```
import os
import random
import string
from pathlib import Path
import numpy as np
from tensorflow.keras.models import load_model
from PIL import Image, ImageOps
def random_name_generator(n):
    return ''.join(random.choices(string.ascii_uppercase + string.digits, k=n))
def recognize(image):
    model=load_model(Path("model/mnistCNN.h5"))
    img = Image.open(image).convert("L")
    img_name = random_name_generator(10) + '.jpg'
    if not os.path.exists(f"static/data/"):
        os.mkdir(os.path.join('static/', 'data'))
    else:
        img.save(Path(f"static/data/{img_name}"))
        img = ImageOps.grayscale(img)
        img = ImageOps.invert(img)
        img = img.resize((28, 28))
        img2arr = np.array(img)
        img2arr = img2arr / 255.0
        img2arr = img2arr.reshape(1, 28, 28, 1)
        results = model.predict(img2arr)
```

```
best = np.argmax(results,axis = 1)[0]

pred = list(map(lambda x: round(x*100, 2), results[0]))

values = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

others = list(zip(values, pred))

best = others.pop(best)

return best, img_name
```