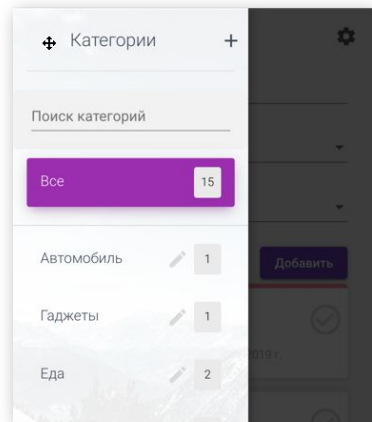
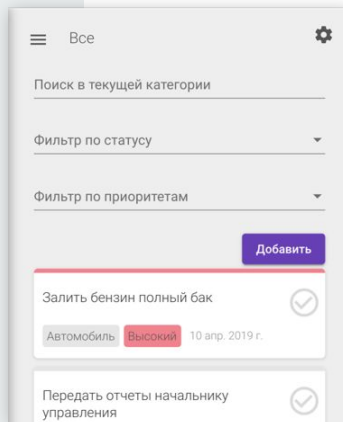
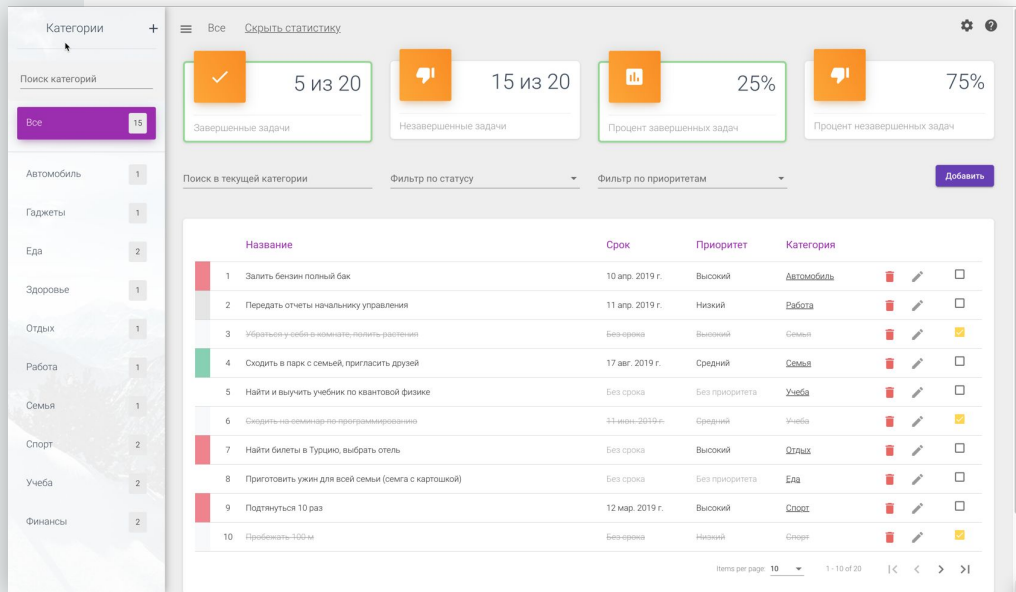




Функционал приложения Todo







Последовательность

1. Прописать функционал - очень важно “видеть” границы приложения - хотя бы на 1-м этапе - потом будет легче проектировать UI и архитектуру

2. Сделать наброски UI - чтобы выбрать подходящий шаблон-дизайна (не будем с нуля создавать вручную, тратить время)

3. Реализовать код на Angular (TypeScript) - использовать все базовые и профессиональные принципы построения приложения

Один из главных принципов, который всегда нужно держать в уме - это разделение UI и логики работы



Функционал

* За основу можно взять Wunderlist (сейчас Microsoft Todo):

CRUD (создание/добавление/удаление/обновление):

- Категории (вложенность одного уровня)
- Задачи со статусами (решено, нерешено)
- Приоритеты с цветом (чтобы выделять задачи)

Статистика:

- решенные
- нерешенные
- проценты

Поиск/фильтрация:

- Категория
- Статус
- Приоритет
- Название задачи

Отображение на мобильном устройстве



Основные объекты

Подходит и для таблиц БД

Задача:

- id
- Название
- Статус
- Приоритет
- Категория
- Срок (до какого числа нужно выполнить)

Приоритет:

- id
- Название
- Цвет

Категория:

- id
- Название



Технологии

В данный момент Todo - это и **backend** и **frontend**:

backend - манипулирует данными напрямую (имеет доступ к БД)

frontend - отображает внешний вид и реагирует на действия пользователя

Используется сервер Node.js

Если бы данные брались с другого сервера с помощью запросов - то текущий проект был бы только как **frontend**



Домашнее задание

- Для каждого свойства объектов (задача, категория, приоритет) прописать их тип, пока без кода
- Типы данных в TS:
 - <https://www.dyclassroom.com/typescript/typescript-basic-types>
 - <https://levelup.gitconnected.com/essential-typescript-types-get-started-with-typescript-5e20a5106c90>