

МЕТОДОЛОГИЯ И ТЕХНОЛОГИЯ ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ В ЭПОХУ ЦИФРОВОЙ ТРАНСФОРМАЦИИ

DOI: 10.24888/2500-1957-2024-3-80-89

УДК
378.147.88

ТЕХНОЛОГИЯ ОБУЧЕНИЯ СТУДЕНТОВ ИТ НАПРАВЛЕНИЙ ИСПОЛЬЗОВАНИЮ SPEECH FRAMEWORK ДЛЯ РАСПОЗНАВАНИЯ РЕЧИ НА ПРИМЕРЕ ПОДКЛЮЧЕНИЯ К МОБИЛЬНОМУ IOS-ПРИЛОЖЕНИЮ

Красавцева Ксения Владимировна
магистрант
dksenia1612@gmail.com
г. Саранск

Национальный исследовательский
Мордовский университет
имени Н. П. Огарева

Щенникова Елена Владимировна
д.ф.-м.н., профессор
schennikova8000@yandex.ru
г. Саранск

Национальный исследовательский
Мордовский университет
имени Н. П. Огарева

Аннотация. Современные технологии стремительно движутся вперед. Появляются все более и более новые технологии. В эпоху цифровой трансформации появляется проблема оперативного обучения новым знаниям студентов высших учебных заведений. Одно из таких быстроразвивающихся направлений – это мобильная разработка. Актуальность данного направления состоит в том, что технологии развиваются, а мобильные устройства человек всегда имеет при себе. Этим и обуславливается постоянное использование девайсов, а также потребность в мобильных приложениях. Из-за дороговизны и малодоступности техники для разработки мобильных iOS-приложений и обучения, которое происходит на фирмах и предприятиях, сотрудничающих с высшими учебными заведениями, наблюдается нехватка специалистов по этому направлению. Такое плодотворное сотрудничество происходит в ходе производственной практики. В статье анализируется речевой фреймворк Speech Framework, предназначенный для распознавания и синтеза речи платформ iOS и macOS, история появления, основные инструменты, преимущества и недостатки. Также разрабатывается методология обучения подключения данного фреймворка на примере разработки мобильного iOS-приложения, предназначенного для распознавания речи. Такого рода приложения могут быть востребованы учебными заведениями для упрощения процесса обучения, ведь с помощью них можно проводить анализ речи обучающихся, диктанты, давать задания студентам и многое другое.

Ключевые слова: Speech Framework, iOS, приложение, фреймворк, технология, распознавание, синтез, речь, голос

Введение

В современном мире развитие IT-технологий движется стремительными шагами. Соответственно, возникает проблема обучения новым знаниям студентов бакалавра и магистратуры по IT-направлениям. Одним из быстро развивающихся и востребованных

направлений является мобильная разработка. Такие специалисты чрезвычайно востребованы на рынке труда. Поэтому необходимо постоянно развивать педагогические технологии и методики обучения в области разработки мобильных приложений. В статье рассмотрим один из этапов iOS-разработки – это поэтапный процесс использования Speech Framework для распознавания речи, на примере подключения к мобильному iOS-приложению.

В связи с тем, что техническое оснащение для iOS-разработки достаточно дорогое, то обычно соответствующие знания и практику студенты получают во время производственной практики на предприятиях или фирмах, сотрудничающих с высшим учебным заведением. С таким положением дел в мобильной разработке iOS-приложений повышается востребованность разработанной технологии обучения. Также важно отметить, что в списке литературы были только интернет-источники фирмы разработчика устройств Apple и Speech Framework. Это обусловлено абсолютной новизной рассматриваемой тематики.

Цель исследования: разработать технологию обучения, подробно описать этот технически сложный процесс – подключение Speech Framework для распознавания речи к мобильному iOS-приложению.

Speech Framework

В современном цифровом мире взаимодействие пользователей с техникой с каждым днем упрощается все больше и больше. Одним из самых распространённых методов взаимодействия пользователя с устройствами – управление голосом. Для того, чтобы устройство взаимодействовало с пользователем, нужно, чтобы оно понимало человека – распознавало речь. Корпорация Apple разработала свой универсальный фреймворк, основанный на нейронных сетях для распознавания речи пользователя, ее синтеза и обработки. Данный фреймворк имеет возможности создания приложений, которые могут способствовать улучшению произношения, а также оценивать его. Этот фреймворк дает возможность создания приложений для обучения иностранных студентов русскому языку, ведь благодаря данному фреймворку можно создавать аудио на основании текста, настраивая параметры произношения, оценивать произношение, а также давать обратную связь.

Speech Framework – это речевой фреймворк для платформ iOS и macOS, предназначенный для распознавания речи. Фреймворк способен не только распознавать произносимую речь, но и речь из аудиофайла.

Данный фреймворк впервые был представлен компанией Apple в 2016 году на конференции WWDC. Предназначался фреймворк для того, чтобы разработчики приложения платформ iOS и macOS имели возможность создавать приложения с функцией обработки и распознавания речи. Speech впервые появился в релизе iOS 10.0.1 того же года (Apple, 2024).

Apple использует свой собственный фреймворк во всех девайсах, вышедших с 2016 года. Speech используется для голосового помощника Siri, который также является разработкой компании Apple. С помощью этого фреймворка, который обеспечивает синтез и распознавание речи пользователя, голосовой помощник может общаться с пользователем, выполнять его команды. Таким образом, Speech является ключевым составляющим, обеспечивающим работу виртуального помощника Siri (Apple, 2024).

Фреймворк Speech имеет ряд преимуществ. Он предоставляет большой выбор различных инструментов, которые позволяют разработчикам приложений обрабатывать и распознавать речь пользователя. Примером необходимости наличия данного фреймворка являются приложения, предназначенные для обучения языкам. В таком приложении важно не только обучающемуся слышать то, что «говорит» приложение, но и самому приложению необходимо воспринимать, распознавать и обрабатывать речь, произносимую пользователем. Apple использует собственные алгоритмы и модели для обработки естественной речи, аудиоданных и преобразования их в текстовый формат.

Speech базируется на передовых технологиях машинного обучения и на технологиях обработки естественного языка. Он использует комбинации глубоких нейросетей и моделей глубокого обучения. В зависимости от поставленной задачи, Speech использует различные архитектуры нейросетей для синтеза, распознавания речи и получения речевых признаков.

Возможности, предоставляемые фреймворком, достаточно велики. Как уже было сказано ранее, он предоставляет возможность перевода не только естественной речи в текст, но и наоборот. Также Speech позволяет создавать аудиофайлы на основании текстовых данных (Apple, 2024).

Синтез речи Speech поддерживает множество настроек, таких как выбор языка, скорость речи и даже эмоциональность. Благодаря Speech разработчики iOS и macOS приложений могут создавать настраиваемые голосовые профили для адаптации речи под потребности пользователей.

Именно благодаря этим функциям и возможностям фреймворка есть возможность создания приложения, которое будет анализировать и оценивать произношение студентов на иностранном языке. Также функции фреймворка можно использовать для создания приложения с заданиями и упражнениями. Студенты должны будут выполнять их, взаимодействуя на языке, который они изучают, а приложение будет оценивать их произношение и речь. Примером одного из таких заданий может быть описание изображений, предоставляемых системой или же вопросно-ответные задания. Благодаря способности анализа, фреймворк может помогать преподавателям в проверке и оценивании речи и произношения студентов, ведь Speech может использоваться для автоматической проверки устных ответов на задания, что преподавателям сэкономит время, а студентам даст возможность самостоятельно улучшать свои навыки, под контролем приложения. Также можно с помощью данного фреймворка можно создать приложение для диктантов, настраивая скорость речи, выдерживание пауз и другие необходимые параметры, что поможет усовершенствовать слуховые навыки студентов и развить навыки понимания и восприятия иностранной речи.

Также немаловажно, что фреймворк поддерживает интеграцию с другими технологиями, что обеспечивает простое слияние с экосистемами разработчиков, для создания голосовых помощников, сохранения, создания и управления аудиофайлами. К тому же, Speech дает возможность управлять потоками шума и настраивать их. Например, предоставляет обработку фонового шума, позволяя его игнорировать, что немаловажно в процессе распознавания речи пользователя.

К сожалению, у фреймворка, как и у всего, есть свои недостатки. Как недостаток можно выделить – ограничение производительности на устаревших устройствах и устаревших версиях iOS или при обработке больших объемов данных. Также недостатком является наличие ограниченного выбора языков и голосовых моделей. Работа фреймворка напрямую зависит от доступа девайса к сети Интернет, что также является недостатком.

Подключение Speech Framework iOS приложению

Давайте рассмотрим, как же можно внедрить и использовать Speech Framework в iOS приложении.

Первый этап

Процесс распознавания речи пользователя включает в себя запись голоса и отправку этих данных на сервера Apple для обработки. Записанный звук голоса является конфиденциальными личными пользовательскими данными, которые нельзя использовать и отправлять на сервера Apple без разрешения пользователя. Именно поэтому разработчик предупреждает пользователя и запрашивает его согласие на отправку данных по сети, и соответственно на их обработку на серверах Apple.

С помощью API-интерфейсов, предоставляемых Speech Framework, запрашиваем авторизацию. Но сначала был подключен сам фреймворк и проинициализирован `SFSpeechRecognizer` (листинг 1).

`class SFSpeechRecognizer: NSObject` – это основной объект, который используется для распознавания речи (Apple, 2024).

```
import UIKit
import Speech

class ViewController: UIViewController, SFSpeechRecognizerDelegate {
    private let speechRecognizer = SFSpeechRecognizer(locale: Locale(identifier:
"ru-RU"))
}
```

Листинг 1. Импорт фреймворка и его инициализация

Locale.init(indentifiew: "ru-RU") – указывает язык, который будет воспринимать и обрабатывать приложение. В данном случае мы установили русский язык.

Затем обязательно добавили в файл Info.plist ключ NSSpeechRecognitionUseagateDescription и установили значение этого ключа (Apple, 2024). NSSpeechRecognitionUseagateDescription – это ключ, который отвечает за объяснение пользователю, для каких целей запрашивается доступ к конфиденциальным данным, а именно к записи и отправке пользовательской речи. Добавление ключа в файл Info.plist представлено на рис. 1.



Рис. 1. Добавление ключа в файл Info.plist

В значении ключа подробно описываем причину, чтобы завоевать доверие пользователя. Доступ к конфиденциальным данным с большей вероятностью разрешат тому приложению, в котором четко и понятно будет описана итоговая цель. Приложения, которые не внушают пользователю доверия, скорее всего будут просто удалены с девайса, или же доступ будет не разрешен (а приложения, основа которых – это взаимодействие с речью, будут просто бесполезны).

Когда приложение запросит авторизацию, система отобразит пользователю значение этого ключа, как часть системного приглашения.

Теперь настроим сам запрос на разрешение доступа к голосовым данным пользователя. Вызываем requestAuthorization(_:) для SFSpeechRecognizer. Этот метод выполняется асинхронно и возвращает результаты в блок (флаг), предоставленный разработчиком. Этот блок используется для определения – одобрил пользователь запрос приложения на доступ к данным или отклонил. Также обязательно устанавливается делегат. Код данного запроса представлен на листинге 2.

После запуска приложение отображает окно с запросом разрешение на доступ к данным пользователя (речи), представленное на рис. 2.

```
speechRecognizer?.delegate = self
// Запрос на авторизацию
SFSpeechRecognizer.requestAuthorization { authStatus in
    OperationQueue.main.addOperation {
        switch authStatus {
            case .notDetermined:
                self.speechButton.isEnabled = false
                self.speechButton
```

```
.setTitle("Распознавание речи в этом приложении еще не разрешено",
for: .disabled)
case .denied:
    self.speechButton.isEnabled = false
    self.speechButton.setTitle("Приложению запрещен доступ к распознаванию
речи",
for: .disabled)
case .restricted:
    self.speechButton.isEnabled = false
    self.speechButton.setTitle("Распознавание речи работает в ограниченном
режиме",
for: .disabled)
case .authorized:
    self.speechButton.isEnabled = true
}
}
```

Листинг 2. Код запроса авторизации

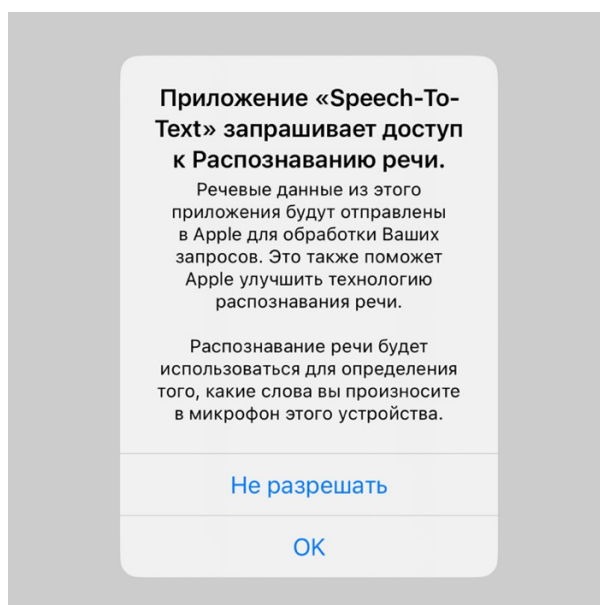


Рис. 2. Окно запроса на доступ к данным пользователя

Второй этап

Теперь настраиваем восприятие речи приложением и ее обработку. Для этого проинициализируем остальные необходимые элементы, необходимые для восприятия речи приложением. Пример инициализации представлен на листинге 3.

`SFSpeechBufferRecognitionRequest` – это запрос с ответом типа `SFSpeechRecognitionRequest`. Запрос захватывает аудио-контент, например, с микрофона устройства, и отправляет для распознавания речи (Apple, 2024).

```
private var recognitionRequest: SFSpeechAudioBufferRecognitionRequest?
private var recognitionTask: SFSpeechRecognitionTask?
private let audioEngine: AVAudioEngine?
```

Листинг 3. Инициализация `SFSpeechBufferRecognitionRequest`, `SFSpeechRecognitionTask`, `AVAudioEngine`

Для отслеживания прогресса распознавания речи используем объект `SFSpeechRecognitionTask` (Apple, 2024).

Также является необходимым аудио-движок – AVAudioEngine, который содержит группу экземпляров AVAudio, использующихся для формирования цепочки по обработке аудио (Apple, 2024).

Теперь напишем сам метод, который позволит приложению воспринимать аудио, обрабатывать и записывать его для дальнейшего вывода. Для этого нужно получить экземпляр для управления аудио-сессией. Для нашей аудио-сессии устанавливаем категорию AVAudioSession.Category.record. Это необходимо для приглушения остальных входных данных, пока идет сессия. Это предотвращает помехи, при восприятии речи приложением (системой). Далее устанавливаем режим измерения аудио AVAudioSession.Mode.measurement, который обеспечивает высокий захват звука, и его обработки, что необходимо и является важным при выполнении задачи распознавания и обработки речи [8]. Также устанавливаем активацию audioSession с флагом notifyOthersOnDeactivation для отслеживания статуса audioSession (Apple, 2024). Код настройки audioSession представлен на листинге 4.

```
let audioSession = AVAudioSession.sharedInstance()
do {
    try audioSession.setCategory(AVAudioSession.Category.record)
    try audioSession.setMode(AVAudioSession.Mode.measurement)
    try audioSession.setActive(true, options: .notifyOthersOnDeactivation)
} catch { print("Ошибка установки свойств AVAudioSession") }
```

Листинг 4. Код настройки AVAudioSession

Далее создали запрос SFSpeechAudioBufferRecognitionRequest, необходимый для распознавания аудио, который нужен для передачи аудио на дальнейшее распознавание, а также установили флаг для запроса распознавания (Apple, 2024). Флаг будет указывать системе, что результаты частичного распознавания должны быть отправлены, когда станут доступны (листинг 5).

```
recognitionRequest = SFSpeechAudioBufferRecognitionRequest()
recognitionRequest?.shouldReportPartialResults = true
```

Листинг 5. Код создания запроса SFSpeechAudioBufferRecognitionRequest и его настройка

Третий этап

Теперь приступаем к написанию части, которая будет отвечать за перехват аудио-буферов с узла ввода. Аудио-буферы необходимы для передачи речи пользователя в процесс распознавания. Получаем узел ввода audioEngine, который предоставляет доступ к входному аудио-потoku устройства audioEngine.inputNode. Далее устанавливаем формат записи звука для этого узла outputFormat(forBus: 0). Он определяет характеристики аудио, а именно: количество каналов, частоту дискретизации и точность битов. И устанавливаем tap («кран») на узле ввода, необходимый для перехвата аудио-буфера, который будет поступать через канал, чтобы выполнять с ним какие-то действия (Apple, 2024). В данном проекте, в случае появления нового аудио-буфера добавляем его к существующему запросу распознавания речи. Теперь записываем метод audioEngine.prepare(), который выполняет действия по подготовке к записи аудио и воспроизведения аудио. Собираем фрагмент кода, для которого мы настроили данные. Код запускает audioEngine для работы с записью звука. Пример настройки audioEngine представлен на листинг 6.

```
let inputNode = audioEngine?.inputNode
let recordFormat = inputNode.outputFormat(forBus: 0)
inputNode.installTap(onBus: 0, bufferSize: 1024, format: recordFormat) { buff, _ in
    self.recognitionRequest?.append(buff)
}
audioEngine?.prepare()
```

```
do {
    try audioEngine?.start()
} catch {
    print("Ошибка запуска AVAudioEngine \(error)")
}
```

Листинг 6. Код настройки audioEngine

Четвертый этап

Наконец после настройки всех необходимых объектов записываем сам запрос на распознавание речи. Запрос возвращает либо response (ответ) типа SFSpeechRecognitionRequest, либо ошибку. Для того, чтобы текст был читаемым, то есть в формате строки, указываем свойство bestTranscription.formattedString (листинг 7). Если запрос отработал успешно, то мы выводим речь пользователя в виде текста в заранее созданное textView. Этот текст можно вывести куда угодно, либо сохранить для дальнейшей обработки.

```
recognitionTask = speechRecognizer?.recognitionTask(with: recognitionRequest) { result, error in
guard let result = result else { return }
self.textView.text = result.bestTranscription.formattedString
}
```

Листинг 7. Код запроса на распознавание речи

Пятый этап

Теперь собираем все куски кода вместе и получаем метод, который будет распознавать аудио пользователя и выводить в UITextView. Готовый метод startRecording() представлен на листинг 8.

Результат

После дополнительной настройки интерфейса было создано мобильное приложение, способное распознавать речь пользователя (рис. 3). После нажатия на кнопку «Начать» текст записывается в режиме онлайн в UITextView, и пользователь через долю секунды видит текст, который он сказал.

```
func startRecording() {
    let audioSession = AVAudioSession.sharedInstance()
    do {
        try audioSession.setCategory(AVAudioSession.Category.record)
        try audioSession.setMode(AVAudioSession.Mode.measurement)
        try audioSession.setActive(true, options: .notifyOthersOnDeactivation)
    } catch { print("Ошибка установки свойств AVAudioSession") }

    recognitionRequest = SFSpeechAudioBufferRecognitionRequest()
    recognitionRequest?.shouldReportPartialResults = true

    let inputNode = audioEngine?.inputNode
    let recordFormat = inputNode?.outputFormat(forBus: 0)
    inputNode?.installTap(onBus: 0, bufferSize: 1024, format: recordFormat) { buff, _ in
        self.recognitionRequest?.append(buff)
    }

    audioEngine?.prepare()

    do {
        try audioEngine?.start()
    } catch { print("Ошибка запуска AVAudioEngine \(error)") }

    guard let recognitionRequest = recognitionRequest else {
        return
    }
    recognitionTask = speechRecognizer?.recognitionTask(
```

```

        with: recognitionRequest
    ) { result, error in
        guard let result = result else { return }
        self.textView.text = result.bestTranscription.formattedString
    }
}

```

Листинг 8. Код метода распознавания речи пользователя

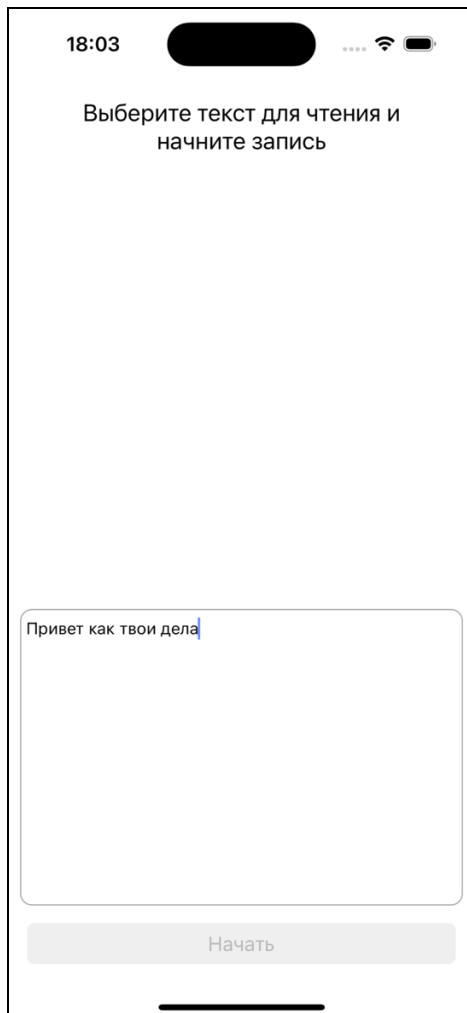


Рис. 3. Интерфейс приложения, способного распознавать пользовательскую речь

Заключение

В данном исследовании разработана технология обучения использованию Speech Framework для распознавания речи. Speech Framework – прекрасный инструмент, предоставляемый корпорацией Apple, для взаимодействия пользователя и мобильного приложения. Использование данного фреймворка открывает студентам – будущим разработчикам – множество возможностей для создаваемых приложений. С помощью Speech разработчики могут создавать любые iOS-приложения с анализом речи и голоса пользователя, например, обучающие. Благодаря данному фреймворку можно создавать приложения для тренировок и диктантов для студентов, изучающих различные языки как русский, так и английский и многие другие, ведь фреймворк поддерживает большое множество языков и диалектов. Также фреймворк можно использовать для создания приложения для проведения интерактивных уроков, в которых студентам необходимо будет разговаривать и взаимодействовать с приложением на необходимом для изучения языке, для выполнения интерактивных заданий и упражнений.

Разработанная технология обучения студентов IT-направлений облегчает и сокращает освоение технически сложного нового материала по мобильной разработке.

Список литературы

- Apple. Framework Speech [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/speech?language=swift,swift,swift,swift> (дата обращения: 22.03.24).
- Apple. Deep Learning for Siri's Voice: On-device Deep Mixture Density Networks for Hybrid Unit Selection Synthesis [Электронный ресурс] // Machine Learning Research. URL: <https://machinelearning.apple.com/research/siri-voices> (дата обращения: 30.03.24).
- Apple. SFSSpeechRecognizer [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/speech/sfspeechrecognizer> (дата обращения: 01.04.24).
- Apple. Managing Your App's Information Property List [Электронный ресурс] // Developer. URL: https://developer.apple.com/documentation/bundleresources/information_property_list/managing_your_app_s_information_property_list (дата обращения: 02.04.24).
- Apple. SFSSpeechAudioBufferRecognitionRequest [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/speech/sfspeechaudiobufferrecognitionrequest> (дата обращения: 04.04.24).
- Apple. SFSSpeechRecognitionTask [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/speech/sfspeechrecognitiontask> (дата обращения: 15.04.24).
- Apple. AVAudioEngine [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudioengine> (дата обращения: 18.04.24).
- Apple. AVAudioSession [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudiosession> (дата обращения: 18.04.24).
- Apple. NotifyOthersOnDeactivation [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudiosession/setactiveoptions/1616603-notifyothersondeactivation> (дата обращения: 19.04.24).
- Apple. OutputFormat(forBus:) [Электронный ресурс] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudionode/1389195-outputformat> (дата обращения: 24.04.24).

TECHNOLOGY FOR TRAINING IT STUDENTS IN THE USE OF SPEECH FRAMEWORK FOR SPEECH RECOGNITION BY THE EXAMPLE OF CONNECTING TO AN IOS MOBILE APPLICATION

Krasavtceva K. V.
undergraduate student
dksenia1612@gmail.com
Saransk

National Research Mordovian State University
named after N. P. Ogarev

Shennikova E. V.
Dr. Sci. (Physical and mathematical)
associate professor
schennikova8000@yandex.ru
Saransk

National Research Mordovian State University
named after N. P. Ogarev

Abstract. Modern technologies are rapidly moving forward. More and more new technologies are appearing. Because of this, the problem of teaching new knowledge to students of higher educational institutions arises. One of these fast-growing areas is

mobile development. The relevance of this area lies in the fact that technologies are developing, and people always have mobile devices with them. This determines the constant use of devices, as well as the need for mobile applications. Due to the high cost and inaccessibility of technology for developing mobile iOS applications, and the training that takes place at firms and enterprises collaborating with higher education institutions, there is a shortage of specialists in this area. The article analyzes the speech framework Speech Framework, designed for speech recognition and synthesis of the iOS and macOS platforms, the history of its appearance, the main tools, advantages and disadvantages. A methodology for teaching how to connect this framework is also being developed using the example of developing a mobile iOS application designed for speech recognition. This kind of application may be in demand by educational institutions to simplify the learning process, because with their help you can analyze students' speech, conduct dictations, give assignments to students, and much more.

Keywords: Speech Framework, iOS, application, framework, technology, recognition, synthesis, speech, voice

References

- Apple. Framework Speech [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/speech?language=swift,swift,swift,swift> (date of the application: 22.03.24).
- Apple. Deep Learning for Siri's Voice: On-device Deep Mixture Density Networks for Hybrid Unit Selection Synthesis [Электронный ресурс] // Machine Learning Research. URL: <https://machinelearning.apple.com/research/siri-voices> (date of the application: 30.03.24).
- Apple. SFSpeechRecognizer [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/speech/sfspeechrecognizer> (date of the application: 01.04.24).
- Apple. Managing Your App's Information Property List [Electronic resource] // Developer. URL: https://developer.apple.com/documentation/bundleresources/information_property_list/managing_your_app_s_information_property_list (date of the application: 02.04.24).
- Apple. SFSpeechAudioBufferRecognitionRequest [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/speech/sfspeechaudiobufferrecognitionrequest> (date of the application: 04.04.24).
- Apple. SFSpeechRecognitionTask [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/speech/sfspeechrecognitiontask> (date of the application: 15.04.24).
- Apple. AVAudioEngine [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudioengine> (date of the application: 18.04.24).
- Apple. AVAudioSession [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudiosession> (date of the application: 18.04.24).
- Apple. NotifyOthersOnDeactivation [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudiosession/setactiveoptions/1616603-notifyothersondeactivation> (date of the application: 19.04.24).
- Apple. OutputFormat(forBus:) [Electronic resource] // Developer. URL: <https://developer.apple.com/documentation/avfaudio/avaudionode/1389195-outputformat> (date of the application: 24.04.24).

Статья поступила в редакцию 18.06.2024

Принята к публикации 16.09.2024