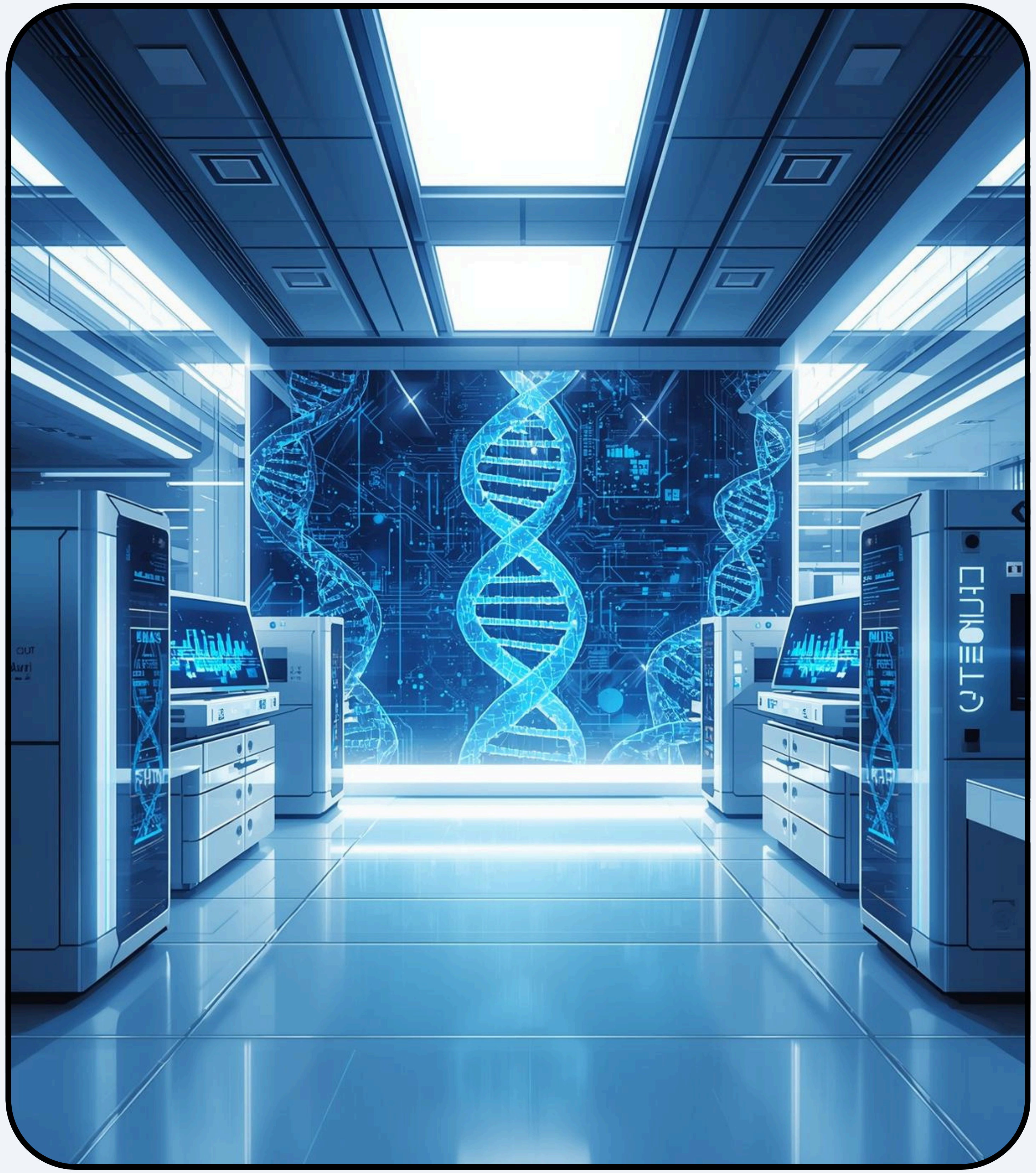
Personalized Medicine

PREDICTIVE TREATMENT RESPONSE

Group 9:- Shristi Mishra (22000793),
Janvi Patel (22000877),
Roma Rajbhar (22000921)



Clinical Challenge

The manual classification of genetic mutations in cancer patients is a time-consuming and complex process. It requires extensive expertise and is hindered by the vast volume of literature, making it difficult for clinicians to keep up with emerging information and evolving treatment protocols.

Automated mutation classification using AI can significantly reduce the workload on healthcare professionals, enhancing their ability to provide timely and effective personalized treatment options for patients.

Overview of the MSKCC Dataset

Comprehensive Data Collection

The MSKCC Personalized Medicine dataset includes a diverse range of **genetic mutation** data essential for understanding various cancers and their treatment options.

Multi-Modal Data Types

This dataset comprises gene names, mutation variations, and clinical evidence text, allowing for a **complete analysis** of mutations alongside textual clinical information.

Extensive Sample Size

With over 3,000 samples and **nine labeled classes**, the dataset provides a robust foundation for training deep learning models to classify genetic mutations effectively.

System Architecture for Multi-Modal Deep Learning

Integrated Approach

The architecture combines convolutional neural networks and dense layers, leveraging both genomic and textual data for enhanced predictive accuracy.

Feature Fusion

By fusing features from different modalities, the model captures intricate relationships, improving the classification of genetic mutations significantly.

Scalability and Flexibility

This architecture allows for easy integration with other data types and models, facilitating future enhancements and scalability in clinical applications.

Detailed Breakdown of Model Architecture

Layer Composition Overview

The architecture utilizes **embedding layers** followed by Conv1D layers to capture features from text inputs effectively.

Fusion of Modalities

Tabular data is integrated through **dense layers**, allowing the model to learn from both genomic and clinical information.

Performance Optimization Strategies

Techniques like regularization and dropout are employed to enhance the model's **generalization capabilities** and prevent overfitting during training.

Technical Implementation Stack Overview and Tools

Core Technologies Used

The implementation relies on **PyTorch** for deep learning, facilitating model development and experimentation with flexibility in architecture design.

Web Frameworks and APIs

FastAPI is employed to develop the backend services, ensuring efficient API management and seamless interaction with the frontend interface.

Version Control and Collaboration

GitHub is used for version control, promoting collaborative coding practices, enabling easy tracking of changes, and facilitating team contributions.

Data Preprocessing Pipeline for Model Training

Importance of Data Quality

High-quality data ensures the model learns accurately. Proper **cleaning and formatting** of input can significantly impact performance and reliability.

Steps in Data Preparation

The preprocessing involves **tokenization**, padding, and encoding, which prepares text and genetic data for the model, ensuring compatibility and effectiveness.

Train-Test Data Split

Splitting the data into training and testing sets is crucial for **evaluating model performance**. This separation helps avoid overfitting and validates generalization.

Training Strategy for Multi-Modal Model

Optimizing Learning Process

The **training strategy** utilizes the Adam optimizer with a learning rate of 0.001 to enhance convergence during model training.

Epochs and Stopping Criteria

The model is trained over **15 epochs** with early stopping and checkpoints to prevent overfitting and ensure robust performance evaluation.

Batch Size Considerations

A batch size of **32** is implemented, balancing memory efficiency with the ability to capture sufficient training data for each iteration.

Evaluating Model Performance Metrics and Results

Key Performance Indicators

The model's performance is assessed through various metrics, including Macro F1, ROC-AUC, and accuracy, providing a comprehensive evaluation.

Class-Specific Analysis

Per-class metrics help identify strengths and weaknesses in classification, allowing targeted improvements and highlighting areas needing further research.

Confusion Matrix Insights

The confusion matrix visualizes classification results, revealing misclassifications and guiding adjustments in the model for enhanced accuracy and reliability.

Key Features of the Web Application

User-Friendly Interface

The application offers an intuitive interface, allowing users to easily input genetic data and receive predictions without prior technical knowledge.

Real-Time Predictions

Users benefit from instantaneous predictions, enabling quick decision-making in clinical settings based on the latest genomic analysis.

Confidence Score Visualization

The application visualizes confidence scores alongside predictions, enhancing interpretability and helping clinicians understand the reliability of the results.

Comprehensive End-to-End Workflow Overview

Input Data Handling

The **workflow starts** with input data from clinical reports and genomic sequences, ensuring all necessary information is captured for processing.

Preprocessing Steps

Data undergoes essential preprocessing, including **text cleaning**, tokenization, and encoding of gene variations, which prepares it for the multi-modal model.

Model Integration

The processed data integrates with the deep learning model, where predictions are generated and probabilities are calculated, **facilitating clinical insights**.

Example Prediction of Genetic Mutation

Overview of Prediction Process

The model predicts mutation classifications based on input features, combining genomic data and clinical text to enhance accuracy and reliability.

Real-World Application

For instance, the gene TP53 with the variation R273C can be classified as pathogenic, showing the model's potential in clinical settings.

Confidence Scores Explained

The confidence score of approximately 27% indicates the model's certainty in its prediction, guiding clinicians in decision-making processes.

Addressing Challenges in Mutation Classification

Model Size Complexity

The large **size of our model** poses challenges in computational efficiency and memory usage, requiring careful resource management during training.

Class Imbalance Issues

Class imbalance among mutation types can lead to biased predictions; strategies such as **oversampling and synthetic data generation** are essential to address this.

Handling Long Texts

Processing lengthy clinical texts is critical; implementing techniques such as **text truncation and summarization** can improve performance without losing essential information.

Key Achievements in Personalized Medicine

Comprehensive AI System

Developed a **robust end-to-end** machine learning system that integrates multiple data types for more accurate mutation classification.

Multi-Modal Fusion Approach

Successfully implemented a **multi-modal fusion** model that combines genomic data and clinical text, enhancing classification precision and reliability.

User-Friendly Web Interface

Created an intuitive web application that allows clinicians to easily input data and receive real-time predictions, streamlining the decision-making process.

Future Enhancements for Personalized Medicine

Incorporating Advanced Algorithms

Future iterations will explore **transformer-based architectures** to improve accuracy and interpretability in genetic mutation classification tasks.

Enhancing User Interface

We aim to develop a more **intuitive user experience**, integrating interactive visualizations for better understanding of predictions and clinical implications.

Expanding Clinical Integration

Plans include seamlessly **integrating our solution** into existing clinical workflows, facilitating easier access for healthcare professionals in real-time scenarios.

Deployment Architecture of the System

Environment Configuration

The deployment architecture utilizes a **cloud-based environment** for scalability, ensuring the application can handle varying loads efficiently and reliably.

Technology Stack

Implementation is based on **Docker containers** for easy deployment, alongside services like AWS for hosting and storage, ensuring accessibility and performance.

Load Balancing Strategy

A load balancing strategy is implemented to **distribute traffic** evenly across multiple instances, enhancing user experience and maintaining system responsiveness during peak usage.

Repository Structure: Organizing Code for Efficiency

Clear Code Organization

The **repository is structured** into distinct directories, enhancing navigation and maintainability for developers and collaborators.

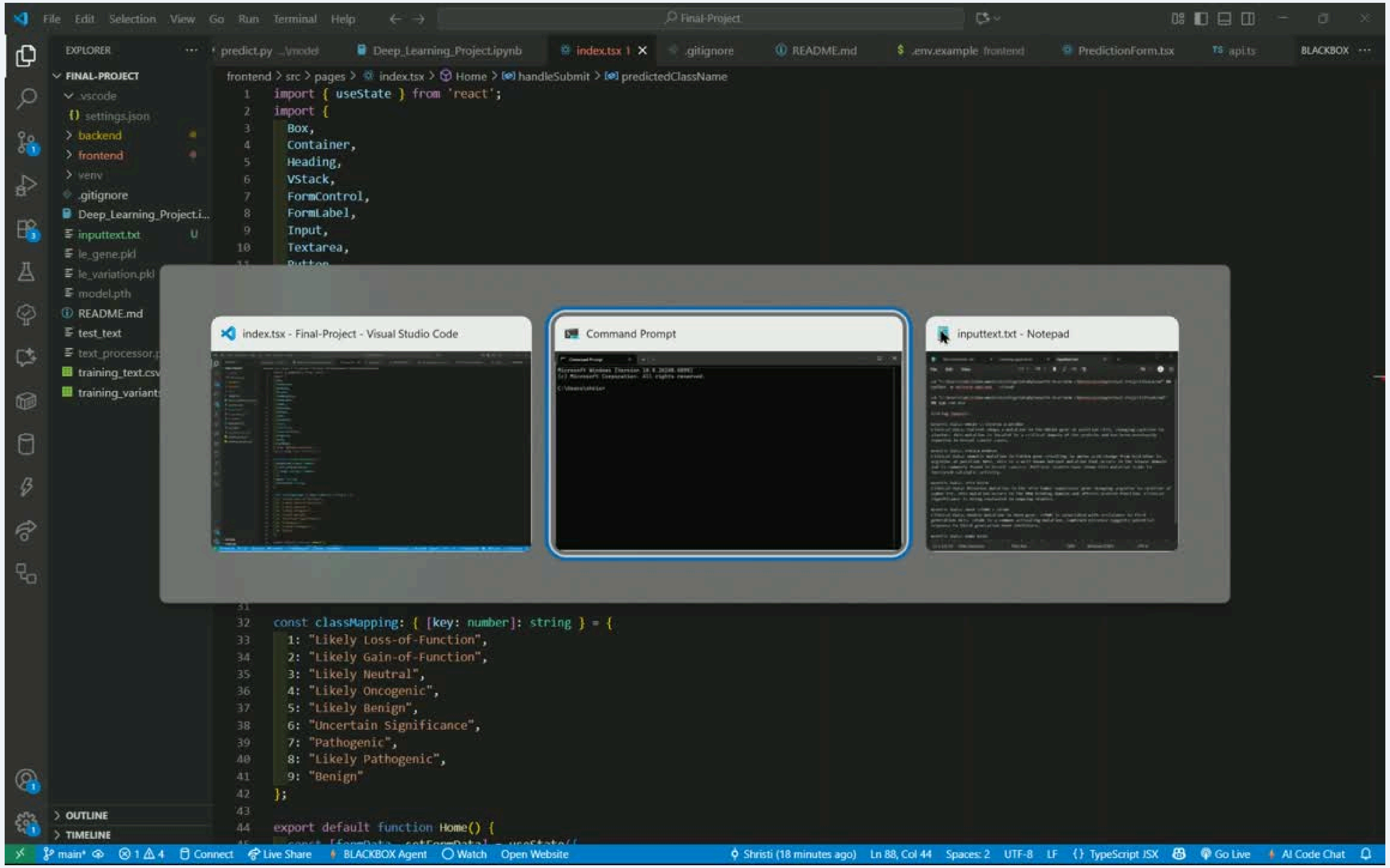
Modular Design Approach

Each module is self-contained, focusing on specific functionalities, which simplifies debugging and updates while improving code readability and efficiency.

Documentation and Resources

Comprehensive documentation accompanies the repository, providing essential guidance for users and developers to understand the setup and usage of the system.

Demo Screenshots of the Web Application



frontend > src > pages > index.tsx > Home > handleSubmit > predictedClassName

```
import { useState } from 'react';
import {
  Box,
  Container,
  Heading,
  VStack,
  FormControl,
  FormLabel,
  Input,
  Textarea,
  Button
```

```
const classMapping: { [key: number]: string } = {
  1: "Likely Loss-of-Function",
  2: "Likely Gain-of-Function",
  3: "Likely Neutral",
  4: "Likely Oncogenic",
  5: "Likely Benign",
  6: "Uncertain Significance",
  7: "Pathogenic",
  8: "Likely Pathogenic",
  9: "Benign"
};

export default function Home() {
```

Loom Demo Video Link:- https://www.loom.com/share/ba96348a95194bbf8701b00f1165bfee

Github Repository Link:- https://github.com/ShristiDCJ/personalized-medicine-ml.git

(Some files were to large to upload, so they are not present in github)

Key Lessons Learned from the Project

Importance of Multi-Modal Data

Utilizing both genomic and clinical text data significantly enhances the model's capability to classify mutations more accurately and efficiently.

Need for Robust Preprocessing

Comprehensive data cleaning and preprocessing are essential to ensure the model's performance and to reduce noise in the training dataset.

Implementation Challenges Faced

Navigating deployment complexities and ensuring model interpretability were critical challenges that required innovative engineering solutions throughout the project.

Impact of Personalized Medicine on Oncology

Improved Patient Outcomes

Personalized medicine **enhances treatment** precision, allowing clinicians to tailor therapies based on individual genetic mutations, ultimately improving patient outcomes and survival rates.

Automation in Decision-Making

The integration of AI facilitates **automated clinical** decision support, reducing the workload on healthcare professionals and enabling them to focus on critical patient care.

Future Research Directions

Continued exploration of multi-modal deep learning approaches will **further advance** personalized medicine, unlocking new opportunities for innovative treatments in oncology and beyond.

Thank You