Health Insurance Charges prediction model Deployment

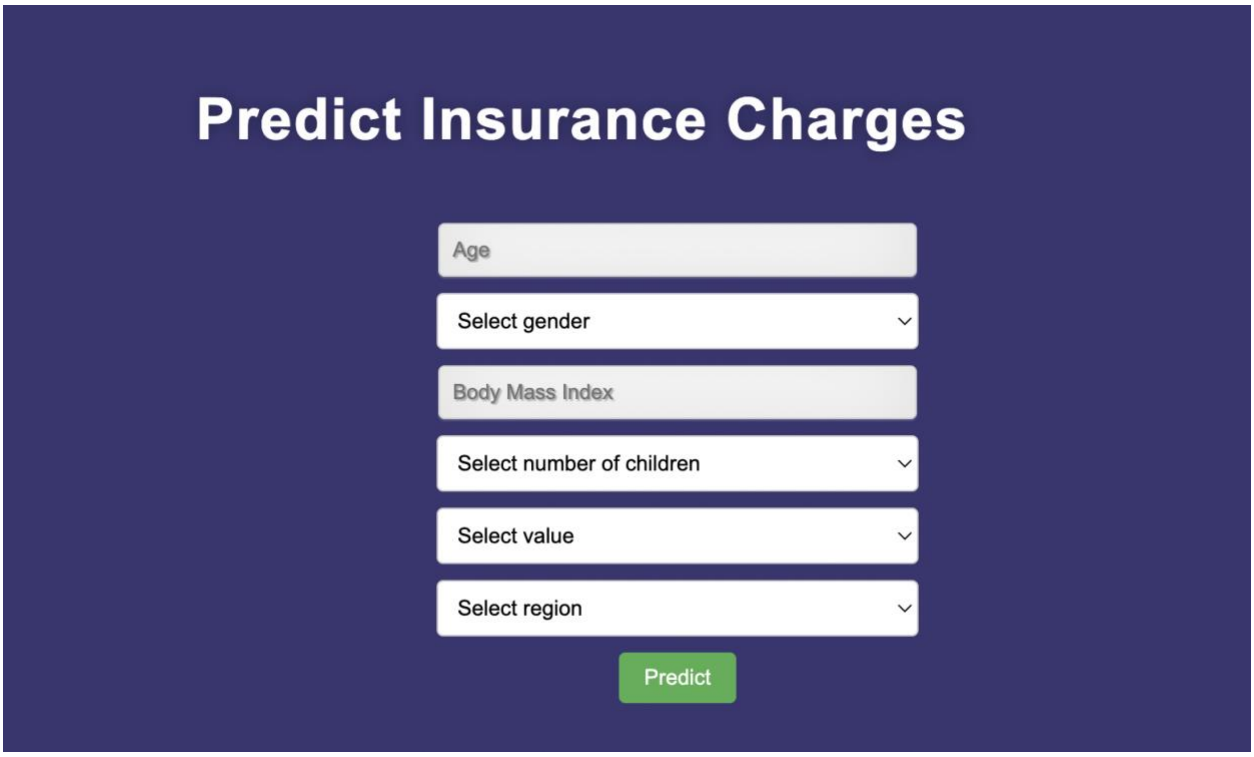
Step 1: Exploratory Analysis on the dataset

```
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 6 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1338 non-null   int64
 1   sex       1338 non-null   int64
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   int64
 5   region    1338 non-null   int64
dtypes: float64(1), int64(5)
memory usage: 62.8 KB
None
   age  sex     bmi  children  smoker  region
0   19    1  27.900         0       1       1
1   18    2  33.770         1       2       2
2   28    2  33.000         3       2       2
3   33    2  22.705         0       2       3
4   32    2  28.880         0       2       3
```

Step 2: Model Deployment using Flask and pickle.

```python
@app.route('/predict',methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    int_features = [int(request.form['age']),
                    categoricalValueToInt(request.form['sex']),
                    float(request.form['bmi']),
                    int(request.form['children']),
                    categoricalValueToInt(request.form['smoker']),
                    categoricalValueToInt(request.form['region'])]

    # Create a DataFrame from int_features
    test_df = pd.DataFrame([int_features], columns=['age','sex','bmi','children','smoker','region'])


    # Retrieve the encoded values from the DataFrame
    encoded_features = test_df.values

    # final_features = [np.array(encoded_features)]
    prediction = model.predict(encoded_features)

    output = round(prediction[0], 2)

    return render_template('index.html', prediction_text='Insurance charge should be $ {}'.format(output))
```

Step 3: Rendering web page using render template and request:



Step 4: Predicted Charges of insurance:

Predicted value for a person whose:
Age = 21
Gender = Male
BMI = 32.23
Children = 1
Smoker= No
Region = southwest