**Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

Отчет по проекту

по дисциплине «Основы UI и UX дизайна»

Выполнила: студентка группы БВТ2001

Павлова Л.Д.

**Оглавление**

[Цель работы 3](#_Toc153888845)

[Задание 3](#_Toc153888846)

[Теоретические сведения 4](#_Toc153888847)

[Выполнение работы 6](#_Toc153888848)

[Вывод 7](#_Toc153888849)

# **Цель работы**

Необходимо разработать SPA исходя из описания задачи.

# **Задание**

Имеется API со списком фильмов, с которым нужно наладить взаимодействие.

Необходимо визуализировать данные полученные от API в виде таблицы, содержащей только важную (на ваш взгляд) информацию о фильме.

К каждому фильму из таблицы можно оставить комментарий (неограниченное количество), а также удалить его в любой момент времени, локальное состояние.

Ссылка на документацию API: <https://yts.mx/api>

Методы, с которыми необходимо работать:

* List Movies;
* Retrieve Movie.

Требования к задаче:

* NextJS;
* Отсутствие перезагрузки страницы;
* В приложении должна присутствовать пагинация.

Правила хорошего тона:

* Адаптивная верстка;
* Приятный и удобный UI;
* Простой и лаконичный код;

DRY.

# **Теоретические сведения**

На JavaScript выполняется большая часть интерактивных элементов на сайтах и в мобильных приложениях. JavaScript отлично работает с HTML/CSS и интегрирован основные браузеры на рынке. Чистый JavaScript используется в вебе, а для общего применения JavaScript разработчики используют различные среды выполнения, например, Node.js.

Node.js — это среда выполнения кода JavaScript. Она позволяет использовать JavaScript как язык программирования общего назначения: создавать на нем серверную часть и писать полноценные десктопные приложения.

Основа Node.js — движок V8. Этот движок был разработан Google и используется в браузере Google Chrome. Он компилирует код JavaScript в машинный код, который понимает процессор. Однако, чтобы сделать из JavaScript язык общего назначения, одного движка недостаточно. Так, например, для создания серверной части нужно, чтобы язык умел работать с файлами, сетью и т.п. Для решения этой проблемы разработчики добавили к V8 дополнительные возможности, с помощью своего кода и сторонних библиотек. В итоге у них получился инструмент, который превращает JavaScript в язык общего назначения.

Node.js стала популярна среди разработчиков благодаря возможности создавать серверную и клиентскую часть на одном языке, скорости работы и NPM.

React Query — отличная библиотека хуков для управления запросами данных, которая полностью избавляет от необходимости помещать удаленные данные в глобальное состояние. Вам просто нужно указать библиотеке, где вы хотите получить данные, и она будет обрабатывать кеширование, фоновые обновления и аннулирование данных без какого-либо дополнительного кода или конфигурации.

React Query также устраняет необходимость использовать хуки useState и useEffect и заменяет их на несколько строк логики React Query. В конечном итоге это определенно поможет вам поддерживать приложение. Если вам интересно узнать больше, не забудьте ознакомиться с документацией по [React Query](https://react-query.tanstack.com/docs/overview).

# **Выполнение работы**

Сначала выполним установку следующих инструментов разработки:

* Среда выполнения кода NodeJS;
* VisualStudio Code c расширениями: Babel JavaScript, Tailwind CSS IntelliSense, PostCSS Language Support.

Запустим в терминале VSC команду npx create-next-app@latest. После активации команды выберем необходимые опции. После установки всех пакетов, необходимо запустим приложение на localhost:3000.

При запуске нас встречает главная страница, на которой представлены карточки фильмов. Они разделены на 4 колонки по 5 строк. Всего на первой странице можно увидеть 20 фильмов. Главная страница представлена на рисунке 1.

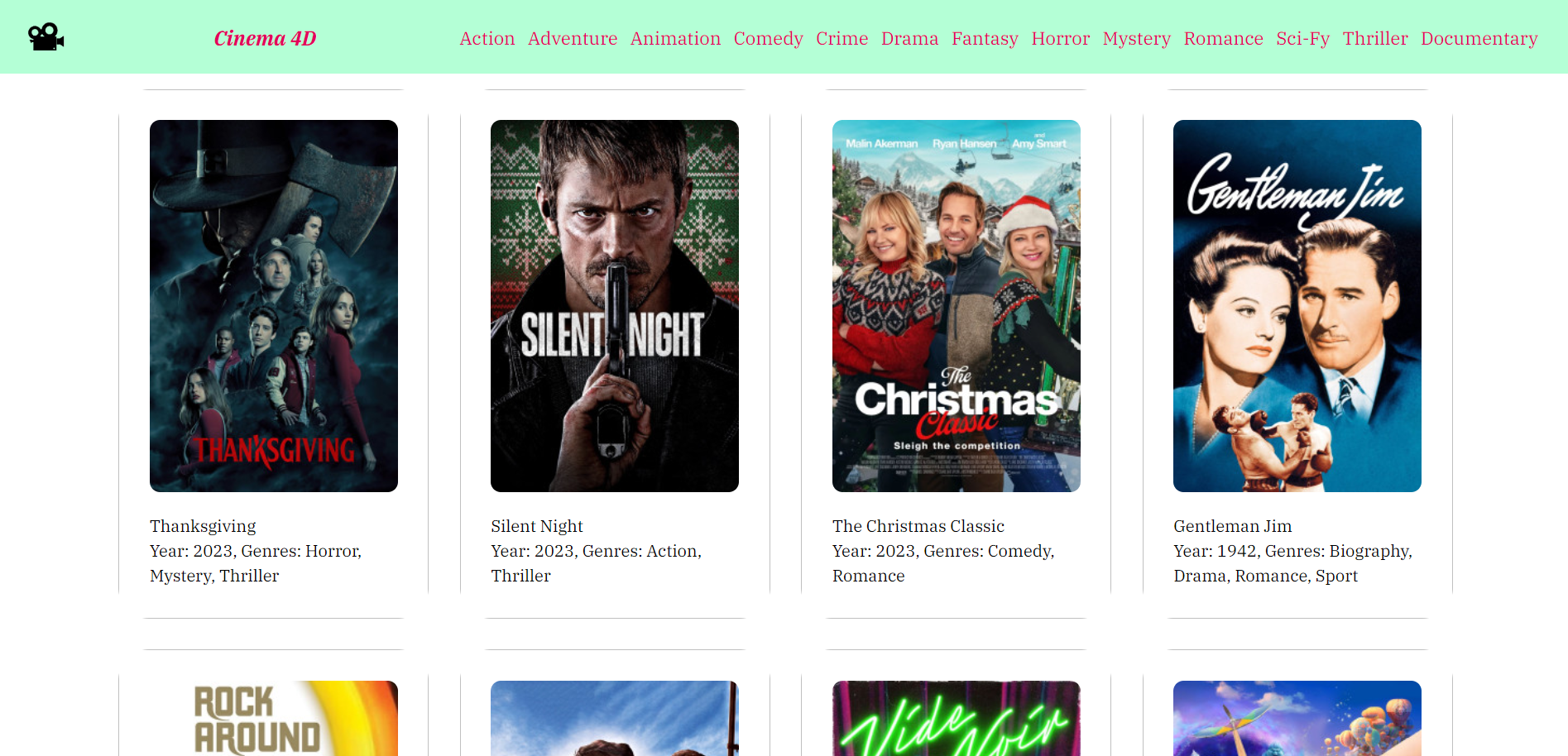
****

Рисунок 1 – Главная страница сайта

В проекте требовалась реализация пагинации. Она представлена на рисунке 2. Для того, чтобы ей было удобно пользоваться, пагинация включает в себя ссылки на соседние страницы, а также на первую и последнюю.

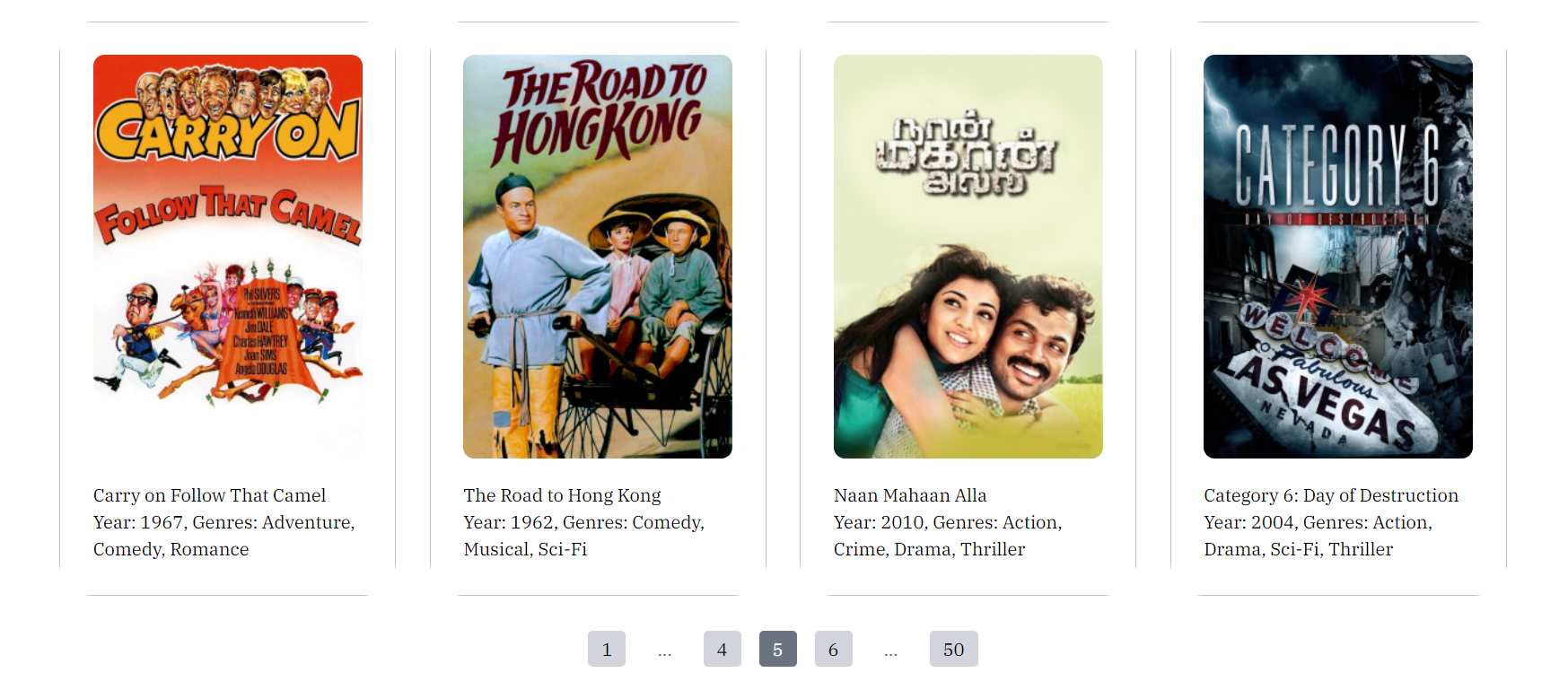
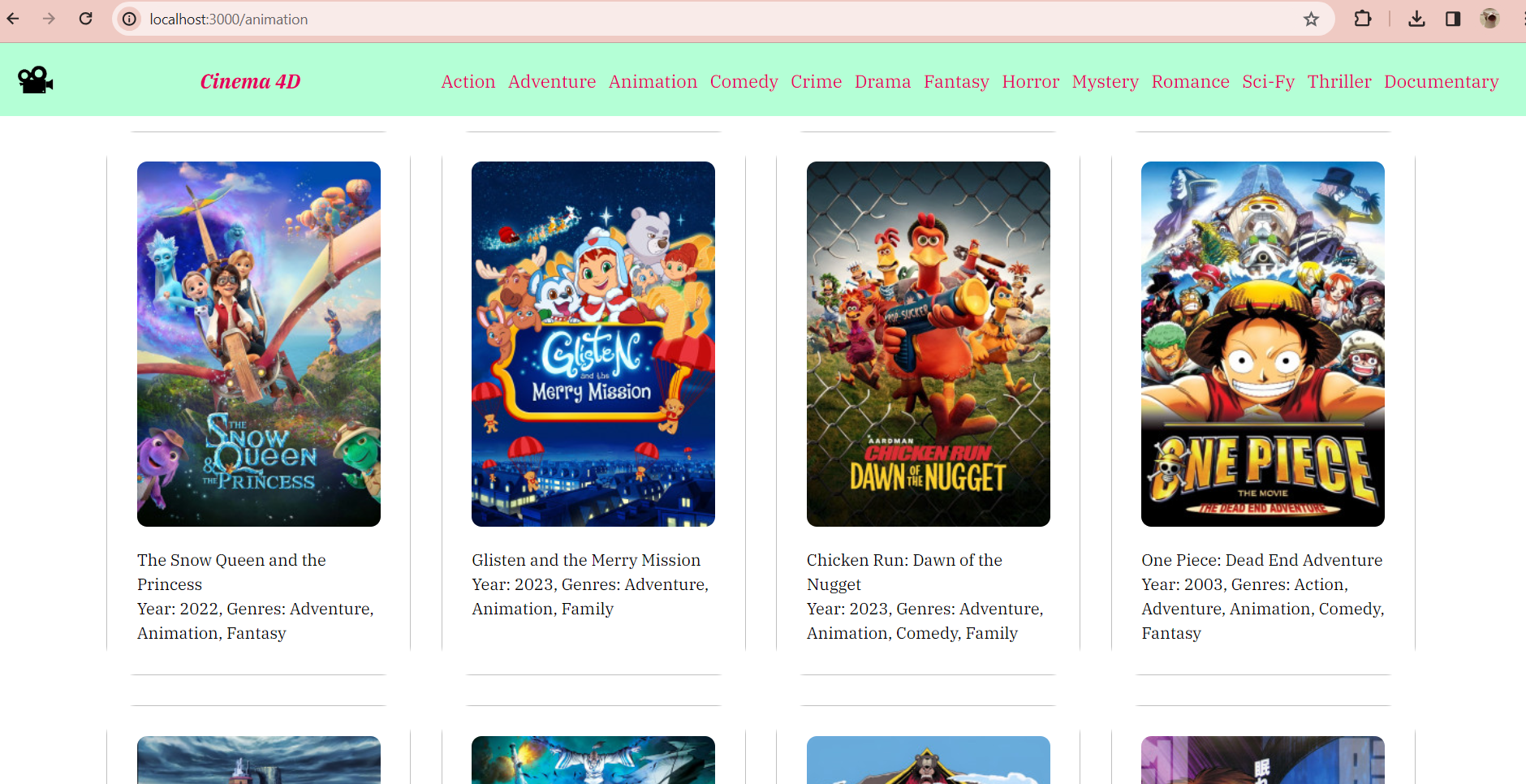
****

Рисунок 2 – Пагинация

Так же было реализовано деление фильмов по жанрам, путем фильтрации фильмов, по указанным в них жанрах. На страницах с жанрами так же реализована пагинация. Страница с жанром “Animation” представлена на рисунке 3.

****

Страница фильмов с жанром “Animation” – Рисунок 3

При клике на любую из карточек фильма, открывается страница с детальным описанием фильма. На ней представлены год выпуска фильма, его жанры, рейтинг и краткое описание. Так же отдельно выведен список актеров, которые представлены для данного фильма в предоставленной API. Страница с детальной информацией о фильме представлена на рисунке 4.

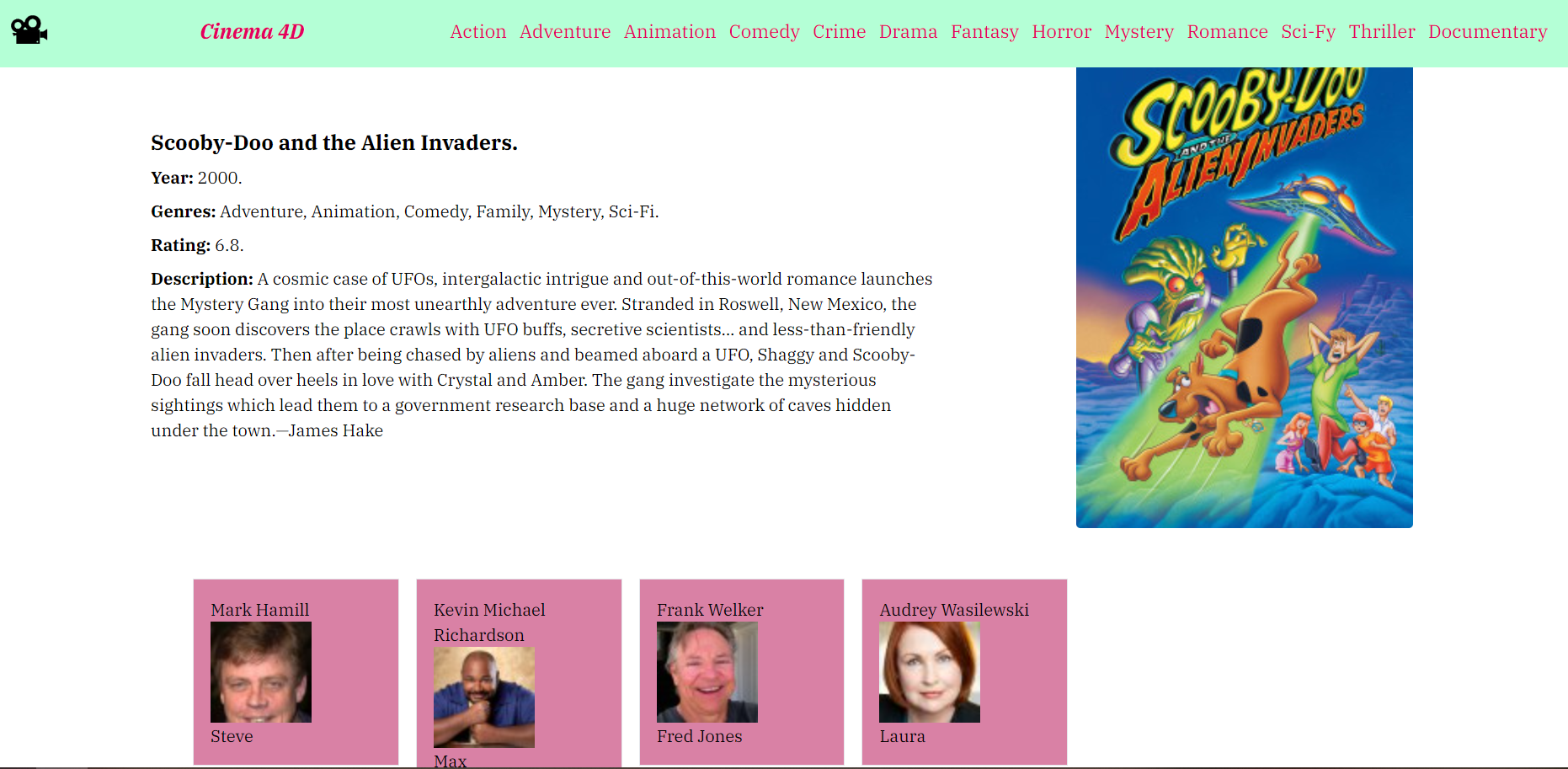
****

Рисунок 4 – Детальная информация о фильме

Так же внизу страницы с детальной информацией реализовано добавление комментариев. Если комментариев нет, выводится фраза “ There no comments... You can be first!”. Комментарии можно удалять, они не пересекаются с другими фильмами. Выполнение представлено на рисунке 5.

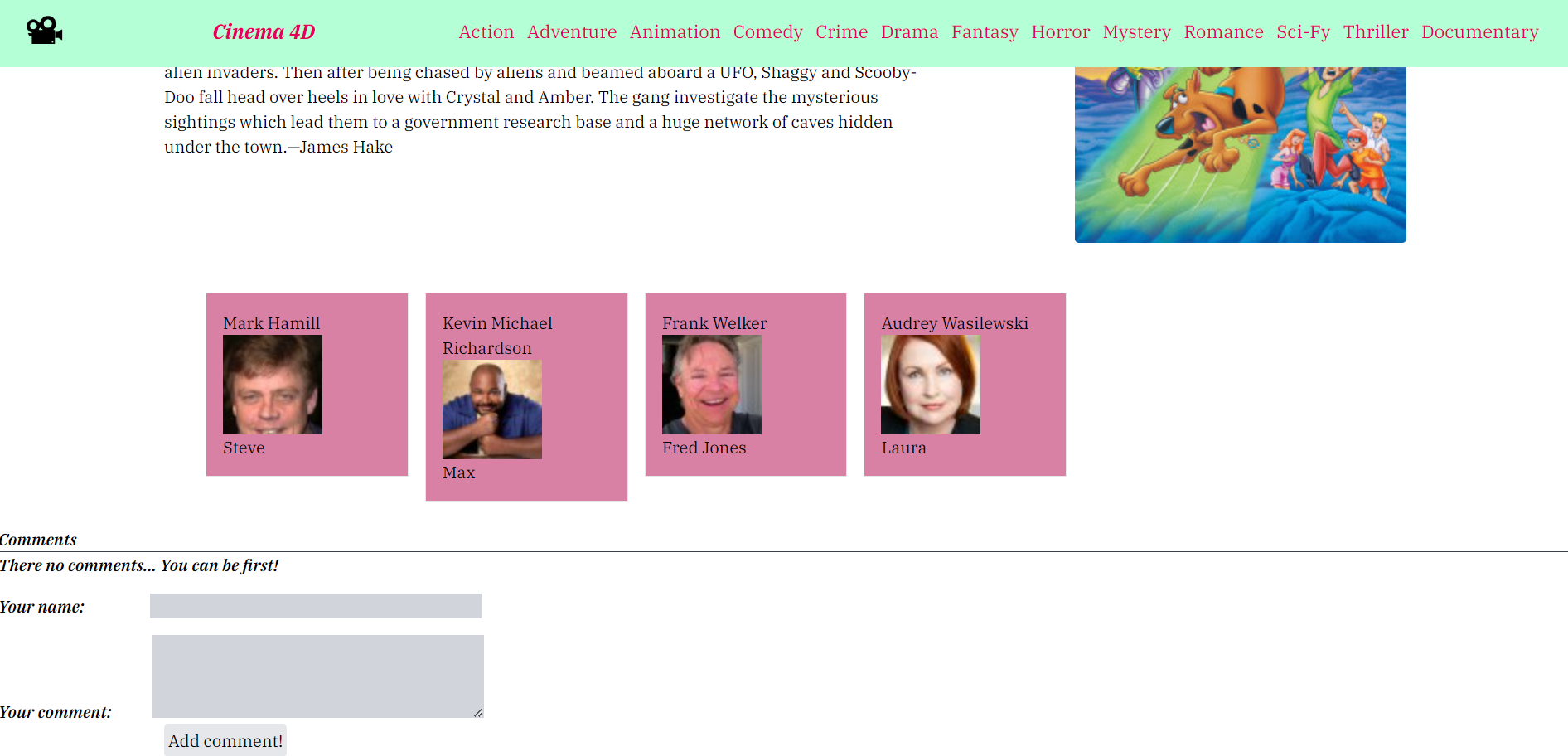
****

Рисунок 5 – Комментарии на странице детальной информации фильма

На рисунке 6 представлен пример с написанием комментария, “заглушки” под отсутствие комментариев уже нет, так же под написанным комментарием есть кнопка его удаления.

****

Рисунок 6 – Написание комментария

Адаптивная верстка так же была реализована. На рисунках 7-9 представлен вид стайте для телефона для разных страниц.

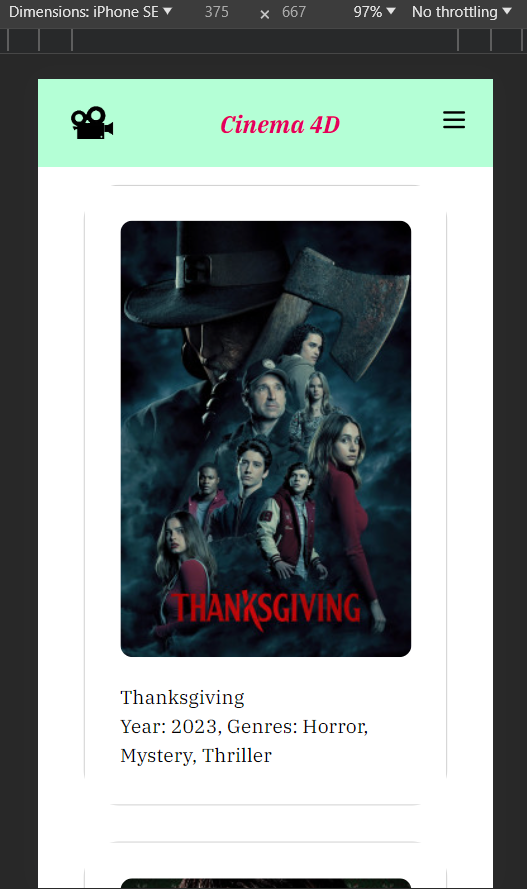


Рисунок 7 – Главная страница на телефоне

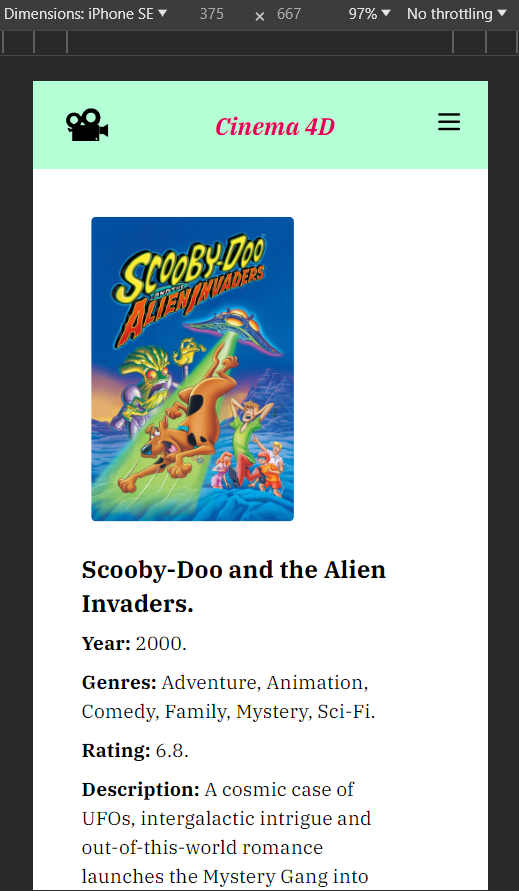


Рисунок 8 – Страница с детальной информацией на телефоне

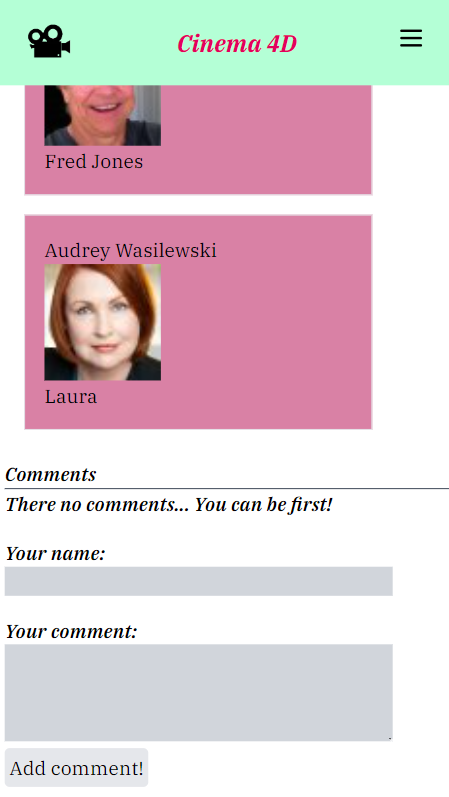


Рисунок 7 – Комментарии на телефоне

# **Вывод**

В результате выполнения данного проекта мною было реализовано SPA-приложение, которое работает с API, содержащим список фильмов. Проект был выполнен с помощью Next.js. Приложение имеет главную страницу с фильмами, представленными в виде таблице, есть возможность отфильтровать фильмы по жанрам, перехода на детальную информацию о фильме с возможностью оставить комментарии. Также были реализованы пагинация и адаптивная верстка.