

# Assignment PHP Backend

## Description

Thanks for applying for the backend engineer position at Temper!

This challenge is the first step of technical evaluation in the hiring process. This is your opportunity to show us how you work. If we decide to continue to the next step, your work will also form the starting point of our technical interview.

**Please read this document carefully before starting to work on the assignment!**

## Time to spend on this assignment

There's no fixed deadline, but we have a **soft deadline of 7 days**. **After you've read it, please let us know when you're planning to complete it.**

After you complete the assignment, we'll review it internally, and, if our evaluation is positive, we'll schedule a technical interview to discuss the results together.

## Tips

- You will have to make some assumptions as you're working on this challenge, so please include these clearly, indicate how the program should be run and any other comments you think the code reviewers will find helpful in the README.md file included with this assignment.
- As with anything in software, you need to strike a balance between extensibility/maintainability & time-spent (cost).
- It's ok if you want to deviate from the non-functional requirements stated below, but keep in mind the time you want to spend on this. If you do deviate, please add the reasons for these deviations in the README.md file included with this assignment.

## Evaluation Criteria

We will evaluate the assignment according to the following aspects:

- **Correctness:** bug free implementation according to the requirements;
- **Code quality:** how clean, readable, and organised your code is; does it have the right abstractions?

- **Code scalability:** how easy would it be to add more functionality without major rewrites
- **Maintainability:** how easy would it be for a new programmer to pick up the code and understand the domain concepts and rules
- **Testing:** how much confidence your automated tests add to the implementation.

## Expectations

- the solution should be written in OOP PHP
- no database connection is needed (e.g. use memory)
- the program should be able to run and see if requirements are implemented (up to you how you want to do that)
- keep it simple (e.g. no frameworks) considering the time constraints

## The assignment

Please implement a software solution that will enable various types of vehicle to park in a parking garage.

- When a vehicle arrives at the parking garage, and if there is space available, we should see a message on the screen saying "Welcome, please go in".
- When a vehicle arrives at the entrance to the parking garage, and it's full, we should see a message saying "Sorry, no spaces left".
- Must accept multiple types of vehicles: cars, vans, motorcycles
- Each type will have different size (e.g. 2 motorcycles can park in 1 car spot, a van takes 1.5 car spots).
- Must have 3 floors, each floor has different capacity.
- A Van can only park in the ground floor.

Next to this document will you find a ZIP archive with a Dockerfile to help you get started quickly.

Good luck!