

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

Реферат

Тема: Обзор предметной области

Студент гр. 4303

Ахриев Р.А.

Преподаватель

Санкт-Петербург

2019

ВВЕДЕНИЕ

Мобильные телефоны играют важную роль в нашем, современном мире. Практически у каждого человека сейчас есть мобильный телефон. Смартфоны имеют широкий функционал. Пользователи используют смартфоны для совершения звонков, просмотра видео, прослушивания аудиокниг, съемки. Мобильные приложения, создаваемые для смартфонов позволяют существенно расширить базовый функционал телефона. Существуют различные приложения для самых разнообразных целей.

В данный момент более 99% рынка занимают операционные системы iOS (14%) и Android (85%).

Цель данной работы состоит в реализации фреймворка, значительно ускоряющего создания приложений для операционной системы iOS.

Для достижения поставленной цели необходимо выполнить следующие задачи:

- Определить подход создания пользовательского интерфейса
- Произвести поиск инструментов, позволяющих оптимизировать создание iOS приложения
- Выполнить сравнительный анализ отобранных инструментов по указанным критериям
- Сформировать перечень требований к разрабатываемому инструменту
- Описать алгоритмы и механизмы работы разрабатываемого инструмента
- Разработать инструмент, оптимизирующий выполнение типичных задач, с которым сталкивается разработчик в ходе создания iOS приложения.

Объектом исследования является API, предоставляемый компанией Apple для создания iOS приложения.

Предметом исследования является фреймворк, позволяющий оптимизировать создание iOS приложений.

1. ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ

1.1. Необходимые критерии

Для тестирования API от компании Apple, необходимо воспользоваться одним из ключевых компонентов UITableView – компонент для отображения списка элементов (новостной ленты, списка задач).

В ходе использования компонента необходимо выяснить следующее:

- Встраиваемость компонента в архитектуру MVC
- Удобство использования
- Недостатки компонента для дальнейшего устранения
- Найти готовые решения оптимизирующие работу UITableView

1.2. Обзор существующих решений.

Существует ряд библиотек и компонентов решающих различные проблемы UITableView:

TableKit- библиотека позволяющая использовать декларативный способ создания интерфейса.

MGSwipeTableViewCell – решает проблему создания инициализации кнопок при жесте свайпа.

SwiftReorder – библиотека решающая проблему пересортировки ячеек таблицы.

ParallaxTableViewHeader – зависимость, позволяющая создавать заголовок таблицы с эффектом паралакс.

Существует ряд библиотек, решающих различные частные случаи и проблемы связанные с хэшированием, переиспользованием данных.

1.3. Анализ проблем существующих решений.

Основной проблемой использования данных библиотек является требование прямой зависимости компонента от данных решений, что препятствует созданию гибкого компонента с возможностью комбинирования функционала готовых инструментов.

1.3. Вывод.

Существует ряд инструментов, решающих локальные проблемы отдельных компонентов, как было рассмотрено на примере UITableView, основным недостатком которого является необходимость использования паттерна «Делегат» напрямую в контроллере, что является нарушением паттерна MVC и приводит к смешиванию слоя данных и слоя представления.

В следствие вышеизложенного принято решение создать фреймворк, решающих основные повседневные задачи iOS разработчика и позволяющий использовать готовые решения через слой интерфейса, что приведет к возможности замены различных компонентов без нарушения целостности проекта.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Список библиотек [Электронный ресурс]. URL:

<https://github.com/vsouza/awesome-ios> —дата обращения: 10.12.2019).