МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Кафедра многопроцессорных систем и сетей

РЕАЛИЗАЦИЯ ПРИЛОЖЕНИЯ ПОД ПЛАТФОРМУ ANDROID ДЛЯ СИНХРОННОГО ПЕРЕВОДА РЕЧИ В АНИМАЦИЮ ЖЕСТОВОГО ЯЗЫКА

Курсовой проект

Ульяницкого Владимира Александровича

студента 3 курса,

специальность «Информатика»

Научный руководитель:

старший преподаватель кафедры МСС  
А. С. Гусейнова

Минск, 2019

# РЕФЕРАТ

Курсовой проект, 21 стр., 3 рис., 12 ист.

***Ключевые слова:*** распознавание речи, сурдоперевод, MOCAP, анимация, Android, MySQL, Leap Motion.

***Объект исследования*** – изучение вопроса использования мобильных приложений в качестве удобного инструмента для осуществления перевода голосовых команд в анимацию языка жестов.

***Цель работы*** – усовершенствовать средства коммуникации для людей с особенностями слуха и речи.

***Результат работы*** – изучены средства распознавания речи и захвата движения и применены для написания демо-приложения.

***Область применения*** – социальная сфера, создание комфортной и безопасной среды для людей с особенностями речи и слуха и облегчение для них интеграции в общество.

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 4](#_Toc27427482)

[ГЛАВА 1. ПОСТАНОВКА ЗАДАЧИ, ЦЕЛИ И ПРОБЛЕМЫ ПРОЕКТА 5](#_Toc27427483)

[1.1 Постановка задачи 5](#_Toc27427484)

[1.2 Цели проекта 5](#_Toc27427485)

[1.3 Проблемы проекта 6](#_Toc27427486)

[1.4 Выводы 6](#_Toc27427487)

[ГЛАВА 2. ОПИСАНИЕ ПРИЛОЖЕНИЯ И ЕГО ИНТЕРФЕЙСА 7](#_Toc27427488)

[2.1 Описание приложения 7](#_Toc27427489)

[2.2 Интерфейс приложения 7](#_Toc27427490)

[ГЛАВА 3. АРХИТЕКТУРА ПРИЛОЖЕНИЯ 9](#_Toc27427491)

[3.1 Основные компоненты приложения 9](#_Toc27427492)

[3.2 Модуль распознавания речи 9](#_Toc27427493)

[3.3 Реализация базы данных 12](#_Toc27427494)

[3.4 Клиентское приложение 12](#_Toc27427495)

[ГЛАВА 4. СРЕДСТВА, ИСПОЛЬЗУЕМЫЕ ДЛЯ РАЗРАБОТКИ ПРИЛОЖЕНИЯ 14](#_Toc27427496)

[4.1 Интегрированная среда разработки 14](#_Toc27427497)

[4.2 Захват движения. Технология Leap Motion 14](#_Toc27427498)

[4.3 Программное обеспечение для 3-D анимации 17](#_Toc27427499)

[ГЛАВА 5. ОБЗОР СУЩЕСТВУЮЩИХ АНАЛОГОВ 19](#_Toc27427500)

[ЗАКЛЮЧЕНИЕ 20](#_Toc27427501)

# ВВЕДЕНИЕ

Речь — один из наиболее важных инструментов взаимодействия между людьми. Мы постоянно пользуемся ей в нашей повседневной жизни для общения друг с другом, выражения своих мыслей и эмоций, зачастую забывая про то, что существуют люди, которые не имеют возможности использовать этот способ коммуникации. Мы привыкли к тому, что можем с лёгкостью воспринимать звуки речи и издавать их. Однако для людей, которые страдают нарушениями слуха или речи, это гораздо более сложная, а то и вовсе невыполнимая, задача. Общение с ними для обычного человека сильно затруднено, так как большая часть людей не умеет пользоваться языком жестов.

Решение данной проблемы имеет высокую социальную значимость. Это может являться частью программы создания комфортной и безопасной среды для людей с особенностями речи и слуха и облегчить для них интеграцию в общество. Это весьма актуально, ведь на учете Белорусского общества глухих на данный момент стоит около 10 тысяч инвалидов по слуху [1]. Приложение-сурдопереводчик могло бы значительно улучшить жизнь этих людей.

Идея о необходимости реализации данного приложения поступила от автошколы “Нужный шаг”, в которой ведется обучение глухонемых и слабослышащих людей. По словам руководства и преподавателей автошколы, они нуждаются в приложении, которое бы могло быстро переводить голосовые указания инструкторов по вождению в анимированные жесты.

# ГЛАВА 1. ПОСТАНОВКА ЗАДАЧИ, ЦЕЛИ И ПРОБЛЕМЫ ПРОЕКТА

## Постановка задачи

Задачей курсового проекта является реализация приложения под платформу Android для синхронного перевода устной речи в анимацию языка жестов. Данная задача была сформулирована исходя из нескольких требований к приложению: способность быстро реагировать на команды пользователя, возможность работы оффлайн, возможность использования без специального оборудования и удобство восприятия результата глухонемыми и слабослышащими пользователями.

Так как данное приложение предполагается использовать непосредственно на дороге в процессе поездки, очень важно, чтобы перевод происходил именно на язык жестов. Как удалось выяснить, язык жестов воспринимается глухонемыми проще, чем обычный текст, в связи с этим пользователи будут более комфортно себя чувствовать в процессе движения, своевременно реагировать на указания, которые будут поступать от инструкторов, и, соответственно, их обучение будет более удобным и качественным, что в дальнейшем позволит избежать аварийных ситуаций на дороге.

Исходя из подобного сценария использования, кажутся очевидными выбор платформы Android, как наиболее распространённой среди целевой аудитории мобильной операционной системы, а также необходимость обеспечить работу приложения без подключения к сети Интернет.

## Цели проекта

Проект ставит целью усовершенствовать средства коммуникации для людей с особенностями слуха и речи. В частности, приложение должно облегчить для таких людей обучение вождению.

Для достижения данной цели планируется предпринять следующие шаги:

1. Изучение проблематики для точного понимания задачи, которую собираемся решать;
2. Общение с целевой группой для понимания их потребностей и обсуждения возможностей будущего приложения;
3. Изучение доступных технологий, которые уже решают данную проблему. Анализ приложений-конкурентов с целью понять, что уже решено, и что может быть улучшено;
4. Определение перечня технических средств: платформы, доступных библиотек и решений, которые собираемся использовать для реализации прототипа;
5. Составление технического задания на основе требований целевой аудитории к приложению и возможностей, выбранных нами, технических средств;
6. Реализация приложения;
7. Тестирование приложения в реальной жизни.

## Проблемы проекта

Основная проблема проекта заключается в том, чтобы использовать технологию распознавания речи без какой-либо серверной составляющей, вместо этого используя исключительно возможности мобильного устройства. Это накладывает определённые ограничения на качество распознавания, однако вместе с этим значительно повышает его скорость.

Существует несколько способов решения проблемы недостаточной точности распознавания речи, основные из них это использование небольшого по объёму (200 – 300 слов) целевого словаря и адаптация акустической модели на входных данных, максимально приближенных к реальным [2].

## Выводы

Таким образом, в данной главе поставлены задачи и цели приложения, требования к разработке. Проанализированы проблемы и способы их решения.

# ГЛАВА 2. ОПИСАНИЕ ПРИЛОЖЕНИЯ И ЕГО ИНТЕРФЕЙСА

## 2.1 Описание приложения

В физическом смысле приложение после сборки представляет собой файл с расширением \*.apk для установки основной части приложения на устройство пользователя. Скачать приложение можно будет напрямую из Google Play Market — официального магазина приложений операционной системы Android. Целевые словари планируется сделать доступными для загрузки отдельно, непосредственно из приложения.

Приложение позволяет пользователю осуществлять перевод устной речи в её жестовую интерпретацию, с дополнительным выводом тестовой версии распознанного текста для контроля возможных неточностей распознавания, а также просматривать все доступные жесты из предварительно скачанных целевых словарей.

## 2.2 Интерфейс приложения

Весь интерфейс приложения можно условно разделить на три части:

* интерфейс главного экрана;
* интерфейс просмотра словаря;
* интерфейс настроек приложения.

Интерфейс главного экрана — интерфейс, предоставляющий пользователю возможность использования основной функции приложения –перевода. Служит для осуществления голосового ввода при нажатии соответствующей кнопки или при голосовой активации и для просмотра жестового перевода.

Интерфейс просмотра словаря служит для просмотра текстовой версии слов и фраз, содержащихся в словаре, а также их анимированных жестовых версий.

Интерфейс настроек приложения предоставляет пользователю возможность скачать требующиеся целевые словари либо удалить ненужные, установить ключевую фразу для активации функции распознавания и уровень чувствительности, с которым она будет распознаваться.

Навигация между элементами интерфейса осуществляется с помощью меню, расположенного в нижней части экрана. Меню создано с помощью средств разработки, встроенных в среду Android Studio [3]. Разметка интерфейса и стилизация объектов реализованы с помощью правил, описанных в структуре xml [4].

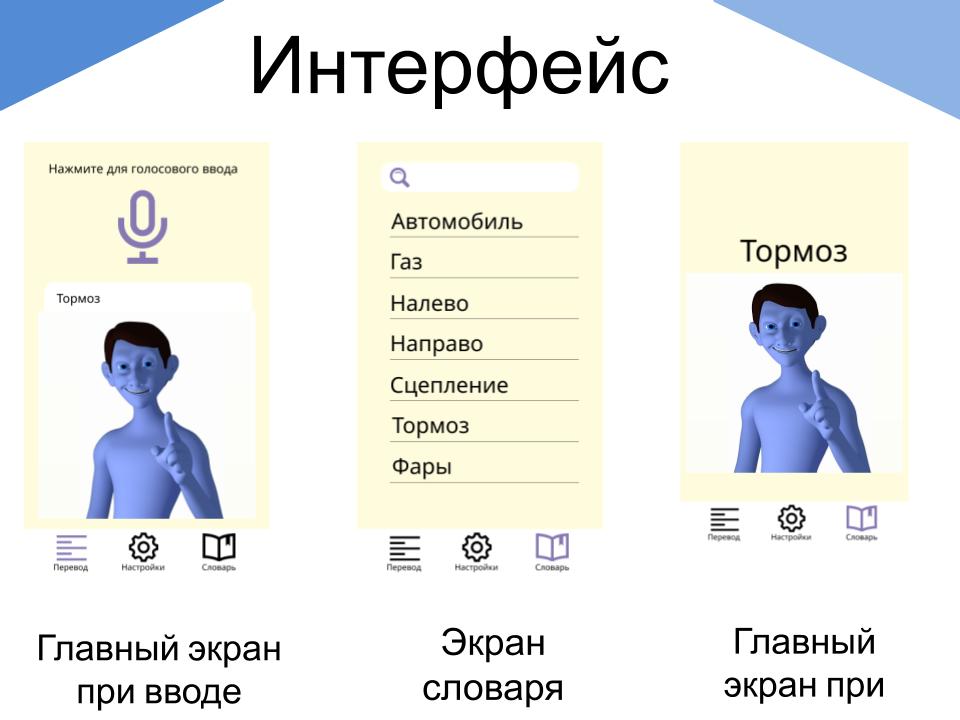


Рисунок 2.0.1 – Интерфейс приложения

# ГЛАВА 3. АРХИТЕКТУРА ПРИЛОЖЕНИЯ

## 3.1 Основные компоненты приложения

В качестве платформы, которая будет использована для разработки приложения (рис. 3.1), выбор был сделан в пользу Android, так как он является одним из самых распространенных и доступных. К тому же Android является более простым для начинающих разработчиков.

В структуре приложения можно выделить несколько основных компонент: модуль распознавания речи, база данных и клиентское приложение. Клиентское приложение при этом является прослойкой между базой данных и модулем распознавания речи. Такой подход позволяет упростить добавление и удаление целевых словарей без необходимости модификации всего приложения.



Рисунок 3.1 – Архитектура приложения

## 3.2 Модуль распознавания речи

Модуль распознавания речи используется для перевода устной речи пользователя в текст, который далее передаётся в клиентское приложение. Подавляющее большинство подобных систем работают онлайн: речь передаётся на удалённый сервер, где и происходит распознавание. Такой вариант не подходит для данного приложения, поэтому был сделан обзор среди систем, имеющих возможность работы офлайн. Также, для большего удобства интеграции в приложение, выбор производился среди систем, распространяемых с open-source лицензиями, т. к. это избавляет от ограничений на распространение итогового продукта и даёт больше возможностей для модификации, в том числе адаптации существующей акустической модели либо создания своей собственной.

Таким образом были рассмотрены следующие системы: CMU Sphinx, HTK, iAtros, Julius, Kaldi и RWTH ASR. Выбор системы зависел от таких критериев, как точность и скорость распознавания, удобство использования в программном коде и внутренняя структура. По точности системы оценивались исходя из наиболее популярных метрик: Word Recognition Rate (WRR), Word Error Rate (WER). По скорости распознавания сравнение было проведено с использованием Real Time Factor — показателя отношения времени распознавания к длительности распознаваемого сигнала, также известного как Speed Factor (SF) [5].

Таблица 3.1 — Результаты сравнения по точности и скорости

|  |  |  |  |
| --- | --- | --- | --- |
| Система | WER, % | WRR, % | SF |
| HTK | 19,8 | 80,2 | 1,4 |
| CMU Sphinx  (pocketsphinx/sphinx4) | 21.4/22.7 | 78.6/77.3 | 0,5/1 |
| Kaldi | 6,5 | 93,5 | 0,6 |
| Julius | 23,1 | 76,9 | 1,3 |
| iAtros | 16,1 | 83,9 | 2,1 |
| RWTH ASR | 15,5 | 84,5 | 3,8 |

Также было произведено сравнение по алгоритмам их работы. Первая стадия распознавания речи — извлечение признаков, значимых для распознавания. Все системы используют для этих целей мел-кепстральные коэффициенты (MFCC), а CMU Sphinx, Kaldi и RWTH ASR также более шумоустойчивые коэффициенты линейного перцепционного предсказания (PLP). Для акустического моделирования все рассмотренные системы используют статистические методы: скрытые марковские модели (HMM) — статистическая модель, имитирующая работу процесса, похожего на марковский процесс с неизвестными параметрами, задачей ставится разгадывание неизвестных параметров на основе наблюдаемых, — Kaldi, iAtros и RWTH ASR также могут использовать обобщённый метод моментов (GMM). Языковое моделирование выполняется во всех системах с помощью N-грамм и конечного автомата-преобразователя (кроме HTK). Непосредственно для распознавания используется алгоритм Витерби (HTK, CMU Sphinx, Julius, iAtros и RWTH ASR) и двухпроходной алгоритм прямого-обратного хода (Kaldi) [6, 7, 8].

Важным требованием также является наличие API. Этому требованию соответствуют только системы HTK, CMU Sphinx и Julius.

На основе вышеописанного для использования в проекте была выбрана система CMU Sphinx. CMU Sphinx включает в себя два декодера – pocketsphinx, реализованный на С, и sphinx4, реализованный на Java. Это позволяет применять данную систему на множестве платформ, в том числе под управлением операционной системы Android, что значительно облегчает ее интеграцию в проекты, написанные на Java. Данная система имеет модульную структуру, что положительно сказывается на возможности быстрого внесения изменений и исправления ошибок. С точки зрения удобства использования CMU Sphinx кроме консольного интерфейса предоставляет API, что существенно упрощает процесс встраивания системы в стороннее приложение. Также она обладает подробной тщательно проработанной документацией, которой сможет воспользоваться даже начинающий разработчик ПО, что сильно упрощает процесс знакомства и работу с системой. Также сильной стороной данной системы является поддержка множества языков по умолчанию, то есть наличие языковых и акустических моделей этих языков в свободном доступе. Среди поддерживаемых языков кроме стандартного английского встречаются также русский, казахский и ряд других. CMU Sphinx распространяется под лицензией BSD, что разрешает её встраивание в коммерческие проекты. Данная система может применяться в коммерческих проектах, так как обладает множеством достоинств, хотя и обеспечивает несколько худшую точность распознавания, а также предоставляет API, которое можно использовать для построения сторонних приложений на базе данной системы.

## 3.3 Реализация базы данных

Для хранения словарей необходима база данных – набор таблиц, связанных определёнными отношениями. В качестве базы данных для проекта было решено использовать SQLite — компактную встраиваемую СУБД. Данный вариант лучше всего походит для не высоконагруженной системы, хранящей достаточно малый объём данных. База данных приложения состоит из набора команд, который был составлен инструктором автошколы и анимированных изображений. Для хранения словаря, доступного пользователям, понадобится таблица, которая имеет поля: ID слова или выражения – сгенерированный искусственный номер лексической единицы, непосредственно сама лексическая единица и путь к анимации, которая хранится в локальных ресурсах приложения.

## 3.4 Клиентское приложение

Клиентское приложение, разработанное для операционной системы Android, используется пользователем для осуществления перевода и удобного доступа к словарям. Кроме того, пользователь использует клиентское приложение для изменения своего настроек приложения и скачивания необходимых ему целевых словарей.

При запуске приложения перед пользователем появляется главный экран, в связи с чем он может быстро и беспрепятственно начать работу.

Разделы приложения служат для группировки информации, предоставляемой пользователю, и включают в себя:

* раздел «Перевод» позволяет осуществлять основную работу по переводу и просмотру его результатов;
* раздел «Настройки» позволяет скачивать дополнительные словари, настраивать чувствительность голосового распознавания;
* раздел «Словарь» позволяет ознакомиться с целевым словарем, загруженным на устройство.

# ГЛАВА 4. СРЕДСТВА, ИСПОЛЬЗУЕМЫЕ ДЛЯ РАЗРАБОТКИ ПРИЛОЖЕНИЯ

## 4.1 Интегрированная среда разработки

Для реализации клиентского приложения и работы с базами данных была выбрана интегрированная среда разработки Android Studio от Google. Основные её преимущества: бесплатность для любого использования, в том числе в коммерческих проектах, и наилучшая поддержка разработки под систему Android, в том числе функции:

* Сборка приложений, основанная на Gradle;
* Рефакторинг кода;
* Статический анализатор кода Lint, выявляющий такие проблемы, как несовместимость версий и недостатки производительности;
* Редактор макетов, позволяющий легко переключаться между редактированием xml-файла и представлением WYSIWYG;
* Возможность предпросмотра макетов на различных конфигурациях экрана;
* Отладка приложения как на реальном устройстве, так и с помощью встроенного эмулятора Android Virtual Device, в котором можно настроить любую комбинацию разрешения, размера экрана и версии операционной системы [9].

## 4.2 Захват движения. Технология Leap Motion

Для создания анимации было решено использовать технологию захвата движения (MOCAP), которая позволяет перенести жесты реального человека на цифровую 3-D модель путём распознавания и синхронизации определённых ключевых точек на теле человека.

Для захвата движения рук предполагается использование Leap Motion. Это технология, основанная на захвате движения, осуществляющая человеко-компьютерное взаимодействие [10]. Ее разработала компания OcuSpec, основанная в 2010 г. Устройство представляет собой небольшой USB-адаптер, верхняя часть которого создает невидимую 3D-область взаимодействия достаточно большого объема, которую можно себе представить как куб с ребром в 61 см. Внутри данной области Leap Motion может отслеживать движение не только пальцев и рук пользователя, но также и других предметов: карандашей, ручек, палочек для еды и т.п. Удается это с помощью установленных в устройстве двух камер и трех ИК-светодиодов. Стоит отметить высокую скорость и точность захвата, она достигает 200 кадров в секунду. Для разработчиков предлагается абсолютно бесплатный SDK, который работает с 14 различных платформ и библиотек. Среди них есть такие известные платформы, как Unity 3D и Unreal Engine, также реализована поддержка различных языков программирования, например Processing (в том числе специально написанная на его базе среда разработки LeapMotionP5), С++, Objective-C, Java, JavaScript, AS3 и другие. Работа с предложенным SDK не составляет большого труда благодаря тому, что к нему прилагается подробная документация и множество примеров. Среди минусов можно отметить небольшое количество предопределенных движений, что вынуждает программиста самому пытаться математически описывать более сложные жесты (например, полет птицы, волну и т.д.). Некоторые разработчики решили самостоятельно начать работу над универсальным плагином или инструментом, который мог бы просто записать показанное движение, а потом его использовать, что могло бы сильно облегчить работу множеству программистов. Принцип работы устройства прост – инфракрасные (ИК) диоды подсвечивают руки, а инфракрасные камеры осуществляют их захват, передавая изображения программному обработчику Leap Motion. На программном уровне используются математические алгоритмы, которые выделяют контуры рук, и отслеживают координаты пальцев. Начиная с версии SDK 2.0 Leap Motion научился выделять составные части руки, проще говоря, алгоритм определяет кости рук и запястье, отслеживает их перемещение в пространстве. Таким образом, открываются новые горизонты для расширения базы распознаваемых жестов.

Распознавание рук происходит достаточно быстро, но скорость зависит от мощности компьютера, на котором, собственно, и происходит обработка данных, полученных с двух камер.

К недостаткам можно отнести невозможность распознавать жесты, которые требуют повернуть руку ребром к устройству. Также Leap Motion некорректно распознает жесты, в которых две руки соединяются вместе. Примеры таких жестов приведены на рисунке 4.1.

Leap Motion — революционный контроллер, он не только заменил сенсорный экран, но и подарил нам управление пространством, сделав еще более прозрачной границу между реальным миром и виртуальной реальностью.

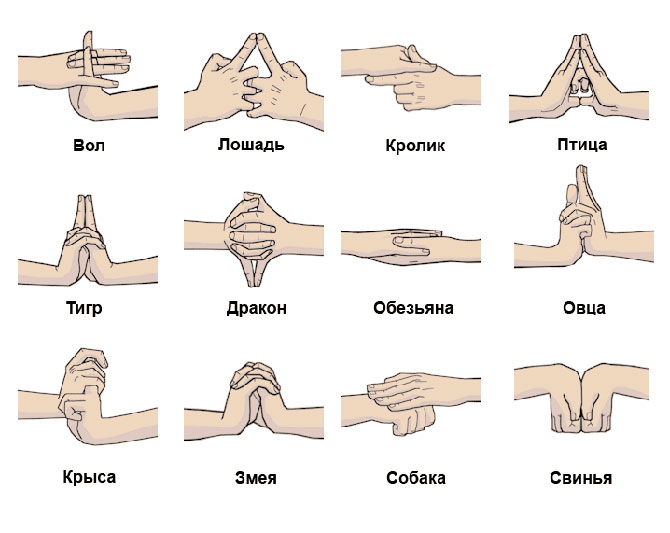


Рисунок 4.1 – Пример некорректно распознаваемых жестов

На уровне разработчика ПО мы получаем удобный программный интерфейс, позволяющий управлять всеми возможностями сенсора. Кроссплатформенные инструменты разработчика дают последнему доступ к устройству на множестве языков программирования, как компилируемых, так и интерпретируемых (пока среди последних только два — Python и JavaScript). Кроме того, API имеет стройную и понятную структуру: в каждый момент времени контроллер снимает изображение, формирует на основе его кадр и посылает на верхний уровень — в прикладную программу, где программист, обработав кадр, работает с такими сущностями, как руки, пальцы, указатели (инструменты) и другое. Из-за наличия в устройстве двух камер оно часто монтируется на очки виртуальной реальности для создания эффекта дополненной реальности, что достигается благодаря наличию в изображениях, снимаемых камерами, измеренных значений яркости инфракрасного излучателя, а также калибровочных данных, необходимых для коррекции сложного объектива.

## 4.3 Программное обеспечение для 3-D анимации

iClone — это программа для 3D-анимации и рендеринга в реальном времени, которая позволяет пользователям создавать 3D-анимационные фильмы [5]. Воспроизведение в режиме реального времени обеспечивается с помощью движка 3D-видеоигр для мгновенной визуализации на экране. Другие функции включают в себя: полную лицевую и скелетную анимацию фигур человека и животных, график для редактирования и объединения движений, язык сценариев (Lua) для взаимодействия персонажей, применение стандартных файлов захвата движения, возможность управлять анимационной сценой так же, как при воспроизведении видеоигры, и импорт моделей из Google 3D Warehouse, среди многих других функций. iClone также примечателен тем, что предлагает пользователям бесплатное использование всего контента, который они создают с помощью программного обеспечения, даже при использовании собственной библиотеки ресурсов Reallusion. iClone разработан и продается компанией Reallusion.

Функция захвата движения — еще одна функция iClone, которая позволяет пользователям подключать несколько устройств захвата движения и объединять их в один или несколько объектов в режиме реального времени. Это делается с помощью универсальной мокап-платформы iClone Motion LIVE. С iClone Motion LIVE пользователи могут получать профили (плагины) для своего оборудования для плавной потоковой передачи, захвата и последующего редактирования данных анимации во встроенном пакете редактирования движений iClone. Эти движения затем можно экспортировать с помощью программного обеспечения iClone 3DXchange Pipeline с использованием FBX и BVH. iClone также включает в себя плагин Unreal Engine 4 LIVE LINK, который транслирует анимацию всего тела, рук и лица непосредственно из iClone в UE4 для создания анимации в реальном времени. Подобный плагин в настоящее время разрабатывается для использования с iClone и системой Unity Mecanim.

Предполагается обрабатывать данные, полученные с Leap Motion, в iClone, а затем экспортировать полученную анимацию в среду разработки Unity 3D. Преимуществами данной среды можно считать кроссплатформенность, популярность, доступность. Платформа поддерживается и постоянно обновляется. Найти необходимую информацию, дополнительное программное обеспечение или расширить возможности стандартной платформы не составит труда.

# ГЛАВА 5. ОБЗОР СУЩЕСТВУЮЩИХ АНАЛОГОВ

Большинство существующих систем машинного сурдоперевода создано для работы на английском языке, примером таких систем может послужить проект ViSiCAST. Наиболее близко к текущему проекту приложение HandTalk, Приложение работает с виртуальным переводчиком Хьюго, который отвечает на голосовые и текстовые команды, преобразуя речь в жесты в режиме реального времени. Это также позволяет слушателям учиться общаться на языке жестов. Однако HandTalk может работать только на португальском языке, на данный момент также реализуется поддержка английского [11].

Примером русскоязычной разработки является российская система «Сурдофон». «Сурдофон» – инновационная компания, создатель первого отечественного программного обеспечения для оказания услуг дистанционного перевода русского жестового языка, онлайн-сервиса перевода русского жестового языка «Сурдофон» и одноименного мобильного приложения, которым сегодня пользуются сотни тысяч людей в России, Казахстане, Украине и других странах, где языком общения глухих является русский жестовый язык. Приложение «Сурдофон» — коммуникатор, который распознает речь говорящего собеседника и переводит ее на русский жестовый язык – это делает анимированная «кукла». Со своей стороны человек с ограничениями по слуху набирает текст, который через эту же «куклу» озвучивается компьютерным синтезатором речи. Таким образом, каждый получает информацию наиболее удобным для себя способом: глухой человек – на жестовом языке, слышащий – на звучащем [12].

К достоинствам программы «Сурдофон» можно отнести большой объём жестового словаря — около 10000 жестов. Однако есть и недостатки: программа требует подключения к серверам компании и, соответсвенно, работает только при наличии интернета. Также «Сурдофон» не может переводить речь синхронно.

# ЗАКЛЮЧЕНИЕ

В ходе работы над проектом была изучена проблематика задачи и обоснована её актуальность. В процессе общения с целевой группой был составлен список необходимых функций и возможностей приложения.

Далее были изучены уже существующие решения с целью анализа их преимуществ и недостатков, определения степени соответствия предъявленным требованиям и необходимости реализации собственного проекта.

После определения того, что все имеющиеся решения в той или иной степени не соответствуют требованиям, было принято решение реализовать приложение и изучены технологии, применимые для решения проблемы.

Была выбрана платформа для реализации приложения и определена его архитектура, определён перечень необходимых технических и программных средств. Был сделан выбор в пользу таких инструментов:

1. Распознавание речи — библиотека CMU Sphinx;

2. Захват движений рук — Leap Motion;

3. Создание анимации — iClone 7 и Unity 3D.

После выбора данных инструментов было проведено обучение работе с ними, рассмотрены примеры использования, и они были успешно применены в прототипе приложения.

Был создан тестовый прототип приложения. На данный момент в нем доступны следующие функции: ввод голосовой команды и получение видеофрагмента, который демонстрирует команду на языке жестов.

Также были определены цели на ближайшее будущее, а именно:

1. Реализация полной версии приложения;
2. Создание словаря, включающего все необходимые фразы и жесты;
3. Тестирование приложения в реальных условиях, исправление возможных недостатков.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Григорян А. Жизнь глухих. Какие проблемы волнуют белорусских инвалидов по слуху? / А. Григорян // tut.by [Электронный ресурс]. – 2016. - Режим доступа: https://news.tut.by/society/422456.html. - Дата доступа: 16.12.2019.
2. CMU Sphinx [Электронный ресурс]. Режим доступа: https://cmusphinx.github.io/. - Дата доступа: 25.11.2019.
3. Android Developers. Меню / Google // Android Developers [Электронный ресурс]. - Режим доступа: https://developer.android.com/guide/topics/ui/menus?hl=ru. - Дата доступа: 15.10.2019.
4. Android – Программирование для профессионалов / Б. Харди [и др.]; под общ. ред. Б. Харди — 2-е изд. — СПб.: Питер, 2016 — (Серия «Для профессионалов»). — 640 с.
5. Беленко, М. В., Сравнительный анализ систем распознавания речи с открытым исходным кодом / М. В. Беленко, П. В. Балакшин // Международный научно-исследовательский журнал [Электронный ресурс]. – 2017. - Режим доступа: https://research-journal.org/technical/sravnitelnyj-analiz-sistem-raspoznavaniya-rechi-s-otkrytym-kodom/ - Дата доступа: 08.10.2019
6. Kaldi [Электронный ресурс]. Режим доступа: http://kaldi-asr.org/doc - Дата доступа: 25.11.2019.
7. CMU Sphinx Wiki [Электронный ресурс]. Режим доступа: http://cmusphinx.sourceforge.net/wiki/ - Дата доступа: 25.11.2019.
8. Гребенкин Д. Почему Kaldi хорош для распознавания речи? / Д. Гребенкин // habr.com [Электронный ресурс]. – 2019. - Режим доступа: https://habr.com/ru/post/470696/. - Дата доступа: 16.12.2019.
9. Android Studio [Электронный ресурс]. Режим доступа: https://developer.android.com/studio/?hl=zh-cn. - Дата доступа: 15.12.2019.
10. Попов, Д. И., Технология Leap Motion и ее применение в образовательных процессах / Д. И. Попов, Д. Г. Григорьевич, М. В. Алпатова // CYBERLENINKA [Электронный ресурс]. – 2015. - Режим доступа: https://cyberleninka.ru/article/n/tehnologiya-leap-motion-i-ee-primenenie-v-obrazovatelnyh-protsessah - Дата доступа: 14.12.2019
11. HandTalk [Электронный ресурс]. Режим доступа: https://www.handtalk.me - Дата доступа: 27.11.2019.
12. Сурдофон [Электронный ресурс]. Режим доступа: https://www.xn--d1ascahfol.xn--p1ai/ - Дата доступа: 27.11.2019.