

Тестовое задание на позицию фронтенд разработчика (от 24 марта 2022)

(!) Общая информация

1. Требования к оформлению письма с выполненным тестовым заданием:
 - a. Разверните Вашу работу на любом хостинге (чтобы ее можно было посмотреть в браузере не разворачивая проект из исходного кода) и вышлите нам ссылку.
 - b. Укажите в письме список выполненных пунктов и выбранные вами инструменты. Если Вы сделали не все пункты тестового задания – пожалуйста, укажите причину по которой Вы их не выполнили (не хватило времени, не хватает опыта/знания, что-то ещё).
 - c. Обязательно приложите к письму максимально подробное и актуальное на дату отправки резюме.
 - d. Укажите ссылку на github/bitbucket репозиторий.
 - e. Укажите сколько часов было потрачено на выполнение задания.
 - f. Высылайте выполненное тестовое задание на электронный адрес **hr@abz.agency**.
2. Материалы данного тестового задания:
 - a. Все материалы, используемые в данном тестовом задании **были специально разработаны** сотрудниками компании abz.agency для проверки знаний соискателей в рамках тестового задания для данной вакансии (mockups, API документация, REST API сервер) и не имеют ничего общего с коммерческими проектами.
 - b. Ваши работы не будут использованы в коммерческих и (даже) некоммерческих целях.
 - c. Вы также **не имеете права** использовать материалы данного тестового задания, а также Ваше решение данного задания в коммерческих и некоммерческих целях (в том числе в портфолио, для составления курсов или в качестве тестового задания в другой компании).

Важно! Тестовое задание будет принято к рассмотрению только в случае если оно соблюдает требования, перечисленные выше.

(←) Тестовое задание

1. Работа с mockups (HTML/CSS). Тут вы найдете Figma project и исходники.
 - a. Только одна страница с преднамеренно минимальным набором стилей и компонентов, чтобы максимально сократить время выполнения тестового задания.
2. Работа с REST API (GET). Тут Вы найдете API документацию (OpenAPI).
 - a. Реализуйте блок "Working with a GET request" в соответствии с макетом и API документацией. Выводим по 6 юзеров на странице результата API запроса. Кнопка "Show more" должна быть скрыта, когда достигнута последняя страница результатов API запросов. Пользователи отсортированы по дате регистрации (новые первые).
 - b. Для отображения радиобаттонов на форме регистрации используйте GET /positions метод из API документации.
3. Работа с REST API (POST) – форма регистрации блок "Working with a POST Request"
 - a. Реализовать валидацию на фронт-енд части в соответствии с макетами и API документацией.
 - b. Реализовать бизнес логику формы регистрации в соответствии с макетами и API документацией.
 - c. После успешной регистрации обновить список пользователей в блоке "Working with a GET request". Если кнопка "Show more" была нажата (т.е. больше чем одна страница пользователей была загружена из API), то свернуть все и отобразить только первую страницу результата GET запроса. В результате новый пользователь будет выведен первым и Вы сможете проверить корректность работы формы без перезагрузки страницы.
4. Оптимизация вебсайта (бонусное задание). Минимизируйте и оптимизируйте css, js, изображения, и т.д. Для этого Вам необходимо развернуть Вашу работу на любом доступном Вам хостинге и направить на него любой доступный Вам домен.
 - a. Проверьте Вашу работу используя Google Page Speed и убедитесь что Ваша работа в зеленой зоне.
 - b. Проверьте Вашу работу используя Google Chrome Performance Audit / Lighthouse и убедитесь что Ваша работа в зеленой зоне для Performance, Best practices, SEO (mobile and desktop для 3G).
 - c. Проверьте Вашу работу используя Webpagetest и убедитесь что показатели близки к AAAAAA.

(Т) Технические требования

- React / Vue;
- Responsive / HTML5 / CSS3;
- CSS препроцессор (Sass/Less) или PostCSS;
- Организуйте Ваши CSS стили в читаемом виде (группируйте CSS по функциональности, пишите разъясняющие комментарии, разделяйте стили на несколько файлов по функциональности и т.д.);
- Обратите внимание на mockups и style guide при работе с блоком юзеров. Задание предполагает ваше внимание к деталям вывода контента. Такие детали как количество блоков, "...", tooltips, перенос строк и т.п.
- Разметка должна быть pixel-perfect;
- Latest browser;
- Самостоятельное тестирование в следующих браузерах (можно использовать BrowserStack):
 - Chrome, Firefox, Edge, Safari (Windows);
 - Chrome, Firefox, Safari (MacOS);
 - Chrome, Safari (iOS);
 - Chrome (Android).

Важно! Мы будем обращать внимание, насколько чистый у Вас подход к написанию CSS и Javascript-кода. Вы можете использовать любые сторонние библиотеки CSS и Javascript без каких-либо ограничений. Сторонние библиотеки CSS / Javascript должны быть подключены через менеджеры зависимостей npm / yarn (а не методом копи-паста). Сборка проекта должна быть реализована с помощью npm / webpack.