**1. Deploy to a Free Hosting Platform**

Two popular options for free or low-cost hosting are:

**Option 1: Render**

1. **Sign Up**:
   * Go to [Render](https://render.com/) and create an account.
2. **Prepare Your Code**:
   * Ensure all your code is in a Git repository (e.g., on GitHub or GitLab).
3. **Create a New Web Service**:
   * Log in to Render and select **New + > Web Service**.
   * Connect your GitHub or GitLab repository containing the Flask app.
   * Select the branch where your code resides.
4. **Set Up Build and Start Commands**:
   * Build command: Install required libraries:

bash

Copy code

pip install -r requirements.txt

* + Start command: Specify Flask app:

bash

Copy code

python app.py

1. **Deploy**:
   * Deploy the app and note the public URL (e.g., https://yourappname.onrender.com).

**Option 2: PythonAnywhere**

1. **Sign Up**:
   * Go to [PythonAnywhere](https://www.pythonanywhere.com/) and create an account.
2. **Upload Your Files**:
   * Log in and upload your project files (app.py, SiS1.py, NewtonRaphson.py, templates/) to your PythonAnywhere account.
3. **Set Up a Web App**:
   * Go to the **Web** tab and create a new web app.
   * Choose **Flask** as the framework and the Python version you're using.
4. **Point to Your App**:
   * Specify the path to your app.py file in the **WSGI configuration file**.
5. **Run Your App**:
   * Save your changes and start the web app.
   * Your app will be live at https://yourusername.pythonanywhere.com.

**2. Deploy to Cloud Providers**

**Option 1: Heroku**

1. **Install Heroku CLI**:
   * Download and install the Heroku CLI.
2. **Prepare Your Project**:
   * Add a requirements.txt:

bash

Copy code

pip freeze > requirements.txt

* + Add a Procfile to specify the Flask app:

makefile

Copy code

web: python app.py

* + Ensure all files are committed to a Git repository.

1. **Create a Heroku App**:

bash

Copy code

heroku create

1. **Deploy the App**:

bash

Copy code

git push heroku main

1. **Access Your App**:
   * Heroku provides a public URL (e.g., https://yourappname.herokuapp.com).

**Option 2: AWS (Amazon Web Services)**

1. **Set Up an EC2 Instance**:
   * Launch a free-tier EC2 instance.
   * Install Python, Flask, and your app dependencies.
2. **Deploy Your Flask App**:
   * Copy your project files to the EC2 instance using SCP or SFTP.
   * Run your Flask app with gunicorn or flask run.
3. **Set Up a Domain**:
   * Use AWS Route 53 or another DNS provider to point a domain to your EC2 instance.

**3. Secure Your App**

For a live app, ensure:

1. **SSL Certificates**:
   * Use services like Let’s Encrypt to secure your site.
2. **Environment Variables**:
   * Store sensitive data (e.g., API keys) securely using .env files or environment variables.

**4. Next Steps**

Once deployed, share the public URL with users to access your live application.

Let me know which option you'd like help with, and I can guide you through it!