**Техническое задание**

1. **Общие сведения**

<https://github.com/toskovattt> ; @password9en\_bot

* **Название проекта:** Telegram-бот «PasswordGen»
* **Исполнитель:** группа студентов 1 курса (Трофимов, Иванов, Пирютко), направление «Информационные системы и программирование»
* **Заказчик:** Преподаватель дисциплины «Оператор ЭВМ» (Попенко Захар Васильевич)
* **Сроки выполнения:** 48 часа с момента выдачи задания

**2. Цель и задачи проекта**

* **Цель:** Разработать Telegram-бота, позволяющего пользователю сгенерировать случайный пароль.
* **Задачи:**
  + Изучить библиотеку pyTelegramBotAPI, String, Random.
  + Создать Telegram-бота с использованием данных библиотек.
  + Реализовать выдачу случайного пароля с указанным числом символом.
  + Обеспечить обработку пользовательского ввода и генерацию ответа от бота.
  + Добавить команды /start, /info для взаимодействия с пользователем.

**3. Функциональные требования**

* При запуске команды /start бот отправляет приветственное сообщение и предлагает сгенерировать пароль.
* Бот предлагает сложность пароля (Простой, Средний, Сложный)
* Бот генерирует пароль на основе выбора пользователя
* После генерации пароля бот предлагает сгенерировать новый пароль (Да, Нет)
* При нажатии на кнопку “Да” бот предлагает заново выбрать сложность пароля
* При нажатии на кнопку “Нет” бот прощается с пользователем и завершает работу

**4. Нефункциональные требования**

* Бот должен быть реализован на языке программирования Python с использованием библиотеки pyTelegramBotAPI, Random, String.
* Код должен быть структурирован и снабжен комментариями для облегчения понимания.
* Бот должен корректно обрабатывать некорректный ввод пользователя.
* Интерфейс бота должен быть на русском языке.

**5. Требования к интерфейсу**

* Использование встроенной клавиатуры Telegram для предоставления вариантов выбора.
* Простое и интуитивно понятное взаимодействие с пользователем.
* Сообщения бота должны быть краткими и информативными.

**6. Технические требования**

* Совместимость с последней стабильной версией Python.
* Использование только стандартных библиотек Python, pyTelegramBotAPI, Random, String.
* Хранение токена бота в отдельном конфигурационном файле или переменной окружения для обеспечения безопасности.

**7. Этапы разработки**

* Изучение документации pyTelegramBotAPI, Random, String.
* 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 /info.
* Тестирование бота на корректность работы.
* Подготовка и сдача проекта.

**8. Критерии приемки**

* Бот успешно запускается и отвечает на команды /start и /info.
* Пользователь может выбрать количество символом, и бот успешно предоставит пароль.
* Интерфейс бота соответствует требованиям.
* Код оформлен согласно требованиям и содержит необходимые комментарии.

