TGCF (Telegram Chat Forwarder) - Usage Instructions

🎉 Application is Running!

The TGCF web interface is successfully running on your system.

📍 Access Information

- **URL:** http://localhost:8501
- **Password:** `tgcf_demo_password_2025`

> **Note:** This localhost refers to the computer that's running the application. If you want to access it from your local machine or deploy it remotely, you'll need to deploy the application on your own system or use port forwarding.

🔐 Getting Telegram API Credentials

Before you can use TGCF, you need to obtain Telegram API credentials. Here's how:

Step 1: Get API ID and API Hash

1. Visit https://my.telegram.org
2. Log in with your Telegram phone number
3. Click on "API Development Tools"
4. Fill in the application details:
   - **App title:** TGCF (or any name you prefer)
   - **Short name:** tgcf
   - **Platform:** Desktop (or appropriate)
   - **Description:** Telegram message forwarding automation
5. Click "Create application"
6. You'll receive:
   - **api_id** (a number, e.g., 123456)
   - **api_hash** (a string, e.g., "abc123def456...")

IMPORTANT: Keep these credentials secure and never share them publicly!

Step 2: Choose Account Type

TGCF can work with two types of Telegram accounts:

Option A: Bot Account (Recommended for most users)

- **Pros:** Easier to set up, no need for session string
- **Cons:** Limited to bot capabilities
- **Steps:**
  1. Open Telegram and search for @BotFather (https://t.me/botfather)
  2. Send `/newbot` command

3. Follow instructions to create your bot
4. Copy the **bot token** provided (e.g., `123456789:ABCdefGHIjklMNOpqrsTUVwxyz`)

Option B: User Account (For advanced features)

- **Pros:** Full user account capabilities, can access user-only chats
- **Cons:** Requires session string generation
- **Steps:**
  1. You'll need to generate a session string
  2. You can use the tool: https://replit.com/@aahnik/tg-login?v=1
  3. Or use the command-line tool: https://github.com/aahnik/tg-login
  4. Enter your API ID, API Hash, and phone number to generate the session string

---

🚀 Configuring TGCF

Step 1: Access the Web Interface

1. Open your browser and go to: http://localhost:8501
2. Enter the password: `tgcf_demo_password_2025`

Step 2: Enter Telegram Credentials

1. Navigate to "🔑 Telegram Login" page (from the sidebar)
2. Enter your credentials:
   - **API ID:** (from my.telegram.org)
   - **API HASH:** (from my.telegram.org)
   - Choose account type (Bot or User)
   - If Bot: Enter **Bot Token**
   - If User: Enter **Session String**
3. Save the configuration

Step 3: Configure Forwarding Rules

1. Go to "🔗 Connections" page
2. Add source and destination chats
3. Configure which messages to forward

Step 4: Set Up Plugins (Optional)

Navigate to "🔌 Plugins" to enable:
- **Filter:** Blacklist/whitelist specific messages
- **Format:** Apply text formatting (bold, italics, etc.)
- **Replace:** Replace text using regex
- **Caption:** Add headers/footers to messages
- **Watermark:** Add watermarks to images/videos
- **OCR:** Extract text from images

Step 5: Run TGCF

1. Go to "🏃 Run" page
2. Choose mode:
   - **Live Mode:** Forward messages from now onwards (recommended)
   - **Past Mode:** Forward existing/old messages

3. Click "Start" to begin forwarding

📚 Features Overview

Core Features:

- ✅ Forward messages between Telegram chats
- ✅ Support for both bot and user accounts
- ✅ Live and past message forwarding modes
- ✅ Advanced filtering and formatting options
- ✅ Plugin system for extensibility

Available Plugins:

- 🔍 **Filter:** Control which messages get forwarded
- 📝 **Format:** Apply text styling
- 🔄 **Replace:** Text replacement with regex support
- 📋 **Caption:** Add custom headers/footers
- 🖼 **Watermark:** Add watermarks to media (requires ffmpeg)
- 🔍 **OCR:** Extract text from images (requires tesseract-ocr)

🛠️ Technical Details

Project Location

```
/home/ubuntu/github_repos/tgcf
```

Configuration Files

- `.env` - Contains the web interface password
- `tgcf.config.json` - Auto-generated configuration file

Running the Application

The application is currently running in the background. To manage it:

```
# Check if running
ps aux | grep streamlit

# Stop the application
pkill -f "streamlit run"

# Restart the application
cd /home/ubuntu/github_repos/tgcf
tgcf-web &
```

CLI Mode (Alternative to Web Interface)

You can also run TGCF from the command line:

```
cd /home/ubuntu/github_repos/tgcf
tgcf --help
tgcf live  # Start in live mode
tgcf past  # Start in past mode
```

📖 Additional Resources

- **Official Documentation:** https://github.com/aahnik/tgcf/wiki
- **Video Tutorials:** https://www.youtube.com/playlist?list=PLSTrsq_DvEgisMG5BLUf97tp2DoAnwCMG
- **Feature Overview Video:** https://youtu.be/FclVGY-K70M
- **GitHub Repository:** https://github.com/aahnik/tgcf

⚠️ Security Best Practices

1. **Keep credentials secure:** Never share your API ID, API Hash, Bot Token, or Session String
2. **Change the password:** Update the password in `.env` file to something more secure
3. **Use strong passwords:** Minimum 16 characters recommended
4. **Backup your config:** Keep a backup of `tgcf.config.json` after configuration

🐛 Troubleshooting

Issue: Can't access the web interface

- **Solution:** Ensure port 8501 is not blocked by firewall
- Check if the application is running: `ps aux | grep streamlit`

Issue: Authentication errors

- **Solution:** Double-check your API credentials from my.telegram.org
- Ensure you're using the correct account type (Bot vs User)

Issue: Watermark/OCR not working

- **Solution:** Install additional dependencies:
  ```bash
  # For Ubuntu/Debian
  sudo apt-get install ffmpeg tesseract-ocr
```

For macOS
brew install ffmpeg tesseract
```

### Issue: Dependency version errors (e.g., jinja2)

- **Problem:** Error like "Pandas requires version '3.1.2' or newer of 'jinja2'"
- **Cause:** Running `tgcf-web` directly uses system pip, not the poetry environment

- **Solution:** Always use `poetry run tgcf-web` or the provided scripts:
  ```bash
  # Correct ways to run:
  poetry run tgcf-web # Direct execution with poetry
  ./tgcf-start.sh # Using the start script
  ./start # Using the simple start script
  ```

If dependencies are out of date:
poetry install # Reinstall dependencies
poetry lock # Regenerate lock file if needed
```

- **Note:** The project includes jinja2>=3.1.2 to ensure pandas compatibility

Issue: Need help or found a bug

- Check the GitHub issues: https://github.com/aahnik/tgcf/issues
- Join discussions: https://github.com/aahnik/tgcf/discussions

---

🎯 Quick Start Checklist

- [ ] Access the web interface at http://localhost:8501
- [ ] Login with password: `tgcf_demo_password_2025`
- [ ] Get API credentials from https://my.telegram.org
- [ ] Create a bot with @BotFather (if using bot mode)
- [ ] Enter credentials in " 🔑 Telegram Login" page
- [ ] Configure forwarding rules in " 🔗 Connections" page
- [ ] (Optional) Enable plugins in " 🧪 Plugins" page
- [ ] Start forwarding in " 🏃 Run" page
- [ ] Monitor and enjoy automated message forwarding!

Enjoy using TGCF! 🎉