# 🎁 Telegram Gift Monitor Pro

Real-time monitoring system for Telegram gift notifications with instant mobile alerts.

## 🚀 Quick Start

### Prerequisites

- Python 3.10+
- Docker & Docker Compose
- Telegram API credentials
- Firebase account (for push notifications)

### 1. Clone and Setup

```bash
# Clone the repository
git clone https://github.com/yourusername/telegram-gift-monitor.git
cd telegram-gift-monitor

# Copy environment example
cp .env.example .env

# Edit .env with your credentials
# Required: TELEGRAM_API_ID, TELEGRAM_API_HASH, TELEGRAM_PHONE
```

### 2. Get Telegram API Credentials

1. Go to https://my.telegram.org
2. Log in with your phone number
3. Click "API Development Tools"
4. Create new application
5. Copy `api_id` and `api_hash` to `.env`

### 3. Test Telegram Connection

```bash
# Install dependencies
pip install pyrogram python-dotenv

# Run test script
python test_monitor.py
```

### 4. Start Full System

bash

```bash
# Start database and Redis
docker-compose up -d postgres redis

# Wait 10 seconds for services to start
sleep 10

# Run database migrations
docker-compose exec postgres psql -U tgm_user -d tgm_db -f /docker-entrypoint-initdb.d/001_initial_schema.sql

# Start API server
cd backend/services/api
python main.py

# In another terminal, start monitor
cd backend/services/monitor
python telegram_monitor.py
```

## 📱 API Endpoints

Access API documentation at: http://localhost:8000/docs

### Key endpoints:

- `POST /api/v1/auth/register` - Register new user
- `POST /api/v1/license/activate` - Activate license
- `GET /api/v1/channels` - List available channels
- `POST /api/v1/channels/subscribe` - Subscribe to channel

## 🔑 Generate Test License

python

```python
from backend.services.api.licenses import LicenseService

service = LicenseService()
license = service.generate_license_key("trial")
print(f"License: {license['key']}")
```

## 📁 Project Structure

```
telegram-gift-monitor/
├── backend/
│   └── services/
│       ├── api/          # FastAPI server
│       ├── monitor/      # Telegram monitor
│       └── shared/       # Shared modules
├── database/           # SQL schemas
├── mobile/             # Flutter app (next phase)
└── docker-compose.yml  # Docker configuration
```

## 🛠️ Development

### Run with Docker

```bash
docker-compose up -d
```

### Run locally

```bash
# Create virtual environment
python -m venv venv
source venv/bin/activate  # On Windows: .\venv\Scripts\activate

# Install dependencies
pip install -r backend/services/monitor/requirements.txt

# Start services
python backend/services/api/main.py
python backend/services/monitor/telegram_monitor.py
```

## 🔍 Monitoring Channels

Add channels to monitor in `.env`:

```
MONITOR_CHANNELS=@channel1,@channel2,@channel3
```

Gift keywords (customize in `.env`):

```
GIFT_KEYWORDS=new limited gift,редкий,подарок,rare
```

## 📊 Features

- ✅ Real-time Telegram monitoring
- ✅ Instant push notifications
- ✅ License management system
- ✅ Multi-channel support
- ✅ RESTful API
- ✅ PostgreSQL database
- ✅ Redis caching
- ✅ JWT authentication
- ✅ Rate limiting
- ✅ Docker support

## 🚧 Coming Soon

☐ Flutter mobile app
☐ iOS support
☐ Web dashboard
☐ Analytics
☐ Auto-buy integration

## 📝 License

This project is proprietary software. All rights reserved.

## 🤝 Support

For issues or questions, create an issue in the repository.