**я уже прошёл изучение этого курса, анкету для получение сертификата можно заполнить уже сейчас, поскольку планка в 70% обучения уже пройдена? Вопрс куратору курса  
ПОСТАНОВКА ЗАДАЧИ**

Известная компания, занимающаяся прокатом велосипедов в крупных городах России, испытывает проблемы с частой кражей их имущества (велосипедов). Как возможное решение проблемы, компания хочет вести учёт этих случаев и отслеживать прогресс. Их собственные разработчики уже подготовили серверную часть приложения, вам же требуется реализовать клиентскую часть.

**Клиентская часть** предназначена как для сотрудников компании, так и для обычных пользователей. Обычному пользователю доступна только ограниченная часть функционала: главная страница и страница с возможностью сообщить о новом случае кражи.

**ОПИСАНИЕ API**

**I. СУЩНОСТИ**

|  |  |  |  |
| --- | --- | --- | --- |
| Сущность**«Ответственный сотрудник»** | | | |
| **Название поля** | **Тип** | **Значение по умолчанию** | **Описание** |
| email | string |  | E-mail адрес сотрудника  **\*Обязательное и уникальное поле** |
| firstName | string |  | Имя сотрудника |
| lastName | string |  | Фамилия сотрудника |
| password | string |  | Пароль  **\*Обязательное поле** |
| clientId | string |  | clientId, уникальный для каждого студента  **\*Обязательное поле** |
| approved | boolean | true для первого пользователя, созданного с конкретным clientId. Для всех последующих — false. | Статус сотрудника: одобрен/не одобрен |

|  |  |  |  |
| --- | --- | --- | --- |
| Сущность**«Сообщение о краже»** | | | |
| **Название поля** | **Тип** | **Значение по умолчанию** | **Описание** |
| status | string | "new" при создании нового сообщения | Статус сообщения Возможные значения:   * + new   + in\_progress   + done   **\*Обязательное поле** |
| licenseNumber | string |  | Номер лицензии  **\*Обязательное поле** |
| type | string |  | Тип велосипеда Возможные значения:   * + general   + sport   **\*Обязательное поле** |
| ownerFullName | string |  | ФИО пользователя (арендатора велосипеда)  **\*Обязательное поле** |
| clientId | string |  | clientId, уникальный для каждого студента  **\*Обязательное поле** |
| createdAt | date | Текущая дата при создании сообщения | Дата создания сообщения  **\*Обязательное поле** |
| updatedAt | date | Текущая дата при обновлении сообщения | Дата последнего обновления сообщения |
| color | string |  | Цвет велосипеда |
| date | date |  | Дата кражи |
| officer | string |  | Ответственный сотрудник  Валидным значением может быть только действующий id ответственного сотрудника из базы |
| description | string |  | Дополнительный комментарий |
| resolution | string |  | Завершающий комментарий |

**II. ЗАПРОСЫ**

Все запросы работают с форматом JSON, поэтому не забывайте указывать заголовок Content-type: application/json.

|  |  |
| --- | --- |
| **POST /api/auth/sign\_up** | Запрос для создания новой учетной записи в системе. |
| **POST /api/auth/sign\_in** | Запрос для авторизации существующего пользователя.  В ответе на этот запрос в поле token вы получите токен, который вам нужно использовать для запросов, доступных только авторизованным пользователям. Токен будет действителен в течение 7 дней с даты отправки запроса.  **Подсказка:** рекомендуется сохранять этот токен в *local storage,* чтобы не проходить авторизацию постоянно. Токен необходимо передавать в каждом запросе, доступном только авторизованным пользователям в следующем заголовке: **Authorization: Bearer <token>.** |
| **GET /api/auth/** | Запрос для проверки действительности токена. В данном запросе не нужно указывать никакие данные, кроме токена в заголовке *Authorization.* |
| **POST /api/public/report** | Запрос для создания нового сообщения о краже. Этот запрос доступен без авторизации. Его нужно отправлять, когда сообщение о краже создаёт неавторизованный пользователь. |

**Важно!** Во всех следующих запросах clientId указывать НЕ НУЖНО. Они доступны только авторизованным пользователям. ClientId будет автоматически браться из токена ↓

|  |  |  |
| --- | --- | --- |
| **POST /api/cases** | | Запрос для создания нового сообщения о краже |
| **PUT /api/cases/:id** | Запрос для редактирования сообщения о краже, где :id — идентификатор редактируемого сообщения  **Примечание:** запись вида :id означает, что запрос имеет динамическую часть, в данном случае id. Двоеточие при запросе опускается, оно нужно только для обозначения динамической части запроса. | |
| **DELETE /api/cases/:id** | Запрос на удаление сообщения о краже | |
| **GET /api/cases** | Запрос на получение всех сообщений о краже | |
| **GET /api/cases/:id** | Запрос на получение информации по конкретному сообщению о краже | |
| **POST /api/officers** | Запрос для создания нового сотрудника | |
| **PUT /api/officers/:id** | Запрос на редактирование сотрудника по id (можно использовать, например, для функции «одобрить») | |
| **DELETE /api/officers/:id** | Запрос на удаление сотрудника | |
| **GET /api/officers** | Запрос для получения списка всех сотрудников | |
| **GET /api/officers/:id** | Запрос для получения информации по конкретному сотруднику | |

⚡ Полная документация к API бэкенд-части находится [по ссылке](https://documenter.getpostman.com/view/18055274/UVRAH6XZ). <https://documenter.getpostman.com/view/18055274/UVRAH6XZ>

**ФУНКЦИОНАЛЬНЫЕ ТРЕБОВАНИЯ**

**ГЛАВНАЯ СТРАНИЦА**

Главная страница должна содержать текстовое описание сервиса, возможно, картинки на ваш выбор. Данная страница доступна всем пользователям без авторизации.

**ФОРМА АВТОРИЗАЦИИ**

Форму авторизации можете расположить на главной странице, в шапке сайта или на отдельной странице — на ваше усмотрение. У авторизованных пользователей должна быть возможность выйти из учетной записи.

**СООБЩИТЬ О КРАЖЕ**

«Сообщить о краже» должна содержать форму для отправки информации об украденном велосипеде. Форма должна содержать следующие поля:

* + Номер лицензии (обязательное поле)
  + ФИО клиента (обязательное поле)
  + Тип велосипеда (выпадающий список с заранее определенными вариантами, обязательное поле)
  + Цвет велосипеда
  + Дата кражи
  + Дополнительная информация

Страница должна быть доступна всем пользователям без авторизации. Однако, если форму заполняет авторизованный сотрудник (например, если клиент сообщил о краже по телефону), ему доступно ещё одно дополнительное поле: ответственный сотрудник. Поле представляет собой выпадающий список с возможностью выбора из списка всех **одобренных** сотрудников, которые есть в базе.

**СТРАНИЦА РЕГИСТРАЦИИ**

На странице регистрации должна находиться форма регистрации со следующими полями:

* + E-mail (обязательное поле)
  + Пароль (обязательное поле)
  + Имя
  + Фамилия
  + Client ID (обязательное поле)

При отправке формы регистрации в базе данных создаётся новый сотрудник. Первый созданный сотрудник с конкретным client ID автоматически получит статус одобренного, остальных сотрудников нужно будет одобрить вручную.

**СООБЩЕНИЯ О КРАЖАХ**

Данная страница должна содержать список всех известных случаев краж (подсказка: это может быть таблица). Отображать служебные поля, например, clientId не нужно. Должна быть возможность удалить сообщение. При клике на одно сообщение из списка должна открываться его детальная страница.

**ДЕТАЛЬНАЯ СТРАНИЦА СООБЩЕНИЯ О КРАЖЕ**

Детальная страница сообщения должна содержать всю информацию о конкретном случае кражи с возможностью редактирования любого поля, кроме createdAt, updatedAt и clientId. Для полей, которые могут принимать значения из списка необходимо сделать поля соответствующих типов.

Помните, что в списке ответственных сотрудников должны отображаться только **одобренные** сотрудники.

Поле завершающего комментария (resolution) должно быть доступно только при выборе статуса «завершен», и в таком случае является обязательным. Т.е. нельзя изменить статус на «завершен», не заполнив поле resolution.

URL детальной страницы должен содержать id сообщения. Пример: localhost:3000/cases/12345 откроет страницу сообщения с id 12345.

**ОТВЕТСТВЕННЫЕ СОТРУДНИКИ**

Данная страница должна содержать список всех доступных сотрудников. Служебные поля (id, clientId, password) отображать не нужно. Должна быть возможность удалить сотрудника. При клике на одну запись из списка должна открываться детальная страница данного сотрудника.

**ДЕТАЛЬНАЯ СТРАНИЦА СОТРУДНИКА**

На этой странице должна содержаться детальная информация по сотруднику с возможностью редактирования. Нельзя редактировать поля email и clientId. Должна быть возможность одобрить сотрудника/снять одобрение (подсказка: для этого можно использовать тип поля *checkbox*).

URL детальной страницы должен содержать id сотрудника. Пример: localhost:3000/officers/12345 откроет страницу сотрудника с id 12345.

**ТРЕБОВАНИЯ К ИНТЕРФЕЙСУ**

В данном проекте нет готового макета и вам необходимо самостоятельно продумать пользовательский интерфейс. Все оформление: цветовая гамма, расположение элементов, шрифты — на ваше усмотрение. Оцениваться интерфейс будет по следующим критериям:

* + **Читабельность:** весь текст должен быть хорошо виден и читабелен. Размер шрифта — не менее 13 пикселей.
  + **Доступность:** все элементы должны быть доступны для взаимодействия. Не допускается перекрытие другими элементами, «уползание» за край экрана и т.д.
  + **Понятность:** пользователю должно быть однозначно понятно, за что отвечает тот или иной элемент интерфейса: кнопка, поле, выпадающий список и т.д.
  + **Адаптивность:** интерфейс должен отображаться корректно на любых размерах экрана.

⚡ Список ресурсов, которые помогут вам разработать классный интерфейс:

* 1. [Базовые принципы веб-дизайна](https://merehead.com/ru/blog/basic-principles-of-web-design/)
  2. [Советы для создания удобных форм ввода](https://habr.com/ru/post/417959/)
  3. [Статья «Четыре правила отличного UI»](https://www.uprock.ru/articles/4-pravila-dlya-otlichnogo-ui)
  4. [Для вдохновения: дизайны веб-сайтов, созданные опытными UX/UI дизайнерами](https://www.behance.net/search/projects?tracking_source=typeahead_search_direct&search=web%20site)

**ТРЕБОВАНИЯ К ВЕРСТКЕ**

* + Соблюдайте **семантическую вёрстку.** На каждой странице должны присутствовать разделы <header>, <main> и <footer>, а также заголовок <h1>. Кнопки должны быть реализованы элементом <button>, элементы дропдауна — списком <select> и так далее.
  + Если какой-либо элемент доступен для взаимодействия (ссылка, кнопка), при наведении курсора должен появляться cursor: pointer. Внешний вид самого элемента тоже должен меняться при наведении курсора. Пример: нижнее подчеркивание текста у ссылки, другой цвет фона у кнопки и т.д.
  + Во всех формах сайта присутствуют обязательные поля, поэтому необходимо обязательно продумать валидацию форм. Если пользователь пытается отправить форму с незаполненными обязательными полями, он должен увидеть сообщение об ошибке.

**ТРЕБОВАНИЯ К REACT**

* + Интерфейс должен быть поделен на компоненты. Перед началом работы хорошенько обдумайте, какие компоненты вы будете использовать. Деление на компоненты должно быть логичным и оправданным.
  + В проекте будет использоваться довольно много данных, поэтому рекомендуется использовать более продвинутый инструмент хранения и изменения данных, чем обычный state, например useReducer или Redux.
  + При написании кода старайтесь следовать принципам KISS (Keep It Short and Simple — не усложняй) и DRY (Don’t Repeat Yourself — не повторяйся).

В остальном техническая сторона реализации проекта — полностью на ваше усмотрение. Можете использовать любые инструменты и дополнительные библиотеки, какие посчитаете нужными, но старайтесь следить за тем, чтобы их применение было оправдано и не усложняло код без необходимости.

**ПРОЧИЕ ТРЕБОВАНИЯ**

* + Пишите код аккуратно, с соблюдением форматирования и отступов.
  + Старайтесь давать компонентам, переменным и функциям осмысленные имена.
  + Старайтесь использовать современный *ES6* синтаксис: стрелочные функции, декомпозиция, *spread* и т.д.
  + При размещении проекта на *GitHub* не забывайте добавить папку *node\_modules* в файл *.gitignore*, чтобы она не попала в ваш репозиторий. О том, как настроить *.gitignore* и почему папки *node\_modules* не должно быть в репозитории, можете прочитать в [этой статье](https://cpab.ru/chto-takoe-fajl-gitignore-i-kak-ego-nastroit-cloudsavvy-it/).

**⭐ НАЧАЛО РАБОТЫ**

Рекомендуется начать работу с визуальной части, т.е. продумать и сверстать интерфейс, а только потом подключать бэкенд.

Последовательность действий при подключении фронтенд-части к бэкенду:

* 1. Запросить у ментора персональный clientId.
  2. Через POST /api/auth/sign\_up создать учетную запись с вашим clientId.
  3. Авторизоваться через POST /api/auth/sign\_in под своей учетной записью.
  4. Получить token и уже с ним делать все авторизованные запросы.

При возникновении сложностей вы можете задать свои вопросы в канале **#финальный\_проект** или в личных сообщениях ментору.

**УДАЧИ!**

**Вебинар «Разбор финального проекта»**

**Спикер:**Юлия Токаревская, ментор курса

После просмотра записи вебинара вам предоставляется возможность оценить его и поделиться своими впечатлениями/комментариями в [**форме обратной связи**](https://forms.gle/f1PzmskpuYvasvm26). Для нас это очень важно. Просим оставить детальный комментарий. Это поможет нам скорректировать/улучшить планы и форматы вебинаров, что позволит сделать ваш процесс обучения более эффективным и интересным.

<https://lms.skillfactory.ru/courses/course-v1:Skillfactory+FR+2020/courseware/94353298670444a7aca29c2471453233/0ca88498b4c64358b9574d364d221485/1?activate_block_id=block-v1%3ASkillfactory%2BFR%2B2020%2Btype%40vertical%2Bblock%407851c84d1338404c8dad8c3fff9d7e15>

<https://docs.google.com/presentation/d/1aFyLRD0qmc5YiUiBX-l6l-5xVg4iGS3-AbjkIZ7M3-o/edit#slide=id.g107e1d84739_0_101>

Финальный опрос SkillFactory

Заполни и отправил 21 ноября 2022 г.

**larisa.v.n1957@gmail.com** (без совместного доступа) [Сменить аккаунт](https://accounts.google.com/AccountChooser?continue=https://docs.google.com/forms/d/e/1FAIpQLSdBaxSyDj_7CgLxob19lRQl4ofegYAj6uQjxzuo8Rb3r3XilQ/viewform?usp%3Dsend_form&service=wise)

Черновик сохранен.

\* Обязательно

Укажите, пожалуйста, данные для составления и отправки сертификата

Мы проверим ваш прогресс на учебной платформе и вскоре вышлем письмо с сертификатом на указанный адрес.  
Срок подготовки сертификата - до 2 рабочих дней включительно.

Фамилия и имя\*



Наливайко Сергей

Email, на который прислать сертификат\*

Sergey\_p\_i@mail.ru



Название курса на русском языке:\*

Профессия "Frontend-разработчик PRO" (FRPRO)

Профессия фронтенд-разработчик Pro(frpro)

Фамилия и имя на латинице, только в случае, если требуется сертификат на английском языке

Nalivaiko Sergei



Название курса на английском языке:

Frontend Developer Specialization

Номер вашего потока\*

Напишите в формате "аббревиатура курса-№", например, DST-11 или QAP-7. В случае, если вы проходили курс «Мастер Google Таблиц‎» или «‎Курс-тренажер по SQL», укажите "0" в данном поле.

FR-108



Ник на kaggle (если в курсе есть хакатоны)

Я не принимал участия в форумах разработчиков и ника у меня нет



Поговорим о результатах обучения

Вспомните, с какой целью вы приступали к обучению на курсе и чего достигли в итоге.

Вспомните, почему вы решили обучаться на данном курсе? Какова была ваша цель?\*

Получить навыки в данной сфере деятельности, сменить профессию и найти новую работу



Достигли ли вы цели обучения? Если нет, то почему?\*

Навыки получил, даже больше, чем ожидал. Прорекламировал свои навыки руководителям организации. Сейчас в режиме ожидания ответа, но без дела не нахожусь.



Как вы планируете применять полученные навыки?\*

С удовольствием! Хочу попробовать объединить полученные навыки во фронденд-разработке с фотографией, рекламой и интерьером жилых, офисных помещений.



А теперь о самом курсе

Помогите нам усовершенствовать учебные материалы, поддержку в рамках курса и процесс обучения в целом

Столкнулись ли вы с какими-либо трудностями во время прохождения курса? Если да, то с какими?\*

В целом курс мне очень понравился, мастерство изложения материалов, чувствуется профессиональная хватка. Был нюанс в одном из модулей, касающийся работы с немного устаревшим проектом, но благодаря поддержке ментеров курса и куратору, вопрос был решён. Курс отличный.



Какие элементы курса вам понравились?\*

Описание материала, видео, ссылки на необходимые ресурсы(очень пригодились). Особенно хочу отметить, понравившиеся мне начальные модули по изучению css. Модули javascript: тестеровка кода, написание функций, работа с таймаутами.В модуле 12, интерес вызвал sass, использование переменных в css коде, 19 модуль React и все модули, связанные с ним, тут вообще без обсуждений, всё супер. 13, 14 модули Webpac и работа с ним. Свойства css @keyframes и др. В общем все модули интересны. Курс сформирован очень здорово.



А что мы можем улучшить?\*

Чуть больше дать информации о Сss коде



Оставьте пожалуйста отзыв о курсе\*

Отличный, полезный, нужный курс. Сам прошёл, теперь знаю что к чему, буду приобщать к учёбе в этом направлении своего ребёнка.



Согласны ли вы разместить отзыв на внешнем ресурсе и получить скидку 50% на следующую покупку? (не суммируется с другими акциями)\*

В случае положительного ответа мы свяжемся с вами для дополнительных инструкций

Я согласен разместить этот отзыв на внешнем ресурсе.

Пришло время оценок!

Насколько вы согласны со следующими утверждениями?

Формулировка заданий была понятной.\*

1

2

3

4

5

6

7

8

9

10

Задания усложнялись постепенно.\*

1

2

3

4

5

6

7

8

9

10

Задания помогали закрепить полученные навыки.\*

1

2

3

4

5

6

7

8

9

10

Ответы экспертов по учебным материалам помогали мне решить вопрос.\*

1

2

3

4

5

6

7

8

9

10

Поддержка была оперативной.\*

1

2

3

4

5

6

7

8

9

10

Насколько вы удовлетворены поддержкой обучения на курсе?\*

1

2

3

4

5

6

7

8

9

10

Насколько вы в целом удовлетворены курсом?\*

1

2

3

4

5

6

7

8

9

10

Насколько вы готовы рекомендовать курс знакомым и друзьям?\*

1

2

3

4

5

6

7

8

9

10

Отправить

Очистить форму

<https://docs.google.com/forms/u/0/d/e/1FAIpQLSdBaxSyDj_7CgLxob19lRQl4ofegYAj6uQjxzuo8Rb3r3XilQ/formResponse>

Ссылка в отзовике на мой отзыв

<https://otzovik.com/myreviews.php>

https://otzovik.com/edit\_rview.php?pid=14018604