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](#_Toc19652574)

[1.1. Наименование Заказчика 3](#_Toc19652575)

[1.2. Наименование Исполнителя 3](#_Toc19652576)

[1.3. Основание для разработки 3](#_Toc19652577)

[1.4. Плановые сроки начала и окончания работ 3](#_Toc19652578)

[1.5. Понятия и термины 3](#_Toc19652579)

[2. Назначение и цели создания приложения 5](#_Toc19652580)

[2.1. Назначение приложения 5](#_Toc19652581)

[2.2. Цели создания приложения 5](#_Toc19652582)

[3. Требования к приложению 6](#_Toc19652583)

[3.1. Общие требования 6](#_Toc19652584)

[3.2. Требования к структуре 6](#_Toc19652585)

[3.3. Требования к безопасности 8](#_Toc19652586)

[3.4. Функциональные требования 8](#_Toc19652587)

[3.4.1. Вариант использования: Зарегистрироваться в приложении 9](#_Toc19652588)

[3.4.2. Вариант использования: Просмотр главной страницы 9](#_Toc19652589)

[3.4.3. Вариант использования: Просмотр подписок 9](#_Toc19652590)

[3.4.4. Вариант использования: Поиск репозиториев 10](#_Toc19652591)

[3.5. Требования к пользовательскому интерфейсу 10](#_Toc19652592)

[3.6. Требования к надёжности 10](#_Toc19652593)

[4. Состав и содержание работ по созданию системы 11](#_Toc19652594)

[5. Порядок контроля и приёмки 12](#_Toc19652595)

[6. Требования к документированию 13](#_Toc19652596)

1. Общие сведения
   1. Наименование Заказчика

Ассистент Тарасов Вячеслав Сергеевич, кафедра программирования и информационный технологий.

* 1. 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. Основание для разработки

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

* 1. Плановые сроки начала и окончания работ

Плановый срок начала работ – Февраль 2019 г.

Плановый срок окончания работ – Июнь 2019 г.

* 1. Понятия и термины

В Техническом Задании используются следующие понятия и термины:

* **«Сайт»** - часть проекта **«GitHubHelper»**, являющийся web-приложением с использование 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.

1. Назначение и цели создания приложения
   1. Назначение приложения

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

* 1. Цели создания приложения

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

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

1. Требования к приложению
   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 репозиториев в список отслеживаемых пользователем;
  + Поиск репозиториев по названию;
  + Просмотри 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

* 1. Требования к структуре

Приложение должно иметь архитектуру, основанную на шаблоне проектирования MVC (Model-View-Controler), а также разделение на клиентскую и серверную часть, общение которых происходит с помощью REST API. Ниже приведена структурная схема приложения.

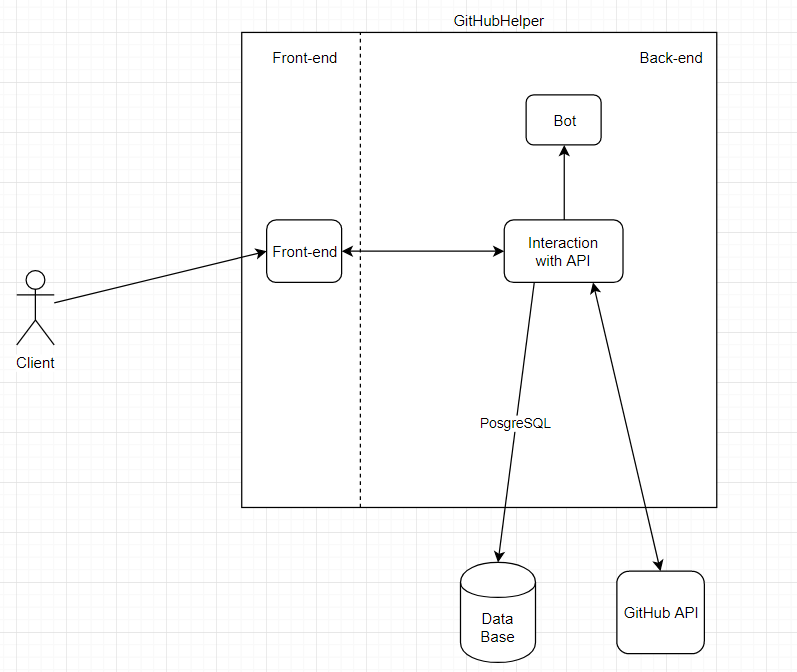


Рисунок 2 - Структурная схема приложения

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.
  1. Требования к безопасности

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

* 1. Функциональные требования

Приложение должно удовлетворять следующим основным функциональным требованиям:

* Приложение должно иметь возможность хранить следующую инфомацию:
  + 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 репозиториев в список отслеживаемых пользователем;
* В приложении должен быть реализован поиск репозиториев по названию;
* Приложение должно поддерживать авторизацию через Github;
* Приложение должно выдавать авторизованному пользователю уникальный идентификатор, связывающий его аккаунт в Github и в Telegram;
  + 1. Вариант использования: Зарегистрироваться в приложении

Гость может зарегистрироваться на сайте. Для этого ему необходимо подписаться на Telegram bot-а, 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.

* + 1. 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 репозиториев, владельцем которых является пользователь;
  + 1. 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.

* + 1. 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 результами поиска.

* 1. 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;
  1. Требования к надёжности

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

1. Состав и содержание работ по созданию системы

Состав и содержание работ по созданию систему включают следующие этапы:

1. Предпроектное исследование

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

1. Анализ предметной области

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

1. Разработка ТЗ

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

1. Разработка модели программы

Результат – описание спецификационных данных, определене связей между сущностями, построение концептуальной модели БД, построение логической модели БД.

1. Разработка программы
2. Тестирование программы и её корректировка
3. Сдача системы в эксплуатацию
4. Порядок контроля и приёмки

Заказчик принимает разработанный продукт при выполнении следующих пунктов:

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

1. 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.