



# HENNGE

## CHALLENGE DETAILS

---

### Introduction

Thank you for your application to HENNGE.

This challenge is intended for **frontend position** applicants.

This challenge consists of 3 missions. After successful completion of Mission 3, you will receive an email with a link to submit your CV and Cover Letter (CL). Once we receive them, our engineers will review your code and documents (CV & CL) altogether, and we will contact you via email with the screening result. If you haven't received your result within a month, please let us know!

**Tips:** Due to the competitiveness of our application process, we highly recommend applicants to explain their motivation clearly in the Cover Letter (CL). In addition, we highly encourage prospective applicants to finish the challenge within **2 weeks**. If some time has passed since the registration, the challenge might get updated and applicants might need to redo the challenge on the new version.

## Mission 1: Implement UI elements

### Description

Development of an `Email Audit` system is currently in progress. Assume that this project is a real project and is being used in a production environment on a trial basis, since the features are not yet complete.

You are tasked with implementing two UI elements to intelligently show email recipients in the `AuditTable` component.

Inside the resource provided below, you will find a `challenge_details` folder and folders for the different frameworks. The framework folders are identical projects implemented in their respective framework.

You may choose any one of the frameworks to implement the `RecipientsDisplay` component. The requirements are specified in `CHALLENGE.md` inside the `challenge_details` directory. Please note



The URL is protected by HTTP Basic Authentication, which is explained on Chapter 2 of RFC2617, so you have to provide an `Authorization: header` field in your POST request.

- For the `userid` of HTTP Basic Authentication, use the same email address you put in the JSON string
- For the `password`, provide a string ASCII value "HENNGECHALLENGE" (not including double quotations)

## Sample Request

```
POST /challenges/004 HTTP/1.1
Authorization: Basic bmluamFAZXhhbXBsZS5jb206SEV0TkdfQ0hBTExFTkdF
Host: api.challenge.henнге.com
Accept: */*
Content-Type: application/json
Content-Length: 136
```

```
{"contact_email":"ninja@example.com", "github_url":"https://gist.github.com/henнге/b8
```

## Sample Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Wed, 1 Jan 2021 00:00:00 GMT
Transfer-Encoding: chunked
```

```
{"message":"Congratulations! You have achieved mission 3"}
```

## Any questions? Found a bug?

Please note that our team will not answer to any inquiry related to the challenge itself.

If you have any (other) questions, please reach us through [recruit-engineer@henнге.com](mailto:recruit-engineer@henнге.com).

---

HENNGE © 2024 | [Contact us here](#) | [Privacy Policy](#)