Build Python Code:

```python
import pickle
import cv2
from skimage import feature

waveModel = pickle.load(open("ML/models/spiralModel.pkl", "rb"))
spiralModel = pickle.load(open("ML/models/spiralModel.pkl", "rb"))

code2cat = {0:"healthy", 1:"parkinson"}


def processImage(imagePath):
    image = cv2.imread(imagePath)
    image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    image = cv2.resize(image, (200, 200))
    image = cv2.threshold (image, 0, 255, cv2.THRESH_BINARY_INV | cv2.THRESH_OTSU) [1]
    return image


def quantify_image(image):
    features = feature.hog(image, orientations=9, pixels_per_cell=(13, 13), cells_per_block=(4, 4), transform_sqrt=True, block_norm="L1")
    return features


def predict(imgPath, modelType):
    img = processImage(imgPath)
    imgFeatures = quantify_image(img)

    if modelType=="spiral":
        pre = spiralModel.predict([imgFeatures])

    if modelType=="wave":
        pre = spiralModel.predict([imgFeatures])

    return code2cat[pre[0]]
```

Python Flask code:

```python
from flask import Flask, request, redirect, flash, send_from_directory, render_template, send_file



app = Flask(__name__, static_url_path='/')
app.config['SECRET_KEY'] = 'secret_key'
app.config['UPLOAD_FOLDER'] = 'static/uploads/'
```

```python
from app import *
import os
from utils.predict import predict

@app.route("/", methods=["GET"])
def index():
    return render_template('index.html')



@app.route("/signin", methods=["POST", "GET"])
def signin():
    if request.method=="GET":
        return render_template('signin.html')


@app.route("/signup", methods=["POST", "GET"])
def signup():
    if request.method=="GET":
        return render_template('signup.html')


@app.route("/upload", methods=["POST", "GET"])
def upload():
    if request.method=="GET":
        return render_template('upload.html')
    if request.method=="POST":
        imgType = request.form["type"]
        file = request.files['file']

        imgPath = os.path.join(app.config['UPLOAD_FOLDER'], "upImage.jpg")
        file.save(imgPath)
        print(imgPath)
        prediction = predict(imgPath, imgType)
        return render_template('result.html', prediction=prediction, path='/uploads/upImage.jpg')
```