**Техническое задание для Telegram бота “Финик"**

**1. Общие сведения**

* **Язык программирования**: Python
* **Библиотеки**: telebot, logging, threading, sqlite3 (через модуль database.py)
* **Функционал**: Управление балансом пользователей, магазин товаров, рейтинг, переводы, администрирование.

**2. Функциональные требования**

**2.1. Основные функции**

* **Регистрация пользователей**:
  + При первом запуске (/start) пользователь регистрируется в системе.
  + Пользователь должен принять пользовательское соглашение.
* **Баланс**:
  + Два типа баланса: активный (для покупок) и пассивный (для переводов).
  + Отображение баланса по команде "💰 Баланс".
* **Магазин**:
  + Пользователи могут просматривать товары (кнопка "🛒 Магазин").
  + Покупка товаров с списанием средств с активного баланса.
* **Переводы**:
  + Ограничение: не более 3 переводов в день.
  + Подтверждение перевода через интерактивные кнопки.
* **Рейтинг**:
  + Топ пользователей по активному балансу (кнопка "🏆 Рейтинг").
  + Отображение позиции пользователя в рейтинге.
* **История операций**:
  + Просмотр истории покупок и переводов (кнопка "📜 История").

**2.2. Административные функции**

* **Штрафы**:
  + Администратор может списывать средства с пользователей (кнопка "😡 Выдать штраф").
* **Настройки бонусов**:
  + Изменение периода начисления бонусов (кнопка "⏱️ Сменить время бонуса").
* **Управление пользователями**:
  + Просмотр списка пользователей (кнопка "🗿 Пользователи").
  + Назначение администраторов (кнопка "📥 Новый администратор").

**2.3. Дополнительные функции**

* **Периодический бонус**:
  + Начисление бонуса на пассивный баланс каждые N минут (настраивается администратором).
* **Поддержка**:
  + Отображение инструкции (кнопка "🆘 Помощь").

**3. Технические требования**

**3.1. База данных**

* **Модуль**: database.py (импортируется в код).
* **Таблицы**:
  + Пользователи (ID, username, активный баланс, пассивный баланс, роль).
  + Покупки (ID пользователя, название товара, стоимость, дата).
  + Переводы (ID отправителя, ID получателя, сумма, дата).
  + Ожидающие переводы (временные данные для подтверждения).

**3.2. Логирование**

* Используется модуль logging для отслеживания ошибок и действий бота.

**3.3. Потоки**

* Отдельный поток (Thread) для периодического начисления бонусов.

**3.4. Константы**

* **rating\_size**: Размер рейтинга (по умолчанию 5).
* **bonus\_period**: Период начисления бонуса в минутах (по умолчанию 10).
* **bonus\_amount**: Размер бонуса (по умолчанию 1000).

**4. Интерфейс**

**4.1. Клавиатуры**

* **Основное меню**: Зависит от роли пользователя (пользователь/администратор).
* **Инлайн-кнопки**: Для подтверждения действий (покупки, переводы).
* **Reply-кнопки**: Для навигации по меню.

**4.2. Сообщения**

* Форматирование: HTML (parse\_mode='html').
* Мультимедиа: Отправка изображений товаров.

**5. Безопасность**

* Ограничение на переводы: не более 3 в день.
* Проверка ролей для административных функций.
* Подтверждение действий через callback-кнопки.

**6. Условия эксплуатации**

* **Сервер**: Локальный или облачный хостинг с поддержкой Python.
* **Зависимости**: Указаны в коде (telebot, sqlite3 и т.д.).
* **Запуск**: Скрипт bot.py с предварительно инициализированной базой данных.

**7. Дополнительные файлы**

* **greeting.txt**: Приветственное сообщение при старте.
* **user\_agreement.docx**: Пользовательское соглашение.
* **instruction\_for\_buttem\_help.txt**: Инструкция по использованию.

**8. Тестирование**

* Проверка всех кнопок и команд.
* Тестирование корректности начисления бонусов.
* Проверка ограничений (например, переводов).

**9. Сроки и доработки**

* Возможные доработки:
  + Добавление новых товаров.
  + Расширение админ-панели.
  + Интеграция с веб-приложением для заданий (указан пример URL).