```python
import requests

from bs4 import BeautifulSoup as bs

import os

from flask import Flask, request, render_template, send_from_directory


APP_ROOT = os.path.dirname(os.path.abspath(__file__))

app = Flask(__name__)


@app.route("/")

def index():

    return render_template("upload.html")


# @app.route('/')

# def index():

#    return render_template('index.html')


@app.route("/upload", methods=["POST"])

def upload():

    target = os.path.join(APP_ROOT, 'images/')


    if not os.path.isdir(target):

        os.mkdir(target)

    else:

        pass
```

```python
    for upload in request.files.getlist("file"):

        filename = upload.filename

        destination = "/".join([target, filename])

        upload.save(destination)


    return render_template("complete.html", image_name=filename)


@app.route('/upload/<filename>')

def send_image(filename):

    return send_from_directory("images", filename)


@app.route('/gallery')

def get_gallery():

    image_names = os.listdir('./images')

    return render_template("gallery.html", image_names=image_names)


@app.route('/about')

def about():

    return render_template('about.html')


@app.route('/contact')

def contact():

    return render_template('contact.html')


@app.route('/home')

def home():
```

```python
        return render_template('home.html')


@app.route('/skills')

def skills():

    return render_template('skills.html')


@app.route('/news')

def news():


    link = 'https://inshorts.com/en/read'

    req = requests.get(link)


    soup = bs(req.content, 'html5lib')

    box = soup.findAll('div', attrs = {'class':'news-card z-depth-1'})


    ha,ia,ba,la = [],[],[],[]

    for i in range(len(box)):

        h = box[i].find('span', attrs = {'itemprop':'headline'}).text


        m = box[i].find('div', attrs = {'class':'news-card-image'})

        m = m['style'].split("'")[1]


        b = box[i].find('div', attrs = {'itemprop':'articleBody'}).text

        l='link not found'


        try:
```

```python
            l = box[i].find('a', attrs = {'class':'source'})['href']
        except:
            pass


        ha.append(h)
        ia.append(m)
        ba.append(b)
        la.append(l)
    return render_template('news.html', ha=ha, ia=ia, ba=ba, la=la, len = len(ha))


@app.errorhandler(404)
def page_not_found(e):
    return render_template('404.html'), 404


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