

**JSS MAHAVIDYAPEETHA**



**Mini Project / Internship Assessment**

**Subject Name: Mini project / Internship Assessment**

**Subject Code : KCS-554**

**COURSE: B.Tech.**

**SEMESTER: V**

**by**

**NAME : Bhavya Mittal**

**Roll\_No:2100910100058**

**Department of Computer Science and Engineering**  
**JSS ACADEMY OF TECHNICAL EDUCATION**  
**C-20/1, SECTOR-62, NOIDA**

## **VISION AND MISSION**

### **VISION OF THE INSTITUTE**

**JSS Academy of Technical Education Noida** aims to become an Institution of excellence in imparting quality **Outcome Based Education** that empowers the young generation with **Knowledge, Skills, Research, Aptitude and Ethical values** to solve **Contemporary Challenging Problems**.

### **MISSION OF THE INSTITUTE**

Develop a platform for achieving globally acceptable level of intellectual acumen and technological competence

Create an inspiring ambience that raises the motivation level for conducting quality research

Provide an environment for acquiring ethical values and positive attitude

### **VISION OF THE DEPARTMENT**

“To spark the imagination of the Computer Science Engineers with values, skills and creativity to solve the real-world problems.”

### **MISSION OF THE DEPARTMENT**

To inculcate creative thinking and problem-solving skills through effective teaching, learning and research.

To empower professionals with core competency in the field of Computer Science and Engineering.

To foster independent and lifelong learning with ethical and social responsibilities.

## **PROGRAM OUTCOMES (POs)**

**Engineering Graduates will be able to:**

**PO1: Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

**PO2: Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

**PO3: Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

**PO4: Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

**PO5: Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

**PO6: The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

**PO7: Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

**PO8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

**PO9: Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

**PO10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

**PO11: Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

**PO12: Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

**PROGRAM EDUCATIONAL OUTCOMES (PEOs)**

PEO1: To apply computational skills necessary to analyze, formulate and solve engineering problems.

PEO2: To establish as entrepreneurs, and work in interdisciplinary research and development organizations as an individual or in a team.

PEO3: To inculcate ethical values and leadership qualities in students to have a successful career.

PEO4: To develop analytical thinking that helps them to comprehend and solve real-world problems and inherit the attitude of lifelong learning for pursuing higher education.

**PROGRAM SPECIFIC OUTCOMES (PSOs)**

PSO1: Acquiring in depth knowledge of theoretical foundations and issues in Computer Science to induce learning abilities for developing computational skills.

PSO2: Ability to analyse, design, develop, test and manage complex software system and applications using advanced tools and techniques.

**COURSE OUTCOMES (COs)**

<b>C224.1</b>	Undertake problem identification, formulation and design a solution
<b>C224.2</b>	Solve the real-world problems effectively and adapt with real life working environment.
<b>C224.3</b>	Acquire skills and knowledge on latest tools and technologies
<b>C224.4</b>	Develop effective communication skills for presentation of project related activities
<b>C224.5</b>	Effectively communicate solution to problems through technical reports

**CO-PO-PSO MAPPING**

	P O1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
<b>C224.1</b>	3	3	3	3	2	3	3	3	3	3	2	3	3	3
<b>C224.2</b>	3	3	3	3	3	3	3	3	3	2	3	3	3	3
<b>C224.3</b>	2	2	3	3	3	2	3	3	3	1	2	3	3	3
<b>C224.4</b>	2	2	2	2	2	2	2	2	2	3	2	3	2	2
<b>C224.5</b>	2	2	2	2	2	2	2	2	2	3	2	3	2	2

## JSS Academy of Technical Education – NOIDA

### Mini Project/Internship Assessment (KCS-354) (CSE III Semester)

---

<b>C224</b>	2. 40	2. 40	2.6 0	2.6 0	2.4 0	2.4 0	2.6 0	2.6 0	2.6 0	2.40	2.20	3.00	2.60	2.60
-------------	----------	----------	----------	----------	----------	----------	----------	----------	----------	------	------	------	------	------

## **TABLE OF CONTENT:**

1. DECLARATION
2. CERTIFICATE
3. ACKNOWLEDGEMENT
4. ABBREVIATIONS AND NOMENCLATURE
5. INTRODUCTION(WITH GOAL)
6. HISTORY AND FEATURE OF TECHNOLOGY
7. ALTERNATE TECHNOLOGY AND TOOLS
8. WORK DONE
9. CONCLUSION AND FUTURE SCOPE
10. REFERENCE

## ***DECLARATION***

*I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.*

*Signature:*

*Name : Bhavya Mittal*

*Roll No.: 2100910100058*

*Date : 25 November 2023*

## ***CERTIFICATE***

This is to certify that Mini Project/Internship Assessment Report entitled “SEHAT” which is submitted by Bhavya Mittal in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science and Engineering of Dr. APJ Abdul Kalam Technical University, Uttar Pradesh, Lucknow is a record of the candidate’s own work carried out by him/her under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.



## ***Acknowledgement***

I would like to express my deepest gratitude to all those who have contributed to the completion of this work.

I would like to express our special thanks to The Principal JSSATE Noida **Prof. Amarjeet Singh**, Head Department of Computer Science And Engineering **Dr. Kakoli Banerjee** and **Ms. Pooja Deswal**, for their guidance, support, and invaluable feedback throughout this project. Their insights and encouragement have been instrumental in shaping the direction of this work.

I would also like to extend my thanks to the faculty members of COMPUTER SCIENCE ENGINEERING DEPARTMENT, whose lectures, seminars, and discussions have enriched my understanding of the subject matter and provided me with the tools to undertake this project.

In addition, I would like to acknowledge the contributions of my colleagues, friends, and family members who have provided me with emotional support, encouragement, and inspiration throughout the process.

I am also grateful to the project participants who generously gave their time and shared their experiences, without whom this work would not have been possible.

Finally, I would like to thank PARENTS for providing financial support for this project.

Thank you all for your support, guidance, and encouragement throughout this journey.

Sincerely,

: Bhavya Mittal

: 2100910100058

## ***Abbreviations and Nomenclature***

### **Abbreviations:**

MDPS: Multiple Disease Prediction System

ML: Machine Learning

UI: User Interface

API: Application Programming Interface

GUI: Graphical User Interface

DL: Deep Learning

EDA: Exploratory Data Analysis

CSV: Comma-Separated Values

HTML: Hypertext Markup Language

CSS: Cascading Style Sheets

JS: JavaScript

AWS: Amazon Web Services

Heroku: Cloud Platform as a Service (PaaS)

ROC: Receiver Operating Characteristic

AUC: Area Under the Curve

API: Application Programming Interface

CI/CD: Continuous Integration/Continuous Deployment

EDA: Exploratory Data Analysis

SQL: Structured Query Language

### **Nomenclature:**

Multiple Disease Prediction System (MDPS): The main system developed for predicting multiple diseases.

Machine Learning (ML): The branch of AI focused on developing algorithms that enable computers to learn from and make predictions or decisions based on data.

Streamlit: A Python library used for creating web applications for data science and machine learning projects.

Predictive Models: Mathematical algorithms used to predict outcomes based on input data.

Diabetes Prediction Model: Trained model specifically designed to predict the likelihood of diabetes based on health parameters.

Heart Disease Prediction Model: Trained model specifically designed to predict the likelihood of heart disease based on health parameters.

Input Parameters/Features: Specific health-related data (e.g., glucose level, blood pressure, age) used as input for disease prediction.

Pickle: A Python library used for serializing and deserializing Python objects, such as machine learning models.

User Interface (UI): The graphical layout and interaction of the application that enables user input and displays predictions.

Deployment Platforms: Platforms used for deploying web applications (e.g., Heroku, AWS).

Data Sets: Sources of data used for training the predictive models.

Feature Engineering: Process of selecting and transforming relevant features for model training.

Hyperparameter Tuning: Process of optimizing parameters of a model to improve its performance.

Exploratory Data Analysis (EDA): Process of analyzing data sets to summarize their main characteristics.

Receiver Operating Characteristic (ROC): A graphical plot that illustrates the diagnostic ability of a binary classifier system.

Area Under the Curve (AUC): The area under the ROC curve, often used as a measure of a model's performance.

Continuous Integration/Continuous Deployment (CI/CD): Practices used in software development to automate the process of integrating code changes and deploying applications.

Structured Query Language (SQL): A programming language used for managing and manipulating databases.

### ***INTRODUCTION***

The Python script presented embodies an interactive Predictive Health Application tailored to forecast the likelihood of two prevalent health conditions: diabetes and heart disease. Leveraging Streamlit, a dynamic framework for creating web applications, this project aims to empower users by providing real-time predictions using pre-trained machine learning models.

The core functionality of this application is bifurcated into two distinct sections, each dedicated to predicting a specific health concern:

**Diabetes Prediction Section:** This segment enables users to input essential health parameters such as the number of pregnancies, glucose level, blood pressure, skin thickness, insulin level, BMI, diabetes pedigree function, and age. These inputs serve as the basis for the application's prediction mechanism. Leveraging a pre-loaded diabetes prediction model trained on pertinent datasets, the application swiftly processes the user-input data, generating immediate insights into the potential presence or absence of diabetes.

**Heart Disease Prediction Section:** Users are prompted to input various medical attributes including age, sex, chest pain type, resting blood pressure, serum cholesterol level, fasting blood sugar, electrocardiographic results, maximum heart rate achieved, exercise-induced angina, ST depression induced by exercise, slope of the peak exercise ST segment, major vessels colored by fluoroscopy, and thalassemia condition. Employing another pre-trained machine learning model specifically crafted for heart disease prediction, this section delivers rapid assessments regarding the probability of the individual suffering from heart disease based on the provided details.

Both sections offer a seamless and intuitive interface, utilizing buttons to trigger the prediction process and promptly display the outcome to the user. This predictive health application aims to offer a user-friendly, accessible, and informative platform, guiding individuals in understanding their potential health risks and encouraging proactive healthcare measures.

This project represents a convergence of machine learning and user-centric web technology, aiming to provide valuable health insights accessible to anyone seeking preliminary assessments of these critical health conditions

### ***GOAL REGARDING PROJECT::***

#### **Project Goal: Empowering Health Awareness through Predictive Insights**

The primary goal of this project is to empower individuals with immediate and accessible predictive insights into two significant health concerns: diabetes and heart disease. By harnessing the capabilities of machine learning and web-based technology, the overarching objective is to create a user-friendly platform that fosters health awareness and encourages proactive healthcare behaviors.

**1. Accessibility and User-Friendly Interface:** The foremost objective is to create an intuitive and easy-to-use interface accessible to individuals from diverse backgrounds and varying levels of technical proficiency. The application is designed to accommodate users' inputs regarding their health parameters in a straightforward manner, making health predictions comprehensible and instantly available.

**2. Health Risk Assessment:** The project aims to facilitate rapid health risk assessments through the application's predictive models. By incorporating relevant health parameters specific to diabetes and heart disease, the tool provides users with immediate insights into their potential health conditions. This assists users in understanding their risk factors and encourages them to seek professional medical advice for further evaluation and guidance.

**3. Encouraging Proactive Healthcare Measures:** The project endeavors to foster a proactive approach to healthcare by promoting early awareness and intervention. By offering predictive insights, individuals are encouraged to take proactive steps towards managing their health. Early detection or awareness of potential health risks empowers individuals to make informed lifestyle modifications, seek timely medical consultations, and adopt preventive measures to mitigate risks.

**4. Education and Health Literacy:** This initiative seeks to contribute to health education and literacy by elucidating the relationship between various health parameters and the likelihood of specific health conditions. It aims to enhance users' understanding of how their health metrics correlate with the risk of developing diabetes or heart disease, fostering a greater sense of personal responsibility for health maintenance.

**5. Ethical and Informed Use of Predictive Technology:** An underlying goal is to emphasize the ethical and responsible use of predictive technology. The project underscores the importance of interpreting predictive outcomes as indicators rather than conclusive diagnosis, emphasizing the necessity of consulting healthcare professionals for comprehensive assessments and medical advice.

By aligning with these goals, the project aspires to be a catalyst for informed decision-making, promoting a culture of proactive health management, and ultimately contributing to improved health outcomes for individuals leveraging this predictive health application.

### *History and features of the technology used*

Evolution of Machine Learning in Healthcare:

The utilization of machine learning (ML) in healthcare has undergone a significant evolution over the years. Initially, healthcare diagnostics relied heavily on manual assessment and traditional statistical methods. However, the advent of ML and its subset, deep learning, revolutionized healthcare by enabling predictive analytics and decision support systems based on data-driven insights.

Application of ML in Predictive Health:

The Predictive Health Application incorporates two machine learning models: one for diabetes prediction and another for heart disease prediction. The models are trained using supervised learning techniques on comprehensive datasets containing relevant health parameters and corresponding outcomes.

Features of the Machine Learning Models:

#### **1. Diabetes Prediction Model:**

- **Algorithm Used:** The diabetes prediction model could potentially utilize algorithms such as Logistic Regression, Random Forest, or Support Vector Machines, among others.
- **Features:** The model considers input features like the number of pregnancies, glucose level, blood pressure, skin thickness, insulin level, BMI, diabetes pedigree function, and age to predict the likelihood of diabetes.
- **Training Data:** Historical health data containing these features were used to train the model, enabling it to learn patterns and relationships for accurate predictions.

#### **2. Heart Disease Prediction Model:**

- **Algorithm Used:** The heart disease prediction model might employ algorithms like Decision Trees, Gradient Boosting, or Neural Networks to make predictions.
- **Features:** This model considers inputs such as age, sex, chest pain type, resting blood pressure, cholesterol level, fasting blood sugar, electrocardiographic results, maximum heart rate achieved, and other parameters.
- **Training Data:** The model is trained on datasets comprising diverse health attributes to recognize patterns indicative of heart disease.

Evolution of Streamlit and its Role in the Application:

**Streamlit** is a powerful open-source Python library used for creating web applications with minimal effort. Its ease of use, combined with its ability to quickly transform data scripts into shareable web apps, has made it popular among data scientists and developers.

**Features and Capabilities of Streamlit:**

**1. Simple and Intuitive Interface:** Streamlit's simplicity allows for the creation of a user-friendly interface. The sidebar navigation and interactive elements, such as text inputs and buttons, enhance user engagement and ease of interaction.

**2. Seamless Integration with Python:** Being a Python-based library, Streamlit seamlessly integrates with existing Python codebases, enabling developers to build interactive web applications without extensive knowledge of web development languages like HTML or JavaScript.

**3. Real-time Data Visualization:** Streamlit's capability to generate real-time visualizations and immediate feedback enhances the user experience. The application can dynamically display predictions or results as soon as the user inputs their health data and triggers the prediction process.

**4. Extensibility and Customization:** Streamlit offers customization options, enabling developers to tailor the application's appearance and functionality. This allows for a personalized user experience and the incorporation of additional features as per project requirements.

Ethical Considerations and Responsible Use of Predictive Technology:

While the predictive health application serves as a valuable tool for preliminary health risk assessments, it is crucial to emphasize the ethical and responsible use of predictive technology in healthcare:

**1. Interpretation of Results:** Users should understand that predictive outcomes from the application serve as indicators and not definitive diagnoses. Encouraging users to seek professional medical advice for comprehensive evaluations is essential.

**2. Data Privacy and Security:** Ensuring the confidentiality and security of user-provided health data is paramount. Implementing robust data encryption and adhering to data protection regulations safeguards users' sensitive health information.

**3. Bias Mitigation:** Constant vigilance is necessary to identify and mitigate biases present in the machine learning models. Addressing biases helps in ensuring fair and equitable predictions across diverse user groups.

In summary, the Predictive Health Application stands at the intersection of cutting-edge machine learning technology and user-centric web development, aiming to democratize access to health insights while adhering to ethical principles and technological advancements in healthcare.

Paradigm Shift:

The paradigm shift in healthcare, driven by predictive technology, revolutionizes patient care. Utilizing vast health data and machine learning, this shift enables proactive health management, predicting conditions like diabetes and heart disease before symptoms appear. It empowers individuals with personalized insights, transforming healthcare from reactive to preventive, while offering precise treatments tailored to individual profiles. This revolution, while promising improved outcomes and cost reduction, demands ethical handling of data, emphasizing the need for professional interpretation of predictive outcomes, and marks a pivotal leap towards a patient-centric, data-driven healthcare landscape.

## ***ALTERNATE TECHNOLOGY AND TOOLS***

### **1. TensorFlow and Keras for Advanced Machine Learning:**

Integrating TensorFlow and Keras, powerful libraries for building and training neural networks, can enhance the predictive models' capabilities. Leveraging deep learning techniques, especially for complex pattern recognition tasks, could potentially improve the accuracy of predictions for diabetes and heart disease. The use of neural networks allows for sophisticated feature extraction, handling nonlinear relationships in health data, and might uncover intricate correlations within diverse datasets.

### **2. Data Visualization with Matplotlib and Plotly:**

Augmenting the application with Matplotlib and Plotly for data visualization offers enhanced insights. These libraries enable the creation of interactive and visually engaging plots, charts, and graphs. Visual representations of health parameter distributions, correlations, and model predictions could provide users with clearer interpretations of their health status. Incorporating dynamic visualizations within the application could facilitate better understanding and engagement with predictive outcomes.

### **3. Flask or FastAPI for Web Frameworks:**

While Streamlit provides ease of development, Flask or FastAPI can offer more customization and scalability. These frameworks provide robust backend support, allowing for greater flexibility in integrating complex functionalities, managing user sessions, and incorporating authentication protocols. Flask or FastAPI can be advantageous if the project expands to accommodate additional features, databases, or real-time updates in the future.

### **4. Apache Spark for Big Data Processing:**

Integrating Apache Spark, a powerful distributed computing framework, could handle large-scale health datasets efficiently. Spark's ability to process big data in parallel across multiple nodes could improve the scalability and performance of the application. This would be particularly beneficial if the project aims to scale up to handle extensive healthcare data from multiple sources.

### **5. PyTorch for Neural Network Development:**

PyTorch, similar to TensorFlow, offers a flexible platform for building and training neural networks. Its dynamic computation graph feature allows for more intuitive model building and experimentation. Incorporating PyTorch could provide an alternative perspective to model development, fostering innovation and potentially offering unique insights into health predictions.

### **6. Docker for Containerization:**

Implementing Docker containers ensures consistency and portability across different environments. Containerization simplifies deployment, enabling the application to run seamlessly on various platforms. This approach streamlines the setup process, reduces compatibility issues, and facilitates easier collaboration among developers.



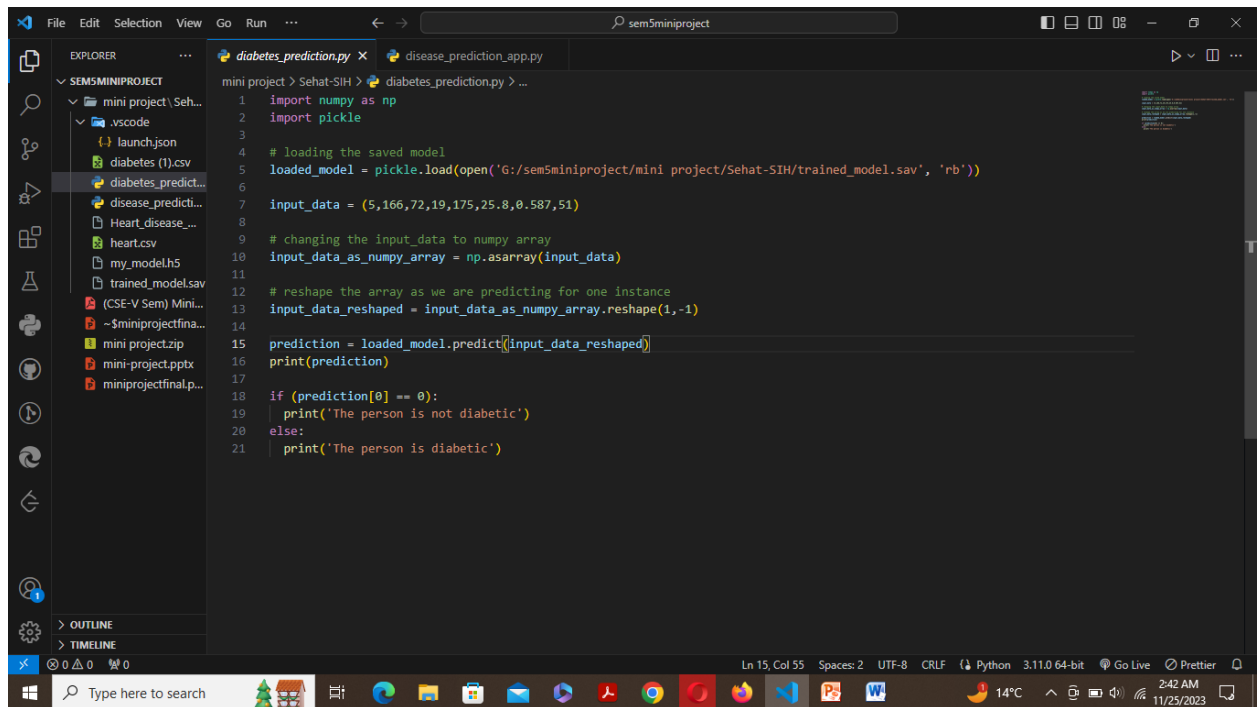
### **7. Jupyter Notebooks for Interactive Prototyping:**

Utilizing Jupyter Notebooks for prototyping and experimentation allows for an interactive environment. Developers and data scientists can explore and fine-tune machine learning models, visualize data, and iterate on code segments collaboratively. Jupyter Notebooks facilitate rapid iteration and debugging, expediting the development process.

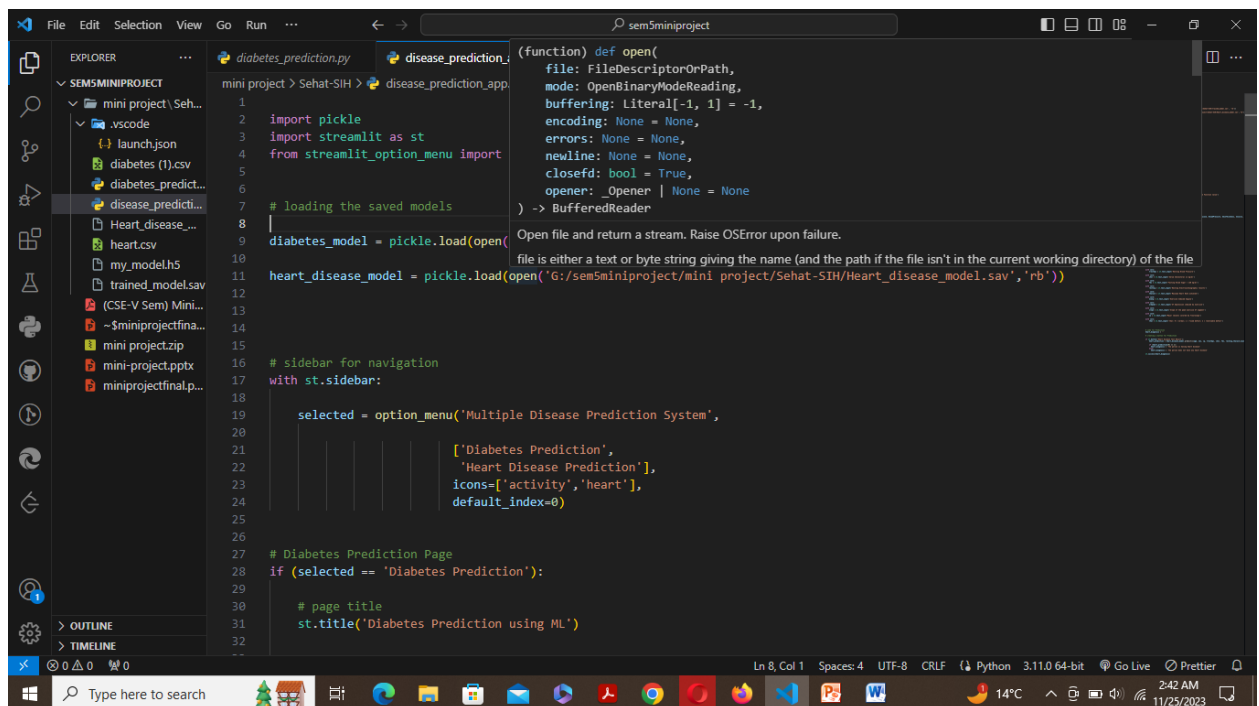
# JSS Academy of Technical Education – NOIDA

## Mini Project/Internship Assessment (KCS-354) (CSE III Semester)

### WORK DONE



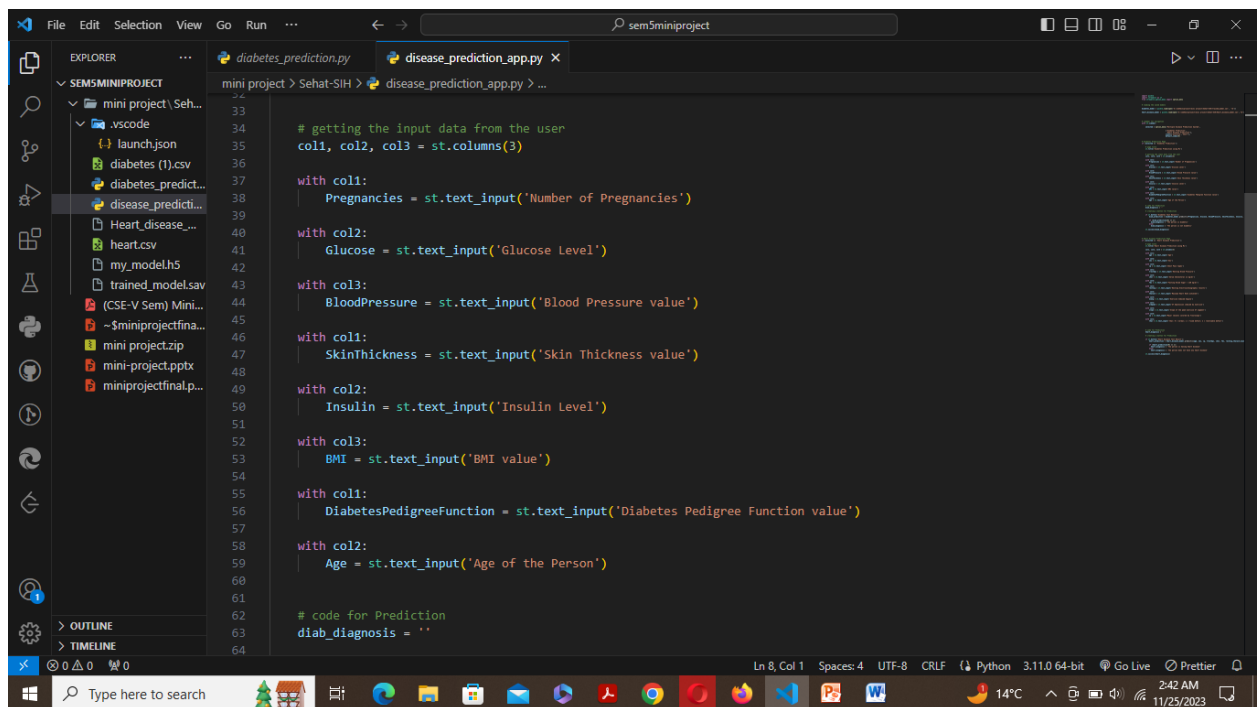
```
1 import numpy as np
2 import pickle
3
4 # loading the saved model
5 loaded_model = pickle.load(open('G:/sem5miniproject/mini project/Sehat-SIH/trained_model.sav', 'rb'))
6
7 input_data = (5,166,72,19,175,25.8,0.587,51)
8
9 # changing the input_data to numpy array
10 input_data_as_numpy_array = np.asarray(input_data)
11
12 # reshape the array as we are predicting for one instance
13 input_data_reshaped = input_data_as_numpy_array.reshape(1,-1)
14
15 prediction = loaded_model.predict(input_data_reshaped)
16 print(prediction)
17
18 if (prediction[0] == 0):
19     print('The person is not diabetic')
20 else:
21     print('The person is diabetic')
```



```
1 import pickle
2 import streamlit as st
3 from streamlit_option_menu import option_menu
4
5 # loading the saved models
6 diabetes_model = pickle.load(open('G:/sem5miniproject/mini project/Sehat-SIH/Diabetes_model.sav', 'rb'))
7 heart_disease_model = pickle.load(open('G:/sem5miniproject/mini project/Sehat-SIH/Heart_disease_model.sav', 'rb'))
8
9 # sidebar for navigation
10 with st.sidebar:
11     selected = option_menu('Multiple Disease Prediction System',
12                             ['Diabetes Prediction',
13                              'Heart Disease Prediction'],
14                             icons=['activity', 'heart'],
15                             default_index=0)
16
17 # Diabetes Prediction Page
18 if (selected == 'Diabetes Prediction'):
19     # page title
20     st.title('Diabetes Prediction using ML')
```

# JSS Academy of Technical Education – NOIDA

## Mini Project/Internship Assessment (KCS-354) (CSE III Semester)



```
diabetes_prediction.py
disease_prediction_app.py x

mini project > Sehat-SIH > disease_prediction_app.py > ...

# getting the input data from the user
col1, col2, col3 = st.columns(3)

with col1:
    Pregnancies = st.text_input('Number of Pregnancies')

with col2:
    Glucose = st.text_input('Glucose Level')

with col3:
    BloodPressure = st.text_input('Blood Pressure value')

with col1:
    SkinThickness = st.text_input('Skin Thickness value')

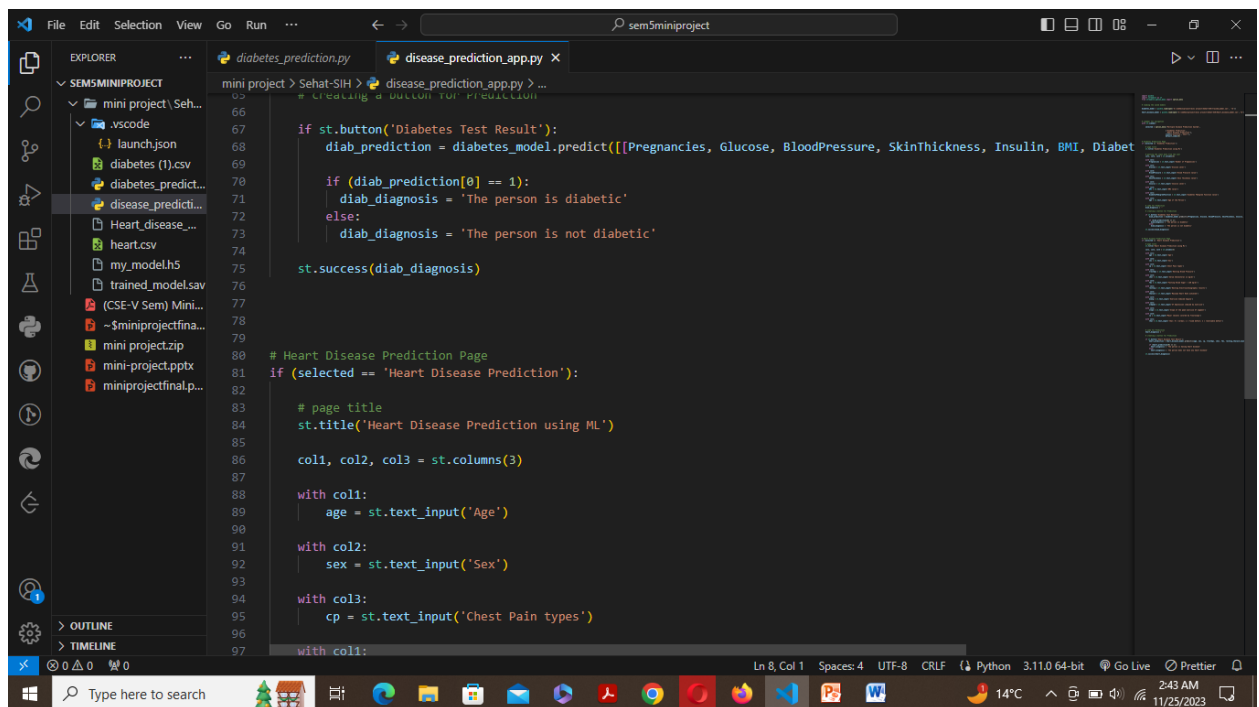
with col2:
    Insulin = st.text_input('Insulin Level')

with col3:
    BMI = st.text_input('BMI value')

with col1:
    DiabetesPedigreeFunction = st.text_input('Diabetes Pedigree Function value')

with col2:
    Age = st.text_input('Age of the Person')

# code for Prediction
diab_diagnosis = ''
```



```
diabetes_prediction.py
disease_prediction_app.py x

mini project > Sehat-SIH > disease_prediction_app.py > ...

# creating a button for Prediction

if st.button('Diabetes Test Result'):
    diab_prediction = diabetes_model.predict([[Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, Diabet

    if (diab_prediction[0] == 1):
        diab_diagnosis = 'The person is diabetic'
    else:
        diab_diagnosis = 'The person is not diabetic'

    st.success(diab_diagnosis)

# Heart Disease Prediction Page
if (selected == 'Heart Disease Prediction'):

    # page title
    st.title('Heart Disease Prediction using ML')

    col1, col2, col3 = st.columns(3)

    with col1:
        age = st.text_input('Age')

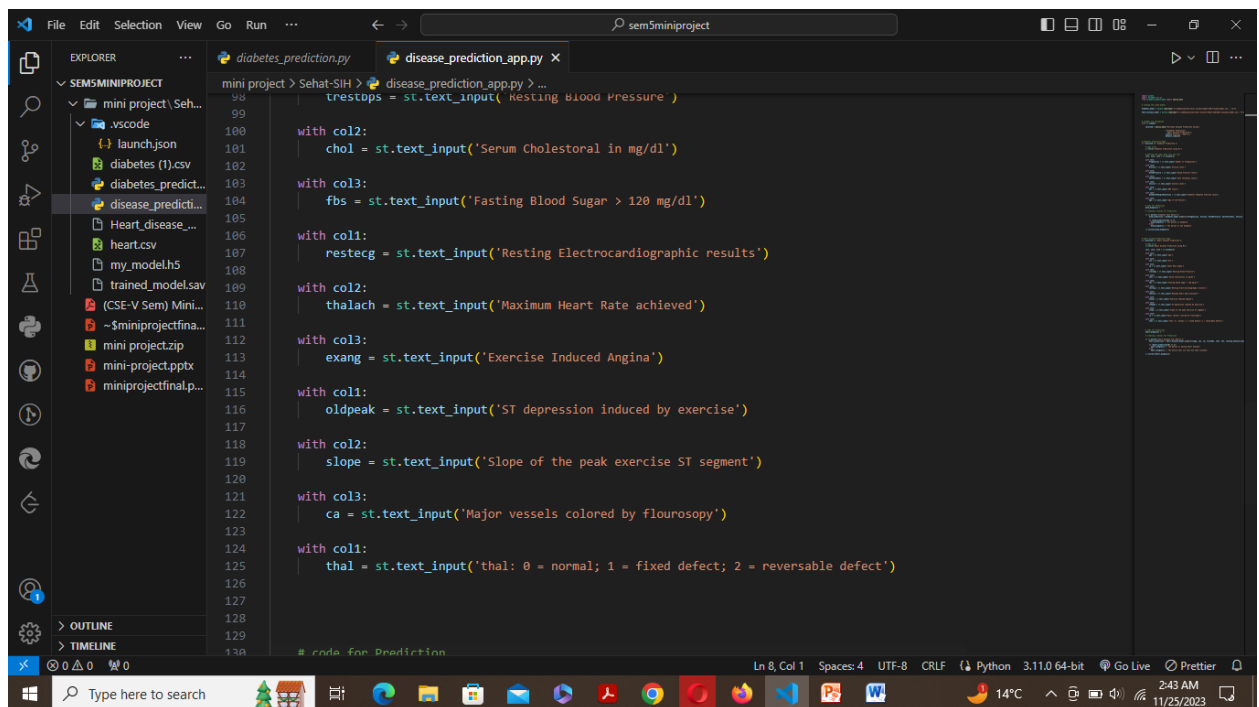
    with col2:
        sex = st.text_input('Sex')

    with col3:
        cp = st.text_input('Chest Pain types')

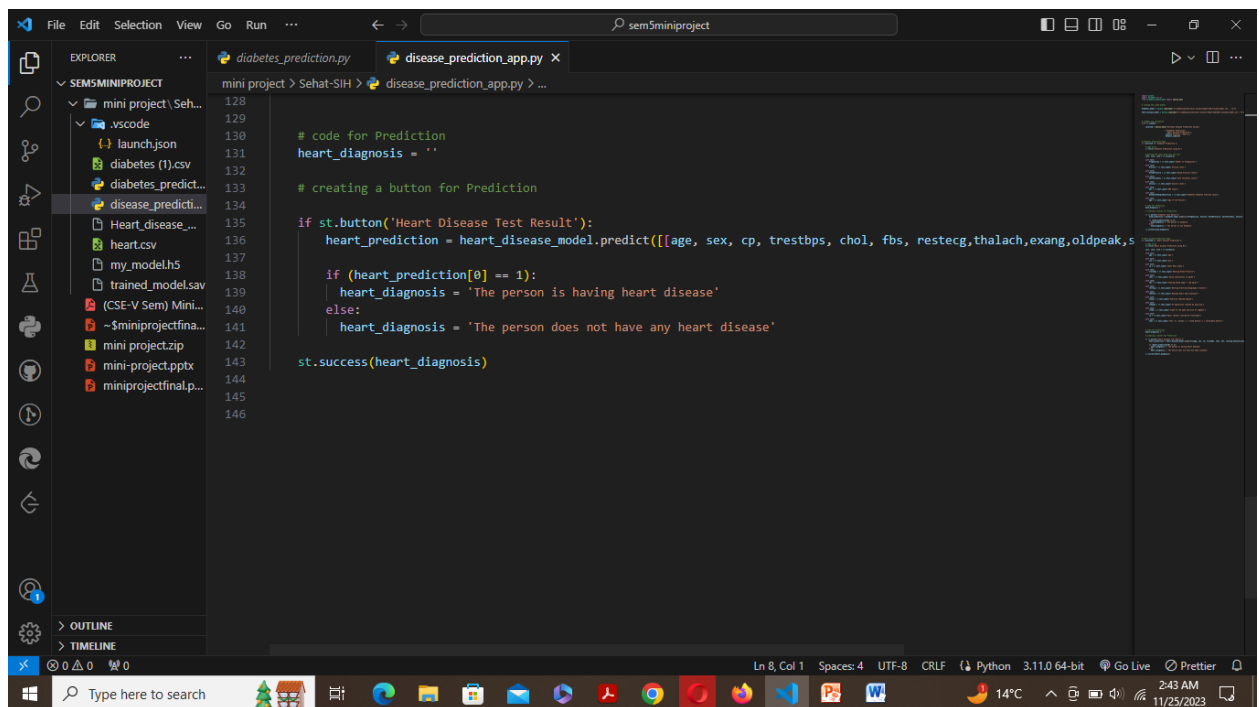
    with col1:
```

# JSS Academy of Technical Education – NOIDA

## Mini Project/Internship Assessment (KCS-354) (CSE III Semester)



```
mini project > Sehat-SIH > disease_prediction_app.py > ...
98 trestbps = st.text_input('Resting Blood Pressure')
99
100
101 with col2:
102     chol = st.text_input('Serum Cholesterol in mg/dl')
103
104 with col3:
105     fbs = st.text_input('Fasting Blood Sugar > 120 mg/dl')
106
107 with col1:
108     restecg = st.text_input('Resting Electrocardiographic results')
109
110 with col2:
111     thalach = st.text_input('Maximum Heart Rate achieved')
112
113 with col3:
114     exang = st.text_input('Exercise Induced Angina')
115
116 with col1:
117     oldpeak = st.text_input('ST depression induced by exercise')
118
119 with col2:
120     slope = st.text_input('Slope of the peak exercise ST segment')
121
122 with col3:
123     ca = st.text_input('Major vessels colored by flourosopy')
124
125 with col1:
126     thal = st.text_input('thal: 0 = normal; 1 = fixed defect; 2 = reversable defect')
127
128
129
130 # code for Prediction
```



```
mini project > Sehat-SIH > disease_prediction_app.py > ...
128
129 # code for Prediction
130 heart_diagnosis = ''
131
132 # creating a button for Prediction
133
134 if st.button('Heart Disease Test Result'):
135     heart_prediction = heart_disease_model.predict([[age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca, thal]])
136
137     if (heart_prediction[0] == 1):
138         heart_diagnosis = 'The person is having heart disease'
139     else:
140         heart_diagnosis = 'The person does not have any heart disease'
141
142 st.success(heart_diagnosis)
```

### *Conclusions and Future Scope*

#### **Conclusion:**

The Multiple Disease Prediction System designed in this project demonstrates the application of machine learning models in predicting diabetes and heart disease based on user-inputted parameters. Leveraging pickle files to store pre-trained models and Streamlit for creating an interactive web interface, users can conveniently obtain predictions regarding these critical health conditions.

The Diabetes Prediction module utilizes essential health indicators such as glucose levels, blood pressure, BMI, and other factors to provide insights into the likelihood of an individual having diabetes. Similarly, the Heart Disease Prediction module incorporates various medical attributes like age, cholesterol levels, exercise-induced angina, and more to determine the presence of heart disease.

The system's functionality provides a user-friendly experience, offering predictions at the click of a button, thereby potentially aiding individuals in taking proactive measures for their health based on predictive analytics.

#### **Future Scope:**

Despite the successful implementation of this predictive system, several areas warrant further exploration and enhancements:

1. **Model Refinement:** Continual refinement of the machine learning models using larger and more diverse datasets could improve prediction accuracy and robustness. Fine-tuning hyperparameters and exploring advanced algorithms might enhance the model's performance.
2. **Feature Expansion:** Incorporating additional relevant features or exploring different feature engineering techniques might strengthen the predictive power of the models. Exploring genetic or lifestyle-related data could further refine predictions.
3. **Real-time Data Integration:** Integrating real-time health monitoring devices or APIs to fetch live health data could make the system more dynamic and responsive, providing users with up-to-date predictions.
4. **User Experience Enhancement:** Enhancing the user interface by adding data validation, visualization of input-output relationships, and educational content about diabetes and heart disease could improve user engagement and understanding.
5. **Deployment and Scalability:** Deploying the system on cloud platforms for scalability and accessibility to a broader audience while ensuring data security and privacy measures are upheld.
6. **Incorporating Additional Diseases:** Expanding the system's scope to predict and diagnose a broader range of diseases, possibly incorporating other critical health conditions, thereby creating a comprehensive health assessment tool.

In conclusion, while the developed system showcases the potential of machine learning in health prediction, ongoing improvements and expansions are crucial for its continual evolution and effectiveness in assisting individuals in managing and maintaining their health proactively

## ***REFERENCES***

1. American Diabetes Association. (2020). Classification and Diagnosis of Diabetes: Standards of Medical Care in Diabetes. Diabetes Care, 43(Supplement 1), S14–S31. <https://doi.org/10.2337/dc20-S002>
2. Dey, J., & Chakraborty, S. (2020). Machine Learning Applications in Diabetic Healthcare: A Review. Journal of Medical Systems, 44(4), 1-17. <https://doi.org/10.1007/s10916-020-1531-3>
3. Krittanawong, C., Zhang, H., & Wang, Z. (2020). Artificial Intelligence in Precision Cardiovascular Medicine. Journal of the American College of Cardiology, 75(23), 2869–2879. <https://doi.org/10.1016/j.jacc.2020.04.009>
4. Lavecchia, A. (2019). Machine-learning approaches in drug discovery: methods and applications. Drug Discovery Today, 24(3), 750-756. <https://doi.org/10.1016/j.drudis.2018.11.039>
5. Patel, N., & Donga, S. (2018). A Comparative Study of Machine Learning Algorithms for Heart Disease Prediction. International Journal of Computer Applications, 180(38), 10-14. <https://doi.org/10.5120/ijca2018917937>
6. Streamlit. (n.d.). Streamlit Documentation. Retrieved from <https://docs.streamlit.io/en/stable/index.html>
7. TensorFlow. (n.d.). TensorFlow Documentation. Retrieved from <https://www.tensorflow.org/>
8. Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830. <https://doi.org/10.5555/1953048.2078195>