

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Российский государственный профессионально-педагогический университет»

ПРИЛОЖЕНИЕ ДЛЯ ВУЗА «МОБИЛЬНЫЙ РГППУ»
Выпускная квалификационная работа
по направлению подготовки 44.03.04 Профессиональное обучение
(по отраслям)
профилю подготовки «Информатика и вычислительная техника»
специализации «Компьютерные технологии»

Идентификационный номер ВКР: 143

Екатеринбург 2017

Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Российский государственный профессионально-педагогический университет»
Институт инженерно-педагогического образования
Кафедра информационных систем и технологий

К ЗАЩИТЕ ДОПУСКАЮ

Заведующая кафедрой ИС

_____ Н. С. Толстова

«_____» _____ 2017 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
ПРИЛОЖЕНИЕ ДЛЯ ВУЗА «МОБИЛЬНЫЙ РГПУ»

Исполнитель:

обучающийся группы № КТ-412

А. С. Гришин

Руководитель:

ст. преподаватель каф. ИС

С. В. Ченушкина

Нормоконтролер:

Б. А. Редькина

Екатеринбург 2017

АННОТАЦИЯ

Выпускная квалификационная работа состоит из приложения для вуза «Мобильный РГППУ» и пояснительной записки на 70 страницах, содержащей 50 рисунков, 23 источников литературы, а также одно приложение на одной странице.

Ключевые слова: ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, МОБИЛЬНОЕ ПРИЛОЖЕНИЕ, РАЗРАБОТКА, ВУЗ.

Гришин А.С. Приложение для вуза «Мобильный РГППУ»: выпускная квалификационная работа / А.С. Гришин; Рос. гос. проф.-пед. ун-т, Ин-т инж.-пед. образования, Каф. информ. систем и технологий. — Екатеринбург, 2017. — 70 с.

Работа направлена на продвижение и представление деятельности вуза в мобильной среде.

Целью работы является создание мобильного приложения на базе мобильной операционной системы Android. Для достижения цели была проанализирована предметная область и аналогичные проекты в сети интернет, подготовлены эскизы и проект продукта, и реализовано мобильное приложение, наполненное данными с официального сайта РГППУ.

СОДЕРЖАНИЕ

Введение.....	5
1 Теоретические основы разработки мобильных приложения	7
1.1 Понятие мобильного приложения.....	7
1.2 Классификация мобильных приложений	9
1.3 Обзор мобильных операционных систем	12
1.4 Основные этапы разработки мобильного приложения.....	17
1.5 Анализ источников, необходимых для разработки мобильного приложения и аналогичных приложений	19
1.5.1 Анализ литературы	19
1.5.2 Анализ интернет-источников	20
1.5.3 Анализ аналогичных мобильных приложений	21
1.6 Выводы по первой главе.....	27
2 Описание разработанного мобильного приложения	28
2.1 Описание заказчика	28
2.2 Описание исходных данных	30
2.3 Преимущества мобильного приложения перед мобильной версткой сайта.....	35
2.4 Выбранная среда разработки приложения	38
2.5 Описание проекта приложения.....	41
2.6 Схема расположения элементов и структура меню	45
2.7 Описание навигационных пунктов и их программная реализация	47
2.7.1 Новости и объявления и выдвигающееся вертикальное меню	47
2.7.2 Пункты «Абитуриенту» и «Студенту»	53

2.7.3 Вывод расписания и сохранение настроек.....	53
2.7.4 Пункт «Мои документы»	57
2.7.5 Пункт «Ссылки».....	57
2.7.6 Пункт «Контакты»	59
2.8 Компиляция приложение и тестовый запуск	59
Заключение	62
Список использованных источников	63

ВВЕДЕНИЕ

На сегодняшний день все больше людей предпочитают мобильный и быстрый доступ к информации. С каждым годом процент пользователей мобильных версий сайтов растет, а, следовательно, компьютерные версии становятся все менее популярны.

Согласно информации сайта Wi-life.ru, количество различных мобильных устройств в 2015 г достигло уровня 7.7 миллиардов, что превысило всю человеческую популяцию в 7.1 миллиардов [5]. Статистика сайта Cossa.ru говорит о том, что мобильным телефоном в России владеет 98 процентов населения [4]. Следовательно, информацию стоит адаптировать именно под смартфоны, чем под персональные компьютеры.

По данным взятым с сайта rg-su.ru [21], на официальный сайт Российского государственного профессионально-педагогического университета ежедневно заходят в среднем 1200 человек. Около 40% из них используют при этом мобильные устройства. Такая ситуация происходит по причине того, что большинство пользователей сайта нашего вуза являются студентами и находятся в возрастном диапазоне 16-28 лет, а этом возрасте люди мобильны и чаще пользуются сайтом через смартфоны и планшеты, чем через стационарные машины.

На данный момент пользование сайтом с мобильного устройства создает ряд неудобств. Нужно запоминать адрес сайта, либо искать его в поисковике. На самом сайте среди мелкого текста искать нужный раздел, среди огромного количества ссылок, разной степени полезности.

Можно сделать вывод, что использование мобильного приложения, посвященного РГППУ, не только упростит восприятие информации на мобильном устройстве, но и позволит пользователю оперативно взаимодействовать с множеством интерактивных элементов сайта.

В связи с этим было принято решение разработать мобильное приложение, полностью включающее в себя информацию с сайта, а также предлагающее дополнительный интерактив.

Объектом выпускной квалификационной работы является процесс продвижения и представление деятельности ВУЗа в мобильной среде.

Предметом выпускной квалификационной работы является мобильное приложение для Российского государственного профессионально-педагогического университета.

Целью выпускной квалификационной работы является создание мобильного приложения РГППУ на базе мобильной операционной систем Android.

В соответствии с поставленной целью в работе определены следующие **задачи**:

1. Проанализировать предметную область и аналогичные проекты в сети Интернет.
2. Проанализировать исходные данные с сайта ВУЗа и подготовить эскизы и проект продукта.
3. Реализовать мобильное приложение под управлением выбранной операционной системы.
4. Наполнить данными, взятыми с официального сайта РГППУ, разместить на сайте и в официальном магазине приложений «Google Play» и обеспечить возможность скачивания приложения.

1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЯ

1.1 Понятие мобильного приложения

Мобильное приложение — это программное обеспечение, специально разработанное под конкретную мобильную платформу (iOS, Android, Windows Phone и т. д.). Предназначено для использования на смартфонах, фаблетах, планшетах, умных часах и других мобильных устройствах.

Мобильные приложения пишутся на языках программирования высокого уровня, а затем компилируются в машинный код операционной системы для получения максимальной производительности. Разработка приложений имеет свои особенности: мобильные устройства работают от батареи и комплектуются менее производительными процессорами, чем персональные компьютеры. Кроме того, современные смартфоны и планшеты повсеместно оснащены дополнительными устройствами, такими как гироскопы, акселерометры и фотокамеры, дающими уникальные возможности для расширения функционала приложения.

Обычно мобильные устройства продаются уже с некоторыми предустановленными приложениями. Остальные по желанию пользователя можно скачать (как платно, так и бесплатно) на специализированных сервисах: Apple AppStore, Google Play, Windows Phone Store и других. Первые магазины приложений, такие как Apple AppStore и Android Market, ставший впоследствии Google Play, появились в 2008 году. Спустя два года Американское диалектическое общество назвало термин «приложение» словом года.

По итогам 2015 года в каталогах двух самых крупных маркетов было порядка 3 млн приложений. Количество загрузок за один год превысило 300 миллионов [5].

Сегодня у каждой организации есть выбор в форме размещения информации. Очевидно, что ПК-версия сайта, это первое, что организация пытается разработать, однако на этом не стоит останавливаться. Дополнительно разработав мобильное приложение, содержащее весь контент ПК-версии, можно получить множество преимуществ и плюсов.

Одним из них является то, что приложение достаточно лишь один раз загрузить на мобильное устройство, и можно пользоваться им в любое время без подключения к интернету и повторного скачивания. Это особенно удобно в тех случаях, когда деньги на балансе телефона закончились, а приложением нужно срочно воспользоваться.

Использование приложений экономит время и избавляет от нудных ожиданий загрузки необходимых элементов (графика, изображение и ряд других), так как программа уже находится на телефоне в своём полном комплекте.

Создатели приложений могут разработать приложения для мобильных, которые будут подходить отдельно для каждого типа мобильного телефона или планшета. Так, для разных устройств могут быть предусмотрены различные функции, но они всегда остаются полезными и актуальными для их владельцев.

Пользователи приложения, как правило, устанавливают связь с нужным сайтом, оставаясь при этом в курсе всех происходящих событий и новостей определённой организации. Сегодня это имеет особую важность для тех, кто в своей деятельности большое внимание уделяет использованию интернета.

Совершенствование приложения и укрепление авторитета и репутации компании взаимосвязаны.

1.2 Классификация мобильных приложений

Классификация мобильных приложений достаточно сложная, так как многие из них являются «гибридами», то есть предлагают пользователю функции различного рода. Но все же некоторые виды мобильных предложений можно классифицировать однозначно.

Контентные приложения очень популярны среди пользователей мобильных приложений. На сегодняшний день такие виды активности, как прослушивание музыки, просмотр различных фильмов, клипов и фотографий, а также чтение цифровых книг являются максимально доступными и удобными для любого владельца мобильного гаджета, что и рождает спрос на данный сегмент мобильных приложений (рисунок 1).

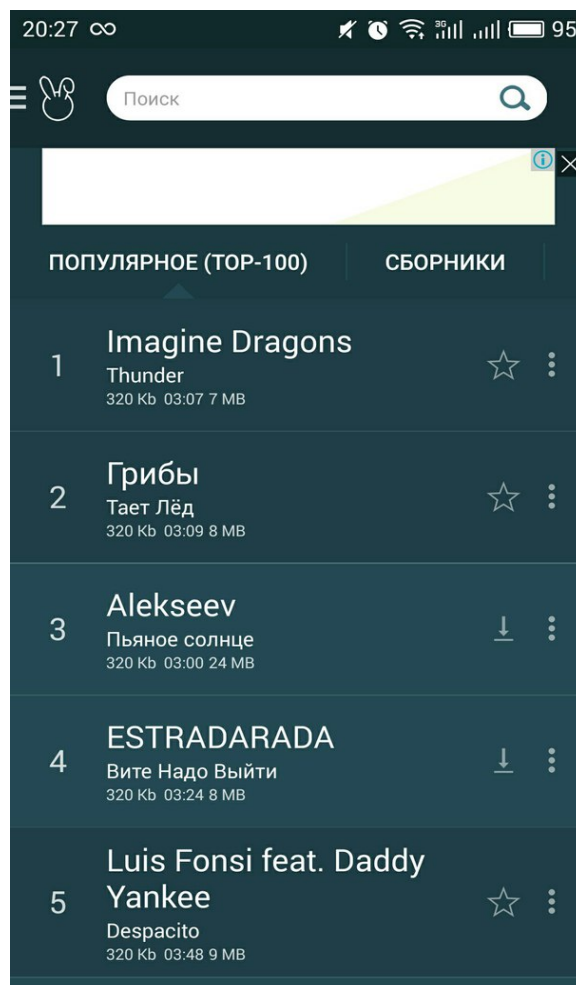


Рисунок 1 — Пример контентного приложения

Бизнес-приложения (рисунок 2) стали необходимым средством для многих пользователей, которое поможет им упростить их офисную работу. В настоящий момент сегмент бизнес-приложений является предпочтительным для инвесторов, но сложность для данного сегмента составляет перевод бизнес-задач на мобильные телефоны.

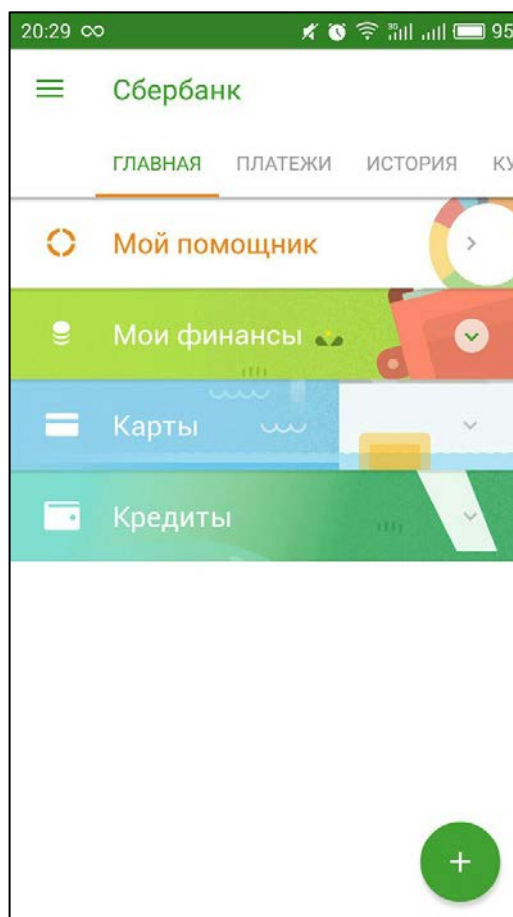


Рисунок 2 — Пример бизнес-приложения

Мобильные игры (рисунок 3) наиболее востребованы на рынке мобильных приложений на сегодняшний день. Разработчики придумывают новые игры или совершенствуют уже выпущенные. Игры притягивают внимание все большей аудитории. Они становятся неотъемлемой частью жизни многих пользователей.



Рисунок 3 — Мобильная игра

Мобильные социальные сети с каждым днем набирают все большую популярность, наращивая многочисленную аудиторию по всему миру. Социальными сетями на сегодняшний день пользуется все большее количество людей, на что оказывает влияние другая уже сложившаяся тенденция: увеличение количества пользователей смартфонов [12]. Можно сказать, что это самый популярный вид мобильных приложений, так как при покупке смартфона на нем уже по умолчанию будет установлена одно или несколько аналогичных мобильных программ.

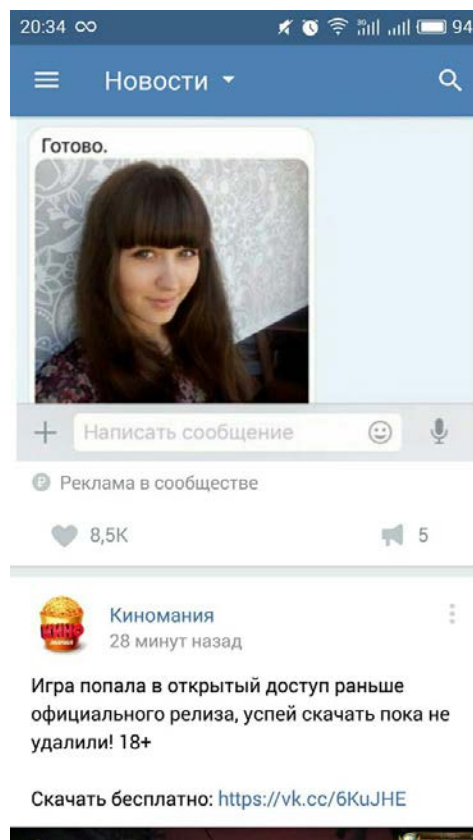


Рисунок 4 — Социальная сеть

1.3 Обзор мобильных операционных систем

В настоящее время существует хороший выбор языков программирования для разработки мобильных приложений. Это связано с тем, что для различных мобильных устройств, приходится использовать различные языки программирования. Обычно это связано с тем, что мобильные устройства имеют различные операционные системы (ОС).

Мною были рассмотрены такие технологии как Windows Phone SDK (написание на языке XAML), iPhone SDK (основной язык – ObjectiveC), Android SDK (основной язык – Java).

Windows Phone SDK

На данный момент, последняя версия инструментария доступна в версии Windows Phone SDK 7.1 ReleaseCandidate в лицензии «GoLive» с возможностью разрабатывать свои приложения и публиковать их в Windows

Phone Marketplace. WindowsPhone SDK 7.1 ReleaseCandidate содержит следующие компоненты:

- Windows Phone SDK 7.1;
- Windows Phone Emulator;
- Windows Phone SDK 7.1 Assemblies;
- Silverlight 4 SDK and DRT;
- Windows Phone SDK 7.1 Extensions for XNA Game Studio 4.0;
- Expression Blend SDK for Windows Phone 7;
- Expression Blend SDK for Windows Phone OS 7.1;
- WCF Data Services Client for Windows Phone;
- Microsoft Advertising SDK for Windows Phone.

Код разрабатываемого приложения описывается на языке XAML. На самом деле — это просто XML файлы с языком разметки XAML.

Платформа Windows Phone (рисунок 5) не просто очередная платформа для мобильных устройств. Она содержит в себе не только технологическую составляющую, но и полностью проработанную концепцию дизайна интерфейса и взаимодействия с пользователем под названием Metro-дизайн или стиль Metro.

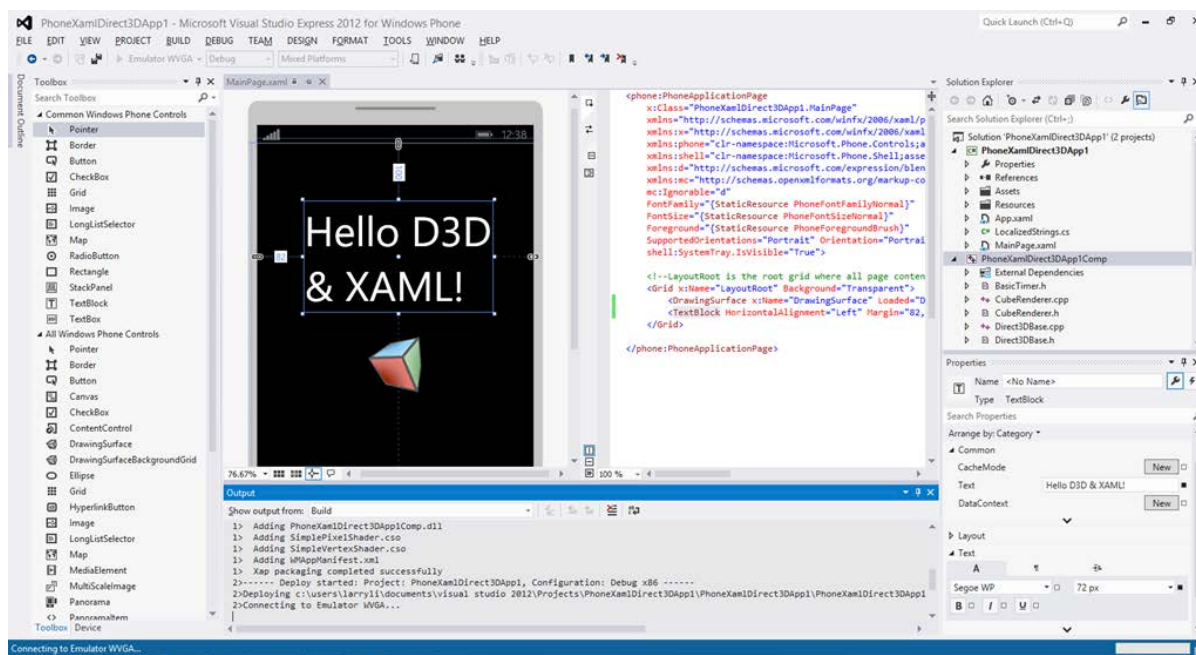


Рисунок 5 — Средство разработки для Windows Phone

Вся разработка под Windows Phone ведется в среде VisualStudio. Среда является очень удобной для разработки и отладки приложений. Для мобильных приложений под Windows Phone отладка происходит с помощью эмулятора Windows Phone с помощью среды разработки Windows Phone.

iPhone SDK

Разработка под iPhone под операционную систему iOS возможна только под Mac OS X. Но в Интернете можно найти статьи, как можно программировать и на Macintosh и даже на VM. Стоит заметить, что Apple предоставляет инструменты бесплатно, платить придется за подписку разработчика.

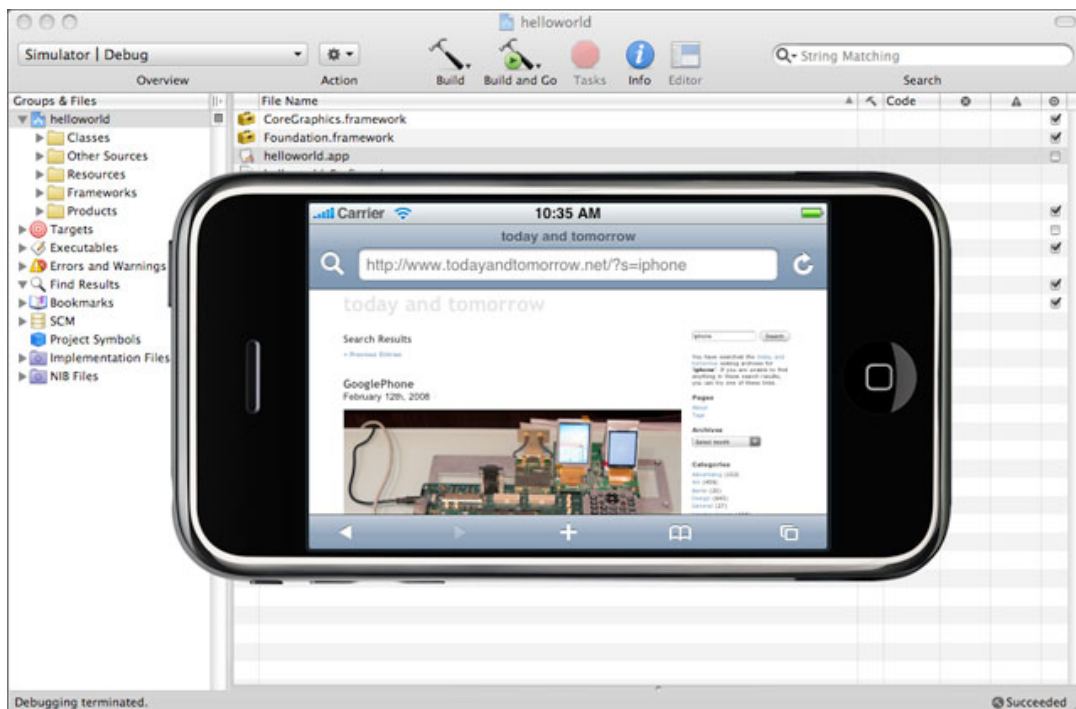


Рисунок 6 — Средство разработки для iPhone

Для написания программ под iPhone предлагается использовать ObjectiveC (рисунок 6). При этом есть возможность писать так же и на C и на C++ (для этого необходимо изменять расширения файлов с .m на .mm). Правда при этом полностью уйти от ObjectiveC не удастся, почти весь API рассчитан именно на него, исключения составляют, например OpenGL (хотя для его инициализации придется использовать несколько строк кода на ObjectiveC), так же полностью доступны стандартные библиотеки C/C++

(так, например, с файловой системой можно работать как средствами SDK на ObjectiveC, так и используя стандартную библиотеку C для ввода/вывода (fopen(), fgetc(), etc())).

Отладка приложения происходит с помощью среды XCode и эмулятора iPhone установленного в ней.

Android SDK

Для разработки под Android можно использовать среду Eclipse которая до недавнего времени был наиболее популярной средой разработки Android, многие опытные разработчики продолжают его использовать или Android Studio (рисунок 7) — более молодая среда разработки, тем не менее быстро набирающая популярность как более интеллектуальная и удобная. Кроме того, Android Studio активно развивается и поддерживается компанией Google как официальная среда разработки андроид-приложений.

Основой среды Android Studio для разработки Android-приложений стал проект российской компании IntelliJ IDEA.

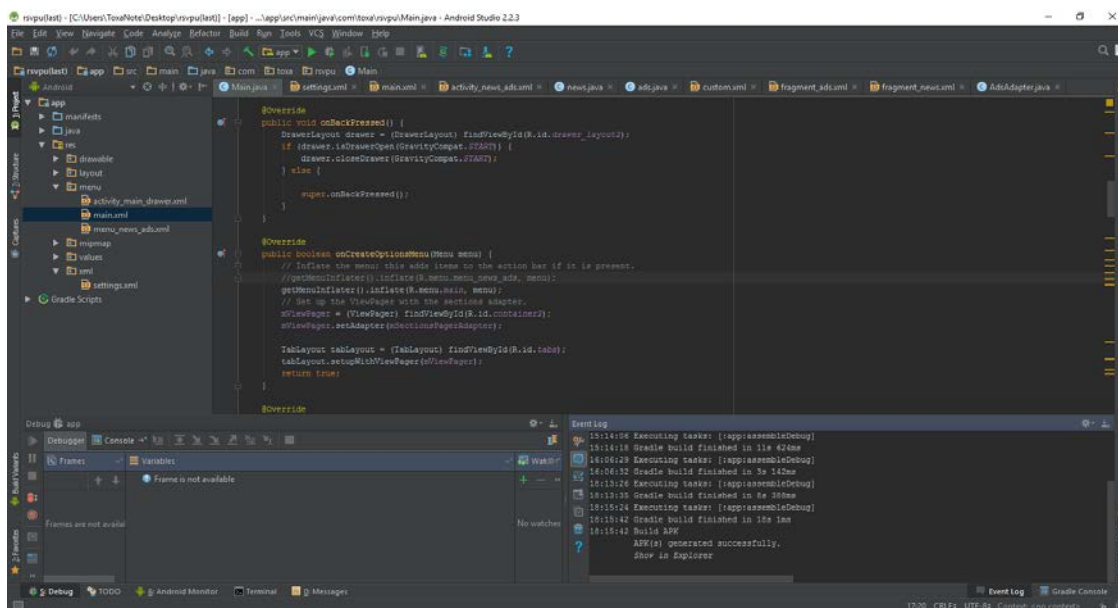


Рисунок 7 — Средство разработки для Android

Компания Google в рамках конференции Google I/O представила собственную интегрированную среду разработки Java-приложений для Android под названием Android Studio. Эта среда базируется на открытой платформе IntelliJ IDEA, которая была разработана питерской компанией JetBrains.

До настоящего времени у Google не было собственной интегрированной среды разработки (Integrated Development Environment – IDE), с помощью которой разработчики бы могли писать Android-приложения. Поэтому они пользовались другими доступными средами. Наиболее популярными являются: Eclipse, свободная среда, поддерживаемая организацией Eclipse Foundation; свободная среда NetBeans и IntelliJ IDEA.

Существует различные версии SDK, которые используются для написания кода для различных версий Android. В настоящее время большое распространения получили версии 2.2 и 2.3. Поддерживается почти полная обратная совместимость версий.

Кроме разработки на языке Java поддерживается возможность более низкоуровневая разработка с использованием Android NDK (Native Development Kit) на языке C/C++.

Если совсем еще недавно можно было утверждать, что создание приложений лучше начинать с iOS-устройства, то на сегодня ответ уже не столь очевиден. Андроид-устройства заполнили рынок и их число уже превысило общее предложение гаджетов, работающих на iOS. Поэтому желая осуществить заказ на игры на Android или другого софта, вы планируете открыть свой бизнес для огромной аудитории [13].

Андроид-аудитория не только широкая, но и весьма разноликая. Скачать на Android программы могут люди самого разного достатка, самых разных возрастов. Компании-разработчики Андроид – устройств изготавливают свои гаджеты не только в сегменте дорогих моделей, но и предлагают бюджетные их варианты. Большинство пользователей могут скачать приложения со специального хранилища Google Play.

Разработка под Android имеет ряд особенностей. Операционная среда Андроид за время своей жизни пережила множество выпусков. Многие даже сейчас еще используют устройства с версией 2.3, которая была выпущена в сравнительно далеком 2010 году, хотя на сегодняшний момент разработка ведется уже под релиз 5 (Lollipop).

Андроид гаджеты характеризуются широким выбором размеров и разрешений дисплеев. Каждый изготовитель старается удивить своих пользователей, побаловать их новыми возможностями. Поэтому в продаже доступны устройства с размером дисплеев от 4 до 10 дюймов. Даже среди смартфонов есть 6–дюймовые девайсы. А разрешительная способность представлена еще шире. Это другая особенность Android гаджетов.

Android имеет большую базу пользователей и, как следствие, у представленного в приложении продукта будет широкая аудитория.

В качестве среды разработки мною был выбран пакет Android Studio, так как эта официальная среда разработки Googlei современнее своего аналога Eclipse.

1.4 Основные этапы разработки мобильного приложения

При разработке мобильного приложения всегда стоит следовать определенному плану, чтобы не упустить из виду множество важных аспектов.

Анализ целевой аудитории. На этом этапе заказчику стоит определиться, зачем он планирует использовать приложение, какова итоговая цель разработки мобильного инструмента коммуникации с аудиторией.

Выработка согласованного решения. Перед началом разработки необходимо получить от заказчика техническое задание (ТЗ) или предоставить ему бриф для заполнения и дальнейшей работы по этому документу.

После получения заполненного брифа и / или ТЗ можно приступать к прототипированию и составлению пользовательских профилей для оценки возможностей итогового продукта.

На основе видения дизайнера, бизнес-оценки и согласования подробностей ТЗ можно запускать процесс разработки.

Прототипы разрабатываются дизайнером и могут быть как статическими, так и интерактивными. Для этого можно воспользоваться одним или

несколькими инструментами для прототипирования, о которых мы рассказывали ранее.

Статические прототипы и интерактивные макеты должны составляться с учетом технической и программной базы, которую планируется использовать для создания приложения.

Написание кода и внедрение технологий. С готовым дизайном приложение переходит к разработчикам: им предстоит на основе языков программирования, фреймворков и различных технологий создать мобильное приложение в соответствии с ТЗ, брифом и утвержденным прототипом.

Тестирование является обязательным на различных этапах разработки приложения, как на симуляторах, так и на реальных устройствах. Цель тестирования — убедиться, что взаимодействие приложения с аппаратной и программной платформой смартфонов и планшетов будет именно таким, как предполагалось на этапе прототипирования.

Создание пред релизной версии. В результате серии тестов и доработок приложения должна быть получена рабочая версия приложения. Именно эту версию и предстоит добавить в магазин приложений: App Store, Google Play, магазин приложений Windows Phone (в зависимости от того, для какой платформы ведется разработка) или любой аналогичный сервис для дистрибуции приложений.

Публикация приложения в магазин — финальный этап работы студии. Приложение добавляется в соответствующий платформе магазин из указанных выше.

Необязательный этап: дальнейшая техническая поддержка и маркетинговое продвижение приложения

При желании заказчика возможно предоставление дополнительных услуг, таких как техническая поддержка приложения, дальнейший выпуск новых версий под обновляемые версии мобильных ОС, а также маркетинговое продвижение.

Поскольку эти услуги предоставляются отдельно от основного пакета услуг, то и оплачиваются отдельно. Помимо маркетинга и техподдержки возможно также размещение приложения в App Store или Google Play от имени заказчика, обеспечение серверной поддержки для приложения [7].

1.5 Анализ источников, необходимых для разработки мобильного приложения и аналогичных приложений

1.5.1 Анализ литературы

Р. Майер «Программирование приложений для планшетных компьютеров и смартфонов» [6]. Данная книга предназначена для людей, которые уже имели опыт программирования, но никогда не программировали для системы Android. Для работы с этой книгой читателю необходимо иметь опыт программирования на любом языке, чтобы понимать основы алгоритмизации. Эта книга по подробным шагам описывает технологию разработки приложений. Минусом книги, конечно, является ее узконаправленность, однако это и является плюсом в виду более подробной и точной информации.

На западе одним из самых популярных источников является книга Д. Гриффитса «Head First. Программирование для Android» [1], так как она имеет простой и понятный язык изложения материала.

Эта книга также предназначена для новичков разработки среды Android, но подразумевает читателя знать основы языка Java. Большим преимуществом этой книги перед предыдущей является год ее издания. Книга датируется 2016 годом, а значит содержит более новую, актуальную и достоверную информацию.

Пособие Л. Дерси «Создание приложений для Android за 24 часа» [2] также рассчитано на новичков и не подразумевает знание языка Java. Также плюсом этой книги является большое количество примеров, на основе кото-

рых можно лучше изучить тему. Также это пособие датировано 2015г. а значит его еще можно использовать к современным версиям Android.

Л. Дэрсси, Ш. Кондер «Разработка приложений для Android-устройств. Том 1. Базовые принципы» [3].

Данное пособие очень обширно и основательно, имеет много полезной информации, а том числе, и для новичка в разработке. У этой книги есть два главных минуса: первый состоит в том, что переведен только первый том книги, и читатель остается на полпути его обучения, второй и главный минус состоит в том, что книга датирована 2014 год, а за период в два года появились новые и совершенно другие версии Android.

1.5.2 Анализ интернет-источников

Интернет предоставляет большое количество информации по данной теме. Плюсом этой информации является то, что она зачастую свежая и актуальная.

Было рассмотрено несколько наиболее популярных ресурсов.

Startandroid.ru [18] — это непосредственно сайт операционной системы Android. Здесь есть очень много практических советов, как для начинающего, так и для уже опытного пользователя. Плюсом этого сайта является и удобная навигация, при которой всегда можно найти нужный материал.

Android-4pda.ru [15] — сайт, являющийся форумом, посвященным программированию на разных платформах, но для программирования в среде Android здесь есть целый раздел. Плюсом этого сайта является то, что каждый пользователь может задать вопрос по интересующей теме и в течении нескольких минут получить ответы от других пользователей, с возможными примерами.

Кроме специфических сайтов, посвященным полностью данной теме, есть множество других порталов или интернет-журналов, где в новостях пе-

риодически встречается тема разработки приложений для системы Android, например:

Intuit.ru [17] — это открытый онлайн-университет, где можно бесплатно пройти курс по разработке приложений, а после чего решить тесты и сдать экзамен по данной теме.

Habrahabr.ru. Раздел «Разработка под Android» [16] — это один из самых популярных IT порталов в России, в нем есть целый раздел, посвященный программированию, в котором содержится много информации и разработке для среды Android.

Хочется сказать, что по данной теме можно найти очень много литературы: как учебников, так и тематических форумов. Также можно найти некоторые источники, которые структурированно подходят к изучению данной темы. Единственный минус, который может возникнуть, при поиске материала для обучения — это быстрое устаревание материала, поэтому внимание уделялось только источникам не старше 1-3 лет.

1.5.3 Анализ аналогичных мобильных приложений

В магазине мобильных приложений «Google Play» можно найти несколько аналогичных приложений. Почти все они разработаны студентами или сотрудниками непосредственно самих вузов и зачастую не профессиональными разработчиками, следовательно, имеют некую схожесть с мобильным приложением РГППУ. Рассмотрим самые популярные и часто скачиваемые из них.

Приложение Московского государственного университета [19] является лидером по скачиванию. Это объясняется популярностью самого вуза. Можно сказать, что приложение имеет очень удобный и приятный интерфейс (рисунок 8), поэтому при разработке собственного мобильного приложения было обращено внимание именно на этот аналог. Дизайнерская работа проведена намного лучше, чем в предыдущем приложении. Стилизована шапка

приложения, неброский, но красивый фоновый рисунок с ночным видом университета удачно сочетается с темной стилистикой остальных элементов. К сожалению, данные с сервера не получаются. Ни одна ссылка не работает, что делает приложение абсолютно не работоспособным.

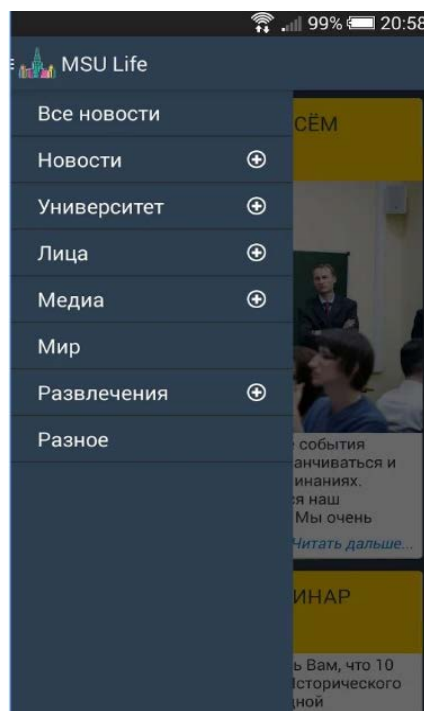


Рисунок 8 — Интерфейс мобильного приложения «MSU Life»

Приложение, посвященное Московскому технологическому институту [11] отличается от интерфейса веб-страницы данного вуза. Это делает интерфейс приложения удобным для просмотра именно на смартфонах и планшетах. На рисунке 9 представлено меню приложения. Оно выглядит очень минималистичным, с применением всего лишь нескольких управляющих элементов и использованием крупных кнопок.

Плюсом данного приложения является наличие множества материалов для обучения. В каком-то смысле оно выступает больше, как электронная библиотека, чем официальное приложение вуза, однако здесь также присутствуют новости вуза и контактная информация.

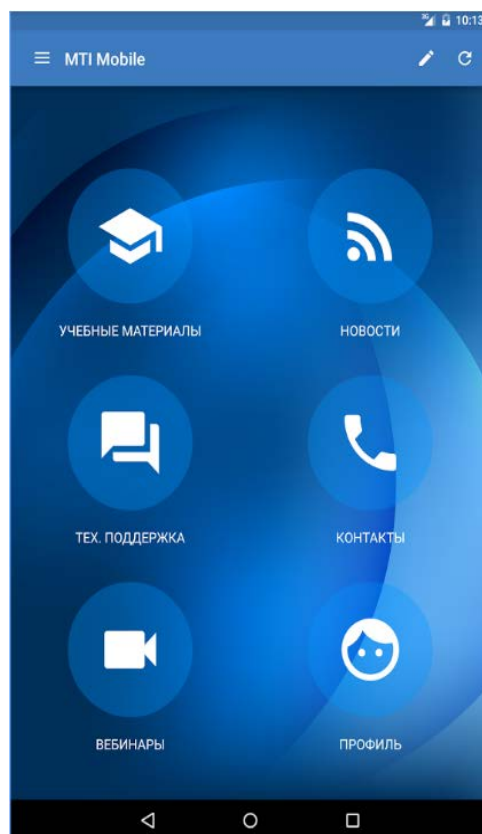


Рисунок 9 — Интерфейс мобильного приложения «МТИ»

Также было обращено внимание на приложение НГУЭУ [9]. Данное приложение было взято во внимание, потому что оно содержит самую удобную интерактивную систему по взаимодействию со студентами. В приложении удобно реализованы такие функции, как расписание (рисунок 10), информация о преподавателях, адреса корпусов университета, контактах.

Плюсом является то, что кроме просмотра информации каждый раздел приложения можно комментировать, обсуждать и ставить оценки. Также пользователь имеет возможность менять цвета некоторых панелей, что улучшает восприятие интерфейса, и позволяет пользователю настраивать приложение под свой вкус.

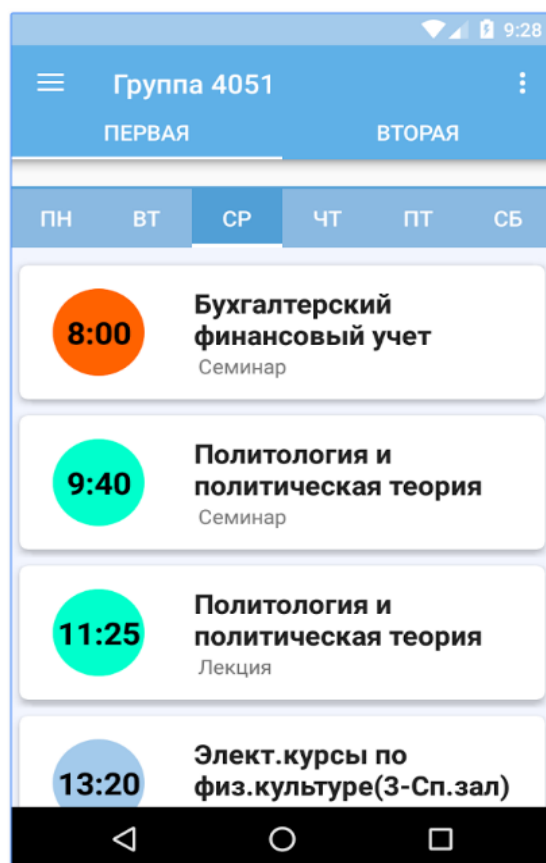


Рисунок 10 — Интерфейс мобильного приложения «НГУЭУ»

Приложение «Университет ИТМО» для Санкт-Петербургского национального исследовательского университета информационных технологий, механики и оптики (рисунок 11) на данный момент является самым популярным и технологичным приложением из всех, что мне удалось найти. Большое количество функционала совместно с технически грамотно сделанной структурой обеспечивает высокую популярность приложения среди студентов и преподавателей данного вуза. В приложении доступно расписание занятий, новости и анонсы университета, информация об университете. После авторизации предоставляется доступ к таким разделам как «Успеваемость», «Стипендия», «Зарплата» и «Обратная связь» для связи с разработчиками. Дизайн приложения яркий и мотивирующий. Каждая вкладка тщательно проработана. Заметно, что над приложением работала команда профессионалов, о чем написано во вкладке «Обратная связь».

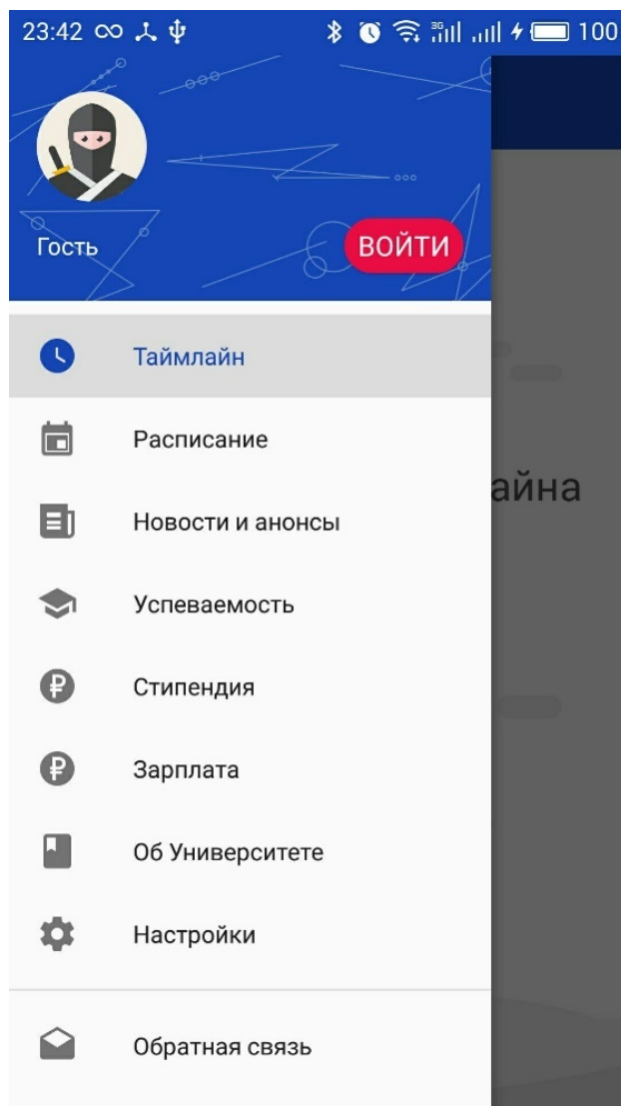


Рисунок 11 — Главное меню приложения «Университет ИТМО»

Следующим рассмотренным приложением будет «УНЦ МГУ» Учебно-научного центра государственного строительства и подготовки кадров МГУ (рисунок 12).

Слишком упрощенный дизайн приложения облегчает его восприятие, но выглядит недоделанным.

Также низкочастотная анимация открывания подпунктов шторки бросается в глаза, и негативно сказывается на общем впечатлении о нем.

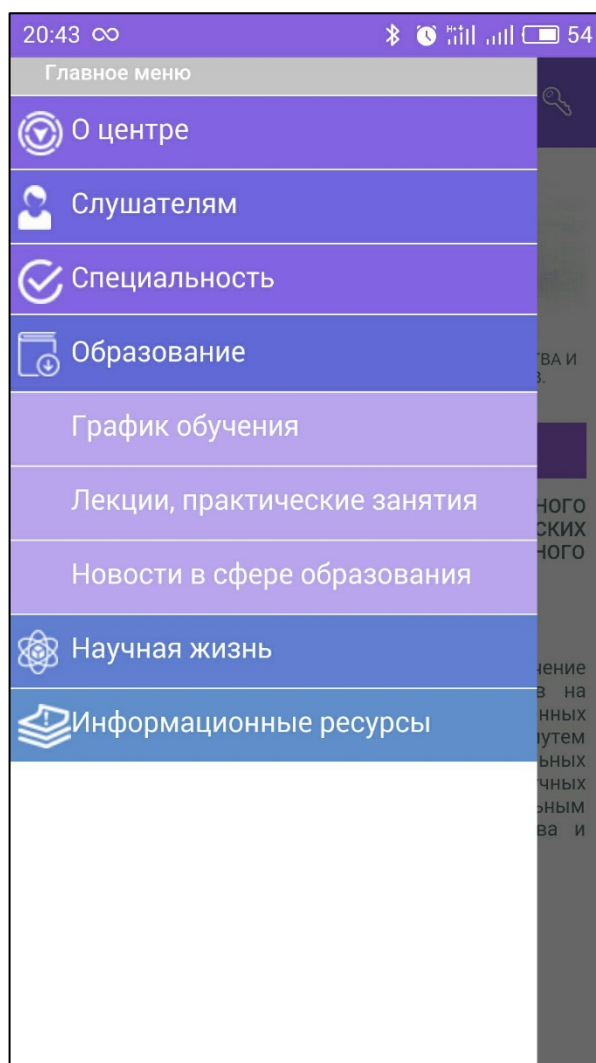


Рисунок 12 — Главная страница приложения «УНЦ МГУ»

Само приложение является скорее большой справкой о вузе, так как содержит много справочной информации, а интерактивность практически отсутствует. Авторизация через личный кабинет не дает абсолютно никаких преимуществ и является бесполезной.

Сегодня существует несколько популярных мобильных приложений, содержащих информацию о вузах. Можно сказать, что каждый из них не имеет никакой четкой классификации. Каждый из них является уникальным с оригинальным дизайном и контентом, что позволяет почерпнуть новые идеи для разработки мобильного приложения РГППУ.

1.6 Выводы по первой главе

Анализ предметной области выявил преимущества мобильного приложения перед вёрсткой.

Проанализировав множество популярных сред разработки было принято использовать среду Android Studio и язык программирования Java. Данные средства являются одними из самых популярных и удобных для разработки.

После анализа источников и аналогичных приложений был сделан вывод, что почти все приложения не имеют четкой классификации и структуры.

Было обнаружено, что мобильное приложение улучшает восприятие информации с мобильного устройства и сокращает время доступа к информации, тем самым выявив актуальность создания мобильного приложения.

2 ОПИСАНИЕ РАЗРАБОТАННОГО МОБИЛЬНОГО ПРИЛОЖЕНИЯ

2.1 Описание заказчика

Российский государственный профессионально-педагогический университет (РГППУ) – это федеральное государственное автономное образовательное учреждение высшего образования, в состав которого входят институты, колледж, филиалы и представительства в различных городах РФ.

Для рекламы университет использует сайт (рисунок 13), группу в социальной сети «ВКонтакте», канал на сайте «Youtube.com».

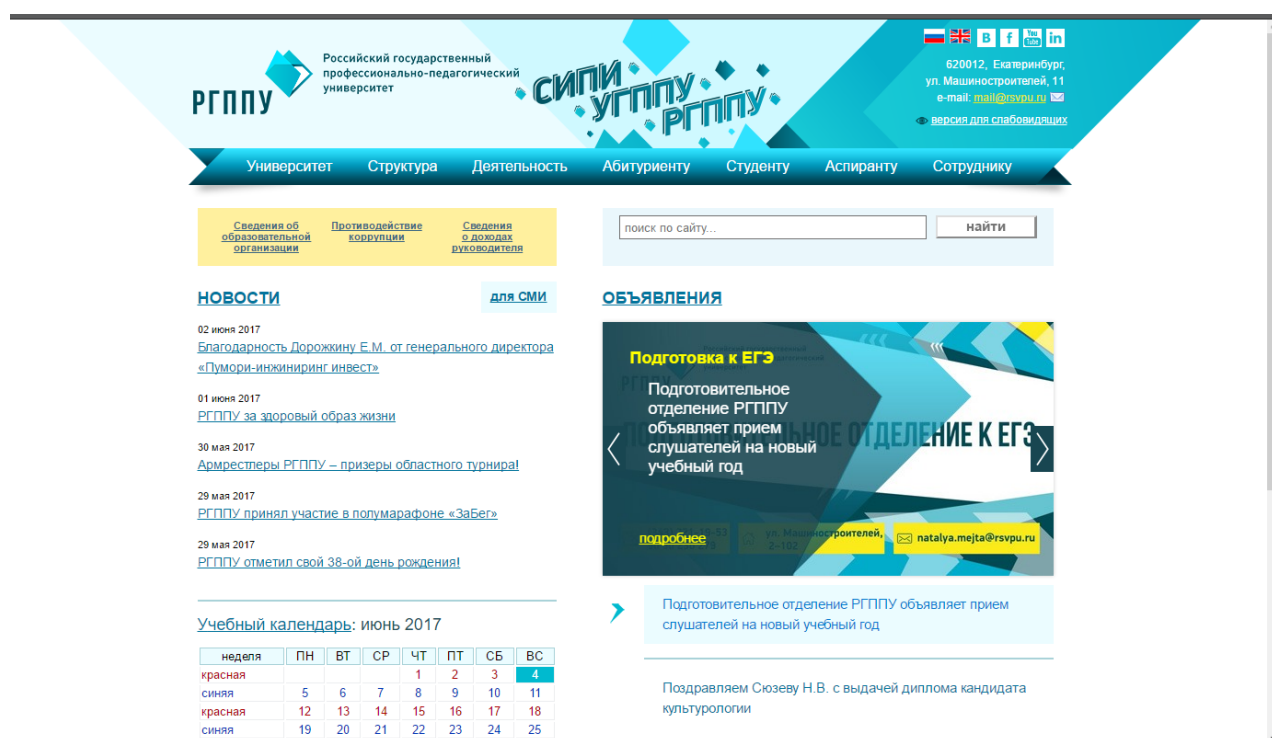


Рисунок 13 — Главная страница официального сайта вуза

Для расширения рынка сбыта, создания имиджа вузу и для рекламных и маркетинговых целей, руководством университета принято решение создать мобильное приложение.

Российским государственным профессионально-педагогическим университетом было представлено техническое задание на разработку мобильного приложения, состоящее из следующих пунктов:

1. Цели и задачи приложения. Целью разработки приложения является повышение степени лояльности студентов и привлечение абитуриентов за счет удобства и простоты доступа к информации университета. Приложение должно содержать информацию о расписании занятий, новостях, контактах и ссылках на официальные электронные ресурсы университета.

2. Язык реализации. Приложение будет реализовано на русском языке. В последующих версиях возможно добавление английского языка для иностранных студентов и абитуриентов.

3. Ориентация экрана. При работе на мобильных телефонах ориентацией экрана является портретной.

4. Стартовый экран. Стартовый экран содержит новостную ленту, загружаемую напрямую из базы университета.

5. Операционная система и устройства, обеспечивающие работу приложения. В приоритете приложение должно работать на устройствах под управлением Android версии 4.0.2 и выше.

6. Язык программирования. При разработке приложения должен использоваться следующий язык программирования: Java. Запросы к базе данных будут написаны на языке PHP с использованием языка запросов Transact-SQL.

7. Регистрация и размещение приложения в Google Play. Публикация приложения Google Play будет осуществляется заказчиком самостоятельно или с помощью Исполнителя при условии наличия оплаченного аккаунта заказчика в Google Play.

8. Оформление. Дизайн приложения должен соответствовать корпоративному стилю университета.

2.2 Описание исходных данных

Для наполнения информации был проанализирован сайт ВУЗа для определения структуры и обозначения данных необходимых для загрузки в мобильное приложение.

В результате анализа было выявлено, что в мобильное приложение будут задействованы в качестве исходных данных следующие разделы:

1. Раздел «Новости» – лента, частично выведенная на главной странице сайта, содержащая информацию о произошедших событиях в университете (рисунок 14).



Рисунок 14 — Новостная лента сайта Российского государственного профессионально-педагогического университета

2. Раздел «Объявления» – аналогичная новостной ленте. Содержит информацию о готовящихся событиях (рисунок 15).

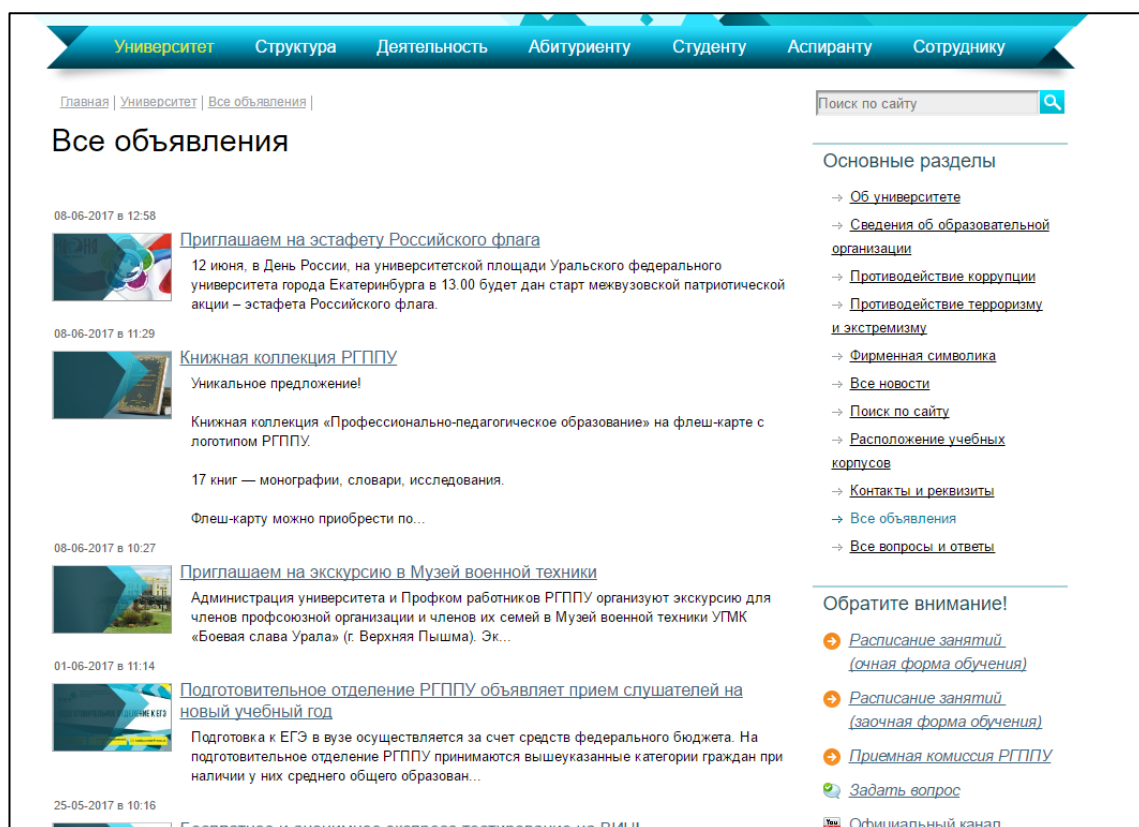


Рисунок 15 — Лента объявлений Российского государственного профессионально-педагогического университета

3. Раздел «Абитуриенту», содержащий информацию для поступающих. Так как раздел очень большой использоваться будет только вкладка «Высшее образование» (рисунок 16).

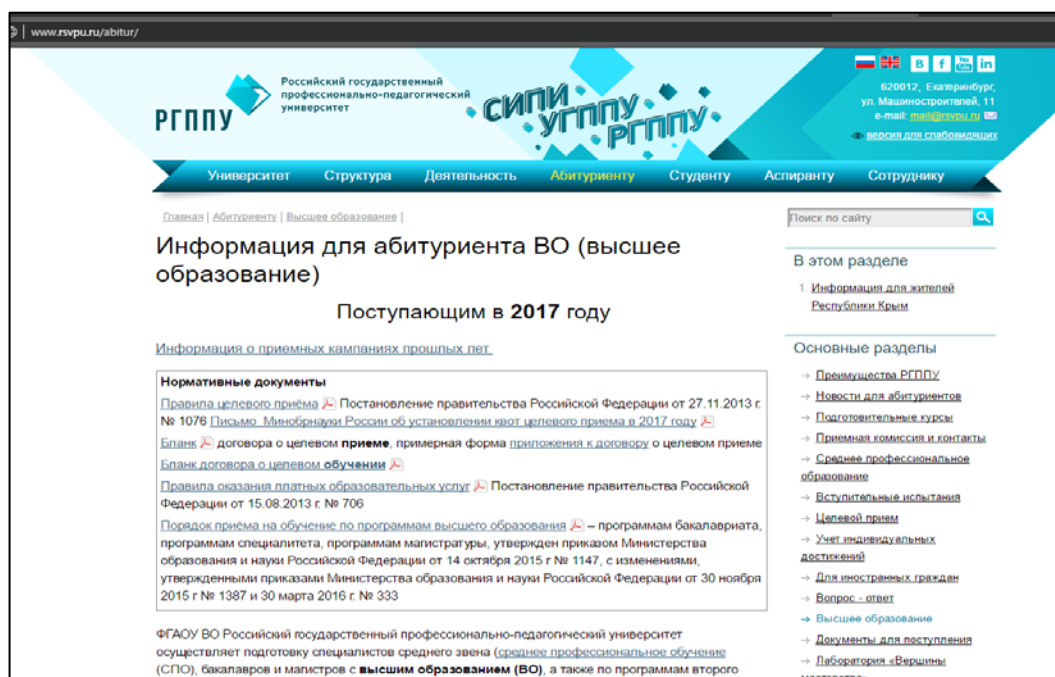


Рисунок 16 — Раздел информации для абитуриента

4. Раздел «Студенту», в котором будет использована только контактная информация институтов и отдельный проект «Мои документы» — личный кабинет студента (рисунок 17), в котором всем желающим, после авторизации через систему «Таймлайн» предоставляется возможность воспользоваться сервисами университета.

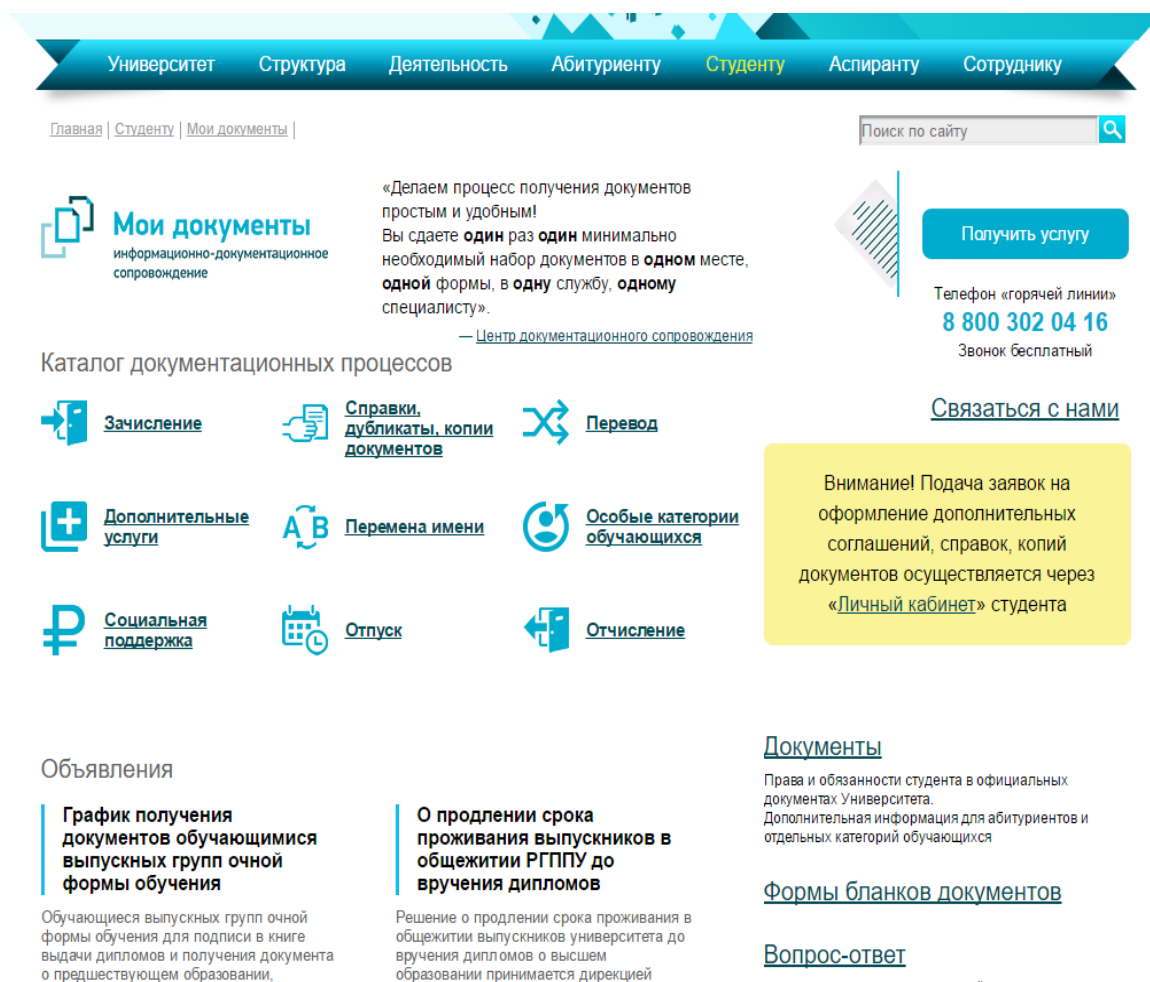


Рисунок 17 — Раздел «Мои документы»

5. Существует две версии раздела «Расписание». Первая версия созданная для настольных компьютеров, неудобна в использовании с мобильного устройства и доступна с любых устройств. Ей пользуется большинство студентов. В приложении будет использована вторая – мобильная версия, созданная Павлом Кунофом. Она становится доступна на главной странице сайта только при заходе на него с мобильного устройства (рисунок 18).

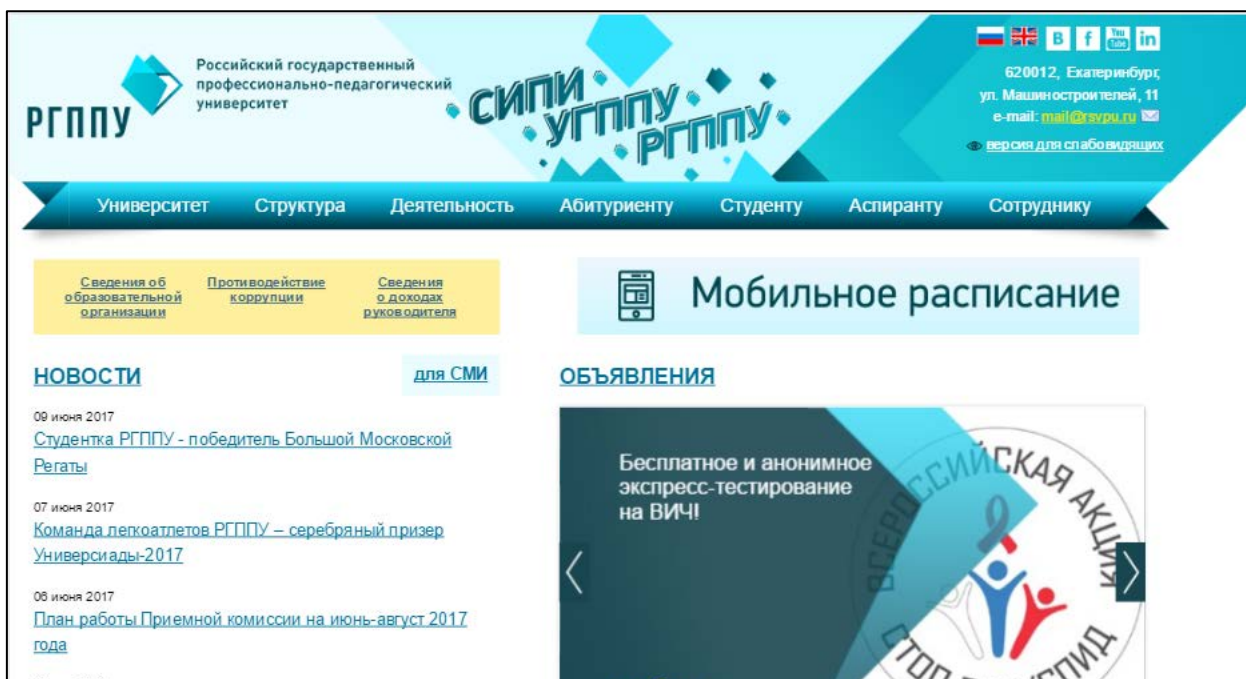


Рисунок 18 — Ссылка на мобильное расписание

На рисунке 19 представлен внешний вид мобильного расписания, открытого через мобильный телефон.

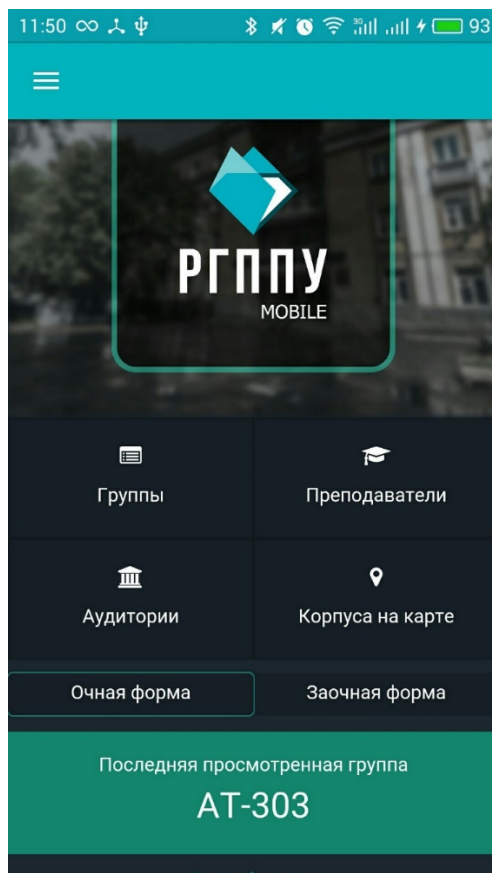


Рисунок 19 — Мобильная версия расписания

6. Официальный канал университета на сайте «youtube.com» [23] на котором сотрудниками размещаются различные видеоролики, связанные с университетской деятельностью (рисунок 20).

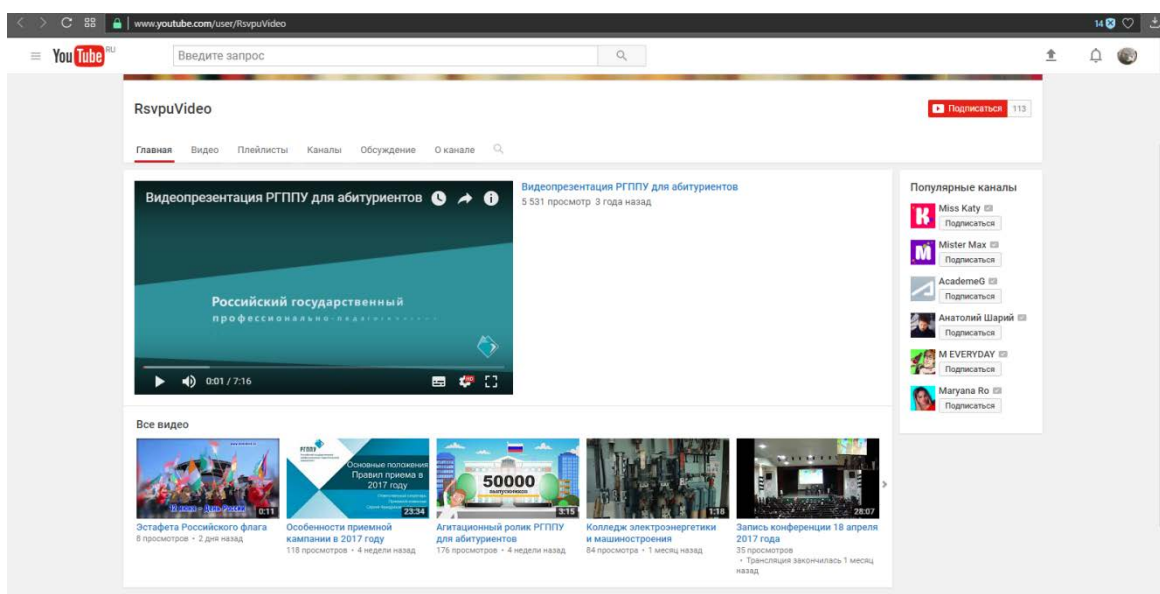


Рисунок 20 — Главная страница официального сайта Российского государственного профессионально-педагогического университета

7. Официальная группа Вконтакте [22] служит для информирования студентов в социальной сети, в которой студенты проводят большое количество времени.



Рисунок 21 — Официальная группа Вконтакте

8. Также был взят за основу логотип сайта, и перерисован в векторном формате, для более высокого качества.

2.3 Преимущества мобильного приложения перед мобильной версткой сайта

Существует два направления технической реализации проекта для мобильных устройств: мобильный web-сайт и мобильное приложение.

Мобильным web-сайтом будем считать специализированный сайт, адаптированный для просмотра и функционирования на мобильном устройстве. Сайт может включать в себя интерактивные компоненты с использованием JavaScript, HTML5, новых API браузеров. В этом случае такую реализацию называют web-приложением. Далее, чтобы избежать путаницы любой вариант, работающий внутри браузера мобильного устройства, будем называть web-сайтом.

Таким образом, можно подготовить мобильную верстку сайта и просматривать посредством мобильного браузера.

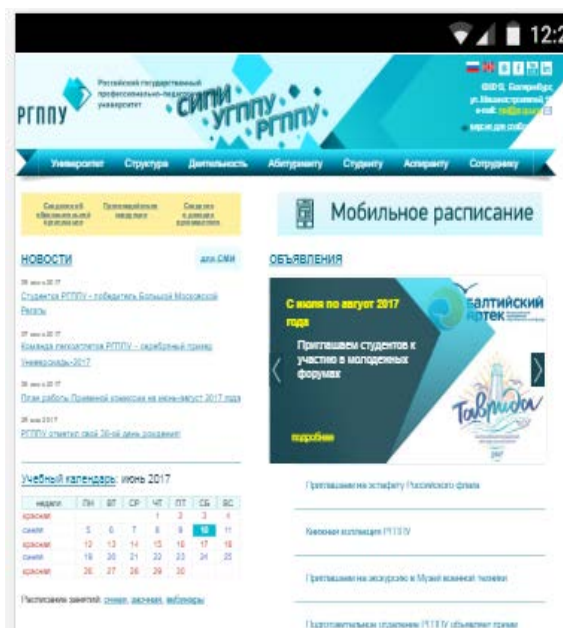


Рисунок 21 — Мобильная версия сайта Российского государственного профессионально-педагогического университета.

Мобильное приложение – это специально разработанное приложение под конкретную мобильную платформу (iOS, Android, Windows Phone). Обычно приложение разрабатывается на языке высокого уровня и компилируется в нативный код ОС, дающий максимальную производительность.

Существует еще третий вариант – мобильное приложение, включающее в себя компонент браузера. В этом случае часть мобильного приложения чаще всего используется для навигации и интеграции с ОС, а web-компонент – для показа контента. Обычные пользователи не могут зачастую отличить такой вариант от нативного мобильного приложения [19].

Одним из первых аргументов, которые приводят сторонники приложений – наиболее близкий к ОС и привычный для пользователей интерфейс. Действительно мобильное приложение наиболее тесно интегрировано с платформой и дает реализовать привычный отзывчивый интерфейс. С другой стороны web-сайт с помощью хорошего форматирования и использования JavaScript может дать вполне понятный метод взаимодействия. На текущий момент отзыв web-сайта значительно уступает приложению, но мощность мобильных устройств продолжает расти и сами браузеры существенно меняются в лучшую сторону. Кроме того, различные версии мобильных ОС могут диктовать свои стандарты, которых приходится придерживаться. При этом некоторые нововведения могут оказаться не совсем понятны обычным пользователям. Существенным в данном случае является то, что наиболее активными пользователями (теми, кто выставляет рейтинг и делает комментарии в магазинах приложений) являются те, кто «фанатеет» от последних новшеств мобильной ОС. На это стоит обратить внимание при продвижении проекта – их можно использовать как союзников, помогающих распространению.

Web-сайт, а особенно интерактивный, существенно уступает приложению с точки зрения быстродействия. Браузеры мобильных устройств пока не могут порадовать высокой производительностью, кроме того, web-разработчики используют не самые оптимизированные версии библиотек

(плохая реализация этих библиотек никак не сказывается на «больших» браузерах, поэтому с этим там можно смириться). Однако и приложение не всегда может радовать хорошим быстродействием – излишняя анимация, сложный интерфейс значительно снижают «отклик». Кроме того, для сложной графики и анимации приходится использовать языки более низкого уровня, разрабатывать или покупать отдельные специализированные библиотеки.

В этой области приложения далеко опережают сайт. В приложении существенно больше возможностей для доступа к устройству. Однако выше упоминался уже третий вариант, когда компонент браузера внедряется в приложение и в этом случае такая разница нивелируется. Кроме того, постоянно растет уровень предоставления доступа к возможностям устройства из браузера через расширяющийся набор API.

Web-сайт запускается из браузера, поэтому требует постоянного соединения с сетью. Это не имеет значения, если проект реализуется исключительно как онлайн-овый. Однако даже в этом случае из-за особенностей мобильного доступа в Интернет переход между частями приложения (навигация) связана с неприятными для пользователя задержками. Возможно, использование API для хранения локальных данных решат эту проблему, но пока примеров такого применения найти не удалось. Мобильные приложения могут осуществлять работу без подключения, выполняя кеширование и обновление данных, если требуется, при появлении соединения. Но все же и приложению нужно подключение в подавляющем большинстве бизнес-решений.

Для реализации проекта на всех или каких-то определенных платформах требуется разработать приложение для каждой из платформ отдельно, причем на каждой свои среда и язык разработки, свои стандарты интерфейса. В случае мобильного сайта одна версия должна покрывать потребности всех платформ. Так выглядит в теории. Но на практике оказывается, что браузеры на различных платформах функционируют по-разному. Приходится поддерживать либо несколько версий одного сайта, либо в коде подстраивать выда-

ваемый контент под текущий запрос. Существенные отличия в размерах экрана также сказываются и на верстке сайта.

2.4 Выбранная среда разработки приложения

Для разработки мобильного приложения выбрана среда разработки Android Studio. Это бесплатный комплекс инструментов, объединенных в одной оболочке, предназначенный для разработки и тестирования приложений, работающих в операционных системах семейства Android. Разработчики данной платформы собрали большой арсенал из необходимых инструментов для эффективного и удобного создания новых приложений и, в случае необходимости, модернизации уже существующих.

Android Studio состоит из следующих компонентов: Android SDK, компоненты для работы с графическим дизайном, приложение для загрузки компонентов различных версий Android, эмулятор мобильного устройства для запуска приложения, инструменты для проведения тестов и отладки приложений.

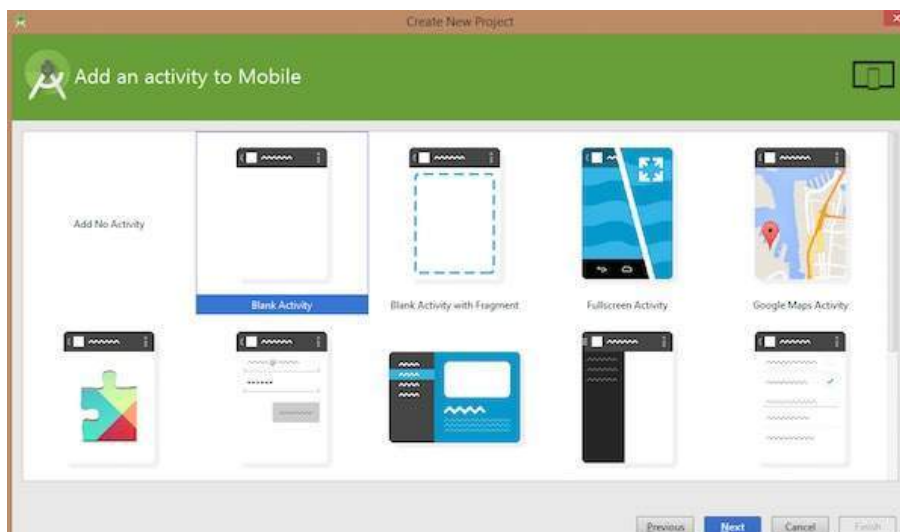


Рисунок 22 — Создание нового проекта в Android Studio

При создании нового проекта (рисунок 22) в программе, полностью доступна вся структура программы вместе со всеми файлами, что несомненно дает возможность максимально эффективно и продуманно организовать сам

процесс разработки. Очень удобно внедрена возможность показывать вносимые изменения и дополнения, выводятся визуально в режиме настоящего времени, происходят преобразования в точной зависимости от заданных алгоритмов.

Важным фактом является то, что Android Studio предоставляет возможность заниматься разработкой программ практически для всех версий ОС Android, начиная с релиза, и заканчивая последней на текущий момент версией 7.2. Существует инструмент для предварительной оценки внешнего вида программы для различных устройств, например, смартфон или планшет. Удобная подсветка кода, позволяет легко ориентироваться в больших объемах кода. Кроме того, большим плюсом является изменение отдельных компонентов методом простого перетаскивания в то место, где пользователь захочет его видеть, данная функция существенно облегчает процесс программирования. Android Studio обладает инструментом Google Cloud Messaging, главной задачей которого является настройка отправок всплывающих уведомлений для приложений, задействуя при этом облачные сервисы под управлением ОС Android. В итоге, можно сделать вывод о многозадачности Android Studio. Одним из преимуществ данной среды является массивное подробное тестирование приложения перед его выпуском. Тестирование проводится во всевозможных режимах работы, что позволяет исправить многочисленные недочеты.

Свойства программы Android Studio:

1. Возможность рендеринга приложений в реальном времени. Такую возможность предоставляет встроенный редактор WYSIWYG.
2. Возможность построения проектов на базе Gradle.
3. Наличие всплывающих подсказок, упрощающих процесс разработки.
4. Возможность реферального отслеживания.
5. Работа в редакторе слоев, в котором есть возможность управлять всеми элементами методом перетаскивания мышью.

6. Наличие мастера шаблонов, который позволяет быстро создавать стандартные дизайны.

7. Наличие инструментов, проверяющих возможности совместимости различных приложений.

Для разработки мобильного приложения недостаточно только Android Studio, необходимо также наличие комплекта разработчика приложений JDK. Оно включает в себя следующие составляющие: среда выполнения компилятор и API.

Для компилирования приложения был использован эмулятор мобильных устройств Genymotion (рисунок 23). Он позволяет запускать приложения на виртуальных образах различных устройств Android (x86 с аппаратным ускорением OpenGL).

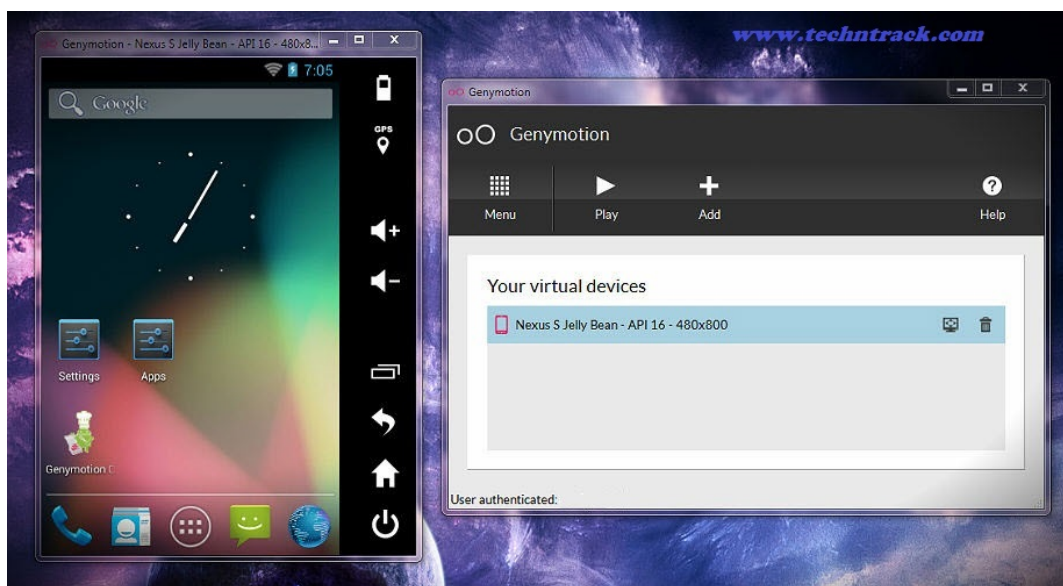


Рисунок 23 — Интерфейс программы Genymotion

По словам разработчиков, цель продукта – заменить эмулятор Android от Google для использования не только разработчиками Android, но и людьми, которые делают презентации мобильных приложений.

Возможности Genymotion:

1. Быстрая загрузка и запуск предварительно настроенных образов Android. Список устройств постоянно расширяется.

2. Функционал GPS-модуля (с настраиваемыми координатами) и батареи (с настраиваемым уровнем батареи).
3. Отображение: аппаратное ускорение с OpenGL, режим полноэкранного просмотра.
4. Консоль, которая позволяет вам взаимодействовать с вашей виртуальной машиной, используя командную строку.
5. Эмуляция передней и задней камеры.
6. Эмуляция работы интернета через GPRS, Edge, 3G, 4G и пр.
7. В будущих релизах ожидается появление новых сенсоров: гироскоп, яркость экрана, температура, вектор поворота и многое другое.
8. Работает на порядок быстрее эмуляторов от Google (не только на процессорах Intel).
9. Общий буфер обмена виртуального устройства и ПК.
10. Редактирование количества процессоров (CPU) и количества оперативной памяти (RAM) в настройках виртуального устройства.
11. Генерирование архива log файлов при обращении в тех поддержку.
12. Исправления ошибок падения эмулятора, добавлена информация в логах.

2.5 Описание проекта приложения

Приложения для Android состоят из компонентов, которые не зависят друг от друга и могут быть запущены, пользователем или сторонним приложением (при наличии соответствующих разрешений), независимо, т.е. одно приложение может использовать компоненты другого. И при этом приложение не будет включать в себя код другого. Такой подход требует того, что бы каждый компонент приложения мог запускаться независимо, по этой причине в Android приложении нет единой точки входа, такой как `main()` в C.

Всего в Android приложениях имеется 4 компонента:

- Activity;

- Fragment;
- Service;
- Adapter.

Activity. Это визуальное представление графического элемента, по сути – окно. То, что пользователь видит, когда запускает приложения. Также может содержать различные диалоги подсказки или вопросы. Интерфейс наполняется различными виджетами (кнопки, формы ввода и т.д.). Все activity реализуются как подкласс базового класса Activity. Приложение может содержать несколько activity, в таком случае при запуске нового activity, предыдущее закрывается и помещается стек.

Fragment. Наследующееся от Activity представление, содержащее урезанный функционал. Главным свойством фрагмента является его модульность. Он должен встраиваться в другое представление. Activity может содержать любое количество фрагментов, переключаемых между собой.

Service. Не имеет визуального представления и выполняется в фоновом режиме. Например, service, который проводит мониторинг телефонной линии и ждет, пока не поступит звонок. Service выполняется до тех пор, пока не закончит свою работу или какое либо приложение не завершит его. Также приложения могут и запускать service.

Adapter. Если говорить в общих чертах, то адаптеры упрощают связывание данных с элементом управления. Он позволяет заполнять массивы строчных значений наборами объектов.

При создании нового проекта в AndroidStudio в проекте появляется стандартная структура проекта с исходным кодом и папками и файлами по умолчанию (рисунок 24). Android Studio предоставляет ряд шаблонов для различных ситуаций, но самыми распространенными являются Basic Activity и Empty Activity. Это самые удобные шаблоны для старта для создания большинства приложений. Empty Activity является просто пустым экраном, а Basic Activity имеет ряд минимальных элементов, для облегчения создания новых проектов.

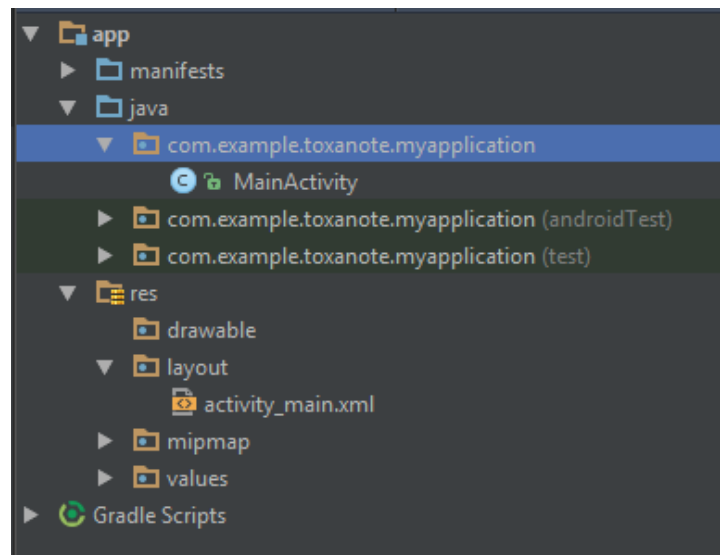


Рисунок 24 — Стандартная структура проекта

В папке manifests находится единственный файл AndroidManifest.xml (рисунок 25) в котором можно изменять название приложения, его значок, стартовую страницу, список Activity и другие технические настройки.

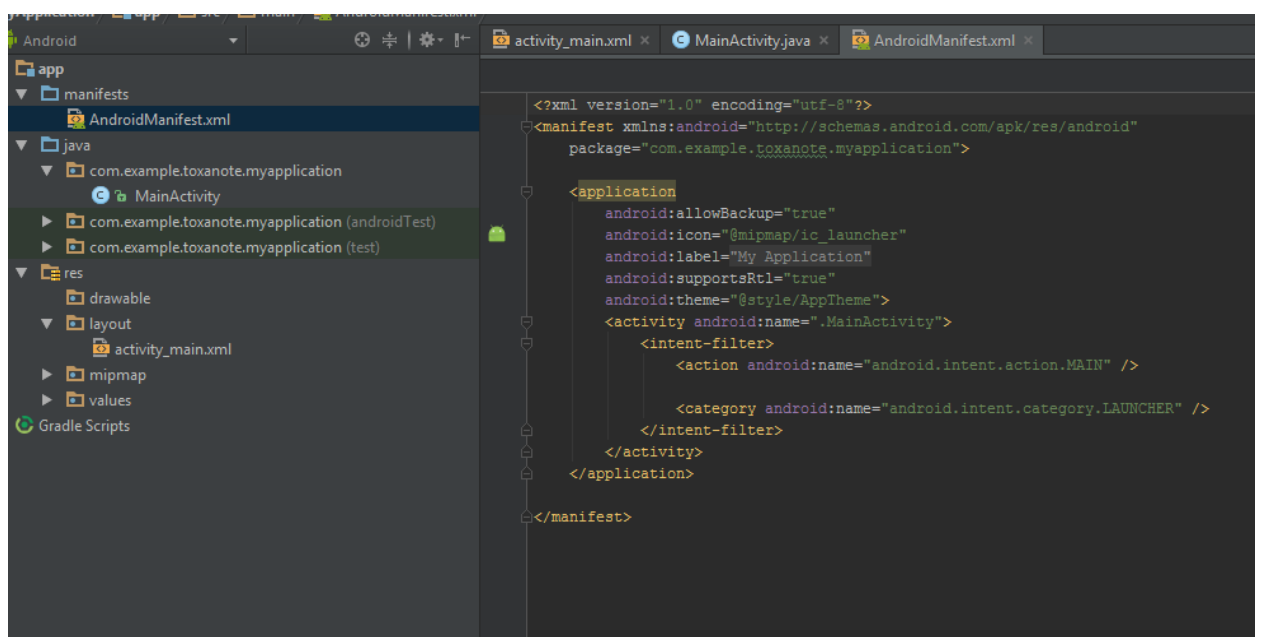


Рисунок 25 — Внешний вид AndroidManifest.xml

Весь исполняемый код на языке java, интерфейсы и классы находятся в папке java. В этой папке находится другие подпапки и файлы, из которых для нас важными являются каталоги androidTest и test предназначенные для хранения файлов тестов приложения. А собственно исходные коды располагаются в папке главном файле с названием проекта (рисунок 26). В нашем

случае это «com.toxa.rsvpu». Странное название обусловлено тем, что в название проекта автоматически добавляется имя пользователя, создавшего проект. Стандартно в этом каталоге находится файл класса MainActivity, который запускается по умолчанию при старте приложения.

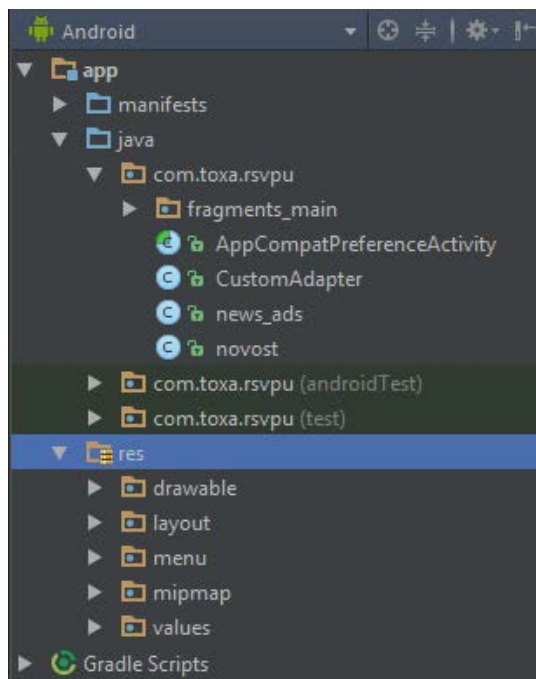


Рисунок 26 — Строение каталога Java

В папке res лежат все ресурсы, которые будет использовать приложение. Они распределены по подпапкам:

- drawable – для графических файлов растровых и векторных форматов;
- layout – для файлов разметки Activity формата xml. По умолчанию здесь есть два файла activity_main.xml и content_main.xml, которые определяют интерфейс для стандартной activity – MainActivity;
- menu для разметки различных меню и подменю;
- mipmap – По сути, это замена ресурсам drawable для значков приложения. Появилось в более поздних версиях android;
- values – содержит текстовые переменные на различных языках и файлы настроек;

Файл build.gradle содержит информацию, которая используется при по-

строении проекта. Здесь можно подключить дополнительные модули и расширения к проекту.

2.6 Схема расположения элементов и структура меню

Для разработки приложения была продумана, и разработана схема взаимного расположения его частей.

От схемы напрямую зависит, сможет ли пользователь найти интересующий его материал.

Приложение «РГППУ» состоит из четырех экранов и семи фрагментов. При такой структуре приложение показывает наилучшие результаты производительности и удобства. Схема приложения согласована, и утверждена заказчиком (рисунок 27).

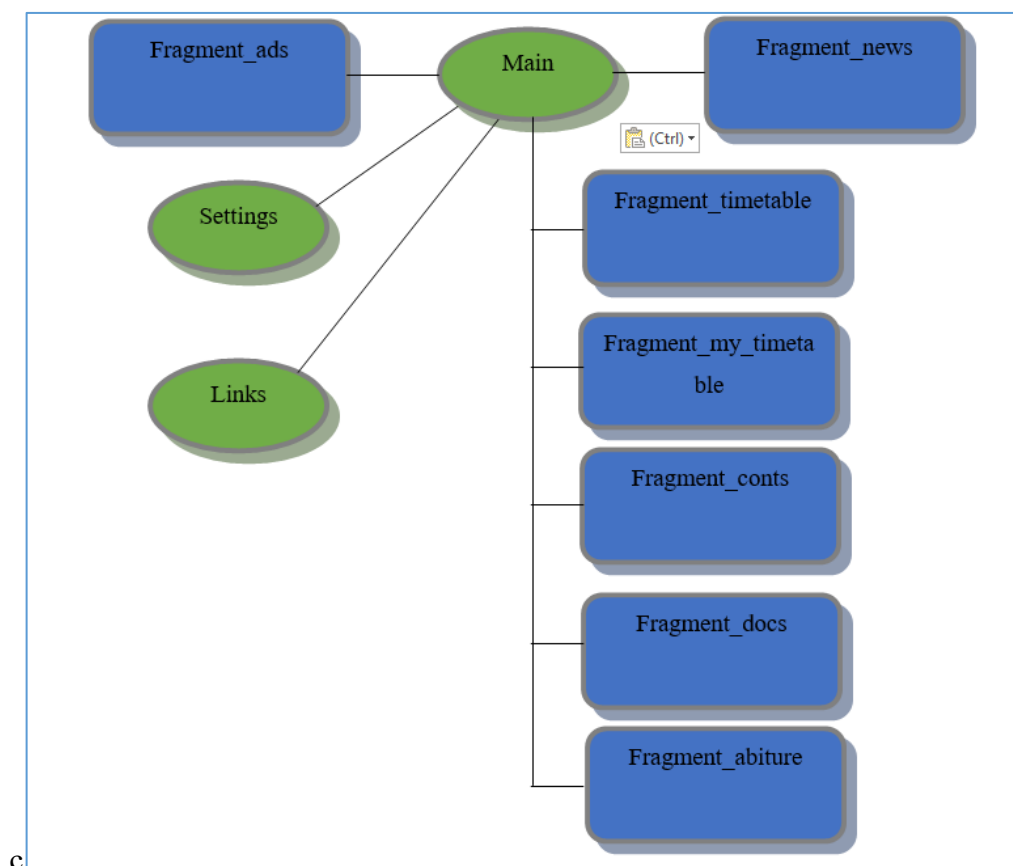


Рисунок 27 — Структура приложения

То, из каких частей состоит приложение, и как эти части располагаются на сайте очень важно для достижения целей разработки приложения.

Тщательно обдумывается возможный путь пользователей (переходы по ссылкам) по экранам приложения. Навигация должна обеспечить доступность предоставляемой информации.

На рисунке 28 схематично показана структура интерфейса мобильного приложения. Она показывает расположение активных элементов на экране приложения.



Рисунок 28 — Структура интерфейса главного экрана

Структура меню приложения и алгоритмы взаимодействия пользователей с приложением. Приложение содержит разделы, доступные для всех пользователей.

- «Новости» – раздел, дублирующий одноименный блок с сайта. Содержит новости о уже прошедших событиях;
- «Объявления» – аналогичный новостям раздел. Тут публикуется информация о планируемых событиях;
- «Абитуриенту» – раздел с информацией для абитуриентов
- «Студенту» – раздел с контактной информацией деканатов и полезных ссылок;
- «Расписание» – мобильная версия расписания занятий, сделанная в рамках курсовой работы;

- «Мое расписание» – то же расписание, но сразу открывающее раздел или определенную группу, предустановленную в настройках;
- «Мои документы» – личный кабинет студента;
- «Контакты» – раздел содержащий основные контактные данные, такие как телефон компании, сайт, адрес;
- «Ссылки» – блок, содержащий ссылки на официальные ресурсы университета;
- «Настройки» – меню, включающее в себя настройки работы со стандартным расписанием.

2.7 Описание навигационных пунктов и их программная реализация

2.7.1 Новости и объявления и выдвигающееся вертикальное меню

Главный экран MainActivity состоит из трех компонентов:

- activity_news_ads_main – форма для объединения всех фрагментов в одно целое;
- activity_news_ads – содержит пустой фрагмент, который будет заменяться другими при взаимодействии с различными элементами (кнопками, ссылками т.д.);
- nav_header_bar – форма выдвигающегося вертикального меню.

При открытии приложения запускается метод onCreate() сразу подставляется фрагмент news (рисунок 29).

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_news_ads_main);
    Toolbar toolbar1 = (Toolbar) findViewById(R.id.toolbar1);
    setSupportActionBar(toolbar1);
}
```

Рисунок 29 — Подставление фрагмента activity_news_ads_main

При свайпе вправо срабатывает метод `getItem()` внутреннего класса `SectionsPagerAdapter`, который как аргумент принимает номер новой страницы и в зависимости от него в операторе `case` выводит на экран требуемый фрагмент. (рисунок 30).

```
public class SectionsPagerAdapter extends FragmentPagerAdapter {

    public SectionsPagerAdapter(FragmentManager fm) { super(fm); }

    @Override
    public Fragment getItem(int position) {
        // getItem is called to instantiate the fragment for the given page.
        // Return a PlaceholderFragment (defined as a static inner class below).
        switch (position){
            case 0:
                news n1 = new news();
                return n1;
            case 1:
                ads ads = new ads();
                return ads;
        }
    }
}
```

Рисунок 30 — Механизм переключения вкладок

Оба приведенных выше фрагмента имеют одинаковую структуру (рисунок 31).

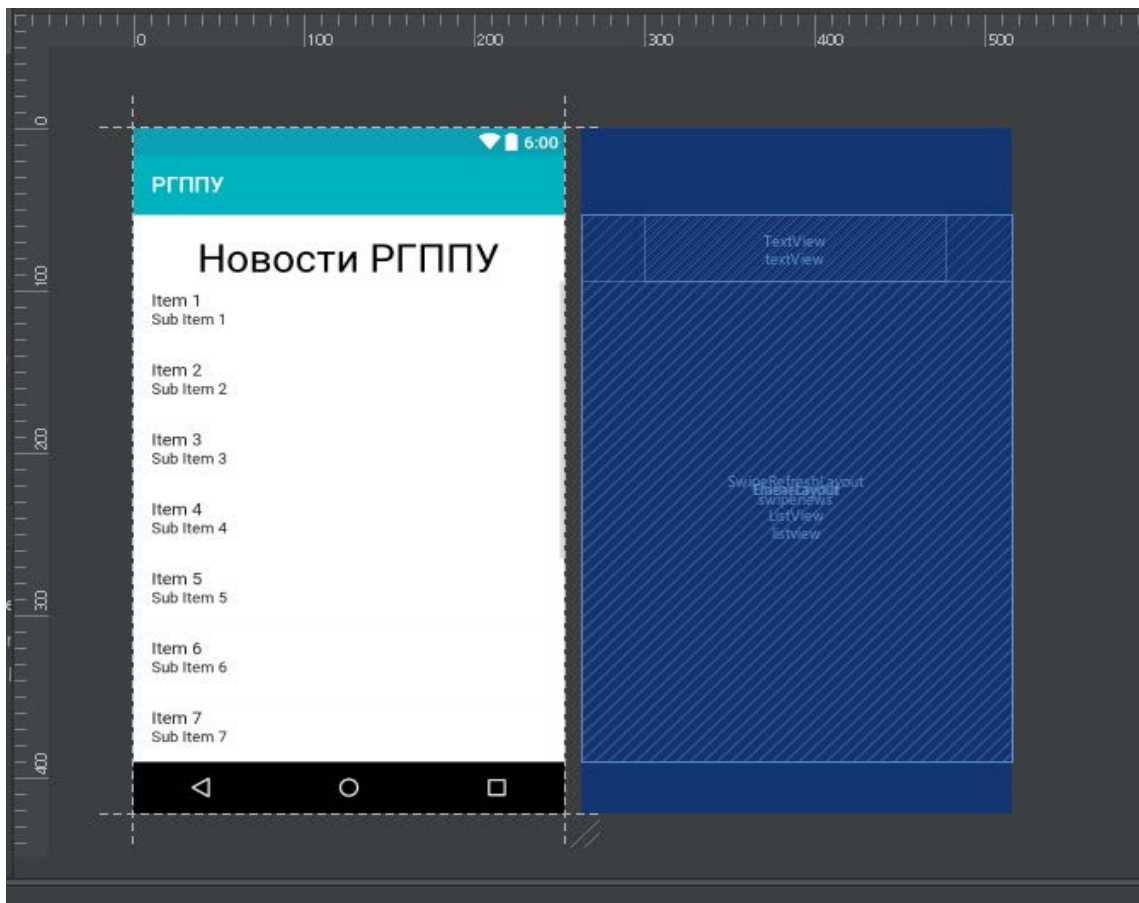


Рисунок 31 — Структура фрагмента news

Как видно из рисунка основной лентой является элемент ListView. С помощью адаптера данный список заполняется отдельными элементами описанными в CustomAdapter, как видно из рисунка 32:

- ImageView – место для картинки (отмечено звездочкой);
- TextView – заголовок;
- TextView2 – дата публикации;
- TextView3 – содержание новости.

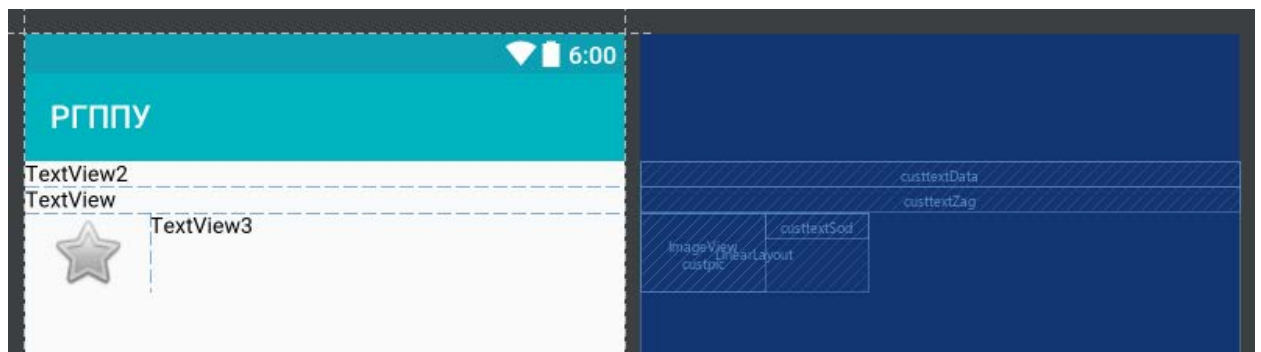


Рисунок 32 — Структура адаптера

Для получения этих данных от сервера мы используем асинхронный метод Async.ParseTask() (рисунок 33).

```
@Override
protected String doInBackground(Void... params) {
    // получаем данные с внешнего ресурса
    try {
        URL url = new URL("http://www.rsvpu.ru/contents/news/news.php");

        urlConnection = (HttpURLConnection) url.openConnection();
        urlConnection.setRequestMethod("GET");
        urlConnection.connect();
        urlConnection.getResponseMessage();

        InputStream inputStream = urlConnection.getInputStream();
        StringBuffer buffer = new StringBuffer();

        reader = new BufferedReader(new InputStreamReader(inputStream));

        String line;
        while ((line = reader.readLine()) != null) {
            buffer.append(line);
        }

        result = buffer.toString();
    }
}
```

Рисунок 33 — Асинхронный метод, получающий данные с сервера

Как видно из рисунка сначала отправляется Http-запрос к файлу news.php на сервере университета, размещенный там сотрудниками. Как выглядит запрос можно увидеть на рисунке 34.

```

7
8 //***** Подключение к базе *****
9 //Sdb=mssql_connect($hostname,$username,$password) or die ( 'Something went wrong while connecting to MSSQL (all)' );
10 $db=mssql_connect($GLOBALS['hostname'],$GLOBALS['username'],$GLOBALS['password']) or die ( 'Something went wrong while connecting to MSSQL (all)' );
11 mssql_select_db($GLOBALS['dbName'],$db);
12 //***** КОНЕЦ Подключение к базе *****
13
14 $select = "
15 SELECT * FROM
16 (
17     SELECT
18         dbo.Publications.Pub_id as 'id',
19         dbo.Publications.Pub_Title as 'title',
20         dbo.Publications.Pub_Description as 'content',
21         t2.create_date,
22         t2.actual_from_date,
23         t2.actual_to_date,
24         dbo.Publications.Pub_page_id as 'page_id',
25         t2.cat_id,
26         dbo.Contents.Content as 'link'
27     FROM
28         dbo.Publications
29         INNER JOIN
30         (
31             SELECT DISTINCT
32                 dbo.Distribution.Pub_id,
33                 dbo.Distribution.cat_id,
34                 Convert(varchar(18), dbo.Publications.Pub_CreateDate, 20) as 'create_date',
35                 Convert(varchar(18), dbo.Distribution.actual_from, 20) as 'actual_from_date',
36                 Convert(varchar(18), dbo.Distribution.actual_to, 20) as 'actual_to_date'
37             FROM
38                 dbo.Publications
39                 INNER JOIN dbo.Distribution ON dbo.Publications.Pub_id=dbo.Distribution.Pub_id
40                 INNER JOIN dbo.Contents ON dbo.Publications.Pub_id=dbo.Contents.pub_id
41             WHERE
42                 dbo.Distribution.date_from<=CURRENT_TIMESTAMP AND
43                 dbo.Distribution.date_to>CURRENT_TIMESTAMP AND
44                 dbo.Distribution.is_active=1 AND
45                 dbo.Contents.contentType='redirect' and
46                 (dbo.Distribution.cat_id = 38)
47             ) as t2 ON dbo.Publications.Pub_id=t2.Pub_id
48             INNER JOIN dbo.Contents ON dbo.Publications.Pub_id=dbo.Contents.pub_id
49         ) as tbl1

```

Рисунок 34 — Асинхронный метод, получающий данные с сервера

Затем программа получает ответ и в методе `doInBackground()`, который выполняется уже после `ParseTask()`, мы делим его на элементы массива. После чего в цикле заполняем элементы адаптера полученными данными (рисунок 35).

```

@Override
protected void onPostExecute(String str) {
    super.onPostExecute(str);

    String[] s = str.split("title");
    for (String i : s) {
        if (i.length() > 10) {
            novost n = new novost();
            int icont = i.indexOf("content");
            int ipic = i.indexOf("picture");
            int idate = i.indexOf("date");
            n.setZagolovok(i.substring(0, icont));
            n.setData(i.substring(idate + 4));
            n.setKartinka(i.substring(ipic + 7, idate));
            n.setSoderzhanie(i.substring(icont + 7, ipic));
            list.add(n);
        }
    }
    listView.setAdapter(adapter);
}

```

Рисунок 35 — Заполнение массив данными

На главном экране – MainActivity, расположены следующие элементы:

TabActivity – расширение для Activity позволяющее переключаться между фрагментами news и ads путем перелистывания вправо и влево. Фрагменты news и ads выводят на экран списки новостей и объявлений соответственно.

Navigation drawer – выдвигающееся вертикальное меню открывающееся при свайпе вправо от левого края экрана. Это событие обрабатывается в классе Main в приемнике событий под названием addDrawerListener() (рисунок 36).

```
Toolbar toolbar2 = (Toolbar) findViewById(R.id.toolbar);
DrawerLayout drawer = (DrawerLayout) findViewById(R.id.drawer_layout2);
ActionBarDrawerToggle toggle = new ActionBarDrawerToggle(
    this, drawer, toolbar2, "Open navigation drawer", "Close navigation drawer");
drawer.addDrawerListener(toggle);
toggle.syncState();

NavigationView navigationView = (NavigationView) findViewById(R.id.nav_view2);
View headerLayout = navigationView.getHeaderView(0);
navigationView.setNavigationItemSelectedListener(this);
```

Рисунок 36 — Слушатель событий меню

На нажатие пунктов данного меню настроено еще одно событие onNavigationItemSelectedListener(). Оно обрабатывает нажатие на пункты и запускает код, который принимает как аргумент ID нажатого элемента и в операторе выбора вариантов «Case» заменяет текущий фрагмент, находящийся в Activity на фрагмент с указанным ID (рисунок 37), тем самым позволяющее переключаться между фрагментами abiture, conts, docs, links, timetable.

Кроме того, в этом же событии функция getSupportActionBar().hide() скрывает шапку с кнопками переключением между новостями и объявлениями при переходе на другие фрагменты, такие как: fragment_conts, fragment_abiture, fragment_docs, fragment_timetable, fragment_links.

```

public boolean onNavigationItemSelected(MenuItem item) {
    // Handle navigation view item clicks here.
    int id = item.getItemId();
    FragmentTransaction ftrans = getFragmentManager().beginTransaction();

    if (id == R.id.nav_timetable) {
        ftrans.replace(R.id.activity_news_ads, ftimetable);
        getSupportActionBar().hide();
    } else if (id == R.id.nav_abiture) {
        ftrans.replace(R.id.activity_news_ads, fabiture);
    } else if (id == R.id.nav_news_ads) {
        getSupportActionBar().show();
    } else if (id == R.id.nav_conts) {
        ftrans.replace(R.id.activity_news_ads, fconts);
        getSupportActionBar().hide();
    } else if (id == R.id.nav_docs) {
        ftrans.replace(R.id.activity_news_ads, fdocs);
        getSupportActionBar().hide();
    }
    else if (id == R.id.nav_links) {
        ftrans.replace(R.id.activity_news_ads, flinks);
        getSupportActionBar().hide();
    }
}

```

Рисунок 37 — Обработчик нажатий на пункты меню

Результат отображения представлен на рисунке 38.

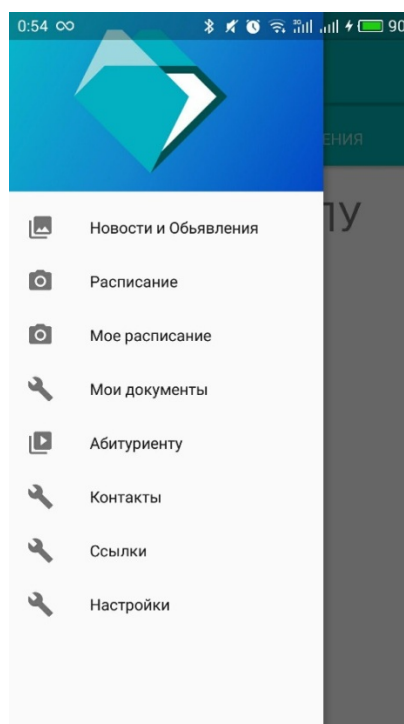


Рисунок 38 — Выдвигающееся вертикальное меню

2.7.2 Пункты «Абитуриенту» и «Студенту»

Фрагменты `fragment_student` и `fragment_abiture` идентичны и не получают информацию с сервера, а целиком содержатся в приложении. Структуру фрагментов можно увидеть на рисунке 39.

В пункте «Студенту» содержатся ссылки на актуальные информационные системы и телефоны деканатов и Управления сопровождения обучающихся.

В пункте «Абитуриенты» содержатся контакты приемной комиссии и ссылки на план приема и видео.

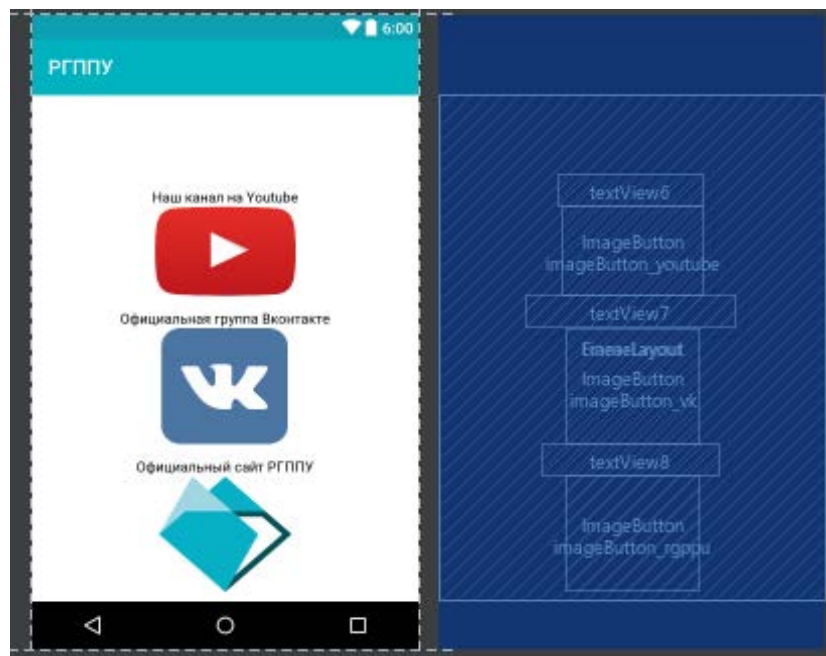


Рисунок 39 — Выдвигающееся вертикальное меню

2.7.3 Вывод расписания и сохранение настроек

Один из фрагментов, открывающихся в `MainActivity` и заменяющих собой стандартный фрагмент. Его структура очень проста. Он состоит из одного элемента `WebView` который является, по своей сути, встроенным в приложение окном браузера. В классе `fragment_timetable.java` в методе `onCreateView()` мы загружаем с помощью функции `loadUrl()` мобильную версию рас-

писания с сайта нашего университета и заполняем ей фрагмент (рисунок 40). С виду данный элемент ничем не будет отличаться от созданного внутри приложения.

```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                          Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_timetable, container, false);
    WebView w1 = (WebView) view.findViewById(R.id.timetable_web);
    w1.getSettings().setJavaScriptEnabled(true);
    w1.getSettings().setJavaScriptCanOpenWindowsAutomatically(true);
    w1.setWebViewClient(new Timetable.MyWebViewClient());
    w1.loadUrl("http://rsvpu.ru/mobile");
    // Inflate the layout for this fragment
    return view;
}
```

Рисунок 40 — Загрузка мобильного расписания с сайта

Меню содержит 8 элементов. При нажатии на любой из них содержащее Activity заменяется одним из 5 фрагментов (рисунок 41).

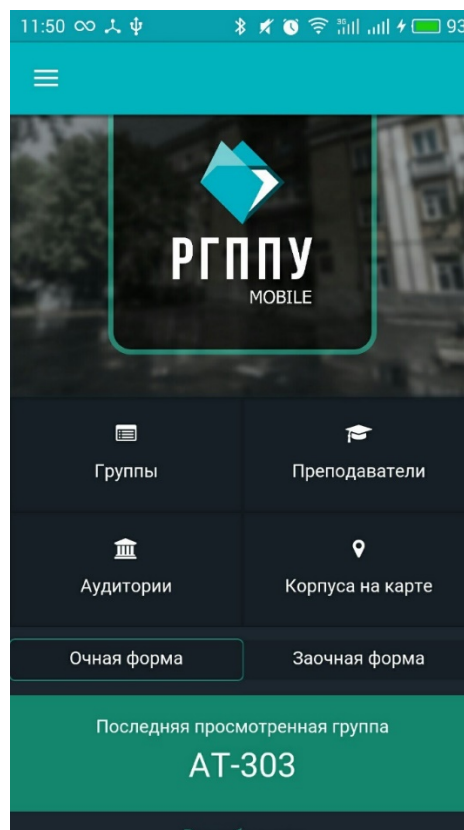


Рисунок 41 — Один из доступных фрагментов

Или же просто открывает другое Activity. Например, SettingsActivity – экран настроек (рисунок 42).

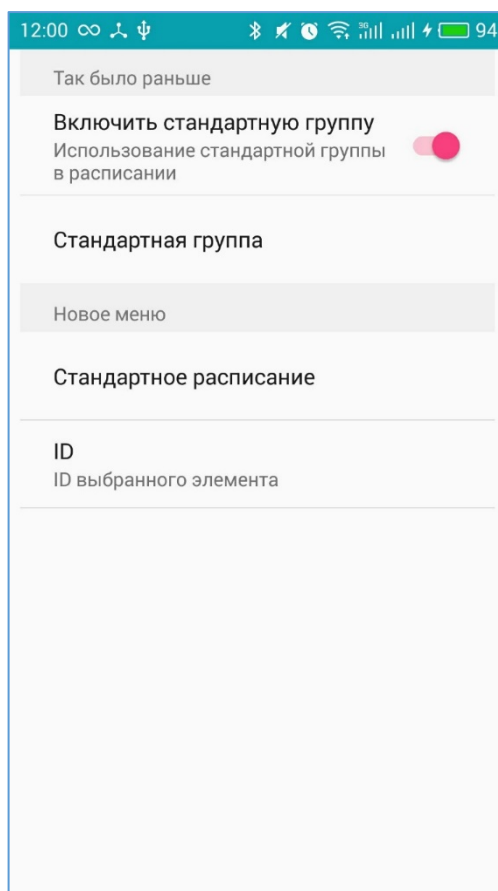


Рисунок 42 — Экран настроек

Пункт «Настройка» содержит выбор формы стандартного расписания и идентификационный номер стандартного элемента (группы, преподавателя или аудитории в зависимости от предыдущей настройки). Номер элемента нужно заранее записать из адресной строки компьютерной версии расписания (рисунок 43).

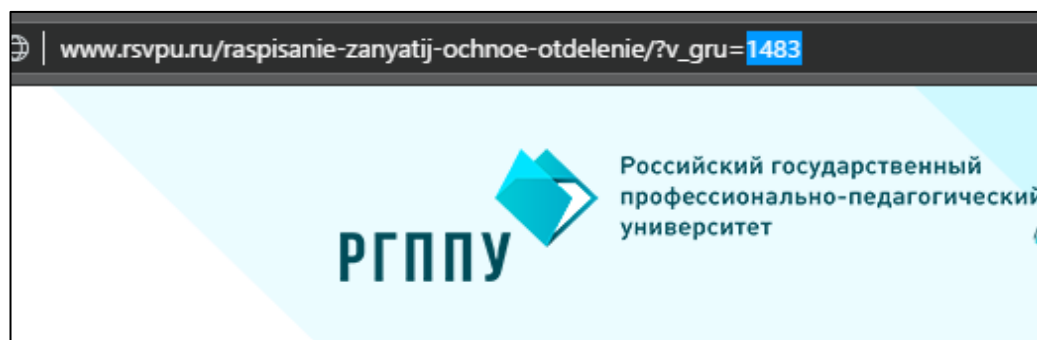


Рисунок 43 — Идентификационный номер элемента расписания

Фрагмент «fragment_my_timetable» отличается тем, что он сразу открывает расписание, предустановленное в SettingsActivity. В зависимости от настройки «chosen_subj» мы вызываем метод loadUrl() с различными ссылками в аргументе, ведущими на расписание по группам, преподавателям или аудиториям (рисунок 44).

```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                          Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_timetable, container, false);
    WebView w1 = (WebView) view.findViewById(R.id.timetable_web);
    w1.getSettings().setJavaScriptEnabled(true);
    w1.getSettings().setJavaScriptCanOpenWindowsAutomatically(true);
    w1.setWebViewClient(new MyTimetable.MyWebViewClient());
    SharedPreferences prefs = PreferenceManager
        .getDefaultSharedPreferences(getActivity().getApplicationContext());
    String grr = prefs.getString("chosen_subj", "");
    String id = prefs.getString("id_chosen", "");
    switch (grr)
    {
        case "gr":
            w1.loadUrl("http://192.168.1.5/?v_gru=" + id);
            break;
        case "pr":
            w1.loadUrl("http://192.168.1.5/?v_prep=" + id);
            break;
        case "au":
            w1.loadUrl("http://192.168.1.5/?v_aud=" + id);
            break;
        case "no":
            w1.loadUrl("http://192.168.1.5/");
            break;
    }

    // Inflate the layout for this fragment
```

Рисунок 44 — Механизм открытия стандартного расписания

Также в этот аргумент подставляется значение переменной ID полученной ранее в методе prefs.getString(), который получает значение, сохраненное в настройках. Таким образом, задав однажды определенный ID (который можно посмотреть в обычной версии расписания, в адресной строке) и тип элемента, в данном фрагменте всегда будет сразу открываться нужное расписание.

2.7.4 Пункт «Мои документы»

Фрагмент docs_fragment содержит компонент WebView (рисунок 45). В него загружается веб-страница личного кабинета студента [20].



Рисунок 45 — Фрагмент «Мои документы»

2.7.5 Пункт «Ссылки»

LinksActivity – экран ссылок (рисунок 46) на официальные электронные ресурсы университета:

- официальный канал университета на портале «YouTube»;
- официальная группа университета в социальной сети «ВКонтакте»;
- официальный сайт РГППУ.



Рисунок 46 — Экран ссылок

При нажатии на элемент откроется новое окно браузера с выбранным ресурсом (рисунок 47).

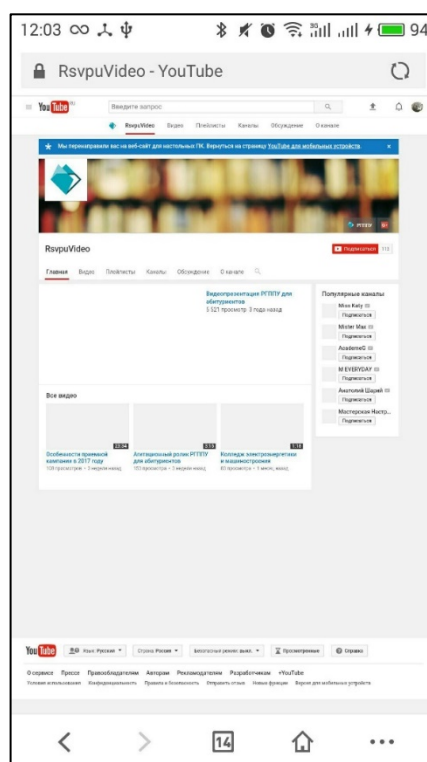


Рисунок 47 — Ресурс, открытый по ссылке

2.7.6 Пункт «Контакты»

Фрагмент контактов (рисунок 48), аналогично предыдущим пунктам, содержит всего один элемент WebView. В него, с локального сервера, загружается веб-страница контактов, сделанная в рамках курсовой работы.



Рисунок 48 — Фрагмент контактов

2.8 Компиляция приложение и тестовый запуск

На текущий момент многие ресурсы загружаются с локального сервера на домашнем компьютере. Сервер развернут через программу OpenServer. Они будут размещены сотрудниками на сервере вуза. После этого будет добавлена ссылка на сайт вуза для загрузки приложения.

В AndroidStudio для компиляции приложения используется Gradle – система автоматической сборки, построенная на принципах Apache Ant и Apache Maven.

С использованием в Android Studio в режиме Build Variant напротив модуля в поле Build Variant переключается режим сборки с debug на release (рисунок 49).

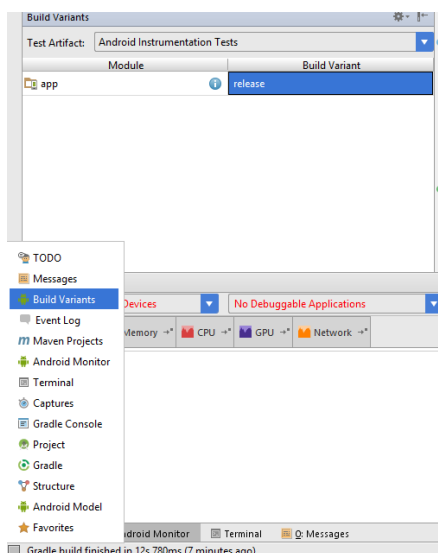


Рисунок 49 – Запуск Build Variant

В основном меню через Build → Generate Signed APK. запускается «Gradle assembleRelease», и получается сгенерированный арк-файл в папке build/apk/. Пример приложения в списке ярлыков представлен на рисунке 50.

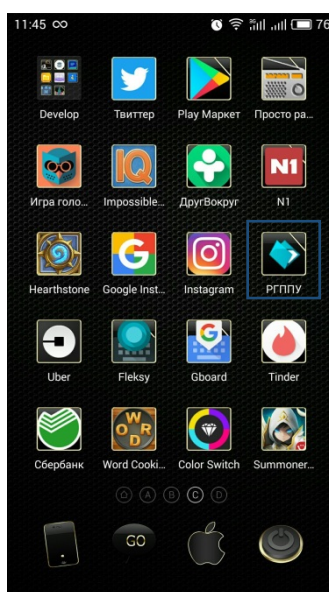


Рисунок 50 — Фрагмент контактов

На текущий момент многие ресурсы загружаются с локального сервера на базе Open Server на домашнем компьютере.

После тестирования и размещения скриптов на сайте ВУЗа, приложение он будет добавлен в Google Play и ссылка будет размещена на сайте вуза для загрузки приложения.

ЗАКЛЮЧЕНИЕ

В рамках выпускной квалификационной работы было создано мобильное приложение, позволяющее пользователю получить информацию об Российском государственном профессионально-педагогическом университете.

Был произведён анализ существующих технологий разработки мобильных приложений, исходя из аналитики и ссылаясь на техническое задание, выбрана операционная система мобильного устройства для работы приложения, а также среда разработки.

Приложение «РГППУ» было разработано по заказу Российского государственного профессионально-педагогического университета.

В ходе выполнения выпускной квалификационной работы были решены следующие задачи:

- проанализированы предметная область и аналогичные проекты в магазине приложений «Google Play»;
- проанализированы исходные данные заказчика;
- подготовлен проект продукта;
- реализовано мобильное приложение под управлением операционной системы Android;
- приложение наполнено данными взятыми из базы данных университета.

Результатом работы является создание мобильного приложения для Российского государственного профессионально-педагогического университета.

Разработанный проект удовлетворяет всем требованиям, технического задания, предоставленного заказчиком.

Таким образом, следует считать, что задачи выпускной квалификационной работы полностью решены и цель исследования достигнута.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Гриффитс Р. Д. Head First. Программирование для Android [Текст] / Р. Д. Гриффитс – Санкт-Петербург: Питер, 2016. – 704 с.
2. Дерси Л. Создание приложений для Android за 24 часа [Текст] / Л. Л. Дерси – Москва: Эксмо, 2015. – 464 с.
3. Дэрси Л. Разработка приложений для Android-устройств. Базовые принципы [Текст] /Л. Дэрси, Ш. Кондер – Том 1. – Москва: Эксмо, 2014. – 598 с.
4. Использование мобильных телефонов в разных странах [Электронный ресурс]. – Режим доступа: <http://www.cossa.ru/152/37433/> (дата обращения: 15.05.2017).
5. Использование мобильных устройств [Электронный ресурс]. – Режим доступа: <http://www.wi-life.ru/stati/wi-fi/marketingovye-stati-2/mobile-devices-use-aruba-research-results> (дата обращения: 15.05.2017).
6. Майер Р. Программирование приложений для планшетных компьютеров и смартфонов [Текст] / Р. Майер – Москва: Эксмо, 2013. – 816 с.
7. Мобильное приложение, что такое, определение, новости, статьи, видео [Электронный ресурс]. – Режим доступа: <https://indicator.ru/tags/mobilnoe-prilozhenie/> (дата обращения: 15.05.2017).
8. Основные этапы разработки мобильных приложений [Электронный ресурс] – Режим доступа: <https://spark.ru/startup/componentix/blog/4499/osnovnie-etapi-razrabotki-mobilnih-prilozhenij> (дата обращения: 15.05.2017).
9. Приложения в Google Play – НГУЭУ [Электронный ресурс]. – Режим доступа: <https://play.google.com/store/apps/details?id=burov.nsuem> (дата обращения: 15.05.2017).
10. Приложения в Google Play – MSU-Life [Электронный ресурс]. – Режим доступа: <https://play.google.com/store/apps/details?id=com.springlabs.msulife&hl=ru> (дата обращения: 15.05.2017).

11. Приложения в Google Play- MTI Mobile [Электронный ресурс]. – Режим доступа: <https://play.google.com/store/apps/details?id=ru.edu.mti.mtimobile> (дата обращения: 15.05.2017).
12. Рынок мобильных приложений в России и мире [Электронный ресурс]. – Режим доступа: http://www.json.ru/poleznye_materialy/free_market_watches/analytics/rynok_mobilnyh_prilozhenij_v_rossii_i_mire/ (дата обращения: 15.05.2017).
13. Технологии разработки мобильных приложений [Электронный ресурс]. – Режим доступа: <https://rae.ru/forum2012/219/2709> (дата обращения: 15.05.2017).
14. Android [Электронный ресурс]. – Режим доступа: <http://www.android.com/> (дата обращения: 15.05.2017).
15. Android-4pda [Электронный ресурс]. – Режим доступа: <http://4pda.ru/forum/index.php?showforum=281> (дата обращения: 15.05.2017).
16. Habrahabr.ru [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/posts/programming/> (дата обращения: 15.05.2017).
17. Intuit.ru [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru/studies/courses/3703/945/info> (дата обращения: 15.05.2017).
18. Start Android – учебник по Android для начинающих и продвинутых [Электронный ресурс]. – Режим доступа: <http://startandroid.ru/ru/> (дата обращения: 15.05.2017).
19. Habrahabr.ru [Электронный ресурс]. – Режим доступа: <https://habrahabr.ru/post/168843/> (дата обращения: 05.06.2017).
20. Личный кабинет студента [Электронный ресурс]. – Режим доступа: <http://uso.rsvpu.ru/Service> (дата обращения: 05.06.2017).
21. Анализ сайта [Электронный ресурс]. – Режим доступа: <http://prcy.ru/> (дата обращения: 05.06.2017).
22. Официальная группа РГППУ [Электронный ресурс]. – Режим доступа: <https://vk.com/rsvpu> (дата обращения: 09.06.2017).

23. Официальный видеоканал РГППУ [Электронный ресурс]. – Режим доступа: <https://www.youtube.com/user/RsvpuVideo> (дата обращения: 09.06.2017).

ПРИЛОЖЕНИЕ

**Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования**

«Российский государственный профессионально-педагогический университет»

Институт инженерно-педагогического образования
Кафедра информационных систем и технологий
направление 44.03.04 Профессиональное обучение (по отраслям)
профиль «Информатика и вычислительная техника»
профилизация «Компьютерные технологии»

УТВЕРЖДАЮ

Заведующий кафедрой

_____ Н. С. Толстова

« ____ » _____ 2017 г.

**ЗАДАНИЕ
на выполнение выпускной квалификационной работы бакалавра**

студента 4 курса, группы КТ-412 Гришина Антона Сергеевича

1. Тема: Приложение для вуза «Мобильный РГППУ»

утверждена распоряжением по институту от 07.02.2017 г. № 73.

2. Руководитель Ченушкина Светлана Владимировна, старший преподаватель кафедры ИС

3. Место преддипломной практики: Российский государственный профессионально-педагогический университет

4. Исходные данные к ВКР: Дэрс Л. «Разработка приложений для Android-устройств. Базовые принципы», Использование мобильных телефонов в разных странах, Основные этапы разработки мобильных приложений, Приложения в Google Play – НГУЭУ

5. Содержание текстовой части ВКР (перечень подлежащих разработке вопросов):

Анализ литературы и интернет-источников.

Анализ аналогичных мобильных приложений.

Обзор мобильных операционных систем

Разработка структуры приложения

Разработка приложения

Компиляция и тестовый запуск приложения

6. Перечень демонстрационных материалов:

Презентация, выполненная средствами Microsoft Office PowerPoint

7. Календарный план выполнения выпускной квалификационной работы

№ п/п	Наименование этапа дипломной работы	Срок выполнения этапа	Процент выполнения ВКР	Отметка руководителя о выполнении
1	Сбор информации по выпускной работе и сдача зачета по преддипломной практике	10.04.2017	15	
2	Выполнение работ по разрабатываемым вопросам их изложение в выпускной работе:			
	Изучение технологии разработки мобильных приложений	30.04.2017	10	
	Анализ литературы и интернет источников	09.05.2017	10	
	Разработка структуры мобильного приложения	15.05.2017	15	
	Разработка мобильного приложения	31.05.2017	25	
	Тестирование приложения на рабочих системах университета	04.06.2017	5	
3	Оформление текстовой части ВКР	05.06.2017	5	
4	Выполнение демонстрационных материалов к ВКР	10.06.2017	5	
5	Нормоконтроль	12.06.2017	5	
6	Подготовка доклада к защите в ГЭК	19.06.2017	5	

8. Консультанты по разделам выпускной квалификационной работы

Наименование раздела	Консультант	Задание выдал		Задание принял	
		подпись	дата	подпись	дата

Руководитель _____
подпись дата

Задание получил _____
подпись студента дата

9. Выпускная квалификационная работа и все материалы проанализированы. Считаю возможным допустить Гришина А. С. к защите выпускной квалификационной работы в государственной экзаменационной комиссии.

Руководитель _____
подпись дата

10. Допустить Гришина А. С. к защите выпускной квалификационной работы в государственной экзаменационной комиссии (протокол заседания кафедры от 14.06.2017 №12)

Заведующий кафедрой _____
подпись дата