# Доклад по проекту: Telegram-бот для бронирования столиков

## 1. Цели и задачи

Целью проекта было создание удобного Telegram-бота для бронирования столиков в ресторане с возможностью визуального контроля занятости через отдельное десктопное окно.

Задачи включали:

- Реализацию команд для бронирования, отмены и просмотра своих броней;

- Добавление роли администратора с доступом к полному списку бронирований;

- Визуализацию занятых столиков по времени и дате в графическом интерфейсе tkinter.

## 2. Структура проекта

Проект состоит из следующих модулей:

- main.py — инициализация Telegram-бота;

- handlers/ — папка с обработчиками команд;

- db.py — инициализация SQLite-базы данных;

- time\_check.py — проверка времени по расписанию;

- view\_db.py — десктопное GUI-приложение;

- config.py — настройки ID и часов работы.

Используется библиотека tkcalendar для выбора даты.

## 3. Интерфейс

Telegram-бот поддерживает команды:

/book, /cancel, /mybookings, /view, /free, /help.

Графический интерфейс view\_db.py включает:

- Календарь выбора даты;

- Таблицу с интервалами времени и столиками;

- Подсветку занятых слотов.

## 4. Этапы разработки

1. Проектирование базы и команд;

2. Реализация Telegram-бота;

3. Обработка времени и проверок;

4. Разработка tkinter-интерфейса;

5. Интеграция с SQLite и тестирование.

## 5. Структура БД

Таблица bookings:

- id: INTEGER — ID брони

- user\_id: INTEGER — Telegram ID

- username: TEXT — имя пользователя

- table\_number: INTEGER — номер столика

- booking\_time: TEXT — начало брони

- booking\_end\_time: TEXT — конец брони

Хранение в файле restaurant.db.

## 6. Реализация view\_db.py

Модуль запускает окно с календарём и таблицей:

- Подключение к базе данных;

- Отображение интервалов времени с шагом 10 минут;

- Подсветка занятых ячеек красным цветом;

- Обновление интерфейса по дате.

Ключевые функции: get\_bookings\_for\_date, round\_time\_down, update\_table.

## 7. Демонстрация

1. Пользователь бронирует столик через бота;

2. Проверка ограничений по времени и дате;

3. Администратор запускает view\_db.py и выбирает дату;

4. Интерфейс показывает таблицу занятости.

## 8. Вывод

Система реализует удобное бронирование через Telegram без сайта.

Позволяет администратору видеть полную картину занятости столиков.

Простая архитектура и локальная база делают проект гибким и расширяемым.