Telegram-бот для подготовки к ЕГЭ по математике

# 1. Введение

Цель проекта:

Создание интерактивного помощника для учащихся, который предоставляет:  
📚 Теоретические материалы по математике  
❓ Практические задания с автоматической проверкой  
📊 Отслеживание прогресса  
📂 Дополнительные ресурсы (файлы, каналы)  
🌤 Интеграцию с сервисом погоды

Целевая аудитория:

Учащиеся 10–11 классов  
Абитуриенты  
Преподаватели математики

Технологии:

Язык: Python 3.x  
Библиотеки:  
- pyTelegramBotAPI  
- SQLAlchemy  
- requests  
База данных: SQLite  
Хостинг: Локальный сервер или облако (PythonAnywhere, Heroku)

# 2. Функционал бота

## 2.1 Основное меню

При запуске (/start) пользователь видит меню с кнопками:  
Профиль – статистика ответов, дата регистрации  
Теория – 15 разделов с теорией (НОД/НОК, графы, теория игр и др.)  
Практика – случайные вопросы с проверкой ответов  
Полезные файлы – PDF-материалы для скачивания  
🏆 Топ-10 – рейтинг пользователей по правильным ответам  
Полезные каналы – ссылки на образовательные Telegram-каналы  
Погода – интеграция с WeatherAPI

## 2.2 Дополнительные функции

Система предложений  
История вопросов  
Объяснения ошибок

# 3. Техническая реализация

## 3.1 База данных

Используется SQLite с таблицами:  
User – данные пользователей  
PracticeQuestion – вопросы  
Suggestion – предложения  
ButtonText – тексты для кнопок  
Пример модели:  
class User(Base):  
 telegram\_id = Column(Integer, primary\_key=True)  
 username = Column(String)  
 registration\_date = Column(DateTime, default=datetime.now)  
 correct\_answers = Column(Integer, default=0)  
 incorrect\_answers = Column(Integer, default=0)

## 3.2 Ключевые модули

main.py – основной код бота  
database.py – настройка подключения к БД  
models.py – модели SQLAlchemy

## 3.3 Интеграции

WeatherAPI – для запроса погоды:  
def get\_weather(city):  
 api\_key = "f430450853714c279f8221723251705"  
 url = f"http://api.weatherapi.com/v1/current.json?key={{api\_key}}&q={{city}}&lang=ru"

# 4. Особенности реализации

## 4.1 Система состояний

Используется словарь user\_suggestions:  
user\_suggestions = {}  
if user\_suggestions.get(chat\_id) == "waiting\_for\_suggestion":  
 # Обработка предложения

## 4.2 Практика

Вопросы выбираются случайно без повторов:  
user\_data = user\_practice\_data.get(chat\_id, {"asked": []})  
remaining = [q for q in all\_questions if q.question not in user\_data["asked"]]

## 4.3 Гибкий интерфейс

Динамические кнопки (InlineKeyboardMarkup)  
@bot.callback\_query\_handler(func=lambda call: True)  
def universal\_handler(call):  
 if call.data == 'profile':  
 show\_profile(call)  
 elif call.data == 'theory':  
 Main\_Menu(call)

# 5. Инструкция по развертыванию

Установка зависимостей:  
pip install pyTelegramBotAPI sqlalchemy requests  
  
Настройка:  
Замените токен бота:  
bot = telebot.TeleBot('ВАШ\_ТОКЕН')  
Добавьте API-ключ для WeatherAPI.  
  
Запуск:  
python main.py

# 6. Перспективы развития

Добавление генератора вариантов ЕГЭ  
Модуль для преподавателей  
Уведомления о новых материалах  
Расширение базы вопросов

# 7. Заключение

Бот предоставляет удобный инструмент для подготовки к ЕГЭ, сочетающий теорию, практику и отслеживание прогресса. Гибкая архитектура позволяет легко расширять функционал.