**Техническое задание**

**1. Общие сведения**

* **Название проекта:** Telegram-бот «BlueWhale» – психологическая помощь и поддержка»
* **Исполнитель:** Тархова Алина, студент 1 курса, направление «Информационные системы и программирование»
* **Заказчик:** Преподаватель дисциплины «Оператор ЭВМ»
* **Сроки выполнения:** 6 часов с момента выдачи задания
* **Юзернейм бота:** [t.me/BlueWhaleHelpAndSupport\_bot](https://t.me/BlueWhaleHelpAndSupport_bot)
* **Мой GitHub:** <https://github.com/Rionec>
* **Репозиторий бота на Github:** <https://github.com/Rionec/BlueWhaleBot>

**2. Цель и задачи проекта**

* **Цель:** Разработать Telegram-бота, отправляющего простые советы нацеленные на улучшение морального состояния пользователя, а так же предоставляющего быстрый доступ к сервисам и контактам экстренной психологической помощи
* **Задачи:**
  + Изучить библиотеку pyTelegramBotAPI.
  + Создать Telegram-бота с использованием данной библиотеки.
  + Добавить команды /start и /help для взаимодействия с пользователем.
  + Добавить ключевые инлайн-кнопки, предоставляющие доступ к остальному функционалу бота: «Случайный совет», «Контакты и сервисы».
  + Инегрировать в работу бота базу данных, созданную в Excel

**Блок схема работы бота**

/start

Предложение пользователю использовать комманды /help

Предоставление пользователю выбора следующего действия

Отправка случайного совета пользователю.   
Предложение получить еще один совет.

Отправка пользователю списка интернет ресурсов психологической помощи и телефонных номеров доверия

«Контакты и сервисы»

«Случайный совет»

«Случайный совет»

**3. Функциональные требования**

* При запуске команды /start бот отправляет приветственное сообщение и предлагает доступные функции
* Команда /help предоставляет краткую информацию о боте, а также выводит три инлайн-кнопки**:** «Список проблем», «Случайный совет», «Контакты и сервисы».
* При выборе «Случайный совет» пользователю отправляется случайный совет
* При выборе «Контакты и сервисы» выводится список ресурсов(сайтов) психологической помощи и контакты телефонов доверия

**4. Нефункциональные требования**

* Бот должен быть реализован на языке программирования Python с использованием библиотеки pyTelegramBotAPI.
* Код должен быть структурирован и снабжен комментариями для облегчения понимания.
* Бот должен корректно обрабатывать некорректный ввод пользователя.
* Интерфейс бота должен быть на русском языке.
* Тон общения бота с пользователем должен быть мягкий и доброжелательный

**5. Требования к интерфейсу**

* Использование инлайн кнопок Telegram для предоставления вариантов выбора.
* Простое и интуитивно понятное взаимодействие с пользователем.
* Сообщения бота должны быть развернутыми и информативными.
* Тон общения бота с пользователем должен быть мягкий и доброжелательный

**6. Технические требования**

* Совместимость с последней стабильной версией Python.
* Использование только стандартных библиотек Python и pyTelegramBotAPI.
* Хранение токена бота в отдельном конфигурационном файле или переменной окружения для обеспечения безопасности.
* Хранение данных и запоминание выборов пользователя.
* Парсинг данных.
* Асинхронность.

**7. Этапы разработки**

* Изучение документации pyTelegramBotAPI.
* Создание бота в Telegram через @BotFather и получение токена.
* Настройка окружения и установка необходимых библиотек.
* Разработка и тестирование основных функций бота.
* Добавление обработчиков команд /start и /help.
* Добавление основной части в виде инлайн кнопок «Случайный совет» и «Контакты и сервисы»
* Создание репозитория на сайте GitHub
* Тестирование бота на корректность работы.
* Подготовка и сдача презентации проекта.

**8. Критерии приемки**

* Бот успешно запускается и отвечает на команды /start и /help.
* Интерфейс бота соответствует требованиям.
* Все данные корректно выводятся
* Все кнопки работают корректно
* Код оформлен согласно требованиям и содержит необходимые комментарии.