

## Тема финальной работы

Разработка популярной онлайн-платформы ВК Маруся, на которой пользователи смогут искать фильмы и ставить им оценки.

## Описание задачи

Благодаря IT-компании VK вы сможете почувствовать себя настоящими разработчиками в большой корпорации. Вам предстоит сделать бета-версию популярного стримингового сервиса от ВК Маруси по макетам, которые предоставили наши партнёры.

Пользователи этого приложения должны иметь возможность искать фильмы и добавлять их в избранное, чтобы не забыть посмотреть. Вам необходимо создать функции поиска фильмов, просмотра категорий фильмов и трейлеров, а также сделать приложение удобным для использования.

## Что нужно сделать

### Описание поведения

Главная страница приложения — удобный дашборд, на котором пользователи могут выбрать случайный фильм и посмотреть список топовых фильмов по IMDb-рейтингу.

На отдельной странице пользователь может просмотреть список существующих киножанров, выбрать один из них и перейти на страницу с фильмами, которые к нему относятся.

При нажатии на карточку фильма пользователь попадает на его страницу. На ней есть детальное описание картины, кнопка для просмотра трейлера и кнопка для добавления фильма в избранное или его удаления из этого списка.

Функция добавления фильма в избранное доступна только для авторизованных пользователей. Вызов формы авторизации происходит при нажатии на кнопку «Войти» в шапке сайта или попытке добавить фильм в избранное (если пользователь ещё не авторизован). Кроме того, после авторизации пользователю становится доступна страница аккаунта с личной информацией и списком фильмов, которые он добавил в избранное.

Пользователю также доступна функция поиска фильма по названию, которой можно воспользоваться, нажав на поле поиска в разделе навигации.

Особенность приложения — возможность выбрать случайный фильм. Это можно сделать на главной странице, нажав на кнопку возле раздела «Избранное».

## Состав продукта

- Главная страница — дашборд, на который пользователь попадает при входе на сайт. На этой странице пользователь видит:
  - обложку случайного фильма, который приложение предлагает для просмотра, и информацию о нём. Этот фильм генерируется случайно, пользователь может получить новый, нажав на кнопку;
  - блок с топом, в котором перечислены лучшие 10 фильмов по IMDb-рейтингу.
- Страница со списком жанров. Пользователь попадает сюда, кликнув по разделу «Жанры» в строке с навигацией по сайту. Структура страницы — список жанров, оформленный в виде карточек.
- Страница конкретного жанра. Пользователь попадает сюда, нажав на карточку жанра в разделе «Жанры». Структура страницы:
  - карточки с фильмами, отфильтрованными по жанру;
  - по умолчанию первые 10 фильмов, отсортированные по рейтингу. При скролле подгружаются следующие 10 фильмов (пока не закончится список).
- Страница фильма. Пользователь попадает, кликнув по карточке фильма в разделе «Топ» на главной странице или на странице жанра. Структура страницы:
  - обложка;
  - полная информация о фильме;
  - кнопка для просмотра трейлера;
  - кнопка для добавления фильма в избранное или его удаления из этого списка.

Если пользователь не авторизован, при нажатии на кнопку добавления фильма в избранное или удаления из него открывается модальное окно с формой регистрации или авторизации. Если же пользователь авторизован, происходит добавление фильма в список или удаление из него, кнопка отражает соответствующее состояние.

При нажатии на кнопку просмотра трейлера открывается модальное окно, в которое встроено видео с трейлером.

- Страница аккаунта пользователя. На ней представлена информация, указанная пользователем при регистрации, а также список фильмов, которые он добавил в избранное. При нажатии на кнопку «Выйти» происходит разлогинивание пользователя, и он попадает на главную страницу.
- Модальное окно авторизации. Вызывается при попытке добавить фильм в избранное, если пользователь ещё не авторизован, или при нажатии на кнопку «Войти» в разделе с навигацией. По умолчанию открывается форма для входа

в аккаунт, но после нажатия на кнопку «Регистрация» содержимое окна меняется на форму регистрации.

При регистрации или авторизации все поля обязательны для заполнения. Если пользователь оставил какие-либо поля пустыми, они подсвечиваются красной рамкой.

После успешной авторизации или регистрации в разделе с навигацией кнопка «Войти» меняется на фамилию пользователя, указанную при регистрации. При нажатии на неё пользователь попадает на страницу аккаунта.

- Поиск фильма. При нажатии на поле «Поиск» в разделе с навигацией по сайту открывается модальное окно, в котором можно ввести название фильма и получить список картин с таким заголовком.

## Описание внешнего вида и UI

Детальное описание внешнего вида платформы и UI вы найдёте на [макете в Figma](#). Возле некоторых страниц мы оставили комментарии с подробным объяснением, как работает логика блоков.

Важно, чтобы внешний вид сайта и отдельных блоков соответствовал макетам.

Также на макетах есть UI-кит — основной набор компонентов, которые встречаются на всех страницах сайта и на основе которых строятся более сложные компоненты. Советуем начать реализацию UI именно с него.

## Технические требования

Приложение должно быть написано с использованием библиотеки React (17-18) и TypeScript. Помимо этого, вы можете использовать CSS-препроцессоры (SASS, SCSS и другие).

Основные библиотеки, которые можно использовать:

- Redux;
- Axios;
- React-Router.

Вы также можете работать с любыми другими библиотеками для реализации какого-либо функционала, за исключением UI-библиотек (Bootstrap, MaterialUI и другие).

Проект должен быть написан с использованием стайлгайда. В нём должны быть:

- разделение на логические директории: страницы, компоненты, стили, логика по работе с API и так далее;

- понятные имена файлов и методов в файлах с кодом;
- имена методов в формате глаголов, которые описывают производимые методами действия;
- имена компонентов в формате существительных, описывающих, что отображают компоненты;
- наименования файлов, методов и переменных, соответствующие стайлгайду (PascalCase для файлов, camelCase для методов и переменных), без сокращений и транслита.

Придерживайтесь правил чистого кода, по которым:

- длина методов не превышает 20 строк;
- количество уровней вложенности блоков кода не выше трёх относительно уровня содержимого метода;
- количество параметров у методов не больше двух (редко — трёх);
- названия переменных описывают их суть;
- имена методов в файлах — глаголы, которые описывают, что делают методы;
- имена компонентов — существительные, которые описывают, что отображают компоненты;
- логика кода понятна при первом прочтении.

## Инструкции по использованию инфраструктуры

Подробную документацию по работе с API вы найдёте [на сайте](https://cinemaguide.skillbox.cc/docs/) <https://cinemaguide.skillbox.cc/docs/>.

Перейдя по ссылке, вы увидите список и описание доступных запросов, а также описание моделей данных, которые принимают и возвращают эти запросы.

## Работа с API

Важные моменты, которые нужно учесть при работе с API:

- Авторизация происходит с использованием сессий и работает следующим образом:
  - 1) отправляется запрос на авторизацию;
  - 2) если данные введены корректно, сервер присылает cookie, в которых находится информация о сессии;
  - 1) при выполнении к запросам автоматически добавляется cookie; поскольку вы, скорее всего, будете отправлять запросы с локалхоста, отправляйте их с опцией включения cookies и авторизационных заголовков в кросс-доменных запросах (credentials: include в fetchApi, withCredentials: true в Axios).
  - 3) работа cookie ограничена по времени. Если оно истекло, сессия авторизации автоматически завершается, и нужно повторить процесс.

Для ручного разлогинивания нужно вызвать соответствующий запрос, который удалит сессию с сервера и cookie клиента. После этого сессия авторизации завершится.

- Чтобы получить список жанров, необходимо отправить соответствующий запрос. В результате вы получите массив строк с названиями жанров. Далее необходимо преобразовать этот массив строк в массив объектов, добавить название и изображение (можно выбрать своё или взять из макета).

## Что можно добавить

Это дополнительные детали и функции, которые вы можете по желанию добавить в приложение, чтобы его улучшить.

- Анимация:
  - при переходах между страницами;
  - при открытии или закрытии модальных окон;
  - при наведении курсора на карточки фильмов и категорий.
- Покрытие тестами сложных (по вашему мнению) частей приложения.
- Настройка уведомлений об ошибках, которые могут возникнуть при выполнении запросов. Например, если пользователь вводит неверные логин или пароль при авторизации.
- Прелоадеры для ситуаций, требующих асинхронных действий: авторизация, загрузка списка фильмов, загрузка фильма и другое.

## Формат сдачи материалов и оценивание

Результат работы должен быть представлен в виде развёрнутого проекта в GitLab Skillbox, готового к запуску. В проекте должны быть реализованы все технические требования, указанные в ТЗ.

Важное условие: при обновлении страницы авторизованным пользователем авторизация не должна сбрасываться. Это же касается и избранных фильмов. Также приложение не должно ломаться при переходе между страницами и выполнении асинхронных действий (запрос случайного фильма, добавление фильма в избранное или его удаление из списка, регистрация или авторизация, поиск).