Урок 6. Домашнее задание

# Задача

За основу можно взять приложение из домашнего задания урока 4. Нужно реализовать приложение, которое отображает список элементов, но не захардкоданных, а скаченных из интернета. Есть такой сервис rapidapi.com, он содержит множество различных API на различные темы, такие как фильмы, музыка, погода и так далее.

Нужно выбрать любой сервис и сделать так, чтобы приложение брало элементы списка из него. Например, это могут быть самые популярные фильмы, популярная музыка, аниме, сериалы и что угодно. Также, нужно добавить на UI строку поиска, которая позволит искать элементы с помощью API. В случае с фильмами это может быть поиск по названию, жанру и так далее.

Поле ввода должно работать так, чтобы поиск начинался автоматически после секундного бездействия пользователя. Т.е. пока пользователь вводит текст – ничего не ищется, когда пользователь закончил и секунду не вводит текст – запускается поиск.

Пока идет поиск или изначальная загрузка данных, должен отображаться спиннер.

Если поле ввода пустое – должны отображаться изначальные элементы (популярные фильмы).

По мере прокрутки страницы, нужно догружать элементы в список порциями по 10 штук. Потенциально, список должно быть можно быть листать очень долго.

Дизайн – на усмотрение студента.

По нажатию на элемент должна открыться страница деталей элемента, где будут загружены дополнительные данные по элементу. Для навигации можно использовать метод Navigator.of(context).push(MaterialPageRoute(страница деталей)…..)

Получать данные должен класс-сервис, реализованный с помощью chopper, модели – с помощью freezed + json\_serializable.

Не обязательно использовать rapidapi.com, можно на свой выбор использовать любое другое API