

Разработка программного обеспечения на языке Python

[Обзорная панель](#) [Мои курсы](#) [Разработка ПО на языке Python](#) [Программирование на языке Python](#)

[Задание 12. Разработка технического задания на создание телеграмм бота](#)

Задание 12. Разработка технического задания на создание телеграмм бота

Разработка Telegram-бота

Цель: познакомиться со структурой технического задания на разрабатываемое программное обеспечение, научиться разрабатывать техническое задание на разрабатываемое программное обеспечение, сформировать навыки разработки программного обеспечения на языке Python согласно разработанному техническому заданию.

Задание:

1. Определиться с назначением своего telegram-бота, например, конвертер валют, глоссарий-бот, погодный бот и др. Составить документ «Техническое задание на разработку Telegram-бота»;
2. Разработать архитектуру Telegram-бота;
3. Разработать интерфейс Telegram-бота;
4. Разработать согласно составленному техническому заданию Telegram-бота.

Методические указания

На сегодняшний день существует несколько стандартов, предъявляющий требования к создаваемой программной системе и устанавливающий порядок, в соответствии с которым должна проводиться ее разработка и прием при вводе в эксплуатацию:

- ГОСТ 34
- ГОСТ 19
- IEEE STD 830-1998

Исходя из этих стандартов, разрабатываемое техническое задание должно включать в себя следующие разделы:

1. Введение (наименование программного продукта, назначение);
2. Общее описание (краткое описание функционала продукта, видение продукта);
3. Детальные требования (функциональные требования);
4. Стадии и этапы разработки;
5. Порядок контроля и приемки.

Архитектура Telegram-бота

Проектирование архитектуры Telegram-бота включает в себя следующий перечень задач:

1. Составление списка вопросов и ответов на них
2. Определение логики (иерархии запросов) для Telegram-бота

Интерфейс чат-бота

Проектирование Интерфейса чат-бота заключается в определении команд чат-бота и способов взаимодействия с пользователем и включает в себя следующий перечень задач:

1. Набор команд чат-бота
2. Элементы управления чат-бота

Пример разработки Telegram-бота

Процедура разработки Telegram-бота включает следующие этапы:

1. Провести регистрацию нового бота в мессенджере.
2. Установить Python-библиотеку для работы с Telegram.

3. Научить бот реагировать на сообщения в пределах чата.
4. Добавить кнопки для взаимодействия с пользователем.

Регистрация нового бота

1. Открыть Telegram
2. При помощи командной строки найти @BotFather. Он несет ответ за регистрацию нового бота.

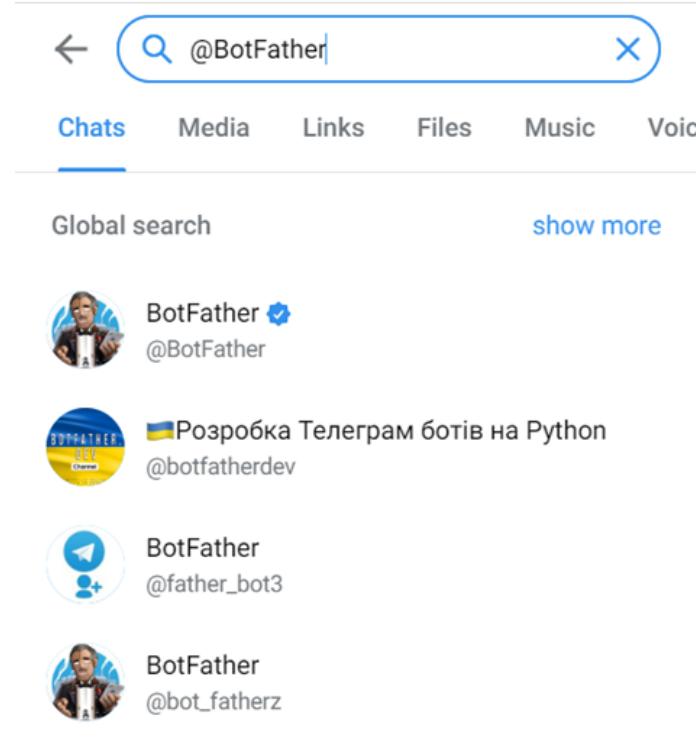


Рисунок 1

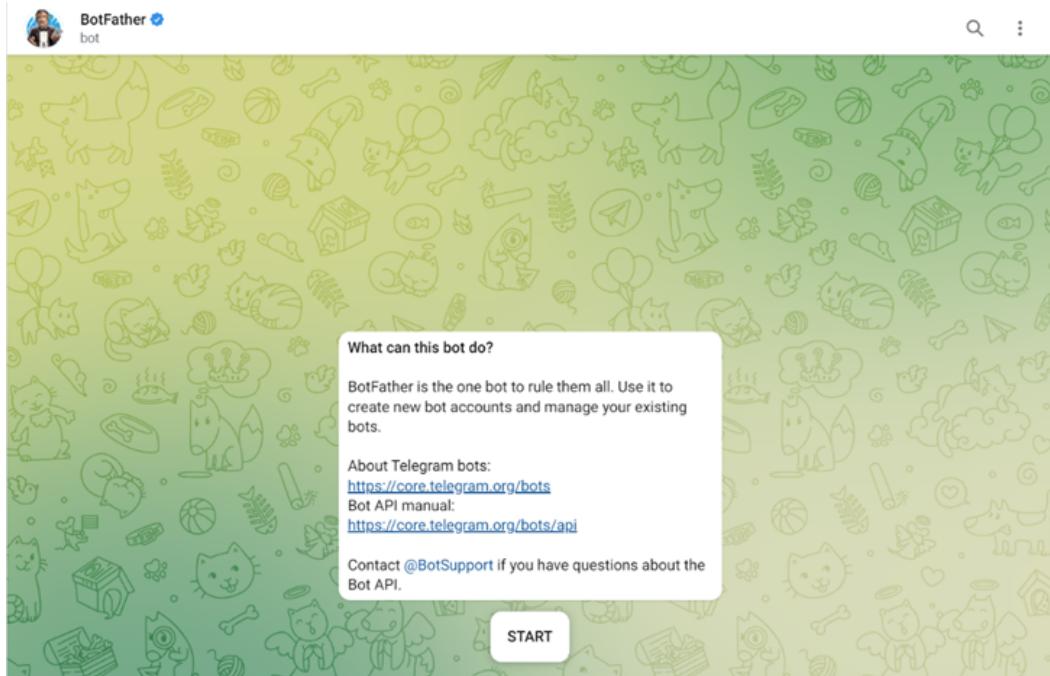


Рисунок 2

1. Кликнуть по надписи Start, а также указать команду /newbot

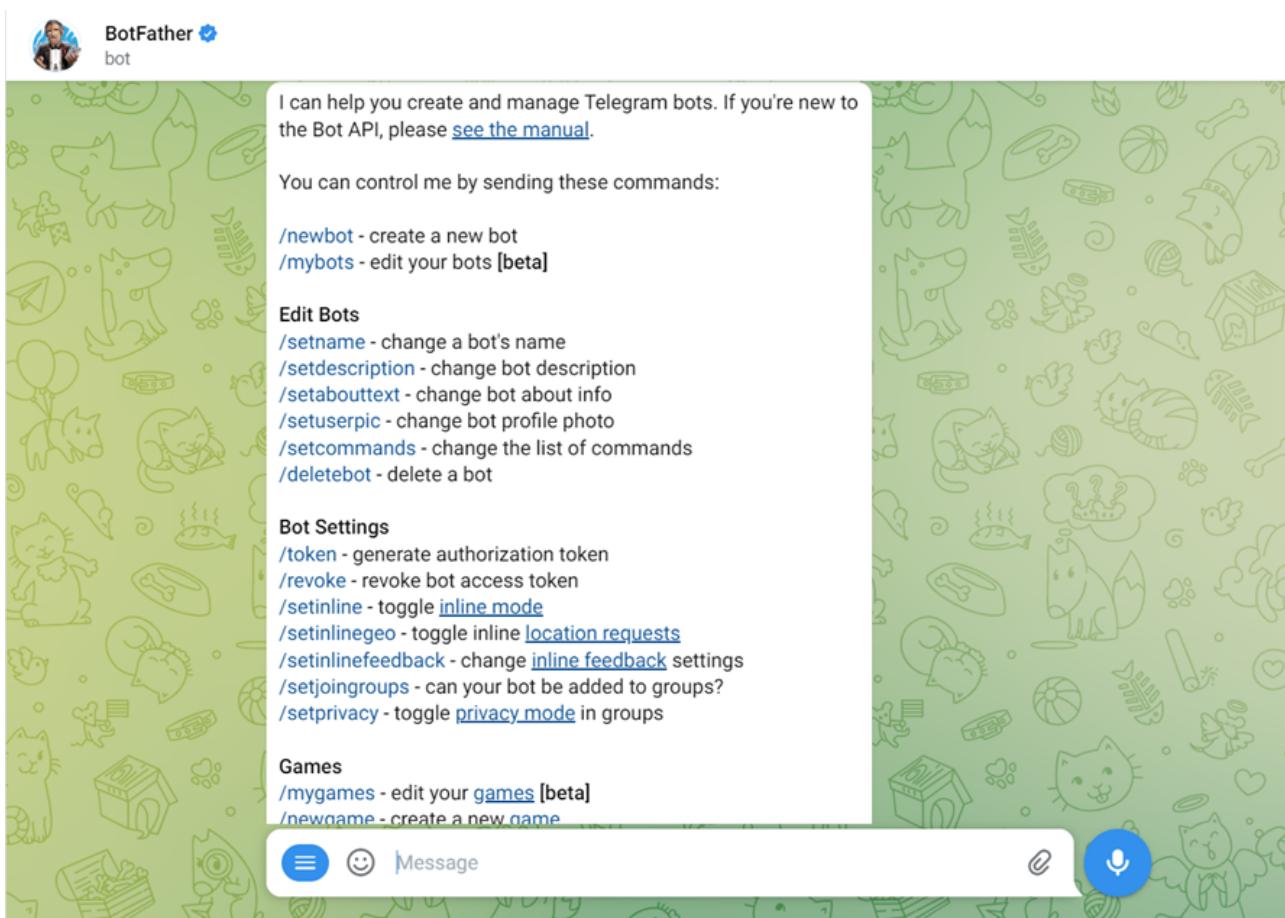


Рисунок 3

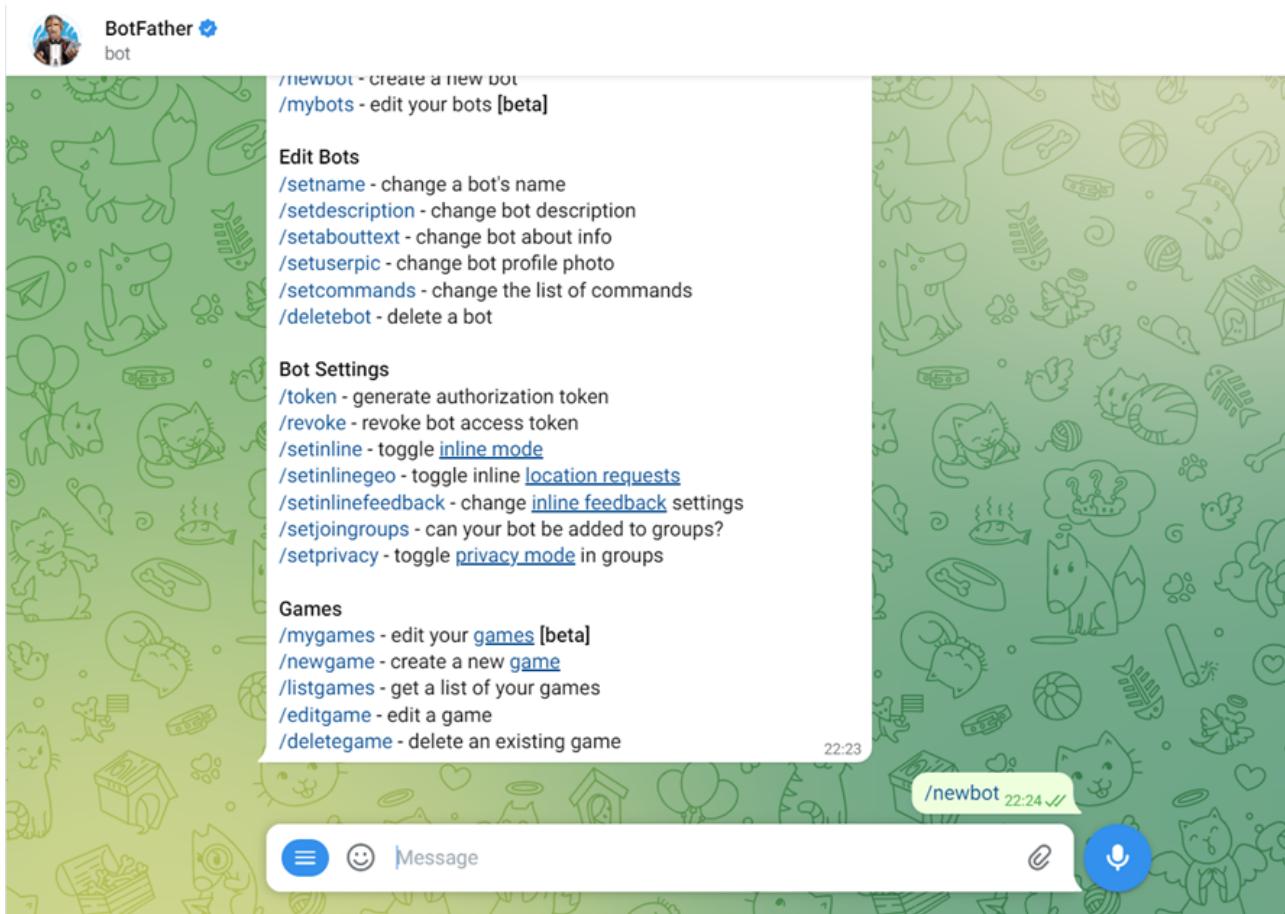


Рисунок 4

- Система задаст поочередно вопросы о названии бота и его нике.

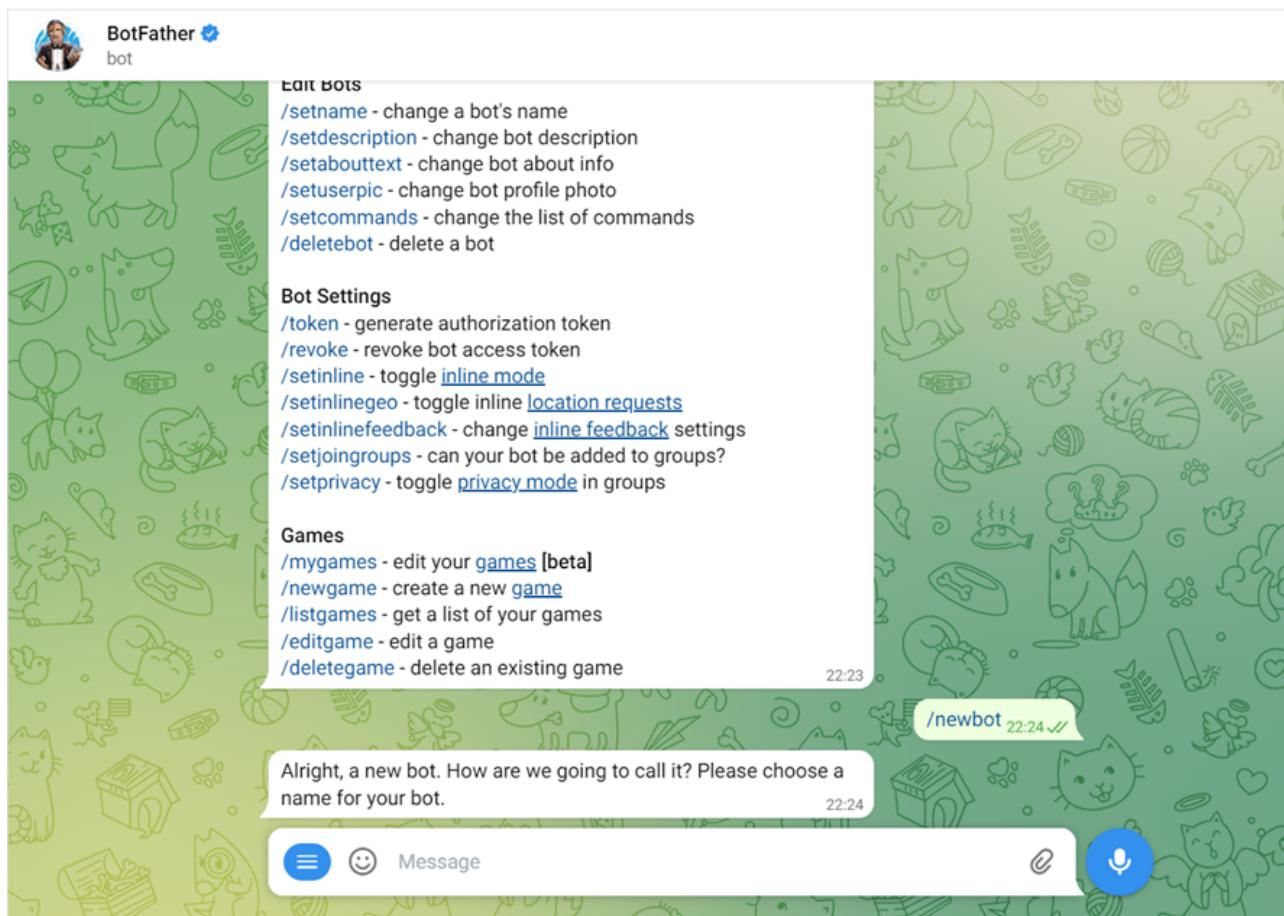


Рисунок 5

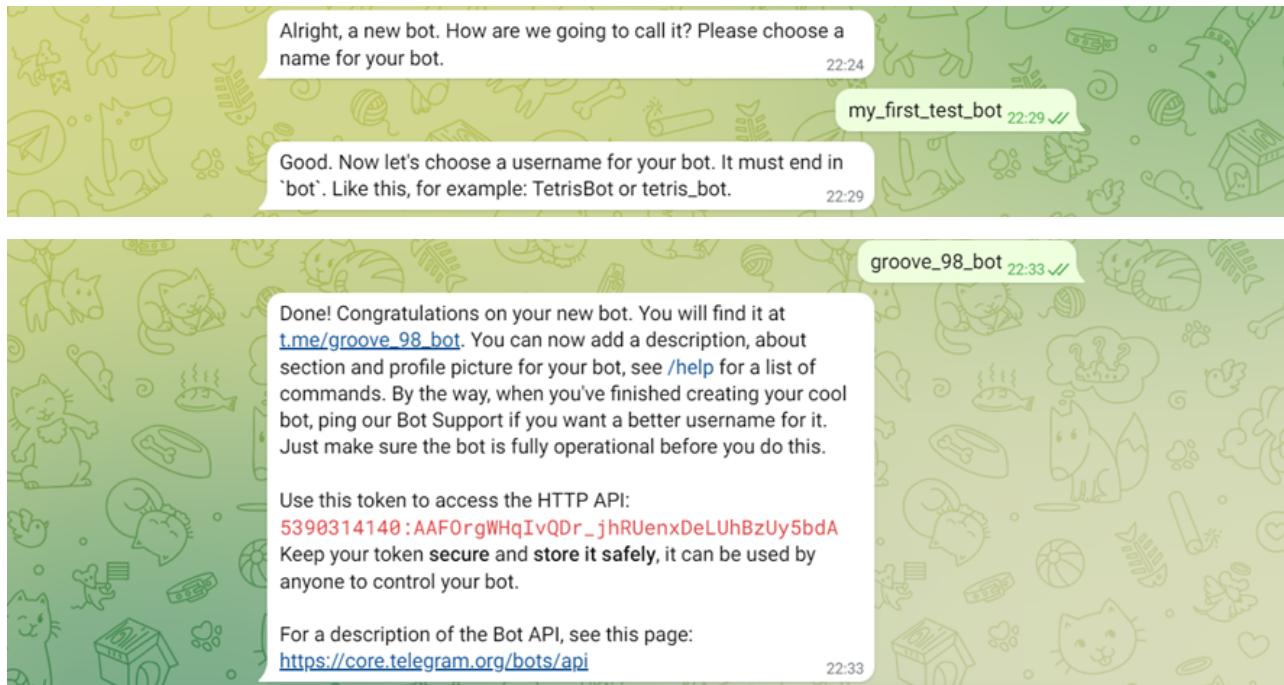


Рисунок 6

Установка библиотеки и ее инициализация

Самый простой вариант для работы с Telegram предоставляет библиотека telebot

1. Запустить командную строку на компьютере.
2. Набрать команду `pip install pyTelegramBotAPI`.
3. Подтвердить обработку операции.

```
C:\Users\asmikhalev>C:\Users\asmikhalev\anaconda3\Scripts\pip.exe install pyTelegramBotAPI
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
Collecting pyTelegramBotAPI
  Downloading pyTelegramBotAPI-4.7.0.tar.gz (210 kB)
    210.7/210.7 kB 713.3 kB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: requests in c:\users\asmikhalev\anaconda3\lib\site-packages (from pyTelegramBotAPI)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\asmikhalev\anaconda3\lib\site-packages (from pyTelegramBotAPI) (1.26.9)
Requirement already satisfied: idna<4,>=2.5 in c:\users\asmikhalev\anaconda3\lib\site-packages (from requests->pyTelegramBotAPI) (2.10)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\asmikhalev\anaconda3\lib\site-packages (from requests->pyTelegramBotAPI) (2.0.12)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\asmikhalev\anaconda3\lib\site-packages (from requests->pyTelegramBotAPI) (2021.10.8)
Building wheels for collected packages: pyTelegramBotAPI
  Building wheel for pyTelegramBotAPI (setup.py) ... done
  Created wheel for pyTelegramBotAPI: filename=pyTelegramBotAPI-4.7.0-py3-none-any.whl size=192795 sha256=df92c773879bd18f0fe7aa46934124e29f3bad28e5a01c195fc9f3d7
  Stored in directory: c:\users\asmikhalev\appdata\local\pip\cache\wheels\f7\29\1d\113c046ac93c2896159bb2c33673d42f21a06c4ac5
Successfully built pyTelegramBotAPI
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
Installing collected packages: pyTelegramBotAPI
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
Successfully installed pyTelegramBotAPI-4.7.0
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution -umpy (c:\users\asmikhalev\anaconda3\lib\site-packages)
WARNING: Ignoring invalid distribution - (c:\users\asmikhalev\anaconda3\lib\site-packages)
C:\Users\asmikhalev>
```

Рисунок 7

1. Создать новый проект.
2. Чтобы приложение понимало бота, подключаем библиотеку с помощью инструкции `import telebot`.
 1. Создаем экземпляр бота с помощью следующего кода `Bot = telebot.TeleBot('токен')`.
 2. Вместо слова «токен» нужно вставить настоящий токен, выданный `@BotFather`.

Реализация механизма реагирования на сообщения в пределах чата

Добавим функционал, который будет обрабатывать команды пользователя. Если пользователь напишет нам «Привет», то скажем ему «Привет, сейчас я расскажу тебе гороскоп на сегодня», а если нам напишут команду «/help», то скажем пользователю написать «Привет». Для этого объявим метод для получения текстовых сообщений `get_text_messages` и добавим следующий код:

```
@bot.message_handler(content_types=['text'])
def get_text_messages(message):
    if message.text == "Привет":
        bot.send_message(message.from_user.id, "Привет, сейчас я расскажу тебе гороскоп на сегодня")
    elif message.text == "/help":
        bot.send_message(message.from_user.id, "Напиши привет")
    else:
        bot.send_message(message.from_user.id, "Я тебя не понимаю. Напиши /help.")
```

Теперь нужно добавить после метода следующий код

```
bot.polling(none_stop=True, interval=0)
```

Это позволит боту постоянно проверять наличие новых сообщений.

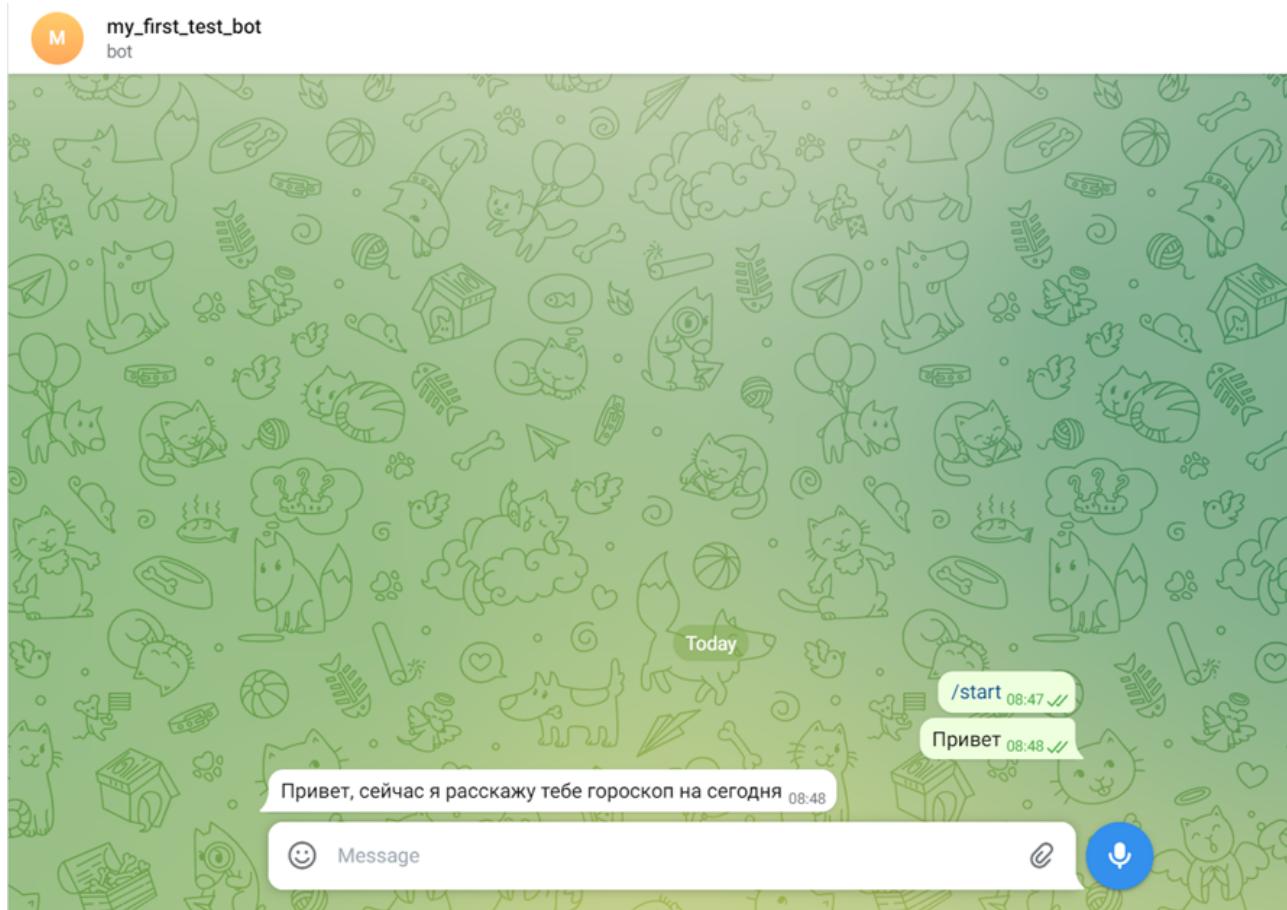


Рисунок 8

Добавление кнопок для взаимодействия с пользователем

- Импортируем типы из модуля telebot, чтобы создавать кнопки

```
from telebot import types
```

- Добавляем код приложения, который отвечает за создание кнопок, соответствующих знакам зодиака

```
# Готовим кнопки
keyboard = types.InlineKeyboardMarkup()
# По очереди готовим текст и обработчик для каждого знака зодиака
key_oven = types.InlineKeyboardButton(text='Овен', callback_data='zodiac')
# И добавляем кнопку на экран
keyboard.add(key_oven)
key_telc = types.InlineKeyboardButton(text='Телец', callback_data='zodiac')
keyboard.add(key_telc)
key_blinzecy = types.InlineKeyboardButton(text='Близнецы', callback_data='zodiac')
keyboard.add(key_blinzecy)
key_rak = types.InlineKeyboardButton(text='Рак', callback_data='zodiac')
keyboard.add(key_rak)
key_lev = types.InlineKeyboardButton(text='Лев', callback_data='zodiac')
keyboard.add(key_lev)
key_deva = types.InlineKeyboardButton(text='Дева', callback_data='zodiac')
```

```

keyboard.add(key_deva)
key_vesy = types.InlineKeyboardButton(text='Весы', callback_data='zodiac')
keyboard.add(key_vesy)
key_scorpion = types.InlineKeyboardButton(text='Скорпион', callback_data='zodiac')
keyboard.add(key_scorpion)
key_strelec = types.InlineKeyboardButton(text='Стрелец', callback_data='zodiac')
keyboard.add(key_strelec)
key_kozerog = types.InlineKeyboardButton(text='Козерог', callback_data='zodiac')
keyboard.add(key_kozerog)
key_vodoley = types.InlineKeyboardButton(text='Водолей', callback_data='zodiac')
keyboard.add(key_vodoley)
key_gyby = types.InlineKeyboardButton(text='Рыбы', callback_data='zodiac')
keyboard.add(key_gyby)
# Показываем все кнопки сразу и пишем сообщение о выборе
bot.send_message(message.from_user.id, text='Выбери свой знак зодиака', reply_markup=keyboard)

```

1. Добавляем обработчик кнопок. Он будет реагировать на слово zodiac.

```

# Заготовки для трёх предложений
first = ["Сегодня – идеальный день для новых начинаний.", "Оптимальный день для того, чтобы решиться на смелый поступок!", "Будьте осторожны, сегодня звёзды могут повлиять на ваше финансовое состояние.", "Лучшее время для того, чтобы начать новые отношения или разобраться со старыми.", "Плодотворный день для того, чтобы разобраться с накопившимися делами."]
second = ["Но помните, что даже в этом случае нужно не забывать про", "Если поедете за город, заранее подумайте про", "Те, кто сегодня нацелен выполнить множество дел, должны помнить про", "Если у вас упадок сил, обратите внимание на", "Помните, что мысли материальны, а значит вам в течение дня нужно постоянно думать про"]
second_add = ["отношения с друзьями и близкими.", "работу и деловые вопросы, которые могут так некстати помешать планам.", "себя и своё здоровье, иначе к вечеру возможен полный раздрай.", "бытовые вопросы – особенно те, которые вы не доделали вчера.", "отдых, чтобы не превратить себя в загнанную лошадь в конце месяца."]
third = ["Злые языки могут говорить вам обратное, но сегодня их слушать не нужно.", "Знайте, что успех благоволит только настойчивым, поэтому посвятите этот день воспитанию духа.", "Даже если вы не сможете уменьшить влияние ретроградного Меркурия, то хотя бы доведите дела до конца.", "Не нужно бояться одиноких встреч – сегодня то самое время, когда они значат многое.", "Если встретите незнакомца на пути – проявите участие, и тогда эта встреча послужит вам приятные хлопоты."]

```

```

@bot.callback_query_handler(func=lambda call: True)
def callback_worker(call):
    # Если нажали на одну из 12 кнопок – выводим гороскоп
    if call.data == "zodiac":
        # Формируем гороскоп
        msg = random.choice(first) + ' ' + random.choice(second) + ' ' + random.choice(second_add) + ' ' +
random.choice(third)
        # Отправляем текст в Телеграм
        bot.send_message(call.message.chat.id, msg)

```

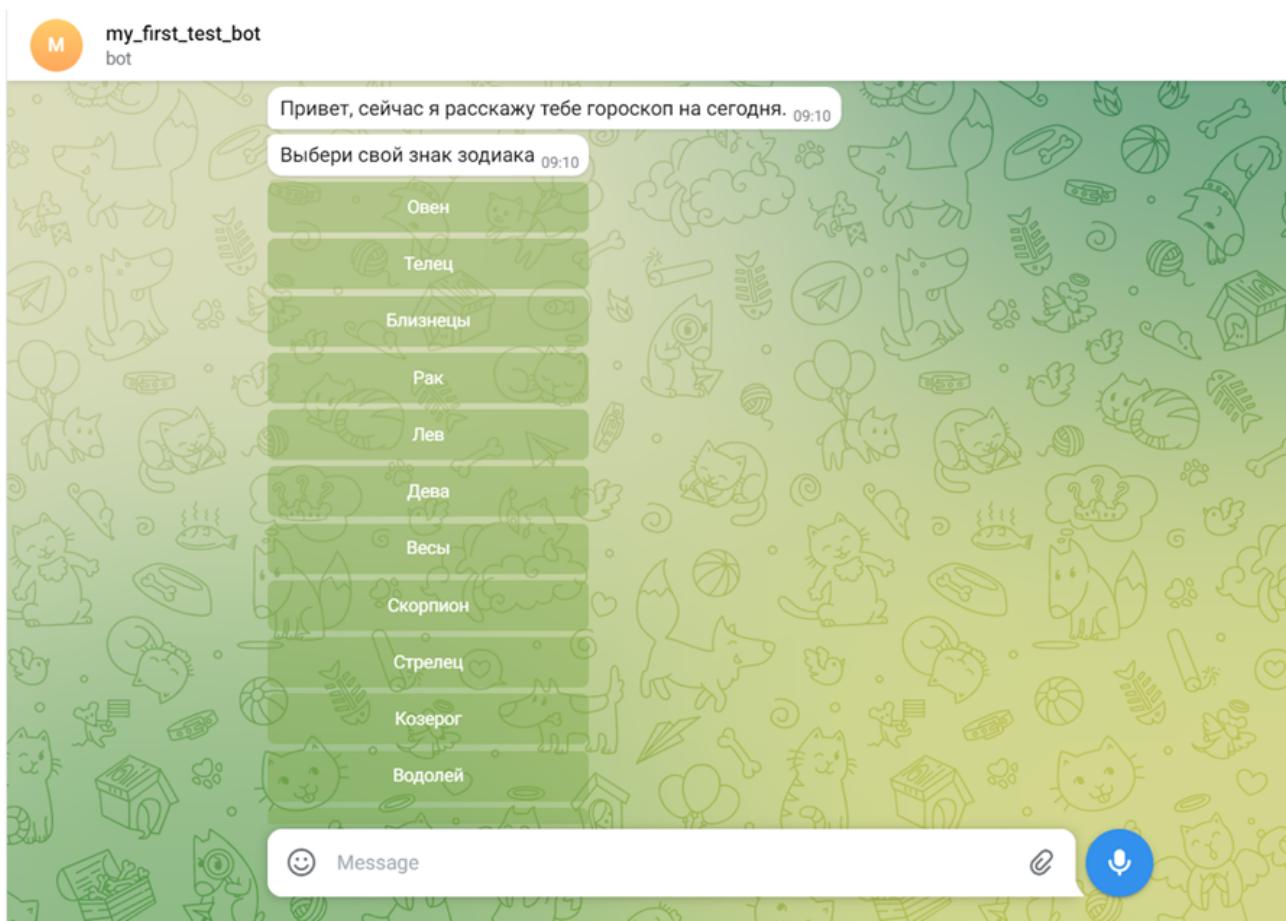


Рисунок 9

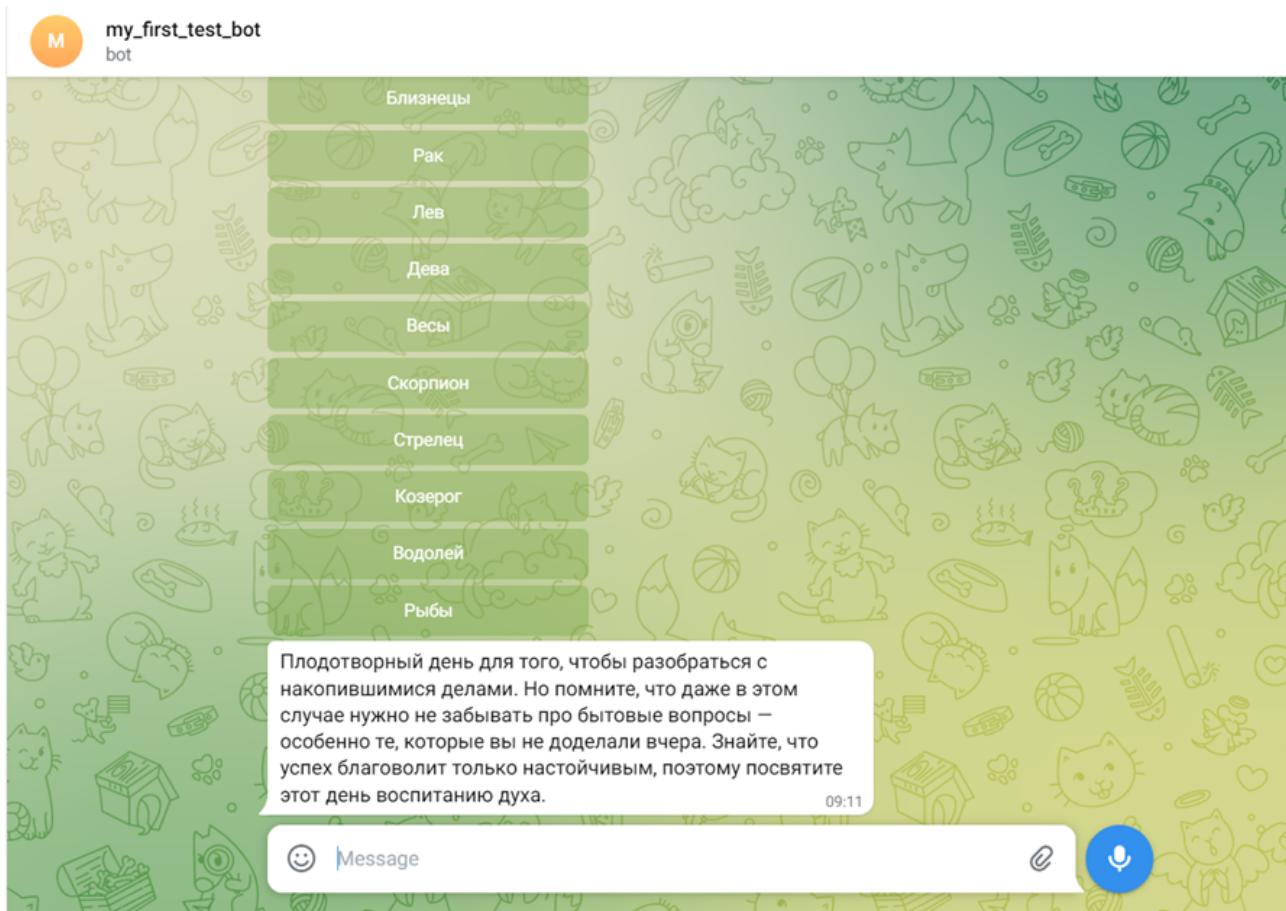


Рисунок 10

Приложение А. ГОСТ 19.201-78 «Техническое задание, требования к содержанию и оформлению»

Согласно ГОСТ 19.201-78 «Техническое задание, требования к содержанию и оформлению» техническое задание должно включать следующие разделы:

1. Введение;
2. Основания для разработки;
3. Назначение разработки;
4. Требования к программе или программному изделию;
5. Требования к программной документации;
6. Технико-экономические показатели;
7. Стадии и этапы разработки;
8. Порядок контроля и приемки;
9. Приложения.

Содержание разделов

В разделе «**Введение**» указывают наименование, краткую характеристику области применения программы или программного изделия и объекта, в котором используют программу или программное изделие.

В разделе «**Основания для разработки**» должны быть указаны:

- документ (документы), на основании которых ведется разработка;
- организация, утвердившая этот документ, и дата его утверждения;
- наименование и (или) условное обозначение темы разработки.

В разделе «**Назначение разработки**» должно быть указано функциональное и эксплуатационное назначение программы или программного изделия.

Раздел «**Требования к программе или программному изделию**» должен содержать следующие подразделы:

- требования к функциональным характеристикам;
- требования к надежности;
- условия эксплуатации;
- требования к составу и параметрам технических средств;
- требования к информационной и программной совместимости;
- требования к маркировке и упаковке;
- требования к транспортированию и хранению;
- специальные требования.

В подразделе «**Требования к функциональным характеристикам**» должны быть указаны требования к составу выполняемых функций, организации входных и выходных данных, временным характеристикам и т. п.

В подразделе «**Требования к надежности**» должны быть указаны требования к обеспечению надежного функционирования (обеспечения устойчивого функционирования, контроль входной и выходной информации, время восстановления после отказа и т.п.).

В подразделе «**Условия эксплуатации**» должны быть указаны условия эксплуатации (температура окружающего воздуха, относительная влажность и т.п. для выбранных типов носителей данных), при которых должны обеспечиваться заданные характеристики, а также вид обслуживания, необходимое количество и квалификация персонала.

В подразделе «**Требования к составу и параметрам технических средств**» указывают необходимый состав технических средств с указанием их основных технических характеристик.

В подразделе «**Требования к информационной и программной совместимости**» должны быть указаны требования к информационным структурам на входе и выходе и методам решения, исходным кодам, языкам программирования и программным средствам, используемым программой.

При необходимости должна обеспечиваться защита информации и программ.

В подразделе «**Требования к маркировке и упаковке**» в общем случае указывают требования к маркировке программного изделия, варианты и способы упаковки.

В подразделе «Требования к транспортированию и хранению» должны быть указаны для программного изделия условия транспортирования, места хранения, условия хранения, условия складирования, сроки хранения в различных условиях.

В разделе «Требования к программной документации» должен быть указан предварительный состав программной документации и, при необходимости, специальные требования к ней.

В разделе «Технико-экономические показатели» должны быть указаны: ориентировочная экономическая эффективность, предполагаемая годовая потребность, экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами.

В разделе «Стадии и этапы разработки» устанавливают необходимые стадии разработки, этапы и содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены), а также, как правило, сроки разработки и определяют исполнителей.

В разделе «Порядок контроля и приемки» должны быть указаны виды испытаний и общие требования к приемке работы.

В приложениях к техническому заданию, при необходимости, приводят:

- перечень научно-исследовательских и других работ, обосновывающих разработку;
- схемы алгоритмов, таблицы, описания, обоснования, расчеты и другие документы, которые могут быть использованы при разработке;
- другие источники разработки.

Приложение Б. ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы»

Согласно ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы» техническое задание на создание системы, в которую входят программное обеспечение, аппаратное обеспечение, люди, которые работают с программным обеспечением, и автоматизируемые процессы должно включать следующие разделы:

1. Общие сведения
2. Назначение и цели создания (развития) системы
3. Характеристика объектов автоматизации
4. Требования к системе
5. Состав и содержание работ по созданию системы
6. Порядок контроля и приемки системы
7. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие
8. Требования к документированию
9. Источники разработки

Подробное описание структуры технического задания.

1. Общие сведения

1.1. Наименование системы

1.1.1. Полное наименование системы

1.1.2. Краткое наименование системы

1.3. Наименование организаций – Заказчика и Разработчика

1.3.1. Заказчик

1.3.2. Разработчик

1.4. Плановые сроки начала и окончания работы

Указываются плановые сроки начала и окончания работ по созданию системы (на основании Договора). Если сроки определены не точно, то указать на какой стадии сроки уточняются.

1.5. Источники и порядок финансирования

Если не целесообразно указывать эти сведения, то дается ссылка на Договор.

1.6. Порядок оформления и предъявления заказчику результатов работ

Определяется порядок оформления и предъявления заказчику результатов работ по созданию системы (ее частей), по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы.

2. Назначение и цели создания системы

2.1. Назначение системы

Указать вид автоматизируемой деятельности (указать для управления какими процессами предназначена система).

Указать перечень объектов автоматизации, на которых предполагается использовать систему, перечень автоматизируемых органов (пунктов) управления объекта автоматизации и управляемых ими объектов (здесь указать в каких подразделениях предусматривается устанавливать систему и привести в разрезе подразделений перечень автоматизируемых бизнес-процессов верхнего уровня).

2.2. Цели создания системы

Наименования и требуемые значения технических, технологических, производственно-экономических или других показателей объекта автоматизации, которые должны быть достигнуты в результате создания АИС; критерии оценки достижения целей создания системы.

3. Характеристика объектов автоматизации

Приводятся краткие сведения об области деятельности Заказчика (или подразделения организационной структуры Заказчика, для нужд которого разрабатывается система) и сферы автоматизации с указанием ссылок на ранее разработанные документы, содержащие более подробные сведения об организации заказчика.

4. Требования к системе

4.1. Требования к системе в целом

4.1.1. Требования к структуре и функционированию системы

Определяется перечень функциональных подсистем, их назначение и основные характеристики, требования к числу уровней иерархии и степени централизации системы. Приводятся требования к характеристикам взаимосвязей со смежными системами. Определяются требования к режимам функционирования системы. Указываются требования по диагностированию системы (какие средства будут использоваться или создаваться, чтобы обеспечить диагностику системы).

4.1.2. Требования к численности и квалификации персонала системы и режиму его работы

4.1.3. Показатели назначения

4.1.3.1. Параметры, характеризующие степень соответствия системы назначению

4.1.3.2. Требования к приспособляемости системы к изменениям

4.1.3.3. Требования к сохранению работоспособности системы в различных вероятных условиях

4.1.4. Требования к надежности

4.1.4.1. Состав показателей надежности для системы в целом

4.1.4.2. Перечень аварийных ситуаций, по которым регламентируются требования к надежности

4.1.4.3. Требования к надежности технических средств и программного обеспечения

4.1.4.4. Требования к методам оценки и контроля показателей надежности на разных стадиях создания системы в соответствии с действующими нормативно-техническими документами.

4.1.5. Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

4.1.6. Требования к защите информации от несанкционированного доступа

4.1.6.1. Требования к информационной безопасности

4.1.6.2. Требования к антивирусной защите

- 4.1.7. Требования по сохранности информации при авариях
- 4.1.8. Требования к защите от влияния внешних воздействий
- 4.1.9. Требования по стандартизации и унификации
- 4.1.10. Дополнительные требования
- 4.1.11. Требования безопасности

4.2. Требования к функциям, выполняемым системой

В данном подразделе приводят:

- 1) по каждой подсистеме перечень функций, задач или их комплексов (в том числе обеспечивающих взаимодействие частей системы), подлежащих автоматизации; при создании системы в две или более очереди - перечень функциональных подсистем, отдельных функций или задач, вводимых в действие в 1-й и последующих очередях;
- 2) временной регламент реализации каждой функции, задачи (или комплекса задач);
- 3) требования к качеству реализации каждой функции (задачи или комплекса задач), форме представления выходной информации, характеристики необходимой точности и времени выполнения, требования к одновременности выполнения групп функций, достоверности выдачи результатов;
- 4) перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

4.3. Требования к видам обеспечения

4.3.1 Требования к математическому обеспечению

Для математического обеспечения системы приводятся требования к составу, области применения (ограничения) и способам использования в системе математических методов и моделей, типовых алгоритмов и алгоритмов, подлежащих разработке.

4.3.2. Требования к информационному обеспечению

Приводятся требования:

- 1) к составу, структуре и способам организации данных в системе;
- 2) к информационному обмену между компонентами системы;
- 3) к информационной совместимости со смежными системами;
- 4) по использованию общесоюзных и зарегистрированных республиканских, отраслевых классификаторов, унифицированных документов и классификаторов, действующих на данном предприятии;
- 5) по применению систем управления базами данных;
- 6) к структуре процесса сбора, обработки, передачи данных в системе и представлению данных;
- 7) к защите данных от разрушений при авариях и сбоях в электропитании системы;
- 8) к контролю, хранению, обновлению и восстановлению данных;
- 9) к процедуре придания юридической силы документам, производимым техническими средствами АС.

4.3.4. Требования к программному обеспечению

Для программного обеспечения системы приводят перечень покупных программных средств, а также требования: к независимости программных средств от используемых СВТ и операционной среды; к качеству программных средств, а также к способам его обеспечения и контроля; по необходимости согласования вновь разрабатываемых программных средств с фондом алгоритмов и программ.

4.3.5. Требования к техническому обеспечению

Приводятся требования:

- 1) к видам технических средств, в том числе к видам комплексов технических средств, программно-технических комплексов и других комплектующих изделий, допустимых к использованию в системе;
- 2) к функциональным, конструктивным и эксплуатационным характеристикам средств технического обеспечения системы.

5. Состав и содержание работ по созданию системы

Данный раздел должен содержать перечень стадий и этапов работ по созданию системы, сроки их выполнения, перечень организаций - исполнителей работ, ссылки на документы, подтверждающие согласие этих организаций на участие в создании системы, или запись, определяющую ответственного (заказчик или разработчик) за проведение этих работ.

6. Порядок контроля и приёмки системы

В разделе указывают:

- 1) виды, состав, объем и методы испытаний системы и ее составных частей (виды испытаний в соответствии с действующими нормами, распространяющимися на разрабатываемую систему);
- 2) общие требования к приемке работ по стадиям (перечень участвующих предприятий и организаций, место и сроки проведения), порядок согласования и утверждения приемочной документации;
- 3) статус приемочной комиссии (государственная, межведомственная, ведомственная).

7. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

В разделе необходимо привести перечень основных мероприятий, которые следует выполнить при подготовке объекта автоматизации к вводу Системы в действие, а также их исполнителей.

В перечень основных мероприятий включают:

- 1) приведение поступающей в систему информации (в соответствии с требованиями к информационному и лингвистическому обеспечению) к виду, пригодному для обработки с помощью ЭВМ;
- 2) изменения, которые необходимо осуществить в объекте автоматизации;
- 3) создание условий функционирования объекта автоматизации, при которых гарантируется соответствие создаваемой системы требованиям, содержащимся в ТЗ;
- 4) создание необходимых для функционирования системы подразделений и служб;
- 5) сроки и порядок комплектования штата и обучения персонала.

Приложение В. Стандарт 830-1998 – «IEEE Recommended Practice for Software Requirements Specifications»

Стандарт 830-1998 – IEEE Recommended Practice for Software Requirements Specifications определяет требования к разрабатываемому программному обеспечению. Согласно стандарту техническое задание должно включать следующие разделы:

1. Введение

- 1. Назначение
- 2. Область действия
- 3. Определения, акронимы и сокращения
- 4. Ссылки
- 5. Краткий обзор

2. Общее описание

- 1. Взаимодействие продукта (с другими продуктами и компонентами)
- 2. Функции продукта (краткое описание)
- 3. Характеристики пользователя
- 4. Ограничения
- 5. Допущения и зависимости

3. Детальные требования (могут быть организованы по разному, например, так)

- 1. Требования к внешним интерфейсам
 - 1. Интерфейсы пользователя
 - 2. Интерфейсы аппаратного обеспечения
 - 3. Интерфейсы программного обеспечения
 - 4. Интерфейсы взаимодействия

- 2. Функциональные требования
- 3. Требования к производительности
- 4. Проектные ограничения (и ссылки на стандарты)
- 5. Нефункциональные требования (надежность, доступность, безопасность и пр.)
- 6. Другие требования

 [Практическая работа 6 .docx](#)

7 сентября 2022, 13:14

Состояние ответа

Состояние ответа на задание	Ни одной попытки
Состояние оценивания	Не оценено
Последнее изменение	-
Комментарии к ответу	 Комментарии (0).

[Добавить ответ на задание](#)

Вы пока не предоставили ответ на задание

ПРЕДЫДУЩИЙ ЭЛЕМЕНТ КУРСА

 [Лекция 6. Жизненный цикл программного обеспечения. Разработка ПО. Взаимодействие программ.](#)

[Перейти на...](#)

СЛЕДУЮЩИЙ ЭЛЕМЕНТ КУРСА

[Задание 13. Властелин колец](#) ►

© 2010-2023 Центр обучающих систем
Сибирского федерального университета, [sfu-kras.ru](#)

Разработано на платформе moodle
Beta-version (3.9.1.5.m)

[Политика конфиденциальности](#)

[Соглашение о Персональных данных](#)

[Политика допустимого использования](#)

Контакты +7(391) 206-27-05
info-ms@sfu-kras.ru

[Скачать мобильное приложение](#)

Инструкции по работе в системе