# Project Structure

instagram-workflow/

├── src/

│ ├── main.py

│ ├── gui/

│ │ ├── \_\_init\_\_.py

│ │ ├── main\_window.py

│ │ ├── review\_widget.py

│ │ └── config\_dialog.py

│ ├── core/

│ │ ├── \_\_init\_\_.py

│ │ ├── photo\_scanner.py

│ │ ├── caption\_generator.py

│ │ ├── instagram\_api.py

│ │ └── scheduler.py

│ └── database/

│ ├── \_\_init\_\_.py

│ └── models.py

├── config/

│ └── settings.json

├── photos/

├── requirements.txt

└── README.md

# Prerequisites

# Create virtual environment

python -m venv venv

source venv/bin/activate # On Windows: venv\Scripts\activate

# Install dependencies

pip install -r requirements.txt

# Instagram Business API Setup

1. **Create Meta App:**
   * Go to [Meta for Developers](https://developers.facebook.com/apps)
   * Create new app → Business → Connect Instagram Business API
   * Add products: Instagram Basic Display API + Instagram Graph API
2. **Get Access Tokens:**
   * Generate User Access Token with these permissions:
     + instagram\_basic
     + instagram\_content\_publish
     + pages\_show\_list
     + pages\_read\_engagement
   * Get Instagram Business Account ID from Graph API Explorer

# Store credentials in .env:

bash

Copy

INSTAGRAM\_ACCESS\_TOKEN=your\_long\_lived\_access\_token

INSTAGRAM\_BUSINESS\_ACCOUNT\_ID=your\_account\_id

OPENAI\_API\_KEY=your\_openai\_key

# Database Setup

# database/models.py

from sqlalchemy import create\_engine, Column, Integer, String, DateTime, Boolean, Text

from sqlalchemy.ext.declarative import declarative\_base

from datetime import datetime

Base = declarative\_base()

class Photo(Base):

\_\_tablename\_\_ = 'photos'

id = Column(Integer, primary\_key=True)

filename = Column(String(255), unique=True)

filepath = Column(String(500))

caption = Column(Text)

hashtags = Column(Text)

status = Column(String(20)) # pending, approved, posted, failed

scheduled\_time = Column(DateTime)

posted\_time = Column(DateTime)

instagram\_post\_id = Column(String(100))

created\_at = Column(DateTime, default=datetime.utcnow)

# Core Implementation

## Main Application Entry Point

# main.py

import sys

from PyQt5.QtWidgets import QApplication

from gui.main\_window import MainWindow

from database.models import init\_db

if \_\_name\_\_ == "\_\_main\_\_":

init\_db()

app = QApplication(sys.argv)

window = MainWindow()

window.show()

sys.exit(app.exec\_())