```dockerfile
# Dockerfile
FROM python:3.11-slim

# Set working directory
WORKDIR /app

# Install system dependencies including FFmpeg
RUN apt-get update && \
    apt-get install -y --no-install-recommends \
    ffmpeg \
    wget \
    ca-certificates \
    && rm -rf /var/lib/apt/lists/*

# Copy requirements first for better caching
COPY requirements.txt .

# Install Python dependencies
RUN pip install --no-cache-dir -r requirements.txt

# Copy application code
COPY telegram_media_bot.py .

# Create downloads directory
RUN mkdir -p /downloads

# Set environment variables
ENV PYTHONUNBUFFERED=1
ENV DOWNLOAD_PATH=/downloads

# Run the bot
CMD ["python", "-u", "telegram_media_bot.py"]

# ===========================================
# docker-compose.yml
# ===========================================
# Save this as a separate file: docker-compose.yml
#
# version: '3.8'
#
# services:
#   telegram-media-bot:
#     build: .
#     container_name: telegram-media-bot
#     environment:
#       - BOT_TOKEN=${BOT_TOKEN}
#       - DOWNLOAD_PATH=/downloads
#       - MAX_FILE_SIZE_MB=${MAX_FILE_SIZE_MB:-50}
#       - ALLOWED_USERS=${ALLOWED_USERS:-}
#     volumes:
#       - ./downloads:/downloads
#       - ./logs:/app/logs
#     restart: unless-stopped
#     logging:
#       driver: "json-file"
#       options:
#         max-size: "10m"
#         max-file: "3"
#
# ===========================================
```

```
# Usage:
# ==========================================
# 1. Create .env file with your BOT_TOKEN
# 2. Run: docker-compose up -d
# 3. View logs: docker-compose logs -f
# 4. Stop: docker-compose down
```

```
# Usage:
# ==========================================
# 1. Create .env file with your BOT_TOKEN
# 2. Run: docker-compose up -d
# 3. View logs: docker-compose logs -f
# 4. Stop: docker-compose down
```