**РОСЖЕЛДОР**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

**ВЫСШЕГО ОБРАЗОВАНИЯ**

**СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ПУТЕЙ СООБЩЕНИЯ (СГУПС)**

|  |  |  |  |
| --- | --- | --- | --- |
| **К защите:** |  |  | |
| **Заведующий кафедрой** | **Информационные** | |
| **технологии транспорта** | | |
|  | д-р техн. наук, проф. | |
|  |  | В. И. Хабаров | |
| *подпись* |  | *инициалы, фамилия* | |
|  |  |  | |
| *дата* |  |  | |

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Тема:** | Разработка системы автоматизированного контроля регламента переговоров | | | | | |
|  | для тренажёрного комплекса оперативного персонала сортировочной горки | | | | | |
|  | |  | БР.БИСТ.09.2022 |  |  |
|  | |  | *шифр документа* |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **Выполнил** |  |  |  | **Руководитель** |
|  |  | К.Е. Коноваленко |  |  |  | канд. техн. наук, доц.  Е. Б. Тарасов |
| *подпись* |  | *инициалы, фамилия* |  | *подпись* |  | *инициалы, фамилия* |
|  |  |  |  |  |  |  |
| *дата* |  |  |  | *дата* |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Консультанты по разделам** |  |  |  |  |
| 1 |  |  |  |  |
|  |  |  |  |  |
| 2 |  |  |  |  |
|  |  |  |  |  |
| 3 |  |  |  |  |
|  |  |  |  |  |
| Нормоконтролер работы |  |  |  | ст. преп.  Т. А. Распопина |
|  |  | *подпись* |  |  |
|  |  |  |  |  |
|  |  | *дата* |  |  |

**2022 г.**

**СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ПУТЕЙ СООБЩЕНИЯ (СГУПС)**

Факультет: Бизнес-информатики

Кафедра: Информационные технологии транспорта

Направление: 09.03.02 «Информационные системы и технологии»

Профиль: Интеллектуальные транспортные системы

|  |  |
| --- | --- |
|  | ***УТВЕРЖДАЮ****: зав. кафедрой «Информационные технологии транспорта»*  д-р техн. наук, проф.  В. И. Хабаров |
|  | *«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_ 2022 г.* |

**З А Д А Н И Е**

**на выполнение выпускной квалификационной работы**

|  |  |  |  |
| --- | --- | --- | --- |
| студенту | Коноваленко Кириллу Евгеньевичу | | |
|  |  | | |
| 1. Тема «Разработка системы автоматизированного контроля регламента переговоров для тренажёрного комплекса оперативного персонала сортировочной горки утверждена приказом № 203/с от «30» мая 2022 г. | | | |
| 2. Задание выдано «12» мая 2022 г. | | | |
| 3. Срок сдачи законченной работы на кафедру «17» июня 2022 г. | | | |
| 4. Исходные данные: данные, полученные в ходе прохождения преддипломной практики | | | |
| 5. Содержание расчетно-пояснительной записки | | | |
| Наименование разделов и вопросов | | Примерное количество страниц | График (сроки) выполнения |
| Введение | | 1 | 13.05.2022 |
| Аналитическое исследование | | 5 | 13.05.2022 |
| Проектирование информационной системы | | 15 | 15.05.2022 |
| Создание приложения | | 16 | 28.05.2022 |
| Заключение | | 1 | 01.06.2022 |

6. Содержание и объемы графической части

|  |  |  |
| --- | --- | --- |
| Наименование графического документа (чертежа, схемы, графика) | Количество  листов  формата А1 | График  (сроки)  выполнения |
| Презентация PowerPoint | 12 | 05.05.2022 |

7. Консультанты по разделам

|  |  |  |
| --- | --- | --- |
| Наименование  раздела | Фамилия, И. О.  консультанта | Подпись консультанта,  дата выдачи задания |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| Руководитель |  | Е. Б. Тарасов |
|  | *(подпись, фамилия, И.О.)* |  |
| Задание к использованию принял |  | К. Е. Коноваленко |
|  | *(подпись студента)* |  |

УДК 004.41

**АННОТАЦИЯ**

В работе 40 страниц, 22 рисунка, 5 таблиц, 10 источников.

Ключевые слова: *тренажерный комплекс, приложение, регламент переговоров, система, автоматизированный контроль.*

Предметная область - тренажерный комплекс оперативного персонала сортировочной горки. Приложение состоит из двух частей: для обучаемого и для инструктора. Приложение для инструктора выполняет административные функции: добавление новых ролей, добавление новых команд, отображение статистики по обучаемым. Также оно играет роль сервера, связывающего всех обучаемых. Приложение для обучаемых имеет более узконаправленный функционал: распознавание и генерация речи, отправка результатов на сервер.

**ABSTRACT**

The work contains 40 pages, 22 figures, 5 tables, 10 sources.

Keywords: *training complex, application, negotiation rules, system, automated control*.

The subject area is the training complex of the operational staff of the sorting hill. The application consists of two parts: for the trainee and for the instructor. The instructor application performs administrative functions: adding new roles, adding new teams, displaying statistics on trainees. It also plays the role of a server connecting all trainees. The application for trainees has more narrowly focused functionality: speech recognition and generation, sending results to the server.

**ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

ИС – информационная система.

ПО – программное обеспечение.

C# – объектно-ориентированный язык программирования высокого уровня.

.NET – программная платформа компании Microsoft.

MS – компания Microsoft.

Visual Studio – среда разработки от компании Microsoft.

БД – база данных.

ИИ – искусственный интеллект.

ER (Entity Relationship) – модель «сущность-связь».

ТК (тренажерный комплекс) – система связанных автоматизированных рабочих мест, предназначенных для обучения и тестирования персонала в рамках предметной области.

Оперативный персонал – персонал компании, находящийся на смене и выполняющий свои непосредственные обязанности.

Сортировочная горка – устройство на сортировочных железнодорожных станциях для ускоренного расформирования подвижных составов путем скатывания вагонов с уклона.

ОПСГ – оперативный персонал сортировочной горки.

ПТЭ – правила технической эксплуатации железных дорог Российской Федерации.

АРМ – автоматизированное рабочее место.

**СОДЕРЖАНИЕ**

[Введение 6](#_Toc106576870)

[1 Аналитическое исследование 7](#_Toc106576871)

[1.1 Определение предметной области 7](#_Toc106576872)

[1.2 Определение проблемы предметной области 8](#_Toc106576873)

[1.3 Решение проблемы 8](#_Toc106576874)

[1.4 Постановка цели, задачи 9](#_Toc106576875)

[1.5 Вывод об аналитическом исследовании 10](#_Toc106576876)

[2 Проектирование информационной системы 11](#_Toc106576877)

[2.1 Моделирование бизнес-процессов информационной системы 11](#_Toc106576878)

[2.2 Описание бизнес-процессов до и после внедрения приложения 17](#_Toc106576879)

[2.3 Структура ИС и ее средства разработки 18](#_Toc106576880)

[2.4 Описание структуры базы данных 20](#_Toc106576881)

[2.5 Обзор технологий распознавания речи 22](#_Toc106576882)

[2.6 Перспективы развития разработанной системы 24](#_Toc106576883)

[2.7 Вывод о проектировании системы 25](#_Toc106576884)

[3 Реализация системы 26](#_Toc106576885)

[3.1 Интерфейс обучаемого 26](#_Toc106576886)

[3.2 Интерфейс инструктора 29](#_Toc106576887)

[3.3 Интеграция базы данных 32](#_Toc106576888)

[3.4 Руководство пользователя 35](#_Toc106576889)

[Заключение 38](#_Toc106576890)

[Список использованных источников 39](#_Toc106576891)

# **ВВЕДЕНИЕ**

Во всех сферах деятельности главное – это взаимодействие людей, большой проект не посилен одному человеку. На железнодорожных станциях, помимо указателей и светофоров, основная часть информации передается через голосовые команды. Для обеспечения однозначности сообщений, во избежание аварийных ситуаций, вызванных неверной интерпретацией услышанного, голосовые команды стандартизируются и закрепляются в регламенте переговоров.

Такие регламенты по значимости сравнимы с нормативно-правовыми актами, без которых не обходится работа никаких предприятий.

В обучении регламентам существует проблема контроля. Так, обучающий одновременно может качественно контролировать максимум двух обучаемых за раз. Это очень медленно и дорого. Также присутствует человеческий фактор, оценивание может быть не объективным. Для решения данной проблемы возникает необходимость в разработке программного продукта, автоматизирующего данный процесс.

Но как программа может оценить устную речь? Для начала звуковую информацию нужно перевести в символьную. Решение проблемы распознавания и генерации речи находится на острие технологического прогресса, для осуществления проекта необходимо прибегнуть к использованию искусственного интеллекта.

В ходе выпускной квалификационной работы будет разработано приложение автоматизированного контроля регламента переговоров на примере внедрения в систему обучения, подготовки и аттестации оперативного персонала сортировочной горки на базе специализированных тренажерных комплексов (ТК ОПСГ) на полигоне ОАО «РЖД». Разработанное приложение может быть внедрено и в другие тренажерные комплексы.

# **1 Аналитическое исследование**

## **1.1 Определение предметной области**

Тренажерный комплекс оперативного персонала сортировочной горки (ТК ОПСГ) – модель сортировочной горки с возможностью отработки основных навыков работы дежурного по станции и операторов сортировочной горки в учебном режиме. ТК ОПСГ является продуктом команды разработчиков федерального государственного бюджетного образовательного учреждения высшего образования «Сибирский государственный университет путей сообщений» (ФГБОУ ВО «СГУПС»), созданным для пользования открытым акционерным обществом «Российские Железные Дороги» (ОАО «РЖД»).

ТК ОПСГ разработан для операторов горочного комплекса с целью практической отработки навыков исполнения основных должностных обязательств [1].

Концептуальные части тренажера:

* сервер – обеспечивает синхронную работу ТК ОПСГ;
* автоматизированное рабочее место «Инструктор», выполняющее настройку и запуск моделируемого процесса;
* автоматизированные рабочие места работников станции, участвующие в моделируемых процессах.

Ключевые процессы тренажерного комплекса:

* моделирование реальной ситуации железнодорожной станции;
* выполнение обучаемыми на автоматизированных рабочих местах надлежащих действий по разрешению ситуации. Действия выполняются согласно правилам технической эксплуатации железных дорог (ПТЭ);
* проверка инструктором корректности решений, принятых обучаемыми, а также соответствия их действий ПТЭ с дальнейшими выводами по работе персонала сортировочной горки в моделируемой ситуации.

## **1.2 Определение проблемы предметной области**

Главным для тренажерного комплекса, как и для реальной железнодорожной станции, является согласованность действий участников процессов. Для обеспечения согласованности действий персонал обменивается сообщениями в устной форме. Для максимизации эффективности коммуникации был разработан регламент переговоров, обеспечивающий четкость и понятность каждого сообщения или команды.

Отработка применения регламента переговоров является важной частью процесса обучения. На данный момент эта ответственность полностью ложится на плечи инструктора, что влечет за собой следующие негативные последствия:

* перегруженность инструктора разнотипными задачами;
* необъективность оценки исполнения регламента ввиду человеческого фактора;
* невозможность параллельного контроля нескольких обучаемых.

Инструктор на ТК ОПСГ должен моделировать реальные ситуации железнодорожной станции, отслеживать правильность действий, направленных на их разрешение. Стоит отметить, что на станции может параллельно работать несколько пар операторов, что значительно затрудняет процесс контроля.

## **1.3 Решение проблемы**

Для устранения описанных проблем было принято решение разработать приложение автоматизированного контроля регламента переговоров.

Данная разработка позволит снизить загруженность инструктора, а также повысить точность оценки исполнения регламента. Также приложение позволит без дополнительных усилий собирать статистку успешности исполнения регламента по каждому обучающемуся.

С внедрением приложения автоматизированного контроля регламента переговоров, на тренажерных комплексах появится возможность контроля и оценки устных сообщений без участия человека, что позволит получать информацию о качестве обучения в виде статистики. Статистику можно будет применять для выявления пробелов в знаниях обучаемых и повышения эффективности проводимых занятий по подготовке персонала.

## **1.4 Постановка цели, задачи**

В ходе выпускной квалификационной работы необходимо разработать приложение автоматизированного контроля регламента переговоров.

Разрабатываемое приложение должно соответствовать следующим требованиям:

* возможность добавления новых команд и сообщений;
* наличие удобного конструктора команд;
* возможность распознавания речи;
* возможность синтеза речи;
* функция сбора и ведение статистики;
* представление статистики в удобном виде – по конкретному обучающемуся и по группе в целом.

Для достижения поставленной цели необходимо:

* провести анализ предметной области;
* изучить структуру регламента переговоров;
* проанализировать требования к системе;
* выбрать технологию распознавания и синтеза речи;
* выбрать способ представления отчетности;
* разработать приложение автоматизированного контроля регламента переговоров с учетом всех требований и особенностей предметной области;
* проверить соответствие разработанного приложения всем требованиям;
* протестировать разработанное приложение и при необходимости исправить его недостатки.

## **1.5 Вывод об аналитическом исследовании**

По результатам проведения анализа предметной области был выявлен следующий список проблем контроля исполнения регламента переговоров, возникающий при обучении операторов железнодорожной станции на ТК ОПСГ:

* высокая загруженность инструктора по тренажеру;
* необъективность оценки соответствия устных сообщений регламенту переговоров ввиду присутствия человеческого фактора;
* отсутствие статистических данных по процессу обучения;
* отсутствие подробной отчетности по исполнению регламента конкретным обучаемым или группе в целом.

Для решения вышеописанных проблем требуется разработать приложение автоматизированного контроля регламента переговоров тренажерного комплекса.

# **2 Проектирование информационной системы**

## **2.1 Моделирование бизнес-процессов информационной системы**

Рассмотрим существующие методологии моделирования бизнес-процессов, необходимые для проектирования информационных систем.

*Методология IDEF0.*

Неудивительным является то, что внешне диаграммы, построенные с помощью методологии IDEF0, несколько похожи на диаграммы, реализованные с помощью методологии SADT, ведь методология IDEF0 является подмножеством методологии SADT.

IDEF0 также, как и SADT, позволяет простроить функциональную модель описания бизнес-процессов предметной области. Различие заключается в том, что IDEF0 более детально останавливается на взаимодействии объектов между друг другом, нежели на последовательности действий отдельных объектов.

Моделирование происходит по принципу «черного ящика» – известно состояние объекта до какого-либо процесса и после него, но сам процесс скрыт от глаз читателя диаграммы. Более подробно процессы раскрываются в диаграммах декомпозиции.

На диаграмме IDEF0, в отличие от диаграммы потоков данных DFD, отображаются сигналы управления. Описанная методология используется для моделирования всех процессов[2].

*Диаграмма потоков данных DFD.*

DFD в дословном переводе с английского языка является диаграммой потоков данных. DFD акцентирует внимание на передаче данных между процессами, способами ее обработки и хранения. Диаграммы поток данных активно используются при разработке программных продуктов. Еще одной сферой использования данной нотации является анализ системы в рамках документооборота. Методология наглядно представляет:

* места хранения данных;
* процесс обмена документации;
* ошибки организации бизнес-процессов;
* и прочее.

DFD не описывает бизнес-процесс, так как не учитывает времени выполнения этого процесса, а также условий, при которых процесс может пойти по альтернативному «сценарию»[3].

*Методология UML.*

Универсальный язык моделирования (UML) является одной из самых популярных методологий для графического описания бизнес-процесс. Большинство разработчиков программного обеспечения используют его при проектировании и сопровождении своих продуктов.

UML состоит из интегрированного набора диаграмм, разработанных для помощи разработчикам систем и программного обеспечении в определении, визуализации, конструировании и документировании результатов разработки программного обеспечения, а также для бизнес-моделирования[4].

*Методология ARIS.*

Методология ARIS представляет собой набор модулей, использующийся для рассмотрения бизнес-процессов сразу с нескольких точек зрения:

* организационной;
* функциональной;
* обрабатываемых данных;
* структуры бизнес-процессов;
* продуктов и услуг[5].

Из рассмотренных методологий для моделирования бизнес-процессов приложения автоматизированного контроля регламента переговоров была выбрана методология UML, так как диаграммы в данной методологии предоставляют всю необходимую информацию о бизнес-процессах предметной области, достаточную для разработки информационной системы.

Для моделирования бизнес-процессов в UML необходимо построить следующие диаграммы:

* диаграмму вариантов использования системы;
* диаграмму последовательности процесса автоматизированного контроля регламента переговоров;
* диаграмму активностей автоматизированного контроля регламента переговоров;
* диаграмму состояний информационной системы;
* диаграмму классов предметной области.

На рисунке 2.1 представлена диаграмма вариантов использования системы автоматизированного контроля регламента переговоров.

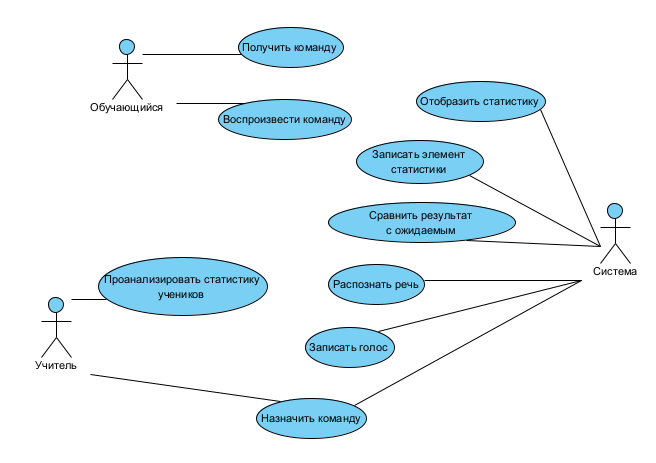


Рисунок 2.1 – Диаграмма вариантов использования приложения автоматизированного контроля регламента переговоров

Как видно из рисунка 2.1, в описываемом бизнес-процессе принимают участие всего три актера (сущности, выполняющие какие-либо действия в описываемом бизнес-процессе). Основной функционал выполняется приложением автоматизированного контроля регламента переговоров, в то время как инструктор занимается лишь анализом полученных статистических данных, а обучаемому не требуется выполнять какие-либо дополнительные действия.

На рисунке 2.2 представлена диаграмма последовательности процесса автоматизированного контроля регламента переговоров. Данная диаграмма более подробно раскрывает диаграмму вариантов использования, показанную на рисунке 2.1. Диаграмма последовательности позволяет увидеть причинно-следственные и временные связи между актерами и действиями, которые они совершают.

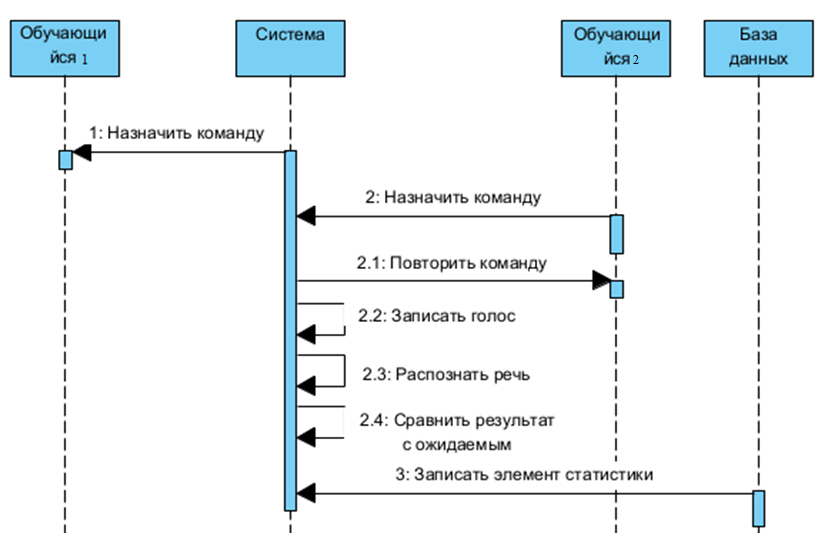


Рисунок 2.2 – Диаграмма последовательности процесса контроля регламента

На рисунке 2.3 изображена диаграмма состояний информационной системы. Данная диаграмма показывает состояния ИС, как они меняются после определенных действий. По состояниям ИС можно отслеживать процесс обучения.

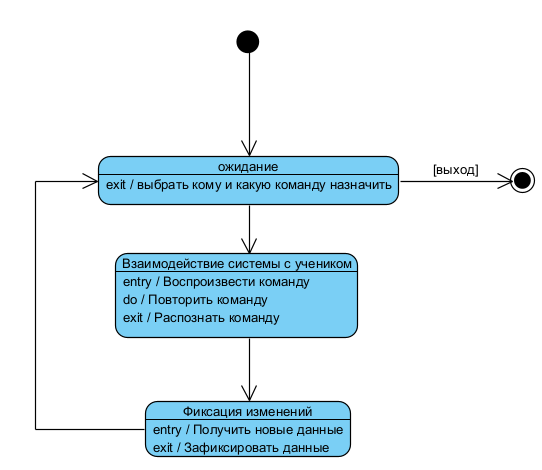


Рисунок 2.3 – Диаграмма состояний ИС

На рисунке 2.4 представлена диаграмма активностей процесса автоматизированного контроля регламента переговоров. Данная диаграмма похожа на диаграмму последовательности, изображенную на рисунке 2.2. Однако данная диаграмма показывает не последовательность действий, а последовательность сообщений, инициализирующих то или иное действие.

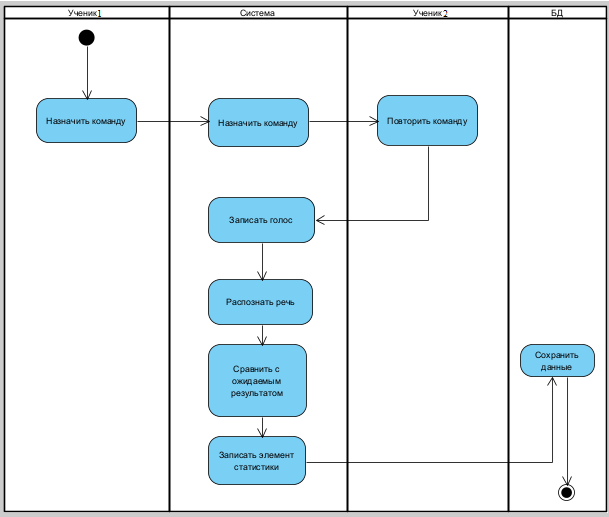


Рисунок 2.4 – Диаграмма активностей процесса контроля регламента

На рисунке 2.5 изображена диаграмма классов предметной области «Система автоматизированного контроля регламента переговоров».

Главной действующей сущностью является класс «Main», он агрегирует модули распознавания и статистики. Его ответственность – совместное управление модулями. Так, например он предоставляет список возможных фраз из модуля статистики, чтобы модуль распознавания мог определить какую фразу пытался произнести пользователь и выполнить оценку корректности произнесенной фразы.

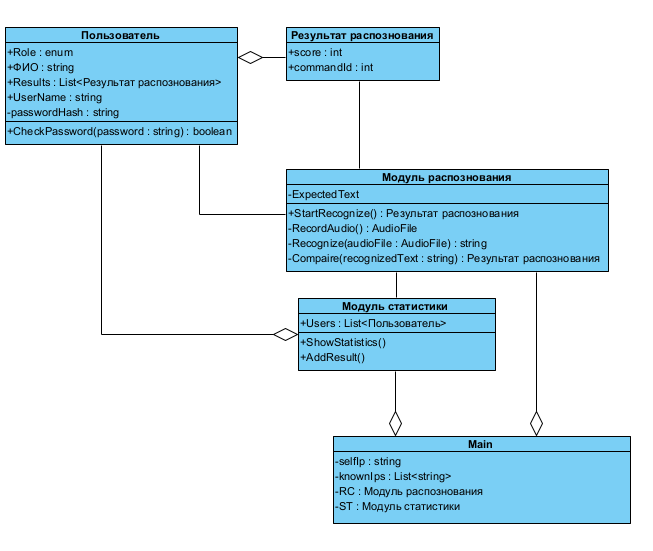


Рисунок 2.5 – Диаграмма классов предметной области

## **2.2 Описание бизнес-процессов до и после внедрения приложения**

До внедрения разрабатываемого приложения контроль регламента был ответственностью инструктора и осуществлялся вручную. Помимо контроля регламента у инструктора так же есть задачи, связанные с моделированием станционных ситуаций, переключение между задачами вызывает неточности в оценке. Данный подход к контролю обладает низкой эффективностью, так как, существует множество допущений со стороны инструктора.

При внедрении приложения автоматизированного контроля регламента переговоров мы сможем получить объективную, не предвзятую оценку его исполнения. Также это позволить повысить скорость массового бучения – система способна контролировать несколько учеников параллельно.

Внедрение разрабатываемого приложения обеспечит автоматизацию описанного процесса контроля регламента переговоров, следовательно, исключит ряд существующих проблем.

## **2.3 Структура ИС и ее средства разработки**

Приложение автоматизированного контроля регламента переговоров применимо для использования на ТК ОПСГ, а также может быть адаптировано под другие задачи, связанные с контролем регламента переговоров. Оно должно выполнять следующие функции:

* разрабатываемое программное обеспечение;
* распознавать и генерировать речь;
* производить сравнение с регламентом;
* редактировать регламент;
* выполнять оценку и вести статистику по пользователям.

Таким образом, приложение автоматизированного контроля регламента переговоров включает в себя следующие модули:

* модуль распознавания речи;
* модуль синтеза речи;
* модуль сетевого взаимодействия;
* модуль статистики.

Техническое задание предполагает разработку приложения автоматизированного контроля регламента переговоров тренажерного комплекса оперативного персонала сортировочной горки на языке программирования C#.

На данный момент C# является довольно распространенным языком программирования, поэтому существует множество сред (IDE) для разработки на нём. Для выбора наиболее подходящей IDE было произведено сравнение наиболее популярных вариантов, был проведен анализ их достоинств и недостатков и соответствия требованиям. Среда разработки должна:

* поддерживать разработку на выбранном языке программирования;
* в полной мере подходить для реализации технической составляющей выпускной квалификационной работы;
* поддерживаться ее разработчиками;
* обладать качественными компилятором, интерпретатором, отладчиком и инструментами автоматизации;
* быть удобной в использовании и обладать исчерпывающим руководством пользователя;
* быть бесплатной.

Рассматриваемые альтернативы: Visual Studio, Project Rider, Eclipse, MonoDevelop [6]. Соответствие альтернатив требованиям наглядно показано в таблице 2.1.

Таблица 2.1 – Сравнение альтернативных IDE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| IDE | Visual Studio | Project Rider | MonoDevelop | Eclipse |
| Бесплатная версия | + | - | + | + |
| Легкость в освоении | - | + | + | - |
| Простота настройки | + | - | + | - |
| Гарантии поддержки | + | + | - | - |
| Широкие возможности отладки и профилирования | + | + | - | - |

Для реализации технической составляющей разрабатываемого приложения IDE должна обладать обширным функционалом, возможностями отладки и профилирования. Поэтому среды разработки MonoDevelop и Eclipse не подходят, т.к. обладают недостаточным функционалом, а также у них отсутствуют гарантии поддержки, что в дальнейшем может стать проблемой.

Среды разработки Eclipse и Visual Studio являются сложными в освоении, что может замедлить разработку. Но сложность использования Visual Studio нивелирует тот факт, что данная среда использовалась в процессе обучения.

Project Rider является платной средой разработки, что исключает данную IDE из списка рассматриваемых сред разработки.

После анализа недостатков и достоинств рассматриваемых сред разработки были сделаны следующие выводы:

* Eclipse сложна в освоении;
* Project Rider не имеет бесплатной версии;
* MonoDevelop не обладает достаточным для выполнения работы функционалом.

Самой подходящей средой разработки для написания технической составляющей выпускной квалификационной работы, учитывая наличия лицензии, является Visual Studio.

## **2.4 Описание структуры базы данных**

После проведения анализа технического задания и смоделированных бизнес-процессов можно сделать вывод, что система автоматизированного контроля регламента переговоров ТК ОПСГ нуждается в разработке базы данных.

На основе анализа предметной области можно спроектировать следующие сущности: «Роль», «Команда», «Пользователь», «Оценка». В таблицах 2.1-2.4 представлено описание каждой сущности, указанием ключевых параметров.

Стоит обратить особое внимание на сущности «Команда» и «Оценка», они являются ключевыми в решении основной задачи.

Таблица 2.1 – Описание сущности «Роль»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип данных | Обязательное | PK | Описание |
| Id | int | + | + | Уникальный идентификатор |
| Name | Varchar(50) | + | - | Наименование роли |

Таблица 2.2 – Описание сущности «Команда»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип данных | Обязательное | PK | Описание |
| Id | int | + | + | Уникальный идентификатор |
| From | int | + | - | Ссылка на роль, которая воспроизводит фразу |
| To | int | + | - | Ссылка на роль, которая принимает фразу |
| Text | varchar(max) | + | - | Текст фразы |
| IsAnswer | int | - | - |  |

Таблица 2.3 – Описание сущности «Пользователь»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип данных | Обязательное | PK | Описание |
| Id | int | + | + | Уникальный идентификатор |
| FIO | varchar(200) | + | - | Наименование |
| UserName | Varchar(50) | + | - |  |
| Passwordhash | Varchar(20) | + | - | Зашифрованный пароль |

Таблица 2.4 – Описание сущности «Оценка»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле | Тип данных | Обязательное | PK | Описание |
| Id | int | + | + | Уникальный идентификатор |
| UserId | int | + | - | Ссылка на пользователя |
| CommandId | int | + | - | Ссылка на команду |
| Score | double | + | - | Оценка |
| Created | date | + | - | Дата выставления оценки |

На рисунке 2.6 представлена ER-диаграмма базы данных рассматриваемой информационной системы. ER-диаграмма позволит нам более наглядно увидеть связи сущностей, более детально оценить их взаимодействие и причины выделения. Так же по диаграмме можно удостовериться в необходимости и достаточности выделенных таблиц.

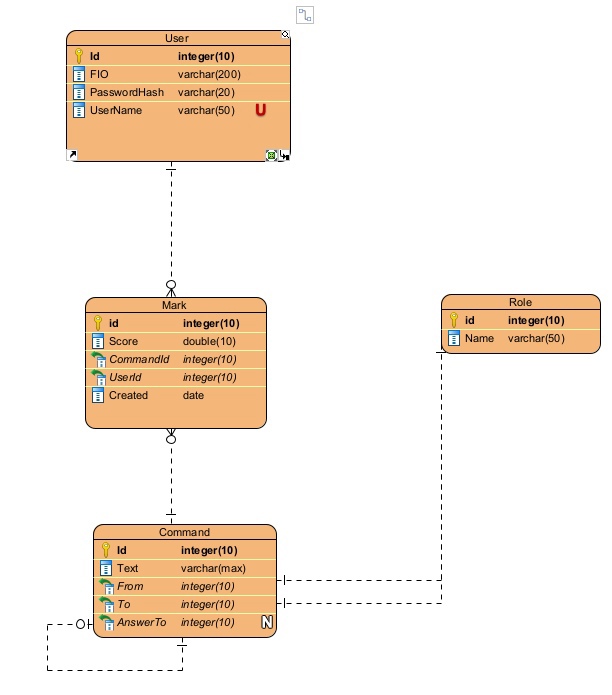


Рисунок 2.6 – ER-диаграмма информационной системы

## **2.5 Обзор технологий распознавания речи**

Для распознавания и синтеза речи используется искусственный интеллект. Существует 3 подхода к внедрению ИИ в систему:

* написание собственной нейросети и её дальнейшее обучение. Этот метод обеспечит наилучший контроль и совместимость, но также он является трудоёмким и затратным по времени;
* использование готовой нейросети и обученной модели. Это даст наилучшую скорость внедрения, но возможны некоторые ограничения в расширении модели;
* использование внешнего сервиса. Такой вариант удобен тем, что модель самостоятельно развивается поставщиком. Также нет необходимости тратить ресурсы компьютера на распознавание. Но при всех этих преимуществах есть и существенные ограничения: отсутствие контроля над работой системы и необходимость наличия интернет-соединения.

Из рассмотренных вариантов был выбран второй – использовать готовую нейросеть и обученную модель. Данный вариант является наилучшим по совокупности параметров.

Для распознавания речи был выбран opensource проект VOSK от Alpha Cephei. Главными преимуществами данного решения является возможность корректирования модели. Также это проект с открытым исходным кодом, что оставляет возможность адаптации алгоритма его работы под поставленные задачи[7].

Были также рассмотрены другие достойные внимания аналоги:

* CMU sphinx – имеет все те же преимущества. Проект имеет более долгую историю в сравнении с VOSK, следовательно внушает больше доверия в плане гарантий поддержки. Так же модель обучена на большем объеме данных. Однако данное решение не имеет готовой реализации для интеграции с .NET, что делает его применением неоправданно сложным[8];
* Microsoft Speech Application Interface – готовое решение, нативно интегрируется с .NET. Из недостатков: полное отсутствие контроля над работой и невозможность самостоятельного изменения модели[9].

Как можно увидеть из приведённых альтернатив, VOSK является наиболее удачным выбором.

Для синтезации речи используется стандартное для .NET решение от Microsoft Speech Synthesis, так как качество синтеза речи не критично на данном этапе развития системы, а важно его наличие как функции. В дальнейшем планируется заменить это решение на более развитое, имеющее возможности адаптации.

## **2.6 Перспективы развития разработанной системы**

В текущей реализации весь функционал расположен в пределах одного приложения. В дальнейшем планируется разделение его на две части. Вкладка «распознавание» войдет в клиентское приложение для обучающегося, а вкладки «конструктор» и «статистика» - в серверное приложение для инструктора.

Ожидается следующий принцип работы разделенных приложений:

* у сервера всегда открыт один порт, к примеру 1337, для авторизации пользователей;
* после авторизации сервер фиксирует нового пользователя в списке активных пользователей;
* клиент отсылает информацию о своей роли и получает в ответ номер порта для общения с сервером. В этот момент сервер создает отдельный поток для прослушивания указанного порта;
* далее клиент общается с сервером только по выделенному порту. Через него клиент получает список активных пользователей, которым может назначать команды. Отправляет на него запись голоса и результаты распознавания речи;
* также и у клиента всегда открыт один порт, для того чтобы сервер мог проверить активность клиента и передать ему назначенную команду.

В дальнейшем планируется развитие модуля распознавания речи. Будет совершен переход от прямого сравнения текста к семантическому анализу. Это позволит повысить точность распознавания и отказаться от ручного выбора роли назначения, так как будет возможно выявление её из контекста.

Еще одной функцией для реализации в перспективе является имитация ответа от виртуального работника. Функционал синтеза речи реализован, но пока его применение не оправданно. Для этого нужно перейти к клиент-серверной модели и реализовать семантический анализ речи, чтобы было возможно генерировать шаблонный ответ, основанный на полученном сообщении, для имитации ответа человека. В текущей реализации невозможно преобразовать вопрос в ответ, в ситуации изначально некорректной команды.

## **2.7 Вывод о проектировании системы**

В ходе проектирования системы ТК ОПСГ было выполнено следующее:

1. были смоделированы бизнес-процессы системы в нотации UML, а именно построены следующие диаграммы;
2. диаграмма вариантов использования системы;
3. диаграмма последовательности процесса контроля регламента переговоров;
4. диаграмма активностей процесса;
5. диаграмма состояний ИС;
6. диаграмма классов предметной области;
7. описаны бизнес-процессы до и после внедрения системы;
8. описаны структура ИС и средства ее разработки;
   1. модули, входящие в приложение системы автоматизированного контроля регламента переговоров ТК ОПСГ;
   2. выбраны язык и среда разработки с помощью которых будет реализовано приложение;
9. спроектирована база данных для рассматриваемой информационной системы;
10. проведен обзор технологий для основной функции системы;
11. указаны направления развития разрабатываемой системы в перспективе.

# **3 Реализация системы**

## **3.1 Интерфейс обучаемого**

Обучаемому предлагается интерфейс, представленный на рисунке 3.1. К нему относится только первая вкладка приложения.

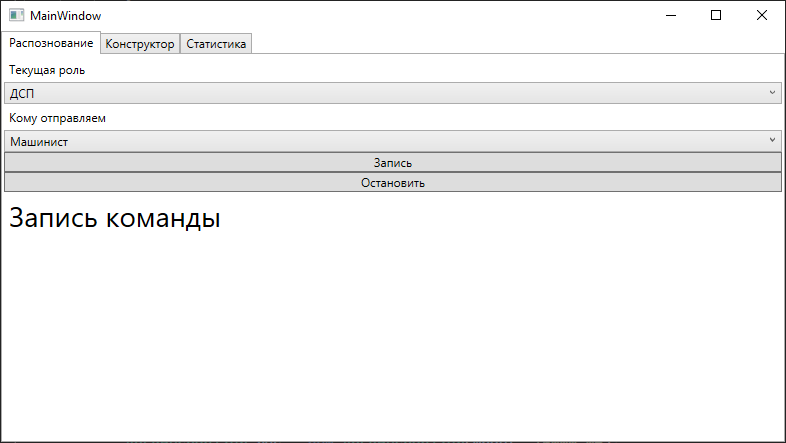


Рисунок 3.1 – Интерфейс обучаемого

Ему доступен выбор текущей роли, то есть та роль, которую он сейчас исполняет и роль назначения «Кому отправляем». Роли выбираются из списка доступных, который загружается из базы данных.

Приложение поддерживает расширение списка ролей, это будет продемонстрированно в следующем пункте, посвящённому интерфейсу инструктора. Дальнейшие развитие системы предполагает наличие учетных записей, и у обучаемого останется лишь выбор назначения.

Для выполнения основной задачи этой части программы добавлены две кнопки: «Запись» и «Остановить». При нажатии кнопки «Запись» происходит запись голоса в файл. Когда фраза закончена, нажимается копка «Остановить», останавливающая запись. После этого аудио файл преобразуется в текст, по полученному тексту определяется исходная фраза. Метод сравнения фраз показан на рисунках 3.2 и 3.3.

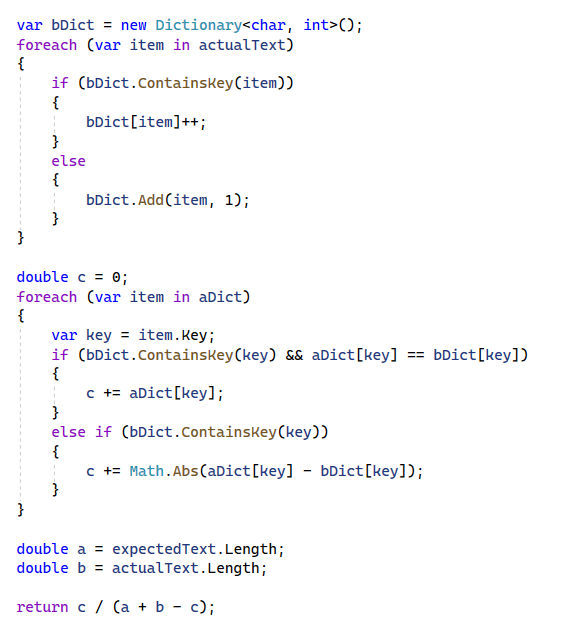


Рисунок 3.2 – Сравнение фраз

Также для определения, какой из вариантов записи происходит в данный момент – ответ или команда, добавлен соответствующий текст, расположенный ниже всех элементов управления. После записи фразы текст меняется на соответствующий новому состоянию, рисунок 3.4.

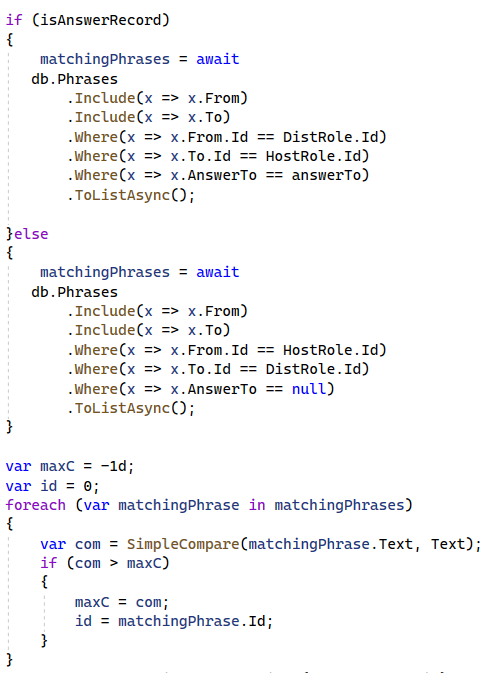


Рисунок 3.3 – Определение наиболее подходящей фразы



Рисунок 3.4 – Состояние ответной фразы.

## **3.2 Интерфейс инструктора**

Для инструктора предполагаются возможности редактирования данных и инструменты статистики. Для этого созданы соответствующие вкладки: «Конструктор» и «Статистика».

Интерфейс конструктора представлен на рисунке 3.5. Он состоит из двух частей: левая половина предназначена для добавления ролей, правая – для создания фраз.

При заполнении фраз используются ранее созданные роли и опционально другие фразы. Это сделано для того, чтобы была возможность создавать фразы, являющееся ответами на другие фразы. Такой подход позволяет нам не создавать дополнительных таблиц в базе данных, а использовать туже самую таблицу фраз.

Пример заполонённых полей конструктора фраз на рисунке 3.6. Так же на этом рисунке видно, что после добавления фразы 3 в поле «AnserTo» появился номер фразы, для которой предназначен ответ. При записи голоса для ответа на фразу по этому полю будут отбираться возможные фразы. Это сокращает время поиска вероятной фразы.

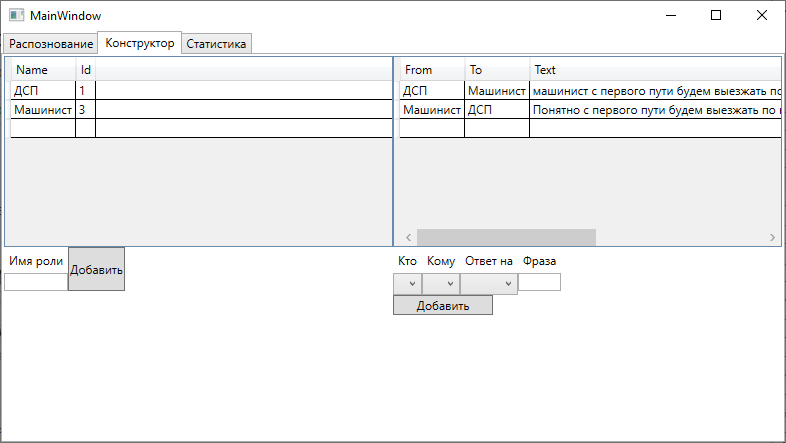


Рисунок 3.5 – Вкладка «Конструктор»

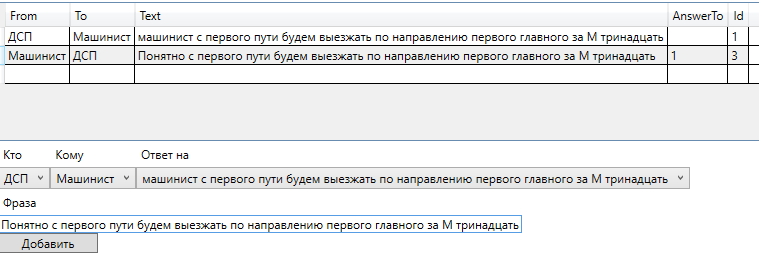


Рисунок 3.6 –Конструктор фраз

Далее рассмотрим вкладку статистики. Её внешний вид представлен на рисунке 3.7.

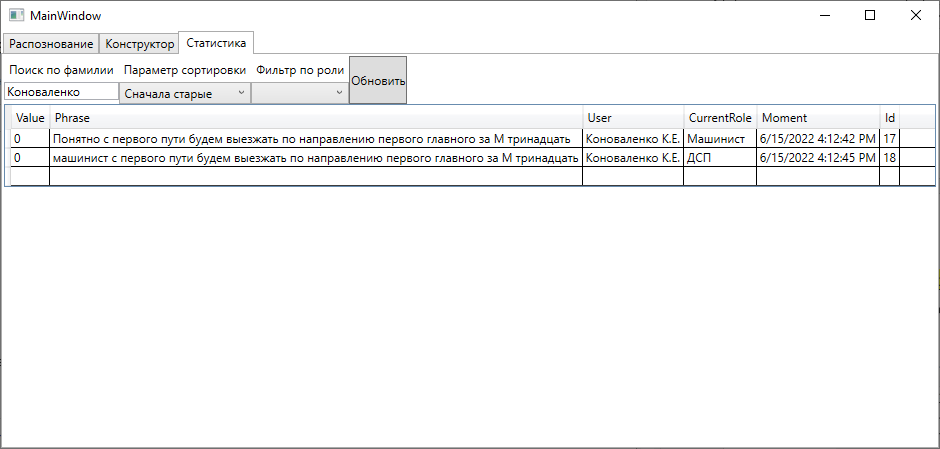


Рисунок 3.7 –Интерфейс статистики

В таблице можно увидеть результаты прохождения задания пользователем. Сам результат находится в колонке «Value», далее указана исполняемая фраза, пользователь, роль которую он исполнял в этой фразе, момент времени в формате мм/дд/гггг чч:мм:сс. Для более удобного анализа полученных результатов были добавлены фильтры и сортировки:

* фильтр по фамилии обучаемого;
* фильтр по роли, исполняемому обучаемым;
* сортировка по некоторым параметрам.

Параметры сортировки представлены на рисунке 3.8.

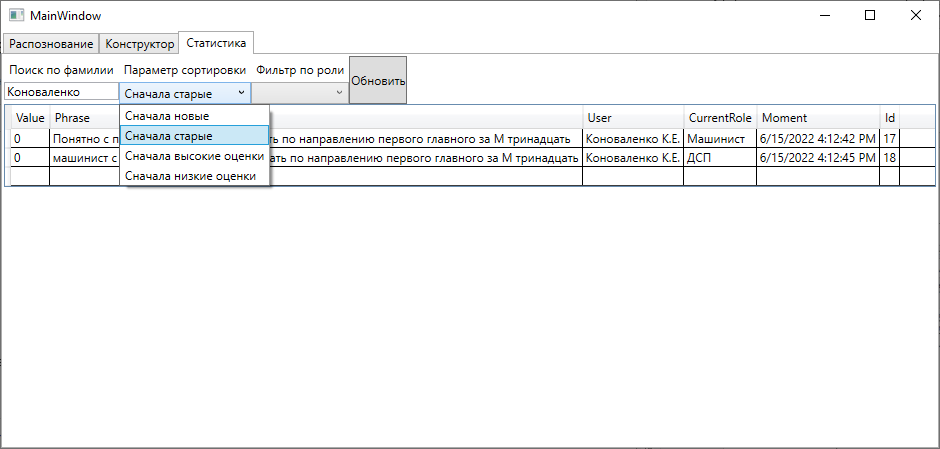


Рисунок 3.8 – Сортировки

## **3.3 Интеграция базы данных**

Работать с базой данных будем по принципу Code first, через Entity Framework.

Согласно этому принципу, мы сначала создаем сущности, необходимые для реализации логики приложения, и уже по ним строим архитектуру базы данных[10].

Так были созданы следующие сущности: роль, фраза, пользователь, оценка. Они показаны на рисунках 3.10-3.13. Все они расширяют базовый класс «Сущность», код которого представлен на рисунке 3.9.

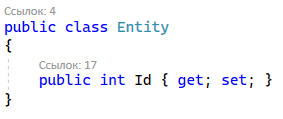


Рисунок 3.9 – Сущность

Введение базового класса призвано упросить задачу создания новых сущностей, так как выделяет одно общее, присущие всем возможным сущностям, свойство в родительский класс.

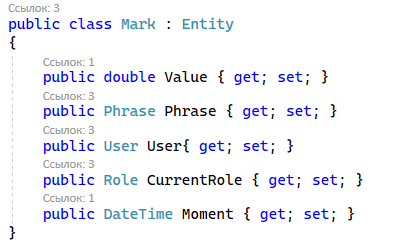


Рисунок 3.10 – Оценка

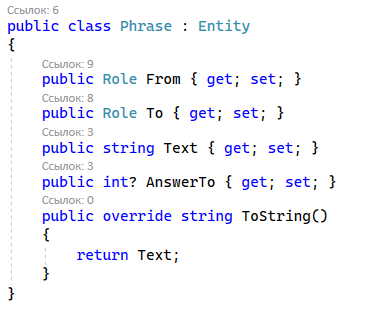


Рисунок 3.11 – Фраза

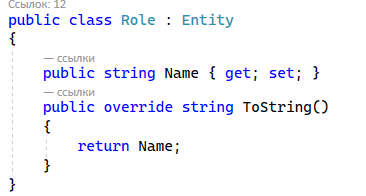


Рисунок 3.12 – Роль

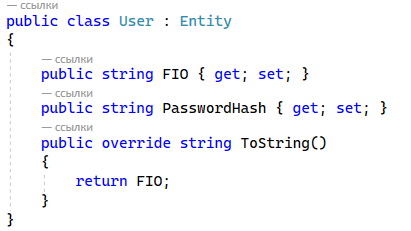


Рисунок 3.13 – Пользователь

Для работы с БД необходимо создать контекст данных, рисунок 3.14.



Рисунок 3.14 – Контекст данных

Для того, чтобы можно было безопасно менять структуру БД, используется механизм миграций БД. Он позволяет иметь историю изменений структуры БД, что позволяет легко поднимать или понижать версию базы данных. Для создания миграций используется инструмент EF Core Tools, позволяющий автоматизировать процесс создания скриптов для обновления структуры данных. Этот инструмент самостоятельно определяет разницу между текущей версией БД и той, что соответствует коду, и генерирует скрипт для устранения этой разницы.

Для создания миграции используется команда *Add-Migration*. Для применения миграций команда *Update-Database*. Пример выполнения данных команд представлен на рисунках 3.15-3.16.

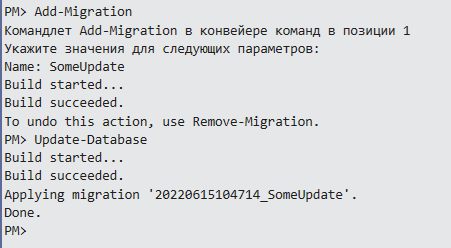


Рисунок 3.15 – создание миграции

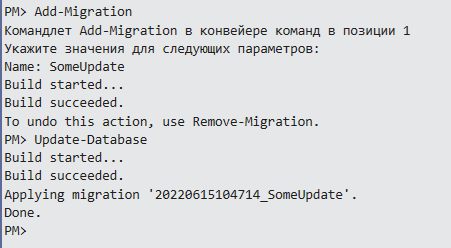


Рисунок 3.16 – применение миграций

## **3.4 Руководство пользователя**

Система автоматизированного контроля регламента переговоров состоит из двух приложений:

* серверное приложение, которое ведет статистику и дает возможность управлять данными. Так же сервер сообщает клиентам о наличии других клиентов и пересылает записи голоса между ними;
* клиентское приложение, в нем можно выбрать исполняемую роль и исполнять регламент. Эта часть системы выполняет генерацию и распознавание речи, после чего отсылает информацию на сервер;
* служба для выполнения списка команд на АРМ.

Помимо указанных служб на сервер необходимо установить СУБД Microsoft SQL Server.

После установки системы автоматического контроля регламента переговоров необходимо проверить работоспособность системы. Для этого на сервере необходимо зайти в конструктор и попробовать добавить первую роль, если все успешно, то серверная часть настроена верно. После этого необходимо запустить клиентскую часть и убедиться, что в списке выбора роли доступна добавленная роль.

Можно приступать к работе.

Сервер работает по следующему алгоритму:

* ожидание подключений;
* добавление нового клиента в список;
* клиент отправляет сообщение;
* предать сообщение, тому кому оно предназначается;
* записать в базу данных оценку;

Алгоритм работы клиента:

* подключиться к серверу;
* авторизация;
* выполнить запись речи;
* выполнить распознавание;
* отправить запись и результаты на сервер;
* ожидание ответа от сервера;
* после получения ответа, переходит в следующий алгоритм.

При получении клиентом сообщений его поведение немного меняется и начинает работу по следующему алгоритму:

* заблокировать весь интерфейс кроме кнопки записи;
* выполнить генерацию речи;
* выполнить запись речи;
* выполнить распознавание;
* отправить запись и результаты на сервер;
* ожидание ответа от сервера;
* после ответа, переходит в предыдущий алгоритм на пункт 3.

Разработанная система обладает полной отказоустойчивостью и не завершается аварийно в случае возникновения какой-либо ошибки. Несмотря на это, при обновлении тренажерного комплекса могут возникать ошибки, вызванные различными ситуациями:

* клиент может не подключаться к серверу из-за сетевых ошибок, для решения проблемы следует попробовать перезапустить сервер и клиенты. Если проблема не решится, перезапустить коммутаторы. Если это тоже не приведет к решению проблемы следует обратиться к системному администратору;
* другое приложение может использовать микрофон в монопольном режиме, для решения данной проблемы необходимо либо закрыть приложение, либо установить дополнительный микрофон.

Разработанная система обладает интуитивно понятным пользовательским интерфейсом, можно дать лишь совет в каком порядке следует начать изучение:

* перейти на вкладку конструктор;
* добавить роли и фразы;
* перейти на клиентскую часть;
* выбрать параметры, соответствующие добавленной фразе;
* воспроизвести фразу;
* теперь в серверной части на вкладке статистика можно увидеть результат выполнения команды;
* в статистике стоит обратить внимание на колонку «Value», в ней содержится процент корректности формулировки.

В случае возникновения ошибок работы системы пользователи могут связаться с командой разработчиков тренажерного комплекса.

# **ЗАКЛЮЧЕНИЕ**

В рамках выпускной квалификационной работы было разработано приложение системы автоматизированного контроля регламента для ТК ОПСГ. Для достижения поставленной цели были выполнены следующие задачи:

* проведен анализ предметной области;
* изучена структура регламента переговоров;
* проведен анализ требований к системе;
* выбрана технология распознавания и синтеза речи;
* выбран способ представления отчетности;
* разработано приложение автоматизированного контроля регламента переговоров с учетом всех требований и особенностей предметной области;
* проверено соответствие разработанного приложения всем требованиям;
* разработанное приложение было протестировано.

В дальнейшем планируется добавить в приложение следующий функционал:

* семантический анализ сообщения;
* выбор назначения по контексту;
* идентификация голоса.

После этого система будет в полной мере готова к внедрению в ТК ОПСГ.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Факультет «Бизнес-информатика» - Абитуриенту / Официальный сайт ВУЗа «СГУПС» URL: <https://www.stu.ru/education/index.php?page=674> (дата обращения: 11.04.2022).
2. Методология IDEF0 / Электронный образовательный ресурс ITteach.RU; [Электронный ресурс]. – URL: <https://itteach.ru/bpwin/metodologiya-idef0> (дата обращения 12.04.2022).
3. Что такое диаграмма DFD и как ее создать? / Сервис для интеллектуального построения диаграмм Lucidchart; [Электронный ресурс]. – URL: <https://www.lucidchart.com/pages/ru/диаграмма-dfd> (дата обращения 12.04.2022).
4. Что такое унифицированный язык моделирования? / Сервис для интеллектуального построения диаграмм Lucidchart; [Электронный ресурс]. – URL: <https://www.lucidchart.com/pages/ru/uml> (дата обращения 12.04.2022).
5. Методология ARIS / Сайт группы компаний Концерн R-Про; [Электронный ресурс]. – URL: <https://www.r-p-c.ru/page/metodologiya-aris.html> (дата обращения 12.04.2022).
6. Орлов Е. Лучшие IDE для C#-разработчика / Е. Орлов – сайт международного IT-колледжа DevEducation; [Электронный ресурс]. – URL: <https://spb.deveducation.com/blog/luchshie-ide-dlya-c-razrabotchika/> (дата обращения 05.05.2022).
7. Vosk-api: Репозиторий пользователя alphacep на сайте GitHub; [Электронный ресурс]. – URL: https://github.com/alphacep/vosk-api (дата обращения 13.05.2022).
8. pocketsphinx: Репозиторий пользователя cmusphinx на сайте GitHub; [Электронный ресурс]. – URL: https://github.com/cmusphinx/pocketsphinx (дата обращения 13.05.2022).
9. Распознавание речи – сайт сайт Документации Microsoft; [Электронный ресурс]. – URL: https://docs.microsoft.com/ru-ru/windows/apps/design/input/speech-recognition (дата обращения 13.05.2022).
10. Справочник по инструментам Entity Framework Core – сайт Документации Microsoft; [Электронный ресурс]. – URL: <https://docs.microsoft.com/ru-ru/ef/core/cli/> (дата обращения 14.05.2022).