

React. Разработка сложных клиентских приложений

Уровень 3, с 27 сентября по 28 ноября 2021

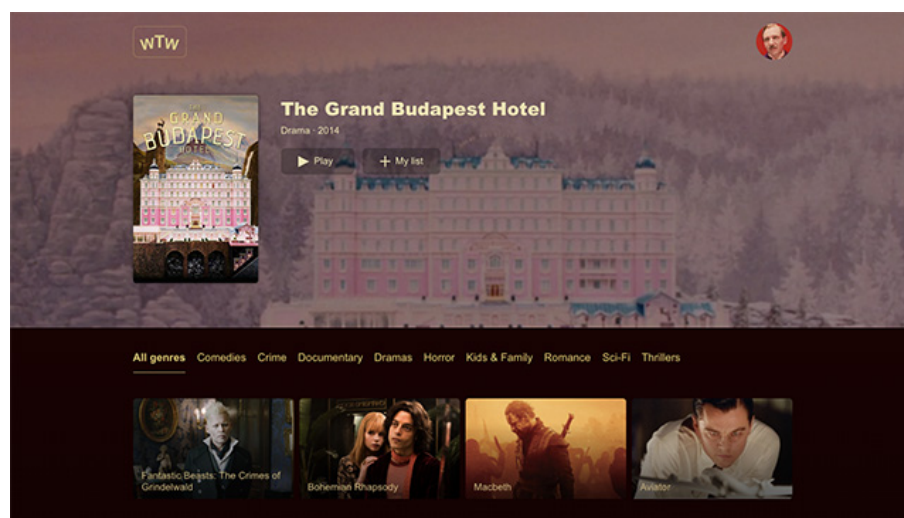


Меню курса



[Главная](#) / [Проекты](#) /

Проект «Что посмотреть»



— [Техническое задание](#)

Техническое задание

О проекте

«Что посмотреть» — онлайн кинотеатр нового поколения. Смотрите новинки абсолютно бесплатно и в лучшем качестве. Оставляйте отзывы, ставьте оценки и выбирайте только лучшее из мира большого кино.

Описание функциональности

Страницы приложения

Приложение состоит из нескольких страниц: `Main (/)`, `Sign In (/login)`, `MyList (/mylist)`, `Film (/films/:id)`, `Add review (/films/:id/review)`, `Player (/player/:id)`.

Страницы `MyList`, `Add review` доступны только авторизованным пользователям. Если пользователь не авторизован, то при переходе к этим страницам выполняется перенаправление на страницу `Sign In`.

Если пользователь не авторизован, то при попытке перехода к приватной странице выполняется перенаправление на страницу «Sign In» (`/login`).



В правом углу шапки отображается аватар пользователя и кнопка «Sign Out» (если пользователь авторизован) или ссылка «Sign In» (если пользователь не авторизован).

Клик по кнопке «Sign Out» приводит к завершению сеанса работы — выходу из закрытой части приложения.

Клик по аватарке пользователя выполняет переход на страницу `MyList` (`/mylist`).

Обращение к несуществующей странице (например, через адресную строку) не приводит к появлению ошибок в приложении, а корректно обрабатывается маршрутизацией.

Пользователь перенаправляется на страницу «404». Дизайн страницы остаётся на усмотрение студента. В самом простом случае это может быть страница с текстом `404 Not Found` и ссылкой для перехода на главную страницу приложения.

Главная страница

На главной странице представлены жанры, превью фильмов.

Страница с детальным описанием фильма доступна всем пользователям.

В шапке страницы отображается постер и обложка промо-фильма.

Промо-фильм можно сразу посмотреть, нажав кнопку «Play» или добавить в список «My List».

Получение промо-фильма для главной страницы выполняется отдельным запросом к серверу (см. «Маршруты»).

После загрузки приложения отображаются 8 карточек фильмов произвольных жанров. В списке жанров выделен «All genres».

Список жанров строится на основании полученной с сервера информации о фильмах.

В списке жанров отображается не больше 9 жанров + `All genres` (выводит в список фильмы любых жанров).

Список фильмов

При смене жанра или получении информации о фильмах с сервера, в списке фильмов отображается не больше 8 фильмов.

Показ дополнительных фильмов выполняется нажатием на кнопку «Show more».

Нажатие на кнопку «Show more» добавляет в список очередные 8 фильмов или оставшиеся фильмы, если их меньше.

После отображения всех фильмов, соответствующих выбранному жанру, кнопка «Show more» скрывается.

При переходе с главной страницы на другие страницы приложения и обратно счётчик показанных фильмов сбрасывается и отсчёт начинается заново.



Карточка фильма в списке фильмов

В карточке фильма выводится следующая информация:

- Изображение (превью фильма).
- Название фильма.

При клике на изображение или заголовок фильма выполняется переход на страницу «Film» (`/films/:id`).

При наведении и удержании курсора мыши на изображении фильма, вместо изображения начинает воспроизводиться видео-превью фильма.

Страница с описанием фильма

Страница с детальным описанием фильма доступна по адресу `/films/:id`, где `id` — идентификатор фильма, полученный с сервера. Например: `/films/123`.

Страница с детальным описанием фильма доступна всем пользователям.

В шапке страницы приведён следующий набор информации:

- Большой постер.
- Обложка фильма.
- Название фильма.
- Жанр.
- Год выхода на экраны.
- Кнопка запуска просмотра.
- Кнопка добавления в список «К просмотру».
- Кнопка перехода на страницу добавления отзыва.

Более детальная информация о фильме представлена на трёх вкладках:

- Overview. Общая информация.
- Details. Расширенная информация.
- Reviews. Отзывы.

Вкладки на странице описания фильма

Overview. Общая информация о фильме:

- Описание фильма.
- Оценка. Например, `8.9` (всегда один знак после запятой).
- Описание оценки (`Bad`, `Normal`, `Good`, `Very good`, `Awesome`).
- Количество голосов.
- Режиссёр.
- Список актёров.



Details. Расширенная информация:

- Режиссёр.
- Актёрский состав.
- Продолжительность (часы, минуты).
- Жанр.
- Год выхода на экраны.

Reviews. Список отзывов пользователей.

Оценка фильма

Текстовое представление оценки фильма формируется по следующим правилам:

- от 0 до 3 — **Bad**.
- от 3 до 5 — **Normal**.
- от 5 до 8 — **Good**.
- от 8 до 10 — **Very good**.
- 10 — **Awesome**.

Похожие фильмы

Блок «More like this» показывает похожие фильмы. В блоке отображается до 4-х карточек схожих фильмов.

Список похожих фильмов загружается с сервера (см. «Маршруты»).

Карточки содержат тот же набор информации, что и карточки на главной странице.

Клик по карточке из блока «More like this» выполняет переход на страницу «Film» соответствующего фильма.

Отзывы

Каждый отзыв содержит:

- Текст отзыва.
- Оценка пользователя.
- Имя пользователя.
- Дата отзыва в формате: **Месяц (полное название) день, год**. Например: **December 24, 2018**.

Добавление нового отзыва выполняется по кнопке «Add review». Кнопка должна отображаться только для авторизованных пользователей.

Форма отправки отзыва

При нажатии на кнопку «Add review» выполняется переход на страницу **Add review**

(/film/{id}/review)



Страница доступна только авторизованным пользователям. Неавторизованные пользователи перенаправляются на страницу [Sign In](#).

Пользователь выставляет оценку фильму от 1 до 10. Оценка выставляется путём выделения определённого количества звёзд. Если пользователи не успели оценить фильм, сервер вернёт в качестве оценки 0.

Текст отзыва должен быть не меньше 50 и не больше 400 символов.

Пока пользователь не поставил оценку и не ввел допустимый объём текста, кнопка отправки отзыва не активна.

При нажатии кнопки «Post» и отправке данных кнопка «Submit» и вся форма должны блокироваться. Разблокировка формы и кнопки происходит в случае успешной отправки или при возникновении ошибки.

В случае успешной отправки формы пользователь перенаправляется в карточку текущего фильма.

В случае возникновения ошибки следует уведомить пользователя. Способ отображения ошибки остаётся на усмотрение разработчика.

Страница MyList

Страница содержит информацию о фильмах, добавленных в список «К просмотру».

Данные для страницы «MyList» всегда загружаются с сервера. Для этого предусмотрен отдельный маршрут (см. раздел [«Взаимодействие с сервером»](#)).

Добавление в список «К просмотру» осуществляется при нажатии на кнопку «+ MyList» на странице «Film» и на главной странице для промо-фильма. Кнопка «+ MyList» заменяется на кнопку «✓ MyList».

Если фильм уже добавлен в список «К просмотру», нажатие на кнопку «✓ My List» удаляет фильм из списка. Кнопка «✓ MyList» заменяется на кнопку «+ MyList».

Страница «MyList» доступна только авторизованным пользователям. Неавторизованные пользователи перенаправляются на страницу [«Sign In»](#).

Клик по карточке фильма (изображение, заголовок) выполняет переход на страницу «Film» с детальным описанием фильма.

Страница Sign In

Страница «Sign in» доступна по адресу [/login](#).

Для входа в сервис пользователь вводит логин (email) и пароль.

Поскольку у сервиса отсутствует возможность регистрации, логин и пароль могут быть любыми, но не пустыми.



В поле «логин» должен вводиться корректный email.

В поле «пароль» должен вводиться валидный пароль. Под валидным паролем подразумевается пароль, который состоит минимум из одной буквы и цифры.

Страница доступна только неавторизованным пользователям. Авторизованных пользователей перенаправляет на главную страницу.

Информация об ошибках выводится в блок ошибок.

Просмотр фильмов

При нажатии на кнопку «Play» отрисовывается плеер и начинается показ выбранного фильма. В момент загрузки фильма отображается анимированный спиннер. Реализация спиннера остаётся на усмотрение разработчика.

Функциональность плеера:

- «Play/Pause». Запуск/остановка видео.
- «Fullscreen». Перевод в полноэкранный режим.
- «Time elapsed». Оставшееся время проигрывания видео. Время выводится в формате `MM:SS` — минуты, секунды или `-HH:MM:SS` — часы, минуты, секунды, если продолжительность видео больше одного часа, например:
 - Менее часа: `-53:12`;
 - Более часа: `-01:45:35`.
- «Exit». Остановка просмотра. Плеер скрывается.

Плеер реализуется с помощью `<video>`.

Для перевода времени проигрывания видео в нужный формат разработчик может воспользоваться дополнительным пакетом на своё усмотрение.

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`).

Взаимодействие с сервером

Все необходимые данные загружаются с сервера.

Сервер доступен по адресу: `https://8.react.pages.academy/wtw`.

В случае недоступности сервера отображается информационное сообщение. Дизайн сообщения остаётся на усмотрение разработчика.

Сервер принимает данные в виде JSON объекта.

Авторизация на сервере происходит на основании токена. Токен передаётся с каждым



запросом в заголовке `X-Token`.

Структуры данных

Film

```
{
  "id": 1,
  "name": "The Grand Budapest Hotel",
  "poster_image": "img/the-grand-budapest-hotel-poster.jpg",
  "preview_image": "img/the-grand-budapest-hotel.jpg",
  "background_image": "img/the-grand-budapest-hotel-bg.jpg",
  "background_color": "#ffffff",
  "video_link": "https://some-link",
  "preview_video_link": "https://some-link",
  "description": "In the 1930s, the Grand Budapest Hotel is a popular European ski resort, presided over by concierge Gustave H. (Ralph Fiennes). Zero, a junior lobby boy, becomes Gustave's friend and protege.",
  "rating": 8.9,
  "scores_count": 240,
  "director": "Wes Anderson",
  "starring": ["Bill Murray", "Edward Norton", "Jude Law", "Willem Dafoe", "Saoirse Ronan"],
  "run_time": 99,
  "genre": "Comedy",
  "released": 2014,
  "is_favorite": false
}
```

CommentGet

```
{
  "id": 1,
  "user": {
    "id": 4,
    "name": "Kate Muir"
  },
  "rating": 8.9,
  "comment": "Discerning travellers and Wes Anderson fans will luxuriate in the glorious Mittel-European kitsch of one of the director's funniest and most exquisitely designed movies in years.",
  "date": "2019-05-08T14:13:56.569Z"
}
```

CommentPost

```
{
  "rating": 8,
  "comment": "Discerning travellers and Wes Anderson fans will luxuriate in the glorious Mittel-European kitsch of one of the director's funniest and most exquisitely designed movies in years."
}
```



```
}
```

User

```
{  
  "email": "Oliver.conner@gmail.com",  
  "password": "12345678"  
}
```

AuthInfo

```
{  
  "id": 1,  
  "email": "Oliver.conner@gmail.com",  
  "name": "Oliver.conner",  
  "avatar_url": "img/1.png",  
  "token": "T2xpdmVyLmNvbW5lcckBnbWFpbC5jb20="`  
}
```

Маршруты

GET /films

Получение списка фильмов.

Коды ответов:

— 200 OK

Пример:

Request:

— URL: GET /films

Response:

— Status: 200 OK

— Body: массив, содержащий структуры типа `Film`

GET /films/: id

Получение фильма с идентификатором `id`.

Коды ответов:

— 200 OK

— 404 Not found

Пример:



Request:

- URL: GET /films/1

Response:

- Status: 200 OK
- Body: структура типа **Film**

GET /films/: id/similar

Получение списка похожих фильмов.

Коды ответов:

- 200 OK
- 404 Not found

Пример:

Request:

- URL: GET /films/1/similar

Response:

- Status: 200 OK
- Body: массив, содержащий структуры типа **Film**

GET /promo

Получение промо-фильма.

Коды ответов:

- 200 OK

Пример:

Request:

- URL: GET /promo

Response:

- Status: 200 OK
- Body: структуры **Film**

GET /favorite

Получение списка фильмов «к просмотру».

Коды ответов:

- 200 OK
- 401 Unauthorized (в случае, если не пройдена авторизация)



Пример:

Request:

— URL: GET /favorite

Response:

— Status: 200 OK

— Body: массив, содержащий структуры типа `Film`

POST /favorite/: film_id/: status

Изменение статуса «к просмотру» у фильма. В теле ответа приходят данные фильма с актуальным состоянием поля `is_favorite`.

Параметры:

- `:film_id` — ID фильма, который нужно убрать/добавить в список «к просмотру»
- `:status` — значения могут быть 1 или 0. 1 добавляет фильм в список «к просмотру», 0 удаляет

Коды ответов:

- 200 OK
- 401 Unauthorized (в случае, если не пройдена авторизация)

Пример:

Request:

— URL: POST /favorite/42/1

Response:

— Status: 200 OK

— Body: структура `Film` с актуальным состоянием поля `is_favorite`

GET /comments/: film_id

Получить список комментариев для фильма по его `id`.

Коды ответов:

- 200 OK
- 400 Bad request

Пример:

Request:

— URL: GET /comments/42

Response:

— Status: 200 OK



- Body: массив, содержащий структуры типа `CommentGet`

POST /comments/: film_id

Отправить новый комментарий к фильму по его `id`.

Коды ответов:

- 200 OK
- 400 Bad request
- 401 Unauthorized (в случае, если не пройдена авторизация)

Пример:

Request:

- URL: POST /comments/42
- Body: структура `CommentPost`

Response:

- Status: 200 OK
- Body: массив, содержащий структуры типа `CommentGet` для конкретного фильма

POST /login

Авторизация пользователя на сервере. В случае успешного запроса сервер вернёт токен, по которому в дальнейшем происходит авторизация. Этот токен необходимо передавать через заголовок `X-Token`. Если авторизация не проходит, сервер вернёт код состояния `401` при запросах на закрытые части сайта.

Коды ответов:

- 200 OK
- 400 Bad request

Пример:

Request:

- URL: POST /login
- Body: структура `User`

Response:

- Status: 200 OK
- Body: структура `AuthInfo`

GET /login

Проверка статуса авторизации пользователя.

Коды ответов:



- 200 OK
- 401 Unauthorized (в случае, если не пройдена авторизация)

Пример:

Request:

- URL: GET /login

Response:

- Status: 200 OK
- Body: структура `AuthInfo`

Коды ответов:

- 200 OK
- 401 Unauthorized (в случае, если не пройдена авторизация)

DELETE /logout

Завершение сеанса работы — выход из закрытой части приложения.

Пример:

Request:

- URL: DELETE /logout

Response:

- Status: 204 No Content

Дополнительно

Покройте код проекта тестами. Напишите тесты для всех компонентов, редьюсеров, асинхронных операций.

Поиск по материалам

Git



[Все материалы](#)

Мой наставник



[Выбор наставника](#)