Департамент образования и науки Тюменской области

Государственное автономное профессиональное

Образовательное учреждение Тюменской области

«Тюменский колледж транспортных технологий и сервиса»

Допущен к защите

Зав. Отделением автоматики,

Информатики и сервиса

О.С.Гордузова

« » 2020г.

КУРСОВОЙ ПРОЕКТ

МДК какой-то там

Тема Разработка программного модуля «Название курсовой»

ТКТТС. О.09.02.07. ПКС 1-17. КП. ПЗ

Пояснительная записка

Выполнил обучающейся Учаев М.А.

Подпись Ф.И.О

Руководитель работы Русанов В.С.

Подпись Ф.И.О

Нормоконтролер Русанов В.С.

Подпись Ф.И.О

Курсовой проект защищен

с оценкой « »

В.С.Русанов

(подпись, Ф.И.О.руководителя)

«10» февраль 2020г.

Тюмень 2020

Содержание

**Введение:**

На сегодняшний день наш мир погружен в глобальную рутину интернета. Миллионы людей, день из-за дня заходят в интернет для поиска интересующей их информации и не только. Сейчас можно воспользоваться большим количеством различных услуг, не выходя из дома. Достаточно иметь перед собой телефон, чтобы зайти в поисковую систему и за доли секунды найти себе подходящую услугу.

Так же, миллионы людей сейчас зарегистрированы в различных социальных сетях, ведь социальные сети на данный момент открывают людям множество различных возможностей, особенно интернет-магазинам, компаниям и так далее. Пользователю достаточно быть подписанным на ту или иную группу, и он будет постоянно получать интересующие его новости, развлекательный контент или же какой-нибудь товар, который только-только появился на складах интернет-магазина.

Особенной популярностью на данный момент пользуются «чат-боты». Чат-бот – это виртуальный собеседник или робот, разработанный для имитации живого человека посредством общения с одним или несколькими пользователями. Возможностей у чат- бота очень много, вплоть от обычного общения до автоматизации чего-либо. Обычно чат-бот используется в роли помощника в решение многих задач, к примеру вы заходите на сайт и через несколько секунд всплывает сбоку окошко «ассистента» или «консультанта», который вас приветствует и предлагает вам помощь, если у вас возникли какие-то вопросы, это и есть чат-бот. Для большинства компаний чат-бот может принести не малую пользу.

Как бы мы не хотели, но наш мир не стоит на месте и год за годом появляются различные технологии, которые облегчают нашу жизнь и чат-боты этому не исключение. Введя свой бизнес или блог, для вас чат-бот будет не заменимой вещью. При большой аудитории одному отвечать на все сообщения и к тому же предоставлять свои услуги будет слишком утомительно, поэтому эту работу можно перекинуть на своего бота, тем самым экономив свое время и направляя его на реализацию более важный задач.

Правда не стоит и забывать про минусы чат-ботов, а минусы у чат-бота следующие:

* Излишество различных функций, которые выполняет чат-бот – при большом количестве функций чат-бот может не справиться ни с одним из них.
* Чат-боты чаще всего могут отвечать на простые вопросы – если написать ему то, что будет выходить за приделы его возможностей, то боль может ничего не ответить и проигнорировать сообщение пользователя
* Примитивные алгоритмы – существует два вида чат-ботов: первый – с искусственным интеллектом, второй – запрограммированный. Чат бота с искусственным интеллектом чаще всего разрабатывают IT-гиганты, поэтому обычным пользователям лучше сосредоточиться на запрограммированном боте, который будет отвечать на определённые сценарии поведения

К плюсам же мы можем отнести следующее:

* Главный – это их быстродействие, пользователю не нужно ждать несколько минут, а и часов, когда им ответит человек, когда бот ответит за считанные секунды.
* Бесплатное обслуживание – вам не приходится платить боту за его работу
* Бота легко настраивать

Целью данного курсового проекта является разработка чат-бота по артам художников в такой социальной сети как «Telegram».

Для достижение поставленной цели необходимо выполнить следующее:

* Описать предметную область «чат-бот по артам художников для социальной сети Telegram»
* Создать чат-бота в Telegram
* Разобраться с TelegramBot API
* Разработать интерфейс модуля
* Сценарий пользователей программного модуля
* Методы тестирования Telegram Bot
* Задеплоить нашего бота на хост-сервер

Объектом разработки курсового проекта является среда разработки Visual Studio Code, такой язык программирование как Python версии 3.7, использование библиотеки telebot для быстрой и упрощённой разработки нашего Telegram бота.

Предметом курсового проекта является создание собственного чат-бота в телеграмме.

В результате мы получим собственного бота, который будет находиться на хост-сервере и выполнять некоторые функции после того как пользователь будет с ним взаимодействовать.

1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ
   1. Спецификация программного модуля «чат-бот по артам художников в социальной сети Telegram»

Выбирая чат-бота на такой платформе как Telegram, я основывался на том факторе, что телеграмм все больше и больше набирает обороты и в скором времени, данная платформа может обогнать такие известные платформы как: VK, Instagram и т.д.

Данный продукт предназначен для людей любых возрастов. Особенно он будет полезен таким сферам труда как арт-студии, при наличии большой аудитории можно внедрят в бота созданные арт проекты, тем самым показывая всем свои новые сделанные продукты.

Бот обладает следующим функционалом:

* Отправлять пользователям картинки / арты
* Отправлять пользователям видеоматериалы, содержащие обучающий контент
* Отправлять пользователям скетчи(быстрые заготовки), кому необходимо отточить своё мастерство в рисование
* Отправлять готовые арт работы художников

Для взаимодействия с ботом, пользователям не обязательно будет ему писать текст, у бота оборудованная кнопочная панель, нажатия на которые бот будет совершать те или иные действия

Входные данные:

* Нажатие пользователем на кнопки из меню бота
* Отслеживание действий пользователей скриптом

Выходные данные:

* При нажатие на определенную кнопку, бот должен отвечать соответствующей функции, для которой она предназначена
* Отклик бота осуществляется в том случае, если действия пользователей с ботом не выходят за рамки его функционала

Требования к системе

Система должна выполнять свою основную функцию, отвечать на отклик пользователей и выдавать им тот материал, на который они ссылаются

Функциональные требования

* Работоспособность всех кнопок в меню бота
* Реагировать на запросы от пользователей
* Отвечать в коротком промежутке времени
* Иметь удобное меню
* Реагировать на обращения, которые не входят в функционал бота
  1. Создание чат-бота в Telegram

Создание чат-бота производится следующим образом:

В Telegram присутствует конструктор ботов, с помощью которого происходит создания оболочки бота, для этого необходимо:

* Написать в поисковой строке Telegram «BotFather»
* Для того что бы начать общаться с BotFather нужно написать команду /start
* Далее бот отправляет сообщение с командами, которые он может делать, выбираем команду /newbot
* После BotFather будет спрашивать имя нашего бота, пишем как имя нашего бота
* Следом после имени, BotFather будет спрашивать username(имя пользователя) нашего бота, по которому его будут искать пользователи, обязательный критерий, username нашего бота должно оканчиваться на bot, например: (Hellobot)
* Далее, после всего проделанного, BotFather отправляет нам сообщение, в котором содержится токен нашего бота, токен нашего бота никому нельзя показывать и отдавать, так как другой пользователь, который узнает токен нашего бота сможет им управлять, поэтому его стоит сохранить в безопасном месте.

Наша оболочка бота создана, далее достаточно нам вбить в поисковую строку @username нашего бота и перейти к переписке с ним.

На данный момент, бот не будет отвечать на сообщения пользователя. Так же у BotFather, есть другие команды, с помощью которых мы можем поменять аватар бота, изменить информацию о боте, добавить информацию о боте на главный экран, когда пользователь только будет начинать с ним общаться, данная информация будет отображаться в самом чате.

Для написания скрипта чат-бота нужно использовать язык программирования, а также среду разработки. Я остановил свой выбор на таком языке программирования как Python и среде разработки Visual Studio Code

Python – высокоуровневый язык программирования, ориентированный на повышение производительности разработчика и читаемости кода. Python, является довольно легким языком программирования из-за своего синтаксиса, к тому же в python присутствует множество библиотек, в частности библиотека telebot, которая позволяет быстро и эффективно разрабатывать скрипт для нашего бота.

Выбирая среду разработки чат-бота, я основывался на некоторых пунктов:

* Удобный и интуитивно понятный пользовательский интерфейс
* Множество различных функция и библиотек, позволяющие хорошо и быстро разрабатывать код
* Красивый дизайн и удобство в использовании

Visual Studio Code – отлично подходит под данные критерии, это один из лучших редакторов кода, которые помогут качественно писать код.

* 1. Telegram bot API

Telegram bot API – это программный интерфейс, позволяющий программировать нашего бота

API включает в себя объекты и команды, которые предназначены для установки поведения нашего бота в Telegram. Используя интерфейс, мы можем создавать собственные программные коды, которые при запуске в Telegram начинают работать как боты.

В Telegram bot API все элементы управления представляют объекты, которые представлены в JSON, иными словами в виде строки, заданной по определённым правилам. Это позволяет обмениваться данными по всему сервису легко и без особых затрат, так как при передаче данных изображается не сам код, а целый набор ключевых значений, которые выходят в текстовом значении. Огромное количество элементов управления предназначено для того, чтобы выстроить команду установщика. Ключевые слова расширяют боле возможностей объектов. Обмениваться текстами между другими клиентами можно в виде соответствующих запросов.

Примеры:

* sendMessage – для получения сообщения
* sendVideo – для отправки видео
* sendAudio – для отправки аудиозаписи

Все элементы управления делятся на категорию групп:

* Получение обновления и информационных запусков
* Общение в чате
* Отправки разнообразных элементов
* Функции со стикерами
* Обновление текстовых сообщений
* Платежного функционала
* Игр и развлекательных порталов
  1. Разобрать интерфейс модуля

Интерфейс модуля, представляет собой стандартную фоновую стилизацию. В модуле размещены 4 главные кнопки, и несколько побочных, для более простого взаимодействия пользователя с ботом. Интерфейс программного модуля представлен на рисунке 1.1 и рисунке 1.2

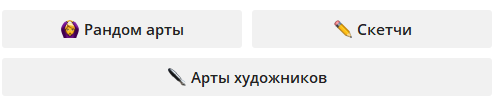


Рисунок 1.1 – Главный интерфейс модуля

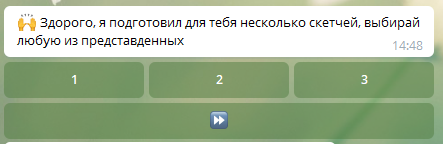


Рисунок 1.2 – Побочный интерфейс пользователя.

* 1. Сценарий программного модуля (программный модуль)

Сценарий: зарегистрироваться в Telegram:

* Скачать и установить Telegram
* Нажать «Start Message»
* Ввести данные
* Войти в Telegram

Сценарий: использования бота:

* Найти бота по его имени
* Написать команду /start
* Нажать на кнопку в главном меню
* Нажать на кнопку в второсортном меню

Сценарий: прекращение использование бота:

* Нажать на 3 точки в правом верхнем углу бота
* Нажать удалить сообщения
* Выбрать также удалить для бота тоже
* Нажать кнопку удалить
  1. Методы тестирования Telegram Bot

Тестирование методом черного ящика. Данная техника не требует каких-либо знаний о внутренней работе приложения. Рассматриваются только основные аспекты системы, не связанные или мало связанные с ее внутренней логической структурой.

Принципы тестирования черного ящика. В этом методе программа рассматривается как [чёрный ящик](https://ru.wikipedia.org/wiki/%D0%A7%D1%91%D1%80%D0%BD%D1%8B%D0%B9_%D1%8F%D1%89%D0%B8%D0%BA).

Целью [тестирования](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F) ставится выяснение обстоятельств, в которых поведение программы не соответствует [спецификации](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B5%D1%86%D0%B8%D1%84%D0%B8%D0%BA%D0%B0%D1%86%D0%B8%D1%8F). Для обнаружения всех [ошибок](https://ru.wikipedia.org/wiki/%D0%9E%D1%88%D0%B8%D0%B1%D0%BA%D0%B0) в программе необходимо выполнить исчерпывающее тестирование, то есть тестирование на всевозможных наборах [данных](https://ru.wikipedia.org/wiki/%D0%94%D0%B0%D0%BD%D0%BD%D1%8B%D0%B5_(%D0%B2%D1%8B%D1%87%D0%B8%D1%81%D0%BB%D0%B8%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%82%D0%B5%D1%85%D0%BD%D0%B8%D0%BA%D0%B0)).

Для большинства программ такое невозможно, поэтому применяют разумное тестирование, при котором тестирование программы ограничивается небольшим подмножеством всевозможных наборов данных. При этом необходимо выбирать наиболее подходящие подмножества, подмножества с наивысшей вероятностью обнаружения ошибок.

Тестирование методом белого ящика. Это подробное исследование внутренней логики и структуры программы. При этом необходимо знание исходного кода.

Этот метод не ставит цели выявление синтаксических ошибок, так как дефекты такого рода обычно обнаруживает компилятор. Методы белого ящика направлены на локализацию ошибок, которые сложнее выявить, найти и зафиксировать. С их помощью можно обнаружить логические ошибки и проверить степень покрытия тестами.

Выводы по теоретической части

В результате рассмотрения теоретического материала была разработана спецификация программного модуля «бот для социальных сетей “Арты”». Рассмотрена структура создания ботов в такой социальной сети как Telegram. Были рассмотрены всевозможные сценарии для пользователей. Был рассмотрен дизайн бота, содержащий стандартную технологию визуальности.

Представлены примеры аналогов данного приложения, а также их преимущества и недостатки. Описаны методы тестирования, при помощи которых можно проверить правильность и корректность работы модуля «бот для социальных сетей “Арты”».

2 ПРАКТИЧЕСКАЯ ЧАСТЬ

2.1 Средства разработки

Telegram боты представляют собой специальные аккаунты, которые могут автоматически обрабатывать и отправлять сообщения. На данный создать бота может абсолютно каждый человек, ведь для его написания после вышедшего обновления сильные знания в языке программирования уже не нужны. Ведь написать собственного помощника можно с помощью команд.

Telegram bot – это приложение, которое запушено на стороне пользователя и осуществляет отправку запросов.

Возможности ботов растут каждый день и в скором будущем наступит день когда эта технология вытеснит другие продукты.

У ботов есть следующие ограничения:

* Принимает аудио только в одном формате (.ogg).
* Бот не может отправлять сообщение другому боту.
* Бот не умеет забирать фото от пользователей.
* Бот не может писать первым. Он должен отозваться на команду пользователя которая будет характеризовать, что пользователь хочет начать общение с ним.

В данном проекте были использованы инструменты разработки такие как:

* Python
* Visual Studio Code

2.1 Язык программирования Python

Python – это интерпретируемый язык программирования. Код написанный на языке Python преобразуется в машинный с помощью интерпретатора.

Язык Python был разработан в 1991 году человеком по имени Гвидо Ван Россумом.

Python активно совершенствуется и по сей день. Python имеет большое комьюнити состоящее из множества разработчиков, тестировщиков, системных программистов.

Python прост в понимании из-за своего простого синтаксиса. Читать код на этом языке программирования достаточно легко, потому что в нем мало вспомогательных элементов.

Python – универсальный язык программирования. Он поддерживает объектно-ориентированное программирование.

Множество разработчиков выбирают именно этот язык программирования по таким причинам как:

* Качество программного обеспечения. Легкость чтения кода позволяет многократно использовать и обслуживать гораздо проще, чем в других языках программирования.
* Библиотеки поддержки. Python поддерживает множество различных библиотек, одна из известный стандартная библиотека.
* Переносимость программ. Большая часть программ на языке Python выполняется без изменений на всех основных платформах.
* Скорость разработки. По сравнению с такими языками программирования как C++, C или Java, Python во много раз повышает производительность работы. Объем программного кода на языке Python обычно составляет треть, или даже пятую часть кода языка программирования Java или C++, что означает меньший объем ввода с клавиатуры, меньшее количество времени на откладку и меньший объем трудозатрат на сопровождение.

Основные структуры данных Python:

* Список
* Объекты и классы
* Кортеж
* Словарь
* Последовательности
* Множество
* Ссылки

Язык Python используется в компаниях начиная от маленьких студий и заканчивая компаниями гигантами:

* Компания Google использует Python в своей поисковой системе.
* Такие компании, как Intel, Cisco, Hewlett-Packard и другие используют Python для тестирования аппаратного обеспечения.
* Служба использования видеоматериалов YouTube в значительной степени реализована на Python.
* Популярная программа BitTorrent для обмена файлами в пиринговых сетях написана на языке Python.
* Популярный веб-фреймворк App Engine от компании Google использует Python в качестве прикладного языка программирования.

2.2 Среда разработки Visual Studio Code

Visual Studio Code – это легкий и в то же время мощный редактор исходного кода, который доступен для Windows, MacOS и Linux. Он поставляется со встроенной поддержкой JavaScript, TypeScript и Node.js и имеет много расширений для других языков программирования, C++, C#, Python, PHP, Go и другие, и работы с .NET и Unity.

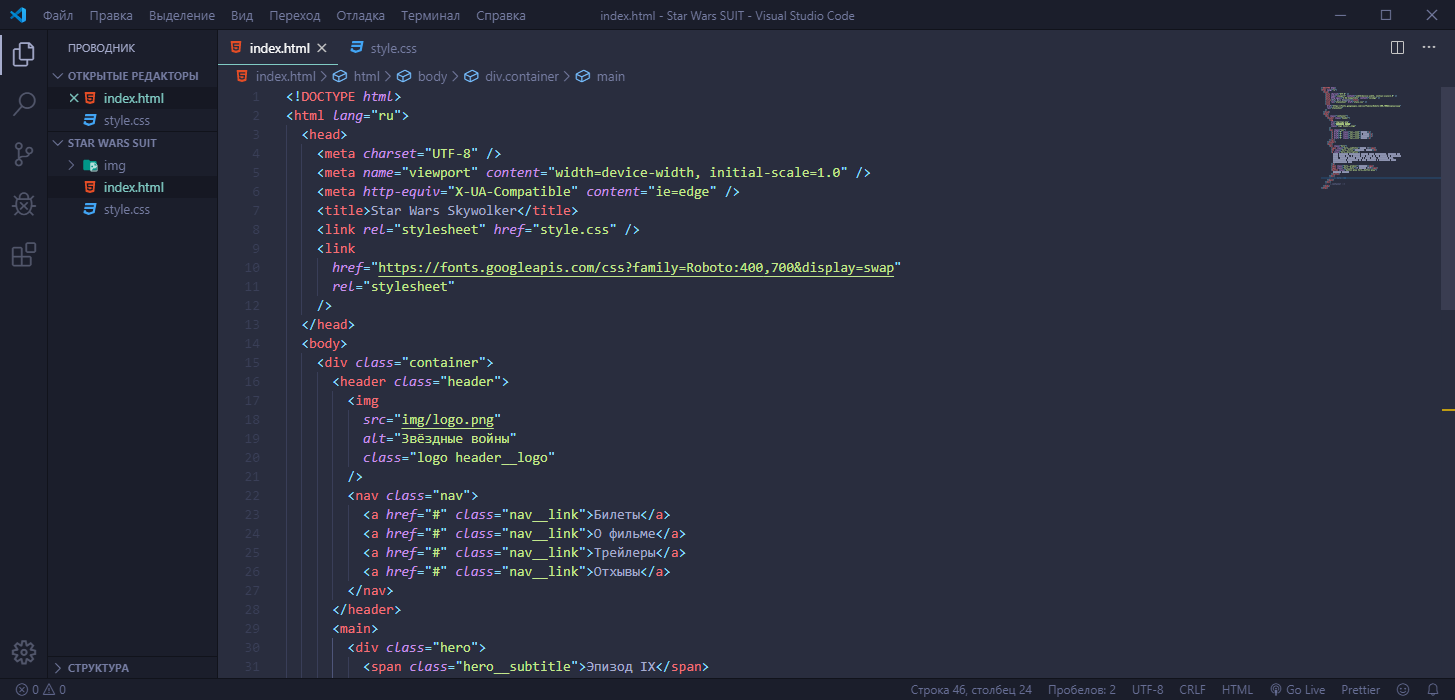


Рисунок 2.1 – Интерфейс Visual Studio Code

Инструментарий Visual Studio Code:

* ASP.NET 5 — гибкий фреймворк создания современных веб-приложений.
* NodeJS (включает NPM) – платформа для создания масштабируемых сетевых приложений.
* git – система контроля версий.
* Yeoman – инструмент «скаффолдинга» примерно тоже самое что и File | New Project в VS.
* generator-aspnet — yeoman генератор для ASP.NET 5 приложений, выполните npm install -g generator-aspnet для установки.
* hottowel — yeoman генератор для быстрого создания AngularJS приложений, выполните npm install -g generator-hottowel для установки.
* Express – фреймворк для Node приложений, использует «движок» шаблонов Jade.
* gulp – инструментарий создания «тасков» для выполнения сопутствующих сборке проекта задач.
* mocha – фреймворк создания модульных тестов на JavaScript/Node.
* bower – клиентский пакетный менеджер.
* TypeScript definition manager – определения TypeScript для популярных JavaScript библиотек, включают поддержку IntelliSense в Code.

VS Code работает с файлами и папками в которых находятся проекты. В простейшем случае вы можете открыть файл на редактирование просто выполнив команду ./code index.html. Более интересным случаем является открытие папки. Например, если в папке находятся файлы index.html, style.css, script.js то VS Code включает подсказки, навигацию по коду, выполнение команд и многое другое.

VS Code обладает интуитивно понятной и простой схемой расположения основных элементов. Интерфейс разделен на четыре основных блоков:

* Редактор, основной блок в котором осуществляется изменение содержимого открытого файла
* Сайдбар, с помощью которого можно увидеть различные представления файлов проекта
* Статусбар, показывающий текущий статус различных операций
* Вьюбар, позволяющий переключаться между режимами сайдбара и индицирующий с помощью иконок различную информацию например, количество входящих изменений git изменений.

2.3 Разработка чат-бота.

Создание бота проходит через несколько этапов и в первую очередь нужно создать аккаунт для нашего бота.

Что бы получить доступ чат-бота к API Telegram, нужно получить токен у бота BotFather.

Ответом на команду /start, BotFather выдаст нам список доступных команд

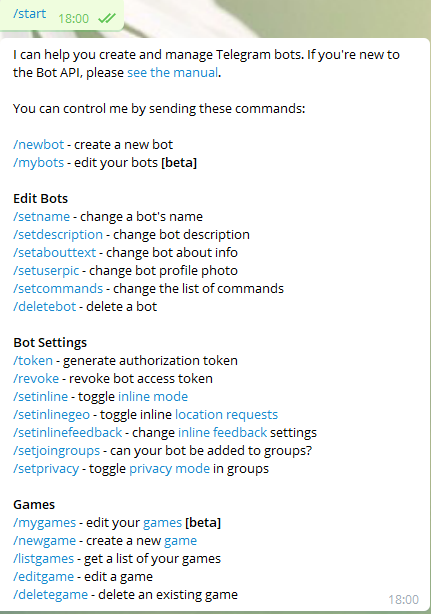


Рисунок 2.2 – Ответ BotFather на команду /start

Что бы получить токен бота, нужно зарегистрировать нашего чат-бота введя команду /newbot. BotFather нужно отправить имя чат-бота и его адрес, по которому его будут находить и добавлять пользователи.

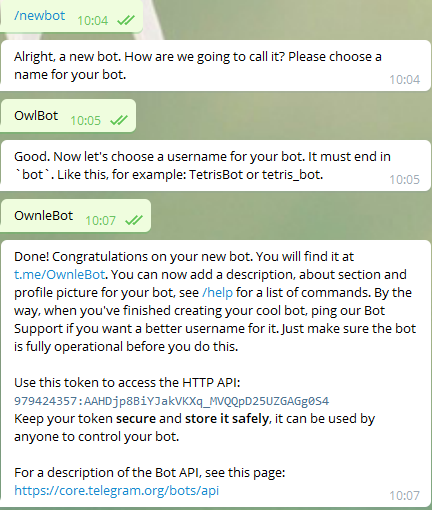


Рисунок 2.3 – Ответ на команду /newbot

Можно добавить описание нашего бота и информацию, которая будет отображаться у него в профиле, используя команды /setdescription и /setabouttext



Рисунок 2.4 – Ответ бота на команды /setdescription и /setabouttext

Чтобы сменить аватар у бота, нужно использовать команду /setuserpick. После отправки команды, необходимо прислать фотографию для вашего аватара.



Рисунок 2.5 – Результат выполнение команды /setuserpick

2.4 Функционал чат-бота.

/start – стартовая команда, которая способствует началу общение с ботом. После приветствия, бот выдает меню с последующими командами.

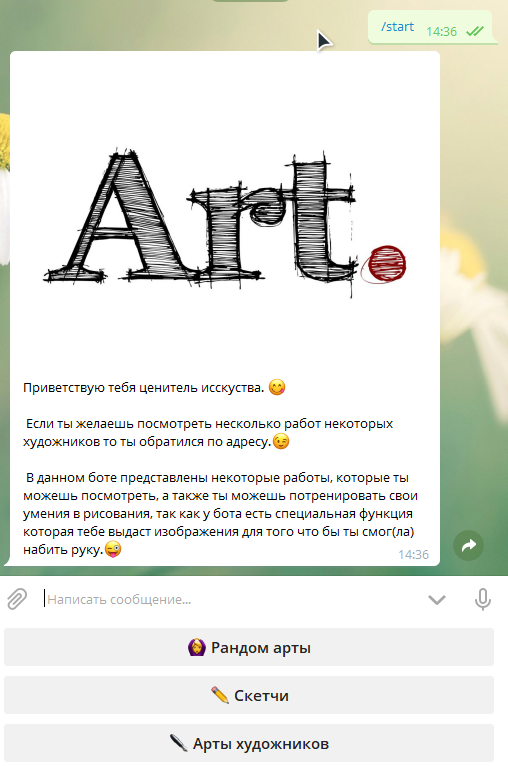


Рисунок 2.6 – Результат выполнение команды /start

/Рандом арты – Бот отправляет отправляет арт картины пользователю, которые определяются случайным порядком



Рисунок 2.7 – Ответ на команду /Рандом арты

/Скетчи – Бот присылает список из 10 скетчей (быстрых заготовок). Под сообщением бота всплывает меню, позволяющая пользователю переключаться между скетчами. Для удобства под каждым скетчем находится кнопка удалить, которая удаляет скетч.

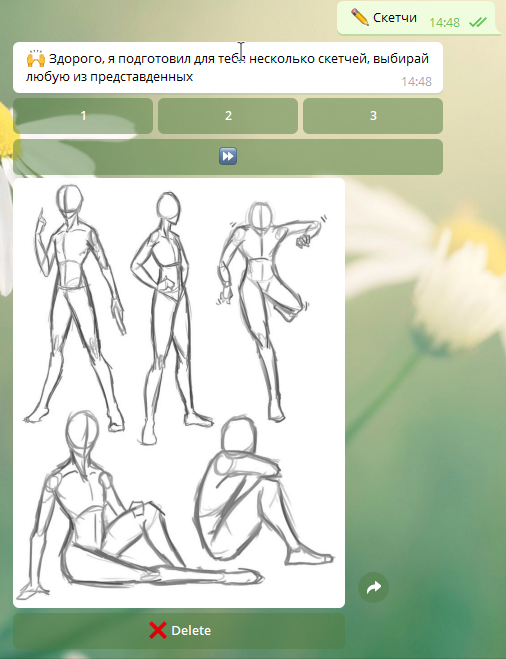


Рисунок 2.8 – Ответ на команду /Скетчи

/Арты художников – Команда позволяет выбрать художника и посмотреть его некоторые работы. Под ответный сообщением бота всплывает меню с четырмя художниками, по нажатию на них, появляется ещё одно меню, где представлены четыре работы выбранного пользователем художника. Так же предусмотрена кнопка «назад» в основном меню.

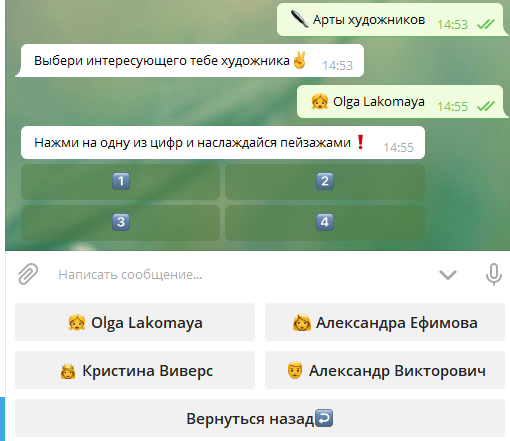


Рисунок 2.9 – Ответ на команду /Арты художников

ЗАКЛЮЧЕНИЕ …