

## Challenge description

You are required to write a web API with 2 endpoints:

1. A "Set timer" endpoint.
  - Receives a JSON containing hours, minutes, seconds and web url
  - This endpoint should return a JSON with a single field - "id"
  - This endpoint should define an internal timer, which shoots a webhook to the defined URL after the time ends
  - For example:  

```
POST /timers {hours: 4, minutes: 0, seconds: 1, url: "https://someserver.com"}
```

  
should return  

```
{id: 1}
```

  
After 4 hours and 1 second, server should call  

```
POST https://someserver.com/1
```
2. A "Get timer status" endpoint
  - Receives timer id
  - Returns a JSON with how long in seconds are left until the timer expires. If timer already expired, return "0"
  - For example:  

```
GET /timers/1
```

  
Should return  

```
{id: 1, time_left: 645}
```

Additional requirements:

- The timers should persist.
- If we shut down the process and restart it, timers should be saved.
- If a timer should have fired when the server was down, the server should fire the web hook after initializing.

Notes:

- Please use Python/ Javascript/ Typescript/ Java or C#
- Please add a readme with clear execution instructions (Solution will be tested in macos environment)
- If you're using another language, please package accordingly (Docker)
- Solution would also be measure by code structure and handling large scale of timers
- Feel free to add any extra functionality you wish
- You're welcome to approach me for any questions you may have.