

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (ФГБОУ ВО
«ВГУ»)

Техническое задание
Разработка Telegram бота
«GitHubHelper»
(в соответствии с ГОСТ 34)

Заказчик

Ассистент Тарасов В.С.

Исполнители

Кравченко Д.А., Старкин М.В., Мущенко И.В.

Воронеж, 2019

Оглавление

| | | |
|--------|--|----|
| 1. | Общие сведения | 3 |
| 1.1. | Наименование Заказчика | 3 |
| 1.2. | Наименование Исполнителя | 3 |
| 1.3. | Основание для разработки | 3 |
| 1.4. | Плановые сроки начала и окончания работ | 3 |
| 1.5. | Понятия и термины | 3 |
| 2. | Назначение и цели создания приложения | 5 |
| 2.1. | Назначение приложения | 5 |
| 2.2. | Цели создания приложения | 5 |
| 3. | Требования к приложению | 6 |
| 3.1. | Общие требования | 6 |
| 3.2. | Требования к структуре | 6 |
| 3.3. | Требования к безопасности | 8 |
| 3.4. | Функциональные требования | 8 |
| 3.4.1. | Вариант использования: Зарегистрироваться в приложении | 9 |
| 3.4.2. | Вариант использования: Просмотр главной страницы | 9 |
| 3.4.3. | Вариант использования: Просмотр подписок | 9 |
| 3.4.4. | Вариант использования: Поиск репозитория | 10 |
| 3.5. | Требования к пользовательскому интерфейсу | 10 |
| 3.6. | Требования к надёжности | 10 |
| 4. | Состав и содержание работ по созданию системы | 11 |
| 5. | Порядок контроля и приёмки | 12 |
| 6. | Требования к документированию | 13 |

1. Общие сведения

1.1. Наименование Заказчика

Ассистент Тарасов Вячеслав Сергеевич, кафедра программирования и информационных технологий.

1.2. Наименование Исполнителя

Студент Кравченко Денис Александрович, кафедра программирования и информационных технологий.

Студент Мущенко Илья Викторович, кафедра программирования и информационных технологий.

Студент Старкин Михаил Владимирович, кафедра программирования и информационных технологий.

1.3. Основание для разработки

Необходимость заказчика в наличии Telegram bot-a, позволяющего отслеживать новые версии интересующих приложений с помощью системы контроля версий Github.

1.4. Плановые сроки начала и окончания работ

Плановый срок начала работ – Февраль 2019 г.

Плановый срок окончания работ – Июнь 2019 г.

1.5. Понятия и термины

В Техническом Задании используются следующие понятия и термины:

- «Сайт» - часть проекта «**GitHubHelper**», являющийся web-приложением с использованием базы данных SQL.
- «Bot» - часть проекта «**GitHubHelper**», специальная программа, выполняющая автоматически и/или по заданному расписанию какие-либо действия через интерфейсы, предназначенные для людей. В данном

случае является чат-ботом, предоставляющим определенный функционал для взаимодействия с web-сервисом GitHub.

- **«Гость»** - неавторизованный на портале человек, пользующийся ограниченным функционалом сайта и не имеющий доступа к функционалу чат-бота.
- **«Пользователь»** - авторизованный на портале человек, пользующийся функционалом сайта и бота.
- **«Репозиторий»**- репозиторий Git представляет собой каталог файловой системы, в котором находятся файлы конфигурации репозитория, файлы журналов, хранящие операции, выполняемые над репозиторием, индекс, описывающий расположение файлов, и хранилище, содержащее собственно файлы.
- **«Коммит»** - (от англ. *commit* - зд. фиксировать) - сохранение, фиксация (в архиве, репозитории и др.) изменений в программном коде.

2. Назначение и цели создания приложения

2.1. Назначение приложения

Приложение будет состоять из веб-интерфейса и чат-бота. Приложение предназначено для отслеживания изменений версий приложений, размещённых в Github.

2.2. Цели создания приложения

Основными целями создания Telegram bot-a «GitHubHelper» с точки зрения пользователя являются:

- Облегчение процесса отслеживания изменений версий приложений.
- Возможность отслеживать проекты, в разработке которых пользователь не участвует (readonly права).
- Поиск репозитория по названию.

3. Требования к приложению

3.1. Общие требования

Разрабатываемое приложение должно удовлетворять следующим основным требованиям:

- Возможность выполнения основных задач приложения:
 - Получение уведомлений о новых коммитах в отслеживаемых репозиториях;
 - Добавление/Удаление репозитория в список отслеживаемых пользователем;
 - Поиск репозитория по названию;
 - Просмотр истории коммитов;
- Создание интуитивно понятного пользовательского интерфейса.
- Авторизация с помощью Github API;

На рисунке ниже представлена use case диаграмма данного приложения, в котором отображены все функции.

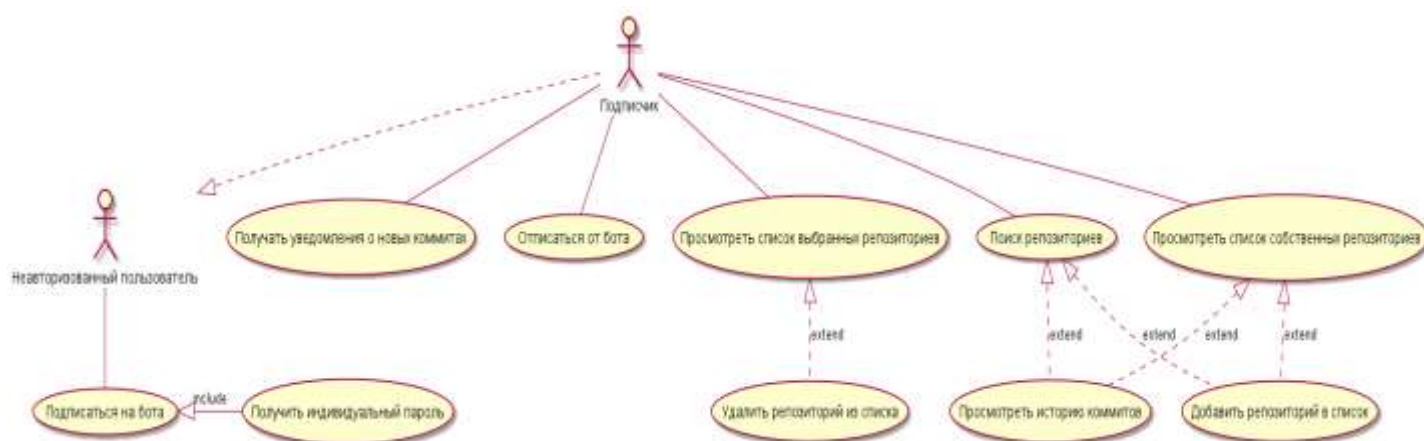


Рисунок 1 - Use case диаграмма приложения

3.2. Требования к структуре

Приложение должно иметь архитектуру, основанную на шаблоне проектирования MVC (Model-View-Controller), а также разделение на

клиентскую и серверную часть, общение которых происходит с помощью REST API. Ниже приведена структурная схема приложения.

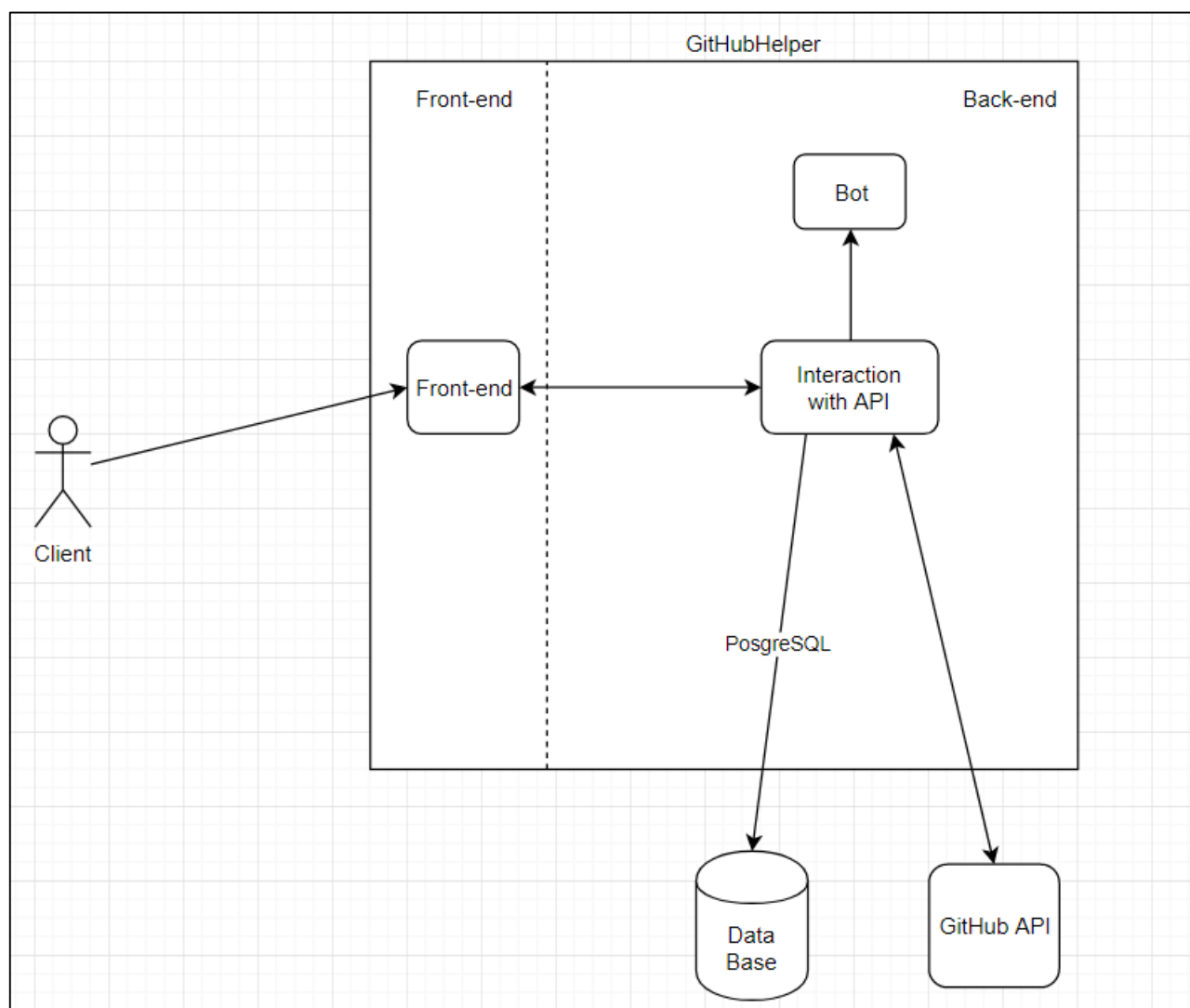


Рисунок 2 - Структурная схема приложения

Шаблон MVC требует реализации трёх отдельных слоёв:

- База данных (Model) – данные, необходимые для корректной работы приложения.
- Графический интерфейс (View), являющегося подписчиком на событие изменения значений свойств, запросов или команд, предоставляемых бизнес-логикой. В случае, если произошли какие-либо изменения свойств или данных базы данных, то происходит оповещение об этом, и пользовательский интерфейс, в свою очередь, запрашивает обновленное значение свойства из блока бизнес-логики. В случае, если

пользователь воздействует на какой-либо элемент интерфейса, вызывается соответствующая команда, предоставленная бизнес-логикой.

- Бизнес-логика (Controller), реализуется на языке программирования Java и является связывающим звеном между графическим интерфейсом и базой данных. Выполняет действия и обрабатывает данные, вводимые пользователем. Также извлекает данные, необходимы для отображения на пользовательском интерфейсе.

3.3. Требования к безопасности

В приложении должна быть реализована авторизация через систему контроля версий Github. Все запросы к внешнему API должны осуществляться по HTTPS протоколу.

3.4. Функциональные требования

Приложение должно удовлетворять следующим основным функциональным требованиям:

- Приложение должно иметь возможность хранить следующую информацию:
 - О пользователе:
 - Идентификатор, используемый Telegram bot-ом для идентификации;
 - Логин пользователя в Github;
 - Имя пользователя в Github;
 - Ссылку на страницу профиля пользователя в Github;
 - О репозитории:
 - Полное название репозитория;
 - Описание репозитория;
 - Ссылку на страницу репозитория на Github;
 - Дату создания репозитория;
 - Ссылку на владельца репозитория;
 - Логин владельца репозитория;

- Приложение должно предоставить возможность добавления/удаления репозитория в список отслеживаемых пользователем;
- В приложении должен быть реализован поиск репозитория по названию;
- Приложение должно поддерживать авторизацию через Github;
- Приложение должно выдавать авторизованному пользователю уникальный идентификатор, связывающий его аккаунт в Github и в Telegram;

3.4.1. Вариант использования: Зарегистрироваться в приложении

Гость может зарегистрироваться на сайте. Для этого ему необходимо подписаться на Telegram bot-a, ссылка на которого находится на главной странице. После подписки бот пришлёт уникальный код, который пользователю необходимо ввести в специальное поле. После этого становится доступна кнопка «Войти», перенаправляющая на страницу авторизации через Github. После успешной авторизации пользователь попадает на главную страницу, как авторизованный пользователь и его данные добавляются в базу данных.

3.4.2. Вариант использования: Просмотр главной страницы

Гость может открыть главную страницу сайта. Открытие главной страницы также возможно с любой страницы сайта для авторизованного пользователя. У авторизованного пользователя на этой странице отображена информация из Github:

- Логин;
- Имя пользователя;
- Список репозитория, владельцем которых является пользователь;

3.4.3. Вариант использования: Просмотр подписок

Авторизованный пользователь может просмотреть список репозиториев, уведомления об изменении которых он получает. На странице «Мои подписки» пользователь может увидеть свои логин и полное имя из Github. Также ниже должен располагаться список репозиториев, которые пользователь выбрал для отслеживания.

3.4.4. Вариант использования: Поиск репозиториев

Авторизованный пользователь может воспользоваться поиском. Для этого ему необходимо ввести название репозитория в специальное поле, доступное из всех страниц сайта. После ввода данных пользователя перенаправит на страницу с результатами поиска.

3.5. Требования к пользовательскому интерфейсу

Пользовательский интерфейс должен удовлетворять следующим основным требованиям:

- Дизайн сайта должен быть достаточно простым (не перегруженным функционалом и внешним видом);
- Отсутствие возможности перейти на другие страницы приложения до авторизации через Github;
- Расположение модулей на страницах должно быть читабельным и понятным, в том числе, низкоуровневому пользователю;
- Все страницы сайта должны быть выдержаны в едином стиле;

3.6. Требования к надёжности

Система должна быть защищена от сбоев и защищена от неверных действия пользователя.

4. Состав и содержание работ по созданию системы

Состав и содержание работ по созданию системы включают следующие этапы:

1. Предпроектное исследование

Был изучен принцип создания ботов в выбранном мессенджере Telegram, а также возможности Github API, выбранного для реализации проекта.

2. Анализ предметной области

Был разработан подробный анализ системы и введение организационных требований к решению задач и целей.

3. Разработка ТЗ

Результат - документация на разрабатываемую систему, в которой описаны все необходимые организационные требования к разработке.

4. Разработка модели программы

Результат – описание спецификационных данных, определены связи между сущностями, построение концептуальной модели БД, построение логической модели БД.

5. Разработка программы

6. Тестирование программы и её корректировка

7. Сдача системы в эксплуатацию

5. Порядок контроля и приёмки

Заказчик принимает разработанный продукт при выполнении следующих пунктов:

- Успешное прохождение большей части тестов;
- Приложение соответствует заявленным требованиям

6. Требования к документированию

Отчётные документы должны включать в себя текстовые материалы (представленные в виде бумажной копии и на цифровом носителе в формате MS Word или PDF) и, при необходимости, графические.

Необходимо предоставить отчёт по курсовому проекту согласно учебному плану.