Министерство науки и высшего образования Российской Федерации

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»

ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНЫХ КИБЕРНЕТИЧЕСКИХ СИСТЕМ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

На правах рукописи

УДК 004.75

БУЛЬДИН ИЛЬЯ ДМИТРИЕВИЧ

СОЗДАНИЕ МОБИЛЬНОГО ПРИЛОЖЕНИЯ SIP-ТЕЛЕФОНИИ НИЯУ МИФИ

Выпускная квалификационная работа бакалавра

Направление подготовки 09.03.01 Информатика и вычислительная техника

|  |
| --- |
| Выпускная квалификационная работа защищена  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |
| Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Секретарь ГЭК \_\_\_\_\_\_\_\_\_\_\_\_\_ |

г. Москва

2023

Студент-дипломник \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Бульдин И.Д. /

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Романов Н.Н. /

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Овчаренко Е.С. /

Заведующий кафедрой №12 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ / Иванов М.А. /

**АННОТАЦИЯ**

СОДЕРЖАНИЕ

[ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 5](#_Toc133762226)

[ВВЕДЕНИЕ 6](#_Toc133762227)

[1 ОБЗОРНАЯ ЧАСТЬ 8](#_Toc133762228)

[1.1 Введение 8](#_Toc133762229)

[1.2 Описание предметной области 8](#_Toc133762230)

[**1.2.1 Естественно-языковая модель предметной области** 8](#_Toc133762231)

[**1.2.2 Графическая модель предметной области** 9](#_Toc133762232)

[1.3 Анализ существующего ПО 10](#_Toc133762233)

[**1.3.1 Grandstream Wave Lite** 10](#_Toc133762234)

[**1.3.2 Zoiper Lite** 11](#_Toc133762235)

[**1.3.3 PortSIP Softphone** 12](#_Toc133762236)

[**1.3.4 VoIP MEPhI для Android** 13](#_Toc133762237)

[**1.3.4 Сравнение существующего ПО** 14](#_Toc133762238)

[**1.3.5 Вывод** 18](#_Toc133762239)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 19](#_Toc133762240)

# **ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ**

SIP (Session Initiation Protocol) – протокол прикладного уровня, который позволяет устанавливать, изменять, завершать мультимедиа сеансы, такие как звонки интернет телефонии.

ПО – Программное Обеспечение.

ТСОП – Телефонная Сеть Общего Пользования.

Софтфон (softphone) – ПО для совершения звонков через Интернет.

IETF ­(Internet Engineering Task Force) – организация, которая создает стандарты, которые часто принимаются пользователями Интернета, сетевыми операторами и поставщиками оборудования.

OS (Operating System) – операционная система.

UX (User Experience) дизайн – процесс определения опыта, который будет испытывать пользователь при взаимодействии с продуктом.

UI (User Interface) дизайн – проектирование пользовательского интерфейса.

VoIP (Voice over IP) – также называемая IP-телефония, это передача голоса и мультимедийного контента через интернет-соединение.

HIG (Human Interface Guidelines) – документ, содержащий рекомендации для разработчиков UI программного обеспечения в рамках некоторой OS или платформы.

# **ВВЕДЕНИЕ**

В наше время мобильные телефоны стали неотъемлемой частью нашей жизни, и возможность голосового общения с помощью интернета стала доступна для большинства людей. IP-телефония

Актуальной научной задачей является изучение существующих решений в области создания мобильных приложений SIP-телефонии, анализ их функциональности и разработка мобильного приложения для операционной системы iOS, которое будет удовлетворять требованиям современной мобильной разработки, учитывая специфику использования внутри организации-заказчика НИЯУ МИФИ.

Актуальность работы подтверждается статистикой использования интернет-телефонии. Согласно финансовой отчетности крупнейшей в России социальной сети ВКонтакте по итогам четвертого квартала 2021 года [1] ежемесячная аудитория сервиса «VK Звонки» составляет 20 млн пользователей.

По статистике [2] сотрудники проводят в разговорах 1.5-2 часа. Более того, в связи с пандемией COVID-19 количество сотрудников, работающих дистанционно, по оценке Минтруда [3] насчитывает миллионы. В связи с этим компании вынуждены предоставлять сотрудникам удобные и качественные решения для голосовой связи, учитывающие особенности использования в организации и требующие только стабильного Интернет-соединения.

Целью работы является снижение количества пользователей стационарными телефонными аппаратами для подключения к ТСОП и информатизация процессов связи между сотрудниками НИЯУ МИФИ для повышения мобильности. Достижение цели осуществимо путем разработки собственного ПО – мобильного приложения SIP-телефонии, которое будет обладать необходимым функционалом, удобным интерфейсом, высокой степенью надежности и предоставлять сотрудникам удобный инструмент для общения с помощью SIP-телефонии.

Для достижения поставленной цели необходимо решить следующие задачи:

* Изучение существующих решений в области мобильных приложений SIP-телефонии, анализ их функциональности и возможностей;
* Выбор технологий и инструментов, необходимых для использования в разработке;
* Формулирование требований к разрабатываемому ПО;
* Проектирование UX и UI дизайна системы;
* Разработка архитектуры и интерфейса мобильного приложения с учитом требований современной SIP-телефонии и особенностей использования на мобильных устройствах с операционной системой iOS;
* Реализация бизнес-логики мобильного приложения, обеспечивающей выполнение основных сформулированных требований;
* Тестирование разработанного ПО (локальное и с помощью сервиса бета-тестирования iOS приложений TestFlight).

# **1 ОБЗОРНАЯ ЧАСТЬ**

## **1.1 Введение**

В данной главе представлены исследование и анализ предметной области, описание существующих решений.

## **1.2 Описание предметной области**

### **1.2.1 Естественно-языковая модель предметной области**

Предметной областью является установление голосовой связи посредством SIP-телефонии (пункт 1.3.) между сотрудниками НИЯУ МИФИ.

На сегодняшний день абонент может использовать один из представленных методов связи: традиционный стационарный ТСОП-телефон, софтфоны с ПО МИФИ (только для операционной системы Android) и со сторонним ПО.

Для совершения звонка пользователю необходимо быть авторизованным в системе. Регистрация нового абонента происходит по заявке на электронную почту службы IP-телефонии управления информатизации университета.

После успешного добавления нового сотрудника в систему, необходимо произвести первоначальные настройки средства связи. Для подключения ТСОП-телефона нужно вызвать соответствующего специалиста. Софтфон со сторонним ПО может быть подключен самостоятельно, следуя подробной инструкции, полученной в мессенджере или по электронной почте. Для настройки софтфона с ПО МИФИ необходимо заполнить имя учетной записи пользователя и пароль в соответствующих полях ввода.

Авторизованному пользователю необходим добавочный номер вызываемого абонента, который доступен в открытом доступе на сайте службы IP-телефонии. Пользователи софтфона с ПО МИФИ могут найти интересующий номер непосредственно в каталоге пользователей внутри приложения.

### **1.2.2 Графическая модель предметной области**

Графическая модель предметной области представлена на рисунке 1.2.2.1.

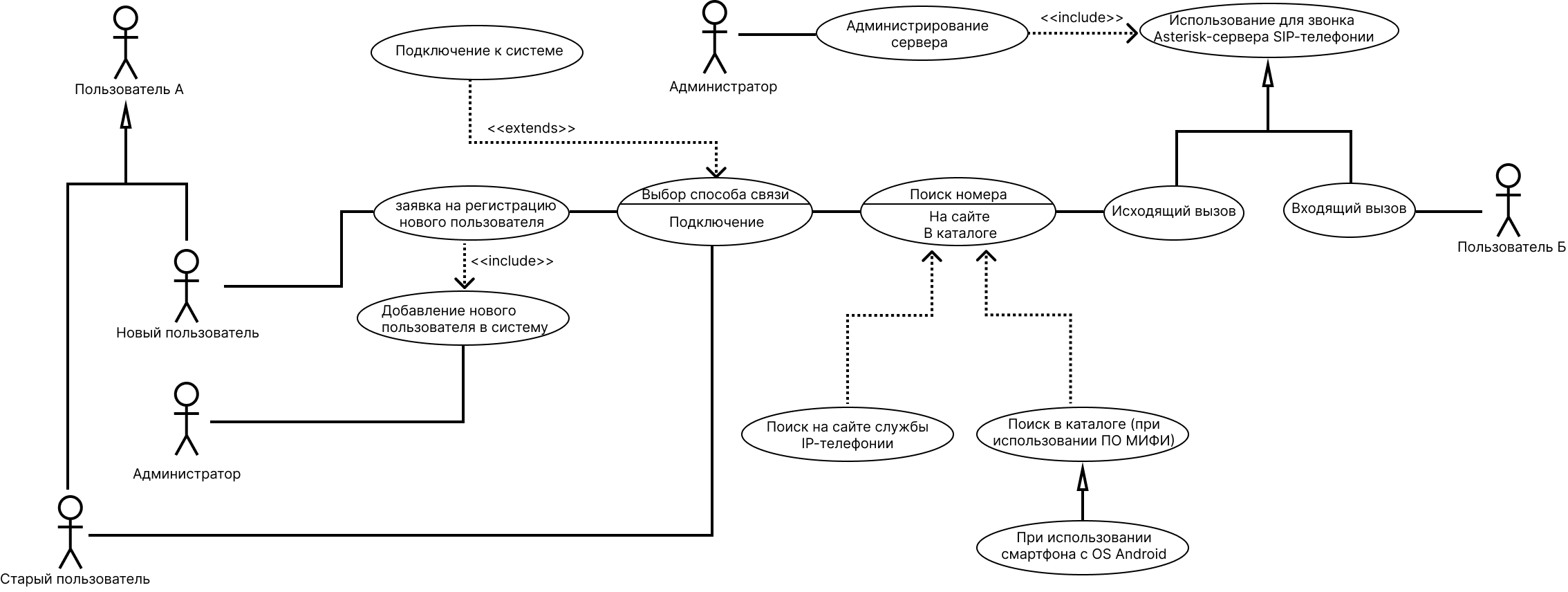


Рисунок 1.2.2.1 – Графическая модель предметной области

## **1.3 SIP-телефония**

Телефония с использованием Session Initial Protocol (SIP) – это метод совершения и приема телефонных звонков через Интернет с использованием широкополосного соединения и учетной записи SIP.

SIP – это стандартизированный протокол, используемый для инициирования, поддержания, изменения и завершения сеансов связи в реальном времени, включающих видео, голос, обмен сообщениями и другие виды связи между устройствами. SIP обычно используется в сочетании с другими протоколами, такими как RTP (Real-Time Transport Protocol) и RTCP (Real-Time Transport Control Protocol).

SIP является открытым стандартом и определен IETF. Протокол описан в RFC 3261 [4], который определяет основные характеристики протокола и форматы сообщений, используемые для связи.

SIP использует сообщения, известные как SIP-requests и SIP-responses, для инициирования и управления сеансами связи. Эти сообщения отправляются между SIP-клиентами, которые представляют собой устройства или программы, использующие SIP, и SIP-серверами, которые представляют собой серверы, маршрутизирующие SIP-запросы. SIP часто используется в системах VoIP (Voice Over IP) для установления и управления голосовыми и видеозвонками через Интернет.

Одним из основных преимуществ SIP-телефонии является то, что протокол позволяет пользователям совершать и принимать звонки из любого места, где есть широкополосное соединение. Это может быть особенно полезно для предприятий и организаций, сотрудники которых работают удаленно или в разных местах.

Существует множество различных приложений для SIP-софтфонов и аппаратных устройств, которые позволяют пользователям использовать преимущества SIP-телефонии. Далее в пункте 1.4 проведен анализ конкурентов разрабатываемой системы.

## **1.4 Анализ существующего ПО**

Рассмотрим перечень решений, предоставляющих ПО для SIP-телефонии.

### **1.4.1 Grandstream Wave Lite**

Прямым конкурентом является решение от компании Grandstream – GS Wave Lite. Это мобильное приложение, которое позволяет пользователям совершать и принимать звонки, используя широкополосное соединение и учетную запись SIP.

Одно из преимуществ Wave Lite заключается в том, что оно имеет широкий функционал и большое количество пользовательских настроек. Приложение включает такие функции, как переадресация вызовов, голосовая почта и текстовые сообщения. На рисунке 1.4.1.1 представлен интерфейс приложения.

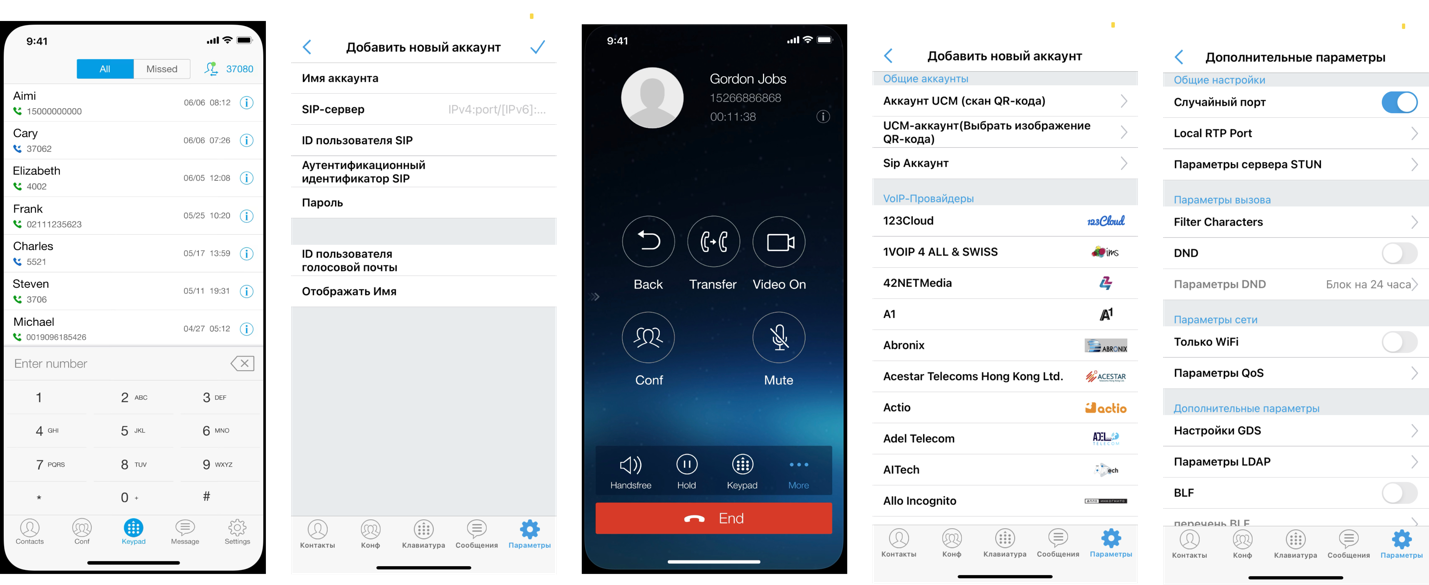


Рисунок 1.4.1.1 – Интерфейс мобильного приложения GS Wave Lite

Анализ отзывов на страницах приложения в AppStore и Google Play [5, 6] позволяет выявить основные недостатки данного решения:

* Высокое энергопотребление;
* Отсутствие возможности принимать входящие звонки, если приложение находится в состоянии жизненного цикла [7] «Not Running» для iOS;
* Прерывания связи;
* Зависания приложения;
* Проблемы с работой микрофона на некоторых версиях iOS;
* Поддерживает до 6-ти SIP-аккаунтов.

### **1.4.2 Zoiper Lite**

Zoiper Lite – еще одно решение, активно используемое на рынке. Приложение имеет несколько особенностей и функциональных возможностей, которые включают поддержку различных VoIP протоколов, что позволяет подключаться к различным сервисам и совершать звонки по предпочитаемому протоколу.

Скачивание приложения, а также отзывы на официальной странице магазина в AppStore и Google Play [8, 9] позволяют выделить основные достоинства и недостатки.

На рисунке 1.4.2.1 представлен интерфейс приложения.

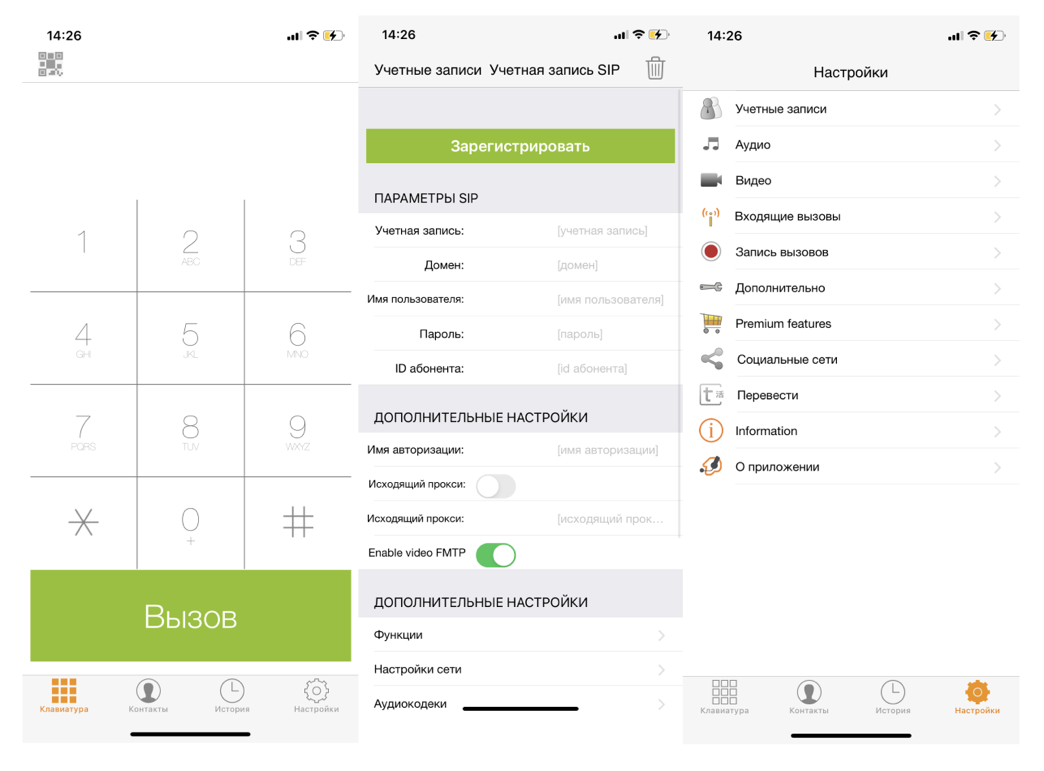


Рисунок 1.4.2.1 – Интерфейс мобильного приложения Zoiper Lite

Достоинства:

* Расширенные возможности, такие как запись звонков, конференцсвязь и голосовая почта;
* Оптимизация и низкое энергопотребление;
* Поддерживает несколько учетных записей SIP без ограничений в количестве.

Недостатки:

* Пользовательский интерфейс, не соответствующий HIG;
* Сложный UX;
* Проблемы с авторизацией пользователя;
* Отсутствие возможности принимать входящие звонки, если приложение находится в состоянии жизненного цикла «Not Running» для iOS.

### **1.4.3 PortSIP Softphone**

PortSIP Softphone - это мобильное приложение, предназначенное для совершения звонков по VoIP с помощью смартфона или планшета. Приложение имеет множество функций и возможностей, включая поддержку различных протоколов VoIP, шифрование вызовов и передовые аудиокодеки.

Скачивание приложения, а также отзывы на официальной странице магазина в AppStore и Google Play [10, 11] позволяют выделить основные достоинства и недостатки.

На рисунке 1.4.3.1 представлен интерфейс приложения.

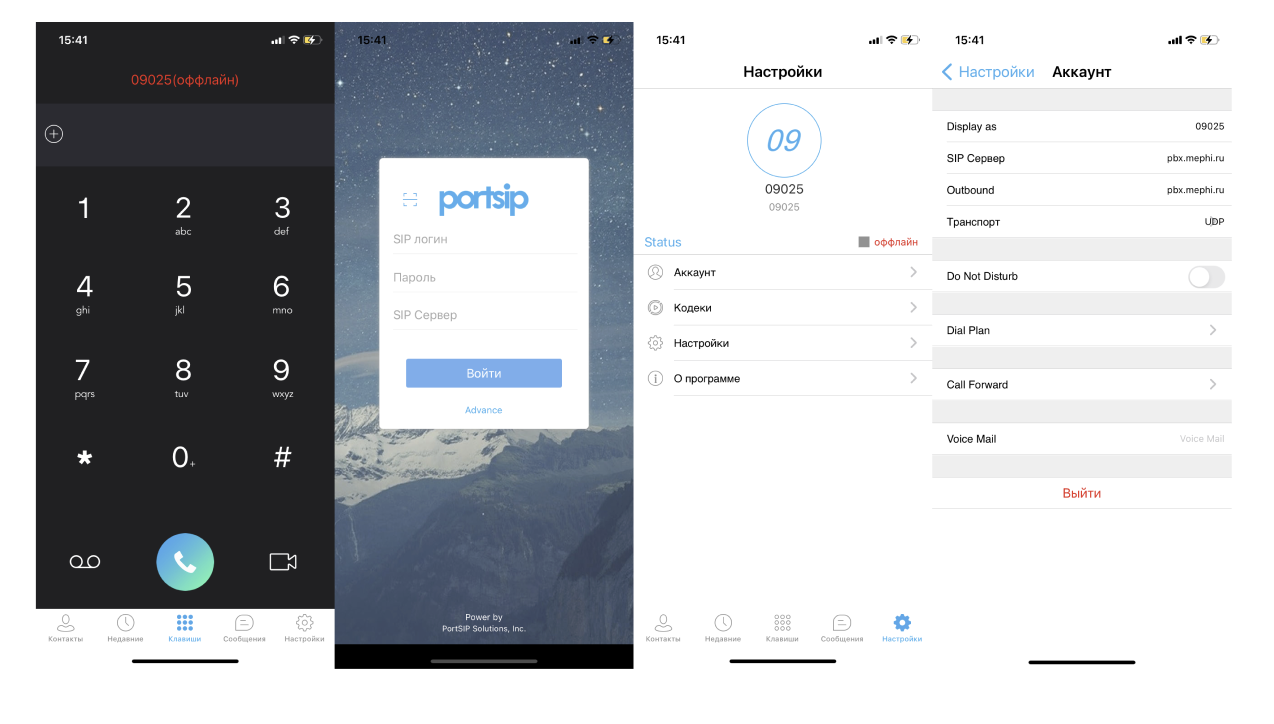


Рисунок 1.4.3.1 – Интерфейс мобильного приложения PortSIP Softphone

Достоинства:

* Поддерживает множество различных VoIP-протоколов, таких как SIP, XMPP и WebRTC. Это позволяет пользователям подключаться к различным сервисам и совершать звонки по предпочитаемому протоколу;
* Поддерживает несколько учетных записей SIP без ограничений в количестве;
* Оснащен расширенными функциями, такими как голосовая почта и отправка текстовых сообщений.

Недостатки:

* Проблемы с авторизацией пользователя;
* Отсутствие возможности принимать входящие звонки, если приложение находится в состоянии жизненного цикла «Not Running» для iOS;
* Назойливые звонки с неизвестных номеров;
* Отсутствие оптимизации, высокое энергопотребление.

### **1.4.4 VoIP MEPhI для Android**

НИЯУ МИФИ предоставляет сотрудникам собственное решение – VoIP MEPhI. Приложение доступно только пользователям мобильных телефонов с OS Android. Приложение имеет специализированные уникальные функции, дающие дополнительную гибкость при работе с IP-телефонией. Так, например, в ПО доступна функция поиска сотрудников организации в каталоге, избранные контакты и упрощенная система авторизации.

На рисунке 1.4.4.1 представлен интерфейс приложения.

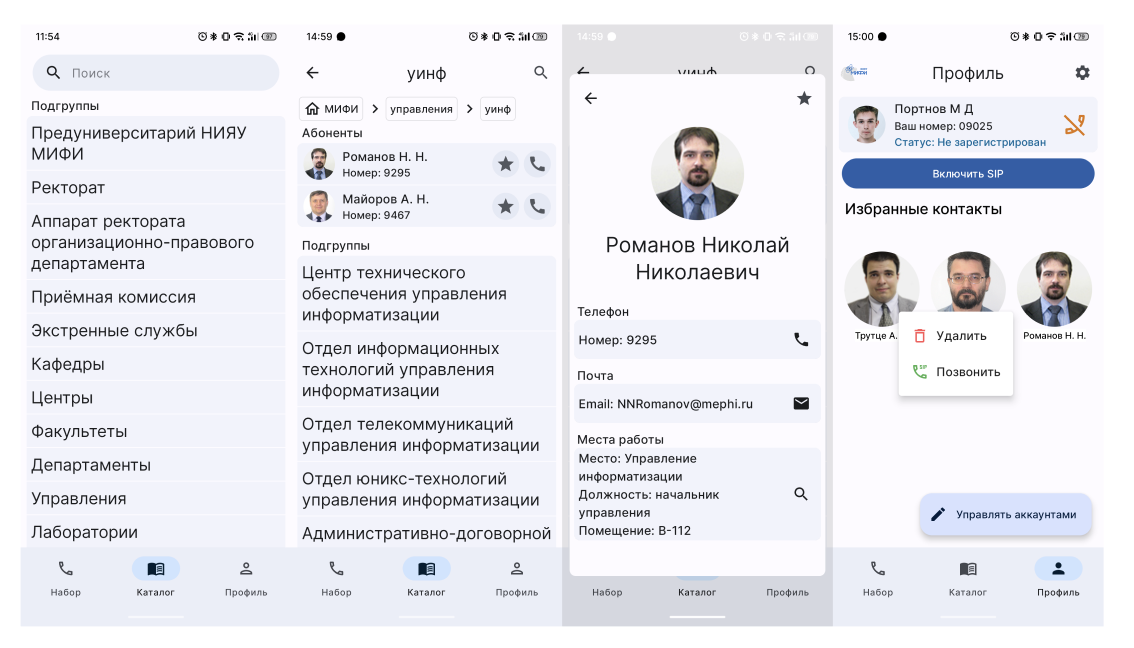


Рисунок 1.4.4.1 – Интерфейс мобильного приложения VoIP MEPhI

Скачивание приложения с официальной страницы в Google Play [12] позволяет выделить основные достоинства и недостатки.

Достоинства:

* Наличие дополнительной функциональности, нацеленной на удобство сотрудников университета;
* Удобный UX и UI, соответствующий HIG;
* Низкое энергопотребление;
* Поддерживает несколько учетных записей SIP без ограничений в количестве;
* Возможность доработки и расширяемости функционала в соответствии с запросами пользователей;
* Обеспечение высокого уровня безопасности.

Недостатки:

* Небольшие недоработки в UX;
* Отсутствие большого количества функционала, таких как текстовые сообщения, запись звонков и т.д.;
* Приложения распространяется только для пользователей устройств с OS Android.

### **1.4.4 Сравнение существующего ПО**

Для сравнения ПО применяется метод упрощенной аналитической иерархии [13]. Выберем критерии и ранжируем их по убыванию степени важности (Таблица 1.4.4.1).

Таблица 1.4.4.1 – Критерии сравнения, упорядоченные по убыванию важности

|  |  |
| --- | --- |
| **Критерий** | **Обозначение** |
| Бесперебойная работа с SIP-аккаунтами | K1 |
| UX и UI системы | K2 |
| Степень интеграции с организацией | K3 |
| Возможность принимать звонки в состоянии жизненного цикла приложения «Not Running» | K4 |
| Низкое энергопотребление | K5 |

Вербально-числовая шкала превосходств представлена в Таблице 1.4.4.2.

Таблица 1.4.4.2 – Вербально-числовая шкала превосходств

|  |  |
| --- | --- |
| **Вербальное обозначение** | **Числовое значение** |
| Важность равна | 1 |
| Умеренно превосходит | 3 |
| Значительно превосходит | 5 |
| Сильно превосходит | 7 |
| Очень сильно превосходит | 9 |

При этом будем использовать промежуточные четные значения. Например, 2 – оценка находится между умеренным превосходством и равной важностью.

Сформируем матрицу A размером n х n попарного сравнения представленных критериев, где n – количество критериев (Таблица 1.4.4.3). В матрице приводится расчет среднего геометрического строки матрицы и компонентов нормализированного вектора приоритетов (НВП).

Среднее геометрическое строки оценивается по формуле (1.4.4.1):

(1.4.4.1)

Компонент НВП оценивается по формуле (1.3.4.2):

(1.4.4.2)

Таблица 1.4.4.3 – Матрица попарного сравнения критериев

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **K1** | **K2** | **K3** | **K4** | **K5** | **Среднее геометрическое строки, αi** | **Компонент НВП, ci** |
| **K1** | 1 | 2 | 3 | 5 | 7 | = 2,914 |  |
| **K2** |  | 1 | 1 | 4 | 9 | = 1,783 |  |
| **K3** |  | 1 | 1 | 7 | 7 | = 1,748 |  |
| **K4** |  | 1/4 | 1/7 | 1 | 3 | =0,464 |  |
| **K5** |  | 1/9 | 1/7 | 1/3 | 1 | =0,238 |  |

Собственное значение матрицы оценивается по формуле (1.4.4.3):

(1.4.4.3)

Рассчитаем собственное значение матрицы:

Показатель случайной согласованности R определяется теоретически и зависит от размера матрицы.

R = 1,12

Отношение согласованности F оценивается по формуле (1.4.4.4):

(1.4.4.4)

Рассчитаем отношение согласованности:

Т.к. F <0.1, то матрица является согласованной.

В таблице 1.4.4.4 приведено соответствие существующего ПО представленным критериям с использованием качественной характеристики.

Таблица 1.4.4.4 – Соответствие ПО критериям (качественная характеристика)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **GS Wave Lite** | **Zoiper Lite** | **PortSIP Softphone** | **VoIP MEPhI** |
| **K1** | Хорошо | Удовлетворительно | Удовлетворительно | Отлично |
| **K2** | Плохо | Очень плохо | Очень хорошо | Очень хорошо |
| **K3** | Очень плохо | Очень плохо | Очень плохо | Отлично |
| **K4** | Очень плохо | Очень плохо | Очень плохо | Очень плохо |
| **K5** | Плохо | Очень хорошо | Плохо | Очень хорошо |

Шкала перевода качественных характеристик в количественные представлена в таблице 1.4.4.5:

Таблица 1.4.4.5 – Соответствие качественных и количественных характеристик

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Качественная характеристика** | **Очень плохо** | **Плохо** | **Удовлетворительно** | **Хорошо** | **Очень хорошо** | **Отлично** |
| Количественная характеристика | 1 | 3 | 5 | 7 | 9 | 11 |

В таблице 1.4.4.6 приведено соответствие существующего ПО представленным критериям с использованием количественной характеристики.

Таблица 1.4.4.6 – Соответствие ПО критериям (количественная характеристика)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **GS Wave Lite** | **Zoiper Lite** | **PortSIP Softphone** | **VoIP MEPhI** |
| **K1** | 7 | 5 | 5 | 11 |
| **K2** | 3 | 1 | 1 | 9 |
| **K3** | 1 | 1 | 1 | 11 |
| **K4** | 1 | 1 | 1 | 1 |
| **K5** | 3 | 9 | 3 | 9 |

В таблице 1.4.4.7 приведено соответствие существующего ПО представленным критериям с использованием количественной характеристики с учетом нормирования.

Нормированная оценка критерия оценивается по формуле (1.4.4.5):

(1.4.4.5)

Таблица 1.4.4.7 – Соответствие ПО критериям (количественная характеристика с ученом нормирования)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **GS Wave Lite** | **Zoiper Lite** | **PortSIP Softphone** | **VoIP MEPhI** |
| **K1** | 0.408 | 2.856 | 2.04 | 2.04 | 4.488 |
| **K2** | 0.249 | 0.747 | 0.249 | 0.249 | 2.241 |
| **K3** | 0.245 | 0.245 | 0.245 | 0.245 | 2.695 |
| **K4** | 0.065 | 0.065 | 0.065 | 0.065 | 0.065 |
| **K5** | 0.033 | 0.099 | 0.297 | 0.099 | 0.297 |
|  | – | 4.012 | 3.589 | 2.698 | 9.786 |

### **1.4.5 Вывод**

По результатам сравнения существующего ПО видно, что VoIP MEPhI для OS Android имеет наивысший балл, соответственно разработка собственного мобильного приложения SIP-телефонии для iOS обоснована. Это позволит адаптировать приложение к своим конкретным потребностям, обеспечить его хорошую интеграцию и предложить дополнительные функции и возможности, которых нет в готовых SIP-софтфонах.

## **1.5 Выбор языка программирования для разработки**

Язык программирования является важным инструментом для разработки ПО, в том числе и мобильных приложений. При разработке iOS приложений существует несколько языков, которые могут быть использованы. В данной главе рассмотрены основные языки, которые можно использовать для разработки iOS приложений, их особенности, возможности и ограничения. Также приведено сравнение этих языков, обоснован выбор наиболее подходящего.

Для разработки iOS приложений можно использовать следующие языки программирования:

* Swift – язык программирования, созданный Apple для разработки iOS, macOS, watchOS и tvOS. Он является официальным языком разработки приложений для Apple и предлагает множество удобных функций и возможностей для разработки;
* Objective-C – язык программирования, который был ранее использован как основной язык разработки приложений для iOS, но сейчас уступает место Swift. Objective-C является объектно-ориентированным языком и использует динамическую типизацию;
* Kotlin/Native – язык программирования, разработанный компанией JetBrains, который позволяет написать кросс-платформенное приложение одной кодовой базой и скомпилировать его в нативный код для iOS;
* C++ – мощный язык программирования, который может использоваться для разработки iOS приложений с помощью таких инструментов как как Clang и LLVM. Однако, так как C++ является низкоуровневым языком, его использование требует большего опыта и знаний, чем более высокоуровневые языки, такие как Swift или Objective-C. В дополнение к этому, C++ поддерживается Apple неофициально, что может привести к нестабильной работе и проблемам с компиляцией;
* Другие языки, такие как Java и Python, не являются официально поддерживаемыми Apple и могут быть очень сложными для использования

Разработка кросс-платформенного приложения не рассматривается организацией. В целом, для нативной разработки iOS приложений рекомендуется использовать Swift или Objective-C.

Swift является более современным и удобным в использовании языком программирования, чем Objective-C. Он имеет синтаксис с высокой читабельностью, что важно для дальнейшей поддержки ПО, использует статическую типизацию, безопасен и предлагает больше современных функций и возможностей, чем Objective-C. Кроме того, Swift быстрее и эффективнее, чем Objective-C, благодаря использованию оптимизированных структур данных и алгоритмов. Поэтому для разработки мобильного приложения SIP-телефонии НИЯУ МИФИ был выбран язык программирования Swift.

## **1.6 Императивный и декларативный подход в разработке**

Императивное и декларативное программирование – это две противоположные парадигмы. Императивное программирование делает акцент на командах или утверждениях, которые описывают, как должна выполняться программа, в то время как декларативное программирование фокусируется на конечном результате или итогах программы, по словам Lloyd J,, "декларативное программирование подразумевает указание того, что должно быть вычислено, но не обязательно как это должно быть вычислено" [14].

UIKit и SwiftUI – это два разных фреймворка, используемых для создания пользовательского интерфейса в iOS-приложениях.

UIKit является стандартным фреймворком для разработки пользовательского интерфейса в iOS с момента выпуска первой версии iOS. Он предоставляет большое количество готовых элементов интерфейса, таких как кнопки, таблицы, коллекции и т.д., а также множество инструментов для управления их взаимодействием и анимацией. Однако UIKit требует большого количества кода для реализации даже простых интерфейсов, что может усложнить разработку и увеличить вероятность ошибок.

SwiftUI – это новый фреймворк, введенный Apple в 2019 году. Он предназначен для создания пользовательского интерфейса с помощью декларативного языка описания, который позволяет описывать интерфейс в терминах его состояний и поведения, а не в терминах низкоуровневых действий, которые нужно выполнить. Это позволяет создавать интерфейсы быстрее и проще, так как не нужно писать много кода для реализации отдельных элементов интерфейса.

В целом, оба фреймворка могут быть использованы для создания пользовательского интерфейса в iOS-приложениях, SwiftUI является более современным и удобным в использовании из-за его декларативного подхода и автоматической обработки изменений состояния. Однако предпочтение при разработке было отдано UIKit.

Далее описаны причины выбора императивного подхода для создания интерфейса iOS приложения:

* UIKit начал использоваться в iOS гораздо раньше, чем SwiftUI, поэтому он более широко распространен [15]. На рисунке 1.6.1 представлен график использования фреймворков для различных iOS. Из графика видно, что прослеживается тенденция увеличения использования SwiftUI, однако стоит учесть, что поддержка только последних версий iOS не выгодна для организации;
* UIKit предоставляет больше возможностей для настройки и контроля поведения элементов интерфейса. Это может быть полезно при создании таблиц с динамической высотой ячеек в зависимости от количества текста;
* UIKit предоставляет больше функциональности и возможностей, чем SwiftUI, включая различные элементы интерфейса, анимации и инструменты управления контентом. Это может быть полезно при создании сложных интерфейсов с множеством индивидуальных элементов.

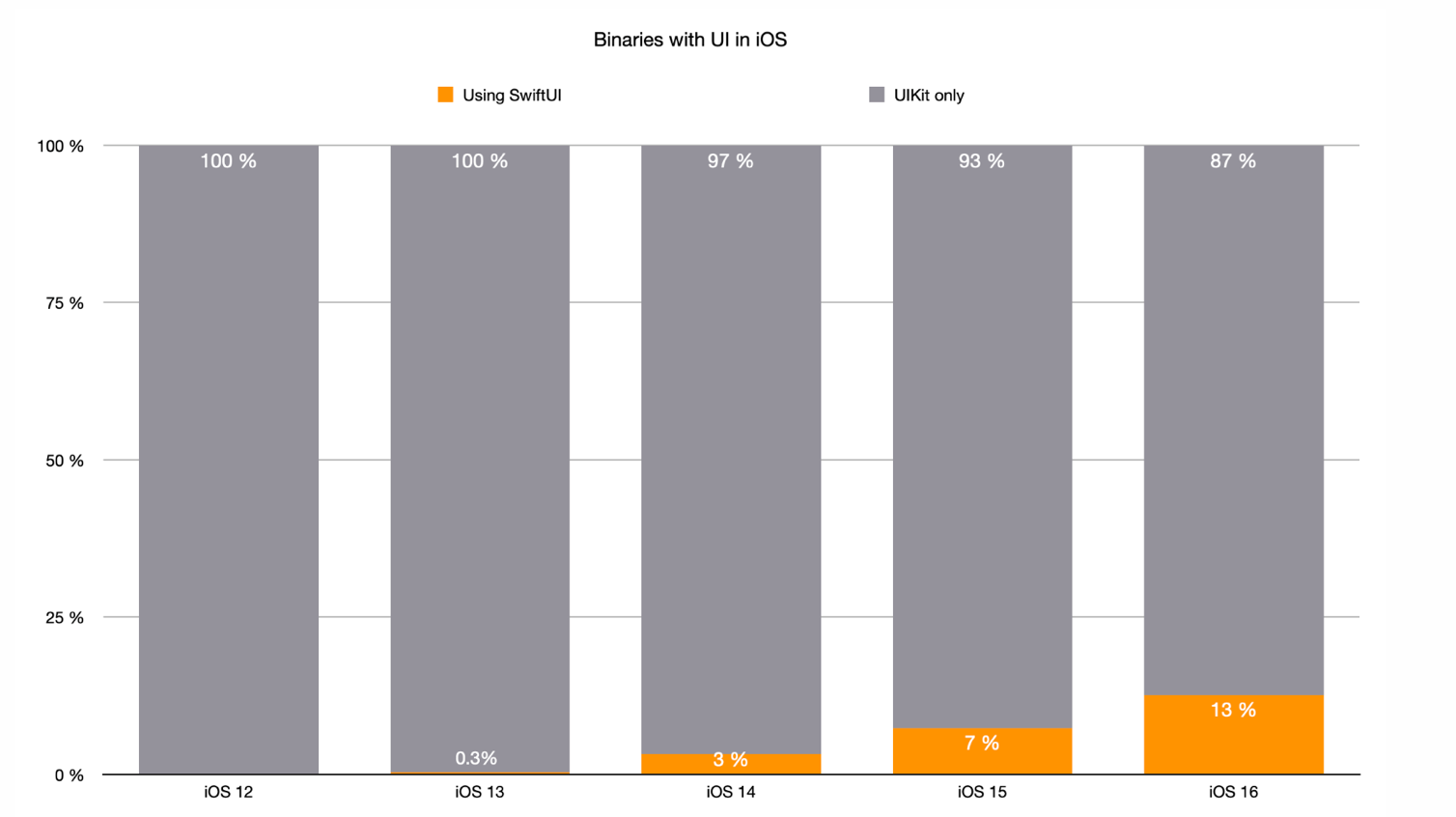


Рисунок 1.6.1 – Использование UIKit и SwiftUI для различных версий iOS

## **1.7 Выбор архитектурного паттерна**

Выбор архитектурного паттерна – важный шаг в разработке приложения для iOS. Архитектурный паттерн определяет структуру приложения и способ взаимодействия его различных частей. Правильный выбор архитектурного паттерна может упростить разработку, улучшить поддержку кода и повысить гибкость программного кода.

Существует несколько популярных архитектурных паттернов, которые можно использовать в приложениях для iOS, таких как MVC (Model-View-Controller), MVP (Model-View-Presenter), MVVM (Model-View-ViewModel) и VIPER. Каждый из этих паттернов имеет свои сильные и слабые стороны, и выбор паттерна зависит от конкретных требований и целей приложения.

Проведено сравнение различных архитектурных паттернов, их ключевые характеристики, чтобы повысить масштабируемость, поддерживаемость и тестируемость программного кода. Разница между MVC, MVP и MVVM несущественна. Controller «знает» и о модели, и о слое представления. Presenter «знает» о модели и зависит от слоя представления. ViewModel похожа на Presenter, но она отвязана от пользовательского интерфейса и слоя данных, потому что преобразует данные, полученные из модели в формат, удобный для слоя представления.

Использование архитектурного паттерна MVС для приложения SIP-телефонии может быть хорошим выбором по нескольким причинам:

* Понятное и удобное разделение логики на изолированные компоненты. Это облегчает внесение изменений и поддержку приложения.
* Возможность переиспользования кода;
* Тестируемость. Разделение задач в MVC облегчает независимое тестирование различных частей приложения, что помогает обеспечить надежность и отсутствие ошибок в приложении;
* Поддерживаемость. Модель MVC облегчает обслуживание приложения с течением времени, поскольку изменения в модели или контроллере не требуют изменений в слое представления, и наоборот

В целом, архитектурный паттерн MVC может помочь сделать разработку приложения SIP-телефонии более эффективной и удобной, что может быть особенно важно для приложения, которое должно быть надежным и отзывчивым. Кроме того, MVC – это широко используемый и устоявшийся архитектурный паттерн, который знаком многим разработчикам, что может облегчить поиск ресурсов и поддерживаемость кода.

# **СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Финансовая отчетность ВКонтакте [Электронный ресурс]. – URL: <https://vk.com/press/q4-2021-results>
2. Организация удаленной работы [Электронный ресурс]. – URL: <https://www.sipnet.ru/helpfull/remote-work>
3. Минтруд. Оценка сотрудников, работающих дистанционно [Электронный ресурс]. – URL: <https://www.interfax.ru/russia/757540>
4. RFC 3261 Session Initial Protocol [Электронный ресурс]. – URL: <https://datatracker.ietf.org/doc/html/rfc3261>
5. Страница Grandstream Wave Lite в AppStore [Электронный ресурс]. – URL: <https://apps.apple.com/ru/app/grandstream-wave-lite/id1029274043>
6. Страница Grandstream Wave Lite в Google Play [Электронный ресурс]. – URL:
7. Жизненный цикл iOS приложения [Электронный ресурс]. – URL: <https://manasaprema04.medium.com/application-life-cycle-in-ios-f7365d8c1636>
8. Страница Zoiper Lite в AppStore [Электронный ресурс]. – URL: <https://apps.apple.com/ru/app/zoiper-lite-voip-soft-phone/id438949960>
9. Страница Zoiper Lite в Google Play [Электронный ресурс]. – URL:
10. Страница PortSIP Softphone в AppStore [Электронный ресурс]. – URL: <https://apps.apple.com/ru/app/portsip-softphone/id426903818>
11. Страница PortSIP Softphone в Google Play [Электронный ресурс]. – URL:
12. Страница VoIP MEPhI в Google Play [Электронный ресурс]. – URL: <https://play.google.com/store/apps/details?id=ru.mephi.voip>
13. Волокобинский Михаил Юрьевич, Пекарская Ольга Анатольевна, Рази Даниил Александрович Принятие решений на основе метода анализа иерархий // Финансы: теория и практика. 2016. №2 (92). URL: <https://cyberleninka.ru/article/n/prinyatie-resheniy-na-osnove-metoda-analiza-ierarhiy-1>
14. Lloyd, J.: Practical advantages of declarative programming. In: Joint Conference on Declarative Programming, GULP-PRODE 1994 (1994)
15. Количество приложений, использующих SwiftUI [Электронный ресурс]. URL: – <https://blog.timac.org/2022/1005-state-of-swift-and-swiftui-ios16/>