Project Development Phase - Sprint Delivery Plan

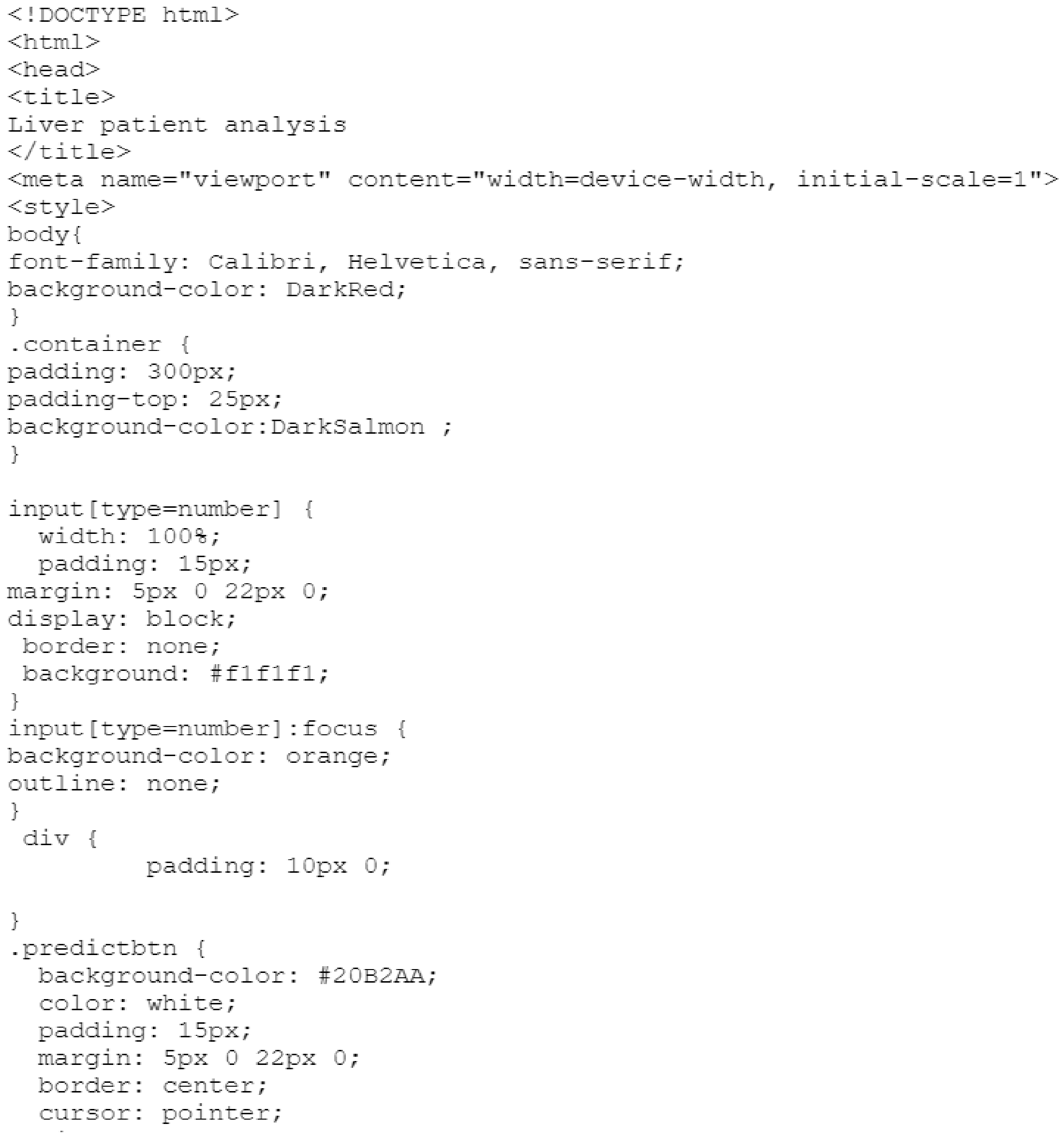
Sprint 4 – Application building & Train the model on IBM

Date	18 November 2022
Team ID	PNT2022TMID52974
Project Name	Statistical Machine Learning Approaches to Liver Disease Prediction

Application Building:

Application Building involves following steps

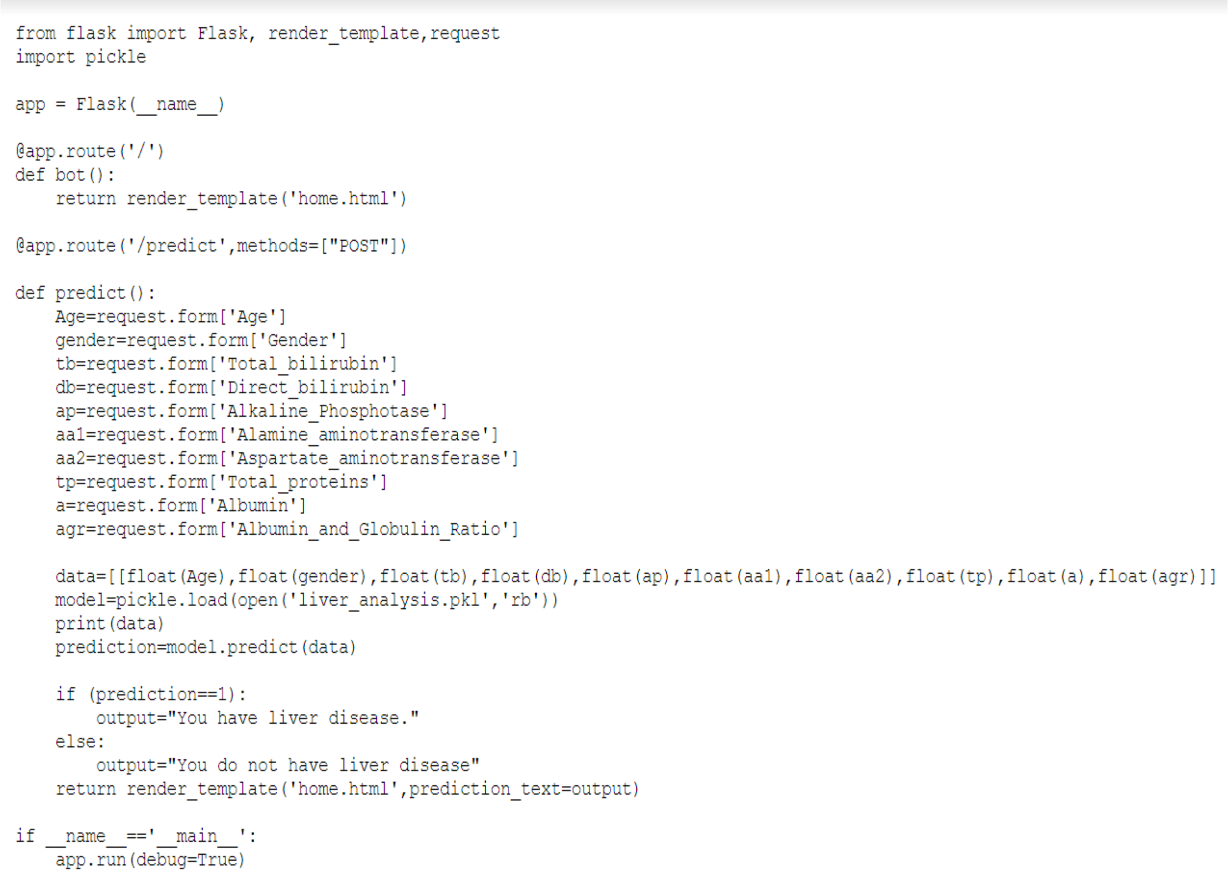
1. Create an HTML file



```
<!DOCTYPE html>
<html>
<head>
<title>
Liver patient analysis
</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body{
font-family: Calibri, Helvetica, sans-serif;
background-color: DarkRed;
}
.container {
padding: 300px;
padding-top: 25px;
background-color:DarkSalmon ;
}

input[type=number] {
  width: 100%;
  padding: 15px;
margin: 5px 0 22px 0;
display: block;
 border: none;
 background: #f1f1f1;
}
input[type=number]:focus {
background-color: orange;
outline: none;
}
 div {
         padding: 10px 0;

}
.predictbtn {
  background-color: #20B2AA;
  color: white;
  padding: 15px;
  margin: 5px 0 22px 0;
  border: center;
  cursor: pointer;
```

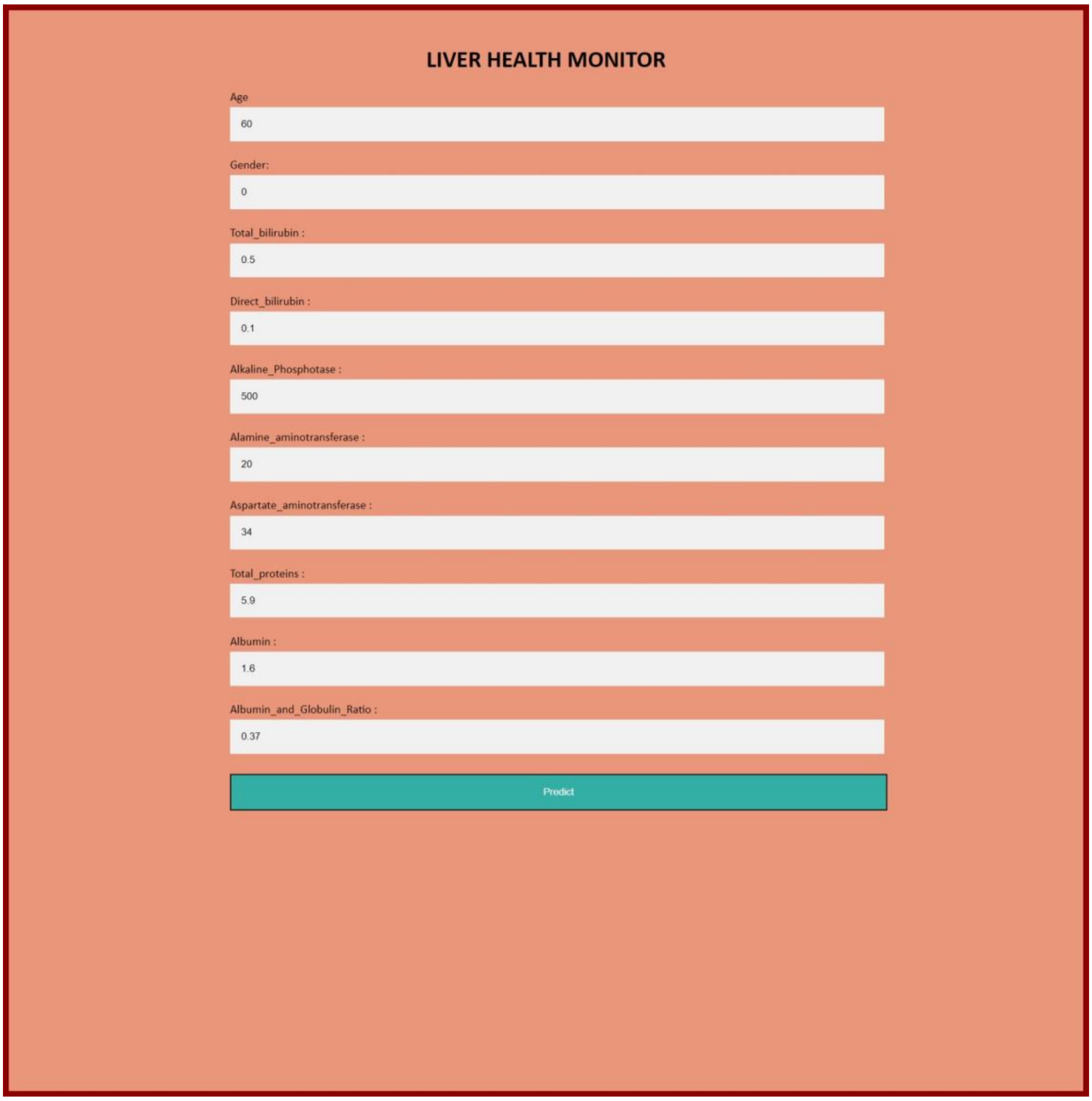
```
</style>
</head>
<body>
<div class="container">
<center>  <h1> LIVER HEALTH MONITOR</h1> </center>
<form action="home.html" method="post">
<label> Age </label>
<input type="number" step=0.01 name="Age" placeholder= "Age" size="15" required />

<label> Gender: </label>
<input type="number" step=0.01 name="Gender" placeholder="Gender (0 for male, 1 for female)" size="15"required />

<label>
Total_bilirubin :
</label>
<input type="number" step=0.01 name="Total_bilirubin" placeholder="Total_bilirubin" size="10" required>

<label>
Direct_bilirubin :
</label>
<input type="number" step=0.01 name="Direct_bilirubin" placeholder="Direct_bilirubin" size="10" required>

<label>
Alkaline_Phosphotase :
</label>
<input type="number" step=0.01 name="Alkaline_Phosphotase" placeholder="Alkaline_Phosphotase" size="10" required>

<label>
Alamine_aminotransferase :
</label>
<input type="number" step=0.01 name="Alamine_aminotransferase" placeholder="Alamine_aminotransferase" size="10" required>


<label>
Aspartate_aminotransferase :
</label>
<input type="number" step=0.01 name="Aspartate_aminotransferase" placeholder="Aspartate_aminotransferase" size="10" required>

<label>
Total_proteins :
</label>
<input type="number" step=0.01 name="Total_proteins" placeholder="Total_proteins" size="10" required>
```

2. Build a Python Code



```python
from flask import Flask, render_template,request
import pickle

app = Flask(__name__)

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

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

def predict():
    Age=request.form['Age']
    gender=request.form['Gender']
    tb=request.form['Total_bilirubin']
    db=request.form['Direct_bilirubin']
    ap=request.form['Alkaline_Phosphotase']
    aa1=request.form['Alamine_aminotransferase']
    aa2=request.form['Aspartate_aminotransferase']
    tp=request.form['Total_proteins']
    a=request.form['Albumin']
    agr=request.form['Albumin_and_Globulin_Ratio']

    data=[[float(Age),float(gender),float(tb),float(db),float(ap),float(aa1),float(aa2),float(tp),float(a),float(agr)]]
    model=pickle.load(open('liver_analysis.pkl','rb'))
    print(data)
    prediction=model.predict(data)

    if (prediction==1):
        output="You have liver disease."
    else:
        output="You do not have liver disease"
    return render_template('home.html',prediction_text=output)

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

3. Run the app



LIVER HEALTH MONITOR

Age
60

Gender:
0

Total_bilirubin :
0.5

Direct_bilirubin :
0.1

Alkaline_Phosphotase :
500

Alamine_aminotransferase :
20

Aspartate_aminotransferase :
34

Total_proteins :
5.9

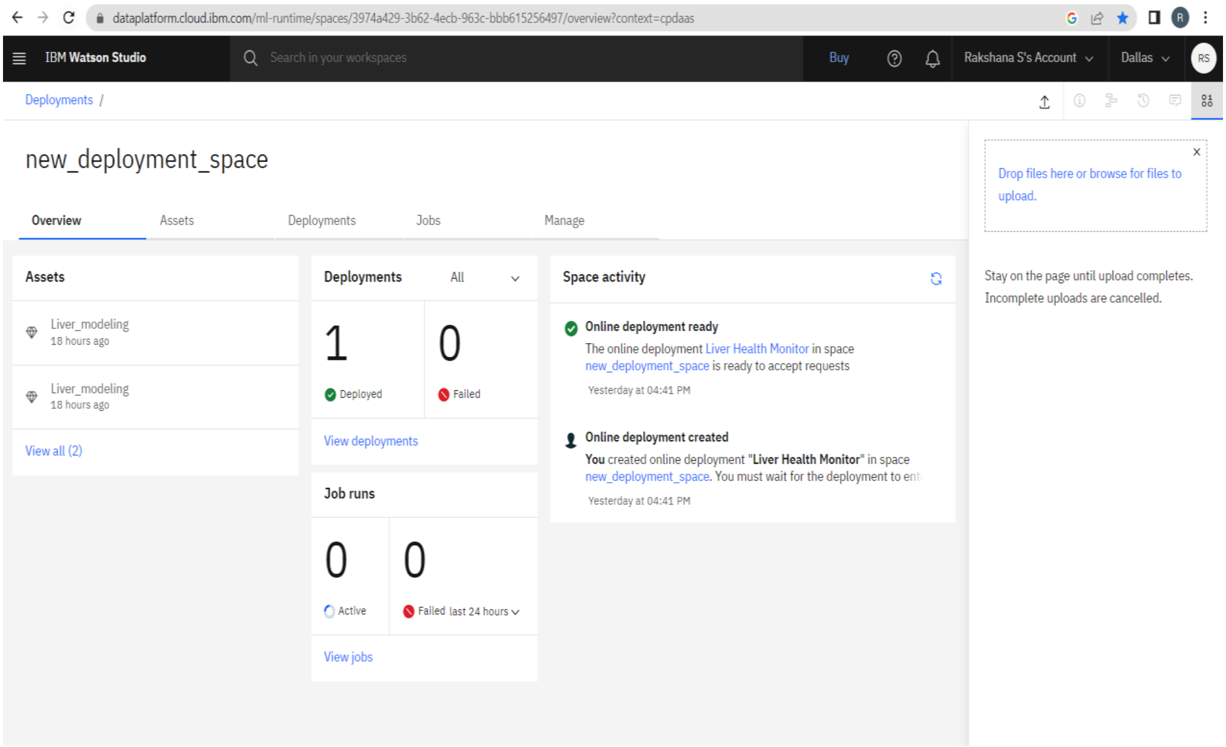
Albumin :
1.6

Albumin_and_Globulin_Ratio :
0.37

Train the model on IBM

- To build a Machine Learning Model and deploy it on the IBM Cloud.

```
!pip install ibm_watson_machine_learning
from ibm_watson_machine_learning import APIClient
wml_credentials={"url":"https://us-
south.ml.cloud.ibm.com","apikey":"eJqAgq9PC4DNCsDnTj0pfeO4-Rk0jXwNWF61-
LDwFYDL"}
client=APIClient(wml_credentials)
def guid_from_space_name(client,space_name):
    space=client.spaces.get_details()
    return(next(item for item in space['resources'] if
item['entity']["name"]==space_name)['metadata']['id'])
space_uid=guid_from_space_name(client,'new_deployment_space')
print("Space UID ="+space_uid)
client.set.default_space(space_uid)
client.software_specifications.list()
software_spec_uid=client.software_specifications.get_uid_by_name("runti
me-22.1-py3.9")
software_spec_uid
model_details = client.repository.store_model(model=svm,meta_props={
    client.repository.ModelMetaNames.NAME:"Liver_modeling",
    client.repository.ModelMetaNames.TYPE:"scikit-learn_1.0",

client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid})

model_id = client.repository.get_model_uid(model_details)
```


dataplatform.cloud.ibm.com/ml-runtime/spaces/3974a429-3b62-4ecb-963c-bbb615256497/overview?context=cpdaas
IBM Watson Studio
Search in your workspaces
Buy
Rakshana S's Account
Dallas
RS
Deployments /
Overview
Assets
Deployments
Jobs
Manage
Drop files here or browse for files to upload.

new_deployment_space

Assets

Liver_modeling 18 hours ago

Liver_modeling 18 hours ago

View all (2)

Deployments All

1 Deployed

0 Failed

View deployments

Job runs

0 Active

0 Failed last 24 hours

View jobs

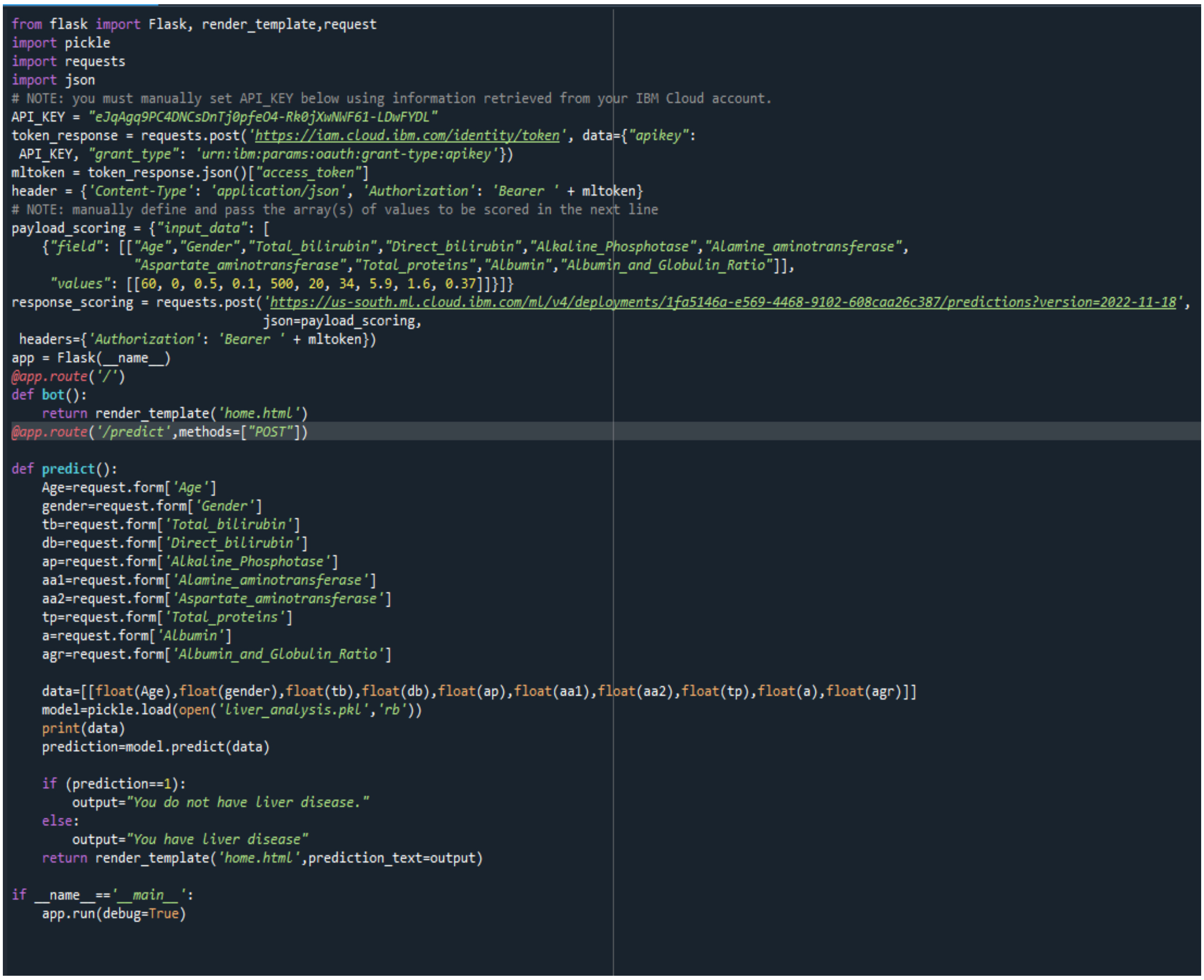
Space activity

Online deployment ready
The online deployment Liver Health Monitor in space new_deployment_space is ready to accept requests
Yesterday at 04:41 PM

Online deployment created
You created online deployment "**Liver Health Monitor**" in space new_deployment_space. You must wait for the deployment to ent
Yesterday at 04:41 PM

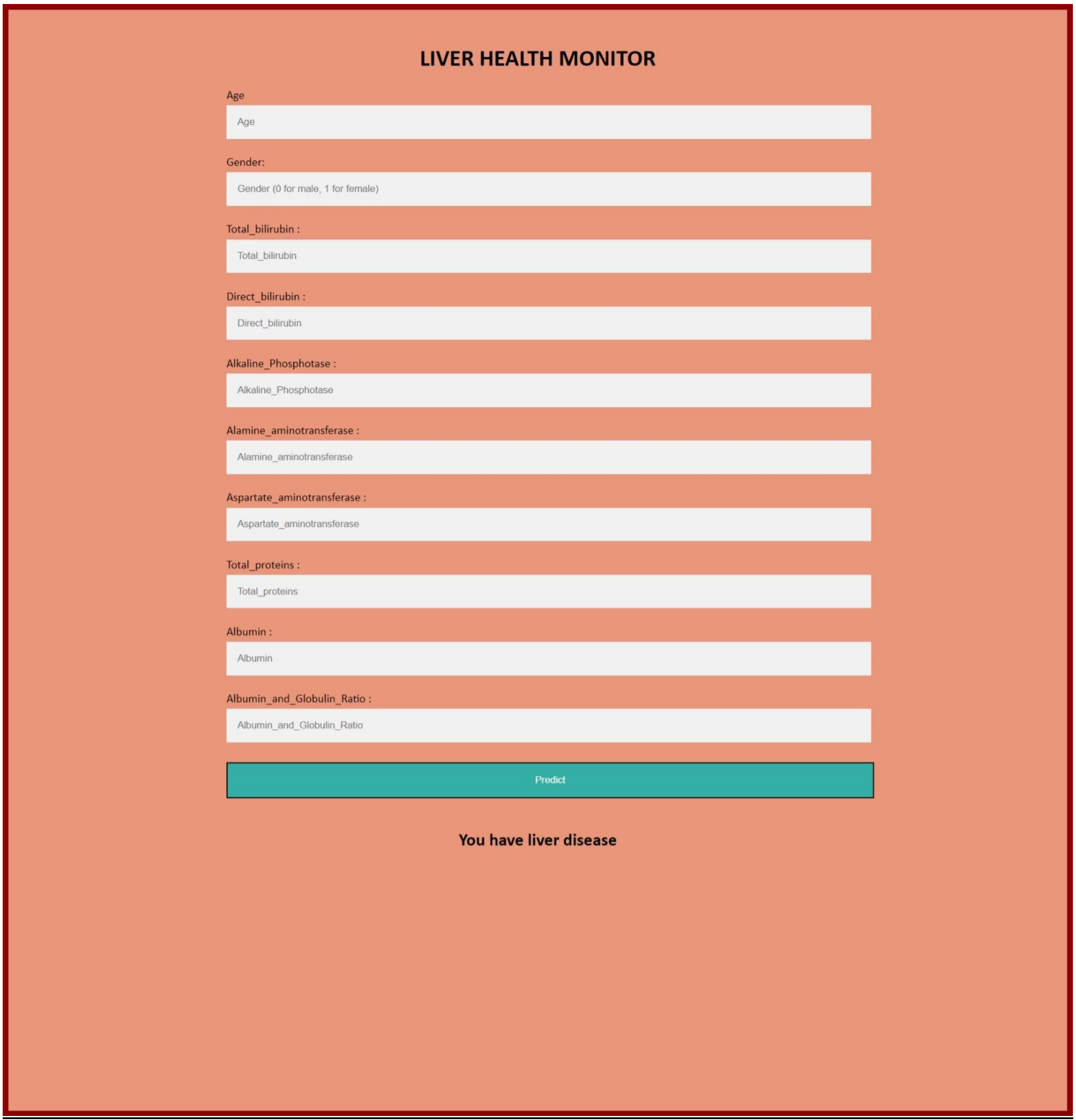
Stay on the page until upload completes. Incomplete uploads are cancelled.

- Integrate Flask with Scoring End points



```python
from flask import Flask, render_template,request
import pickle
import requests
import json
# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "eJqAgq9PC4DNCsDnTj0pfeO4-Rk0jXwNWF61-LDwFYDL"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]
header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}
# NOTE: manually define and pass the array(s) of values to be scored in the next line
payload_scoring = {"input_data": [
    {"field": [["Age","Gender","Total_bilirubin","Direct_bilirubin","Alkaline_Phosphotase","Alamine_aminotransferase",
                "Aspartate_aminotransferase","Total_proteins","Albumin","Albumin_and_Globulin_Ratio"]],
     "values": [[60, 0, 0.5, 0.1, 500, 20, 34, 5.9, 1.6, 0.37]]}]}
response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/1fa5146a-e569-4468-9102-608caa26c387/predictions?version=2022-11-18',
                                 json=payload_scoring,
 headers={'Authorization': 'Bearer ' + mltoken})
app = Flask(__name__)
@app.route('/')
def bot():
    return render_template('home.html')
@app.route('/predict',methods=["POST"])

def predict():
    Age=request.form['Age']
    gender=request.form['Gender']
    tb=request.form['Total_bilirubin']
    db=request.form['Direct_bilirubin']
    ap=request.form['Alkaline_Phosphotase']
    aa1=request.form['Alamine_aminotransferase']
    aa2=request.form['Aspartate_aminotransferase']
    tp=request.form['Total_proteins']
    a=request.form['Albumin']
    agr=request.form['Albumin_and_Globulin_Ratio']

    data=[[float(Age),float(gender),float(tb),float(db),float(ap),float(aa1),float(aa2),float(tp),float(a),float(agr)]]
    model=pickle.load(open('liver_analysis.pkl','rb'))
    print(data)
    prediction=model.predict(data)

    if (prediction==1):
        output="You do not have liver disease."
    else:
        output="You have liver disease"
    return render_template('home.html',prediction_text=output)

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

Final Output:



LIVER HEALTH MONITOR

Age

Age

Gender:

Gender (0 for male, 1 for female)

Total_bilirubin :

Total_bilirubin

Direct_bilirubin :

Direct_bilirubin

Alkaline_Phosphotase :

Alkaline_Phosphotase

Alamine_aminotransferase :

Alamine_aminotransferase

Aspartate_aminotransferase :

Aspartate_aminotransferase

Total_proteins :

Total_proteins

Albumin :

Albumin

Albumin_and_Globulin_Ratio :

Albumin_and_Globulin_Ratio

You have liver disease