

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
**«Национальный исследовательский ядерный университет «МИФИ»
(НИЯУ МИФИ)
Предуниверситарий НИЯУ МИФИ**

**Выпускная работа обучающегося IT-класса Предуниверситария НИЯУ
МИФИ**

«Гид в мире цифрового контента»

Обучающиеся

Ошаров Александр Андреевич, 10 “П” класс
Добычин Артем Константинович, 10 “П” класс
Гасанов Рамиль Эльсеверович, 10 “П” класс

Дата защита:

Результат защиты:

Реферат

Постоянный переход из разных приложений для просмотра интересующей информации и “хранение” на устройстве несколько разных приложений является неудобным и неэкономичным по памяти.

Для того, чтобы повысить комфорт просмотра информации, было принято решение объединить несколько приложений в одно.

В работе рассматриваются основные способы парсинга информации, база данных, анализ интересов потенциальных пользователей и учет интересов пользователей с помощью системы категорий.

Отчет состоит из 20 страниц, включающих в себя:

- схемы;
- скриншоты;
- таблицы.

Ключевые слова: Parsing, API, Telegram, VK, Reddit, Firebase, RealtimeDatabase, MVVM, Android.

Оглавление

Реферат.....	1
Введение	2
Описание	4
Единомышленники	4
Техническая составляющая	5
Исследование	6
Основная часть.....	10
Список литературы.....	14

Введение

Всемирная паутина ведет свой прогрессивный путь, начиная с 1964 года, тогда, когда понятие “у меня дома свой компьютер” не существовало. Можно было лишь работать с электронными машинами в специализированных больших комнатах за 26-метровым контейнером, не говоря уже об Интернете... Прошло целых 59 лет, за такое время много чего изменилось. Сегодня уже очень трудно будет найти человека, у которого не будет компьютера, телефона или планшета, даже моя бабушка, будучи не приверженицей гаджетов, пользуется смартфоном. А там, где есть спрос, всегда будут продавать: игры, научные приложения, музыку, фильмы и так далее. Началась эпоха “Восхождения” и “Купли-продажи”, начиная с грандиозных проектов, как vk, reddit, telegram, tiktok, которые предоставляют своим пользователям безграничные возможности. Все эти хорошо структурируемые системы сочетают в себе две необходимые человеку для развития потребности. И прежде всего это ресурсы. Именно ресурсы дают ключ к неизмеримым пространственным взаимодействиям. Благодаря ресурсам человеческий мозг начинает развиваться, прогрессировать, эволюционировать. Наши восприятия относительно дома, в котором живем, парка, в котором гуляем, города, страны, планеты, галактической системы целиком и полностью зависят от окружающих нас ресурсов, которые пододвигают к исследованиям, следовательно, к новым открытиям, которые предоставляют новые ресурсы, которые вновь пододвигают к исследованиям и так далее. Только благодаря необходимым ресурсам человек может обучаться, познавать науки и защищать диссертации. Именно ресурсы двигают человечество к прогрессу, закрывая доступ к регрессу.

Многие люди, живущие в деревнях, селах, маленьких городишках еле-еле могут позволить себе Интернет, который наполнен этими ресурсами, но которые в большинстве своём требуют денежную оплату. Очевидно, что не все готовы платить за курс, например, программирование на C++ 30 000 рублей в месяц, это слишком большая сумма, не позволяющая талантливым умам показать свои возможности. Упускается, то единственное, за что так многие борются — знания.

Все мы в детстве верим в чудеса. Будучи шестилетними детишками, мы мечтали не о самолетиках на пульте управления, нет, мы мечтали о мире возможностей. Ведь самолетик мы можем построить сами, даже не имея должного опыта, но создать летающий ранец без необходимых навыков в 6 лет довольно трудно.

Сегодня мы живем в мире технологий, нас окружают нейронные сети, способные написать диплом. Это двигает на путь будущего, будущих свершений, будущих реализаций. Но именно это и рушит всю экономическую, политическую, социальную сферы. Такой подход не позволяет развиваться, лишь способствует деградации. Но есть выход: те самые ресурсы можно брать и другим путём, который доступен гораздо большему количеству людей, который заставляет думать, а значит — созидать. И этот путь нам дарят, перед нами широко растворяют ворота и предлагают погрузиться в сладостные познания мира с помощью крупнейших компаний в области научного, публицистического, развлекательного контента таких, как vk, reddit, telegram, tiktok. Эти сети содержат в себе терабайты информации, которая так нужна миру. Предприятия несут в себе огромную ценность и потенциал, но в различной форме, под разными предложениями. Посещая данные виртуальные площадки довольно часто, становится невероятно неудобно на постоянной основе переходить с одного приложения на другое, с одного сайта на другой. Память устройства начинает забиваться от большого многообразия, что приводит к ужасным последствиям — люди не выдерживают и отказываются от предложенной им возможности познания. Именно поэтому данной системе необходима новая реализация, новый подход в виде целостного приложения, заключающего в себе все необходимые данные, информацию, ресурсы и контент в удобном и привычном формате.

Описание

Метwor – это мобильное андроид приложение, которое нужно для просмотра контента с разных источников и составления ленты на основе интересов пользователя. Приложение позволяет клиентам не задумываться о нахождении того или иного канала, об его ново выпущенном tutorialе. Метwor даёт возможность забыть об 4ех приложениях и работать в одном, ища в нем при необходимости нужную информацию по различным категориям. Приложение берёт крупнейшие платформы vk, reddit, telegram, tiktok и в зависимости от интересов добавляет контент в его собственную “ленту”.

Такой подход работы с крупными приложениями, содержащими в себе огромное количество каналов, принуждает к хранению информации для дальнейшей в будущем ускоренной работы с пользователем. Но добавление всего разом в базу данных просто невозможно, поэтому мы дали пользователям возможность самостоятельно расширять базу путем добавления новых сообществ. Важно отметить: главной целью приложения является преподношение людям возможности упрощенного бесплатного безаграничного знания, а уже посредственной – повышение удобства просмотра информации крупных приложений.

Единомышленники

Довольно лестно было бы сказать, что наша идея является уникальной. Ведь тогда бы многие компании не согласились с таким заявлением от Метwor. Каждый делец хочет видеть в своём детище изюминку, ту отличительную черту, по характеристике которой не подходит не одно другое. “Если мир кажется Вам холодным, разожгите огни, чтобы сделать его теплее!” — Люси Ларком.

В наших сердцах разгорелся огонь и, дабы разжечь его в других, мы вдохновились нашим проектом. Перед нашими глазами через Метwor мы видим светлый улучшенный мир, построенный всеми нами — людьми.

В сети уже существует нескольких проектов со схожими задумками. Например, файловый хостинг компании Dropbox Inc., включающий персональное облачное хранилище, синхронизацию файлов и программу-клиент. Необходимо сразу отметить схожую черту — “хранилище”, ведь именно хранилищем контента является Метwor. Но, несмотря на это, у приложения Метwor кардинально другой смысл: Метwor преподносит знания, то есть является неким “учебником” для пользователей, и заменяет использование четырех структур на одну тогда, как Dropbox выступает в роли “виртуального жесткого диска”. Невозможно отрицать, что Dropbox не является единомышленником, но он не несет за собой те возможности, которые несет Метwor, и направлен лишь на хранение.

Ещё одним единомышленником можно считать “энциклопедию мемов” — memepedia. Этот сервис основывается на публикации актуальной информации — выступает в роли газеты или же вестника. Сравнивая его с приложением Memwor, невозможно не заметить сильного различия: сервис memepedia работает за счет алгоритма создания — без сбора, без проверки на качество, через ручной расклад. Когда же Memwor предоставляет всевозможную информацию, производит поиск среди уже востребованных компаний, тем самым расширяя круг обзора, охватывая целевую аудиторию, проверяя и показывая только качественный продукт, исключая мат, бранную лексику и так далее.

Подводя итог, можно сделать следующий вывод: Memwor схож со многими идеями мировых производителей, но ни один из них не имеет функционал, умения и исполнимость, которые имеет Memwor.

Техническая составляющая

При разработке проекта использовался ряд технологий:

- язык программирования Kotlin/Java;
 - среда разработки Android Studio;
 - база данных от Firebase;
 - для отрисовки ленты используются сторонние библиотеки (Swiperefreshlayout, Picasso...);
 - для формирования запросов на сервера используются сторонние библиотеки (Retrofit, OkHttpClient).
- 1) При разработке приложения основным языком программирования является Kotlin, являющийся более современным, продвинутым и со своими уникальными особенностям аналогом языка программирования Java. Преимущества которого: упрощенный синтаксис, оформленный код, написание которого на Kotlin, намного компактнее по сравнению с Java — на 30–40 % и т. д. Но полностью отказываться от Java было бы нерационально, например, по параметру скорости Java превосходит Kotlin на 12–15 % для чистых сборок (то есть Kotlin компилируется в разы медленнее, за исключением особых ситуаций). Поэтому в приложении используется 2 языка в связке, тем более, оба языка преобразуют код в байт-код, исполняемый JVM, то есть Java без проблем можно вызвать из Kotlin или наоборот.
 - 2) Firebase RealtimeDatabase - это NoSQL база данных, размещенная в облаке. Данные хранятся в формате JSON и синхронизируются в реальном времени для каждого подключенного клиента. Создание кроссплатформенного приложения с помощью платформ Apple, Android и

JavaScript SDK позволяет пользователям совместно использовать один экземпляр базы данных реального времени и автоматически получать обновления с новейшими данными.

- 3) Вместо типичных HTTP-запросов база данных Firebase Realtime использует синхронизацию данных — каждый раз, когда данные изменяются, любое подключенное устройство получает это обновление в течение миллисекунд.

Исследование

Для создания приложения недостаточно простых знаний и навыков в языках программирования. Необходимо четкое понимание желаний от потенциальных пользователей, чтобы реализовать наиболее подходящую, уникальную и актуальную модель под каждого клиента. Поэтому общее решение команды — было провести социальный эксперимент.

В ходе работы было привлечено небольшое количество людей (349) разных возрастов и профессий:

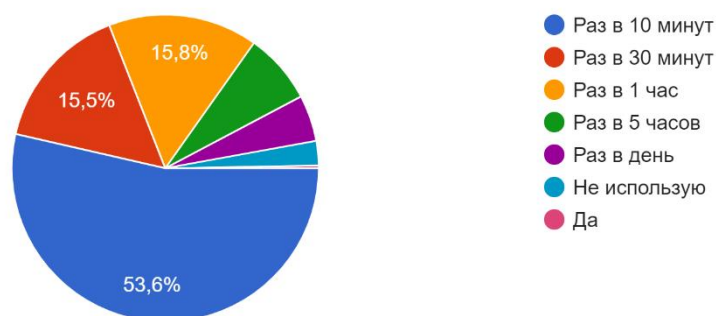
1. Детей и подростков
2. Молодых людей
3. Людей старшего поколения

Всем участникам предлагалось пройти маленький опрос, состоящий из 6 вопросов, 3 из которых были необязательными. Планировалось узнать важные аспекты для понимания важности проекта и значимости его в этом мире. Ниже будут представлены 5 вопросов и диаграмма ответов на них.

1)

Как часто Вы используете приложения vk, telegram, tiktok, reddit?

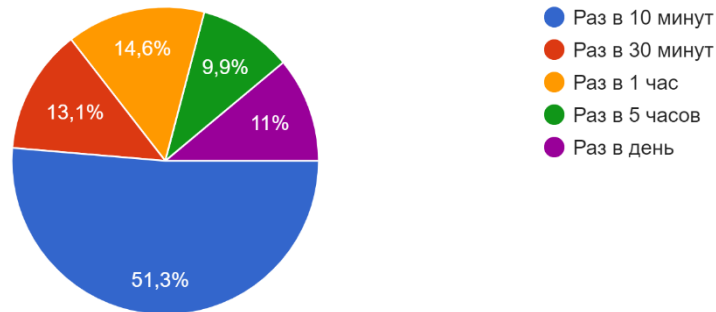
349 ответов



2)

Как часто Вы переходите из одного приложения в другое (речь идёт о вышеуказанных) ?

335 ответов



3)

Зачем Вы используете вышеперечисленные приложения?

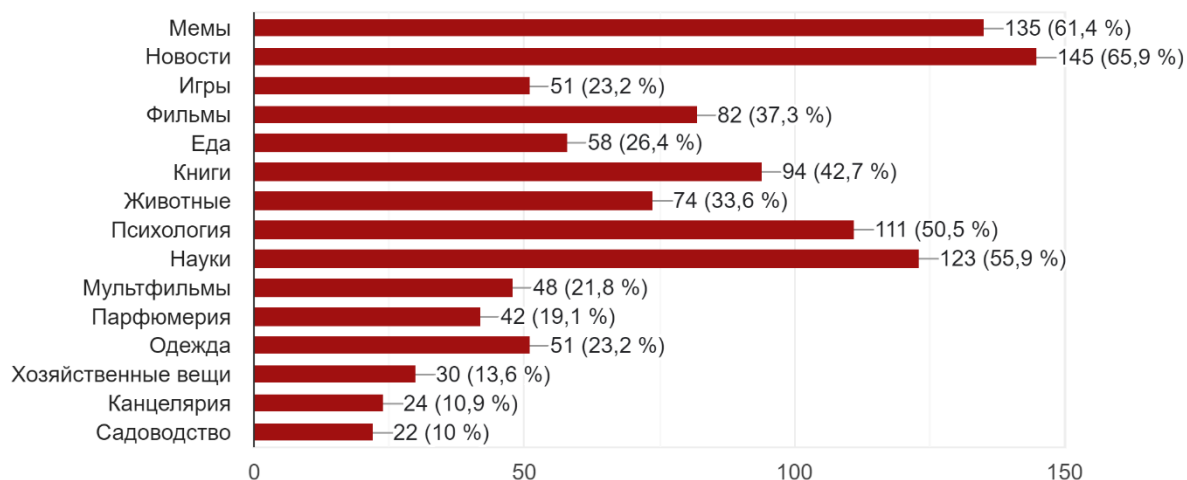
349 ответов



4)

Что Вы чаще всего смотрите в этих приложениях?

220 ответов



5)

Появилось уникальное приложение, способное объединить все 4 в одно...

349 ответов



По приведенным выше ответам на вопросы принимались важнейшие решения:

- из-за большого проведения людей за гаджетами (основываясь по первым трем вопросам) процента устройства может не хватать на весь

рабочий день человека, поэтому основные данные держать локально, а при выходе из приложения очищать кэш по максимуму насколько возможно;

- из-за меньшего числа пользователей “зарабатывающих” (исходя из третьего вопроса) на первый запуск приложения полную синхронизацию с аккаунтами пользователей убрать по причине не востребуемости и уменьшения скорости работы приложения;
- создать “базовые настройки”, опираясь на которые, будет составляться лента под пользователя (исходя из ответов на четвертый вопроса), не ориентируя первые показы контента на определенную тематику, из-за большого расхождения во мнениях.

Последний вопрос непохож на предыдущие, он ориентирован на полноценные запросы пользователей. Но, к сожалению, на него ответила лишь малая доля “испытываемых” ($\approx 11,2\%$), причем многие из них не несли особого смысла, но встречались и рассудительные ответы, на которые можно было опираться при разработке:

Я хотел(а) бы, чтобы в нём было...

39 ответов

Все интуитивно и просто

Было бы хорошо, если в одно приложение можно было бы синхронизировать только мои подписки и чаты, а не обширный поиск. Также можно включить опцию добавления только части из возможных приложений. (Чтобы если бесит тикток, можно было выключить доступ к нему) Тогда я бы скачала.

были только те каналы, на которые я подписан. Чтобы не лезло миллион других на тематику, которая мне интересна.

Больше добра

Все хорошо

Возможность вести переписку в разных месенджерах и возможность просматривать контент развлекательного характера

Версия для iOS

Голосование ещё продолжается, оно доступно по ссылке:

<https://forms.gle/9ABbFeTV76ecLu199>

Основная часть

Парсинг — это автоматизированный сбор и структурирование информации с сайтов при помощи программы или сервиса. Эта программа называется парсер и её задачей является сбор информации в соответствии с заданными параметрами.



Существуют два основных способа парсинга, отличающиеся только способом получения информации:

- Парсинг “в лоб”. Парсер, грубо говоря “вручную” получает доступ к содержимому веб-страницы или веб приложения и производит поиск информации.
У данного способа существует ряд проблем, самая главная из которых – это проблема пагинации (Paging). Чтобы полностью запарсить информацию с веб-страницы необходимо, чтобы она полностью была загружена. Большинство современных сайтов прогружаются в процессе того, как пользователь просматривает их. Поэтому полностью запарсить такой источник – трудная задача. Существуют инструменты, которые предоставляют решение данной проблемы, например Selenium для Python. Он позволяет полностью прогружать страницу, но данных процесс относительно долгий и трудоемкий.
- Парсинг с использованием API, способ, больше всего подходящий для приложения Memwor. API (Application programming interface) – это инструменты, предусмотренные самими разработчиками веб-приложений, программный интерфейс приложения, через который

Метвог отправляет запросы на сервера веб-приложений и получать от них ответ, содержащий целевую информацию. Кроме того, это решает проблему пагинации, так как приложение обращается сразу к серверам веб-приложений, а не к html коду, использование API позволяет использовать сортировку постов от рекламы, их классификацию, а также указание точного количества интересующих постов.

Чтобы пользоваться API нужно использовать соответствующие каждой платформе команды и данные. Под данными подразумевается информация о сообществах.

Во всех используемых приложениях существует структурирование на каналы и сообщества. Чтобы получить контент с источника необходимо где-то хранить информацию о нем:

- Домейн сообщества (Краткая ссылка на сообщество) – ключевой параметр
- Имя сообщества
- Категория сообщества (характер контента)
- Платформа сообщества (VK, Telegram, Reddit)

Домейн – это ключевой параметр при формировании запроса к API

Данная информация хранится в Firebase (Realtime database).

Структура базы – древовидная.



Запросы формируются с помощью сторонних библиотек Retrofit и OkHttpClient.

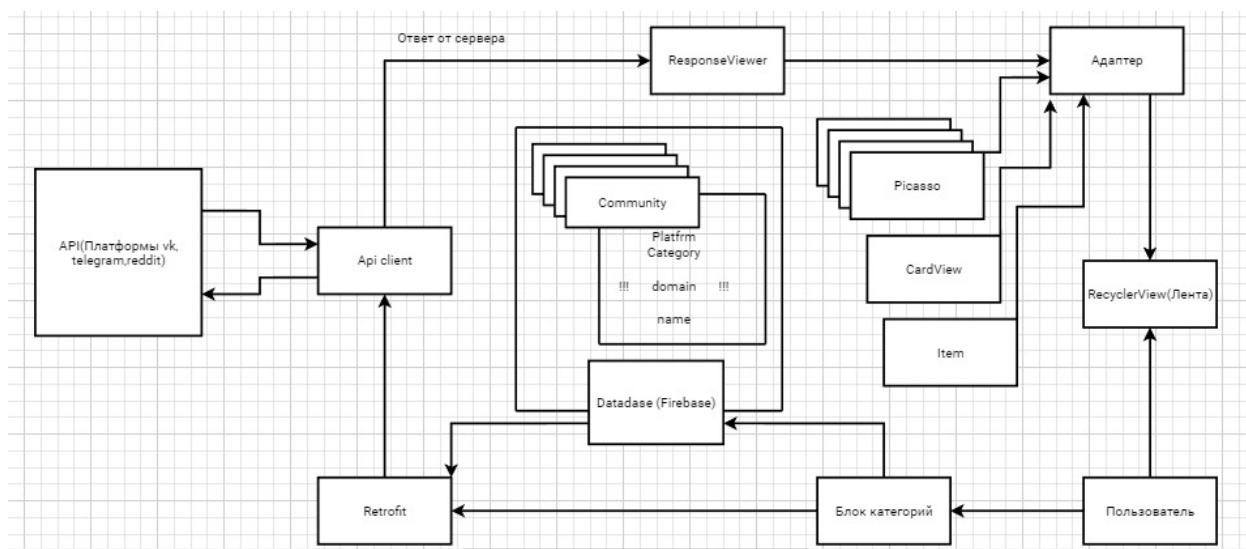


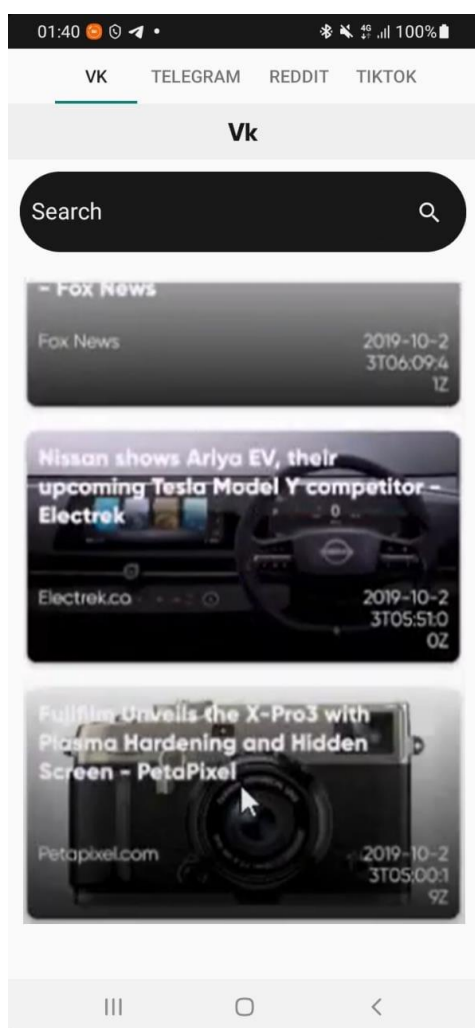
Схема взаимодействия основных компонентов приложения.

Пользователь при использовании приложения выбирает интересующие его категории. Осуществляется запрос в базу данных.

На основе этих категорий происходит поиск в базе данных, и возвращаются домейны (краткие ссылки) сообществ, по которым формируются запросы. Далее запросы отправляются на сервера приложений. От серверов приложение получает ответ. Полученный ответ перенаправляется в ResponseViewer, где происходит подготовка для его отображения в ленту. После этого подготовленные данные перенаправляются в Adapter

Получив необходимые данные, Адаптер прикрепляет эти данные в заранее описанное тело поста, которое также подгружает картинки из интернета с помощью сторонней библиотеки Picasso. После этих операций пост с помощью адаптера передаётся в ленту, где происходит отображение. (Саша)

На данный момент в приложении реализован метод парсинга для платформы VK, сделан подбор информации и составление ленты на основе категорий для платформы VK и разработана общая методика работы с веб-приложениями, поэтому сейчас остается лишь реализовать выбранный нами метод парсинга для остальных платформ и наполнить базу большим количеством источников.



- скриншот работы приложения

Список литературы

- Видеоматериалы с платформы YouTube: каналы Neco Ru, Mobile Developer и др.;
- Официальные документации по: среде разработки Android Studio, языкам программирования Kotlin и Java, базе данных от Firebase, инструментам платформ (VK API, Reddit API, Telegram API);
- Вспомогательные источники: Habr, Skillbox, Geeksforgeeks, StackOverflow;
[Kotlin Build a full-stack web app with Kotlin Multiplatform](#),
[Хабр Пишем простой RESTful сервис на kotlin](#),
[Солонько М.К. - Язык программирования Kotlin](#),
[Хабр Kotlin vs Java](#),
[Checkroi Java vs Kotlin](#),
[skillbox.ru Java или Kotlin: что выбрать начинающему андроид-разработчику](#),
[VK для разработчиков - API](#),
[reddit Документация API](#),
[Core.telegram - api](#),