



---

## Frontend Recruitment test

---

### Environment

- NodeJS 8+
- NPM 6+
- Webpack 4+
- VueJS
- ExpressJS

### Rules

- All JS must be written in ES6 flavoured Javascript
- Project startup procedure must be documented in a readme.MD file
- The project must be presented in public GitHub (or BitBucket, GitLab) repo owned by your regular account
- Each task must be done and added to the repository in a separate git commit
- Application presentation must be done by responsive web design principles
- Custom styling is nice to have, but we are ok with using something as Bootstrap for presentation
- VueJS single file components (<https://vuejs.org/v2/guide/single-file-components.html>) are nice to have
- Both applications (front and back) must be mounted on localhost with different ports
- Browser support is needed for modern browsers only
- Task 4 is optional, but it is nice to have

## Task 1 - New instance

Build a new VueJS instance with Webpack. Add router (<https://router.vuejs.org/>) to VueJS instance. Add navigation element to application.

## Task 2 - Home page

Create a home page with following elements:

- Title
- Image
- Generic text

## Task 3 - Contact us page

Create a contact form with the following inputs:

- E-mail
- Message
- Submit button

When a user submits the form and task 4 is resolved, its payload is sent as POST with JSON formatted payload to “/API/contact” and validation is done on ExpressJS side.

If you have opted-out of task 4, then client side validation as specified is enough:

- Must have a valid email address
- The message must be longer than 30 characters

Validation error or success event must be presented to the user.

## Task 4 [OPTIONAL] - New endpoint

Create ExpressJS instance as a separate application and add “/API/contact” endpoint with following payload validation rules:

- Must have a valid email address
- The message must be longer than 30 characters

This endpoint must confirm with standard HTTP response codes and have response payload as following:

- 200 - status OK
  - Response payload: JSON formatted message “Your message has been sent!”
- 422 - validation error
  - Response payload: JSON formatted message containing the list of errors