

## Backend coding challenge

The goal of the task is to create a mini Doodle. You should design and implement a high-performance simulation of a meeting scheduling platform using Spring Boot and Java technologies. The service should enable users to manage their time slots, schedule meetings, and view their custom calendar availability.

In this service, users should be able to define available slots, which can later be converted into meetings. Each user should have a personal calendar where their time is managed. Calendar as the term in the task should be present only in the domain in the service. A slot can be booked as a meeting with a specific title and participants. The system should support querying free or busy slots, providing an aggregated view for a selected time frame. All data should be persisted to allow for proper management and querying.

### Functionalities to implement:

**Time slot management** - allow users to create available time slots with configurable duration in calendars, delete or modify existing time slots, and mark time slots as busy or free according to their availability.

**Meeting scheduling** - enable users to convert available slots into meetings, add meeting details such as title, description, and participants.

Assume the platform may be used by hundreds of users with thousands of slots. Strive to design your solution according to that.

### Instructions:

Create a new Git repository with an initial commit containing README.md.

Develop your solution with regular, meaningful commits and share the repository link when complete.

Your solution should be runnable locally using docker-compose.

Don't forget to include all the dependencies of your service in the composer file.

Solution should document clearly how the service can be consumed.

Metrics and documentation is a plus.

Implementation of tests is a plus and something we would appreciate.

The goal of the task is to see your design and tech decision making.

Feel free to deliver the solution that is not complete but explain the overall idea.