

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
“ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ”

Факультет компьютерных наук
Кафедра информационных технологий управления
Telegram Bot

Курсовая работа

09.03.02 Информационные системы и технологии
Информационные системы и технологии в управлении предприятием

Допущен к защите

Зав. Кафедрой _____ *М. Г. Матвеев, д.т.н., профессор* _____.20__

Обучающиеся _____ *И. А. Фирсова, А. Г. Толчеева* 3 курс, д/о

Руководитель _____, *Полещук Х.А.*

Воронеж 2018

Содержание

1. Введение	3
2. Анализ	4
2.1. Анализ предметной области	4
2.2. Анализ задач	4
2.3. Анализ аналогов	5
2.3. Анализ средств реализации	5
2.4. Диаграммы	6
2.4.1. Диаграмма прецедентов	6
2.4.2. Диаграмма классов	7
2.4.3. Диаграмма объектов	8
2.4.4. Диаграмма коммуникации	8
2.4.5. Диаграмма последовательности	9
2.4.6. Диаграмма активности	10
2.4.7. Диаграмма развертывания	12
2.4.8. Диаграмма состояний	12
3. Основная часть	14
4. Тестирование	17
4.1 Smoke testing	18
4.2 Decision Table	22
4.3 Check-List	30
4.4 State Transition Testing	43
4.5 Confirmation testing	44
5. Заключение	52

1. Введение

Бот – это помощник, который общается с пользователем посредством сообщений и обладает рядом функций. Чаще всего он используется для рассылки информации или для ее сбора.

Боты позволяют пользователям получать различную информацию (в том числе в виде консультаций по интересующим вопросам), общаться, и осуществлять заказы в онлайн-сервисах в разы быстрее.

Примером автоматизации можно назвать большое количество ботов для приема заявок на доставку еды, заказа билетов или мест в гостинице, автоматизации звонков горячей линии, рассылки рекламы или полезной информации и др.

Такие боты позволяют увеличить прибыль компаний, так как бот может обрабатывать заявки с большей скоростью, чем человек, и уменьшить нагрузку на рабочий персонал.

На сегодняшний момент большим спросом обладают боты мессенджеры. Так как наиболее удобным способом получения информации для большинства людей является диалог. Уже сейчас приложениями для обмена сообщениями пользуются более 2 миллиардов человек, а если верить прогнозам скоро число пользователей увеличится до 2,5млрд.

На момент появления Telegram, пользователи успели протестировать такие мессенджеры как WhatsApp и Viber. Отсутствие у новинки русскоязычной версии привело к тому, что первыми пользователями были жители зарубежных стран (Америки, Испании и др.), большинство которых было задействовано в сфере IT. Это привело к расширению функционала Telegram и возможности создания ботов, что обеспечило его популярность среди пользователей. Именно поэтому для разработки нашего чат-бота мы выбрали кроссплатформенный мессенджер Telegram.

2. Анализ

2.1. Анализ предметной области

В современном мире люди хотят быстро получать нужную им информацию. Для многих сайтов уже разработаны боты для более удобной навигации по сайту.

Наш бот позволит пользователю в формате диалога получить интересующую информацию и сократить время поиска по сайту ВГУ.

Основным плюсом разработанного продукта является то, что он размещен в Telegram, что позволит пользователю в любой момент найти ответ на свой вопрос. Наиболее запрашиваемая информация уже содержится в боте в виде реализованных команд на основе быстрых ссылок.

2.2. Анализ задач

Постановка задачи

Основной целью создания Системы является предоставление пользователям удобного и бесплатного сервиса, позволяющего оперативно находить информацию и узнавать о новостях ВГУ.

Задачи проекта

- автоматизация процесса приёма обращений от пользователей;
- разработка интерфейса чат-бота;
- создание механизма информационного взаимодействия между чат-ботом и сайтом ВГУ;
- демонстрация прототипа чат-бота, обеспечивающего:
 - приём обращения от пользователя;
 - интеграцию с существующей информацией;
 - обеспечение пользователю повсеместным доступом к информации – с любого мобильного устройства (смартфона), подключенного к сети Интернет.
- ввод Системы в действие:
 - проведение испытаний Системы;
 - организация работы эксплуатационного персонала;
 - формирование отчетности по результатам испытаний.

Система работает следующим образом:

1. Для начала работы пользователь запускает бота. В ответ на запуск бот выдаёт ему список функций, которые он может выполнить.
2. Затем пользователь может выбрать:
 - вопрос из шаблона:
Тогда бот отправит выбранный запрос в базу данных, получит на него ответ и покажет его клиенту.
 - подписку на рассылку уведомлений:
В этом случае бот отправит этот запрос в базу данных и id клиента будет добавлено в базу для рассылок, что обеспечит отправку уведомлений о новых записях.
 - отписку от рассылки:
Бот отправит в базу запрос об отписке и id пользователя удалится из базы для рассылок.
 - возможность задать свой вопрос для получения развёрнутого ответа, которого нет в базе:
Бот передаст это сообщение менеджеру, который напишет ответ на вопрос и добавит его в базу данных. После этого бот пришлёт ответ пользователю (Обращение к пользователю осуществляется по id).
3. Помимо функций пользователь может завершить чат, тогда бот проведёт закрытие соединения.

2.3. Анализ аналогов

2.3. Анализ средств реализации

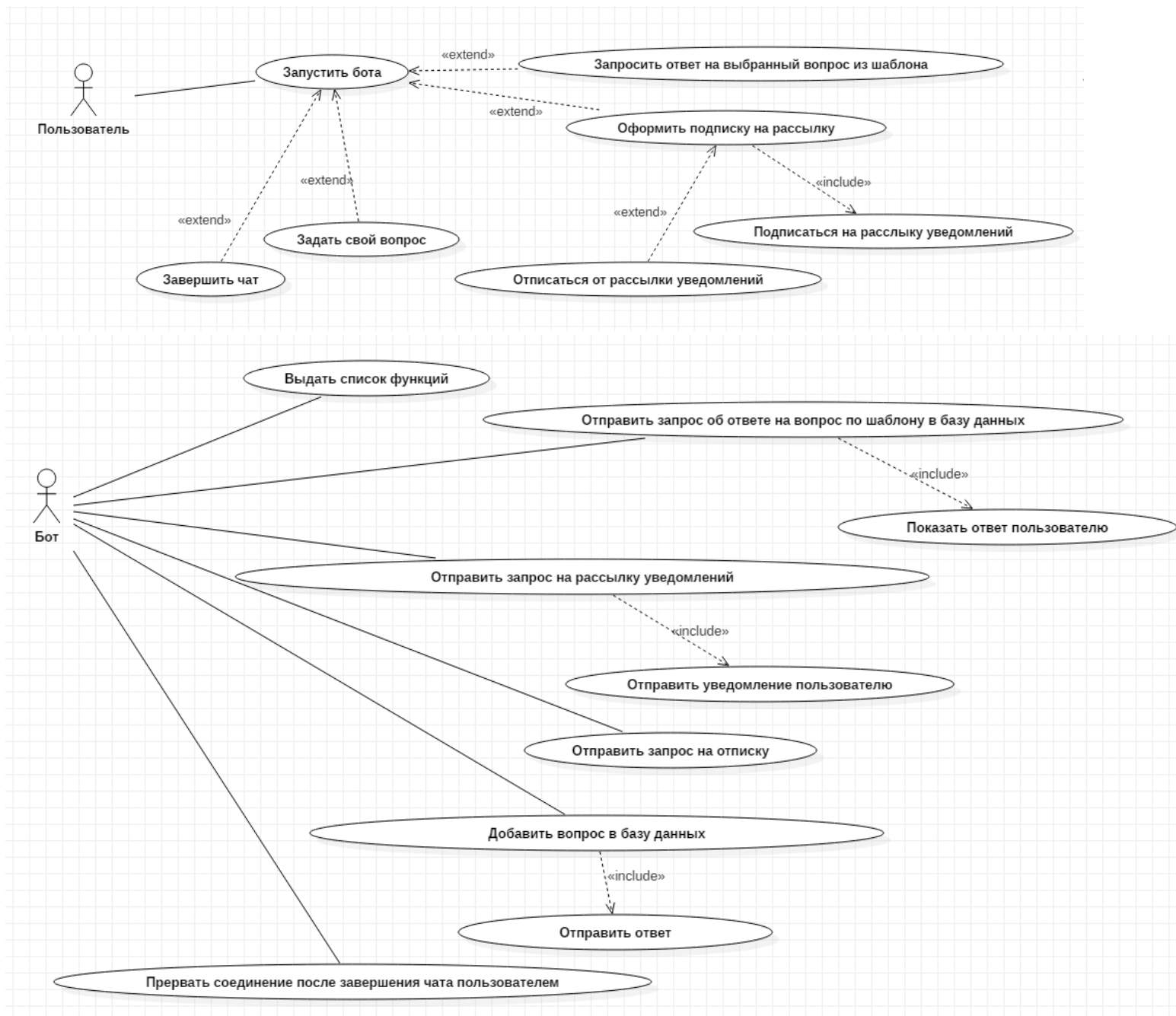
На этапе проектирования для создания диаграмм и схем использовались следующие средства:

- StarUML;
- онлайн-сервис Draw.io;

В качестве языка программирования для реализации приложения был выбран язык Python, встроенные библиотеки pyTelegramBotAPI. СУБД – SQLite3.

2.4. Диаграммы

2.4.1. Диаграмма прецедентов



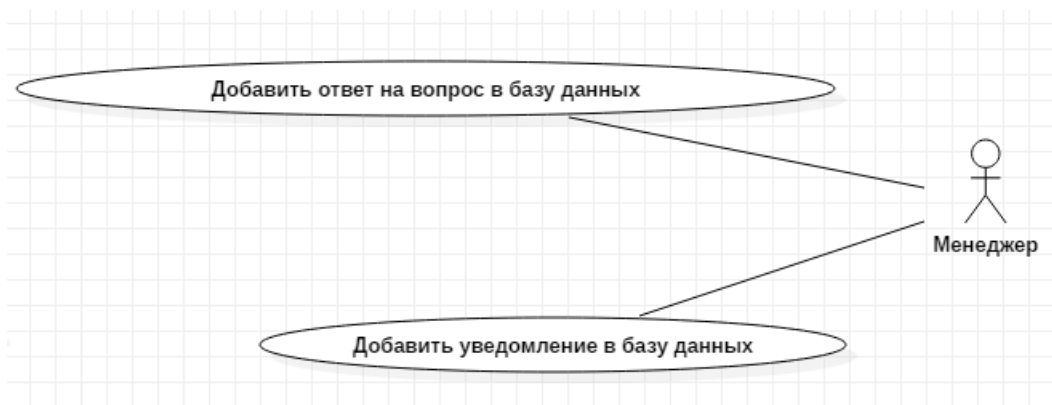


Рис. 1 Use-case диаграмма

2.4.2. Диаграмма классов

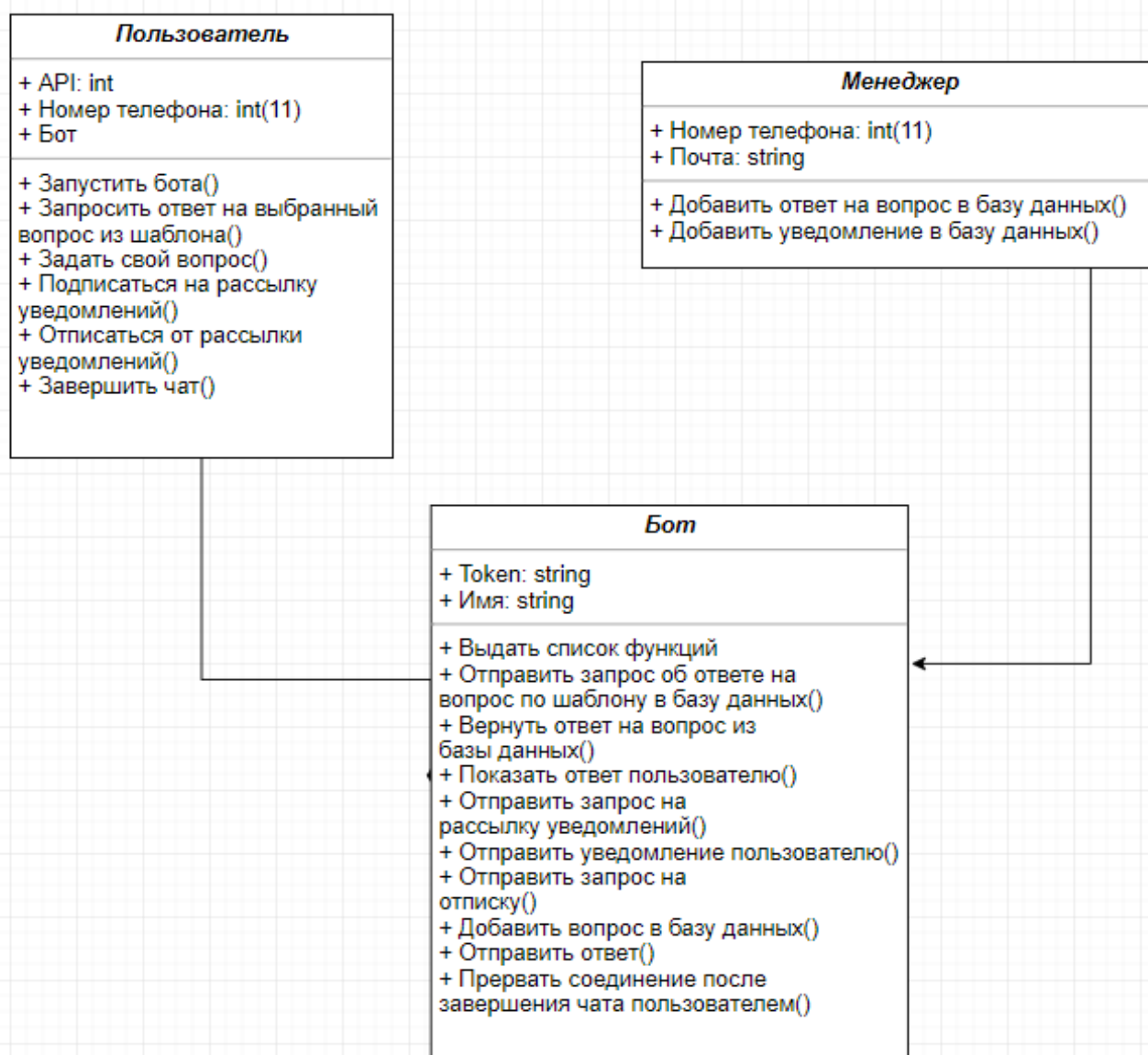


Рис. 2 Диаграмма классов

Описание предполагаемых классов:

Класс пользователь - пользователь может запустить бота, запросить ответ на выбранный вопрос по шаблону, подписаться или отписаться от рассылки, задать свой вопрос.

Класс бот- бот может отправить запрос, вернуть ответ из базы данных и показать его пользователю, отправить запрос в базу данных о подписки на рассылку и отписки, присылать новости каждый день, передать вопросы с развёрнутым ответом в базу данных и вернуть ответ менеджера, закрыть соединение после завершения чата.

Класс Менеджер – менеджер отвечает на развёрнутые вопросы, отправляет новости с сайта каждый день в базу данных.

2.4.3. Диаграмма объектов

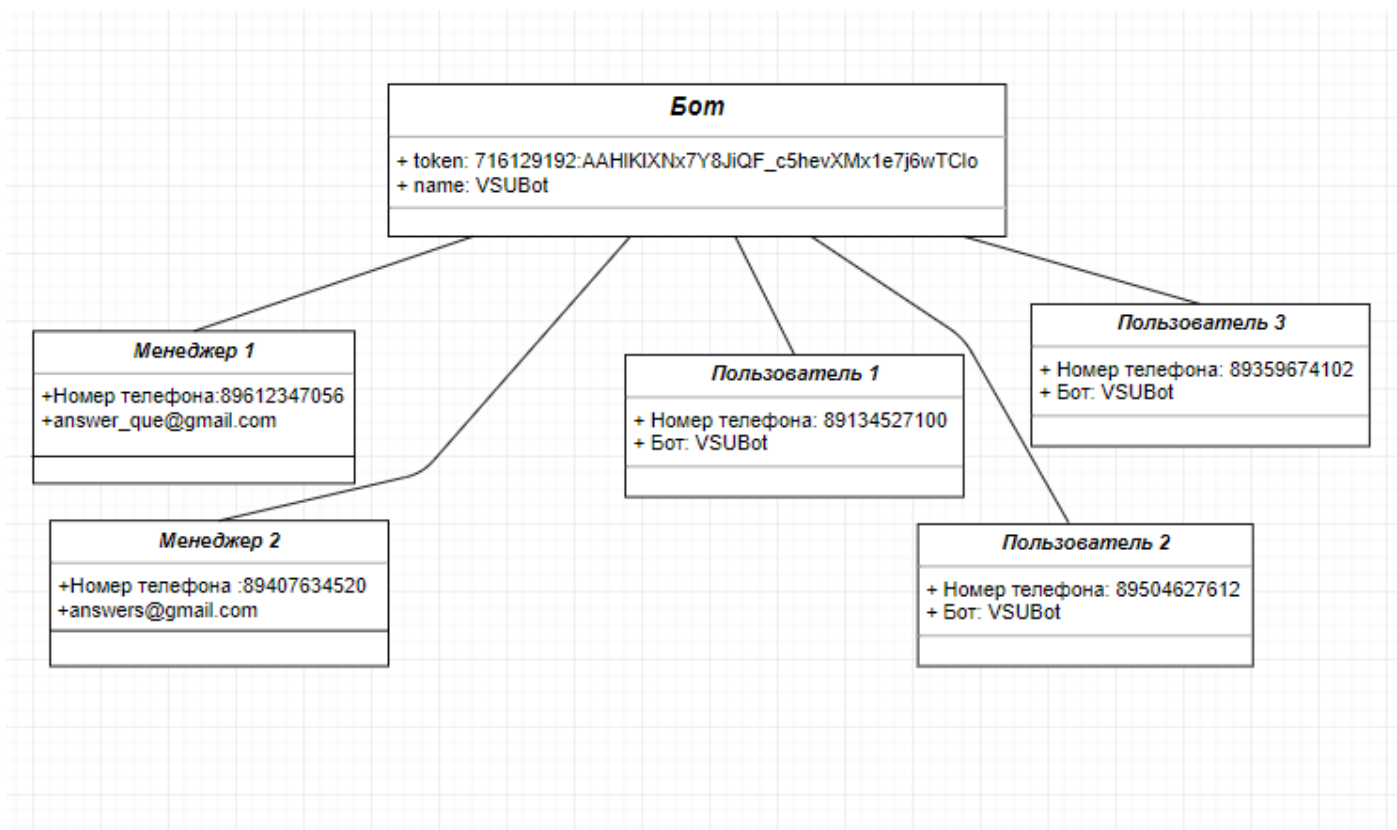


Рис. 3 Диаграмма объектов

2.4.4. Диаграмма коммуникации

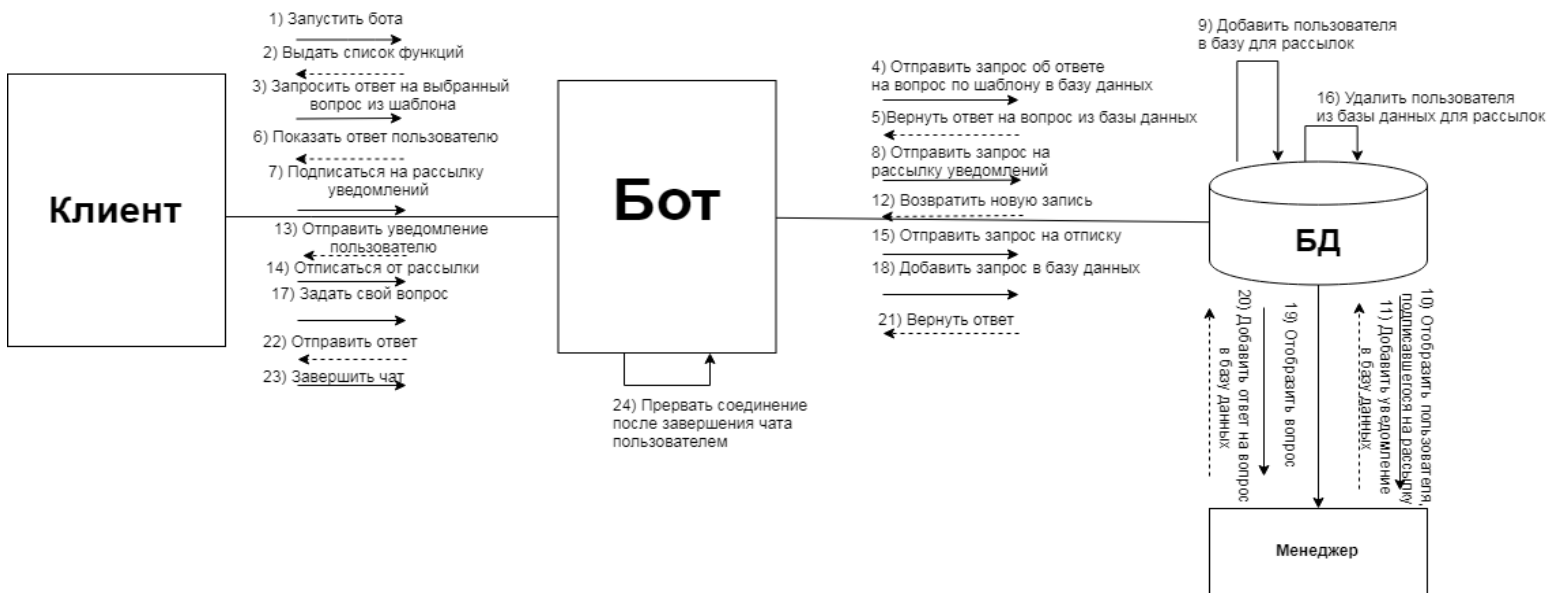


Рис. 4 Диаграмма коммуникаций

На диаграмме коммуникаций также приведена для ситуаций запуска бота, запроса на выбранный вопрос по шаблону, подписки на рассылку, отписки от рассылки, вопрос с развёрнутым ответом и завершение чата

2.4.5. Диаграмма последовательности

Здесь приведена диаграмма последовательностей для ситуаций запуска бота, запроса на выбранный вопрос по шаблону, подписки на рассылку, отписки от рассылки, вопрос с развёрнутым ответом и завершение чата.

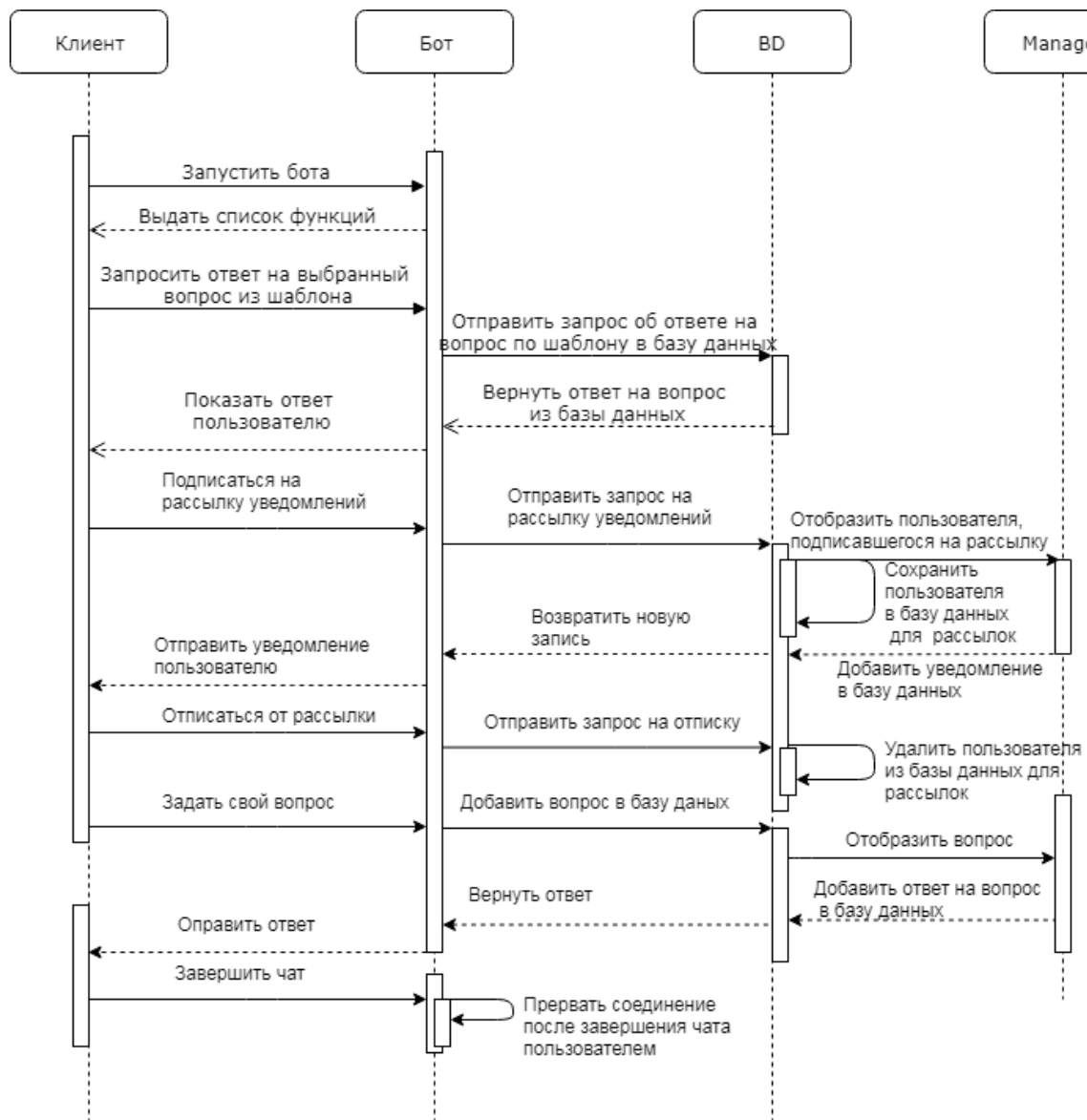


Рис. 5 Диаграмма последовательности

2.4.6. Диаграмма активности

Диаграмма активности является расширением диаграммы состояния

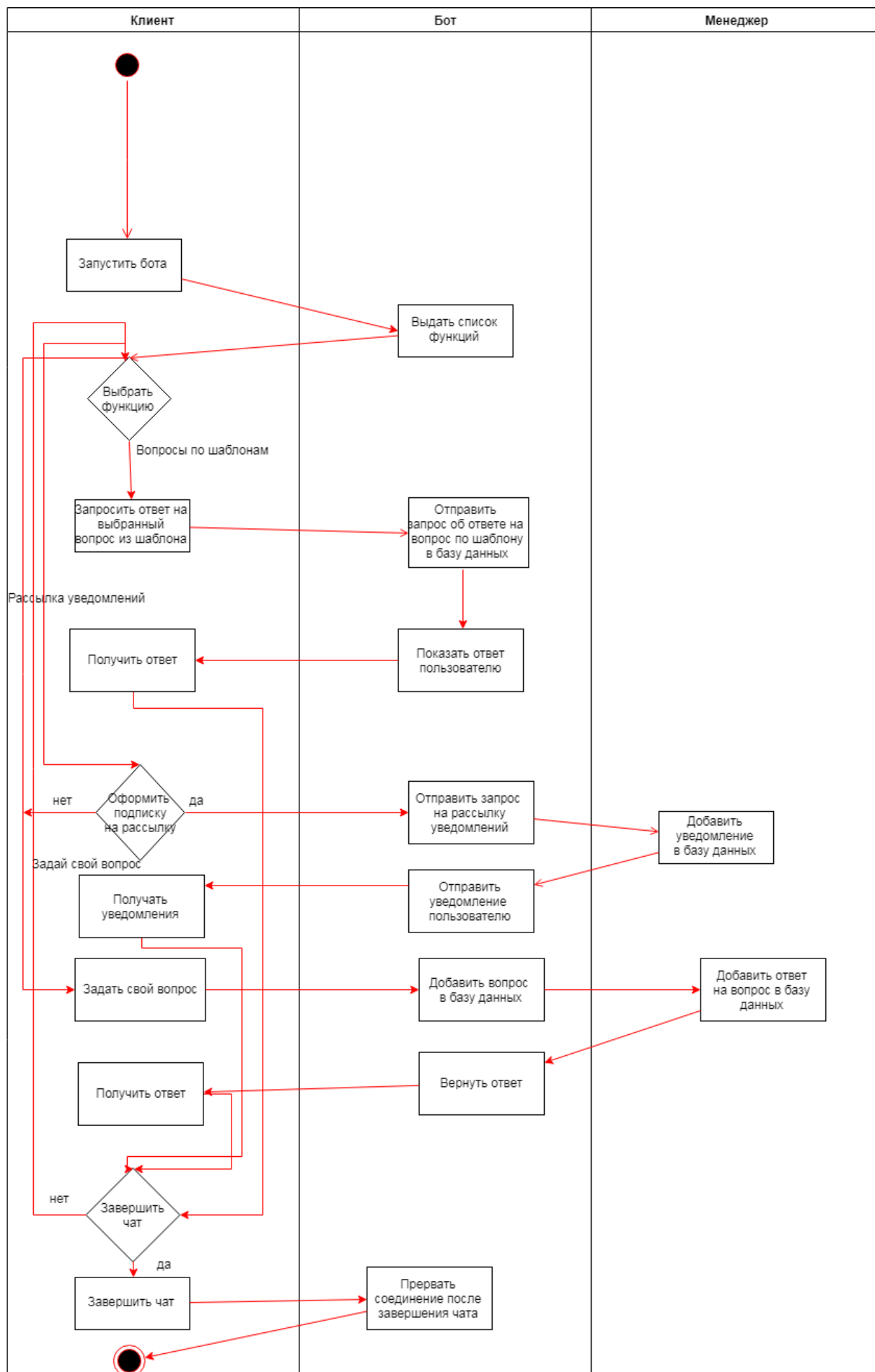


Рис. 6 Диаграмма активности

На диаграмме активности присутствуют 3 части (дорожки):

- Клиент (пользователь),
- Бот (отвечающий на функции),
- Менеджер (отвечающий на развёрнутые вопросы)

2.4.7. Диаграмма развертывания

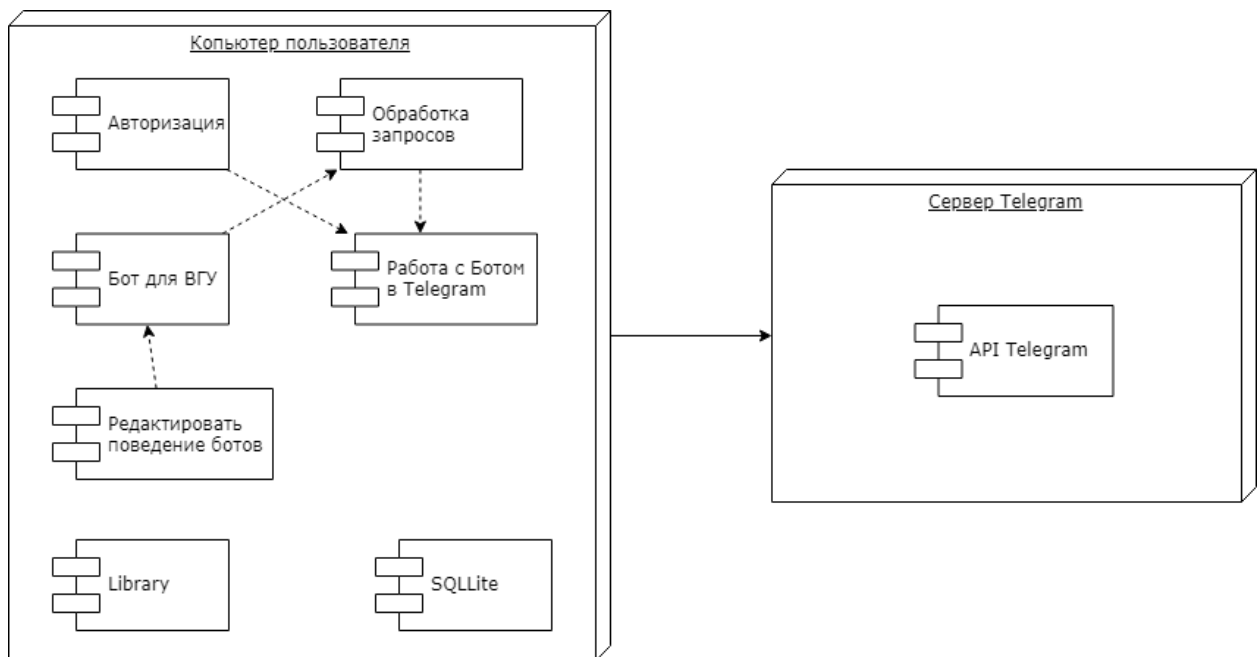


Рис. 7 Диаграмма развёртывания

2.4.8. Диаграмма состояний

На диаграмме состояний приводится набор состояний бота при определённом выборе функций и последовательность переходов от одного состояния к другому.

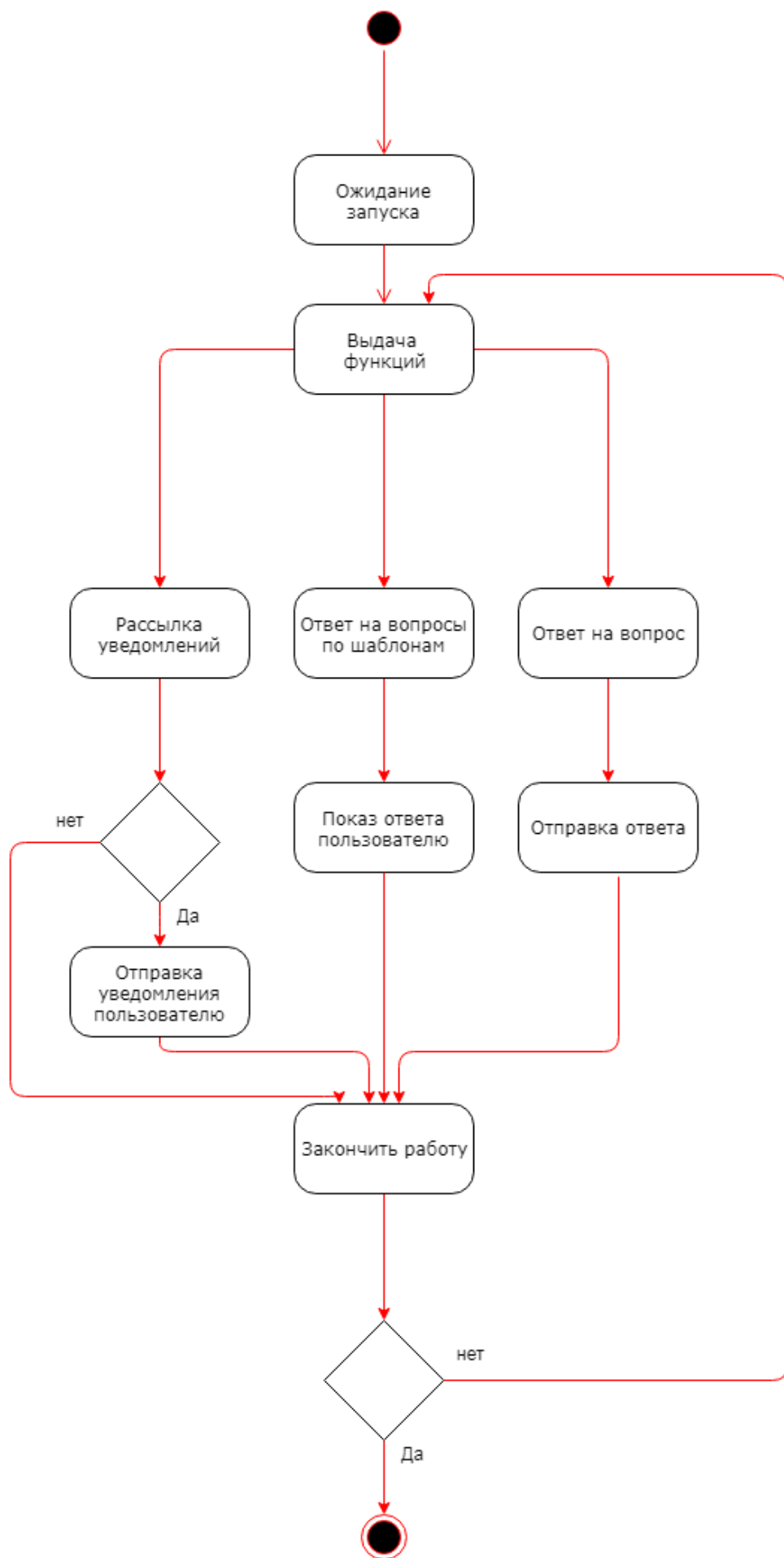


Рис. 8 Диаграмма состояний

3.Основная часть

Язык программирование Python

Python это язык программирования общего назначения, нацеленный в первую очередь на повышение продуктивности самого программиста, нежели кода, который он пишет. Говоря простым языком, на Python можно написать практически что угодно (веб-/настольные приложения, игры, скрипты по автоматизации, комплексные системы расчёта, системы управления жизнеобеспечением и многое другое) без ощутимых проблем. Более того, порог вхождения низкий, а код во многом лаконичный и понятный даже тому, кто никогда на нём не писал. За счёт простоты кода, дальнейшее сопровождение программ, написанных на Python, становится легче и приятнее по сравнению с Java или C++. А с точки зрения бизнеса это влечёт за собой сокращение расходов и увеличение производительности труда сотрудников.

Несомненным достоинством является то, что интерпретатор Python реализован практически на всех платформах и операционных системах. Первым таким языком был Си, однако его типы данных на разных машинах могли занимать разное количество памяти и это служило некоторым препятствием при написании действительно переносимой программы. Python же таким недостатком не обладает.

Так же, немаловажная черта - расширяемость языка, этому придается большое значение и, как пишет сам автор Гвидо ван Россум, язык был задуман именно как расширяемый. Это означает, что имеется возможность совершенствования языка всеми заинтересованными программистами. Интерпретатор написан на Си и исходный код доступен для любых манипуляций. В случае необходимости, можно вставить его в свою программу и использовать как встроенную оболочку. Или же, написав на Си свои дополнения к Python и скомпилировав программу, получить "расширенный" интерпретатор с новыми возможностями.

Следующее достоинство - наличие большого числа подключаемых к программе модулей, обеспечивающих различные дополнительные возможности. Такие модули пишутся на Си и на самом Python и могут быть разработаны всеми достаточно квалифицированными программистами

Модуль TeleBot

Этот модуль является оболочкой над запросами к Telegram Bot API, используется для упрощения и минимизации написанного кода. Все типы

определены в `types.py`. Все они полностью соответствуют определению типов API Telegram, за исключением `from` поля `Message`, которое переименовано в `from_user` (поскольку `from` это зарезервированный токен Python). Таким образом, к таким атрибутам как `message_id` можно обращаться напрямую, например: `message.message_id`. Стоит обратить внимание, что атрибут `message.chat` может принадлежать как определенному пользователю, так и групповому чату.

Объект `Message` также имеет `content_type` атрибут, который определяет тип сообщения. Атрибут `content_type` может быть одним из следующих строк: `text`, `audio`, `document`, `photo`, `sticker`, `video` и так далее. Можно использовать несколько типов в одной функции. Все методы API расположены в классе `TeleBot`. Они переименованы, чтобы следовать общим соглашениям об именах Python. Например: `sendMessage` переименован в `send_message`, `editMessageText` в `edit_message_text`.

Обработчик сообщений - это функция, украшенная декоратором экземпляра `TeleBot`. Обработчики сообщений состоят из одного или нескольких фильтров. Каждый фильтр возвращает `True` или `False` для определенного сообщения, и если возвращается `True`, то обработчик получает разрешение на обработку сообщения. Пример объявления обработчика сообщений приведен в листинге 2.1

Листинг 2.1 – Структура декоратора

```
@bot.message_handler(filters)
def function_name(message):
    bot.reply_to(message, "This is a message handler")
```

Имя функции не имеет никаких ограничений, кроме тех, что установлены языком. Функция должна принимать не более одного аргумента, которым является сообщение, которое должна обрабатывать функция. Поле `filters` - список аргументов, ключевых слов. Фильтр объявляются следующим образом: `имя=аргумент`. Один обработчик может иметь несколько фильтров.

Модуль `CherryPy` – это объектно-ориентированный веб-фреймворк, написанный на языке программирования Python. Спроектирован для быстрой разработки веб-приложений для сети Интернет. Представляет собой надстройку над HTTP-протоколом, но остаётся на низком уровне и не выходит за рамки требований RFC 2616. Модуль `CherryPy` устанавливается точно так же, как и любой другой модуль Python, поэтому в данном случае не приходится говорить о каталоге установки. В отличие от других, более

сложных серверных технологий, модуль CherryPy просто становится доступным для использования всякий раз, как только будет импортирован с помощью инструкции `import`, как и любой другой модуль Python. В действительности CherryPy – это не более чем самостоятельное приложение на языке Python, которое управляет своим собственным многопоточным веб-сервером, благодаря этому выполнить «сценарий на стороне сервера» так же просто, как запустить простую команду в окне терминала. По умолчанию сервер запускается локально на 8080 порту, и выполняет переданную ему функцию, но так же можно задать некоторые параметры вручную, такие как: IP-адрес, порт, открытый и закрытый ключи SSL. Пример задания параметров и запуск веб-сервера приведен в листинге 2.2.

Листинг 2.2 – Пример запуска веб-сервера

```
bot.remove_webhook()
bot.set_webhook(url=WEBHOOK_URL_BASE + WEBHOOK_URL_PATH,
certificate=open(WEBHOOK_SSL_CERT, 'r'))
cherrypy.config.update({
    'server.socket_host': WEBHOOK_LISTEN,
    'server.socket_port': WEBHOOK_PORT,
    'server.ssl_module': 'builtin',
    'server.ssl_certificate': WEBHOOK_SSL_CERT,
    'server.ssl_private_key': WEBHOOK_SSL_PRIV
})
cherrypy.quickstart(WebhookServer(), WEBHOOK_URL_PATH, {'/': {}})
```

Модуль Requests Requests – библиотека Python, которая просто выполняет HTTP-запросы (HyperText Transfer Protocol). В листинге 2.3 приведен пример получения информации о веб странице через запрос.

Листинг 2.3 – Получение публичного тайм-лайн GitHub

```
import requests
resp = requests.get('https://api.github.com/events')
```

После выполнения этого кода, в объекте `resp` хранится вся необходимая информация об этом объекте.

4.Тестирование

Цели:

1. Выявление дефектов на ранних стадиях.
2. Проверить, все ли требования были выполнены.
3. Убедиться, что приложение работает так, как ожидает пользователь.
4. Уверенность в качестве.

Почему тестирование необходимо:

1. Помогает избежать риски, так как мы прописываем условия, при которых могут возникнуть трудности.
2. Выявить дефекты, тем самым повысить качество продукта.

Нами были проведены такие тестовые мероприятия:

1. Планирование. Определить цели тестирования и подход для их достижения.
2. Контроль и управление. Составить план тестирования, что выполнено и будет выполнено на определённых стадиях разработки.
3. Тест Анализ. Рассмотреть особенности текст – кейсов, расставить приоритеты.
4. Тест дизайн. Разработать тест – кейсы. Прописать ожидаемые результаты.
5. Выполнение тестов. Протестировать приложение. Сравнить полученные результаты с ожидаемыми. Проверить, чтобы тесты соответствовали документации
6. Завершение тестов. Сбор данных после тестирования. Тест-report (доклады о тестах). Сохранить информацию для дальнейших проектов.

Так как нами была использована Циклический жизненный цикл проекта, то тестирование будет тоже проходить по определённым итерациям:

1. Первая итерация: Smoke testing для базовых функций
2. Вторая итерация: была составлена Decision Table для функции «Вопросы по шаблону»

3. Третья итерация: Cheat list для функции «Задай свои вопросы», тест для функции «Рассылка уведомлений»
4. Четвёртая итерация: State Transition Testing для всех функций бота
5. Пятая итерация: Conformation testing для проверки тест-кейсов, которые не удалось пройти ранее

Далее

4.1 Smoke testing

Тестирование охватывает основные функций программного обеспечения, но ни одну из них в глубине. Результат этого теста используется, чтобы решить, следует ли продолжить дальнейшее тестирование. Если испытание проходит успешно, продолжайте дальнейшие испытания. Если это не помогло, остановите дальнейшие тесты и попросите новую сборку с необходимыми исправлениями. Если приложение сильно сломано, детальное тестирование может быть пустой тратой времени и усилий.

На Рис. 3.1 Smoke testing для тестирования были выбраны основные команды (кнопка СТАРТ, функции: «Вопросы по шаблонам», «Задай свой вопрос», «Рассылка уведомлений», «Завершить чат»)
Все тесты были пройдены успешно. Можем перейти к следующему этапу тестирования

№	Test Name	Actions	Expected Results	Actual results	Priority	Verdict	Bugs	Tester	Date
1	Кнопка СТАРТ	Зайти в VSU_BOT Нажать на СТАРТ	Сообщение от бота: Привет! Я помогу тебе лучше узнать ВГУ и найти полезную информацию. Ты можешь воспользоваться актуальными вопросами и сразу получить на них ответ. А можешь задать свой, но ответ тебе придёт не сразу. Подписавшись на уведомления, ты сможешь получать ежедневные новости и анонсы о мероприятиях.	Рис. 3.1.1 Кнопка старт Сообщение от бота: Привет! Я помогу тебе лучше узнать ВГУ и найти полезную информацию. Ты можешь воспользоваться актуальными вопросами и сразу получить на них ответ. А можешь задать свой, но ответ тебе придёт не сразу. Подписавшись	High	passed		Толчеева А.Г.	01.05.2019

			Надеюсь, тебе понравится!	на уведомления, ты сможешь получать ежедневные новости и анонсы о мероприятиях. Надеюсь, тебе понравится!					
2	Кнопка СТАРТ	Зайти в VSU_BOT Нажать на СТАРТ	Внизу кнопки с функциями: Вопросы по шаблонам Задай свой вопрос Рассылка уведомлений Завершить чат	Рис. 3.1.1 Кнопка старт Внизу кнопки с функциями: Вопросы по шаблонам Задай свой вопрос Рассылка уведомлений Завершить чат	High	passed		Толчеева А.Г.	01.05.2019
3	Функция "Вопросы по шаблонам"	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам"	Сообщение от бота: "Вы можете получить ответы на такие вопросы:"	Рис. 3.1.2 Функция «Вопросы по шаблону» Сообщение от бота: "Вы можете получить ответы на такие вопросы:"	High	passed		Толчеева А.Г.	01.05.2019
4	Функция "Задай свой вопрос"	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Задай свои вопросы"	Сообщение от бота: "Что ты хочешь узнать о ВГУ?"	Рис. 3.1.3 Функция «Задай свой вопрос» Сообщение от бота: "Что ты хочешь узнать о ВГУ?"	High	passed		Толчеева А.Г.	01.05.2019
5	Функция "Рассылка уведомлений"	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка"	Сообщение от бота: "/on_notifications - подписаться на рассылку уведомлений /off_notifications - отписаться от рассылки уведомлений"	Рис. 3.1.4 Функция «Рассылка уведомлений» Сообщение от бота: "/on_notifications - подписаться на рассылку"	High	passed		Толчеева А.Г.	02.05.2019

				уведомлений /off _notifications- отписаться от рассылки уведомлений"				
6	Функция "Завершит ь чат"	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Заверши ть чат"	Сообщение от бота: "/off - отписаться от бота и завершить чат"	Рис. 3.1.5 Функция «Завершить чат» Сообщение от бота: "/off - отписаться от бота и завершить чат"	High	passed	Толчеева А.Г.	02.05.2019

Рис. 3.1 Smoke testing

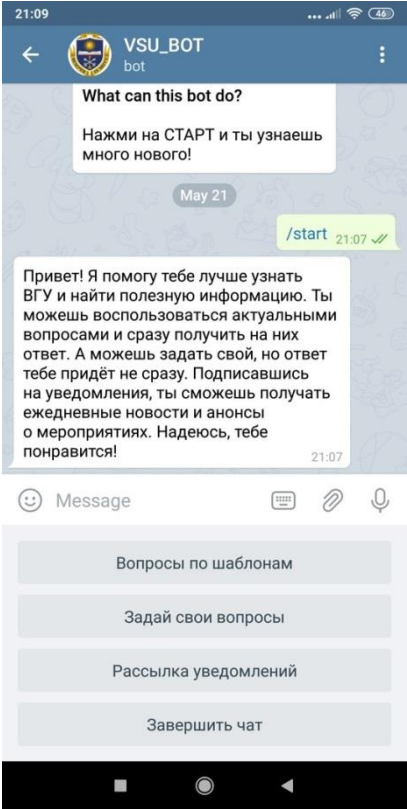


Рис. 3.1.1 Кнопка
старт

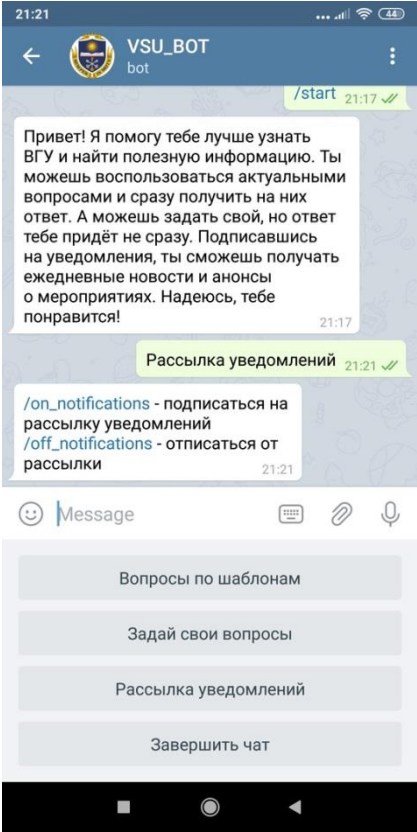


Рис. 3.1.4 Функция
«Рассылка уведомлений»

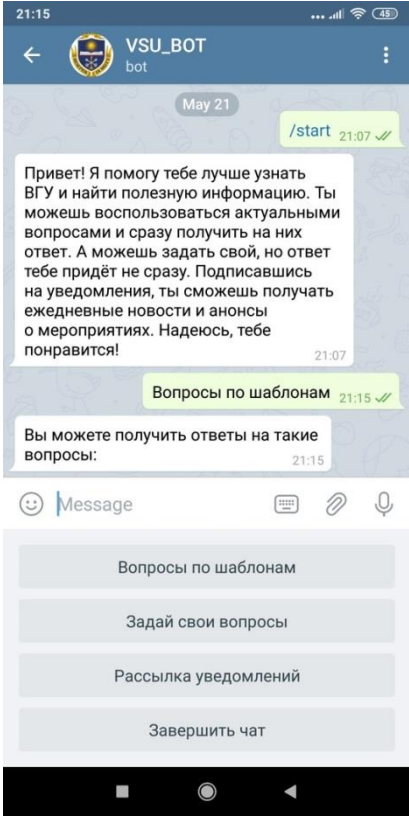


Рис. 3.1.2 Функция
«Вопросы по шаблону»

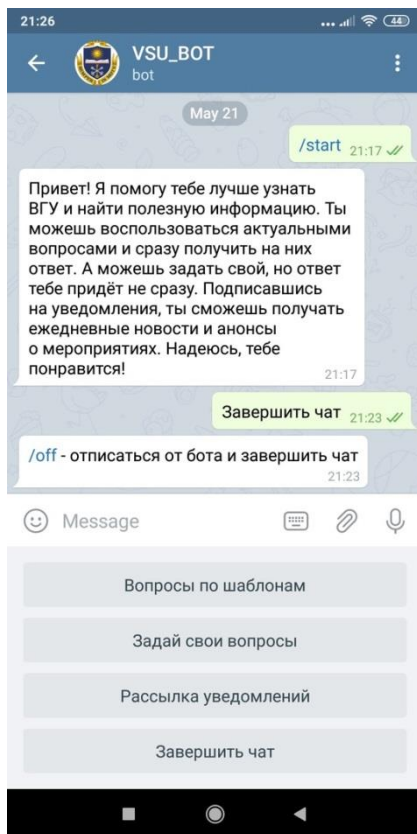


Рис. 3.1.5 Функция «Завершить чат»

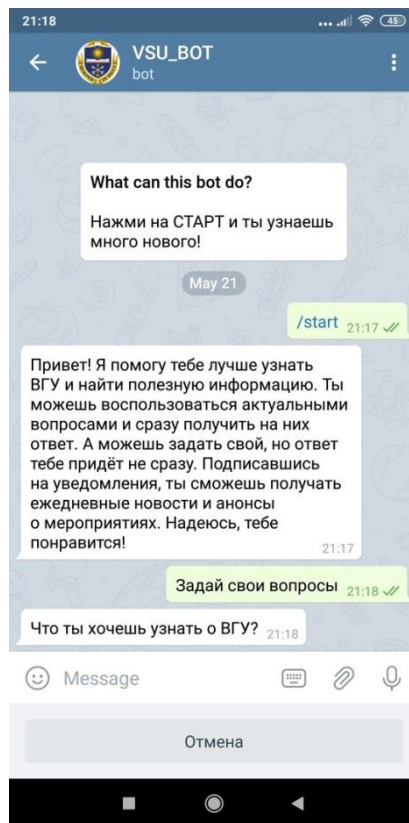


Рис. 3.1.3 Функция «Задай свой вопрос»

4.2 Decision Table

Design Techniques бывает двух видов:

Static Test Design Techniques

1. Informal Reviews – пробежаться глазами по приложению, без документации о особых знаний о функционале.
2. Walkthroughs – Поочерёдное выполнение шагов для сбора информации.
3. Technical Review – Процесс обнаружения ошибок, дискуссия в команде и достижение единого мнения.
4. Inspection – Визуальная экспертиза, нарушение стандартов, несоответствие документации.

Dynamic Test Design

Black-Box. Основан на анализе соответствующей тестовой базы (например, документов формальных требований, спецификаций, вариантов использования, пользовательских историй или бизнес-процессов). Применимы как для функционального, так и нефункционального тестирования. Концентрируются на «входных» и «выходных» значениях.

White – Box. Основывается на внутренней структуре (код, архитектура, рабочие потоки и / или потоки данных в системе) или реализации системы. Тщательность тестирования белого ящика может быть измерена через структурное покрытие. Структурное покрытие - это степень, в которой определенный тип структурного элемента был испытан тестами, и выражается в процентах от типа охватываемого элемента.

Experience – based. Используют опыт разработчиков, тестировщиков и пользователей; для разработки, реализации и выполнения тестов. Часто сочетаются с методами белого и черного ящика.

Specification based (black - box)

- Decision Table. Таблицы решений - хороший способ записать сложные бизнес-правила, которые должна реализовать система.
- При создании таблиц решений тестер определяет условия (чаще input) и результирующие действия (чаще output) системы.
- Они формируют строки таблицы, обычно с условиями вверху и действиями внизу.

- Каждый столбец соответствует правилу принятия решения, которое определяет уникальную комбинацию условий, которая приводит к выполнению действий, связанных с этим правилом.
- Значения условий и действий обычно отображаются как логические значения (true или false) или дискретные значения (например, красный , зеленый , синий), но также могут быть числами или диапазонами чисел.
- Эти различные типы условий и действий объединены в одной таблице.

На Рис. 3.2 Decision Table было проведено тестирование корректного отображения ответов бота на быстрые ссылки. Тест-кейсы были составлены на функции «Вопросы по шаблонам». Были использованы следующие ссылки на вопросы: /how_old_VSU - сколько лет ВГУ? /department - сколько факультетов в ВГУ? /which_departments_at_VSU - какие факультеты есть в ВГУ? /document - какой перечень документов нужен для поступления? /branches - какие филиалы есть у ВГУ? /deadline_for_submission - какой срок подачи документов для поступления?"

№	Test Name	Actions	Expected Results	Actual results	Priority	Verdict	Bugs	Tester	Date
Вопросы по шаблонам									
1	Функция "Вопросы по шаблонам"	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам"	Сообщение от бота: "Вы можете получить ответы на такие вопросы: /how_old_VSU - сколько лет ВГУ? /department - сколько факультетов в ВГУ? /which_departments_at_VSU - какие факультеты есть в ВГУ? /document - какой перечень документов нужен для поступления? /branches - какие филиалы есть у	Рис. 3.2.1 Функция "Вопросы по шаблонам" Сообщение от бота: "Вы можете получить ответы на такие вопросы: /how_old_VSU - сколько лет ВГУ? /department - сколько факультетов в ВГУ? /which_departments_at_VSU - какие факультеты есть в ВГУ? /document - какой перечень	High	passed		Фирсова И.А.	09.05.2019

			ВГУ? /deadline_for_submission - какой срок подачи документов для поступления?"	документов нужен для поступления? /branches - какие филиалы есть у ВГУ? /deadline_for_submission - какой срок подачи документов для поступления?"				
3 1	Сколько лет ВГУ?	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам" Выбрать /how_old_VSU - сколько лет ВГУ?	Сообщение от бота: "100 лет. ВГУ был основан 18 мая 1918"	Рис. 3.2.2 Вопрос «Сколько лет ВГУ?» Сообщение от бота: "100 лет. ВГУ был основан 18 мая 1918"	High	passed	Фирсова И.А.	9.05.2019
3 2	Сколько факультетов в ВГУ?	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам" Выбрать /department - сколько факультетов в ВГУ?	Сообщение от бота: "18 факультетов"	Рис. 3.2.3 Вопрос «Сколько факультетов в ВГУ?» Сообщение от бота: "18 факультетов"	High	passed	Фирсова И.А.	9.05.2019
3 3	Какие факультеты есть в ВГУ?	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам" Выбрать /document - какой перечень документов нужен для поступления?	Факультет географии, геоэкологии и туризма Геологический факультет Исторический факультет Математический факультет Медико-биологический факультет Факультет прикладной математики, информатики и	Рис. 3.2.4 Вопрос «Какие факультеты есть в ВГУ?» Факультет географии, геоэкологии и туризма Геологический факультет Исторический факультет Математический факультет Медико-биологический	High	passed	Фирсова И.А.	10.05.2019

			механики Факультет романо- германской филологии Физический факультет Филологический факультет Химический факультет Экономический факультет Юридический факультет Факультет журналистики Факультет философии и психологии Фармацевтическ ий факультет Факультет компьютерных наук Факультет международных отношений Факультет военного образования (Военная кафедра, Учебный военный центр)	факультет Факультет прикладной математики, информатики и механики Факультет романо- германской филологии Физический факультет Филологический факультет Химический факультет Экономический факультет Юридический факультет Факультет журналистики Факультет философии и психологии Фармацевтическ ий факультет Факультет компьютерных наук Факультет международных отношений Факультет военного образования (Военная кафедра, Учебный военный центр)			
--	--	--	--	--	--	--	--

3 4	Какой перечень документов нужен для поступления?	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам" Выбрать /which_departments_at_VSU - какие факультеты есть в ВГУ?	<p>Перечень основных документов:</p> <ul style="list-style-type: none"> - Документ, удостоверяющий личность, гражданство (паспорт, военный билет, свидетельство о рождении, пр.); - Документ об образовании (аттестат, диплом, пр.); - 4 фотографии 3х4 см; - Согласие абитуриента на обработку его персональных данных (образец); - Заявление о приеме на обучение в бакалавриат и специалитет (образец); - Заявление о приеме на обучение в магистратуру (образец); - Договор об образовании на обучение по основной профессиональной образовательной программе <p>Более полную информацию можно получить по ссылке http://www.abitur.vsu.ru/admission/examples_and_blanks</p>	<p>Рис. 3.2.5 Вопрос «Какой перечень документов нужен для поступления?»</p> <p>Перечень основных документов:</p> <ul style="list-style-type: none"> - Документ, удостоверяющий личность, гражданство (паспорт, военный билет, свидетельство о рождении, пр.); - Документ об образовании (аттестат, диплом, пр.); - 4 фотографии 3х4 см; - Согласие абитуриента на обработку его персональных данных (образец); - Заявление о приеме на обучение в бакалавриат и специалитет (образец); - Заявление о приеме на обучение в магистратуру (образец); - Договор об образовании на обучение по основной профессиональной образовательной программе <p>Более полную информацию можно получить по ссылке http://www.abitur.vsu.ru/admission/examples_and_blanks</p>	High	passed	Фирсова И.А.	10.05.2019
--------	--	---	--	--	------	--------	--------------	------------

3 5	Какие филиалы есть у ВГУ?	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам" Выбрать /branches - какие филиалы есть у ВГУ?	Борисоглебский филиал Лискинский филиал	Рис. 3.2.6 Вопрос «Какие филиалы есть у ВГУ?» Борисоглебский филиал Лискинский филиал	High	passed	Фирсова И.А.	10.05.2019
3 6	Какой срок подачи документов для поступления?	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Вопросы по шаблонам" Выбрать /deadline_for_submission - какой срок подачи документов для поступления?	По программам бакалавриата срок начала приема документов 20 июня, срок завершения – 26 июля. По программам магистратуры срок начала приема документов 20 июня, срок завершения – 5 августа.	Рис. 3.2.7 Вопрос «Какой срок подачи документов для поступления?» По программам бакалавриата срок начала приема документов 20 июня, срок завершения – 26 июля. По программам магистратуры срок начала приема документов 20 июня, срок завершения – 5 августа.	High	passed	Фирсова И.А.	10.05.2019

Рис. 3.2 Decision Table

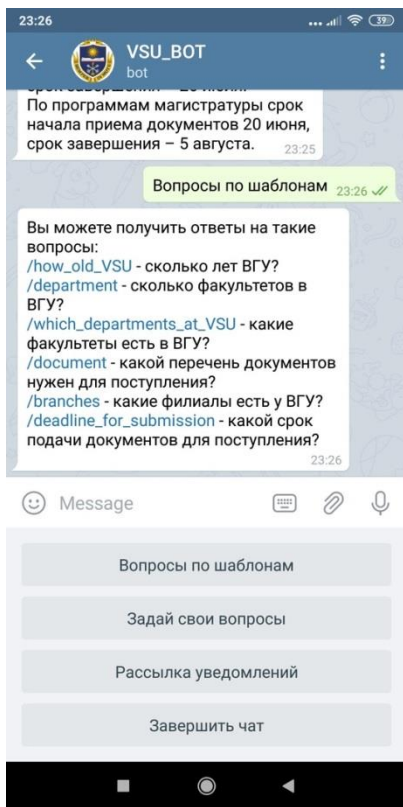


Рис. 3.2.1 Функция "Вопросы по шаблону"

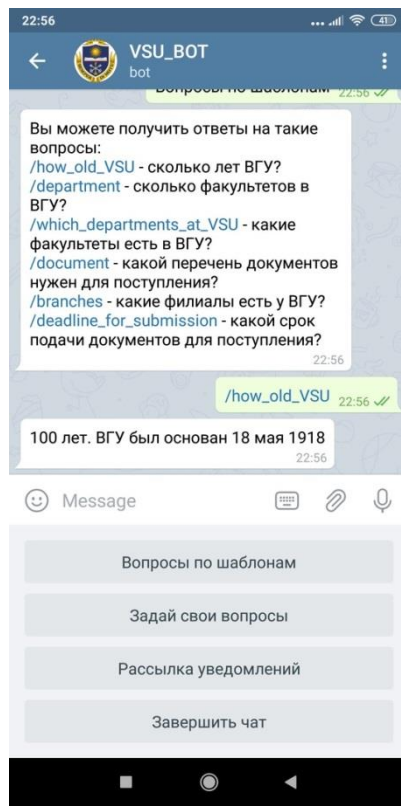


Рис. 3.2.2 Вопрос «Сколько лет ВГУ?»

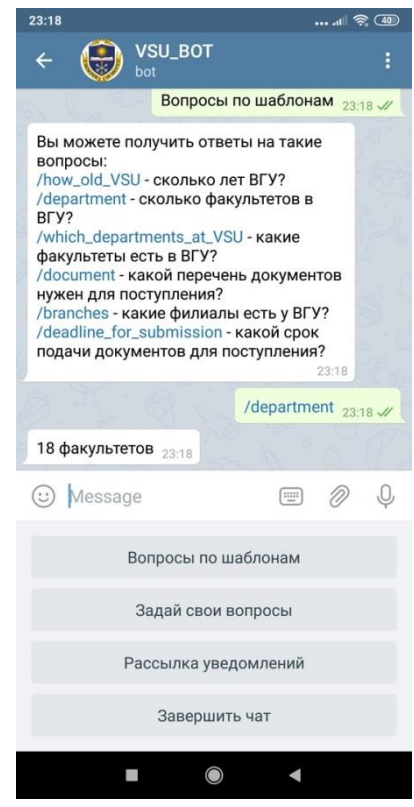


Рис. 3.2.3 Вопрос «Сколько факультетов в ВГУ»

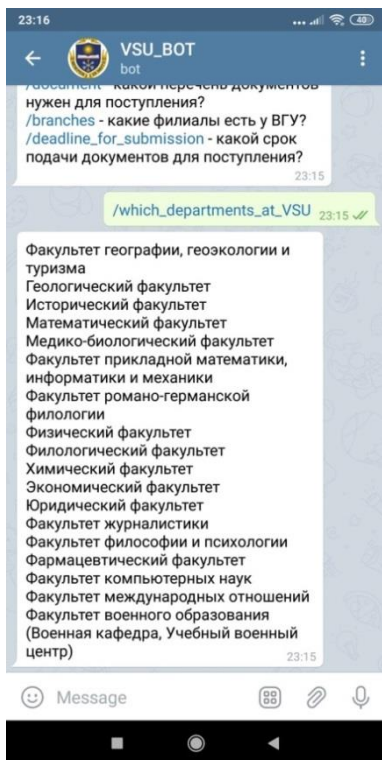


Рис. 3.2.4 Вопрос «Какие факультеты есть в ВГУ?»

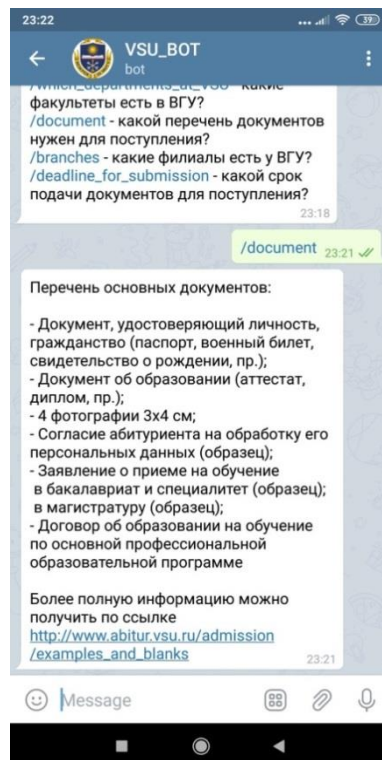


Рис. 3.2.5 Вопрос «Какой перечень

документов нужен для поступления?»

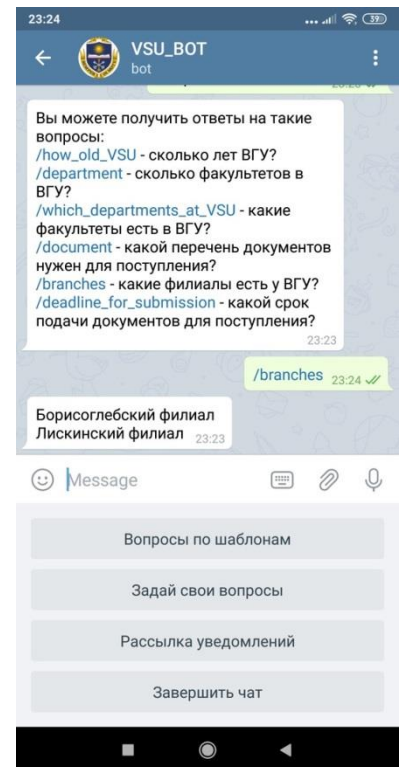


Рис. 3.2.6 Вопрос
«Какие филиалы есть у
ВГУ?»

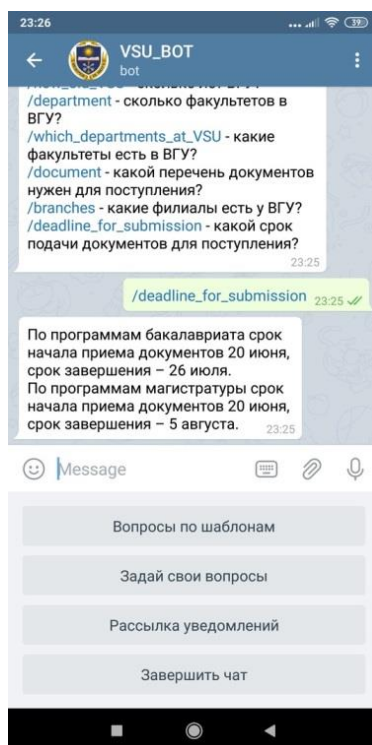


Рис. 3.2.7 Вопрос
«Какой срок подачи
документов для поступления?»

4.3 Check-List

Check-List - это документ, описывающий что должно быть протестировано.

Критерии Check-List:

- функциональность
- вложенные уровни
- формализованность
- описывает элементы, которые надо протестировать
- не описывает ожидаемый результат

В нашем проекте мы создадим Check-List (Рис 3.3.1) для функции «Задай свои вопросы». Тестировать будем поле ввода вопроса. Так как это вводит сам пользователь, следовательно, это редактируемое поле. Далее по этому шаблону создадим тесты и проверим выполняемость.

#	Текстовое поле			заголовок	
#	placeholder		Medium		Message фоном
#	допустимые значения			заголовок	
#	длина сообщения = 6		Medium		Минимальная длин вопроса
#	6 < длина сообщения < 50		High		Обязательно для проверки
#	длина сообщения = 50		Low		Максимальная длина вопроса
#	Русский алфавит А-Я		High		
#	Английский алфавит А-Z		High		
#	недопустимые значения			заголовок	
#	Цифры		High		
#	Специальные символы @#\$%^&()"№;:*		Low		
#	Юникодные символы ® ☒ § © ☹		Low		
#	длина сообщения < 6		Medium		
#	длина сообщения > 50		Low		

Рис. 3.3.1 Check-List

По нашему Check-List мы создали тест-кейсы для функции «Задай свой вопрос» (рис. 3.3.2)

№	Test Name	Actions	Expected Results	Actual results	Priority	Verdict	Bugs	Comments	Tester	Date
Задай свои вопросы										
7	Строка для сообщений	Зайти в VSU_BOT Нажать Старт	Строка для сообщений с placeholder "Message"	Рис. 3.3.2.1. Строка для сообщений Строка для сообщений с placeholder "Message"	Medium	passed			Фирсова И. А.	15.05.2019
Допустимые значения										
8	Длина сообщения = 6	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, длина = 6	Сообщение от Бота: "Готово!"	Рис. 3.3.2.2. Длина сообщения = 6 Сообщение от Бота: "Готово!"	Medium	passed			Фирсова И.А.	15.05.2019
9	Длина сообщения от 6 до 50	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, 6 < длина < 50	Сообщение от Бота: "Готово!"	Рис. 3.3.2.3. Длина сообщения от 6 до 50 Сообщение от Бота: "Готово!"	High	passed			Фирсова И.А.	15.05.2019
10	Длина сообщения > 50	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, длина = 50	Сообщение от Бота: "Готово!"	Рис. 3.3.2.4. Длина сообщения > 50 Сообщение от Бота: "Готово!"	High	passed			Фирсова И.А.	15.05.2019

11	Сообщение на русском языке	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение на русском языке	Сообщение от Бота: "Готово!"	Рис. 3.3.2.4. Сообщение на русском языке Сообщение от Бота: "Готово!"	High	passed		Были использованы данные прошлого кейса	Фирсова И.А.	15.05.2019
12	Сообщение на английском языке	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение на английском языке	Сообщение от Бота: "Готово!"	Рис. 3.3.2.5. Сообщение на английском языке Сообщение от Бота: "Готово!"	High	passed			Фирсова И.А.	15.05.2019
Недопустимые значения										
13	Сообщение состоит из цифр	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение из цифр	Сообщение от Бота: "Извини я тебя не понимаю.."	Рис. 3.3.2.6. Сообщение состоит из цифр Сообщение от Бота: "Извини я тебя не понимаю.."	High	passed			Фирсова И.А.	16.05.2019
14	Сообщение состоит из специальных символов	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение из специальных символов @#\$%^&()" №;:.*	Сообщение от Бота: "Извини я тебя не понимаю.."	Рис. 3.3.2.7. Сообщение состоит из специальных символов Сообщение от Бота: "Извини я тебя не понимаю.."	Low	passed			Фирсова И.А.	16.05.2019

15	Сообщение состоит из юникодных символов	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение из юникодных символов ® ☒ § © ☹	Сообщение от Бота: "Извини я тебя не понимаю.."	Рис. 3.3.2.8. Сообщение состоит из юникодных символов Сообщение от Бота: "Готово"	Low	failed	1 [Задай свои вопросы] сообщение из юникодных символов принимается	Фирсова И.А.	16.05.2019
16	Длина сообщения < 6	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, длина < 6	Сообщение от Бота: "Сообщение слишком короткое, попробуй ещё раз"	Рис. 3.3.2.9. Длина сообщения < 6 Сообщение от Бота: "Извини я тебя не понимаю.."	Medium	failed	2 [Задай свои вопросы] при вводе сообщения длиной < 6 сообщение от бота можно улучшить Enhancement - улучшение. Если бот будет отвечать "Сообщение слишком короткое", то пользователю будет понятна причина некорректности вопроса"	Фирсова И.А.	16.05.2019
17	Длина сообщения >50	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, длина >50	Сообщение от Бота: "Сообщение слишком длинное, попробуй ещё раз"	Рис. 3.3.2.10. Длина сообщения >50 Сообщение от Бота: "Извини я тебя не понимаю.."	Low	failed	3 [Задай свои вопросы] при вводе сообщения длиной >50 сообщ	Фирсова И.А.	16.05.2019

						ение от бота можн о улучш ить		
--	--	--	--	--	--	---	--	--

Рис. 3.3.2 Тест-кейсы для функции «Задай свои вопросы»

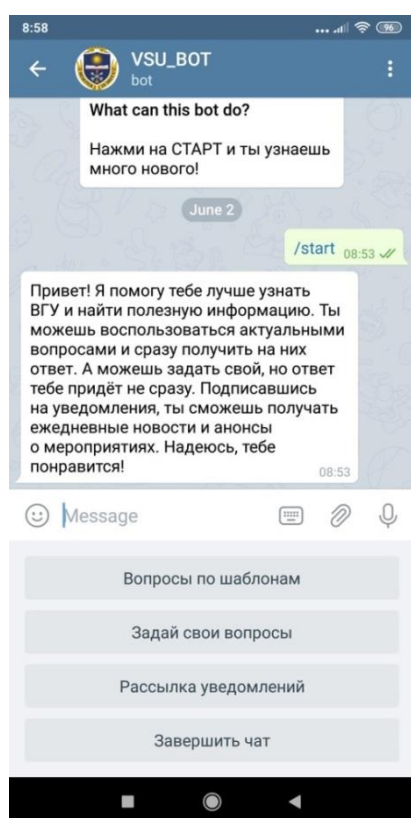


Рис. 3.3.2.1. Строка для сообщений

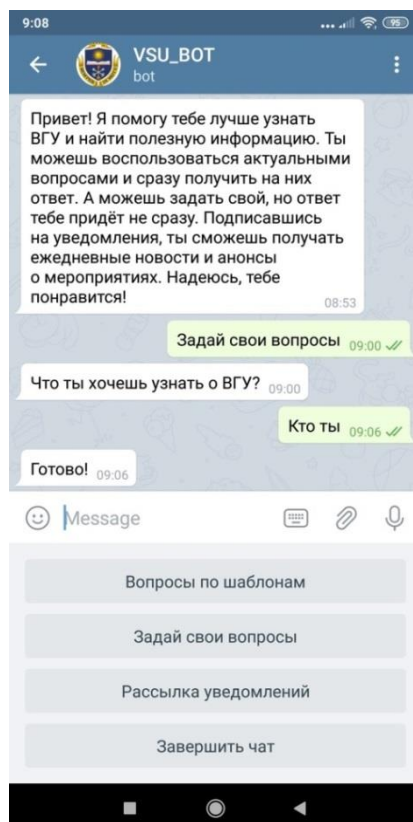


Рис. 3.3.2.2. Длина сообщения = 6

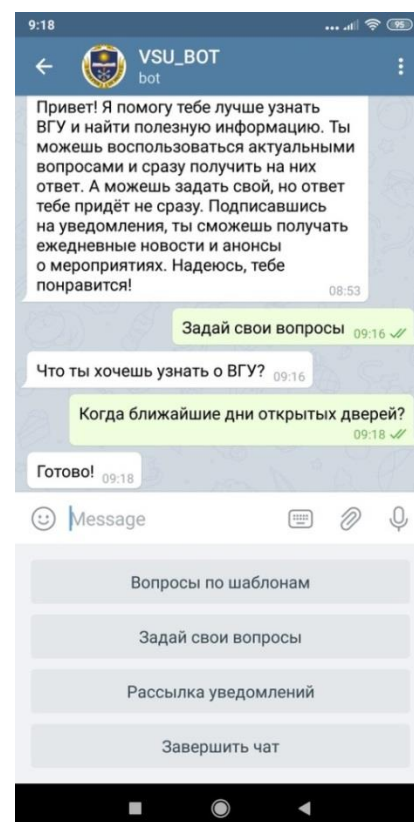


Рис. 3.3.2.3. Длина сообщения от 6 до 50

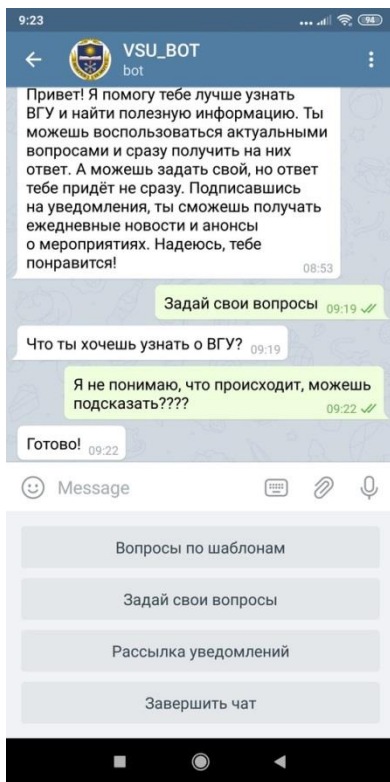


Рис. 3.3.2.4. Длина сообщения > 50

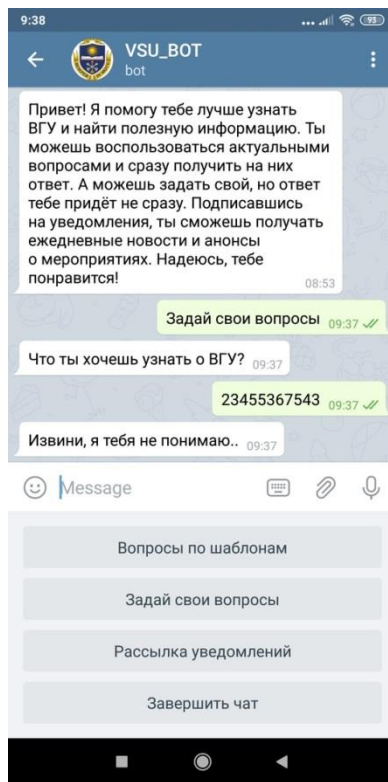


Рис. 3.3.2.6. Сообщение состоит из цифр

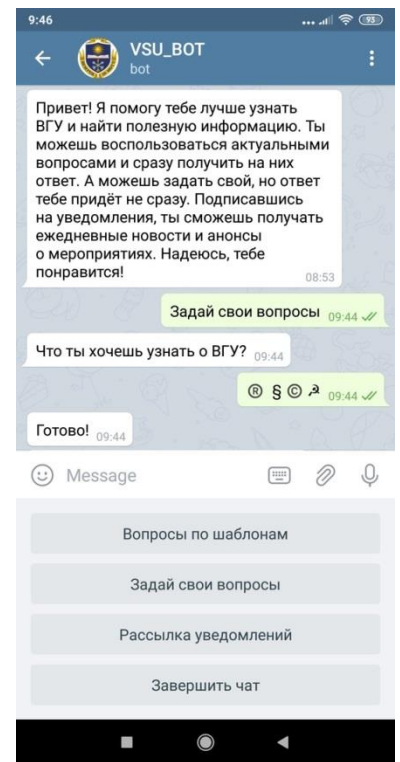


Рис. 3.3.2.8. Сообщение состоит из юникодных символов

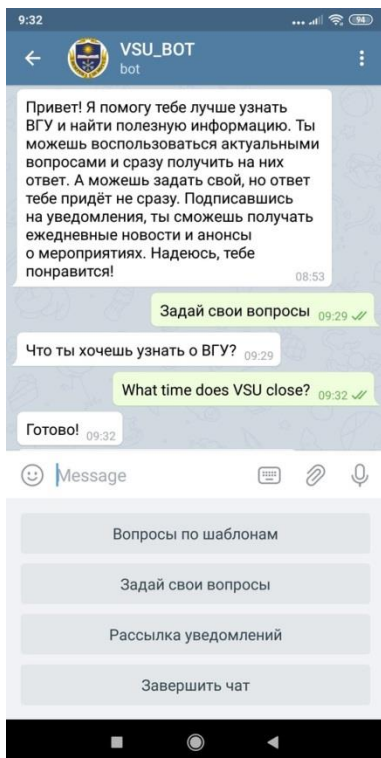


Рис. 3.3.2.5. Сообщение на английском языке

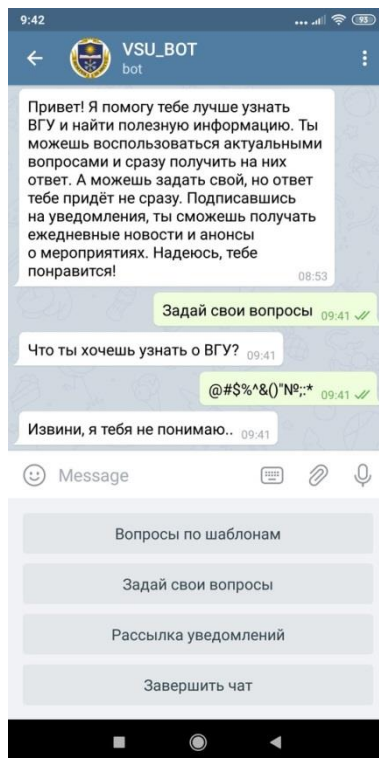


Рис. 3.3.2.7. Сообщение состоит из специальных символов

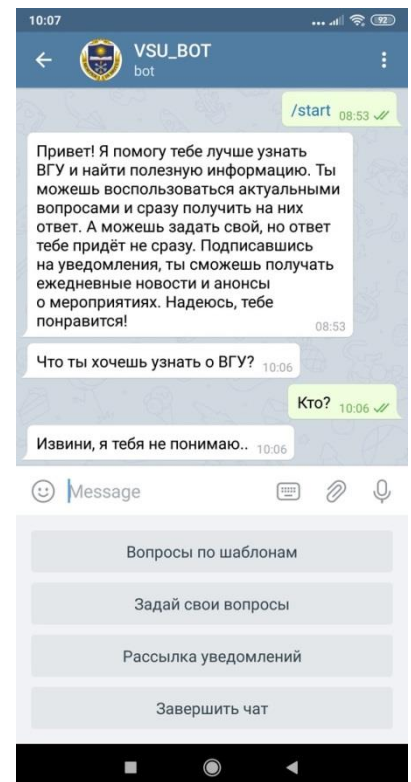


Рис. 3.3.2.9. Длина сообщения < 6

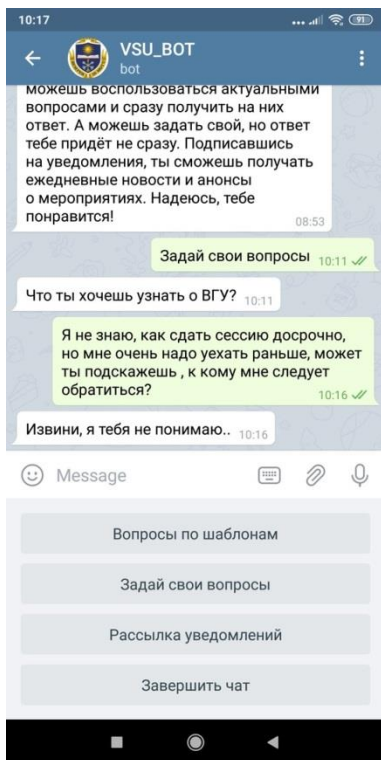


Рис. 3.3.2.10. Длина сообщения >50

По итогам данного тестирования были обнаружены три Bug: один Minor и два Enhancement. Чтобы правильно их завести, воспользуемся Bug Report Template – шаблон для оформления Bug.

Добавлен:

Название:

Статус:

Приоритет:

Назначение:

1. Версия приложения
2. Платформа, на которой проводилось тестирование
3. Набор компонентов, которые тестировали
4. Шаги воспроизведения Bug
5. Фактический результат, который мы получили
6. Ожидаемый результат, который надо было получить

Добавлен: Фирсова И.А.

1 [Задай свои вопросы] сообщение из юникодных символов принимается

Статус: New

Приоритет: Minor

Назначение: Толчеева А.Г.

Версия:

1. Версия:
ver: 1
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги воспроизведения
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Задай свой вопрос"
 4. Ввести сообщение из юникодных символов ® ☒ § © ☯ ㄅ
5. Фактический результат:
Сообщение от Бота: "Готово"
6. Ожидаемы результат:
Сообщение от Бота: "Извини я тебя не понимаю.."

Добавлен: Фирсова И.А.

2 [Задай свои вопросы] при вводе сообщение длинной < 6 сообщение от бота можно улучшить

Статус: New

Приоритет: Enhancement

Назначение: Толчеева А.Г.

1. Версия:
ver: 1
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги воспроизведения
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Задай свой вопрос"
 4. Ввести сообщение, длина < 6
5. Фактический результат:
Сообщение от Бота: "Извини я тебя не понимаю.."
6. Ожидаемы результат:
Сообщение от Бота: "Сообщение слишком короткое, попробуй ещё раз"

Добавлен: Фирсова И.А.

3 [Задай свои вопросы] при вводе сообщение длинной > 50 сообщение от бота можно улучшить

Статус: New

Приоритет: Enhancment

Назначение: Толчеева А.Г.

1. Версия:
ver: 1
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги воспроизведения
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Задай свой вопрос"
 4. Ввести сообщение, длина > 50
5. Фактический результат:
Сообщение от Бота: "Извини я тебя не понимаю.."
6. Ожидаемы результат:
Сообщение от Бота: "Сообщение слишком длинное, попробуй ещё раз"

На данном этапе была так же составлена Decision Table для функции «Рассылка уведомлений» (Рис 3.3.3)

№	Test Name	Actions	Expected Results	Actual results	Priority	Verdict	Bugs	Comments	Tester	Date
Рассылка уведомлений										
51	Подписаться на рассылку, когда ещё не подписан	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка" Выбрать /on_notifications	Сообщение от Бота: "Поздравляю! Вы подписались на рассылку уведомлений! Теперь вы ничего не пропустите!"	Рис. 3.3.3.1 Подписаться на рассылку, когда ещё не подписан Сообщение от Бота: "Поздравляю! Вы подписались на рассылку уведомлений! Теперь вы ничего не пропустите!"	Hight	passed			Фирсова И.А.	20.05.2019
52	Отписка от рассылки, когда уже подписан	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка" Выбрать /on_notifications Выбрать /off_notifications	Сообщение от Бота: "Вы отписались от рассылки уведомлений"	Рис. 3.3.3.2. Отписка от рассылки, когда уже подписан Сообщение от Бота: "Вы отписались от рассылки уведомлений"	Hight	passed			Фирсова И.А.	20.05.2019
53	Подписка на рассылку, когда уже подписан	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка" Выбрать /on_notifications Ещё раз выбрать /on_notifications	Сообщение от Бота: "Вы уже подписаны на рассылку"	Рис. 3.3.3.3. Подписка на рассылку, когда уже подписан Сообщение от Бота: "Поздравляю! Вы подписались на рассылку уведомлений! Теперь вы ничего не пропустите!"	Medium	failed	3 [Рассылка уведомлений] пользователь подписывается на рассылку дважды		Фирсова И.А.	20.05.2019

5 4	Отписка от рассылки, когда ещё не подписан	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка" Выбрать /off_notifications	Сообщение от Бота: "Вы не были подписаны на рассылку, поэтому не можете отписаться"	Рис. 3.3.3.4. Отписка от рассылки, когда ещё не подписан Сообщение от Бота: "Вы отписались от рассылки уведомлений"	Medium	failed	4 [Рассылка уведомлений] пользователь отписывается от рассылки уведомлений, даже если не был подписан	Фирсова И.А.	20.05. 2019
--------	--	--	---	--	--------	--------	--	--------------	-------------

Рис. 3.3.3 Рассылка уведомлений

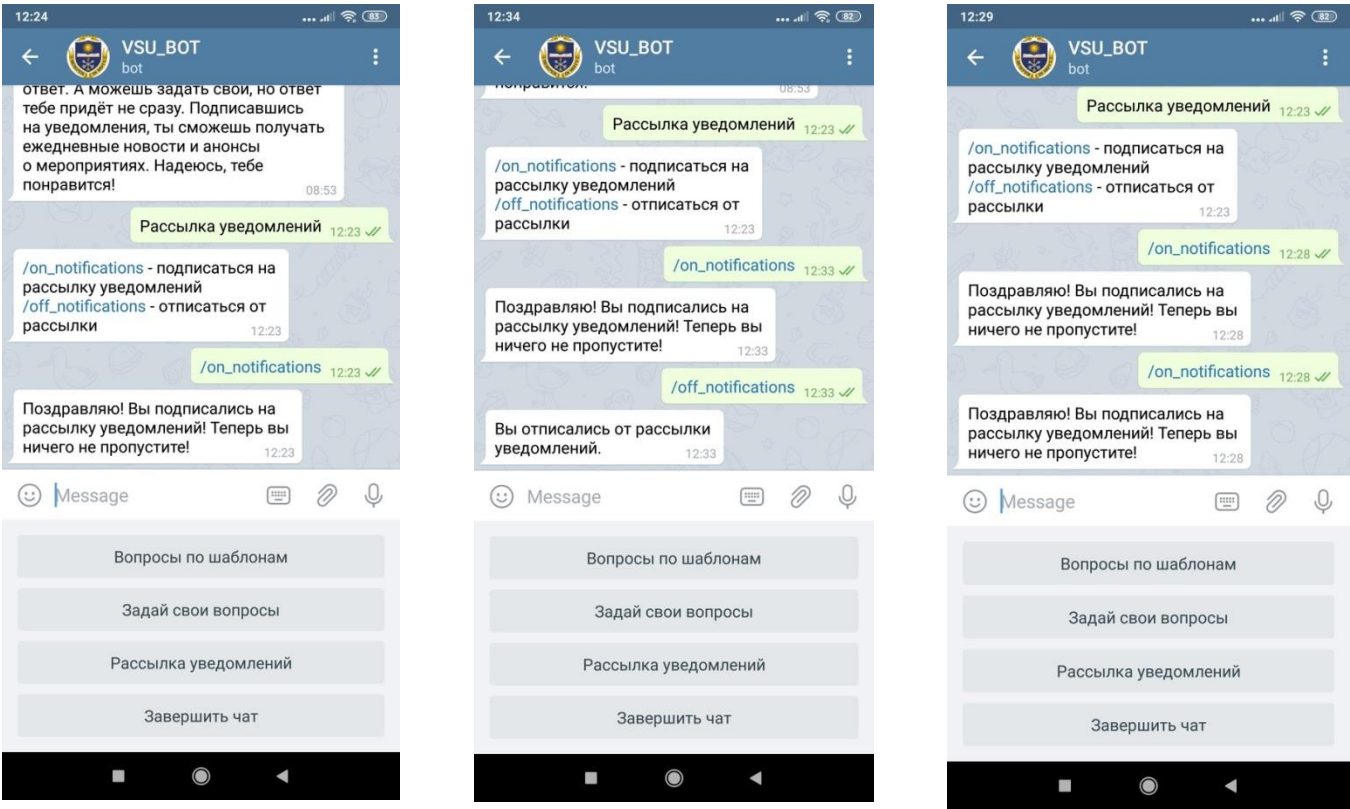


Рис. 3.3.3.1
Подписаться на рассылку, когда ещё не подписан

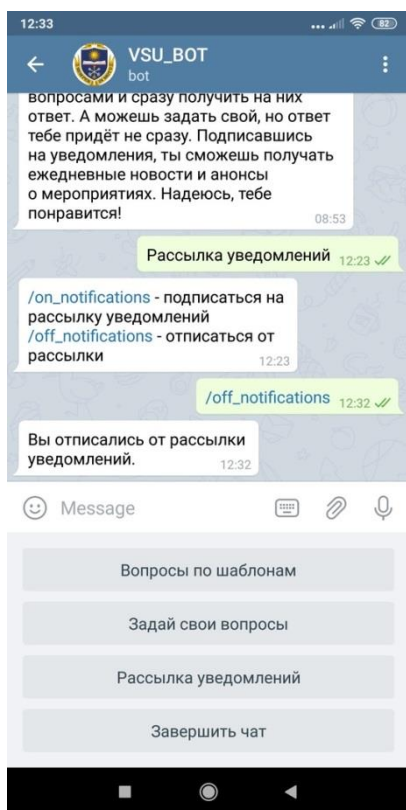


Рис. 3.3.3.2. Отписка от
Рис. 3.3.3.3. Подписка
на рассылку, когда уже
подписан

рассылки, когда
уже подписан

Оформление Bug

Добавлен: Фирсова И.А.

4 [Рассылка уведомлений] пользователь подписывается на рассылку дважды

Статус: New

Приоритет: Normal

Назначение: Толчеева А.Г.

1. Версия:
ver: 1
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги воспроизведения
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Рассылка"
 4. Выбрать /on_notifications
 5. Ещё раз выбрать /on_notifications
5. Фактический результат:
Сообщение от Бота: "Поздравляю! Вы подписались на рассылку уведомлений! Теперь вы ничего не пропустите!"
6. Ожидаемы результат:
Сообщение от Бота: "Вы уже подписаны на рассылку"

Добавлен: Фирсова И.А.

5 [Рассылка уведомлений] пользователь отписывается от рассылки уведомлений, даже если не был подписан

Статус: New

Приоритет: Normal

Назначение: Толчеева А.Г.

1. Версия:
ver: 1
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги воспроизведения
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Рассылка"
 4. Выбрать /off_notifications
5. Фактический результат:
Сообщение от Бота: " Вы отписались от рассылки уведомлений "

6. Ожидаемы результат:

Сообщение от Бота: " Вы не были подписаны на рассылку, поэтому не можете отписаться"

3.4 State Transition Testing

Тестирование переходов состояний (state transition testing): Разработка тестов методом черного ящика, в котором сценарии тестирования строятся на основе выполнения корректных и некорректных переходов состояний.

Состояние (state)- это состояние приложения, в котором оно ожидает 1 или более событий. Состояние помнит входные данные полученные до этого и показывает как приложение будет реагировать на полученные события. События могут вызывать смену состояния и/или инициировать действия.

Переход (transition) - представляет переход одного состояния в другое, происходящий по событию.

Тестирование переходов состояния для бота:

- Пользователь нажал на кнопку старт
 - Бот показал приветственное сообщение
 - Бот показал набор функций
 - Пользователь выбрал функцию «Вопросы по шаблонам»
 - Бот показал сообщение с вопросами
 - Пользователь выбрал вопрос: «Сколько лет ВГУ?»
 - Бот ответил на вопрос
 - Пользователь выбрал вопрос: «Сколько факультетов в ВГУ?»
 - Бот ответил на вопрос
 - Пользователь выбрал вопрос: «Какие факультеты есть в ВГУ?»
 - Бот ответил на вопрос
 - Пользователь выбрал вопрос: «Какой перечень документов нужен для поступления?»
 - Бот ответил на вопрос
 - Пользователь выбрал вопрос: «Какие филиалы есть у ВГУ?»

- Бот ответил на вопрос
- Пользователь выбрал вопрос: «Какой срок подачи документов для поступления?»
- Бот ответил на вопрос
- Пользователь выбрал функцию: «Задай свои вопросы»
 - Пользователь задал корректный вопрос
 - Бот ответил: «Готово!»
 - Пользователь задал некорректный вопрос
 - Бот ответил: «Извини, я тебя не понимаю..»
- Пользователь выбрал функцию: «Рассылка уведомлений»
 - Пользователь подписался на рассылку
 - Бот присылает ежедневные сообщения
 - Пользователь отписался от рассылки
- Пользователь выбрал функцию: «Завершить чат»
- Соединение завершилось

4.5 Confirmation testing

Подтверждающее тестирование или повторное тестирование: если тест не пройден из-за дефекта, то сообщается об этом дефекте и ожидается новая версия программного обеспечения, в которой этот дефект был исправлен. В этом случае нам нужно выполнить тест еще раз, чтобы подтвердить, был ли дефект исправлен или нет.

Во время тестирования Bug имеет следующий жизненный цикл (Рис. 3.5.1)

Вначале создаётся баг, который имеет статус New, затем назначается на разработчика, который будет его исправлять, разработчик оценивает баг и принимает решение по его исправлений, присваивая ему статус, подходящий к ситуации. Если баг исправлен, тестирующий присваивает ему статус Verified и закрывает его. Если же баг не был исправлен, то он переоткрывается, назначается на этого же разработчика, если он не может исправить, то переназначается на другого. Так же может перейти в состояние новый и пройти весь цикл заново.

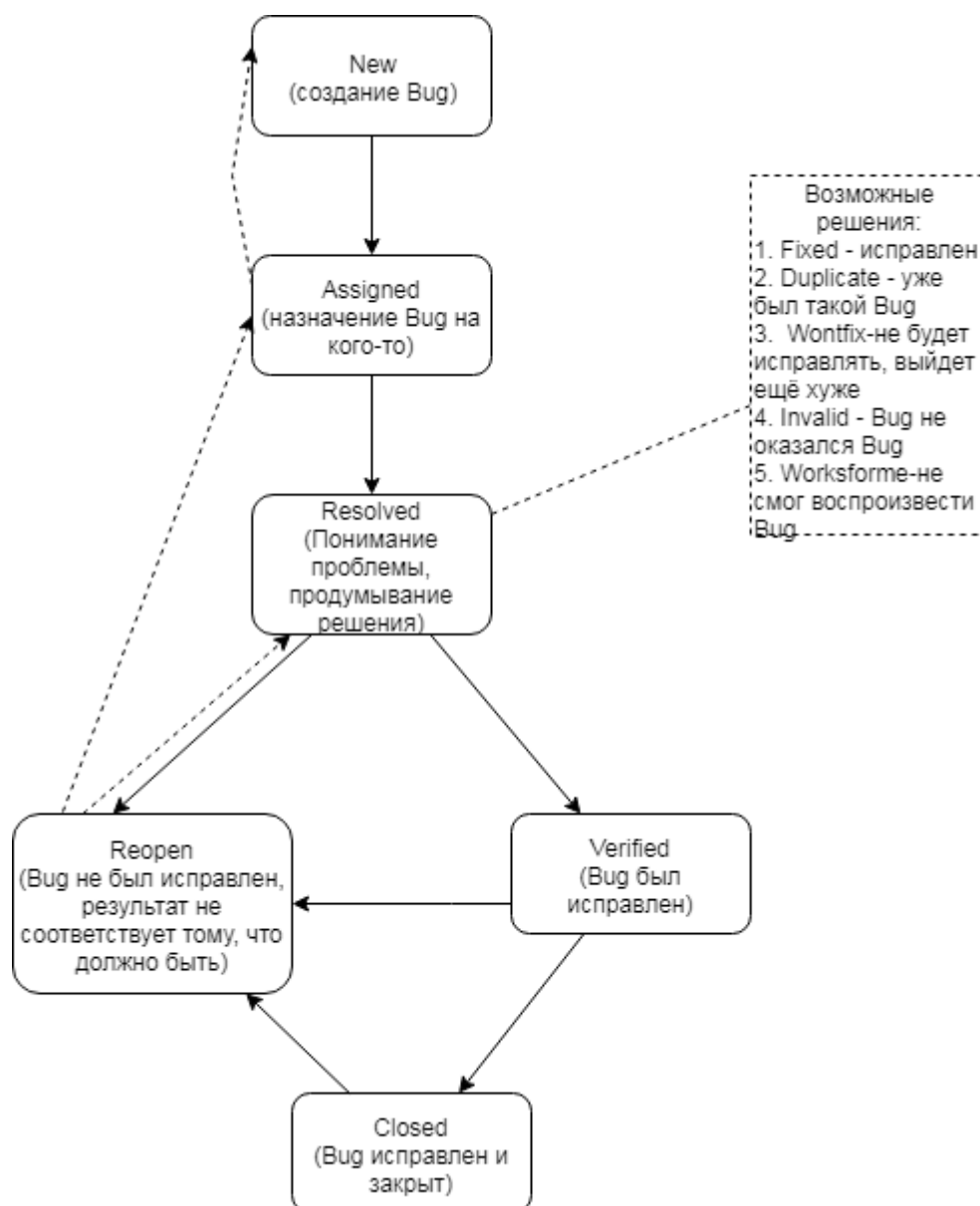


Рис. 3.5.1 Жизненный цикл Bug

После наших тестирований было заведено 5 багов, на этапе заключительного тестирования все они были исправлены (Рис. 3.5.2)

№	Test Name	Actions	Expected Results	Actual results	Priority	Verdict	Bugs	Comments	Tester	Date
15	Сообщение состоит из юникодных символов	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение из юникодных символов ® ☒ § © ☹	Сообщение от Бота: "Извини я тебя не понимаю.."	Рис. 3.5.2.1 Сообщение состоит из юникодных символов Сообщение от Бота: "Извини я тебя не понимаю.."	Low	passed	1 [Задай свои вопросы] сообщение из юникодных символов принимается		Фирсова И.А.	25.05.2019
16	Длина сообщения < 6	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, длина < 6	Сообщение от Бота: "Сообщение слишком короткое, попробуй ещё раз"	Рис. 3.5.2.2. Длина сообщения < 6 Сообщение от Бота: "Сообщение слишком короткое, попробуй ещё раз"	Medium	passed	2 [Задай свои вопросы] при вводе сообщение длиной < 6 сообщение от бота можно улучшить	Enhancement - улучшение. Если бот будет отвечать "Сообщение слишком короткое", то пользователям будет понятна причина некоррект	Фирсова И.А.	25.05.2019

								ности вопр оса"		
17	Длина сообщ ения >50	Зайти в VSU_BOT Нажать Старт Выбрать "Задай свой вопрос" Ввести сообщение, длина >50	Сообщени е от Бота: "Сообщен ие слишком длинное, попробуй ещё раз"	Рис. 3.5.2.3 Длина сообщения >50 Сообщени е от Бота: "Сообщен ие слишком длинное, попробуй ещё раз"	Low	passed	3 [Задай свои вопросы] при вводе сообщение длинной >50 сообщение от бота можно улучшить	см. пред ыдуц ий кейс	Фирс ова И.А.	25.05. 2019

53	Подписка на рассылку, когда уже подписан	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка" Выбрать /on_notifications Ещё раз выбрать /on_notifications	Сообщение от Бота: "Вы уже подписаны на рассылку"	Рис. 3.5.2.4. Подписка на рассылку, когда уже подписан Сообщение от Бота: "Вы уже подписаны на рассылку"	Medium	passed	4 [Рассылка уведомлений] пользователь подписывается на рассылку дважды		Фирсова И.А.	26.05.2019
54	Отписка от рассылки, когда ещё не подписан	Зайти в VSU_BOT Нажать на СТАРТ Выбрать "Рассылка" Выбрать /off_notifications	Сообщение от Бота: "Вы не были подписаны на рассылку, поэтому не можете отписаться"	Рис. 3.5.2.5. Отписка от рассылки, когда ещё не подписан Сообщение от Бота: "Вы не были подписаны на рассылку, поэтому не можете отписаться"	Medium	passed	5 [Рассылка уведомлений] пользователь отписывается от рассылки уведомлений, даже если не был подписан		Фирсова И.А.	26.05.2019

Рис 3.5.2 - Confirmation testing

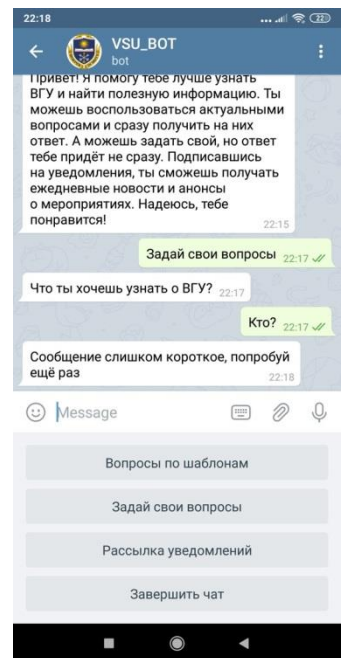
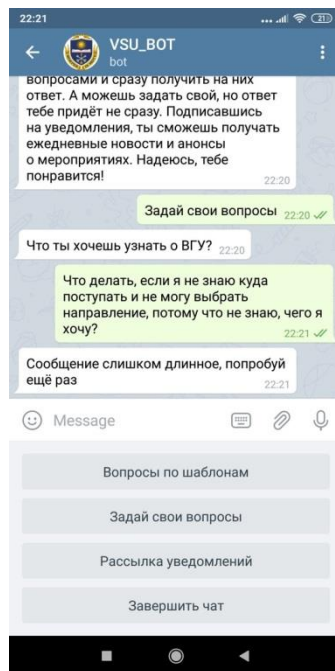
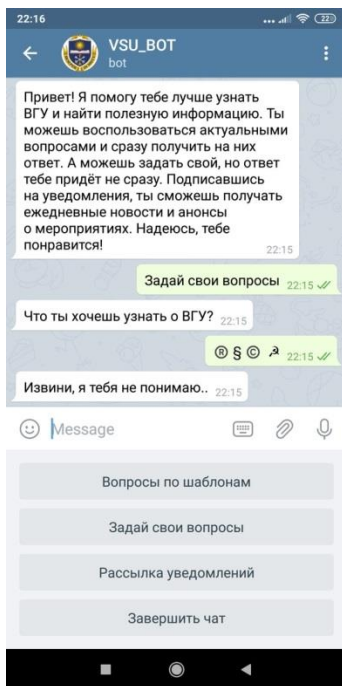
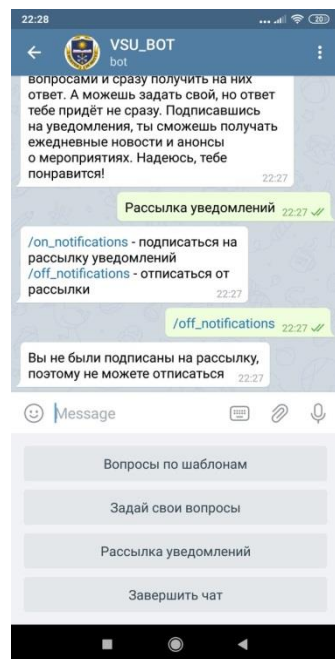
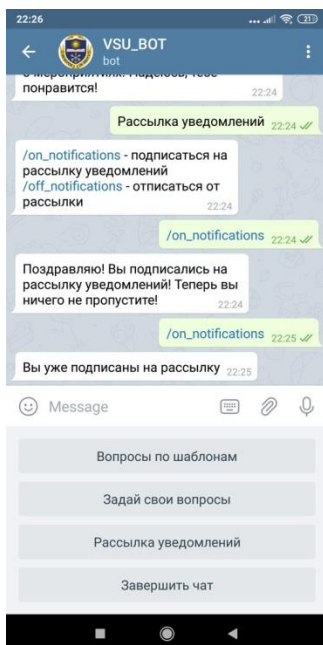


Рис. 3.5.2.1 Сообщение состоит из юникодных СИМВОЛОВ



Чтобы провести закрытие багов, воспользуемся шаблоном Bug Verifying Template – для проверки бага.

Название
Статус

1. Версия:
2. Платформа:

3. Компоненты:
4. Шаги для проверки
5. Фактический результат:

Проводим закрытие багов:

1 [Задай свои вопросы] сообщение из юникодных символов принимается

Статус: Verifying

1. Версия:
ver: 2
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги для проверки
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Задай свой вопрос"
 4. Ввести сообщение из юникодных символов ® ☒ § © ☯ ㄩ
5. Фактический результат:
Сообщение от Бота: "Извини я тебя не понимаю.."

2 [Задай свои вопросы] при вводе сообщение длинной < 6 сообщение от бота можно улучшить

Статус: Verifying

1. Версия:
ver: 2
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги для проверки
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Задай свой вопрос"
 4. Ввести сообщение, длина < 6
5. Фактический результат:
Сообщение от Бота: "Сообщение слишком короткое, попробуй ещё раз"

3 [Задай свои вопросы] при вводе сообщение длинной > 50 сообщение от бота можно улучшить

Статус: Verifying

1. Версия:
ver: 2

2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги для проверки
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Задай свой вопрос"
 4. Ввести сообщение, длина > 50
5. Фактический результат:
Сообщение от Бота: " Сообщение слишком длинное, попробуй ещё раз"

4 [Рассылка уведомлений] пользователь подписывается на рассылку дважды
Статус: Verifying

1. Версия:
ver: 2
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги для проверки
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Рассылка"
 4. Выбрать /on_notifications
 5. Ещё раз выбрать /on_notifications
5. Фактический результат:
Сообщение от Бота: "Вы уже подписаны на рассылку"

5 [Рассылка уведомлений] пользователь отписывается от рассылки уведомлений,
даже если не был подписан

Статус: Verifying

1. Версия:
ver: 2
2. Платформа:
Android 8.1
3. Компоненты:
Frontend
4. Шаги для проверки
 1. Зайти в VSU_BOT
 2. Нажать Старт
 3. Выбрать "Рассылка"
 4. Выбрать /off_notifications
5. Фактический результат:

Сообщение от Бота: "Вы не были подписаны на рассылку, поэтому не можете отписаться"

5.Заключению

В ходе работы проведен анализ предметной области, актуальных технологий и программных решений, так же была изучена документация программного интерфейса Bot API, после чего было спроектировано и разработано программное обеспечение для сервиса обмена сообщениями Telegram с использованием Bot API, которое, впоследствии, было протестировано и отлажено.

Нами были реализованы задачи:

- автоматизация процесса приёма обращений от пользователей;
- разработка интерфейса чат-бота;
- создание механизма информационного взаимодействия между чат-ботом и сайтом ВГУ;
- демонстрация прототипа чат-бота, обеспечивающего:
 - приём обращения от пользователя;
 - интеграцию с существующей информацией;
 - обеспечение пользователю повсеместным доступом к информации – с любого мобильного устройства (смартфона), подключенного к сети Интернет.
- ввод Системы в действие:
 - проведение испытаний Системы;
 - организация работы эксплуатационного персонала;
 - формирование отчетности по результатам испытаний.