

## Для ознакомления:

1. Установить [Xcode](#)
2. [Создание простого приложения](#)
3. [UIKit](#)
4. [Human interface guidelines](#)
5. Язык программирования: [Swift/Objective C](#).

## Требования:

1. Язык - [Swift/Objective C](#)
2. Архитектура - MVC
3. Код отдавать в любом git репозитории
4. min version OS 11
5. В примерах указан рекомендуемый дизайн, можно использовать свои UI элементы, но важно придерживаться [Human interface guidelines](#)

## Основное задание:

Реализовать приложение для создания и отображения заметок. Пользователь должен иметь возможность добавить заметку и просмотреть список всех заметок.

Структура приложения:

- экран списка заметок
- экран деталей и добавления заметки

### 1) Экран списка заметок

Отображать все заметки в вертикальном списке. Переход на добавление новой заметки реализовать через кнопку “+”. Для вывода списка на экран рекомендуем использовать [UITableView](#). При отображении заметок в списке обрезать текст, если его длина превышает 100 символов.

По нажатию на ячейку с заметкой переход на экран “детали заметки”.

По нажатию на кнопку “+” переход на экран “добавить заметку”

### 2) Экран добавления/деталей заметки

Экран имеет два состояния - добавление, детали

#### Добавление

Отображать поле для ввода текста и кнопку “сохранить”.

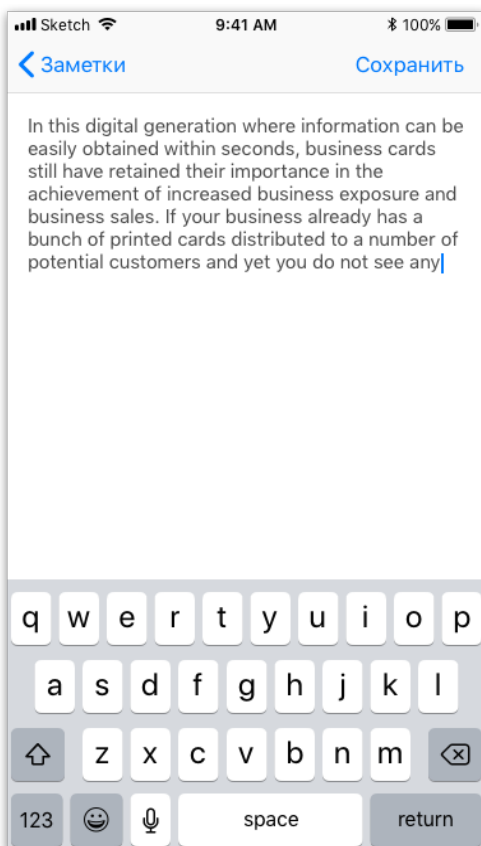
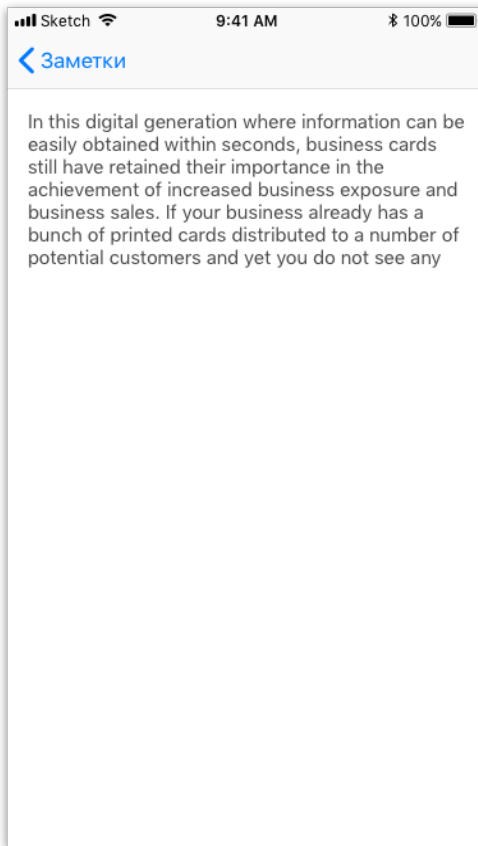
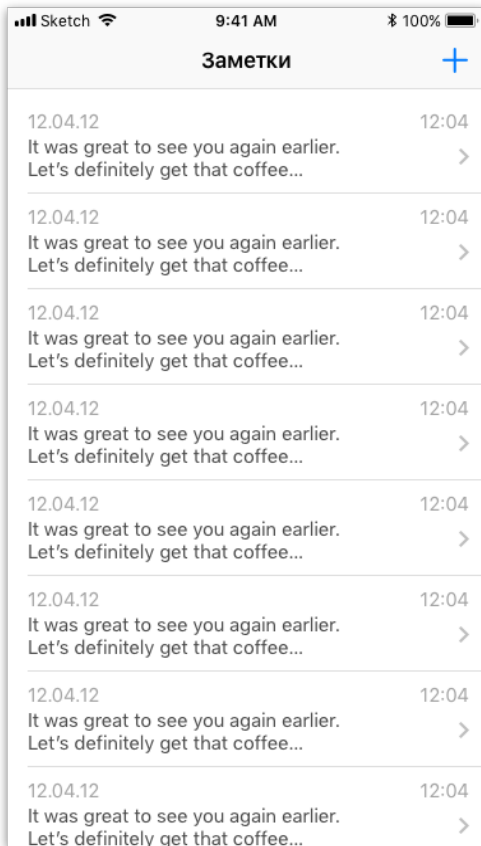
Пользователь должен иметь возможность ввести текст заметки и сохранить ее.

После успешного добавления заметки, пользователь должен вернуться в экран списка заметок с обновленными данными.

#### Детали

Отображать текст заметки без возможности редактировать.

Пример дизайна



## Дополнительные задания:

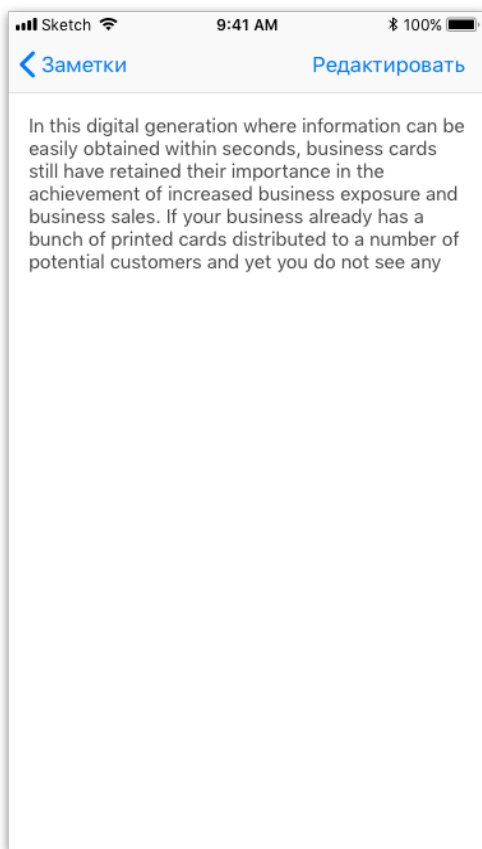
### Межсессионное хранение данных:

Пользователь должен иметь возможность сохранять и извлекать данные, без привязке к сессии. Для реализации данного функционала можно использовать сохранение в файл или любую базу данных ([CoreData](#), Realm, SQLite).

### Редактирование данных:

Пользователь должен иметь возможность редактировать заметку с помощью UITableViewRowAction и кнопки “редактировать” на экране списка заметок. По нажатию показывать экран добавления заметки с предзаполненными данными. Пользователь должен иметь возможность редактировать текст заметки и сохранить изменения.

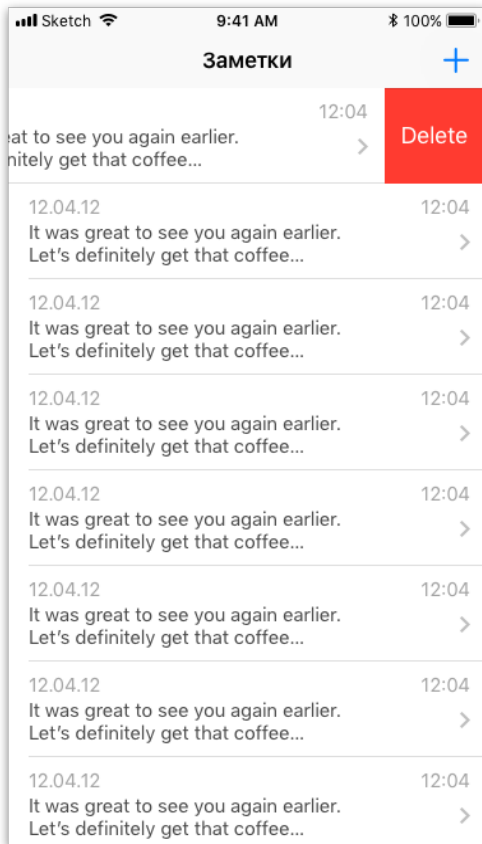
#### Пример дизайна



### Удаление данных:

Пользователь должен иметь возможность удалить заметку с помощью row action и кнопки “удалить” на экране списка заметок.

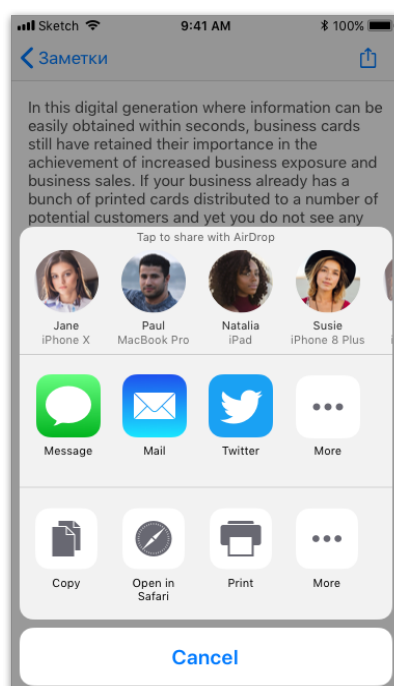
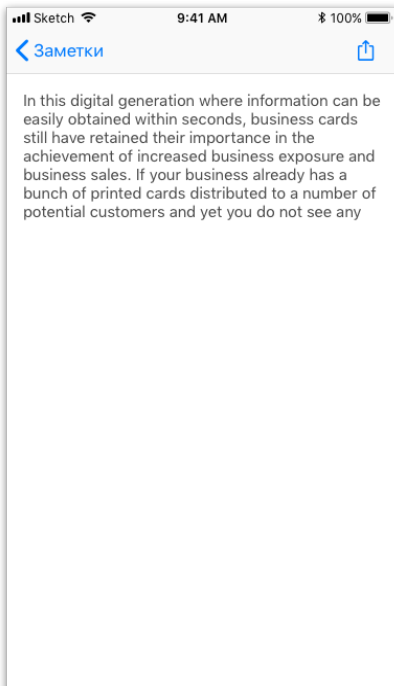
#### Пример дизайна



### Поделиться заметкой:

Пользователь должен иметь возможность поделиться текстом заметки с помощью кнопки “поделиться” на экране деталей заметок. По нажатию на кнопку показывать нативный UI для шеринга данных.(UIActivityViewController)

Пример дизайна



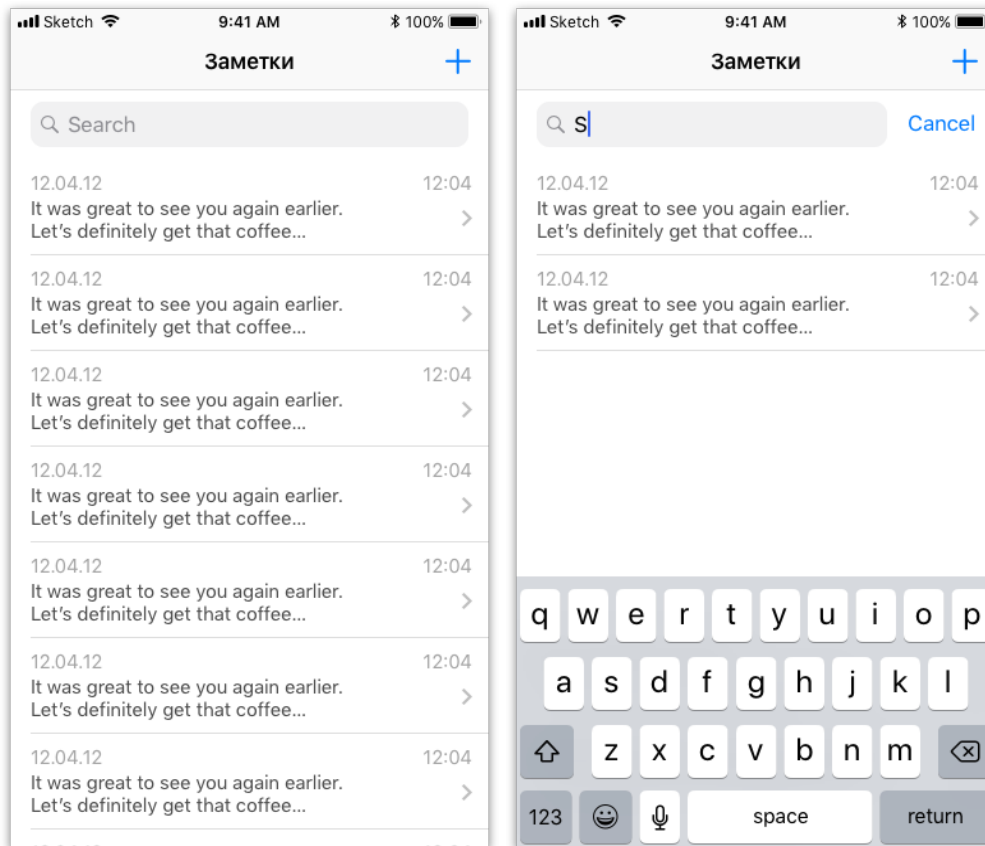
### Поиск по заметкам:

Добавить элемент UISearchBar вверху экрана с кнопкой “cancel”.

#### **Сценарий использования:**

- 1) пользователь ввел символ: отобразить отфильтрованный список по поисковой строке
- 2) пользователь нажал кнопку “cancel”: убрать клавиатуру.

Пример дизайна



### Пагинация:

Реализовать постраничную выгрузку данных из БД (по 20 заметок). Следующая страница должна быть загружена и отображена при достижении конца списка.

### Сортировка:

Добавить кнопку “сортировать” на экране списка заметок.

По нажатию показывать список вариантов сортировки. Для вывода вариантов сортировки на экран рекомендуем использовать [UIAlertController](#) (actionSheet style). Варианты сортировки: По алфавиту, по дате создания/изменения.

Пример дизайна

