Bazar

# Краткое описание

Проект Базар - проект для продажи и обмена товарами (лотами).

Проект содержит функциональность регистрации и авторизации, просмотра, создания, редактирования, удаления лота, создании заявки на лот от других пользователей.

Проект Базар содержит бекенд на Java и фронтенд на ReactJS и SQL базу данных

# Обозначения

1. Символ \* в описании полей для ввода данных, означает, что данное поле является обязательным для заполнения, при его отсутствии поле считается опциональным

# Регистрация пользователя

## Форма регистрации

Поля для ввода данных

* \* Username - Идентификатор пользователя в системе
* \* Password - Пароль
* \* Repeat password - Повтор пароля
* \* Email - электронный адрес для отправки уведомлений

Кнопка отправки формы Sign Up

## Описание работы

### Пример запроса

POST <http://bazar.jettycloud.com:8080/signup>

Body: ​​{"username":"Usr","password":"1234","email":"usr@tmp.org"}

### Клиентская часть

После отправки данных на сервер возможны два случая

1. Пользователь зарегистрирован и авторизован
2. Дублирующийся логин, выводится ошибка

### Серверная часть

Происходит поиск пользователя в системе по Login

Если пользователь с таким логином не найден:

1. Создается запись в таблице user с соответствующими значениями полей
2. Пароль сохраняется в зашифрованном виде
3. Происходит авторизация и редирект на страницу нотификаций: http://bazar.jettycloud.com:8080/notifications

Если пользователь с таким логином уже существует выбрасывается ошибка

# Авторизация пользователя

Авторизация доступна только для зарегистрированных пользователей. Для идентификации авторизованного пользователя используется Authorization header.

## Форма авторизации

Поля для ввода данных

* \* Login - Идентификатор пользователя в системе
* \* Password - Пароль

Кнопка отправки формы Sign In

## Описание работы

### Пример запроса

POST <http://bazar.jettycloud.com:8080/authenticate>

Body: ​​{"username":"Usr","password":"1234"}

### Клиентская часть

После отправки данных на сервер возможны два случая

1. Пользователь найден в системе, осуществляется переход на страницу со списком уведомлений
2. Пользователь не найден в системе, выводится ошибка

### Серверная часть

Происходит поиск пользователя в системе по Login

1. Пользователь с таким логином не найден, выбрасывается ошибка
2. Пользователь с таким логином найден, осуществляется проверка пароля

Если пароли совпадают, то возвращается token (JWT токен) используемый в дальнейшем для авторизованных запросов от пользователя, в противном случае, выбрасывается ошибка

Пример заголовка с токеном:

Authorization: Bearer eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJVc3JFVCIsImV4cCI6MTY3MTY2MzY0NiwiaWF0IjoxNjcxNjI3NjQ2fQ.fYe0ECZruSsc2BC497ol4iEeTx065t4rvIKF-xIL9m8

# Страница уведомлений

Авторизованный в системе пользователь, может видеть список своих уведомлений на странице: GET <http://bazar.jettycloud.com:8080/notifications>

В списке уведомлений должны отображаться события вызванные другими пользователями, например:

* Подача заявки на лот текущего пользователя
* Изменение статуса заявки текущего пользователя

Должна быть возможность быстрого перехода на форму лота при нажатии на уведомление.

# Создание лота

Авторизованный в системе пользователь, может создавать лоты для выставления на продажу. Лот может быть создан в статусе “Новый” путём нажатия кнопки “Сохранить”..

## Форма создания

Поля для ввода данных

* \* Name - Название лота
* Description - Описание
* Price - Цена (0$ при отсутствии)
* Форма загрузки фото

Кнопка отправки формы Save

Путь к форму: <http://bazar.jettycloud.com/lots-add>

## Описание работы

### Пример запроса

POST <http://bazar.jettycloud.com:8080/lot>

Body: ​​{"name":"MyLot","price":10000,"description":"Some description","files":[27]}

### Клиентская часть

После отправки данных на сервер происходит сохранение лота в системе и осуществляется редирект на страницу лота.

GET <http://bazar.jettycloud.com/lots/28>

### Серверная часть

В лоты пользователя добавляется новый лот в статусе “Новый” с ссылками на файлы с фотографиями лота.

## Загрузка фото

Осуществляется при нажатии на кнопку Choose files, после загрузки, фото присваивается id, который будет сохранен при сохранении лота.

Работа с файлами осуществляется по пути: <http://bazar.jettycloud.com:8080/file>

Для получения файла можно обратиться по пути выше с указанием id: GET <http://bazar.jettycloud.com:8080/file/57>

# Список лотов

В списке лотов должны отображаться все доступные лоты (не проданные).

Должна быть возможность посмотреть свои лоты, в случае своих лотов, должны отображаться в том числе и проданные с пометкой об этом.

Путь к форме: <http://bazar.jettycloud.com/lots>

### Отображаемая информация

* Имя пользователя
* Название лота
* Цена

При нажатии на конкретный лот, должен быть переход на страницу лота.

# Страница лота

На странице должна отображаться вся информация по лоту, а также по поданным на неё заявкам и функционал для работы с ними.

Путь к форме: <http://bazar.jettycloud.com/lots/2>

Форма должна отличаться для владельца и других пользователей:

* У владельца лота должны быть возможность видеть список заявок, принять одну заявку, остальные заявки должны быть автоматически отклонены
* У других пользователей должна быть возможность подать заявку и видеть её статус  
  POST <http://bazar.jettycloud.com:8080/offer>   
  Body: {"lotId":"5"}

После продажи на странице лота должно отображаться, что он продан и лот должен пропадать из общего списка.