```python
import numpy as np
import os
from PIL import Image
from flask import Flask, request, render_template, url_for
from werkzeug.utils import secure_filename, redirect
#from gevent.pywsgi import WSGIServer
from keras.models import load_model
from keras.preprocessing import image
from flask import send_from_directory

UPLOAD_FOLDER ='data'

app = Flask(__name__,template_folder='static')
app.config['UPLOAD_FOLDER'] = UPLOAD_FOLDER

model = load_model('models\mnistCNN.h5')

@app.route('/')
def index():
return render_template('index.html')

@app.route('/predict', methods=['GET', 'POST'])
def upload():
if request.method == 'POST':
f = request.files["image"]
filepath = secure_filename(f.filename)
f.save(os.path.join(app.config['UPLOAD_FOLDER'], filepath))

upload_img = os.path.join(UPLOAD_FOLDER, filepath)
img = Image.open(upload_img).convert("L") # convert image to monochrome
img = img.resize((28, 28)) # resizing of input image

im2arr = np.array(img) # converting to image
im2arr = im2arr.reshape(1, 28, 28, 1) # reshaping according to our requirement
```

```python
    pred = model.predict(im2arr)

    num = np.argmax(pred, axis=1) # printing our Labels

    return render_template('predict.html', num=str(num[0]))


if __name__ == '__main__':
    app.run(debug=True)
```