

```
<!DOCTYPE html>

<html lang="en">

<head>

  <!-- Bootstrap 5 css-->

  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet"

    integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftJbbrCEXSU1oBoqyl2QvZ6jIW3"
crossorigin="anonymous">


  <!-- Local css -->

  <link rel="stylesheet" href="../static/style.css">

  <meta charset="UTF-8">

  <meta http-equiv="X-UA-Compatible" content="IE=edge">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Base</title>

</head>

<body>

  {% with messages = get_flashed_messages() %}

    {% if messages %}

      {% if flag == 1 %}

        {% for message in messages %}

          <div class="alert alert-danger" role="alert">

            {{message}}

          </div>

        {% endfor %}

      {% endif %}

    {% endif %}

  {% endwith %}
```

```

{% else %}

    {% for message in messages %}

        <div class="alert alert-success" role="alert">

            {{ message }}

        </div>

    {% endfor %}

{% endif %}

{% endif %}

{% endwith %}


<div class="position-relative">

    <div class="container my-2">

        <h1 class="my-3 h1title">SPAM EMAIL DETECTION</h1>

        <form class="row" method="POST">

            <div class="col-lg-12 my-5">

                <input type="text" name="SUB" class="form-control form-control-lg" id="SUB"
placeholder="SUBJECT">

            </div>

            <div class="col-lg-12 my-5">

                <textarea class="form-control" id="CON" name="CON" rows="8"
placeholder="CONTENT"></textarea>

            </div>

            <div class="col-6 my-3">

                <div class="d-flex justify-content-end">

                    <button type="submit" class="btn btn-outline-success btn-
lg">Predict</button>

                </div>

```

```

        </div>

    </form>

    <!-- {%for Username in username %}

        <h1>{{username}}</h1>

    {% endfor %} -->

</div>

</div>

<!-- Bootstrap 5 script -->

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"

    integrity="sha384-
ka7Sk0Gln4gmtz2MlQnikT1wXgY5Og+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"

    crossorigin="anonymous"></script>

</body>

</html>

```

```

from copyreg import pickle

from flask import Blueprint, render_template, request, flash

```

```

import nltk

from nltk.corpus import stopwords

import string, pickle

```

```

nltk.download('stopwords')

```

```

#loading classifier from pickle

```

```
with open("models/classifier.pkl", "rb") as f:
```

```
    classifier = pickle.load(f)
```

```
#loading cv from pickle
```

```
with open("models/vectorizer.pkl", "rb") as v:
```

```
    cv = pickle.load(v)
```

```
#processing and tokenizing
```

```
def process(text):
```

```
    nopunc = [char for char in text if char not in string.punctuation]
```

```
    nopunc = ''.join(nopunc)
```

```
    clean = [word for word in nopunc.split() if word.lower() not in stopwords.words('english')]
```

```
    return clean
```

```
views = Blueprint("views", __name__)
```

```
@views.route("/", methods = ["POST", "GET"])
```

```
def index():
```

```
    if request.method == "POST":
```

```
        subject = request.form["SUB"]
```

```
        content = request.form["CON"]
```

```
        text = subject + content
```

```
        processed_text = process(text)
```

```
        vectorized_text = cv.transform(processed_text)
```

```
        flag = classifier.predict(vectorized_text)[0]
```

```
if flag == 1:
    flash("This is a SPAM email.")
    print("spam")
    return render_template("base.html", flag = flag)
elif flag == 0:
    flash("This is a HAM email.")
    print("ham")
    return render_template("base.html", flag = flag)
print("get")
return render_template("base.html")
```

```
from flask import Flask
```

```
server = Flask(__name__)
```

```
server.config['SECRET_KEY'] = "ASJdjhadjhurvikodimagnahiahihgkghgk"
```

```
from .views import views
```

```
server.register_blueprint(views, url_prefix="/")
```

```
from distutils.log import debug
```

```
from server import server
```

```
if __name__ == "__main__":
```

```
    server.run()
```