

Тестовое задание iOS

Задание: создать клиент-серверное приложение для поиска интересных фильмов.

Экраны:

1. Main – экран с двумя таблами:
 - List of current popular movies.
 - List of favorite movies.
2. Genre selection.
3. Search for movies.
4. Movie details.

“List of current popular movies” – список всех популярных фильмов. Может быть отфильтрован по определенному жанру, если пользователь выбрал таковой. Каждый элемент списка должен содержать изображение, название, описание. Использовать пагинацию.

“List of favorite movies” – список избранных фильмов. Предусмотреть возможность удаления фильмов из избранного. Каждый элемент списка должен содержать изображение, название, описание. Использовать пагинацию.

“Genre selection” – отдельный экран для выбора жанра, который должен быть сохранен до следующего запуска приложения.

“Search for movies” – поиск фильмов.

“Movie details” – детальная информация о фильме. На данном экране пользователь может добавить или удалить фильм из избранного. Решение о количестве отображаемой информации возлагается на исполнителя.

Требования:

- Swift
- Структурированный код, архитектура построенная на MVC.
- Использовать CocoaPods для сторонних библиотек.
- Для локального хранения данных использовать Core Data (сторонняя библиотека MagicalRecord <https://github.com/magicalpanda/MagicalRecord>).
- UI сделать на базе Storyboard/Autolayout.

- Кеширование картинок на стороне приложения.
- Многопоточность (асинхронное общение с API и асинхронная загрузка изображений).
- Автотесты Quick (<https://github.com/Quick/Quick>).

Дополнительные задания:

- Предусмотреть отображение сообщения в случае пустых списков.
- Добавить пагинацию на экране поиск фильмов.
- Pull-to-refresh для обновления списков.
- Добавить возможность сортировать избранные фильмы по дате добавления в избранное, названию фильма, году выхода фильма или рейтингу.
- Построить архитектуру на MVVM с использованием KVOController (<https://github.com/facebook/KVOController>).

Ресурсы:

1. API Key: “fbe4e6280f6a460beaad8ebe2bc130ac” (можно создать свое приложение и использовать свой ключ). Сайт: <https://developers.themoviedb.org/3/getting-started/introduction>
2. API для получения списка фильмов: <https://api.themoviedb.org/3/discover/movie>
3. API для получения списка официальных жанров для фильмов: <https://api.themoviedb.org/3/genre/movie/list>
4. API для получения основной информации о фильме: https://api.themoviedb.org/3/movie/{movie_id}
5. API для поиска фильмов: <https://api.themoviedb.org/3/search/movie>