

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Образовательная программа «Программная инженерия»

СОГЛАСОВАНО
Научный руководитель,
заместитель директора
по работе с НИУ ВШЭ, 1С
_____ Н. Ю. Старичков
«___» _____ 2023 г.

УТВЕРЖДАЮ
Академический руководитель
образовательной программы
«Программная инженерия»
профессор департамента программной
инженерии, канд. техн. наук
_____ В. В. Шилов
«___» _____ 2023 г.

ГОЛОСОВАЯ КЛАВИАТУРА

Пояснительная записка

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.02.02-01 81 01-1-ЛУ

Исполнитель
студент группы БПИ202
_____ / Е. В. Галюта /
«___» _____ 2023 г.

Подп. и дата	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл	

Москва 2023

УТВЕРЖДЕН
RU.17701729.02.02-01 81 01-1-ЛУ

ГОЛОСОВАЯ КЛАВИАТУРА

Пояснительная записка

RU.17701729.02.02-01 81 01-1

Листов 30

<i>Подп. и дата</i>	
<i>Инв. № дубл.</i>	
<i>Взам. инв. №</i>	
<i>Подп. и дата</i>	
<i>Инв. № подл</i>	

Москва 2023

СОДЕРЖАНИЕ

1 ВВЕДЕНИЕ.....	4
1.1 Наименование программы.....	4
1.2 Краткая характеристика области применения программы.....	4
2 НАЗНАЧЕНИЕ РАЗРАБОТКИ.....	5
2.1 Функциональное назначение.....	5
2.2 Эксплуатационное назначение.....	5
3 ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ.....	6
3.1 Постановка задачи для разработки программы.....	6
3.2 Описание и обоснование алгоритма функционирования программы.....	6
3.2.1 Описание алгоритма функционирования программы.....	6
3.2.2 Обоснование выбора алгоритмов.....	7
3.3 Описание и обоснование метода организации входных и выходных данных	8
3.3.1 Описание организации входных и выходных данных.....	8
3.3.2 Обоснование выбора метода организации входных и выходных данных.....	8
3.4 Описание и обоснование выбора технических и программных средств.....	8
3.4.1 Состав технических и программных средств.....	8
3.4.2 Обоснование выбора технических и программных средств.....	9
4 ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ.....	10
4.1 Предполагаемая потребность.....	10
4.2 Ориентировочная экономическая эффективность.....	10
4.3 Технические преимущества по сравнению с аналогами.....	10
5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ.....	11
ПРИЛОЖЕНИЕ 1 ТЕРМИНОЛОГИЯ.....	12
ПРИЛОЖЕНИЕ 2 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ	14
ПРИЛОЖЕНИЕ 3 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ, МЕТОДОВ И СВОЙСТВ.....	17
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ.....	31

Изм.	Лист	№ докум.	Подп.	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

1 ВВЕДЕНИЕ

1.1 Наименование программы

Наименование программы: «Настольное приложение Голосовая клавиатура».

Наименование программы на английском языке: «Desktop application Voice keyboard».

Краткое наименование программы: «Voice Keyboard app».

1.2 Краткая характеристика области применения программы

Программа является настольным приложением. Приложение предназначено для повышения удобства при работе пользователя с персональным компьютером. Также может быть использовано для повышения производительности и эффективности в работе с различными сочетаниями клавиш.

В интерфейсе приложения пользователь может добавить сочетание клавиш и назначить одну или несколько голосовых команд для его выполнения. Также возможно удалить, просмотреть все сочетания и импортировать или экспортировать настройки.

Программа поставляется для следующих операционных систем: MacOS [14], Windows [19].

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

2 НАЗНАЧЕНИЕ РАЗРАБОТКИ

2.1 Функциональное назначение

Программа предназначена для предоставления интерфейса взаимодействия пользователя с локальным сервером.

Приложение позволяет пользователю добавлять новые сочетания клавиш и присваивать им команды, удалять существующие, а также импортировать и экспортировать настройки приложения. Также есть возможность временно отключить распознавание команд для удобства пользователя.

2.2 Эксплуатационное назначение

Программа должна эксплуатироваться в качестве интерфейсной части приложения «Голосовая клавиатура». Программа предоставляет пользовательский интерфейс для взаимодействия с функционалом локального сервера приложения «Голосовая клавиатура». Другие виды использования представленной программы не рассматриваются.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

3 ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

3.1 Постановка задачи для разработки программы

Приложение должно обеспечивать возможность выполнения следующих функций:

1. Добавление команды и сочетания клавиш, которое активируется данной командой.
2. Редактирование команд, назначенных сочетанию клавиш.
3. Редактирование сочетания клавиш, назначенного командам.
4. Удаление команды и назначенного ей сочетанию клавиш.
5. Просмотр существующих команд и сочетаний клавиш.
6. Экспорт настроек команд и сочетаний клавиш в файл в формате JSON [16].
7. Импорт словаря команд и сочетаний клавиш из файла формата JSON [16], созданного данным приложением.
8. Вывод всплывающих подсказок о невозможности сопоставить распознанный текст голосовой команде.
9. Приостановка распознавания приложением команд.

Перечисленные требования должны поддерживать работу с речью на русском языке, а также корректно работать в двух приложениях, для MacOS [17] и Windows [19] соответственно.

3.2 Описание и обоснование алгоритма функционирования программы

3.2.1 Описание алгоритма функционирования программы

Готовое приложение может быть загружено из релизов в репозитории GitHub [15] или может быть самостоятельно собрано из открытых исходников. Приложение предоставляет комфортный для пользователя интерфейс, максимально упрощающий взаимодействие с приложением и сервером соответственно. Для этого были разработаны два различных приложения для MacOS [17] и Windows [19] с одинаковым набором функций.

Интерфейс удовлетворяет следующим требованиям:

- 1) Пользовательский интерфейс должен быть реализован в виде приложения в строке меню для MacOS [17], системном трее для Windows [19].
- 2) Данный программный продукт должен иметь интерфейс стилизованный под MacOS [17] или Windows [19].
- 3) Дизайн приложения должен иметь понятный для пользователя интерфейс.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- 4) Элементы интерфейса должны быть выполнены в одном стиле с использованием минимального количества цветов и видов шрифтов.
- 5) Рабочие элементы должны отличаться от статических.
- 6) Рабочие элементы должны иметь названия и/или иконки, позволяющие однозначно имитировать выполняемые ими функции.
- 7) Язык интерфейса – русский. Смена языкового пакета не предусмотрена.
- 8) Сигнализация об ошибках должна стилистически (цветом) выделяться и сопровождаться подсказками о дальнейших действиях.
- 9) Интерфейс должен быть спроектирован с учетом различных экранных разрешений.
- 10) Для элементов, размещенных вне видимой области элементов управления, должна быть предусмотрена возможность прокрутки.
- 11) Пространственная компоновка форм и размерные характеристики кнопок должны обеспечивать быстрый доступ к функциям приложения.

Интерфейс реализован с помощью различных view, взаимодействующих между собой и имеющих элементы для взаимодействия пользователя с приложением таких как кнопки, поля для ввода и другое.

Приложение представляет из себя окно, которое сворачивается в строку меню на MacOS [17] и в системный трей на Windows [19], что позволяет пользователю в любой момент обратиться к интерфейсу при этом не нагружая строку с вкладками дополнительным приложением.

При запуске приложения происходит запуск сервера посредством исполнения bash команды. Приложение не активируется пока не будет установлено соединение с сервером, то есть приложение ждет пока сервер полностью запустится и будет готов к работе. Далее пользователь попадает на основную страницу с выбором различных действий и с неё попадает на другие view, в зависимости от выбора функционала.

Взаимодействие с сервером осуществляется посредством gRPC, настроенного как на стороне сервера, так и на стороне клиента. Реализованы сервисы для запросов к серверу и обработки ответов. Каждый ответ, приходящий от сервера, обрабатывается, что исключает возникновение непредвиденных ошибок при взаимодействии с сервером.

3.2.2 Обоснование выбора алгоритмов

Было разработано два аналогичных по функционалу приложения под такие операционные системы как MacOS [17] и Windows [19]. Это было сделано для достижения

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

максимального удобства и эстетически приятного интерфейса взаимодействия пользователя с системой. Все элементы в обоих приложениях соблюдают общий стиль конкретной операционной системы, а также используются уникальные части интерфейса MacOS [17] и Windows [19].

Также два приложения были разработаны для увеличения потенциального количества пользователей.

3.3 Описание и обоснование метода организации входных и выходных данных

3.3.1 Описание организации входных и выходных данных

В виде входных данных будут представлены: нажатия клавиш, а также различные варианты взаимодействия пользователя с интерфейсом (например: нажатие, прокрутка и т. д.).

В качестве выходных данных предоставлено изменение интерфейса приложения в зависимости от действий пользователя. При невозможности сохранить/удалить команду, а также импортировать/экспортировать настройки приложения, пользователь получит всплывающее уведомление с ошибкой. Также уведомления об ошибках будут приходить в случае невозможности получить все команды с сервера или же приостановить распознавание команд приложением. Любое взаимодействие с сервером обрабатывается и в случае ошибок пользователь будет проинформирован о произошедшем.

3.3.2 Обоснование выбора метода организации входных и выходных данных

Выбранные методы организации данных обусловлены удобством и корректностью взаимодействия пользователя с разрабатываемым приложением. Основной акцент делается на комфортное использование приложения «Голосовая клавиатура». Иные способы взаимодействия будут не так удобны для начинающих пользователей персональных компьютеров и лишь затруднят использование приложения.

3.4 Описание и обоснование выбора технических и программных средств

3.4.1 Состав технических и программных средств

Для работы разрабатываемого приложения необходимо устройство с операционной системой Windows [19] 10 и выше или MacOS [17] 13.0 и выше, характеристики которого позволяют работать и пользоваться всеми основными функциями данного приложения.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Для надежной и бесперебойной работы программы требуется следующий состав технических средств:

1. Оперативная память не меньше 2 Гб.
2. Свободное место на встроенной памяти не меньше 1 Гб.
3. Монитор с разрешением не менее 1024 на 768 пикселей.
4. Клавиатура
5. Компьютерная мышь

Данные условия обусловлены функциональностью разрабатываемого приложения, примерным размером исполняемых файлов для каждой операционной системы, а также минимальными системными требованиями указанных операционных систем.

3.4.2 Обоснование выбора технических и программных средств

Вышеуказанные технические требования – это минимальные требования для запуска разрабатываемого приложения. Приложение не нуждается в больших технических средствах, оно может быть использовано на довольно маломощных устройствах.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

4 ОЖИДАЕМЫЕ ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

4.1 Предполагаемая потребность

«Voice Keyboard» - приложение для работы с сочетаниями клавиш посредством голосовых команд. Предполагаемый конечный пользователь – люди, ежедневно использующие персональный компьютер для работы в программах с наличием сочетаний клавиш, а также желающие упростить использование различных сочетаний в работе.

4.2 Ориентировочная экономическая эффективность

В рамках данной работы расчет экономической эффективности не предусмотрен.

4.3 Технические преимущества по сравнению с аналогами

Программа предназначена исключительно для использования в приложении “Голосовая клавиатура” и предоставления интерфейса для взаимодействия пользователя с сервером. Иных образцов или аналогов с необходимым для данного приложения функционалом выявлено не было.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

5. ИСТОЧНИКИ, ИСПОЛЬЗОВАННЫЕ ПРИ РАЗРАБОТКЕ

- 1) ГОСТ 19.101-77 Виды программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 2) ГОСТ 19.102-77 Стадии разработки. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 3) ГОСТ 19.103-77 Обозначения программ и программных документов. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 4) ГОСТ 19.104-78 Основные надписи. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001;
- 5) ГОСТ 19.105-78 Общие требования к программным документам. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001;
- 6) ГОСТ 19.106-78 Требования к программным документам, выполненным печатным способом. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 7) ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 8) ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 9) ГОСТ 19.401-78 Текст программы. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 10) ГОСТ 19.404-79 Пояснительная записка. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 11) ГОСТ 19.505-79 Руководство оператора. Требования к содержанию и оформлению. //Единая система программной документации. – М.: ИПК Издательство стандартов, 2001.
- 12) ГОСТ 19.602-78. ЕСПД. Правила дублирования, учета и хранения программных документов, выполненных печатным способом. — Москва: Стандартинформ, 2005
- 13) ГОСТ 19.603-78 Общие правила внесения изменений. // Единая система программной документации. – М.: ИПК Издательство стандартов, 2001;

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

- 14) C# (язык программирования). [Электронный ресурс] – URL: <https://dotnet.microsoft.com/en-us/languages/csharp> режим доступа: свободный (дата обращения: 15.01.2023).
- 15) GitHub. [Электронный ресурс] – URL: <https://github.com/> режим доступа: свободный (дата обращения: 25.01.2023).
- 16) JSON. [Электронный ресурс] – URL: <https://www.json.org/json-en.html> режим доступа: свободный (дата обращения: 12.01.2023).
- 17) MacOS (операционная система). [Электронный ресурс] – URL: <https://developer.apple.com/macos/> режим доступа: свободный (дата обращения: 15.12.2022).
- 18) Swift (язык программирования). [Электронный ресурс] – URL: <https://www.apple.com/swift/> режим доступа: свободный (дата обращения: 13.01.2023).
- 19) Windows (операционная система). [Электронный ресурс] – URL: <https://support.microsoft.com/ru-ru/windows> режим доступа: свободный (дата обращения: 15.12.2022).

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 1 ТЕРМИНОЛОГИЯ

Персональный компьютер (ПК) — однопользовательская ЭВМ, имеющая эксплуатационные характеристики бытового прибора и универсальные функциональные возможности.

Git-репозитории — каталог файловой системы, в котором находятся файлы конфигурации репозитория, файлы журналов, хранящие операции, выполняемые над репозиторием, индекс, описывающий расположение файлов, и хранилище, содержащее собственно файлы.

Сочетание клавиш — разновидность интерфейса взаимодействия с вычислительным устройством, представляющая собой нажатие кнопки/клавиши на клавиатуре, которому назначено некое действие - команды, исполняемые данной системой.

Импорт — добавление данных, вставка данных из внешних источников в текущий файл/документ/базу данных.

Экспорт — сохранение документа программы не в родном для программы формате файла, а в формате другой программы.

Пиксель — это минимальный и неделимый элемент (точка), из которого состоит изображение на экране монитора.

Трей Windows [19] – это область уведомлений на панели задач, в нижнем правом углу экрана. В нем находятся системные значки (часы, громкость и т. д.) и иконки программ.

Языковой пакет – пакет, предоставляющий пользователю локализованный язык интерфейса, устанавливаемый поверх двоичных файлов операционной системы.

Экранное разрешение – размеры получаемого на экране изображения в пикселях.

Интерфейс – оболочка, предоставленная пользователю для удобного взаимодействия с программой.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 2 ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ КЛАССОВ

Далее приведено описание классов, написанных собственноручно во время реализации данного приложения.

Описание и функциональное назначение классов приложения для MacOS [17].

Таблица 1. Описание и функциональное назначение классов и структур

Название	Тип	Назначение
CommandInfo	struct	Формат для представления команды и соответствующего сочетания клавиш.
CommandsResponse	struct	Формат для представления ответа сервера обо всех командах пользователя.
AppControl_AppControlNIOClient	class	Сгенерированный класс для взаимодействия с сервером через gRPC.
Commands_CommandsNIOClient	class	Сгенерированный класс для взаимодействия с сервером через gRPC,
AppControlService	class	Сервис для взаимодействия с сервером. Приостановка микрофона.
CommandService	class	Сервис для взаимодействия с сервером. Работа с командами и сочетаниями.
Connection	class	Утильный класс для проверки и установки соединения с сервером.
AddShortcutView	struct	Окно для добавления команд.
HomeView	struct	Основное окно для выбора функционала.
ShortcutsView	struct	Окно для отображения

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

		всех команд, а также их удаления.
VoiceKeyboardApp	struct	Основная точка запуска приложения и сервера.

Описание и функциональное назначение классов приложения для Windows [19].

Таблица 2. Описание и функциональное назначение классов

Название	Тип	Назначение
AppControlGrpcClient	class	Клиент для взаимодействия с сервером через gRPC для работы с микрофоном.
CommandsGrpcClient	class	Клиент для взаимодействия с сервером через gRPC для работы с командами.
GrpcClient	class	Класс для установки соединения с сервером.
GrpcClientUtil	class	Класс для проверки наличия соединения с сервером.
CommandModel	class	Модель сочетания клавиш и команд.
AddCommandViewModel	class	Модель для взаимодействия с окном добавления команд.
CommandsViewModel	class	Модель для взаимодействия с окном просмотра и удаления команд.
MainWindowViewModel	class	Модель для взаимодействия с окном из системного трее.
AddCommandWindow	class	Инициализация окна добавления команды.
CommandsWindow	class	Инициализация окна просмотра и удаления команд.
MainWindow	class	Инициализация контекстного меню в системном трее.
App	class	Точка запуска приложения.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

ПРИЛОЖЕНИЕ 3**ОПИСАНИЕ И ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ ПОЛЕЙ, МЕТОДОВ И СВОЙСТВ**

В данной записке указаны свойства, поля и методы, относящиеся к данной части приложения, написанные собственноручно. Некоторые классы имплементируют встроенные классы, но описание уже написанных (в Swift [18]/C# [14]) методов приводится не будет. Также не будут расписаны классы, сгенерированные GRPC для клиента (AppControl_AppControlNIOClient и Commands_CommandsNIOClient).

Описание и функциональное назначение полей, методов и свойств приложения для MacOS [17].

Таблица 3. Описание и функциональное назначение свойств класса CommandInfo

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
id	public	Int	-	Идентификационный номер команды.
command	public	String	-	Строчное представление команды.
shortcut	public	String	-	Строчное представление сочетания клавиш.

Таблица 4. Описание и функциональное назначение полей класса CommandsResponse

Наименование	Модификатор доступа	Тип	Назначение
response	public	[CommandInfo]	Список команд с сочетаниями клавиш.
error	public	String	Строчное представление ошибки сервера

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Таблица 5. Описание и функциональное назначение методов класса AppControlService

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
host	private	String	-	Поле для хранения хоста сервера.
port	private	Int	-	Поле для хранения порта сервера.
id	private	Int	-	Поле для установления id командам с сервера.
turnOnMicro	public	String	Bool	Включение/выключение распознавания команд пользователя.

Таблица 6. Описание и функциональное назначение методов класса CommandService

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
host	private	String	-	Поле для хранения хоста сервера.
port	private	Int	-	Поле для хранения порта сервера.
id	private	Int	-	Поле для установления id командам с сервера
addCommand	public	String	command: String, hotkey: String	Делает запрос на сервер на добавление команды и сочетания клавиш.
deleteCommand	public	String	command: String, hotkey:	Делает запрос на сервер на удаление

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

			String	команды и сочетания клавиш.
getCommands	public	CommandsResponse	-	Делает запрос на сервер на получение всех команд и сочетаний клавиш.
exportCommands	public	String	path: String	Делает запрос на сервер на экспорт команд и сочетаний клавиш.
importCommand	public	String	path: String	Делает запрос на сервер на импорт команд и сочетаний клавиш.

Таблица 7. Описание и функциональное назначение методов класса Connection

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
established	public	Bool	host: String, port: Int	Проверяет установлено ли соединение с сервером.

Таблица 8. Описание и функциональное назначение полей структуры AddShortcutView

Наименование	Модификатор доступа	Тип	Назначение
hotKey	private	String	Текущее сочетание клавиш, введенное пользователем.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

command	private	String	Текущая команда, введенная пользователем.
showingAlert	private	Bool	Флаг, показывающий, нужно ли показывать уведомление пользователю.
response	private	String	Ответ сервера о добавлении команды.
body	public	View	Полноценное окно для добавления команды с сочетанием клавиш.

Таблица 9. Описание и функциональное назначение полей структуры HomeView.

Наименование	Модификатор доступа	Тип	Назначение
taskProcess	private	Process	Процесс, в котором работает локальный сервер.
commandService	private	CommandService	Сервис для управления командами на сервере.
showingAlert	private	Bool	Флаг, показывающий, нужно ли показывать уведомление пользователю.
response	private	String	Ответ сервера о добавлении команды.
body	public	View	Полноценное окно для выбора различных действий с командами.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

appControlService	private	AppControlService	Сервис для управления статусом микрофона.
status	private	Bool	Текущий статус микрофона, ведется ли распознавание команд.
state	private	String	Статус ошибки/уведомления от сервера
directoryName	private	String	Путь к выбранной пользователем директории.
fileName	private	String	Путь к выбранному пользователем файла.

Таблица 10. Описание и функциональное назначение полей структуры ShortcutsView

Наименование	Модификатор доступа	Тип	Назначение
commandService	private	CommandService	Сервис для управления командами на сервере.
commands	private	[CommandInfo]	Список всех команд пользователя.
showingAlert	private	Bool	Флаг, показывающий, нужно ли показывать уведомление пользователю.
response	private	String	Ответ сервера о добавлении команды.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

body	public	View	Полноценное окно для просмотра команд с сочетаниями клавиш, а также удаления команд.
selection	private	CommandInfo.ID?	Текущая, выбранная пользователем команда из списка.

Таблица 11. Описание и функциональное назначение полей структуры VoiceKeyboardApp

Наименование	Модификатор доступа	Тип	Назначение
commandService	private	CommandService	Сервис для управления командами на сервере.
taskProcess	private	Process	Процесс, в котором работает локальный сервер.
host	private	String	Поле для хранения хоста сервера.
port	private	String	Поле для хранения порта сервера.
body	public	View	Полноценное окно для просмотра команд с сочетаниями клавиш, а также удаления команд.
appControlService	private	AppControlService	Сервис для управления статусом микрофона.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Описание и функциональное назначение полей, методов и свойств приложения для Windows [19].

Таблица 12. Описание и функциональное назначение полей и методов класса AppControlGrpcClient

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
client	private	AppControlClient	-	Клиента для обращения к серверу для работы с микрофоном.
instance	private	AppControlGrpcClient	-	Инстанс для обращения к клиенту.
GetInstance	public	AppControlGrpcClient	-	Получение инстанса.
ChangeMicrophoneStatus	public	void	bool	Изменения статуса микрофона на сервере.

Таблица 13. Описание и функциональное назначение полей и методов класса CommandsGrpcClient

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
client	private	CommandsClient	-	Клиента для обращения к серверу для работы с командами.
instance	private	CommandsGrpcClient	-	Инстанс для обращения к клиенту.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

GetInstance	public	CommandsGrpcClient	-	Получение инстанса.
AddCommand	public	void	String command, string hotKey	Добавление команды и сочетания клавиш.
DeleteCommand	public	void	String	Удаление команды и сочетания клавиш.
ImportCommands	public	void	String	Импорт команд и сочетаний клавиш.
ExportCommands	public	void	String	Экспорт команд и сочетаний клавиш.
GetCommands	public	IDictionary<string,string>	-	Получение списка команд и сочетаний клавиш.

Таблица 14. Описание и функциональное назначение полей и методов класса GrpcClient

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
Channel	protected	GrpcChannel	-	Адрес и порт к серверу приложения.
TryMakeRequest	protected	void	Action	Обращение к серверу и обработка ответа.

Таблица 15. Описание и функциональное назначение полей и методов класса GrpcClientUtil

Наименование	Модификатор	Тип возвращаемого	Аргументы	Назначение
Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

	доступа	значения		
ServerHost	public	string	-	Хост сервера.
ServerPort	public	int	-	Порт сервера.
PingServer	public	bool	String hostUri, int portNumber	Проверка соединения с сервером.

Таблица 16. Описание и функциональное назначение полей и методов класса CommandModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
command	private	string	-	Команда.
hotKey	private	string	-	Сочетание клавиш.
ToString	public	string	-	Строчное представление команды и сочетания клавиш.

Таблица 17 Описание и функциональное назначение свойств класса CommandModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
Command	public	string	string	Получение или присвоение значения команде.
HotKey	public	string	string	Получение или присвоение значения сочетанию клавиш.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Таблица 18. Описание и функциональное назначение полей и методов класса AddCommandViewModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
commandsClient	private	CommandsGrpcClient	-	Клиент для работы с командами на сервере.
commandModel	private	CommandModel	-	Представление команды и сочетания клавиш.
commands	private	ObservableCollection<CommandModel>	-	Список всех команд с сочетаниями.
CreateModels	private	void	-	Создание моделей для окна.
CreateCommands	private	void	-	Создание команд для биндингов окна.
UpdateCommandsList	private	void	-	Обновление списка команд.

Таблица 19. Описание и функциональное назначение свойств класса AddCommandViewModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
AddCommand	public	ICommand	-	Получение

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Command				команды добавления команды.
CommandModel	public	CommandModel	CommandModel	Получение или запись модели команды.
CommandsList	private	ObservableCollection<CommandModel>	ObservableCollection<CommandModel>	Получение или запись списка команд.

Таблица 20. Описание и функциональное назначение полей и методов класса CommandsViewModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
commandsClient	private	CommandsGrpcClient	-	Клиент для работы с командами на сервере.
commandModel	private	CommandModel	-	Представление команды и сочетания клавиш.
commands	private	ObservableCollection<CommandModel>	-	Список всех команд с сочетаниями.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

CreateModels	private	void	-	Создание моделей для окна.
CreateCommands	private	void	-	Создание команд для биндингов окна.
UpdateCommandsList	private	void	-	Обновление списка команд.

Таблица 21. Описание и функциональное назначение свойств класса CommandsViewModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
DeleteCommand	public	ICommand	-	Команда удаления команды.
CommandModel	public	CommandModel	CommandModel	Получение или запись модели команды.
CommandsList	private	ObservableCollection<CommandModel>	ObservableCollection<CommandModel>	Получение или запись списка команд.

Таблица 22. Описание и функциональное назначение полей и методов класса MainWindowViewModel

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
commandsClient	private	CommandsGrpcClient	-	Клиент для работы с командами на сервере.
appControlClient	private	AppControlGrpcClient	-	Клиент для работы с командами на сервере.
commandModel	private	CommandModel	-	Представление команды и сочетания клавиш.
commands	private	ObservableCollection<CommandModel>	-	Список всех команд с сочетаниями.
CreateModels	private	void	-	Создание моделей для окна.
CreateCommands	private	void	-	Создание команд для биндингов окна.
UpdateCommandsList	private	void	-	Обновление списка команд.

Таблица 23. Описание и функциональное назначение свойств класса MainWindowViewModel

Наименование	Модификатор доступа	Тип возвращаемого значения	Аргументы	Назначение
ImportCommandsCommand	public	ICommand	-	Команда импорта

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

				команд.
ExportCommandsCommand	public	ICommand	-	Команда экспорта команд.
ChangeMicrophoneStatusCommand	public	ICommand	-	Команда изменения статуса микрофона команд.
CommandModel	public	CommandModel	CommandModel	Получение или запись модели команды.
CommandsList	private	ObservableCollection<CommandModel>	ObservableCollection<CommandModel>	Получение или запись списка команд.

Классы AddCommandWindow, CommandsWindow, MainWindow являются классами инициализации окон и не имеют свойств, методов, полей, только конструктор.

Класс App является классом инициализации приложения, точкой старта, и также не имеет свойств, методов, полей.

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата

[illegible]

Изм.	Лист	№ документа	Подпись	Дата
RU.17701729.02.02-01 81 01-1				
Инв. № подл.	Подп. и дата	Взам. Инв. №	Инв. № дубл.	Подп. и дата