Machine Scheduler

✅ Build a small application that lets a user create machines and schedule jobs on those machines. Implement it such that its functionality can be validated through calls to its API (using a browser or software like *Postman*). If you want to make a client app for it, that's cool but it's not expected.

💡 The purpose of this challenge is to see how you code and understand how you solve problems. We will go over your code together and discuss your implementation, design choices, etc.

🐍 Code your app in any language and framework.

Problem 1

Create an endpoint at /machine/ that facilitates basic CRUD functionality for objects in a `machine` table. A user should be able to:

- Create a `machine` object in the database. The object need only have `id` and `name` attributes (but feel free to add more if it makes sense).
- Retrieve a list of machines.
- Update and delete specific machines.

Problem 2

Create and endpoint at /job/ that facilitates basic CRUD functionality for objects in a `job` table. You may follow the same steps as in Task 1. `job` objects have the

following attributes: `id`, `machine_id` (referencing an object in the `machine` table), `start_datetime`, and `duration_seconds`.

Problem 3

Create an endpoint at /machine/<id>/show-jobs/ which returns all jobs on a given machine.

Problem 4

Handle the following exceptions gracefully:

- A job is created on a machine that does not exist.
- A job is created/updated on a machine that does exist, but there is a time-collision with an existing job on that machine.
- A machine is deleted, but it had one or more jobs assigned to it.