**Резюме по проекту**

**Табл.1 Состав:**

|  |  |  |
| --- | --- | --- |
| **№** | **Требование** | **Комментарий** |
| **1** | **Главная страница**, содержит: | HomePage |
| 1.1 | header: Логотип сайта (при клике осуществляется переход на главную страницу) | + |
| 1.2 | Избранные (переход на страницу избранных фильмов) | + |
| 1.3 | Login пользователь или кнопка "войти"( переход на страницу авторизации) | + |
| 1.4 | На главной странице должен отображаться список фильмов. Список фильмов можно отфильтровать (например по дате, алфавиту и т.д.) | + |
| 1.5 | Реализовать кастомную пагинацию для вывода списка фильмов | + |
| 1.6 | Реализовать поиск фильма по введенной строке из поля поиска | + |
| 1.7 | При клике по карточке фильма - открывается страница с информацией о конкретном фильме | + |
| **2** | **Страница с информацией о конкретном фильме** | CardInfoPage |
| 2.1 | Есть кнопка "like", которая добавляет текущий фильм в "избранные" | + (появляется при авторизации/регистрации) |
| 2.2 | Только авторизованные пользователи могут сохранять фильмы в "избранные". | + |
| **3** | **Страница авторизации** | RegistrationPage |
| 3.1 | Должна присутствовать кнопка "вход" и кнопка "регистрация" | AuthorizationModal |
| 3.2 | При клике по "вход" отображается форма с полями login и password, при правильном вводе данных пользователь становится авторизованным и в header отображается его login. | AuthPage |
| 3.3 | При клике на "регистрация" отображается форма для регистрации нового пользователя с 3 полями: login, пароль, повторите пароль. Для пароля требуется предусмотреть минимальную валидацию. | RegistrationPage |
| **4** | Страница со списком "избранных фильмов" (данные хранятся в LS) | FavoritesPage, FavoritesProvider |
| 4.1 | Реализовать пагинацию для избранных фильмов | + |
| **5** | 404 страница | NotFoundPage |
| **6** | Страница для Error Boundary | ErrorFallback |

**Табл. 2 Требования:**

|  |  |  |
| --- | --- | --- |
| **№** | **Требование** | **Комментарий** |
| 1 | В проекте используются функциональные компоненты c хуками | + |
| 2 | Есть четкое разделение на умные и глупые компоненты | + |
| 3 | Есть рендеринг списков в компонентах | HomePage, FavoritesPage |
| 4 | Реализованы формы в компонентах | HomePage, RegistrationPage, AuthPage |
| 5 | Есть применение React Контекст API | Card, CardInfoPage, FavoritesPage |
| 6 | Есть применение предохранителя | router for HomePage |
| 7 | Есть кастомные хуки (минимум 1) | useDebounce |
| 8 | Используется PropTypes (для проектов без TS) (минимум в 3 компонентах) | Card, ItemInfo, SearchInput |
| 9 | Есть кейс с применение Debounce | HomePage |
| 10 | Есть применение lazy + Suspense | router for FavoritesPage |
| 11 | В проекте используется Redux Toolkit | store, authSlice |
| 12 | В проекте используется RTK Query | rickAndMortyApi |
| 13 | В проекте есть кастомная мидлвара (минимум 1) | perfomanceMiddleware |
| 14 | Регистрация пользователей в приложении осуществляется через LocalStorag | + |
| 15 | Дополнительно: возможность разлогиниться | LogOutModal |