Purpose:

GuardianID is a technology-driven solution designed to help identify elderly individuals quickly and compassionately, especially in situations where they may be lost, confused, or unable to communicate. It combines facial recognition and QR code technology to create personalized ID cards that link to emergency contact and medical information.

How It Works

Facial Recognition

A photo of the individual is processed using facial recognition software to create a unique digital identity.

QR Code Generation

A QR code is created that links to the person’s name, emergency contact, and other relevant details.

ID Card Creation

The photo, name, contact info, and QR code are compiled into a printable ID card that can be worn or carried.

Flask API Integration

A web-based interface allows caregivers or responders to upload a photo and retrieve the person’s identity and contact info instantly

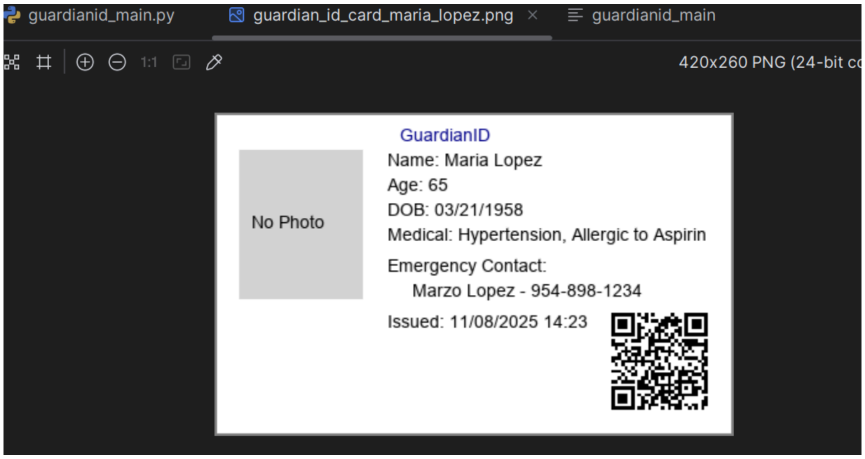
Benefits

- Safety: Helps prevent wandering and ensures quick reunification with caregivers.
- Dignity: Offers a respectful way to identify individuals without invasive tracking.
- Accessibility: Easy to use for families, healthcare workers, and community responders.
- Scalability: Can be expanded to include mobile apps, secure databases, and partnerships with senior centers.

Code for PyCharm (Python Program 3.14.0):

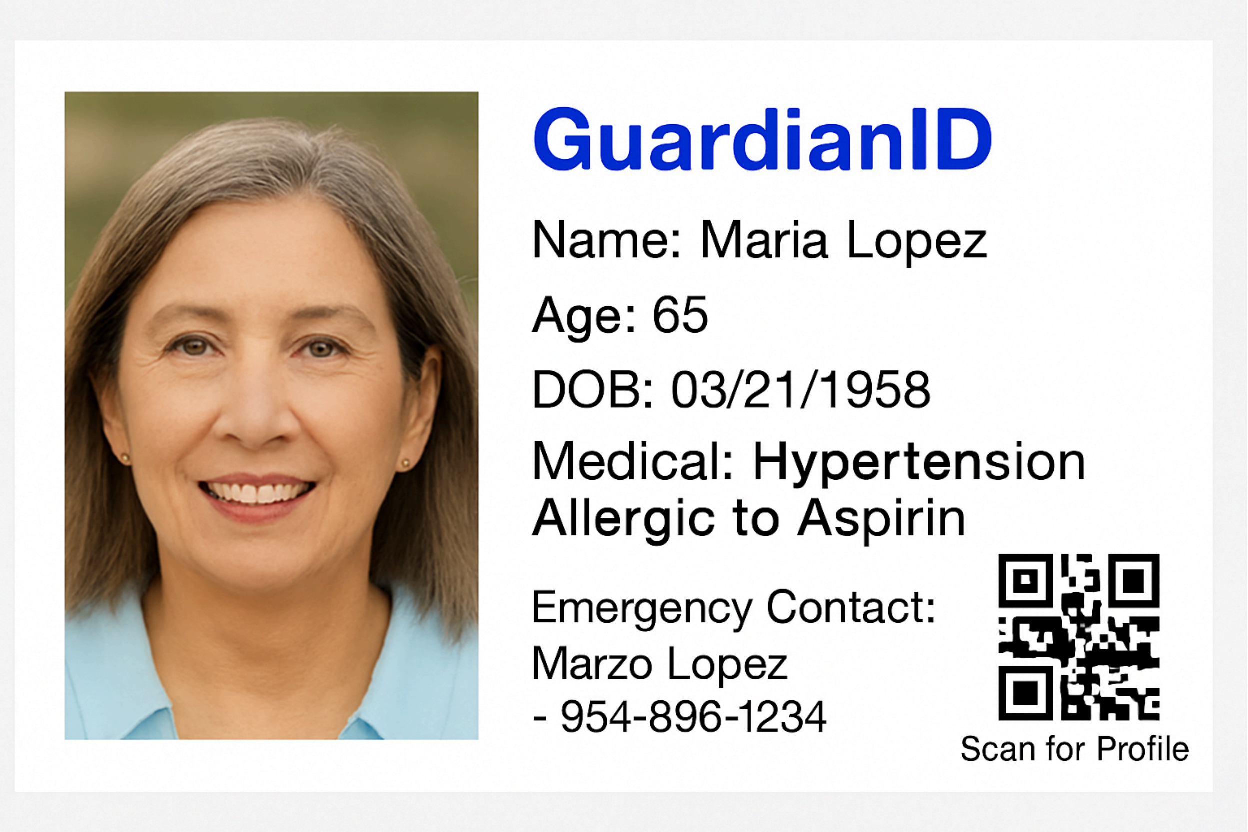
```
from PIL import Image, ImageDraw, ImageFont
import qrcode
import os
from datetime import datetime
import textwrap

# ------------------------------
# STEP 1: User Information
# ------------------------------
user_data = {
    "name": "Maria Lopez",
    "age": 65,
    "dob": "03/21/1958",
    "medical": "Hypertension, Allergic to Aspirin",
    "emergency_contact": "Marzo Lopez - 954-898-1234",
    "profile_link": "http://localhost:5000/guardianid",
    "photo_path": "assets/maria_photo.jpg"  # Ensure this image exists
}

# ------------------------------
# STEP 2: Generate GuardianID Card
# ------------------------------
def generate_id_card(data):
    # Canvas setup
    width, height = 420, 260
    card = Image.new("RGB", (width, height), "white")
    draw = ImageDraw.Draw(card)

    # Font setup (smaller for better fit)
    try:
        font = ImageFont.truetype("arial.ttf", 14)
    except:
        font = ImageFont.load_default()

    # Header
    draw.text((width // 2 - 60, 10), "GuardianID", font=font, fill="darkblue")

    # Photo
    try:
        photo = Image.open(data["photo_path"]).resize((100, 120))
    except FileNotFoundError:
        photo = Image.new("RGB", (100, 120), "lightgray")
        ImageDraw.Draw(photo).text((10, 50), "No Photo", font=font, fill="black")
    card.paste(photo, (20, 30))

    # Text layout
    x_text = 140
    y_text = 30
    line_spacing = 20

    # Basic Info
    draw.text((x_text, y_text), f"Name: {data['name']}", font=font, fill="black")
    draw.text((x_text, y_text + line_spacing), f"Age: {data['age']}", font=font, fill="black")
    draw.text((x_text, y_text + 2 * line_spacing), f"DOB: {data['dob']}", font=font, fill="black")

    # Medical Info (wrapped)
    medical_lines = textwrap.wrap(data["medical"], width=34)
    for i, line in enumerate(medical_lines):
        prefix = "Medical: " if i == 0 else "         "
        draw.text((x_text, y_text + (3 + i) * line_spacing), f"{prefix}{line}", font=font,
fill="black")

    # Emergency Contact (wrapped)
    contact_y = y_text + (3 + len(medical_lines)) * line_spacing + 5
    draw.text((x_text, contact_y), "Emergency Contact:", font=font, fill="black")
    contact_lines = textwrap.wrap(data["emergency_contact"], width=34)
    for i, line in enumerate(contact_lines):
        draw.text((x_text + 20, contact_y + (i + 1) * line_spacing), line, font=font, fill="black")

    # Timestamp
    timestamp_y = contact_y + (len(contact_lines) + 1) * line_spacing + 5
    draw.text((x_text, timestamp_y), f"Issued: {datetime.now().strftime('%m/%d/%Y %H:%M')}", font=font, fill="black")

    # QR Code — bottom-right corner with padding
    qr = qrcode.make(data["profile_link"]).resize((100, 100))
    qr_x = width - 110  # 10px from right edge
    qr_y = height - 110  # 10px from bottom edge
    card.paste(qr, (qr_x, qr_y))

    # Border
    draw.rectangle([(0, 0), (width - 1, height - 1)], outline="gray", width=2)

    # Save card
    os.makedirs("generated_cards", exist_ok=True)
    filename = f"generated_cards/guardian_id_card_{data['name'].lower().replace(' ', '_')}.png"
    card.save(filename)
    print(f"✅ ID card saved as {filename}")

# -------------------------------
# STEP 3: Execute
# -------------------------------
if __name__ == "__main__":
    generate_id_card(user_data)
```

Results:



GuardianID
Name: Maria Lopez
Age: 65
DOB: 03/21/1958
Medical: Hypertension, Allergic to Aspirin
Emergency Contact:
Marzo Lopez - 954-898-1234
Issued: 11/08/2025 14:23

Future Results Outcome once GuardianID program has been updated:



GuardianID

Name: Maria Lopez

Age: 65

DOB: 03/21/1958

Medical: **Hypertension**
Allergic to Aspirin

Emergency Contact:
Marzo Lopez
- 954-896-1234

Scan for Profile