Context

We use GIT to version control our entire code base. All our projects are part of a single GIT repository. We organize and manage that monorepo using a tool called Nx.

There are many advantages to using a monorepo (https://monorepo.tools). We believe Nx (https://nx.dev) is the best tool out there.

Technical assessment

Before your technical assessment, we ask you to prepare an exercise.

You should be familiar with our `patient-booking` app in the `staging` environment since you used it to book appointments for the recruitment process.

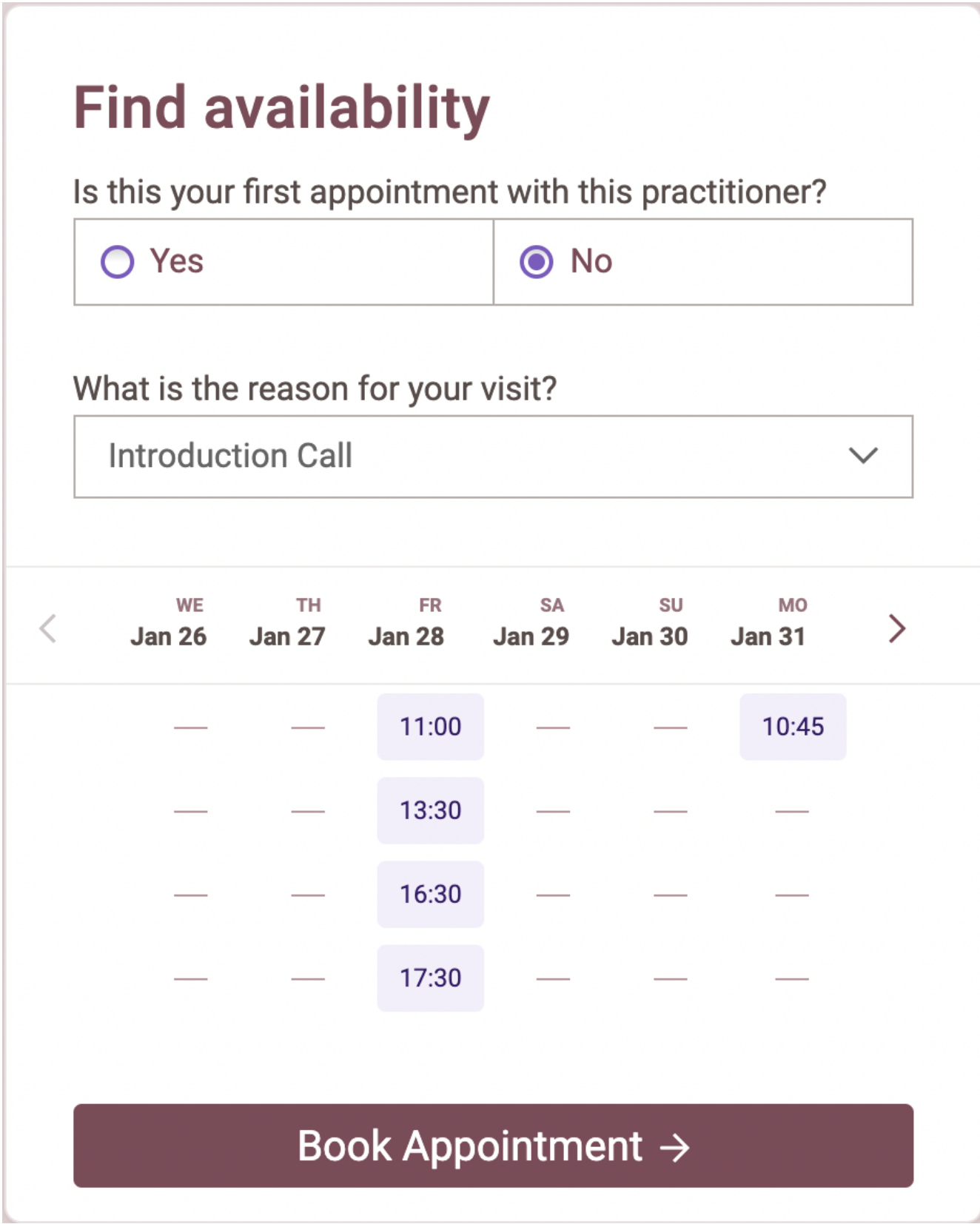
Frontend and Mobile

Your task is to copy-cat a basic/simple version of the booking component shown on: https://staging-patient-booking.rosa.be/fr/antoine-staging-pairet/detail

Based on the docs from Nx (https://nx.dev/getting-started/intro, https://nx.dev/react-native/overview#react-native-plugin) and the tech you are the most familiar with, we expect you to:

- Create a new Nx workspace
- Query our `staging` api to retrieve availabilities (use the Network tab in your console to understand what API call you need to fire)
- Create a UI to display availabities

During the technical assessment, we will discuss the code that you wrote. We will not judge the aesthetic aspects of what you produce. You should not rush to build all the features that exist.



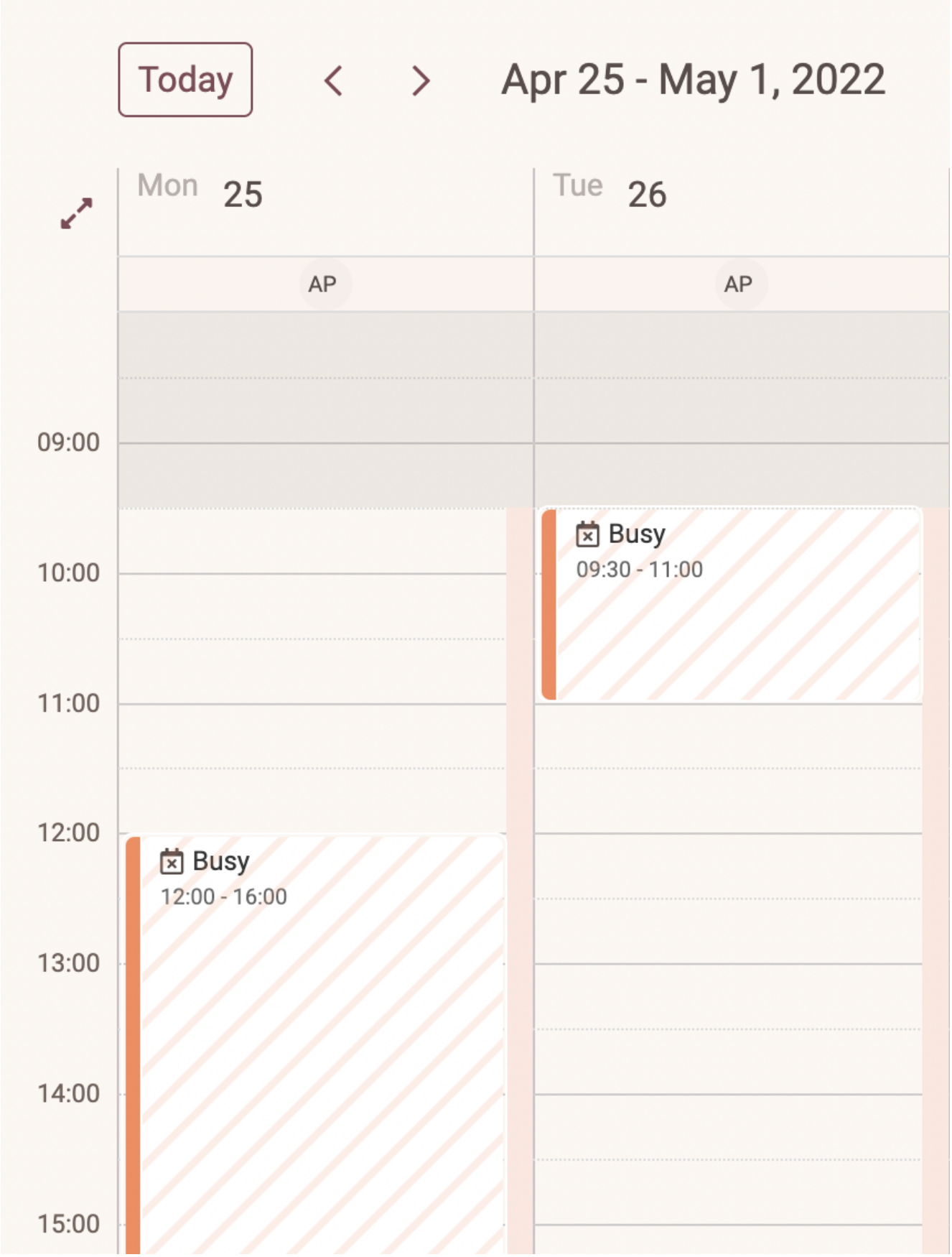
Backend

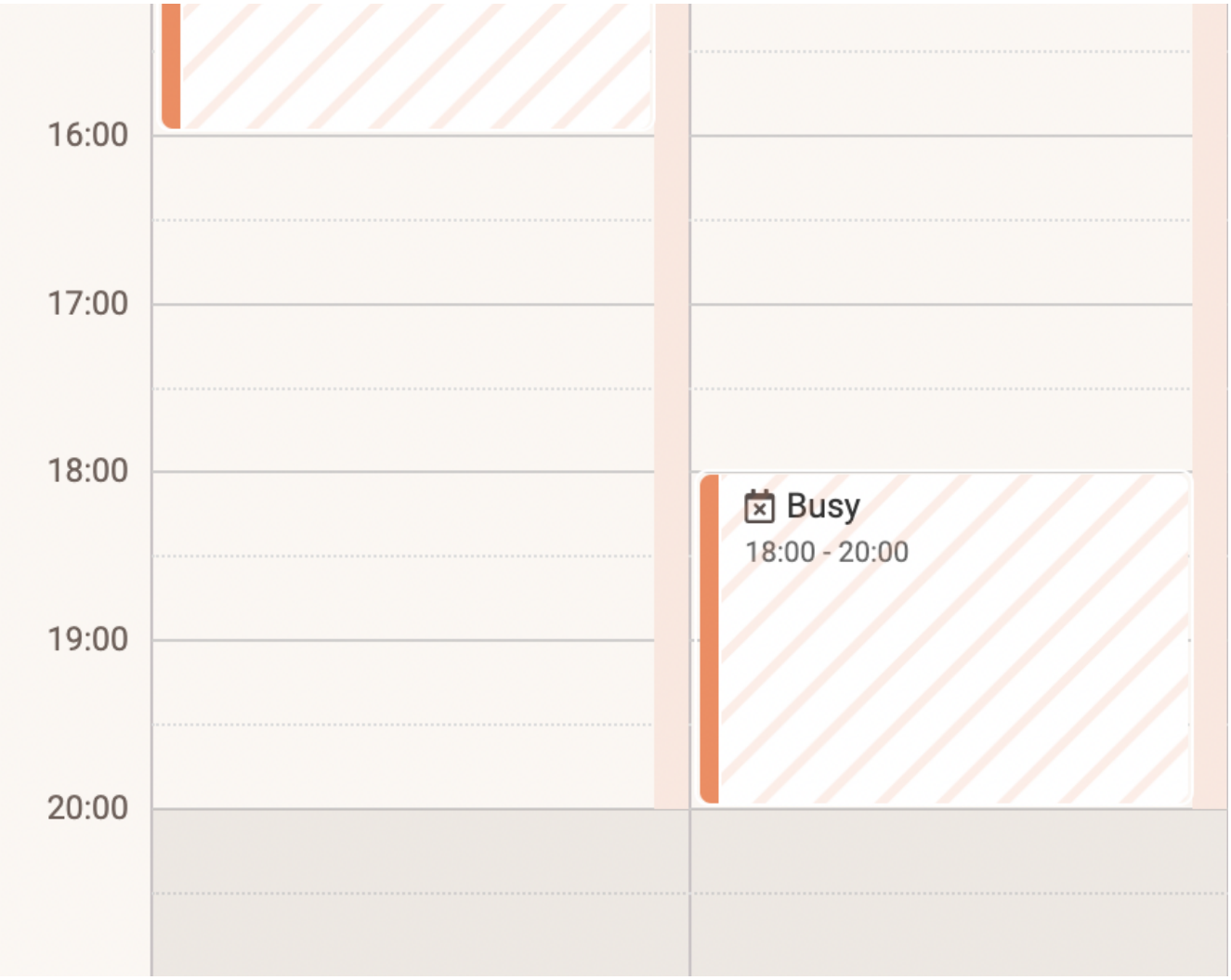
Your task is to copy-cat a basic/simple version of the API endpoint powering the booking-widget:
https://staging-patient-booking.rosa.be/fr/antoine-staging-pairet/detail

Based on the docs from Nx (https://nx.dev/getting-started/intro, https://nx.dev/react-native/overview#react-native-plugin) and the tech you are the most familiar with, we

expect you to:

- Create a new Nx workspace
- Create an API with 2 endpoints
  - Based on `from` and `to` the first endpoint should return a list of availabilities
  - Based on a `date`, the second endpoint should return the first availability after that `date`





The availabilities should be computed based on the schedule of a health professional (HP) and the existing appointments. In the example above

- the HP has 2 `appointmentSlots` on Monday and Tuesday from 9:30 to 20:00
- there are 3 `appointments` that are booked
  If the API is queried to retrieve `availabilities` for these 2 days, the answer should contain 3 availabilities:

1. Monday from 9:30 to 12:00
2. Monday from 16:00 to 20:00
3. Tuesday from 11:00 to 18:00

You can choose whatever framework you are comfortable with to build the API (express, nest.js, etc.). The only requirement is to write it in Typescript within a Nx mono-repo.
During the technical assessment, we will discuss the code that you wrote, the architecture, challenge design decisions and the algorithmic part.