

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
Национальный исследовательский университет «МИЭТ»
Кафедра информатики и программного обеспечения вычислительных систем

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к выпускной работе по теме:

Разработка мобильного приложения сопровождения учебного
процесса студентов в системе ОРИОКС
(Шифр МП СУПС)

Руководитель проекта
доцент, к.п.н.

_____ Е.Л. Федотова

Исполнитель
студент гр. МП-45

_____ О.А. Фаткуллин

Москва 2018

РЕФЕРАТ

Ключевые слова: разработка мобильного приложения, разработка для Android, язык программирования Kotlin, принципы проектирования SOLID, чистая архитектура, Material дизайн.

Отчет содержит 24 стр., 5 рис., 2 табл., 8 источников, 2 прил.

СОДЕРЖАНИЕ

| | |
|---|----|
| Введение | 7 |
| 1 Исследовательский раздел..... | 9 |
| 1.1 Почему мобильное приложение? | 9 |
| 1.2 Обзор аналогичных решений..... | 9 |
| 1.2.1 Сайт ОРИОКС | 11 |
| 1.2.2 Приложение “Расписание для МИЭТ” | 11 |
| 1.2.3 Приложение “ОРИОКС Live” | 12 |
| 1.2.4 Telegram-бот “Open Orioks” | 14 |
| 1.2.5 Итоги | 15 |
| 1.3 Обзор мобильных платформ..... | 16 |
| 1.4 Исследование структуры ОРИОКС | 18 |
| 1.5 Входные и выходные данные | 18 |
| 1.6 Постановка задачи | 18 |
| 2 Конструкторский раздел..... | 19 |
| 2.1 Выбор языка программирования..... | 19 |
| 2.2 Выбор среды разработки | 19 |
| 2.3 Выбор стека технологий | 19 |
| 2.4 Архитектура и алгоритм работы МП СУПС | 19 |
| 2.5 Разработка пользовательского интерфейса | 19 |
| 3 Технологический раздел..... | 20 |
| 3.1 Разработка под Android | 20 |
| 3.1.1 Система контроля версий | 20 |
| 3.1.2 Структура проекта..... | 20 |
| 3.1.3 Разработка пользовательского интерфейса | 20 |
| 3.1.4 Работа с данными | 20 |

| | | |
|-------|--|----|
| 3.1.5 | Безопасность | 20 |
| 3.1.6 | Многопоточность и работа с сетью | 20 |
| 3.2 | Сборка приложения | 20 |
| 3.2.1 | Система автоматической сборки Gradle | 20 |
| 3.2.2 | Подключение зависимостей | 20 |
| 3.2.3 | Подпись приложения | 20 |
| 3.2.4 | Публикация приложения | 20 |
| 3.2.5 | Непрерывная интеграция | 20 |
| 3.3 | Тестирование и отладка в Android разработке | 20 |
| 3.3.1 | Методы тестирования программного обеспечения | 20 |
| 3.3.2 | Методология разработки TDD | 20 |
| 3.3.3 | Тестирование пользовательского интерфейса | 20 |
| 3.3.4 | Отладка в IntelliJ IDEA | 20 |
| | Заключение | 21 |
| | Список использованных источников | 22 |
| | Приложение А Текст программы | 23 |
| | Приложение Б Руководство программиста | 24 |

ОПРЕДЕЛЕНИЯ

В настоящем отчете применяют следующие термины с соответствующими определениями.

Push-уведомления — способ рассылки уведомлений. У конечного пользователя такие уведомления выглядят как небольшое всплывающее окошко на экране телефона или в браузере.

Браузер — программное обеспечение для просмотра информации из сети.

Веб-приложение — клиент-серверное приложение, в котором клиент взаимодействует с сервером при помощи браузера, а за сервер отвечает — веб-сервер.

Кэширование — один из способов оптимизации приложений. Заключается в том, чтобы сохранять на некоторое время и переиспользовать результаты медленных операций..

Мобильное приложение — программное приложение, предназначенное для работы на смартфонах или планшетных компьютерах.

Платформа — среда выполнения, в которой должен выполняться фрагмент программного обеспечения или объектный модуль с учётом накладываемых средой ограничений и предоставляемых возможностей.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

ОРИОКС — организация распределенного информационного обмена в корпоративных средах.

ПК — персональный компьютер.

API — application programming interface — внешний интерфейс взаимодействия с приложением.

БД — база данных.

HTML — hypertext markup language — язык гипертекстовой разметки.

ОС — операционная система.

ВВЕДЕНИЕ

В современных учебных заведениях всё чаще практикуют использование систем сопровождения учебного процесса, которые позволяют учащимся видеть свои оценки, получать домашние задания, узнавать о событиях учебного процесса. В МИЭТе такой системой является ОРИОКС. Рассмотрим существующие недостатки ОРИОКСа.

Слабым местом ОРИОКСа является невозможность оперативного оповещения учащихся о событиях учебного процесса. Это обусловлено тем, что ОРИОКС представляет из себя веб-приложение и подразумевает регулярное посещение через браузер. Если же студент долгое время не заходит в систему, что бывает довольно часто, то он может пропустить информацию о важных событиях, например о пересдачах. Так же, в ОРИОКС невозможна установка напоминаний о предстоящих событиях.

Еще одна проблема веб-приложений – неудобство использования на мобильных устройствах. Большая часть студентов посещает ОРИОКС именно с мобильных устройств, и, хотя, дизайн сайта оптимизирован для работы на небольших экранах, всё же мобильные устройства накладывают свои ограничения. Скорость интернет соединения, чаще всего, ниже чем на стационарных ПК, кроме того, соединение может часто разрываться, что приводит к дискомфорту при использовании веб-приложения, т.к. для каждой страницы браузер должен загрузить, помимо данных, разметку и таблицу стилей.

Все эти недостатки можно устранить, создав мобильное приложение, которое будет получать данные от сервера ОРИОКС через API). Это позволит запрашивать и получать только ту информацию, которая нужна для работы приложения в конкретный момент времени, так как приложению не нужно скачивать таблицу стилей и разметку. В виду малого потребления трафика приложение сможет работать на смартфоне в фоновом режиме и отображать уведомления и напоминания в тот момент когда они только пришли.

Удобство работы с системой ОРИОКС при нестабильном интернет соединении тоже повысится за счёт меньшего объёма пересылаемых данных. Кроме того,

приложение может кэшировать данные и использовать их даже при отсутствии интернет соединения.

На данный момент нет ни одного приложения работающего с ОРИОКС через API. Более того, нет ни одного приложения, позволяющего студентам использовать все возможности ОРИОКС. Целью данной работы является создание такого приложения, чтобы повысить оперативность оповещения студентов о событиях учебного процесса.

1 Исследовательский раздел

1.1 Почему мобильное приложение?

Мобильные приложения могут быть инструментом для быстрой доставки информации, чего нельзя добиться при помощи обычного веб-приложения.

Индустрия мобильных устройств очень быстро развивается. На смену старым устройствам приходят более новые, современные и обладающие большим спектром возможностей. Количество пользователей с каждым годом растёт. Сейчас смартфоны есть почти у всех студентов и они редко с ними расстаются надолго. Конечно, важную роль играет программная составляющая — мобильные приложения. Они существуют совершенно разной направленности:

- развлекательные (игры, музыкальные и видео проигрыватели и т.д.);
- коммуникационные (мессенджеры, навигаторы и т.д.);
- справочные (словари, базы знаний);
- прикладные (все остальные от графического редактора до калькулятора).

Кроме того, популярность мобильных приложений повлекла за собой появление мощных инструментов разработки, большого количества библиотек и фреймворков, что, в свою очередь, сделало разработку приложений быстрой, лёгкой и продуктивной.

1.2 Обзор аналогичных решений

Был произведен поиск существующих решений в Windows Phone Store, Google Play, Apple App Store, на GitHub, и было найдено 3 аналога. К ним можно, так же, добавить сам сайт ОРИОКС.

Рассмотрим преимущества и недостатки каждого решения по отдельности. В качестве критериев будем брать:

- способ получения данных;
- возможность push-уведомлений;

- возможность просмотра расписания;
- возможность просмотра текущих предметов;
- возможность просмотра успеваемости;
- возможность просмотра списка долгов;
- возможность просмотра списка пересдач;
- наличие графического интерфейса;
- оффлайн доступ.

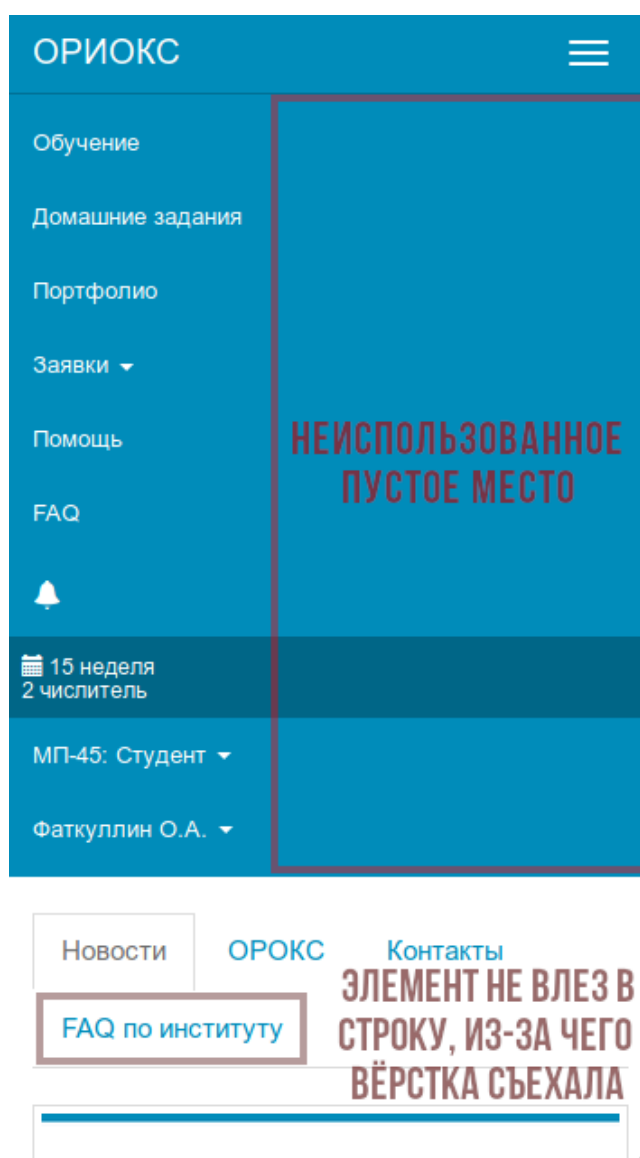


Рисунок 1.1 — Главная страница ОРИОКС с открытым меню

1.2.1 Сайт ОРИОКС

Платформа: Браузер.

Из плюсов: сайт ОРИОКС получает информацию напрямую из БД. Это позволяет запрашивать только ту информацию, которая нужна в данный момент для отображения страницы. Можно, так же, отметить наличие всех перечисленных выше возможностей, за исключением просмотра расписания (но его можно посмотреть на сайте МИЭТ) [2]. Графический интерфейс присутствует, но не полностью адаптирован для мобильных устройств, то есть в некоторых местах элементы интерфейса не помещаются на экране, а в других — наоборот слишком много неиспользованного места (см. рис. 1.1).

К минусам можно отнести отсутствие push-уведомлений, невозможность просмотра информации без интернет соединения и необходимость загружать таблицы стилей и HTML разметку для просмотра страницы.

1.2.2 Приложение “Расписание для МИЭТ”

Платформа: Android. Дополнительная информация: около тысячи установок; рейтинг на Google Play — 4.7 из 5; дата последнего обновления — 22.04.2018.

Приложение предназначено для просмотра новостей МИЭТ и расписания любой группы с возможностью скачать его и использовать в оффлайн-режиме. Раньше присутствовал функционал просмотра успеваемости, но из-за изменений на сайте ОРИОКС этот функционал стал недоступен [3].

Получение расписания реализовано через API, это плюс. Для получения текущей успеваемости использовался синтаксический анализ сайта. При таком подходе любое изменение в таблице стилей или HTML разметке сайта приводит к неработоспособности приложения, что и случилось.

Из требуемых возможностей присутствует просмотр и кэширование расписания, это позволяет просматривать его без интернет-соединения. Просмотр текущих предметов, успеваемости, списка долгов и пересдач невозможен.

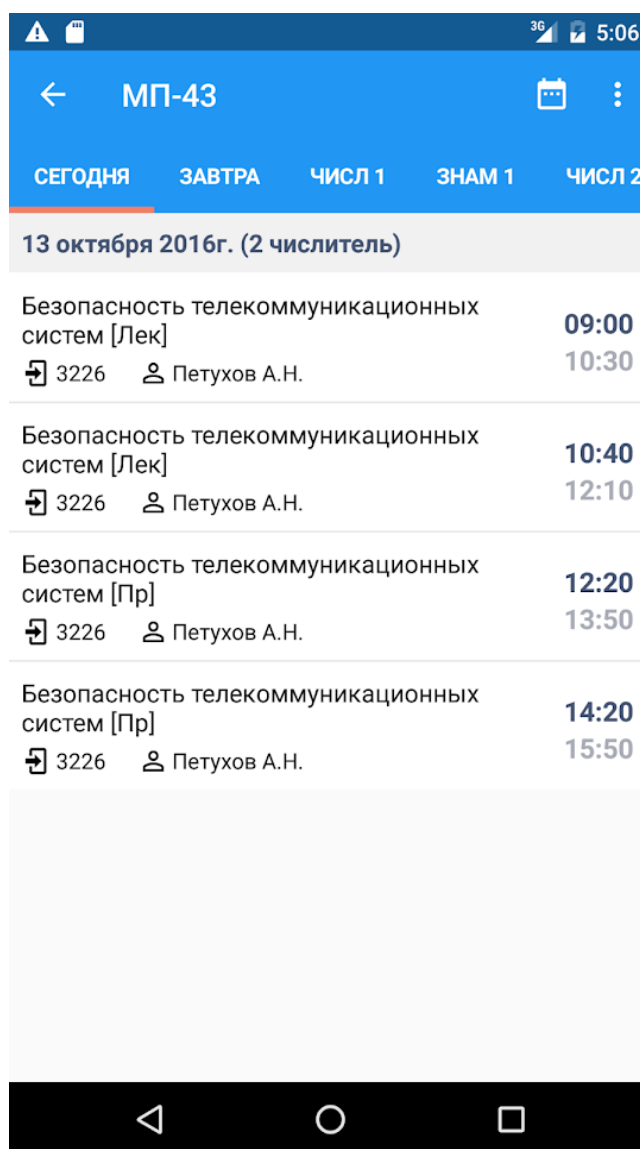


Рисунок 1.2 — Экран расписания в приложении “Расписание для МИЭТ”

Графический интерфейс присутствует, но не соответствует требованиям Material Design. Например, слишком маленькие отступы от краёв экрана (см. рис. 1.2).

1.2.3 Приложение “ОРИОКС Live”

Платформа: Android. Дополнительная информация: около тысячи установок; рейтинг на Google Play — 3.9 из 5; дата последнего обновления — 30.09.2015.

Приложение предназначено для просмотра успеваемости, списка контрольных мероприятий и информации о преподавателях [4].

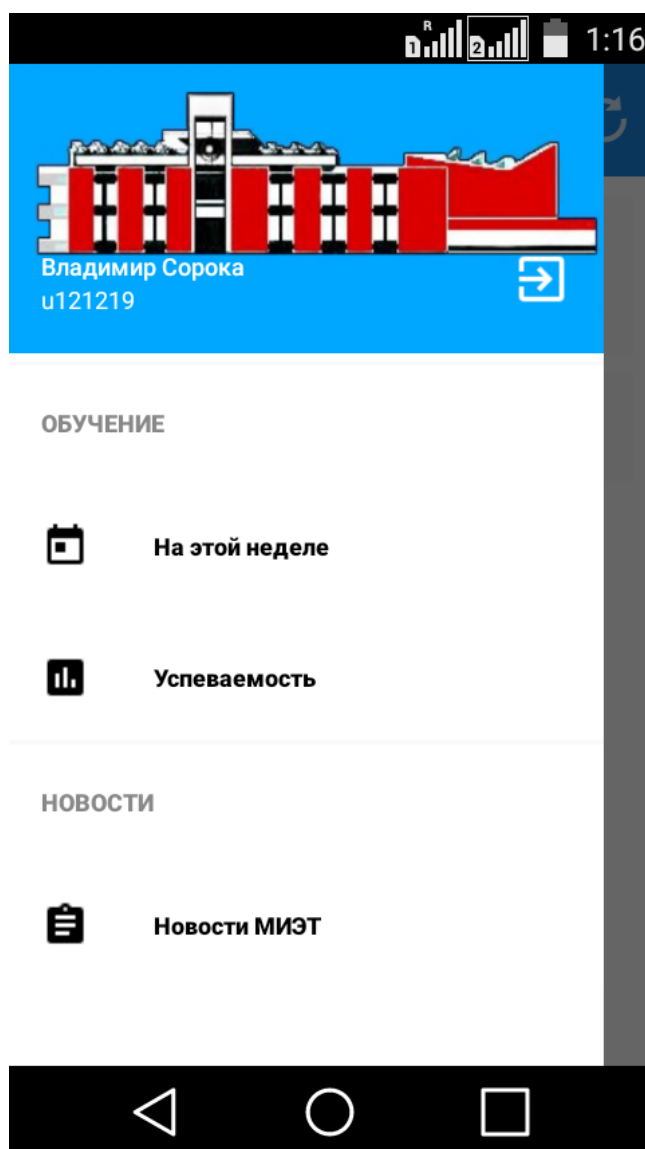


Рисунок 1.3 — Главный экран с открытым меню в приложении “Ориокс Live”

Данные получаются при помощи непубличного программного интерфейса, но интерфейс был удалён, т.к. был сделан неофициально и работа с ОРИОКС стала невозможна. Приложение не обновлялось с 2015 года и на данный момент не работает.

Заявленный функционал проверить не удалось из-за невозможности авторизации, поэтому все эти возможности отметим как отсутствующие. Рабочей осталась только возможность просмотра новостей.

Графический интерфейс присутствует, но не соответствует требованиям Material Design. Неправильные отступы, слишком контрастные цвета (см. рис. 1.3).

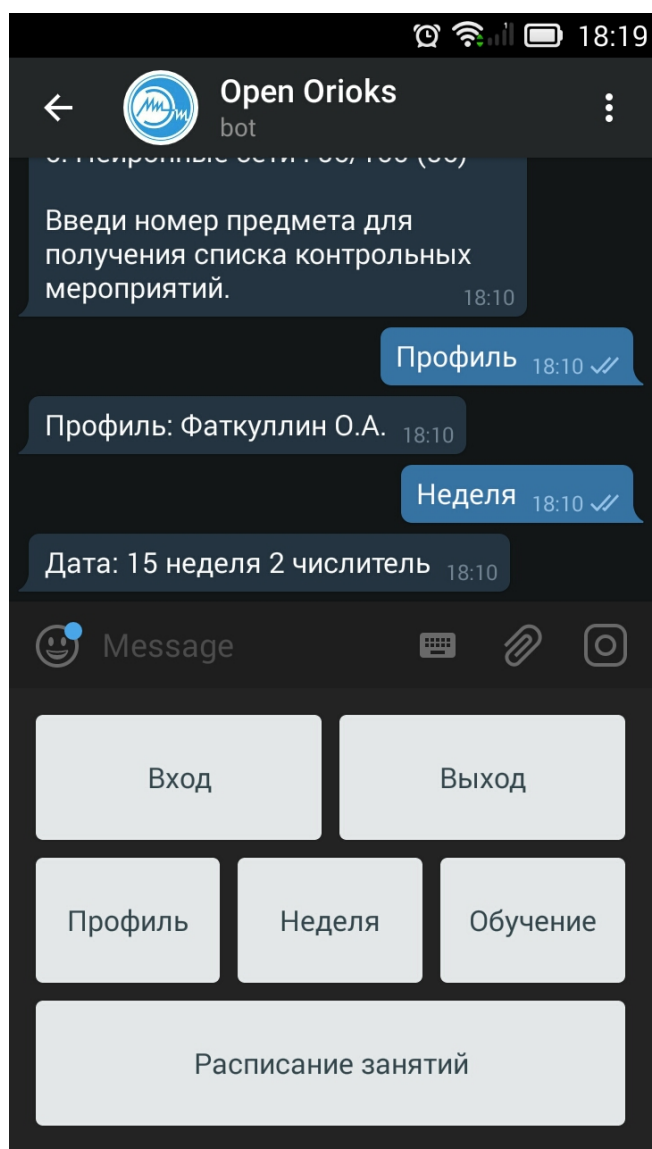


Рисунок 1.4 — Интерфейс управления Telegram-ботом “Open Orioks”

1.2.4 Telegram-бот “Open Orioks”

Платформа: Telegram. Дополнительная информация: последнее обновление — 12.10.2017.

Бот позволяет просматривать расписание на день, текущую успеваемость и список контрольных мероприятий по каждому предмету.

Для получения данных используется синтаксический анализ (как и в пункте 1.2.2), но за счёт того, что данные всех студентов обновляются раз в полчаса и сохраняются в хранилище бота, скорость получения данных конечным пользователем сравнима со скоростью получения данных из БД [5].

Собственного графического интерфейса нет. Взаимодействие с ботом производится через текстовые сообщения в мессенджере Telegram. Использование при отсутствии интернета невозможно, но можно просматривать предыдущие ответы бота, что можно считать частичным кэшированием.

1.2.5 Итоги

По итогам обзора аналогичных решений была построена таблица 1.1. Как видно из таблицы, нет ни одного решения, которое бы соответствовало всем необходимым параметрам, что еще раз доказывает актуальность задачи.

Таблица 1.1

Сравнение аналогичных решений

| Критерий | Сайт ОРИОКС [2] | Приложение “Расписание для МИЭТ” [3] | Приложение “ОРИОКС Live” [4] | Бот “Open Orioks” [5] | МП СУПС |
|-------------------------------|--------------------|--|------------------------------------|--------------------------|---------|
| Скорость получения информации | Средняя | Высокая | Низкая | Средняя | Высокая |
| Push-уведомления | — | — | — | + | + |
| Расписание | ± | + | — | ± | + |
| Текущие предметы | + | — | — | + | + |
| Успеваемость | + | — | — | + | + |
| Долги | + | — | — | — | + |
| Пересдачи | + | — | — | — | + |
| Графический интерфейс | + | — | — | — | + |
| Оффлайн доступ | — | + | — | ± | + |

+ – указанная возможность присутствует

± – указанная возможность частично присутствует

— – указанная возможность отсутствует

В приложении “Расписание для МИЭТ” есть кэширование, но работает оно только для расписания, так как остальные функции недоступны. Только Telegram-бот “Open Orioks” предоставляет возможность push-уведомлений, но не обладает всеми требуемыми возможностями, т.к. при увеличении количества выполняемых задач, бот становится неудобен в использовании.

1.3 Обзор мобильных платформ

Смартфоны, в виде, похожем на нынешний, появились в 2007 году, когда Стив Джобс на выставке “Macworld Conference & Expo” показал миру iPhone. Все существующие телефоны мгновенно стали устаревшими. Рынок операционных систем для смартфонов был практически пуст и было понятно, что один iPhone его не покрывает. В Google в это время только думали о выпуске своего телефона, но о смартфоне речи не шло.

За четыре года до этого, в 2003 году, Энди Рубин и три его друга (Ник Сир, Крис Уайт и Рич Майнер) решили создать операционную систему для носимых устройств, которые могли бы подстраиваться под нужды пользователя. Они основали компанию Android Inc. и назвали свою операционную систему Android. Никто тогда не оценил этот стартап и в 2005 году компания была на грани банкротства, но Энди Рубин смог убедить Google, которая занималась скупкой инновационных проектов, в том, что у Android есть будущее. Так и оказалось.

В 2007 году Google вспоминает, что они покупали стартап, направленный на разработку операционной системы для переносных устройств. Команда Android была только рада заняться разработкой ОС, но нужно было найти производителя смартфонов, который был бы готов выпустить на рынок устройство с совершенно новой ОС. У Nokia уже была своя ОС — Symbian. Motorola в это время была ослеплена успехом Razr и вряд ли обратила бы внимание на Android. Оставались еще LG и HTC, но LG уже решили развивать Windows Mobile вместе с Microsoft, поэтому выбор пал на HTC. HTC была рада сотрудничеству с большой компанией и могла обеспечить быстрый выпуск прототипов устройств. В 2007–2008 году Google и HTC интенсивно работали над первым смартфоном на базе Android — HTC Dream. 22 октября 2008 года устройство уже поступило в продажу.

TODO: Добавить графику с историей развития мобильных платформ.

В 2009 году Microsoft тоже попыталась выйти на рынок смартфонов и выпустила Windows Phone 7. Проект оказался неудачным, своего рода “мобильная Vista”. Продажи устройств на Windows Mobile и Symbian упали и остались только две развивающиеся ОС — Android и iOS. Apple не позволяла сторонним компаниям использовать iOS, поэтому все взгляды обратились на Android.

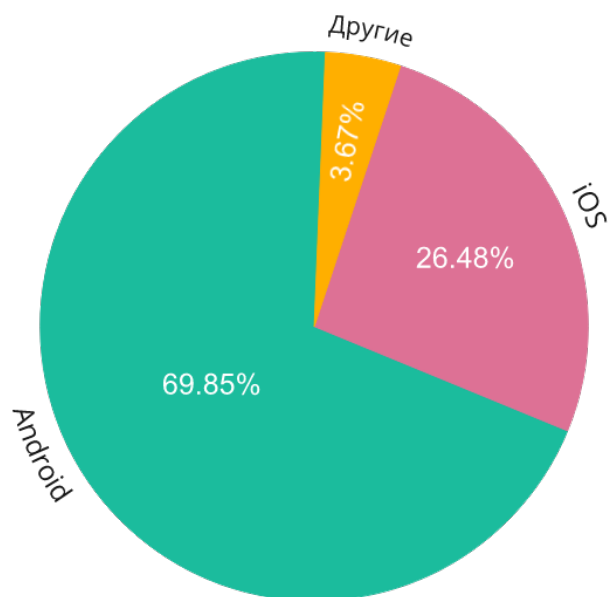


Рисунок 1.5 — Доля устройств (данные по России)

Началась гонка Apple и Google. Компании улучшали свои ОС чтобы превзойти друг друга по быстродействию, безопасности, интерфейсу, функциональности, удобству использования и т.д. В данный момент обе ОС достигли высокого уровня по всем направлениям, но в силу того, что Android — открытая система и лицензии (Apache v2 и GNU GPL v2) не запрещают устанавливать адаптировать её под любые устройства, она гораздо популярнее чем iOS.

Исходя из доли устройств с каждой мобильной ОС в России (см. рис. 1.5), будем выбирать из двух вариантов: iOS и Android. Разработка для iOS ведется на языке Swift или Objective-C, для работы среды разработки требуется устройство под управлением MacOS [6]. Для разработки под Android можно использовать Java или Kotlin, среда разработки не накладывает ограничений на ОС разработчика, так как доступна для Windows, Linux и MacOS [7].

Для сравнения была составлена таблица 1.2. Исходя из покрытия устройств, открытости исходного кода, ограничений на ОС для разработки и наличия опыта программирования на языках Java и Kotlin, была выбрана платформа Android.

Таблица 1.2

Сравнение мобильных платформ

| Название | Поддерживаемые языки программирования | Доля устройств (Россия) [8] | Исходный код | ОС для разработки |
|-------------|---------------------------------------|-----------------------------|--------------|---------------------------------|
| iOS [6] | Swift, Objective-C | 26.48% | Закрытый | MacOS |
| Android [7] | Kotlin, Java | 69.85% | Открытый | GNU/Linux, Windows, MacOS |

1.4 Исследование структуры ОРИОКС

1.5 Входные и выходные данные

1.6 Постановка задачи

2 Конструкторский раздел

2.1 Выбор языка программирования

2.2 Выбор среды разработки

2.3 Выбор стека технологий

2.4 Архитектура и алгоритм работы МП СУПС

2.5 Разработка пользовательского интерфейса

3 Технологический раздел

3.1 Разработка под Android

3.1.1 Система контроля версий

3.1.2 Структура проекта

3.1.3 Разработка пользовательского интерфейса

3.1.4 Работа с данными

3.1.5 Безопасность

3.1.6 Многопоточность и работа с сетью

3.2 Сборка приложения

3.2.1 Система автоматической сборки Gradle

3.2.2 Подключение зависимостей

3.2.3 Подпись приложения

3.2.4 Публикация приложения

3.2.5 Непрерывная интеграция

3.3 Тестирование и отладка в Android разработке

3.3.1 Методы тестирования программного обеспечения

3.3.2 Методология разработки TDD

3.3.3 Тестирование пользовательского интерфейса

3.3.4 Отладка в IntelliJ IDEA

ЗАКЛЮЧЕНИЕ

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Методические указания по подготовке выпускной квалификационной работы по направлению подготовки бакалавров 09.03.04 «Программная инженерия» / Гагарина Л.Г., Коваленко Д.Г., Федотова Е.Л и др. ; Под ред. Б.В. Черникова. — МИЭТ, 2016. — 20 с.
2. ОРИОКС. ОРИОКС. — 2018. — Режим доступа: <https://orioks.miet.ru/> (дата обращения: 20.04.2018).
3. Google. Приложения в Google Play — Расписание для МИЭТ. — 2018. — Режим доступа: <https://play.google.com/store/apps/details?id=com.alex.miet.mobile> (дата обращения: 25.04.2018).
4. Google. Приложения в Google Play — ОРИОКС Live. — 2018. — Режим доступа: <https://play.google.com/store/apps/details?id=com.eva.oriokslive> (дата обращения: 25.04.2018).
5. GitHub. dzmuh97/OpenOrioks — Уведомления из ОРИОКС и ОРОКС для студентов. — 2018. — Режим доступа: <https://github.com/dzmuh97/OpenOrioks> (дата обращения: 26.04.2018).
6. Apple. Develop - Apple Developer. — 2018. — Режим доступа: <https://developer.apple.com/develop/> (дата обращения: 26.04.2018).
7. Google. Application Fundamentals | Android Developers. — 2018. — Режим доступа: <https://developer.android.com/guide/components/fundamentals> (дата обращения: 26.04.2018).
8. Statista. Russia mobile OS market share 2012-2017 | Statistic. — 2018. — Режим доступа: <https://www.statista.com/statistics/262174/market-share-held-by-mobile-operating-systems-in-russia/> (дата обращения: 27.04.2018).

ПРИЛОЖЕНИЕ А

ТЕКСТ ПРОГРАММЫ

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

ПРИЛОЖЕНИЕ Б

РУКОВОДСТВО ПРОГРАММИСТА

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.