

Личный проект «Большое путешествие»

Техническое задание

О проекте

«Большое путешествие» — современный сервис для настоящих путешественников. Сервис помогает детально спланировать маршрут поездки, рассчитать стоимость путешествия и получить информацию о достопримечательностях. Минималистичный интерфейс не даст повода отвлечься и сфокусирует внимание на планировании путешествия.

1. Описание функциональности

Приложение состоит из двух экранов: «Маршрут поездки» и «Статистика».

Обратите внимание, оживление наименования маршрута, дат и общей стоимости путешествия — часть дополнительного задания. Если вы не планируете его выполнять, удалите этот блок из разметки в вашем проекте.

1.1 Общий контейнер

- Новая точка маршрута создаётся нажатием на кнопку «New Event». После нажатия пользователь видит форму создания новой точки маршрута, где ему необходимо

заполнить несколько полей (о полях будет рассказано позже).

1.2 Маршрут поездки

- Путешествие строится из отдельных точек маршрута. Точка маршрута — отдельное событие в путешествии. Точкой маршрута может быть остановка в каком-нибудь месте (Check, Sightseeing, Restaurant) или поездка на транспортном средстве (Taxi, Bus, Train, Ship, Transport, Drive и Flight).
- Точка маршрута принадлежит к одному из типов: Taxi, Bus, Train, Ship, Transport, Drive, Flight, Check, Sightseeing, Restaurant.
- Точки маршрута загружаются с сервера при старте приложения.
- В случае отсутствия точек маршрута вместо списка отображается текст: «Click New Event to create your first point».

1.3 Точка маршрута

- По умолчанию точка маршрута в списке отображается в режиме просмотра.
- В колонке «Time» отображается время и продолжительность нахождения в точке маршрута (разность между окончанием и началом события). Время маршрута отображается в формате «Начало» —

«Окончание» (10:30—11:00). Формат продолжительности нахождения в точке маршрута зависит от длительности:

- Менее часа: минуты (например «23M»);
- Менее суток: часы минуты (например «02H 44M»);
- Более суток: дни часы минуты (например «01D 02H 30M»);
- В колонке «Offers» отображаются не более 3-х дополнительных опций, применённых к точке маршрута. Остальные опции пользователь может посмотреть открыв карточку точки маршрута.

1.4 Новая точка маршрута

- Новая точка маршрута создаётся нажатием на кнопку «New Event». Форма создания новой точки маршрута появляется в самом начале списка. Кнопка при этом блокируется на время создания новой точки.
- Если в момент нажатия на кнопку «New Event» был выбран фильтр или применена сортировка, то они сбрасываются на состояния «Everything» и по умолчанию соответственно.
- Нажатие кнопки «Esc» скрывает форму создания новой точки маршрута. Несохранившиеся изменения пропадают.
- Нажатие на кнопку «New Event» закрывает открытую форму редактирования точки маршрута, если таковая

была. Изменения при закрытии не сохраняются.

- В форме создания новой точки маршрута пользователю доступны следующие элементы управления:
 - Кнопка «Save». Сохранение точки маршрута и закрытие формы создания точки маршрута.
 - Кнопка «Cancel». Отмена изменений и закрытие формы создания точки маршрута.
- Для создания новой точки маршрута пользователь заполняет:
 - Тип точки маршрута (один из: Taxi, Bus, Train, Ship, Transport, Drive, Flight, Check-in, Sightseeing, Restaurant.).
 - Пункт назначения. Выбирается из списка предложенных значений, полученных с сервера. Пользователь не может ввести свой вариант для пункта назначения.
 - Дата и время начала события. Выбор времени и даты осуществляется с помощью библиотеки flatpickr.js. Выбранная дата и время отображаются в поле в формате: день/месяц/год часы:минуты (например «25/12/2019 16:00»).
 - Дата и время окончания события. Формат и требования аналогичны дате начала. Дата окончания не может быть меньше даты начала события.
 - Стоимость. Целое число.
 - Дополнительные опции. В зависимости от типа точки маршрута пользователь может выбрать дополнительные опции (offers).

- Информация о дополнительных опциях загружается с сервера.
- Дополнительные опции отображаются после выбора (изменения) типа точки маршрута.
- Пользователь может отметить одну или несколько дополнительных опций.
- Дополнительная опция применяется к точке маршрута один раз. Повторный клик на дополнительную опцию отменяет её применение.
- Список дополнительных опций доступен не для всех типов точек маршрута. Для некоторых типов точек дополнительные опции могут отсутствовать. В этом случае контейнер для вывода дополнительных опций не отображается.
- Выбор дополнительных опций влияет на общую стоимость путешествия. Стоимость точки маршрута, которую ввёл пользователь в соответствующее поле ввода, при этом не изменяется.
- При изменении типа точки маршрута выбранный ранее список дополнительных опций очищается.
- При выборе типа точки маршрута должен обновляться вспомогательный текст (плейсхолдер). Текст плейсхолдера зависит от типа точки:

- Для точек регистрирующих перемещение пользователя на транспортном средстве (Flight, Taxi и т. д.): <action> to. Например: Flight to, Taxi to и т. д.
- Для точек регистрирующих прибытие в определённое место (Sightseeing, Restaurant, Check-in): <action> in. Например: Sightseeing in, Restaurant in и т. д.
- После выбора пункта назначения появляется блок «Destination». В нём отображается информация о месте назначения (изображения, текст). Информация для блока «Destination» загружается с сервера.
- Если информация о пункте назначения отсутствует, блок «Destination» не отображается. Информация в блоке «Destination» всегда соответствует выбранному пункту назначения. При изменении пункта назначения информация в блоке «Destination» обновляется.
- После сохранения точка маршрута располагается в списке точек маршрута в порядке определённом текущей сортировкой (по умолчанию, по длительности или по стоимости).
- Введённые пользователем данные экранируются.

1.5 Редактирование точки маршрута

- Для перехода к форме редактирования точки маршрута пользователь кликает по кнопке с изображением «Стрелка вниз» в правом углу карточки точки маршрута.
- В форме редактирования пользователю доступны следующие элементы управления:
 - Кнопка «Save». Сохраняет изменения, закрывает форму редактирования.
 - Кнопка «Delete». Удаляет точку маршрута, закрывает форму редактирования.
 - Кнопка «Favorite» (отображается в виде звёздочки). Добавляет точку маршрута в избранное. После добавления в избранное, кнопка меняет состояние — закрашенная звезда. Повторный клик по кнопке удаляет точку маршрута из избранного и возвращает кнопку в исходное состояние. Добавление/удаление в избранное происходит сразу и не зависит от нажатия кнопок «Save».
- В форме редактирования представлены все поля, которые пользователь заполняет при создании новой точки маршрута (см. раздел «Новая точка маршрута»). Правила их поведения сохраняются.
- При нажатии кнопки «Esc» форма редактирования закрывается. Несохраниённые изменения пропадают.
- Одновременно может быть открыта только одна форма создания/редактирования.

- Если пользователь внёс изменения в точку маршрута, не выполнил сохранение и пытается перейти к редактированию другой точки маршрута либо создать новую, то в этом случае открытая форма редактирования скрывается без сохранения изменений. После открывается другая форма редактирования или форма создания новой точки маршрута.
- После сохранения точка маршрута располагается в списке точек маршрута в порядке определённом текущей сортировкой (по умолчанию, по длительности или по стоимости).

1.6 Фильтры

- В приложении предусмотрено несколько фильтров:
 - Everything — полный список точек маршрута;
 - Future — список запланированных точек маршрута, т. е. тех точек, у которых дата начала больше, чем текущая;
 - Past — список пройденных точек маршрута, т. е. точек у которых дата окончания маршрута меньше, чем текущая.
- Выбор любого из фильтров приводит к отрисовке только тех точек маршрута, которые соответствуют выбранному фильтру. При смене фильтра разбивка по дням сохраняется.

1.7 Сортировка

- Пользователю доступна возможность сортировки точек маршрута по стоимости (клик по заголовку колонки «Price») и длительности (клик по заголовку колонки «Time»). Сортировка работает в одном направлении — от максимального к минимальному: при сортировке по стоимости в начале списка окажутся самые дорогие точки маршрута, при сортировке по длительности — самые долгие. В режиме сортировки точки маршрута не разбиваются по дням.
- Для отмены сортировки и возвращению к исходному порядку (разбивка по дням) пользователь кликает по заголовку колонки «Event».

1.8 Статистика

- Статистика «Stats» позволяет пользователю проанализировать расходы и время, затраченное на путешествие. Для удобства просмотра и анализа эта информация представлена в виде нескольких диаграмм: «Money» (финансовые расходы по каждому типу точек маршрута), «Transport» (количество использований того или иного транспорта) и «Time-spend» (затраченное время относительно типа точки маршрута).
- Для просмотра статистики пользователь нажимает на кнопку «Stats».
- Диаграмма «Money» показывает финансовые затраты в разрезе точек маршрута. Она позволяет ответить на вопрос: «Сколько за время путешествия было потрачено

на такси или рестораны?». Диаграмма «Money» не учитывает дополнительные опции, только стоимость точек маршрута.

- Диаграмма «Transport» показывает статистику по транспортным средствам, которыми придётся воспользоваться в путешествии. Диаграмма отвечает на вопрос: «Сколько раз нам придётся воспользоваться самолётом, такси и так далее».
- Диаграмма: «Time-Spend» показывает сколько времени было затрачено на отдельные точки маршрута. Диаграмма позволяет ответить на вопрос: «Сколько дней пользователь проведёт в самолёте, такси, ресторане и так далее».

1.9 Взаимодействие с сервером

- Сервер расположен по адресу `https://11.ecmascript.pages.academy/big-trip/`.
- Каждый запрос к серверу должен содержать заголовок `Authorization` со значением `Basic $ {случайная строка}`. Например, `Basic er883jdzbdw`. Случайная строка формируется однократно при старте приложения.

1.9.1 Ресурсы

Структуры данных:

- Destination

{

```
  "description": "Chamonix, is a beautiful city, a  
true asian pearl, with crowded streets.",  
  "name": "Chamonix",  
  "pictures": [  
    {  
      "src": "http://picsum.photos/300/200?  
r=0.0762563005163317",  
      "description": "Chamonix parliament  
building"  
    }  
  ]  
}
```

- Offer:

```
{  
  "type": "taxi",  
  "offers": [  
    {  
      "title": "Upgrade to a business class",  
      "price": 120  
    }, {  
      "title": "Choose the radio station",  
      "price": 60  
    }  
  ]  
}
```

- Offer.type — одно из следующих значений:

```
["taxi", "bus", "train", "ship", "transport",  
"drive", "flight", "check-in", "sightseeing",  
"restaurant"]
```

- Point:

```
{  
  "base_price": 1100,  
  "date_from": "2019-07-10T22:55:56.845Z",
```

```

    "date_to": "2019-07-11T11:22:13.375Z",
    "destination": $Destination$,
    "id": "0",
    "is_favorite": false,
    "offers": [
      {
        "title": "Choose meal",
        "price": 180
      }, {
        "title": "Upgrade to comfort class",
        "price": 50
      }
    ],
    "type": "bus"
  }
}

```

- Point.type — одно из следующих значений:

```

["taxi", "bus", "train", "ship", "transport",
"drive", "flight", "check-in", "sightseeing",
"restaurant"]

```

- LocalPoint:

```

{
  "base_price": 222,
  "date_from": "2019-07-10T22:55:56.845Z",
  "date_to": "2019-07-11T11:22:13.375Z",
  "destination": $Destination$,
  "is_favorite": false,
  "offers": [
    {
      "title": "Choose meal",
      "price": 180
    }, {
      "title": "Upgrade to comfort class",
      "price": 50
    }
  ]
}

```

```
],  
  "type": "taxi"  
}  
•      AuthorizationError:  
  
{  
  "error": 401,  
  "message": "Header Authorization is not correct"  
}  
•      NotFoundError:  
  
{  
  "error": 404,  
  "message": "Not found"  
}
```

Точки маршрута /points

GET /points

- Код ответов:
 - 200 OK;
 - 401 Unauthorized.

Пример:

- Request:
 - URL: GET /points;
 - Headers: Authorization: Basic kTy9gldsz2317rD.
- Response:
 - Status: 200 OK;
 - Body: массив из элементов вида Point.
- Request:
 - URL: GET /points.
- Response:

- Status: 401 Unauthorized;
- Body: структура `AuthorizationError`.

POST /points

Создание новой точки маршрута.

- Код ответов:
 - 200 OK;
 - 401 Unauthorized.

Пример:

- Request:
 - URL: POST /points;
 - Headers: Authorization: Basic kTy9glDsz2317rD;
 - Body: структура `LocalPoint`.
- Response:
 - Status: 200 OK;
 - Body: структура `Point`.

PUT /points/: id

Редактирование точки маршрута.

- Код ответов:
 - 200 OK;
 - 401 Unauthorized;
 - 404 Not found.

Пример:

- Request:
 - URL: PUT points/31;

- Headers: Authorization: Basic kTy9gldsz2317rD;
- Body: структура Point.
- Response:
 - Status: 200 OK;
 - Body: структура Point.

DELETE /points/: id

Удаление точки маршрута.

- Код ответов:
 - 200 OK;
 - 401 Unauthorized;
 - 404 Not found.

Пример:

- Request:
 - URL: DELETE points/31;
 - Headers: Authorization: Basic kTy9gldsz2317rD.
- Response:
 - Status: 200 OK;
 - Body: текст OK.

Пункты назначения /destinations

GET /destinations

Получение доступных точек маршрута.

- Код ответов:
 - 200 OK;
 - 401 Unauthorized.

Пример:

- Request:
 - URL: GET /destinations;
 - Headers: Authorization: Basic kTy9glpsz2317rD.
- Response:
 - Status: 200 OK;
 - Body: массив из элементов вида Destination.

Общий список дополнительных предложений / offers

GET /offers

- Код ответов:
 - 200 OK;
 - 401 Unauthorized.

Пример:

- Request:
 - URL: GET /offers;
 - Headers: Authorization: Basic kTy9glpsz2317rD.
- Response:
 - Status: 200 OK;
 - Body: массив из элементов вида Offer.

Синхронизация с сервером /points/sync

Этот метод потребуется для реализации дополнительного задания лекции 9.

Пример:

- Request:
 - URL: POST /points/sync;
 - Headers: Authorization: Basic kTy9glpsz2317rD;
 - Body: [\$Point\$, \$Point\$].
- Response:
 - Status: 200 OK;
 - Body:

```
{
  "created": [$Point$, $Point$],
  "updated": [$Point$, $Point$],
  "deleted": [$Point$, $Point$]
}
```

1.10 Обратная связь интерфейса

- На время загрузки вместо маршрута нужно вывести сообщение «Loading...».
- При нажатии на кнопки «Save», «Delete» формируется запрос к серверу на добавление/изменение/удаление данных. На время выполнения запроса текст заголовка кнопки изменяется:
 - «Save» -> «Saving...»
 - «Delete» -> «Deleting...»
- Заголовки кнопок возвращаются в исходный вид после завершения запроса.
- В момент отправки запроса на создание/удаление/изменение данных по точке маршрута форма блокируется от внесения изменений. Разблокировка формы происходит после завершения выполнения запроса (неважно, успешно

выполнен запрос или нет).

- Обновление элементов (удаление точки маршрута, обновление точки маршрута и т.д) в DOM происходит после успешного выполнения запроса к серверу.
- Если запрос не удалось выполнить (сервер недоступен, произошла ошибка), форма редактирования/создания остаётся открытой, к ней применяется эффект «покачивание головой».

2. Дополнительные задания

- Наименование маршрута путешествия формируется автоматически и состоит из пунктов назначения (названий городов), разделённых тире: «Amsterdam — Geneva — Chamonix». Если городов больше 3-х, то в наименовании маршрута отображается первый и последний город, разделённые многоточием: «Amsterdam — ... — Chamonix».
- Даты путешествия в шапке заполняются автоматически. Дата начала всего путешествия соответствует дате начала первой точки маршрута. Дата окончания — дате завершения последней точки маршрута. Например, «18 AUG—6 OCT».
- Общая стоимость путешествия пересчитывается при создании/изменении/удалении любой точки маршрута.

- Кнопка фильтра становится недоступной, если фильтру не соответствует ни одна из точек маршрута.
- Offline режим. Реализуйте в приложении поддержку оффлайн-режима. Приложение должно иметь возможность запускаться без интернета и сохранять пользовательский функционал. При появлении интернета все изменения должны отправляться на сервер.

3. Разное

- В зависимости от состояния, некоторым элементам управления применяются соответствующие классы оформления. Например, активный фильтр и так далее. Примеры доступны в директории с вёрсткой (markup).