

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и информационных технологий

**РАЗРАБОТКА КЛИЕНТ-СЕРВЕРНОГО МОБИЛЬНОГО
ПРИЛОЖЕНИЯ ПОД ОПЕРАЦИОННЫЕ СИСТЕМЫ IOS И ANDROID
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 4 курса 421 группы
направления 09.03.01 Информатика и вычислительная техника
факультета КНиИТ
Фоминова Сергея Михайловича

Научный руководитель

ассистент

М. А. Боринос

Заведующий кафедрой

доцент, к. ф.-м. н.

Л. Б. Тяпаев

Саратов 2019

ВВЕДЕНИЕ

Сегодня мир движется семимильными шагами и то, что раньше человек даже и представить не мог – сейчас это лишь повседневная вещь, которая улучшает человеческое качество жизни и повышает производительность на работе. Ультрабуки, умные дома, часа и смарфтоны - всего лишь часть современного человека. Особенно заметен стремительный взлет рынка мобильных приложений за последние 10 лет. Первоначально они использовались для быстрой проверки электронной почты или короткой связи, но, начиная с поглощением малой компании Android Inc коорпорации Google и до представления первого iPhone 4 от Apple, они начинают применяться и в других областях, таких как игры, общение, просмотр видео и использование интернета. В настоящее время рынок неуклонно растет и развивается, предоставляя для разработчиков все новые решения для построения мобильных приложений.

Актуальной задачей является создание мобильного приложения для iOS, не имеющего аналога на веб-форуме 2channel и создание приложения для Android, с обновленным интерфейсом для пользователя.

Целью данной работы является создание мобильного приложения для веб-форума 2channel, в котором пользователь сможет читать интересные тематики и обсуждения, просматривать прикрепленные изображения или мультимедиа файлы от других пользователей, а также настраивать навигационное меню и добавлять понравившиеся обсуждения в избранное.

Поставленная цель определила следующие задачи:

1. Изучение литературы по разработке мобильного приложения
2. Реализация взаимодействия мобильного приложения с API веб-форума
3. Проектирование системной архитектуры в мобильном приложении
4. Использование Material Design по разработке дизайна мобильного приложения
5. Подключение мобильного приложения к БД

Бакалаврская работа состоит из введения, 5 разделов, заключения и списка использованных источников.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Описание форума» посвящен описанию веб-форума состоящего из тематических разделов, где они впервые появились и какие сейчас популярны в СНГ регионе.

В данных веб-форумах пользователи могут обсуждать различные темы, не регистрируясь на сайте, а при комментарии присутствует возможность добавлять изображения или мультимедиа файлы с расширением MP4 или WebM.

Самым известным веб-форумом подобной тематики является 2channel. Также он предоставляет API документацию для отображения данных с сайта.

Второй раздел «Анализ разработки мобильного приложения» посвящен выбору инструментов при создании мобильных приложений. В основном они делятся на две категории: мультиплатформенная или нативная разработка и кросс-платформенная. В каждой из них есть свои плюсы и минусы.

Для разработки мобильного приложения была выбрана мультиплатформенное решение, т.е. создание двух кодовых баз для iOS и для Android.

Третий раздел «Описание инструментария и проектирование архитектуры» посвящен разбору инструментов, необходимые для создания приложений и способов проектирования кодовой базы. В подразделе 3.1 «Инструменты для работы с HTTP запросами в iOS и Android приложения» приводится обзор работы сетевых запросов при создании клиент-серверных приложений. В подподразделе 3.1.1 «Использование Alamofire в iOS приложения» описывается функционал библиотеки Alamofire и ее базовое применение в разработке iOS приложения. В подподразделе 3.1.2 «Использование OkHttp3 и Retrofit в Android приложения» описывается функционал библиотек OkHttp3 и Retrofit и их совместное применение для сетевых запросов в создании Android приложения. [2]

В подразделе 3.2 «Дополнительные инструменты, используемые при разработке iOS приложения» приводится обзор применяемых библиотек в

разрабатываемом iOS приложении. В подподразделе 3.2.1 «CoreData» описывается применение мобильной базы данных в iOS приложении, которая имеет специальный интерфейс в Xcode для создания требуемых сущностей. Основную работу производит класс Persistent Store Coordinator, который позволяет проводить операции CRUD в background потоке. [3] В подподразделе 3.2.2 «Kingfisher» описывается библиотека для работы с изображениями в iOS приложении и ее возможности. В подподразделе 3.2.3 «AVFoundation и OGVKit» приводится стандартный инструментальный и дополнительная библиотека для работы с мультимедиа файлами с форматами MP4 и WebM в iOS приложении.

В подразделе 3.3 «Дополнительные инструменты, используемые для создания Android приложения» приводится обзор применяемых библиотек в разрабатываемом Android приложении. В подподразделе 3.3.1 «Room» описывается применение мобильной базы данных в Android приложении, которая является оберткой для SQLite, встраиваемой в приложение. Функционал данной библиотеки представляет из себя сам класс БД, через который происходит вызов методов, размещаемых в DAO интерфейсе. Сущности для работы с БД помечаются аннотацией @Entity. В подподразделе 3.3.2 «RxJava2» описывается библиотека для работы с многопоточностью в Android приложении. Чаще всего она применяется для работы, реализующие сетевые запросы на сервер или для фильтрации каких-либо данных. В подподразделе 3.3.3 «Glide» описывается библиотека для работы с изображениями в Android приложении и ее возможности. В подподразделе 3.3.4 «ExoPlayer» приводится разбор библиотеки для работы с мультимедиа файлами в Android приложении. В подподразделе 3.3.5 «Moxy» рассматривается библиотека для устранения потери данных при повороте пользовательского смартфона.

В подразделе 3.4 «Управления зависимостями проекта в мобильном приложении» приводится обзор менеджера пакетов, которые необходимы для подключения рассмотренных ранее библиотек. В подподразделе 3.4.1 «CocoaPods в iOS приложении» описывается использование менеджера пакетов

CocoaPods и его необходимые терминальные команды при установке библиотек.

[1] В подподразделе 3.4.2 «Gradle в Android приложении» рассматривается система автоматической сборки проектов, называемой Gradle, с помощью которой также можно подключать необходимые библиотеки.

Четвертый раздел «Описание инструментария и проектирование архитектуры» посвящен разбору архитектуры кодовой базы, применяемых в создании мобильных приложений, а также рассматриваются способы проектирования; монолитное, модульное и сервис-ориентированное.

В подразделе 4.1 «Архитектура MVP, MVVM, MVC» приводится обзор проектирования систем с использованием стандартных паттернов, а также рассматриваются их отличия друг от друга. В подподразделе 4.2 «Выбор CleanArchitecture» рассматриваются плюсы и минуса выбора данной архитектуры, ее описание и внедрение в каждый из проектов бакалаврской работы.

Пятый раздел «Создание мобильного приложения с использованием iOS и Android SDK's» посвящен разработке мобильного приложения под две операционные системы с использованием IDE, таких как Xcode и Android Studio.

В подразделе 5.1 «Реализация нижней панели в iOS приложении» и в подразделе 5.2 «Реализация нижней панели в Android приложении» описывается разработка нижней панели в обоих приложениях. В iOS используется Custom View элемент и класс UICollectionView, который может отображать горизонтальный список элементов. [4] В Android приложении применяется BottomNavigationViewEx элемент, который расширяет базовый View элемент – BottomNavigationView.

В подразделе 5.3 «Реализация swipe жеста экрана в iOS приложении» и в подразделе 5.4 «Реализация swipe жеста экрана в Android приложении» описывается разработка swipe жеста в главном экране приложения. Swipe жест – это специальный жест, при котором пользователь кладет палец на экран

смартфона или планшета и ведет его в нужном направлении по экрану. В iOS приложении для этого применяются два View компонента – Container View, внутри которых располагаются два UITableView, способных отображать вертикальный список предоставляемых данных. В Android приложении используется ViewPager компонент, который имеет встроенную обработку пользовательского жеста. За отображением данных в виде вертикального списка применяется RecyclerView элемент. [5]

В подразделе 5.5 «Создания экрана закладок в Android приложении» и в подразделе 5.6 «Создания экрана закладок в iOS приложении» описывается разработка экрана Закладок, в котором отображаются пользовательские сохраненные обсуждения. В Android приложении используется RecyclerView элемент с подключенной библиотекой Room. Внутри каждой ячейки отображаемым View элементе используется таймер, который каждый 30 секунд обновляет количество новых сообщений. В iOS приложении реализуется аналогичный функционал, но с применением UITableView элемента для отображения данных, а для работы с БД используется CoreData.

В подразделе 5.7 «Transition анимация изображения в iOS приложении» и в подразделе 5.8 «Transition анимация изображения в Android приложении» рассматривается разработка анимации масштабирования изображения с использованием iOS и Android SDK's. В iOS приложении используется метод animationController(), который создает между двумя экранами промежуточный экран, где и происходит анимация масштабирования изображения. В Android приложении применяется TransitionManager класс, представляющий из себя реализацию анимации View элементов, id которых помечаются и в дальнейшем располагаются в контейнере, который передается между двумя экранами.

В подразделе 5.9 «Работа с мультимедиа файлами в iOS приложении» и в подразделе 5.10 «Работа с мультимедиа файлами в Android приложении» описывается работа с отображением видео в мобильном приложении. В iOS приложении используется стандартный класс из iOS SDK, который может

отображает MP4 формат. Для отображения WebM формата применяется OGVPlayer, в котором уже реализован базовый функционал: кнопки проигрывания и паузы, текст с общим временем, с временем, прошедшем со старта видео, а также нижнего ползунка. В Android приложении используется библиотека ExoPlayer, которая умеет работать с двумя рассматриваемыми видео-форматами. Она реализует идентичный функционал, что в iOS приложении.

В подразделе 5.11 «Реализация экрана просмотра комментариев в текущем обсуждении в iOS приложении» и в подразделе 5.12 «Реализация экрана просмотра комментариев в текущем обсуждении в Android приложении» рассматривается разработка экрана просмотра комментариев в текущем обсуждении, а также возможность просмотра сообщений на ответный комментарий. В iOS приложении используется UITableView элемент, для отображения вертикального списка элементов, а также CustomView элемент, который появляется при просмотре на отвечаемый комментарий. В Android приложении реализуется идентичный функционал, только вместо CustomView используется DialogFragment, который автоматически затемняет предыдущий экран при просмотре на отвечаемый комментарий поста.

В подразделе 5.13 «Реализация навигационного меню в iOS приложении» и в подразделе 5.14 «Реализация навигационного меню в Android приложении» описывается разработка навигационного меню, с помощью которого пользователь может сохранять интересующие его тематики и удалять. В iOS приложении используется CustomView, называемый NavigationContainer, который внутри себя содержит ячейки, реализованные таблично с использованием класса UICollectionView. Для обработки удаления ячеек применяется анимация с применением класса CABasicAnimation. В Android приложении используется стандартный View компонент – BottomSheetView, внутри которого реализуется RecyclerView элемент. При удалении применяется аналогичная анимация с использованием XML языка, содержащий в себе атрибут rotate. [6]

Каждый из экранов включает в себя дополнительную ячейку со знаком плюс, при нажатии которого открывается дополнительное окно, с выбором тематик, предлагаемых на форуме.

В рамках бакалаврской работы было разработано мобильное приложение под операционные системы iOS и Android, с помощью которых пользователь может просматривать содержимое веб-форума.

ЗАКЛЮЧЕНИЕ

При выполнении данной бакалаврской работы была достигнута цель создания мобильного приложения под две операционные системы: iOS и Android, где пользователь может просматривать форум, сохранять интересующие его обсуждения в закладки, настраивать навигационное меню, смотреть видео и изображения прикрепленными другими людьми, и т.д. Так же в работе было выполнено ряд задач - приобретение навыков работы в Android Studio и Xcode, современное использования библиотек, таких как Retrofit, OkHttp, Glide, RxJava, Alamofire, Kingfisher, Room и CoreData. Стоит отметить, что созданное приложение может распространяться и на другие устройства, на которых минимальная операционная система Android превышает или равняется 5.0, а на iOS - с 12.2 или выше. Код реализации приложен на диске в конце бакалаврской работы.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Using CocoaPods [Электронный ресурс] URL: <https://guides.cocoapods.org/using/using-cocoapods.html> (дата обращения 16.05.2019) Загл с экрана Яз англ
2. OkHttp (Электронный ресурс) URL: <https://square.github.io/okhttp/> (Дата обращения 05.05.2019) Загл с экрана Яз англ
3. Core Data [Электронный ресурс] URL: <https://developer.apple.com/documentation/coredata> (дата обращения 13.05.2019) Загл с экрана Яз англ
4. UICollectionView [Электронный ресурс] URL: <https://developer.apple.com/documentation/uikit/uicollectionview> (дата обращения 06.05.2019) Загл с экрана Яз англ
5. RecyclerView [Электронный ресурс] URL: <https://developer.android.com/reference/android/support/v7/widget/RecyclerView> (дата обращения 04.05.2019) Загл с экрана Яз англ
6. The Busy Coder's Guide to Android Development; CommonsWare LLC, Mark Murphy, 2018, 4419с