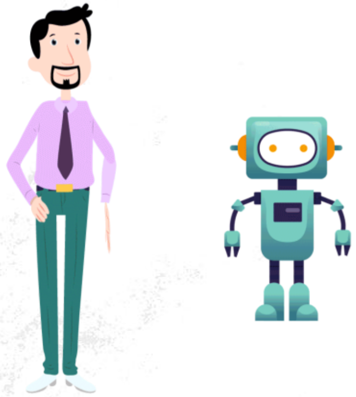
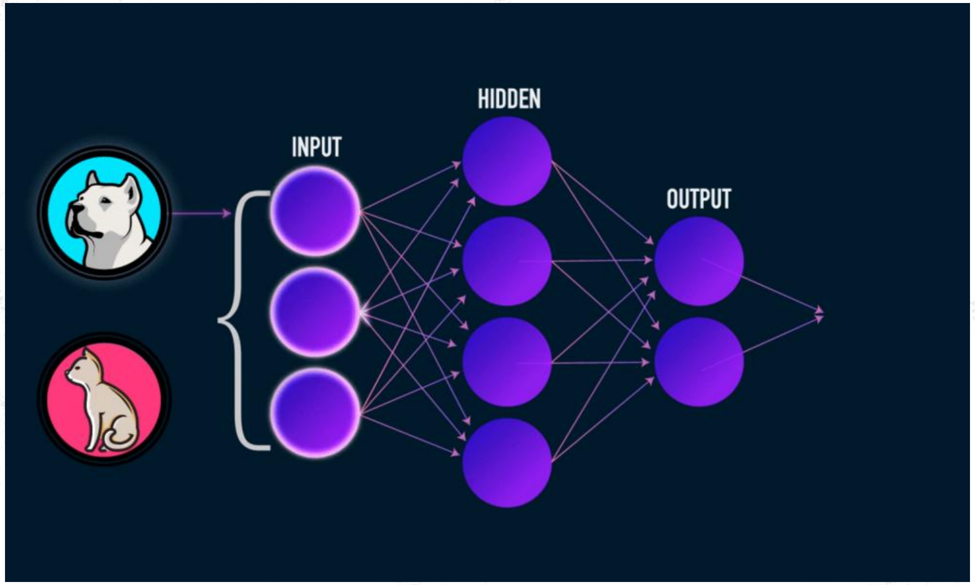


AI/ML PROJECTS

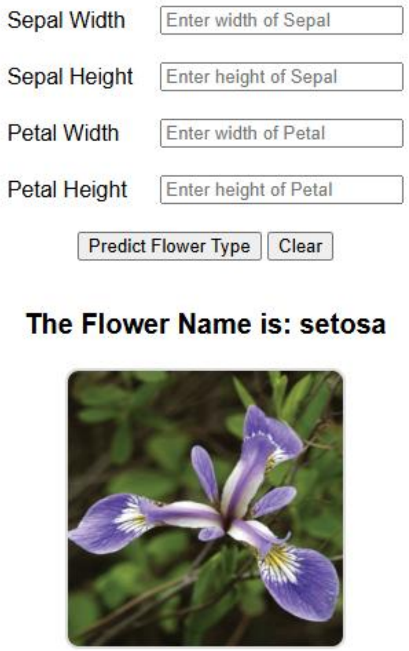


NIELIT Chandigarh/Ropar

"Innovating the Future: Transforming Ideas into AI-Powered Solutions"

Iris Flower Classification Project using ML

- Iris Classifier using Scikit-Learn: Built-in Dataset Approach
- **Source Code**: https://glitch.com/edit/#!/sklearn
- **Live View**: https://sklearn.glitch.me/



Sepal Width Enter width of Sepal

Sepal Height Enter height of Sepal

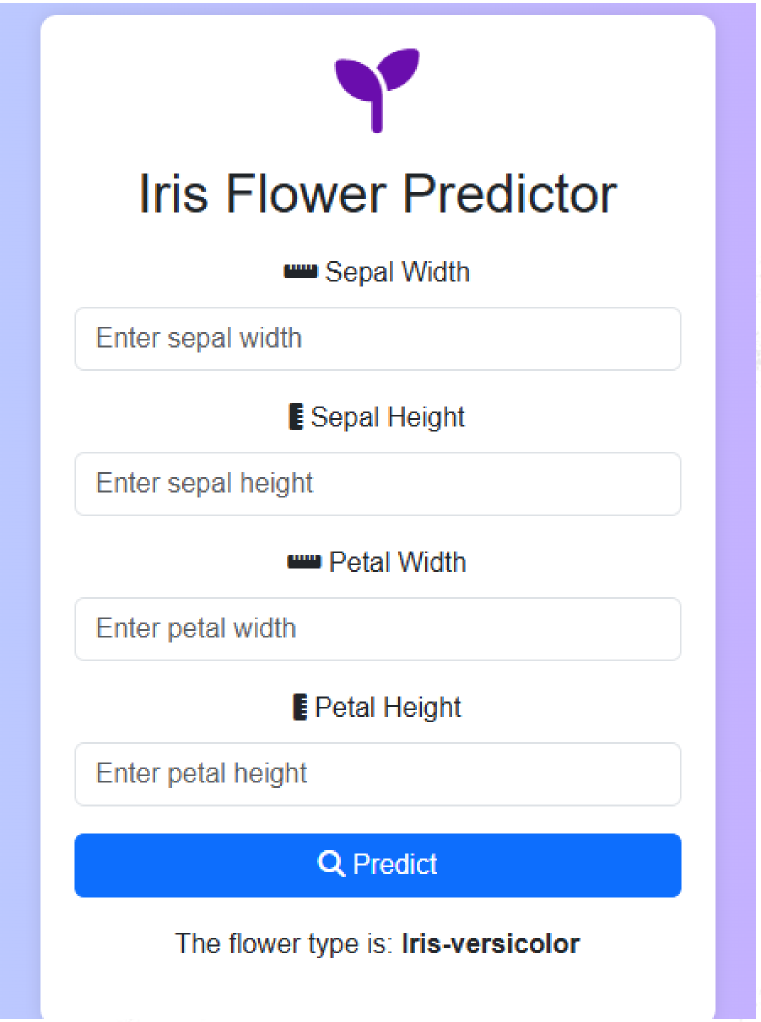
Petal Width Enter width of Petal

Petal Height Enter height of Petal

The Flower Name is: setosa

Classifier using Pandas: CSV-Based Data Handling

- Iris Classifier using Scikit-Learn: CSV-Based Data Handling Approach
- **Source Code**: https://glitch.com/edit/#!/helloworldprogramofmachinelearning
- **Live View**: https://helloworldprogramofmachinelearning.glitch.me/



Iris Flower Predictor

📏 Sepal Width

Enter sepal width

📏 Sepal Height

Enter sepal height

📏 Petal Width

Enter petal width

📏 Petal Height

Enter petal height

The flower type is: **Iris-versicolor**

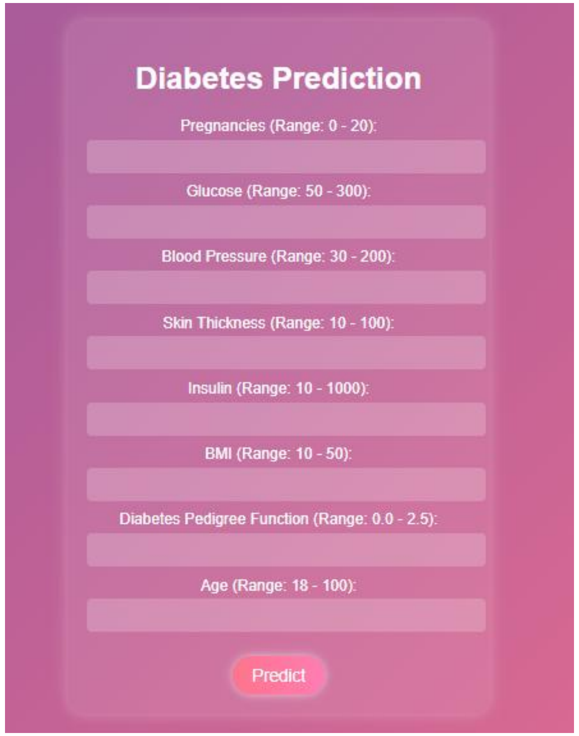
Diabetes Prediction: A Flask-Based ML App

- Diabetes Prediction Project Using Machine Learning. This app is a simple web application using the Flask framework, where users can input health data (like glucose levels, BMI, etc.) to predict if they are diabetic or not based on a Logistic Regression model.

- **Source Code**:

https://glitch.com/edit/#!/datasci

- **Live View**: https://datasci.glitch.me/



Diabetes Prediction

Pregnancies (Range: 0 - 20):

Glucose (Range: 50 - 300):

Blood Pressure (Range: 30 - 200):

Skin Thickness (Range: 10 - 100):

Insulin (Range: 10 - 1000):

BMI (Range: 10 - 50):

Diabetes Pedigree Function (Range: 0.0 - 2.5):

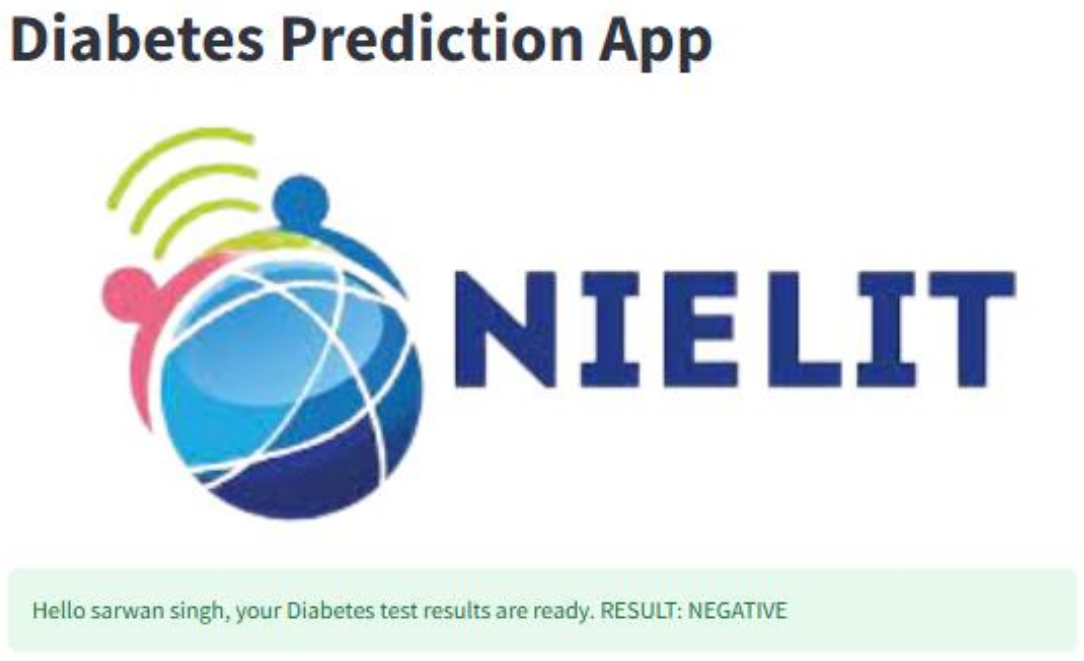
Age (Range: 18 - 100):

Diabetes Prediction App Using Machine Learning

- This AI-powered Diabetes Prediction App utilizes a Machine Learning model trained on medical data to predict the likelihood of diabetes based on patient details. Built with **Streamlit**, the app provides an intuitive interface where users can enter clinical parameters such as Glucose Level, BMI, Insulin, Blood Pressure, Age, and more to receive instant predictions.
- **Source Code**:

https://huggingface.co/spaces/nielitropar/diabetes/tree/main

- **Live View**: https://huggingface.co/spaces/nielitropar/diabetes

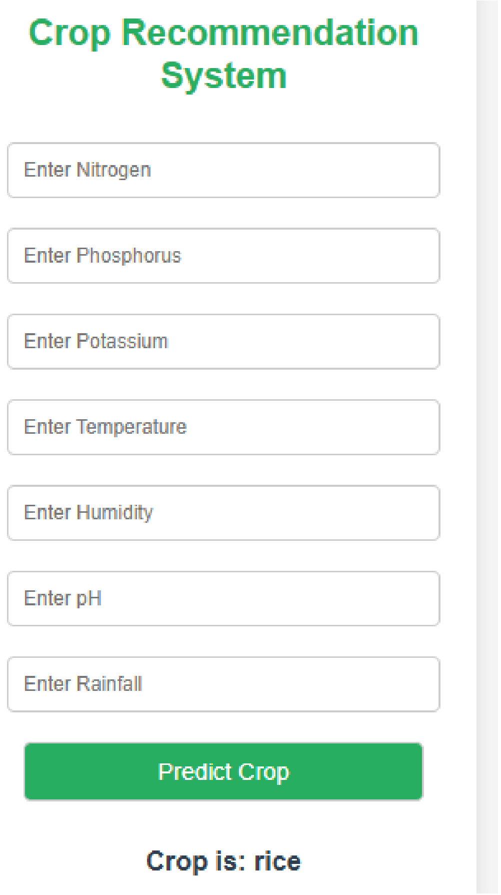


Diabetes Prediction App

Hello sarwan singh, your Diabetes test results are ready. RESULT: NEGATIVE

Smart Crop Recommendation System

- This Flask-based web application helps farmers and agricultural experts determine the best crop to grow based on soil composition and environmental conditions. Using a Logistic Regression model, it predicts the most suitable crop based on factors like NPK (Nitrogen, Phosphorus, Potassium) levels, temperature, humidity, pH, and rainfall.
- **Source Code**: https://glitch.com/edit/#!/croppredict
- **Live View**: https://croppredict.glitch.me/



Crop Recommendation System

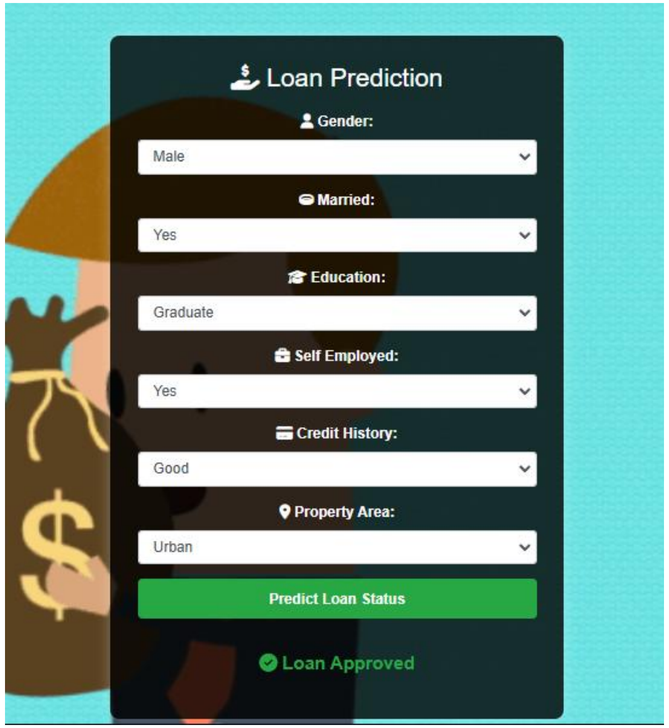
Crop is: rice

Loan Approval Prediction System

- The Loan Approval Prediction System is a Flask-based web application that utilizes Machine Learning (Random Forest Classifier) to predict whether a loan application will be approved or not. Users enter details such as Gender, Marital Status, Education, Employment Status, Credit History, and Property Area, and the system analyzes the input using a trained model to provide an approval prediction.
- **Source Code**:

https://glitch.com/edit/#!/loanprediction

- **Live View**: https://loanprediction.glitch.me/



Loan Prediction

Gender: Male

Married: Yes

Education: Graduate

Self Employed: Yes

Credit History: Good

Property Area: Urban

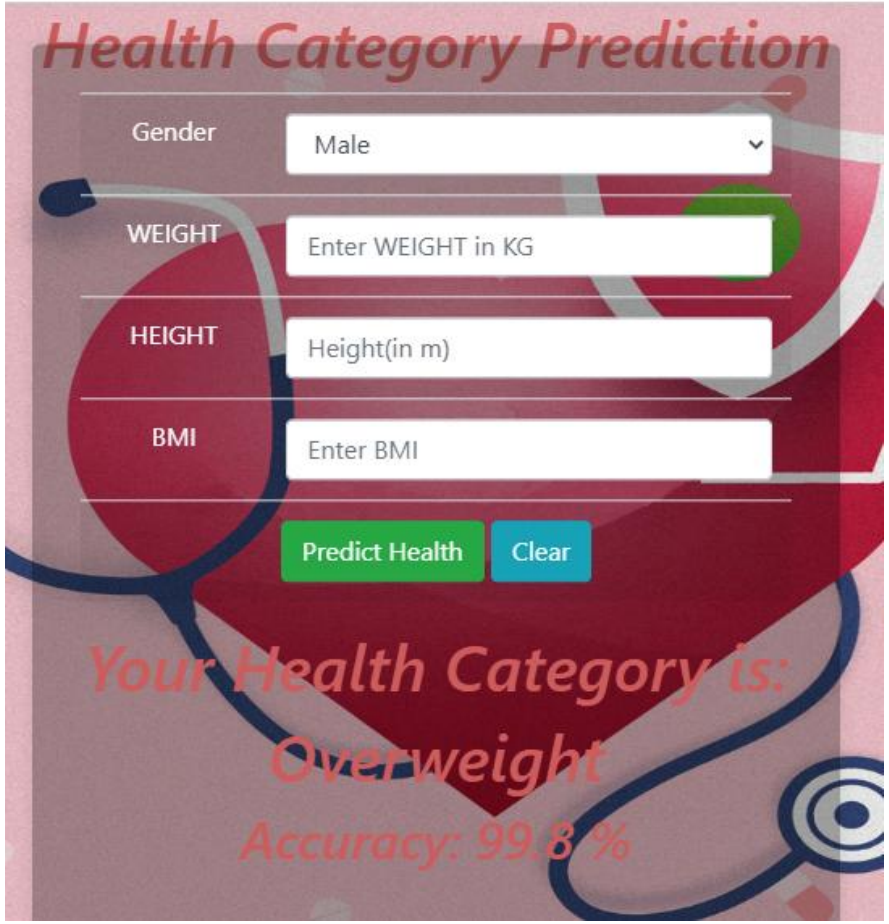
✓ Loan Approved

BMI Health Prediction Using Machine Learning

- This Flask-based web application predicts a person's health category based on their BMI (Body Mass Index), gender, weight, and height using a Random Forest Classifier. Users input their data, and the model predicts their health status while displaying the prediction accuracy. The system uses a pretrained dataset (BMI_Data.csv) to train the model dynamically every time a request is made.
- **Source Code**:

https://glitch.com/edit/#!/health-category-predict

- **Live View**: https://health-category-predict.glitch.me



Health Category Prediction

Gender: Male

WEIGHT: Enter WEIGHT in KG

HEIGHT: Height(in m)

BMI: Enter BMI

Your Health Category is: Overweight

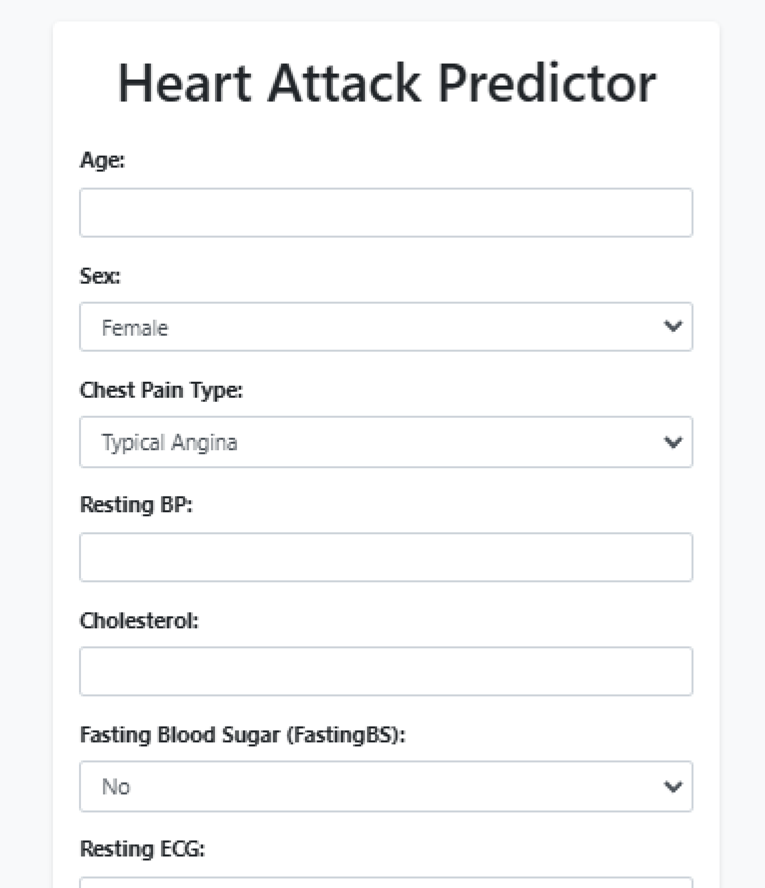
Accuracy: 99.8 %

Heart Attack Prediction Using Machine Learning

- This Flask-based web application predicts the likelihood of heart disease using a Random Forest Classifier trained on a dataset of heart health indicators. Users input key medical parameters such as age, cholesterol level, blood pressure, and chest pain type, and the model predicts whether they are at risk for heart Attack.
- **Source Code**:

https://glitch.com/edit/#!/heartattackpredict

- **Live View**: https://heartattackpredict.glitch.me



Heart Attack Predictor

Age:

Sex: Female

Chest Pain Type: Typical Angina

Resting BP:

Cholesterol:

Fasting Blood Sugar (FastingBS): No

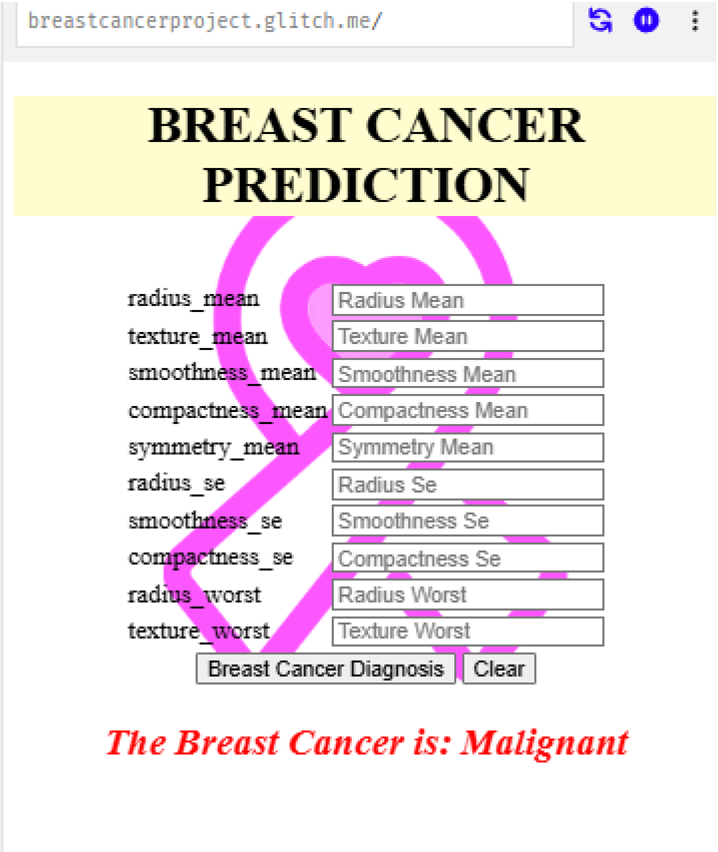
Resting ECG:

Breast Cancer Prediction System

- This Flask-based web application predicts whether a tumor is benign or malignant using Logistic Regression. It takes ten key tumor characteristics as input and classifies the tumor based on preprocessed breast cancer dataset (bdata.csv).
- **Source Code**:

https://glitch.com/edit/#!/breastcancerproject

- **Live View**: https://breastcancerproject.glitch.me/



breastcancerproject.glitch.me/

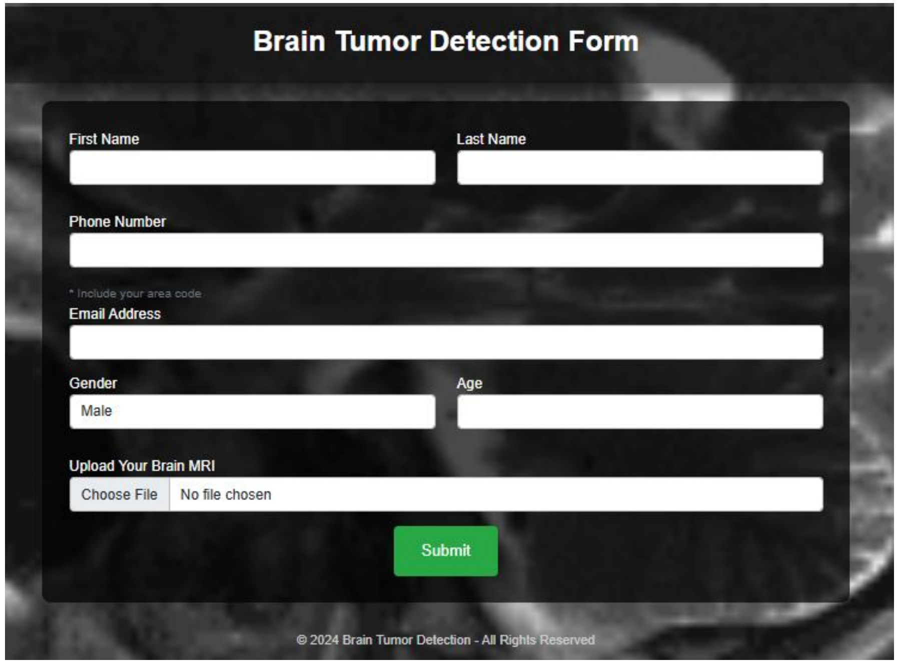
BREAST CANCER PREDICTION

radius_mean	Radius Mean
texture_mean	Texture Mean
smoothness_mean	Smoothness Mean
compactness_mean	Compactness Mean
symmetry_mean	Symmetry Mean
radius_se	Radius Se
smoothness_se	Smoothness Se
compactness_se	Compactness Se
radius_worst	Radius Worst
texture_worst	Texture Worst

The Breast Cancer is: Malignant

Brain Tumor Detection Using Deep Learning

- This **Flask web application** utilizes **Deep Learning (CNN)** to detect brain tumors from MRI images. Users can upload an MRI scan, and the system will analyze the image using a **pre-trained Convolutional Neural Network (CNN) model** to determine the presence of a brain tumor. The results are stored securely in **MongoDB Atlas**, allowing for patient record management and statistical analysis.
- **Dataset:** https://www.kaggle.com/datasets/princelv84/brain-tumor-dataset-yesno-class
- **Training Code:** https://colab.research.google.com/drive/1c7S07QIDgW4K73jo5AcxIaBMfcbvU2GL?usp=sharing
- **Source Code**: https://huggingface.co/spaces/LovnishVerma/braintumor/tree/main
- **Live View:** https://huggingface.co/spaces/LovnishVerma/braintumor



Brain Tumor Detection Form

First Name

Last Name

Phone Number

* Include your area code

Email Address

Gender: Male

Age

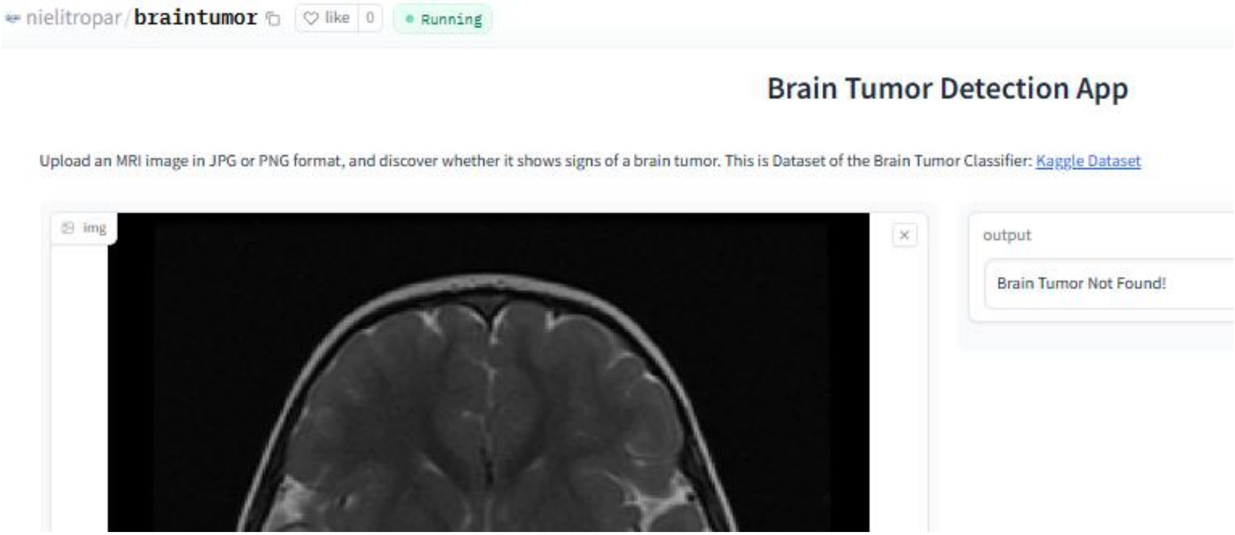
Upload Your Brain MRI

No file chosen

© 2024 Brain Tumor Detection - All Rights Reserved

Brain Tumor Detection Using Deep Learning

- This project is a Deep Learning-based Brain Tumor Detection App that utilizes **Streamlit** and **Gradio** to provide an interactive user experience. The system allows users to upload MRI images in JPG or PNG format and detects the presence of a brain tumor using a pre-trained CNN model (VGG16-based). The app processes images, performs predictions, and displays results with confidence scores.
- **Source Code** : https://huggingface.co/spaces/nielitropar/braintumor/tree/main
- **Live View : https://huggingface.co/spaces/nielitropar/braintumor**



nielitropar/braintumor

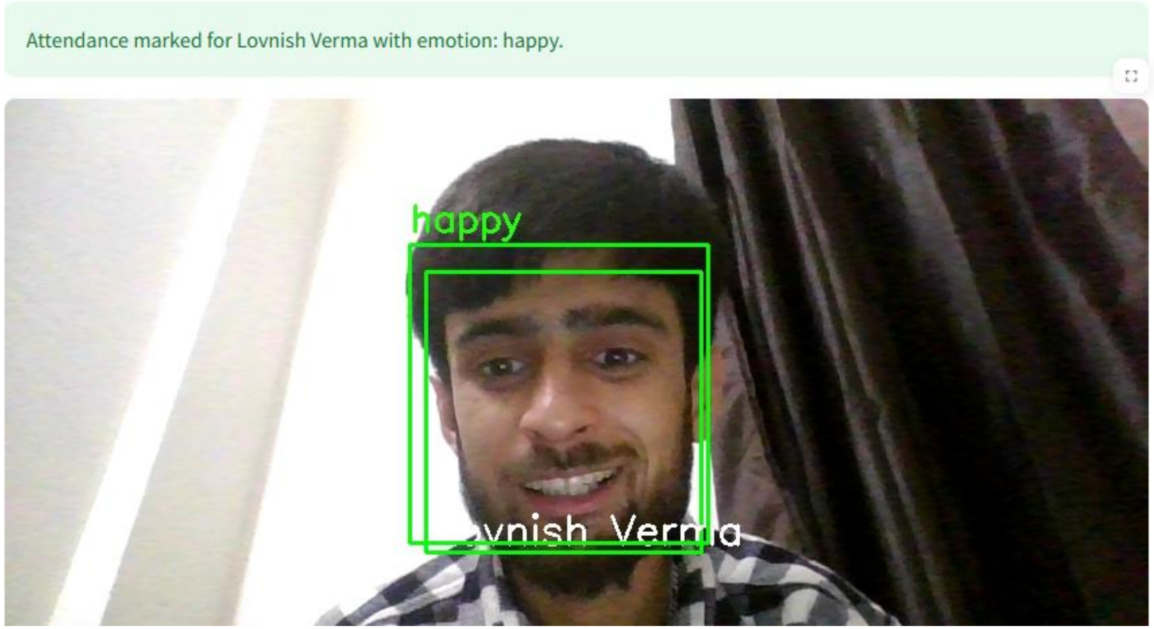
Brain Tumor Detection App

Upload an MRI image in JPG or PNG format, and discover whether it shows signs of a brain tumor. This is Dataset of the Brain Tumor Classifier: Kaggle Dataset

output: Brain Tumor Not Found!

Face and Emotion Recognition-Based Attendance System

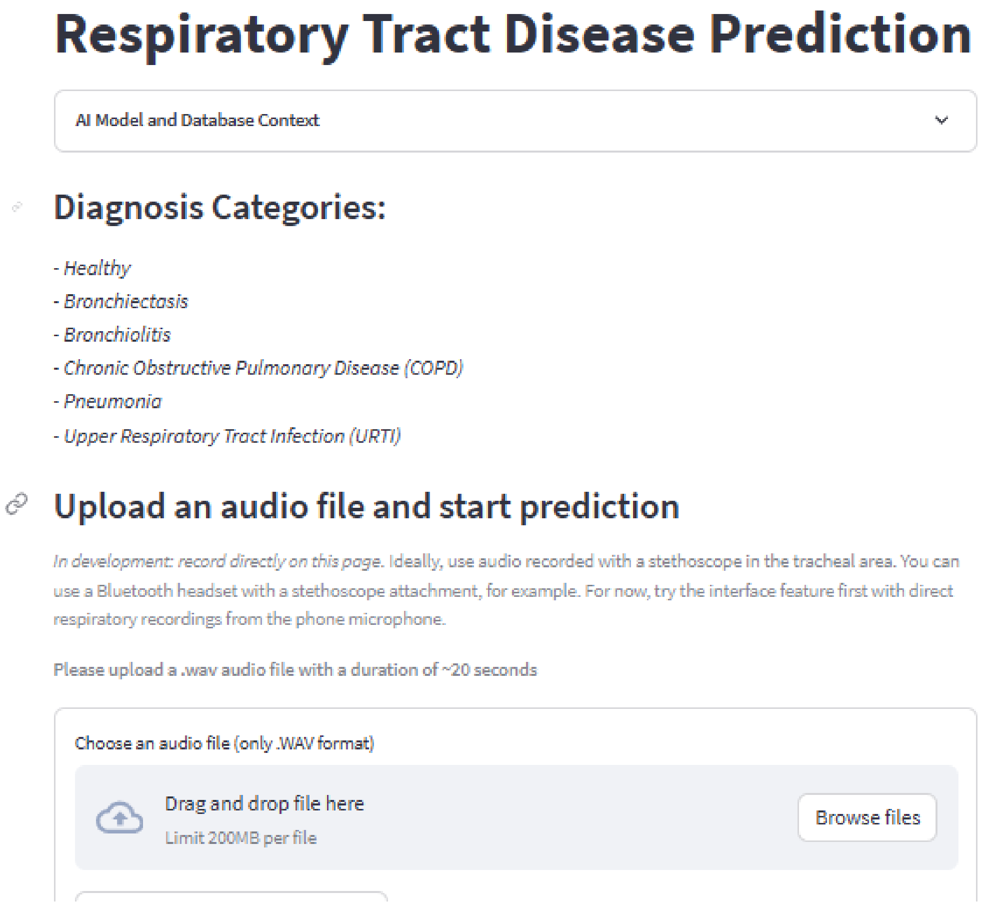
- This project is an AI-powered attendance system that utilizes face recognition and emotion detection to mark attendance automatically. Built using Streamlit, OpenCV, Face Recognition, and Deep Learning (CNN), the system captures real-time video input from a webcam, recognizes registered faces, and detects emotions. Attendance data, including name, roll number, date, time, and detected emotions, is stored in an SQLite database for easy retrieval. The system also allows users to register new faces and view attendance records in a structured format. This project is ideal for schools, colleges, offices, or any organization that wants to automate attendance while incorporating an innovative emotion-tracking feature.
- **Source Code**: https://github.com/lovnishverma/facial-sentiment-analysed-ai-attendance-tracker
- **Live View [Password is (nielit)]: https://faceemotionielit.streamlit.app/**



Attendance marked for Lovnish Verma with emotion: happy.

AI-Powered Respiratory Disease Prediction Using CNN & MFCCs

- This AI-driven Respiratory Disease Prediction App leverages Deep Learning to analyze respiratory sounds and detect potential lung diseases. Using a Convolutional Neural Network (CNN) trained on the Respiratory Sound Database, the app processes audio recordings (preferably from a stethoscope) and extracts Mel-Frequency Cepstral Coefficients (MFCCs) to identify patterns associated with various respiratory conditions.
- **Source Code**: https://huggingface.co/spaces/nielitropar/respiratory/tree/main
- **Live View : https://huggingface.co/spaces/nielitropar/respiratory**



Respiratory Tract Disease Prediction

AI Model and Database Context

Diagnosis Categories:

- Healthy
- Bronchiectasis
- Bronchiolitis
- Chronic Obstructive Pulmonary Disease (COPD)
- Pneumonia
- Upper Respiratory Tract Infection (URTI)

Upload an audio file and start prediction

In development: record directly on this page. Ideally, use audio recorded with a stethoscope in the tracheal area. You can use a Bluetooth headset with a stethoscope attachment, for example. For now, try the interface feature first with direct respiratory recordings from the phone microphone.

Please upload a .wav audio file with a duration of ~20 seconds

Choose an audio file (only .WAV format)

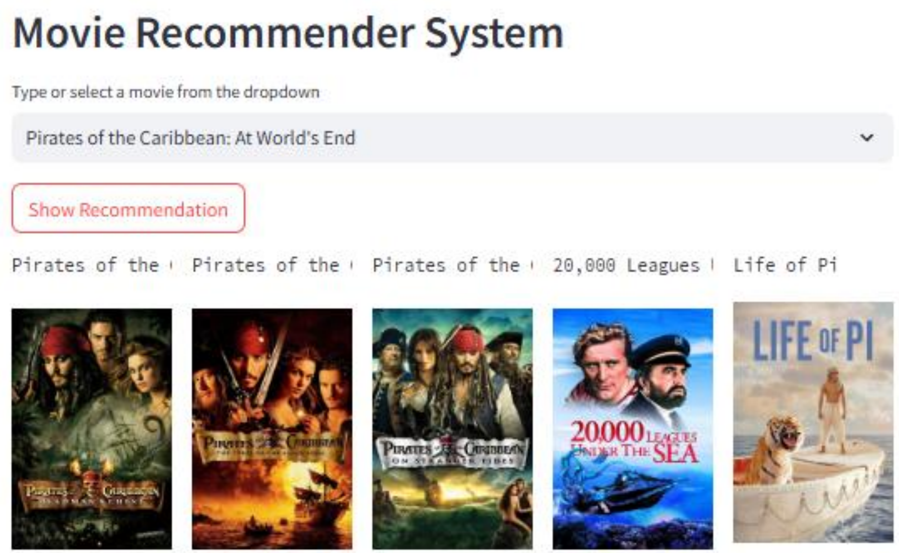
Drag and drop file here
Limit 200MB per file

Movie Recommender System

- This is a Movie Recommender System built using Streamlit, TMDb API, and machine learning techniques. The system suggests five similar movies based on user-selected input by analyzing movie similarity scores.

- **Key Features**:

1. Interactive UI: Users can type or select a movie from a dropdown menu.
2. Movie Recommendations: Uses a pre-trained similarity model to recommend five similar movies.
3. Poster Fetching: Retrieves movie posters dynamically from the TMDb API.
4. Machine Learning Model: Uses a precomputed similarity matrix (from similarity.pkl) to find and rank similar movies.
5. Efficient Storage: Loads preprocessed movie data (movie_names.pkl) to provide quick recommendations.



Movie Recommender System

Type or select a movie from the dropdown

Pirates of the Caribbean: At World's End

Source Code: https://huggingface.co/spaces/LovnishVerma/movie_recommendation_system/tree/main

Live View : [**https://huggingface.co/spaces/LovnishVerma/movie_recommendation_system**](https://huggingface.co/spaces/LovnishVerma/movie_recommendation_system)