular 18)
- Projeto da **API** (Node.js + TypeScript + MongoDB)
- Instruções claras para execução local (README)

**Compartilhe o repositório** com os usuários abaixo:

- [diegoa83](#)
- [dcastrohappmobi](#)



## Prazo

Data limite de entrega: **05/02/2026** até o final do dia.



## **Critérios de Avaliação**

A avaliação será baseada nos seguintes aspectos:

- Qualidade da implementação da API
- Correta implantação das rotas
- Segurança e proteção das rotas
- Aplicação das regras de negócio
- Uso adequado das tecnologias solicitadas
- Arquitetura, organização e estrutura do código
- Clareza, legibilidade e manutenibilidade
- Organização dos arquivos e módulos



## **Diferenciais Avaliados**

Não são obrigatórios, mas contarão pontos positivos:

- Testes automatizados (unitários e/ou integração)
- Containerização da aplicação com Docker
- Documentação da API utilizando Swagger