Federated Learning Based Lung Cancer Classification

By
Mostafizur Rahman
212-15-4130

Gool A Tabassum
212-15-4139

FINAL YEAR DESIGN PROJECT REPORT

This Report Presented in Partial Fulfillment of the Requirements for the **Degree of Bachelor of Science in Computer Science and Engineering**

Supervised by
Mohammad Jahangir Alam
Assistant Professor
Department of Computer Science and Engineering
Daffodil International University

Co-Supervised by
Md. Jihan Parvez
Lecturer
Department of Computer Science and Engineering
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

May 14, 2025

APPROVAL

This Project titled **"Federated Learning Based Lung Cancer Classification"**, submitted by Mostafizur Rahman, ID No: **212-15-4130** and Gool A Tabassum, ID No: **212-15-4139** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **14 May, 2025**.

BOARD OF EXAMINERS

Dr. Fizar Ahmed — **Chairman**
Associate Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Amatul Bushra Akhi — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Mr. Abdullah Al Mamun — **Internal Examiner**
Sr. Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Mohammed Nasir Uddin — **External Examiner**
Professor
Department of Computer Science and Engineering
Jagannath University

DECLARATION

We hereby declare that this project has been done by us under the supervision of **Mohammad Jahangir Alam, Assistant Professor,** Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

13.05.25

Mohammad Jahangir Alam
Assistant Professor
Department of Computer Science and EngineeringDaffodil International University

Co-Supervised by:

Md. Jihan Parvej
Lecturer
Department of Computer Science and EngineeringDaffodil International University

Submitted by:

Mostafizur

Mostafizur Rahman
Student ID: 212-15-4130
Department of Computer Science and EngineeringDaffodil International University

Gool A Tabassum

Gool A Tabassum
Student ID: 212-15-4139
Department of Computer Science and EngineeringDaffodil University

ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions of many individuals over the past two semesters. We are deeply grateful to everyone who has assisted us in one way or another.

First, we express our heartfelt thanks and gratefulness to the almighty for His divine blessing making it possible for us to complete the **Final Year Design Project(FYDP)** successfully.

We are grateful and wish our profound indebtedness to **Mohammad Jahangir Alam**, **Assistant Professor**, Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh. Deep knowledge and keen interest of our supervisor in the field of **Deep Learning** to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting themat all stages have made it possible to complete this project.

We would like to express our heartfelt gratitude to the Head of the Department of Computer Science and Engineering, for his kind help in finishing our project and also to other faculty members and the staff of the Department of Computer Science and Engineering, Daffodil International University.

We would like to thank our entire course-mates at Daffodil International University, who took part in this discussion while completing the coursework.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

ABSTRACT

In this paper, an interpretable and privacy-preserving lung cancer classification framework based on federated learning (FL) combined with Explainable AI (XAI) approaches is proposed. The system comprises of CNN, LeNet5, AlexNet, ResNet50 deep learning model that have been trained using IQ-OTH/NCCD CT scan images to detect lung abnormalities to be malignant, benign, and normal. In this research centralized and federated training scheme were used. Three aggregation algorithms are also applied for federated learning based on FedAvg, FedProx and FedADMM, on both IID and non-IID permutations of the dataset. We show that ResNet50 with FedProx performed best in these cases: it exhibits high global accuracy in non-IID setups while maintaining high privacy of data. Visual explanations were generated using Grad-CAM to improve the interpretability and clinical confidence in the AI predictions. A web application was included that is built on top of Django, which allows clinicians to provide CT scan and get the corresponding classification output as well as to see the interpretation maps calculating heatmaps. Standardization The system complies with ISO/IEC 25010 for software quality and carries the data through HTTPS, and later it is prepared for integration with the HL7 FHIR. Despite computational and data limitations, the work successfully proved that the integration of federated deep learning and explain ability is practical for secure, accurate, and transparent diagnostic tasks in medicine. This work provides a scalable approach for real-world clinical translation, encouraging both technical innovation and the ethical application of AI in healthc

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

This chapter provides an overview of the research background, motivation, objectives, adopted methodology, and the expected outcomes of the project. It also outlines the structure of the complete report.

1.1 Introduction

Lung cancer is a worldwide health problem and often a leading cause of cancer-associated mortality. Lung cancer kills about 1.8 million people a year, and there is an urgent need for effective diagnostics. Early detection and diagnosis is essential, as it substantially increases the 5-year survival rate from below 20% in metastatic form to beyond 60% if detected early. However, classical methods for medical image classification, such as centralized training-based methods are confronted with significant challenges. These consist of laws such as the US Health Insurance Portability and Accountability Act (HIPAA) and Europe's General Data Protection Regulation (GDPR) that make it difficult to share sensitive patient data between organizations. Furthermore, the centralized models can have restricted access to heterogeneous datasets, which can hinder the generalization of the model in different populations and imaging protocols. Federated Learning (FL) provides a promising approach for addressing these challenges.[1]

FL allows multiple medical institutions to jointly train a shared machine learning model without exchanging raw patient data, and to make full use of various data while preserving patient privacy. This work aims to provide an implementation of a federated learning approach targeting lung cancer classification using Convolutional Neural Networks (CNNs), based on the IQ-OTH/NCCD lung cancer dataset. The framework is to become accurate privacy-preserving, distributed learning environments with high classification accuracy, precision, and recall. To tackle crucial challenges like data heterogeneity, model convergence and high communication overheads in federated learning systems, it leverages state-of-the-art aggregation methods, i.e., FedAvg [2], FedProx [3], and FedADMM [4].

The project also includes centralized learning with established CNN [10], LeNet5 [11], ResNet50 [20], and AlexNet [25] in order to offer a baseline to the performance comparison with federated learning. We particularly instantiate the learning approach federated learning for CNN, ResNet50 and LeNet5 to test their generalization to distributed scenarios. Additionally, Explainable AI (XAI) methodologies will be integrated to improve model interpretability. [20] XAI guarantee that the diagnostic decisions of the model are transparent and interpretable to the clinician allowing to earn trust and to be adopted in real clinical scenarios. This framework also highlights the inclusiveness by integrating multi-file data cohorts to reduce the bias in the model and to demonstrate that the model is generalizable and applicable across different demographic in a variety of imaging conditions. By tackling these intertwined challenges, this work intends to contribute to building robust, scalable and interpretable diagnostic tools for lung cancer detection.

1.2 Motivation

This project is motivated by computational needs to produce privacy-preserving machine learning-based medical imaging models when ethical or legal concerns limit data sharing. Federated learning permits cooperative learning in distributed healthcare systems, such that models can learn from heterogeneous, real-world data without direct exposure to patient information. Solving this problem is important to enable the advancement of medical diagnostics since it would enable the building of robust and highly generalizable lung cancer detection models. Moreover, Explainable AI provides transparent and trustworthy model predictions that can help adopt clinical setups. This research serves the general medical community with scalable, secure, and interpretable solutions for early cancer detection and, eventually, patient prognosis.

1.3 Objectives

The objectives of this study are as follows:

- Develop a federated learning framework using CNN, ResNet50 and LeNet5 for lung cancer classification on the IQ-OTH/NCCD dataset.
- Implement and compare centralized training using CNN, LeNet5, ResNet50, and AlexNet with federated learning approaches to evaluate performance metrics such as accuracy, precision and recall.
- Ensure data privacy and security through decentralized training across multiple clients, adhering to regulations like HIPAA and GDPR.
- Optimize the federated learning process by applying aggregation strategies, including FedAvg, FedProx, and FedADMM, to enhance model performance under IID and non-IID data distributions.
- Analyze the impact of data heterogeneity on model performance in federated learning settings.
- Integrate Explainable AI techniques to provide interpretable insights into model predictions, enhancing trust and usability in clinical applications.

1.4 Methodology

The study applies the federated learning approach for lung cancer classification on CNN, ResNet50 and LeNet5, which utilized IQ-OTH/NCCD dataset being divided on four clients and a centralized server. Each client trains a local model with CNN, ResNet50 or LeNet5 separately and shares the model parameters (not raw data) with a central server that uses FedAvg, FedProx or FedADMM to aggregate/update the global model. CNN, LeNet5, ResNet50 and AlexNet are utilized for comparisons of performance of centralized training. The framework tests model performance under IID and non-IID data distributions from the perspectives of accuracy, precision, and recall, as well as system computational overhead. Explainable AI methods are used to interpret model predictions, which is essential for transparency. The approach can cope with problems such as data heterogeneity, model convergence, communication overhead that is payable in large scale, all the time giving the priority to data privacy and security.

1.5 Project Outcome

The project aims to develop a reliable federated learning scheme for lung cancer classification with high accuracy, and space and time efficiency without compromising patient privacy. The framework will outperform the centralised training especially when dealing with the heterogeneous dataset. These combined FedAvg, FedProx, and

FedADMM models are expected to provide empirical evidence for federated learning parameter tuning in medical image analysis. Explainable AI will improve the interpretability of the model's predictions, enabling the model to be used in the clinic. Results will inform the construction of scalable, secure and interpretable diagnostic systems, and potentially lead to early lung cancer detection which is more accurate and which results in improved patient outcomes.

1.6 Organization of the Report

This thesis is organised into six chapters, it covers the understanding, design, and evaluation of the presented lung cancer classification system. The process can be represented as:

Chapter 1: Introduction

Introduction This chapter opens the subject of the research project by giving background introduction on lung cancer diagnosis and discussing the need for AI assisted clinical tools. It justifies why federated learning and explainable AI should be used in health for CT image classification. The chapter also provides the problem statement, objectives of the project, methodology used, anticipated findings and overview of what is in the report.

Chapter 2: Literature Review

This chapter reviews the existing literature related to the project. It touches on the latest developments in convolutional neural networks (CNNs) for medical image classification, the use of federated learning in healthcare, and how explainable AI (XAI) techniques such as the Grad Dual-constrained CAM (Grad-CAM). Comparison with others and the bridge types of the proposed approach are presented to strengthen the justification for the proposed approach.

Chapter 3: Methodology

The entire technical architecture and implementation approach is described in this chapter. It represents the centralized training pipelines of CNN, LeNet5, ResNet50, and AlexNet models on the IQ-OTH/NCCD dataset. We define federated learning settings (client-server model, aggregation schemes FedAvg, FedProx, FedADMM, data partitioning in IID and non-IID settings). The section also elaborates on the incorporation of Grad-CAM for interpretability and, lastly, on the creation and deployment of the Django-based web application.

Chapter 4: Results and Discussion

This chapter provides a detailed analysis of the results obtained. The comparison criteria, in addition to the accuracy and loss, are precision, recall, and F1-score for centralized and federated learning models. Graphs and confusion matrices demonstrate the model's performance in various scenarios. The effects of data heterogeneity and the efficacy of different aggregation strategies are examined. The explainability section presents insights from Grad-CAM visualizations, explaining how the models attend to important lung regions in CT scans.

Chapter 5: Standards, Design and Design Challenges

This chapter discusses conformity to the applicable software, hardware, and communication standards of clinical-grade AI systems. It assesses the social and environmental impact of the solution, discusses ethical aspects, and presents sustainability plans. The chapter also has a section on project management and cost estimation that covers a budget and an alternative financing schema. Complex engineering problem mappings and associated activities supporting educational accreditation criteria are also addressed.

Chapter 6: Conclusion and Future Work

The last chapter concludes this project by reviewing its contributions, identifying its limitations, and suggesting future works. It highlights the significance of federated learning in privacy-preserved healthcare AI and opportunities and challenges regarding model scaling, real-time inference, and standardization regarding the incorporation of the system into clinical workflows.

Chapter 2
Background

This chapter provides the foundational knowledge necessary to understand the research, including a literature review of lung cancer detection, federated learning, and explainable AI, followed by a gap analysis identifying areas for contribution.

2.1 Introduction

Lung cancer remains a leading cause of cancer-related death worldwide, with early detection being critical for improving survival rates. Profound learning advances—especially convolutional neural networks (CNNs)—have greatly improved medical image classification for lung cancer diagnosis. Still, problems, including data privacy, limited datasets, model interpretability, and computational efficiency, endure. While Explainable AI (XAI) enhances model transparency for clinical adoption, Federated Learning (FL) answers privacy issues by allowing cooperative model training without sharing sensitive patient data. Emphasizing federated learning for lung cancer classification using the IQ-OTH/NCCD dataset, this chapter reviews pertinent literature, summarizes similar applications and related research, and notes gaps this project aims to address.

2.2 Literature Review

Since lung cancer is among the major cancer deaths globally, early detection is crucial for survival rates for patients with this disease. Deep learning, especially in lung cancer diagnosis processes, shows excellent progress in medical image classification. The unchangeable difficulties in dataset limits, as well as model interpretation and data correctness, demand immediate attention since they affect federated learning (FL) development of safe medical applications.

Key research on lung cancer detection, federated learning, and explainable artificial intelligence is compiled in this part together with their approaches, results, constraints, and future directions. The studied material is succinctly compiled in the table below.

Table 2.1: Summary of Literature Reviewed.

Author (s)	Year	Title	Methodology	Key Findings
Maleki et al. [5]	2023	An intelligent algorithm for lung cancer diagnosis using extracted features from Computerized Tomography images	Feature extraction, dimensionality reduction (PCA, LDA), and genetic algorithms with SVM and RF	SVM and RF achieved 95% accuracy
S. Wankhade et al. [6]	2023	A novel hybrid deep learning method for early detection of lung cancer using neural networks	Hybrid neural network combining 3D-CNN with other NN techniques	3D-CNN achieved 98.8% accuracy

I. Shafi et al. [7]	2022	An Effective Method for Lung Cancer Diagnosis from CT scan Using Deep Learning-Based Support Vector Network	Deep learning integrated with SVM, capsule networks for segmentation	Capsule networks with SVM achieved 94% accuracy
K.-H. Yu et al. [8]	2020	Reproducible Machine Learning Methods for Lung Cancer Detection Using Computed Tomography Images: Algorithm Development and Validation	Evaluation of reproducible machine learning modules using Docker containers	CNN performed well but faced generalization issues
A. A. Ahmed et al. [9]	2022	Deep Learning Approaches in Histopathology	Surveyed DL methods for analyzing tumor pathology	Up to 83% accuracy for HER2 determination
B. K. Hatuwal et al. [10]	2020	Lung cancer detection using convolutional neural network on histopathological images	CNN for classifying benign tissue, adenocarcinoma, and squamous cell carcinoma	CNN achieved 97.2% validation accuracy
S. Mehmood et al. [11]	2022	Malignancy detection in lung and colon histopathology images using transfer learning with class selective image processing	Modified AlexNet with CSIP	AlexNet + CSIP achieved 98.4% accuracy
S. Vijh et al. [12]	2020	Hybrid bio-inspired algorithm and convolutional neural network for automatic lung tumor detection	WOA_APSO algorithm combined with CNN	WOA_APSO achieved 97.18% accuracy
D. M. Ibrahim et al. [13]	2021	Deep-chest: Multi-classification deep learning model for diagnosing COVID-19, pneumonia, and lung cancer chest diseases	Deep learning model using VGG19-CNN, ResNet152V2, ResNet152V2 with GRU/Bi-GRU for multi-classification of chest diseases	VGG19-CNN: 98.05% accuracy, recall, and precision, 99.66% AUC
P. Nanglia et al. [14]	2020	A hybrid algorithm for lung cancer classification using SVM and Neural	Hybrid model combining SVM and Neural Networks, feature extraction	SVM + Neural Networks: 98.08% accuracy, 98.17% precision, 96.5% recall,

		Networks	using SURF, and optimization with Genetic Algorithm	97% F-measure
F. F. Babar et al. [15]	2022	Federated Active Learning with Transfer Learning: Empowering Edge Intelligence for Enhanced Lung Cancer Diagnosis	Implements Federated Active Learning with Transfer Learning (FAL-TL) and entropy-based uncertainty assessment for efficient labeling.	99.20% accuracy (IQ-OTH/NCCD), 98.70% (Chest CT)
C. Usharani et al. [16]	2025	FedLRes: enhancing lung cancer detection using federated learning with convolution neural network (ResNet50)	Uses federated learning with ResNet50 and adaptive feature optimization for privacy-preserving lung cancer detection.	99.40% accuracy
E. S. N et al. [17]	2021	3D CNN with Visual Insights for Early Detection of Lung Cancer Using Gradient-Weighted Class Activation	3D CNN model with Grad-CAM for interpretability and visual insights to aid radiologists in early detection of lung cancer	3D CNN with Grad-CAM: 97.17% accuracy
A. Shimazaki et al. [18]	2022	Deep learning-based algorithm for lung cancer detection on chest radiographs using the segmentation method	Deep learning-based segmentation model for detecting lung cancer on chest radiographs	Sensitivity: 0.73, mFPI not specified
M. A. Heuvelmans et al. [19]	2021	Lung cancer prediction by Deep Learning to identify benign lung nodules	Introduced a pre-defined benign nodule rule-out test with 99% sensitivity threshold.	LCP-CNN, AUC: 94.5%, Sensitivity: 99%, 22.1% nodules ruled out as benign
Y. K. S et al. [20]	2024	Explainable lung cancer classification with ensemble transfer learning of VGG16, Resnet50 and InceptionV3 using grad-cam	Uses ensemble learning with VGG16, ResNet50, and InceptionV3 for lung cancer classification with Grad-CAM for interpretability.	98.18% accuracy
Q. Pei et al. [21]	2022	Artificial intelligence in clinical applications for lung cancer: diagnosis,	Overview of AI applications in diagnosis, treatment, and prognosis using AI techniques	Improved diagnostic accuracy and treatment prediction (no specific metrics)

		treatment and prognosis		
M. J. P. Priyadarsini et al. [22]	2023	Lung diseases detection using various deep learning algorithms," Journal of Healthcare Engineering	Uses CNN-based models (Sequential, Functional, Transfer learning) for lung disease classification (pneumonia, tuberculosis, lung cancer).	Pneumonia: 98.43% accuracy, Tuberculosis: 99.4% accuracy, Cancer: 99.9% accuracy
N. V. R et al. [23]	2023	EXtRANFS: an automated lung cancer malignancy detection system using extremely randomized feature selector	Combines transfer learning (VGG16) with an Extremely Randomized Tree Classifier for feature selection and MLP for classification.	99.09% accuracy, 98.33% sensitivity, 98.33% F1-score
S. P. Primakov et al. [24]	2022	Automated detection and segmentation of non-small cell lung cancer computed tomography images	Automated pipeline for detecting and segmenting NSCLC using modified 2D U-Net and image preprocessing techniques	Improved reproducibility and clinical relevance compared to manual segmentation

2.2.1 Similar Applications

A few documents studied deep and federated learning for lung cancer detection using different techniques and datasets. Maleki et al. [5] proposed PCA, LDA, and GA with SVM and Random Forest, getting 95% accuracy on small CT data. Wankhade et al. [6] used a hybrid 3D-CNN on the LIDC-IDRI dataset and achieved an accuracy of 98.8% without the detail of preprocessing. Shafi et al. [7] combined capsule networks and SVM on LUNA16 with 94% accuracy, but data insufficiency issues were not explored. Yu et al. [8] analyzed the performance of CNNs with 'Docker' containers and pointed out the problems of data de-identification. Ahmed et al. [9] also used deep learning on tumor pathology and got 83% accuracy for HER2 status, although their method is difficult to use in the clinical setting. Hatuwal et al. [10] applied CNNs on histopathological images and achieved 97.2% accuracy without providing dataset details. Mehmood et al. [11] adapted AlexNet for malignancy detection with 98.4% accuracy, applicable to certain diseases only. Vijh et al. [12] applied the WOA_APSO to CNNs, obtaining 97.18% accuracy on a small dataset. Ibrahim et al. [13] used a VGG19-CNN-based multi-classification model with 98.05% success but a dilemma in between the diagnoses. Nanglia et al. [14] presented a mixed SVM-Neural Network model and scored 98:08% accuracy for 500 CT images.

In the context of federated learning, Babar et al. [15] proposed a Federated Active Learning with Transfer Learning (FAL-TL) framework for the IQ-OTH/NCCD dataset with an accuracy of 99.20% using an entropy-based uncertainty estimation. Usharani et al. [16] applied federated learning to ResNet50 on IQ-OTH/NCCD, and they achieved 99.40% accuracy while concentrating on privacy-preserving diagnosis. Interpretable AI has been studied by E. S. N et al. [17], where 3D CNNs with Grad-CAM were applied to achieve 97.17% accuracy.

2.2.2 Related Research

Related research focuses on deep learning, federated learning, and explainable AI in medical imaging. A hybrid approach with CT images was developed by N. Maleki et al. (2023) which joined feature extraction with PCA and LDA dimension reduction algorithms along with genetic algorithms. The study applied SVM and Random Forest classifiers which delivered results with 95% accuracy. The study stressed early diagnosis importance although its small image dataset displayed 364 pictures (238 cancerous with 126 noncancerous images) as the major limitation. Developing a larger CT imaging database constitutes one of the main future development pathways while researchers strive to implement more sophisticated machine learning methods for performance improvement. [5]

The hybrid neural network described by Wankhade et al. (2023) integrated 3D-CNN architecture to evaluate CT images within the LIDC-IDRI database. Lung cancer detection accuracy reached 98.8% by using the applied model. The article did not present sufficient information about preprocessing techniques, which led to reduced reproducibility of results. Future applications require extended datasets to reach better results and make outcomes consistent according to the study results. [6]

The team of I. Shafi et al. (2020) applied their deep learning framework to LUNA16 data, which led to a 94% accuracy when capsule networks worked with SVM classifiers. The developed model proved to be superior to several other established segmentation methodologies. The research omitted solutions for potential data collection limitations like computational efficiency and biases in the data samples. The research must tackle present problems while looking into real-time diagnostic system design. [7]

K.-H. Yu et al. (2020) evaluated the algorithms from multiple winning Kaggle competitions designed to detect lung cancer. The paper concentrated on reproducible benchmarking using Docker containers as its core methodology. CNN-based approaches showed effectiveness in the study, yet researchers highlighted crucial barriers due to data de-identification along with dataset constraints and insufficient test samples. Future research attempts to reduce existing barriers in order to enhance the trustworthy operation of machine learning technology in medical environments. [8]

In the past A. A. Ahmed et al. (2022) explored the use of deep learning for tumor pathology analysis. Models showed up to 83% accuracy in HER2 status determination, which they improved over conventional tests, as evidenced in the study. While these positive results, the study also pointed out issues with methodological and even protocol standardization, and the interpretative challenges they present, which have slowed clinical implementation. Future work will be to improve in clinical adoption and model reliability. [9]

CNNs were used by Hatuwal et al. (2020) to analyze histopathological images of CRCLs achieving up to 97.2% validation accuracy. However, the description of preprocessing methods and more detail regarding the dataset used is lacking, which prevents reproducing its findings. Future directions consist of preprocessing standards, data set diversity in order to provide robust performance across different clinical settings. [10]

In particular, S. Mehmood et al. (2022) used a CSIP-based modified AlexNet architecture, with an accuracy of 98.4%. Even though the study showed that transfer learning is effective for malignancies, it suggested that it also be extended to cover other diseases for broader applicability. Finally, future work also involves advancing architectures for better accuracy and performance. [11]

In their research in 2020, S. Vijh et al. came up with a brand new WOA_APSO algorithm for feature selection coupled with CNNs. The accuracy of the model was 97.18%, and it can detect lung tumors very well. Nevertheless, the dataset was too small to generalize.

Future research is aimed to validate the algorithm in more significant data and more extensive clinical cases. [12]

To denote that, D. M. Ibrahim et al. (2021) built a deep learning mechanism for the multi-classification of chest diseases based on chest X-ray and CT images. These datasets were combined and used with models like VGG19-CNN and ResNet152V2 with GRU/Bi-GRU based on this study. The AUC obtained was 99.66% and an accuracy of 98.05%. One challenge was overlapping diagnoses, and the second challenge was image quality. Future work entailed expanding datasets and employing other architectures. [13]

In P. Nanglia et al. (2020) they proposed a hybrid SVM and Neutral Network model to enhancing lung cancer classification. With Genetic Algorithm used for optimization, 500 CT images were used to produce an accuracy of 98.08%, precision of 98.17%, and recall of 96.5%. Additional work is needed, as they were compared to Deep Neural Networks in the case of small dataset size. [14]

To improve lung cancer diagnosis while preserving the patient privacy, F. F. Babar et al. (2022) presented a Federated Active Learning with Transfer Learning (FAL-TL) framework. A second dataset used for this study is the IQ-OTH/NCCD dataset (1190 images from 110 patients), jointly along with the publicly available Chest CT Scan dataset. An entropy based uncertainty assessment on the proposed model was done with a view to employ it with efficient sample annotation and achieved an accuracy of 99.20 and 98.70 percent respectively on the IQ-OTH/NCCD dataset and the Chest CT dataset. The study, however, does not point out any specifics limitations; instead, it provides a basis for future exploration of dynamic transfer learning strategies and more sophisticated active learning strategies to maximize performance. [15]

FedLRes framework with ResNet50 to conduct privacy preserving lung cancer diagnosis was presented by C. Usharani et al. (2025). A dataset with 822 CT images used for training and 275 CT images used for validation from the Iraq Oncology Teaching Hospital was used for the study. According to model 99.40 % was achieved in accuracy. Specific limitations were not detailed, but future research will extend the use of federated learning in medical imaging in a more general sense, specifically to build a more general application of federated learning in the medical imaging, from the perspective of maintaining data privacy while utilizing the distributed data. [16]

To address the problem of detecting and segmenting nonsmall cell lung cancer in CT scans, S. P. Primakov et al. (2022) presented an automated pipeline. It was performed on 1328 scans using a modified 2D U-Net, which did better than the manual methods. Future work consists of increasing the training set and applying the technique in radiomic studies to improve results. [17]

B. In 2022, Shimazaki et al. validated a deep learning based lung cancer segmentation model for chest radiographs with 73% sensitivity. The model had potential of further improvement using 629 training radiographs and 151 testing radiographs. In future work, this method will be enhanced to improve the segementation accuracy and applied to larger datasets for better generalization. [18]

To maintain at least 99% sensitivity, a benign nodule rule out test was proposed by M. A. Heuvelmans et al. (2021). The LCP-CNN was used in the study using the same CT scans from the NLST and LUCINDA datasets (2106 nodules + 205 lung cancers) which yielded an AUC of 94.5%, 99.0% sensitivity (allowing 22.1% of nodules to be ruled out as benign). If this approach is adopted, 18.5% of patients could bypass diagnostics and unnecessary follow ups. Nevertheless, the model needs to be validated prospectively in lung cancer screening programs before more widespread use. The usefulness of the LCP-CNN will be further studied in future research to validate this in real world applications. [19]

In Y. K. S et al., 2024, an ensemble learning approach using VGG16, ResNet50 and InceptionV3 is used to improve the accuracy of lung cancer classification. The Grad-CAM was used to improve interpretability of the model. Though the exact number of points in the IQ-OTH/NCCD dataset is not mentioned, it was used. The result of the model proposed was 98.18%. The easiest difficulty was to exclude all pathology in normal cases. Future work will refine training methodologies and add orthogonal imaging for improved diagnosis. [20]

Q. Pei and his colleagues (2022) surveyed the applications of AI in diagnosis, treatment and prognosis of lung cancer, prospectively embracing its accuracy. Although the study addressed AI techniques, it did not give concrete performance measurements. Sampling bias and disease differentiation were the most significant challenges, but future work to standardize the data and improve the interpretability of AI for clinical use was needed [21]

In M. J. P. Priyadarsini et al. (2023), different CNN architectures were used for the disease class identification problem of lungs, including the pneumonia, tubercle disease, and lung cancer. It consisted of 907 lung CT images, 692 from cancer patients, and 215 from normal individuals. The model performed with high classification accuracy of 98.43% for pneumonia, 99.4% tuberculosis, and 99.9% lung cancer. Due to the lack of discussion on limitation, we are planning a future of research of improving model performance by testing different optimizers, changing the learning rates, as well as introducing additional data augmentation techniques along. [22]

In N. V. R et al. (2023), transfer learning with VGG 16, Extremely Randomized Tree Classifier for feature selection and MLP Classifier for lung cancer malignancy detection were used in this paper. The framework, named ExtRanFS, was proposed. The dataset used for study was the IQ-OTH/NCCD dataset consisting of community (25) and individual (85) CT scans from 110 patients (40, malignant cases; 15, benign; 55, normal). Accuracy of the model was at 99.09%, with Sensitivity as 98.33%, F1 score was 98.33%. Although no explicit limitations of the study were mentioned, it provides no further research directions. [23]

To address the problem of detecting and segmenting nonsmall cell lung cancer in CT scans, S. P. Primakov et al. (2022) presented an automated pipeline. It was performed on 1328 scans using a modified 2D U-Net, which did better than the manual methods. Future work consists of increasing the training set and applying the technique in radiomic studies to improve results. [24]

2.3 Gap Analysis

The literature survey demonstrates substantial progress in lung cancer detection methods at the same time showing empty spaces where this project should focus its efforts. The proposed system will be compared against current solutions through a table analysis which addresses federated learning capabilities and explain ability and clinical practicality aspects.

Table 2.2: Comparative Gap Analysis of Existing Lung Cancer Classification Approaches

Features	Maleki et al. [1]	Wankhade et al. [2]	Babar et al. [17]	Usharani et al. [20]	Y. K. S et al. [16]	Proposed System
Federated Learning	No	No	Yes	Yes	No	Yes
Privacy-Preserving Training	No	No	Yes	Yes	No	Yes
Multiple Aggregation Strategies (FedAvg, FedProx, FedADMM)	No	No	No	No	No	Yes
Explainable AI (e.g., Grad-CAM, SHAP)	No	No	No	No	Yes	Yes
Centralized Training Comparison	Yes	Yes	No	No	Yes	Yes
IID and Non-IID Data Analysis	No	No	No	No	No	Yes
Bias Reduction via Diverse Data	No	No	Yes	Yes	Yes	Yes
Clinical Interpretability	No	No	No	No	Yes	Yes
Dataset Size (IQ-OTH/NCCD or equivalent)	Small (364 images)	Not specified	1190 images	1097 images	Not specified	1097 images
Model Accuracy	95%	98.8%	99.20%	99.40%	98.18%	Target: >99%

Most literature depends on centralized training because it causes privacy problems and restricts available dataset variety. Babar et al. [15] and Usharani et al. [16] establish federated learning solutions to address privacy issues without investigating diverse aggregation techniques or handling data distribution types. Y. K. S et al. [20] integrate Explainable AI in their system without applying it to a federated environment. The proposed system brings together federated learning methods with three distinct aggregation strategies (FedAvg, FedProx, FedADMM) while performing centralized training using CNN, LeNet5, ResNet50 and AlexNet and includes explainable AI interpretation features and non-IID and IID data interaction analyses. The approach works to improve both precision and data size scalability along with clinical usability by using diverse information sets to cut down bias.

2.4 Summary

The chapter provided a comprehensive information system about lung cancer detection and discussed deep learning techniques with federated learning principles and explainable AI theory. When reviewing classification improvements with CNN methods, researchers found that these approaches continued to develop. Yet, they face challenges related to dataset specifications, learn data preprocessing methods, and face privacy risks when training occurs at one centralized location. The research application of federated learning maintains strong data protection, and XAI demonstrates interpretation effectiveness. The gap analysis shows that the current methods lack integration between collective aggregation approaches and, IID/non-IID research, and framework side-by-side analysis, which XAI explanatory systems should support. Federated Learning employs ResNet50 and LeNet5 under this system while performing centralized training with LeNet5, ResNet50, and AlexNet and implementing XAI for clinical trust through the IQ-OTH/NCCD dataset.

Chapter 3
Research Methodology

The research design provides step-by-step development of a federated learning platform for lung cancer diagnosis through requirement evaluation followed by design features and implementation protocols for different solution tests. It plans for task distribution and a web application for model deployment. The system includes privacy-protecting training features, diverse aggregation techniques, and AI explainability functions integrated with an interface optimized for clinical use.

3.1 Methodology/Requirement Analysis & Design Specification

3.1.1 Overview

This work offers a thorough and privacy-conscious approach to the federated learning framework integrated with deep learning models to classify lung cancer.[16] By spreading the training process over several client nodes without moving actual patient data, the system is intended to handle data security issues and the demand for accurate, interpretable medical image analysis. First, preprocessing the publicly available IQ-OTH/NCCD dataset—resizing to 224×224 pixels, normalizing intensity values, and augmenting images to ensure robustness—results in 1190 CT scan images across 110 patients. Built on CNN architectures—including custom CNN [10], AlexNet [11], ResNet50 [20], LeNet5 [25], and which are trained in both centralized and federated environments to assess performance under various training paradigms except AlexNet—the framework builds upon Four simulated clients under both IID (independent and identically distributed) and non-IID data scenarios run federated learning to replicate real-world clinical data distribution over several healthcare facilities. Three aggregation methods—FedAvg [2], FedProx [3], and FedADMM [4]—have the central focus of the federated learning approach. Every technique is used to evaluate the resilience and adaptability of model convergence trained on heterogeneous data while keeping compliance with privacy criteria, including HIPAA and GDPR. A baseline, centralized training lets one compare quantitative results using standard classification metrics—accuracy, precision, recall, and F1-score—alongside computational elements, including training time and resource use. Explainable AI techniques are included to improve the clinical trustworthiness of the model's decisions, mainly using Grad-CAM [20] to create heat maps that highlight areas of interest in CT images, enabling clinicians to evaluate the decision-making regions of the neural networks visually. Using HTML, CSS, and JavaScript interfaces, a Django-based web application provides functionalities including image uploads, real-time predictions, confidence score displays, and visual feedback via Grad-CAM overlays, so deploying the complete system. Besides offering a valuable proof of concept, this deployment guarantees medical practitioners' accessibility and usability. Combining federated learning, deep CNN models, explainability, and clinical integration yields a safe, interpretable, and robust solution for lung cancer diagnostics.

3.1.2 Proposed Methodology/ System Design

Below is a conceptual diagram illustrating the Federated Learning setup used in this study:

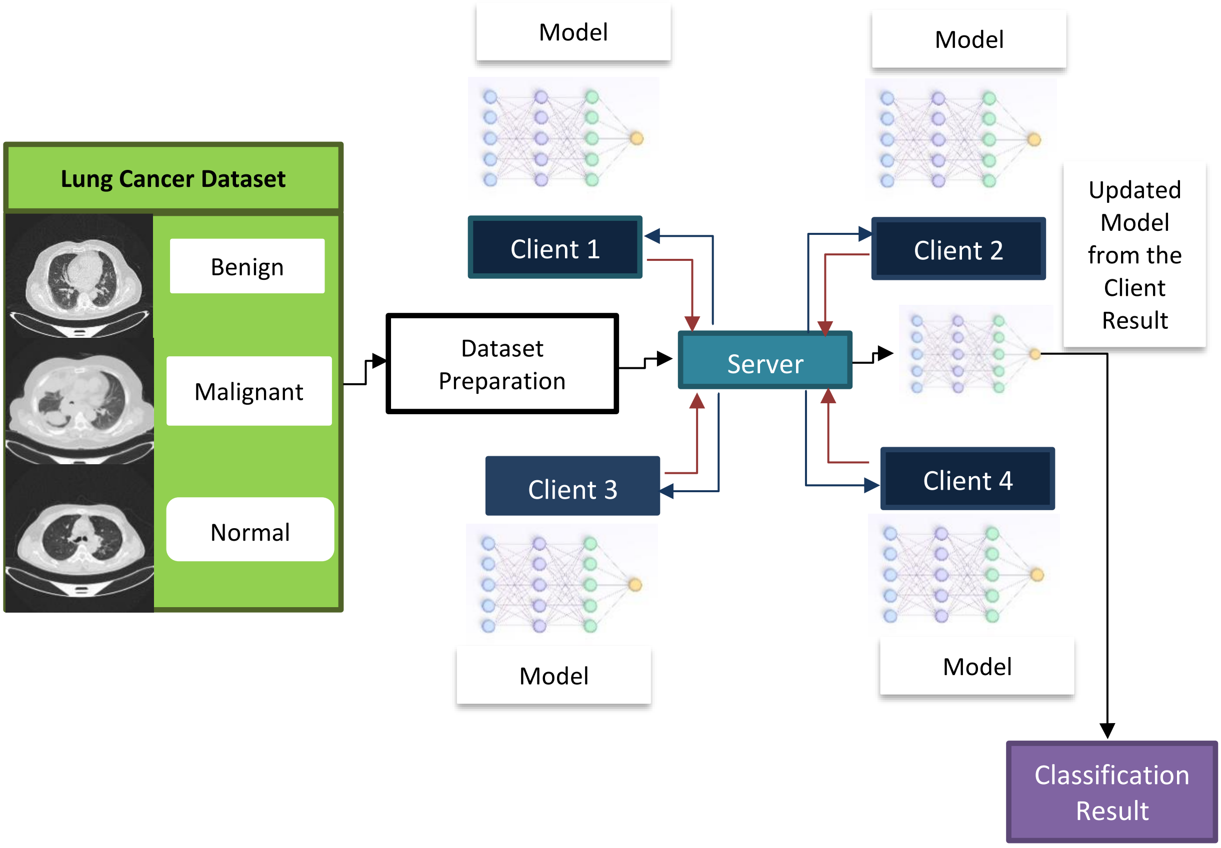

Fig 3.1: Federated Learning Framework with Four Clients and a Central Server

The proposed system adopts Federated Learning (FL) as its base to develop secure distributed classification methods for CT scan analysis. Multiple decentralized client nodes and a central server form the architecture that mirrors a hospital network that retains data storage at each location. [16] Clients receive portions of the IQ-OTH/NCCD data while executing local training sessions that employ LeNet5 and ResNet50 and a custom CNN architecture. The system allows clients to transmit their model parameters (including weights and gradients) to the central server only after each training round. The server applies aggregation with three implementation approaches: Federated Averaging (FedAvg) [2] and Federated Proximal (FedProx) [3], as well as Federated ADMM (FedADMM) [4], depending on the experimental parameters. The processed global model returns to all clients for the next training iteration. The process repeats indefinitely until the training converges or accomplishes the desired rounds. The system operates under IID and non-IID data distributions to evaluate its robust performance against real-world data sets with diverse class distributions between clients. The performance evaluation of the model involves collecting metrics that include accuracy, recall, F1-score, and training loss figures. Each architecture receives the full dataset during centralized training sessions to create performance benchmarks. A combination of Grad-CAM [20] integration into the interpretability framework creates heatmaps that display vital anatomical regions from CT images that affect model decision-making. The trained models are integrated into a Django-based web application that enables clinical users to perform CT scan uploads and receive predictions with visible explanations through the interface. The framework contains a modular structure that delivers privacy protection while being interpretable and adaptable according to medical data privacy standards.

3.1.3 Functional and Nonfunctional Requirements

A set of well stated functional and nonfunctional criteria drives the development of the proposed federated learning system for lung cancer classification. These criteria guarantee that the system not only satisfies technical goals but also follows important performance, usability, and regulatory expectations in the healthcare sector.

Functional Requirements:

- **Data Preprocessing:** The input CT images are normalized to a pixel value range [0, 1] and resized to 224×224 pixels. Rotation, translation, flipping, and Gaussian blur are among the data augmentation methods used to enhance class balance so that every category—benign, malignant, normal—has more than three hundred images.
- **Federated Learning Framework:** The framework of federated learning consists in four distributed clients and one central server. Every customer trains their own local model—using CNN, LeNet5, or ResNet50—on locally kept data, forwarding just model parameters to the central server, so maintaining data privacy.
- **Aggregation Technique:** Three federated aggregation technique are federated averaging (FedAvg), federated proximal (FedProx), and federated ADMM (FedADMM)—are supported by the framework. Performance and stability of these techniques are assessed under both IID and non-IID settings.
- **Centralized Training:** CNN, LeNet5, ResNet50, and AlexNet models are also trained on the complete dataset using a centralized approach to benchmark accuracy and computational efficiency against federated configurations.
- **Explainable AI(XAI):** Explainable artificial intelligence (XAI) uses Gradient-weighted Class Activation Mapping (Grad-CAM) to visualize which areas of a CT image most influenced the model's prediction, so improving interpretability and clinical trust.
- **Web Application:** HTML, CSS, JavaScript, and Django help to create an intuitive web interface. It lets doctors access Grad-CAM-based visual explanations, upload CT images, view classification outputs (malignant, benign, normal).
- **Performance Evaluation:** Accuracy, precision, recall, and F1-score are among the system's computed key performance measures. Furthermore, created to show classification performance over time are confusion matrices and training curves.

Non-functional Requirements:

- **Privacy Compliance:** HIPAA and GDPR rules are followed in design by making sure patient data stays on local devices and only model updates are shared during training.
- **Scalability:** The federated architecture makes the system flexible for actual hospital networks by supporting several clients with different data volumes and distributions.
- **Efficiency:** Targeting a performance benchmark of over 99% on test data, the system is optimized to minimize communication overhead and training time while yet maintaining high classification accuracy.
- **Interpretability:** Grad-CAM visualizations guarantee that the system stays transparent and that clinicians may easily understand its predictions, so improving diagnosis confidence.

- **Usability:** With clinical users in mind, the web interface offers a simple workflow from image upload to diagnosis output and explanation.
- **Robustness:** The federated system guarantees consistent and broad performance over many patient populations and healthcare environments by handling heterogeneous (non-IID) data distributions.

This comprehensive analysis guarantees that the functional as well as nonfunctional features of the system are fully addressed to support dependable, scalable, and interpretable lung cancer diagnosis in clinical settings.

3.1.4 Context Diagram

Context Diagram (DFD-0) of Lung Cancer Classification System:

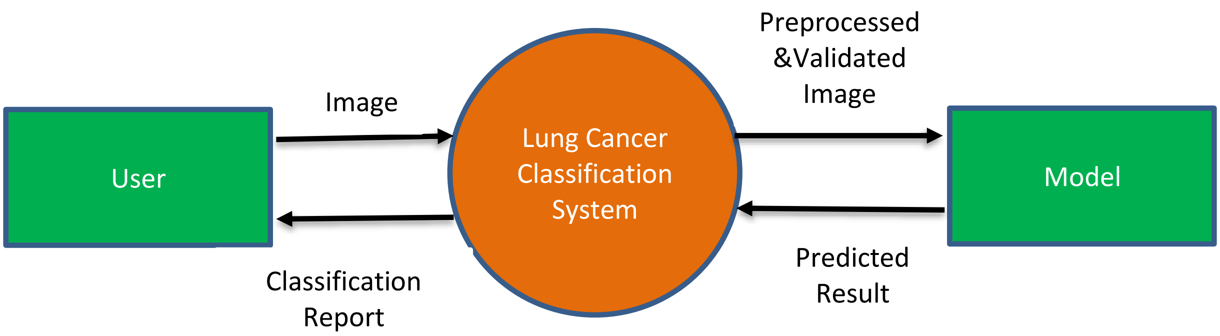

Fig 3.2: Context Diagram of Lung Cancer Classification System

3.1.5 Data Flow Diagram Level 1

Data Flow Diagram Level 1 of Lung Cancer Classification System:

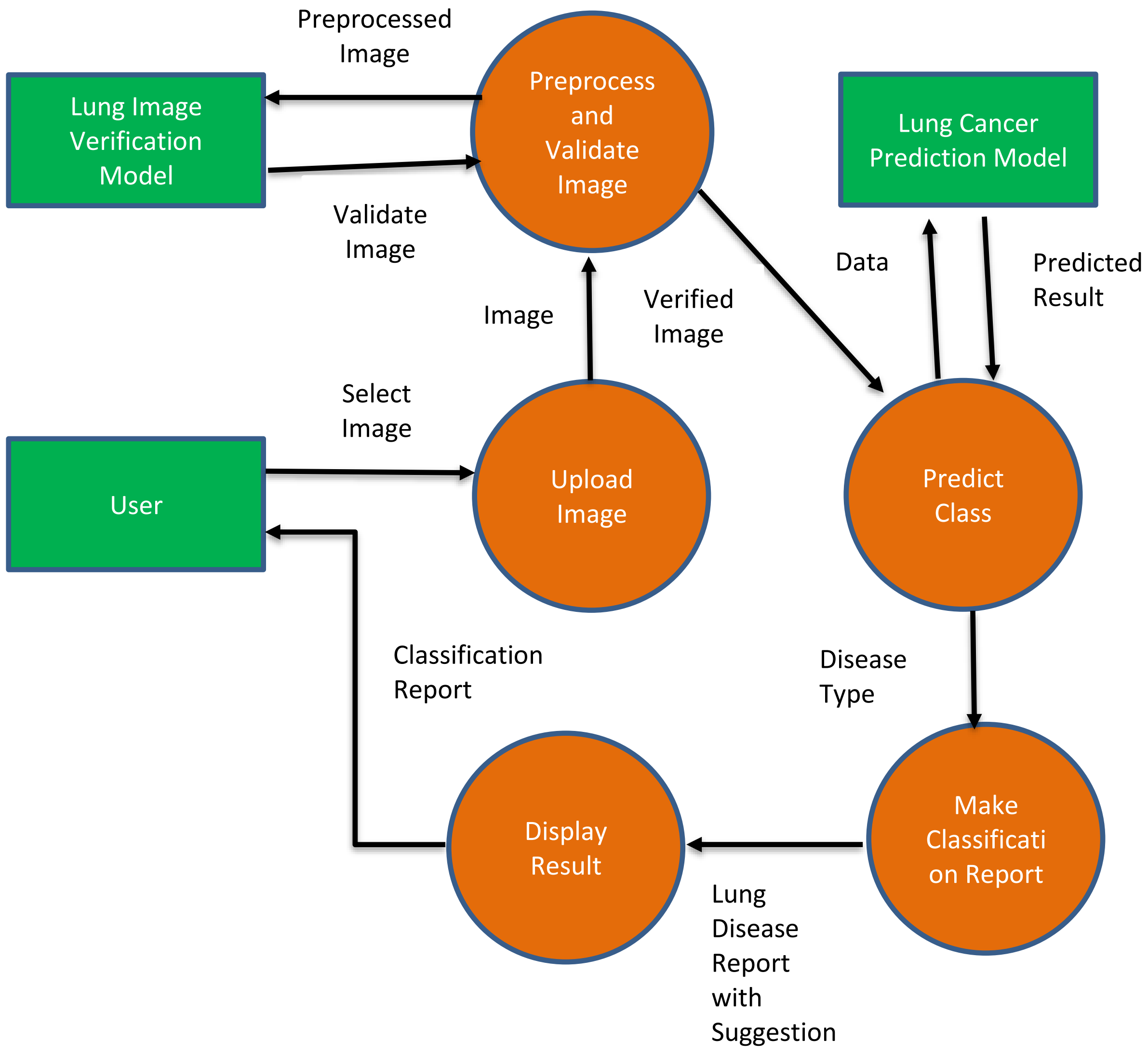

Fig 3.3: DFD-1 of Lung Cancer Classification System

3.1.5 UI Design

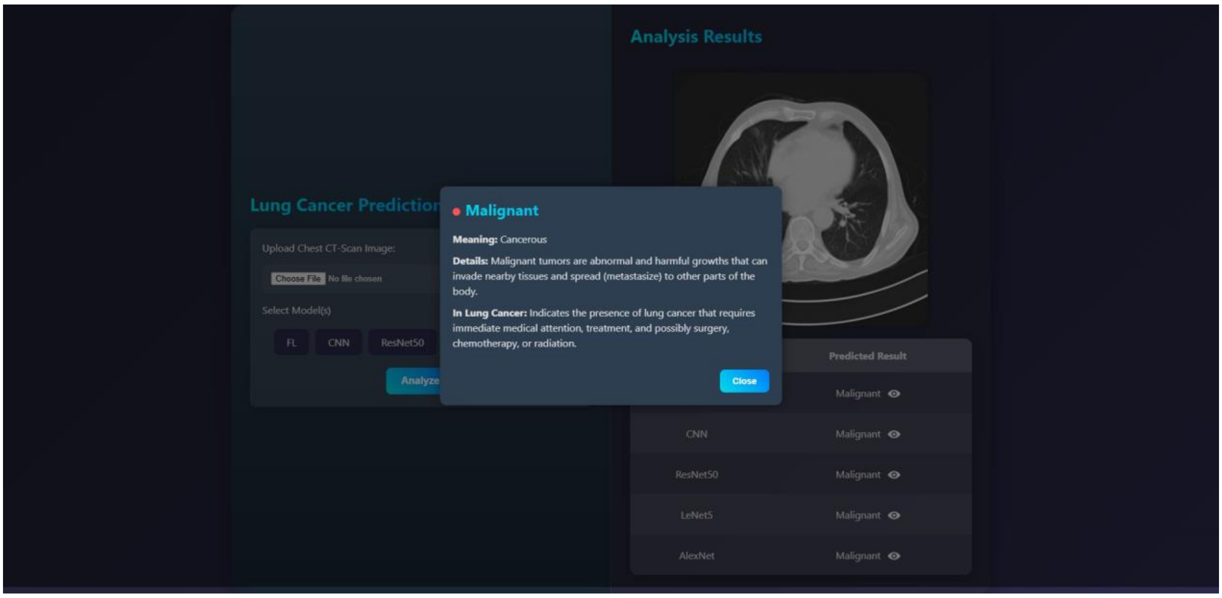

Fig 3.4: User Interface of Lung Cancer Classification System

3.2 Detailed Methodology and Design

The proposed methodology combines data preprocessing with both federated and centralized training along with aggregation algorithms and XAI implementation and performance evaluation and web application creation. The system chose a solution that meets the requirements of privacy protection alongside performance optimization along with interpretability and ease of use for its user base.

3.2.1 Data Collection and Pre-processing

For this study, the data was collected from Kaggle and its IQ-OTH/NCCD lung cancer repository, compiled by the IQ-OTH/NCCD from the Iraq-Oncology Teaching Hospital during three months of the Fall of 2019. This dataset contains 1190 CT scans from 110 patients divided into three classes: malignant (forty cases), benign (fifteen cases), and normal (fifty-five cases). The CT scans were obtained from a Siemens SOMATOM scanner and were archived in DICOM format. The scanning parameters used were: 120kV energy level, 1mm slice thickness, 350-1200HU window width, 50-600 HU window center, and breath-hold at full inspiration. Annotated by oncologists and radiologists.

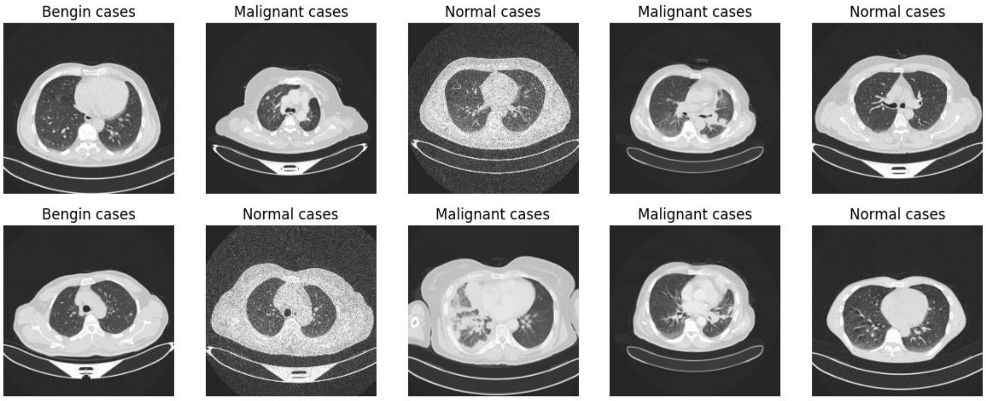

Fig 3.5: IQ-OTH/NCCD Lung Cancer Dataset Sample Images

This research uses multiple phases to develop a ready-to-use federated learning model that can accurately identify lung cancer. Pre-processing follows training and evaluation in addition to aggregation strategy comparison. The model requires this step throughout its development process to achieve better performance and protect privacy alongside delivering efficient distributed learning operations.

Pre-processing: An important step is pre-processing of input image to standardize the inputs and make the model generalize well for different dataset. Thus the multiple transformations of the raw images include.
Resizing: All images are resized to 224 × 224 pixels to keep consistent with different clients and for compatibility with the deep learning model.
Normalization: This helps with the training and getting a convergent model as normalization: each pixel value is divided by 255, scaling to [0,1].
Augmentation: Augmentation techniques are applied to the data as such medical datasets tend to be small, meaning limited data, and imbalanced meaning the distribution of positive data to negative data is highly skewed. The augmentation process includes:

- Rotation: Random rotation of images around within 10 degrees to improve stability to variations in orientation.
- Translation: Translating an image by shifting it up to 10% and simulating other perspectives.
- Horizontal Flipping: An implementation of introducing variability in the lung cancer patterns through flipping images.
- Controlled Blurring Effects (Gaussian Blur): These are introduced to imitate variations in image quality because of medical imaging devices.

Each class of the dataset is augmented by using the Albumentations library to implement the augmentation, and all classes are kept so that there will be > 300 samples for each class to balance the dataset and reduce model instability.

3.2.2 Federated Learning Framework

This study adopts a federated learning (FL) framework to enable privacy-preserving lung cancer classification. In this approach, raw medical data remains localized within each client (e.g., local hospitals or healthcare institutions), ensuring compliance with medical privacy regulations such as HIPAA and GDPR. Instead of sharing data, clients independently train machine learning models on their private datasets. Only the locally trained model parameters (weights) are transmitted to a centralized server for aggregation. The federated learning process in this study follows the standard client-server architecture and can be summarized into several main phases:

- **Initialization:** The central server initializes the global model and prepares the dataset. During preparation, data augmentation techniques are applied to enhance the diversity and size of the dataset. Additionally, the data is partitioned into Independent and Identically Distributed (IID) and Non-Independent and Identically Distributed (non-IID) subsets to simulate different client data distributions.
- **Client Selection:** In each communication round, a subset of clients is randomly selected to participate. This simulates real-world federated environments where client availability may vary.
- **Local Training:** Selected clients receive the current global model and perform local training using their own datasets. Clients typically utilize optimization algorithms such as Stochastic Gradient Descent (SGD) or Adam for local updates.
-

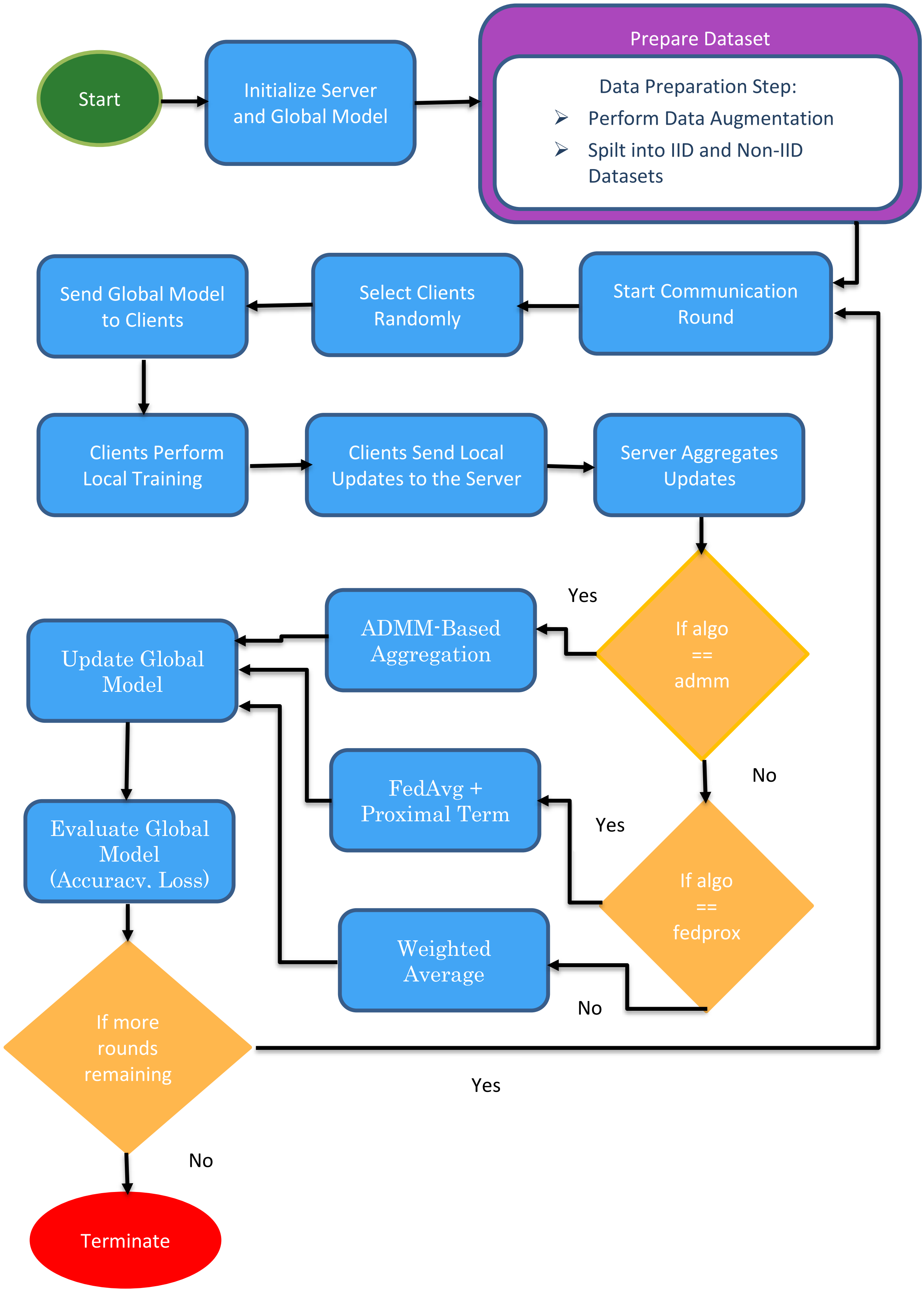

Fig 3.5: Federated Learning Workflow

- **Communication and Aggregation:** After local training, clients send their updated model parameters back to the central server. The server aggregates the local updates using different strategies based on the selected algorithm.
- **Global Model Update and Evaluation:** The aggregated model parameters are used to update the global model. After each round, the global model is evaluated based on classification accuracy and loss.
- **Termination:** The process repeats for a predefined number of communication rounds or until convergence is achieved.

The entire workflow of the federated learning framework is illustrated in Figure 3.6, with major steps highlighted, including dataset preparation (augmentation and data splitting), client selection, local training, aggregation strategies, and global evaluation.

Three different aggregation strategies for combining the locally trained models to a global model are studied.
Federated Averaging (FedAvg): The most widely used aggregation method for federated learning is FedAvg. [2] In this approach:

- In each client, a local model is trained on the dataset for a fixed number of epochs.
- All the model parameters (weights and biases) made after training are sent to the central server.
- These parameters are subsequently weighted based on the number of data samples available for each client and the server computes a weighted average of them.
- Afterwards, the averaged model is distributed back to the clients for the next round of the training.

Federated Proximal (FedProx): FedProx is a variant of FedAvg, by adding a regularization term to the process of local training. [3] The term prevents drastic change in the model updates and stabilizes training when client data distributions are not IID and useful when the distribution of the clients is known. It is Same as FedAvg, but during local training, clients optimize the following objective:

$$\min h_k(w; w^t) = F_k(w) + \frac{\mu}{2} \| w - w^t \|^2$$

where:

- $F_k(w)$ is the local loss function of client
- w^t is the global model from the previous round.
- μ is a hyperparameter controlling the strength of the proximal term.

Federated Adaptive ADMM (FedAdmm): The more advanced optimization method used to augment federated learning is actually using Alternating Direction Method of Multipliers (ADMM) as FedAdmm. [4] It presents an adaptive update to domain balance convergence and computation time.
Key Features of FedAdmm:

- FedAdmm solves an optimization problem instead of averaging model weights like FedAvg, in which the best mix of local and global updates is found.
- In particular, it uses dual variables to better control the learning process and experience better performance on highly heterogeneous environments (extreme non-IID data).
- This helps the adaptive learning rate adapt itself based on each client's data distribution and leading to more stable training.

3.2.3 Centralized Training

Centralized training is conducted using CNN, LeNet5, ResNet50, and AlexNet models on the full IQ-OTH/NCCD lung cancer dataset to establish a baseline for comparison with federated learning strategies. The centralized approach aims to evaluate key performance metrics, including accuracy, precision, recall, F1-score, and computational efficiency, such as training time and resource utilization.

3.2.3.1 Proposed CNN Model Architecture

The custom Convolutional Neural Network (CNN) implemented in this study combines high accuracy performance with straightforward computational processes. The architectural specifications appear in Table 3.1. The CNN implements two convolutional layers containing 5×5 kernels, followed by an average pooling technique that minimizes spatial dimensions while keeping vital characteristics. The model applies two dense layers with ReLU non-linear transformations to one-dimensional vectors obtained by flattening feature maps. A fully connected activation layer implementing softmax function determines the probability distribution of three lung cancer categories in the final network output. The design of this architecture constitutes an efficient lightweight solution which functions well within centralized and federated learning frameworks.

Table 3.1: Proposed CNN Model Architecture

Layer	Description	Output Shape
Input	Input image	(3, 224, 224)
Convolution 2D + ReLU	32 filters, 5×5 kernel	(32, 220, 220)
Average Pooling 2D	2×2 pooling	(32, 110, 110)
Convolution 2D + ReLU	64 filters, 5×5 kernel	(64, 106, 106)
Average Pooling 2D	2×2 pooling	(64, 53, 53)
Flatten	Converts 2D feature maps to 1D vector	(179,776)
Fully Connected + ReLU	Dense layer with 64 neurons	(64)
Fully Connected + ReLU	Dense layer with 32 neurons	(32)

3.2.3.2 Proposed LeNet5 Model Architecture

Initially designed for handwritten digit classification, the LeNet5 model has been modified in this work for lung cancer classification employing the IQ-OTH/NCCD dataset. The design is changed to manage three output classes and three input channels—RGB images. Table 3.2 shows all the architectural details as a whole. Three convolutional layers interleaved with two max-pooling layers make up this model, progressively lowering spatial dimensions while raising feature map depth. The output is flattened following the convolutional blocks and passed through two fully connected layers; the last layer generates class probabilities.The design guarantees good feature extraction capacity for lung cancer detection and preserves a balance between complexity and efficiency. Hence, it is appropriate for centralized training.

Table 3.2: Proposed LeNet5 Model Architecture

Layer	Description	Output Shape
Input	Input image	(3, 224, 224)
Convolution 2D + ReLU	6 filters, 5×5 kernel	(6, 220, 220)
Max Pooling 2D	2×2 pooling	(6, 110, 110)
Convolution 2D + ReLU	16 filters, 5×5 kernel	(16, 106, 106)
Max Pooling 2D	2×2 pooling	(16, 53, 53)
Convolution 2D + ReLU	120 filters, 5×5 kernel	(120, 49, 49)
Flatten	Converts 2D feature maps to 1D	(120 × 49 × 49 = 288,120)
Fully Connected + ReLU	Dense layer with 84 neurons	(84)
Fully Connected	Dense layer with 3 neurons (three lung cancer types)	(3)

3.2.3.3 Proposed AlexNet Model Architecture

The research adapted AlexNet to perform three-class output classification for lung cancer applications. The modifications involve transforming the final fully connected layer to generate probabilities for the three classification classes. The network details appear in Table 3.3. The model architecture contains five convolutional layers that use ReLU activation functions and max-pooling layers to extract complex features, simultaneously reducing image dimensions. The model completes its operation with three fully connected layers that use extracted features for classification tasks. The more profound architecture of AlexNet allows the network to extract higher levels of features that work efficiently for high-dimensional image datasets, including IQ-OTH/NCCD.

Table 3.3: Proposed AlexNet Model Architecture

Layer	Description	Output Shape
Input	Input image	(3, 224, 224)
Convolution 2D + ReLU	64 filters, 11×11 kernel, stride 4, padding 2	(64, 55, 55)

Layer	Description	Output Shape
Max Pooling 2D	3×3 pooling, stride 2	(64, 27, 27)
Convolution 2D + ReLU	192 filters, 5×5 kernel, padding 2	(192, 27, 27)
Max Pooling 2D	3×3 pooling, stride 2	(192, 13, 13)
Convolution 2D + ReLU	384 filters, 3×3 kernel, padding 1	(384, 13, 13)
Convolution 2D + ReLU	256 filters, 3×3 kernel, padding 1	(256, 13, 13)
Convolution 2D + ReLU	256 filters, 3×3 kernel, padding 1	(256, 13, 13)
Max Pooling 2D	3×3 pooling, stride 2	(256, 6, 6)
Flatten	Converts 2D feature maps to 1D	(256 × 6 × 6 = 9216)
Fully Connected + ReLU	Dense layer with 4096 neurons	(4096)
Fully Connected + ReLU	Dense layer with 4096 neurons	(4096)
Fully Connected	Dense layer with 3 neurons (three lung cancer types)	(3)

3.2.3.4 Proposed ResNet50 Model Architecture

The research incorporates the ResNet50 model as the framework for classifying lung cancers by analyzing IQ-OTH/NCCD dataset images. The model architecture receives three RGB image inputs with three class output labels. The technical specifications of the whole system appear in Table 3.4. A deep learning architecture uses the residual framework to deliver results through its blocks with two convolutional layers, batch normalization and ReLU activations. The model utilizes 1×1 convolutions as its downsampling technique across different stages. The network performs several residual stages and applies global average pooling, after which a fully connected layer generates the class predictions. The design includes deep feature extraction capabilities and gradient problem resolution components that make it usable for centralized training systems and accurate lung cancer classification.

Table 3.4: Proposed ResNet50 Model Architecture

Layer	Description	Output Shape
Input	Input image	(3, 224, 224)
Convolution 2D + BatchNorm + ReLU	64 filters, 7×7 kernel, stride 2	(64, 112, 112)

Layer	Description	Output Shape
Max Pooling 2D	3×3 pooling, stride 2	(64, 56, 56)
Residual Block ×3	Two 3×3 convolutions, 256 filters	(256, 56, 56)
Residual Block ×6	Two 3×3 convolutions, 256 filters	(256, 28, 28)
Residual Block ×3	Two 3×3 convolutions, 512 filters (first block downsampling with 1×1 conv, stride 2)	(512, 14, 14)
Global Average Pooling	Adaptive pooling to (1,1)	(512, 1, 1)
Flatten	Converts feature maps to 1D	(512)
Fully Connected	Dense layer with 3 neurons (three lung cancer types)	(3)

3.2.4 Explainable AI Integration

The framework includes Explainable Artificial Intelligence (XAI) techniques, which enhance the interpretability of deep learning models built for lung cancer classification. The research implemented Grad-CAM as a visualization technique to produce explanations that show important image areas that influence model prediction outcomes. The implementation of Grad-CAM creates a dual benefit by increasing doctors' trust in model decisions and enabling them to compare predicted target areas against their clinical understanding, thus improving the integration of models into the healthcare workflow. The visualization tool highlights those areas inside an input picture that the prediction model references most during its decision-making process. This research applied Grad-CAM to examine the attention areas models used to detect lung cancer during classification. Identifying crucial input regions through Grad-CAM helps researchers evaluate if the model attention points toward meaningful clinical features such as tumor locations and healthy tissue areas. The research utilized Grad-CAM consistently between CNN, LeNet5, AlexNet, and ResNet50 model architecture to produce equivalent visualization results. Implementation followed these procedures during the process:

Model and Layer Selection: Pre-trained versions of each model were loaded using the weights obtained from centralized training. The target convolutional layers selected for Grad-CAM were as follows:

Table 3.5: Selected model layers for XAI

Model	**Selected Layer**
CNN	Second convolutional layer
LeNet5	Third convolutional layer
AlexNet	Final ReLU layer following the last convolutional layer
ResNet50	Final layer of the fourth residual block

Image Preprocessing: A single CT image served as the representative choice from the IQ-OTH/NCCD database for model evaluation. Images were transformed into 224×224 pixels resolution before being converted to RGB format and adding a batch dimension for tensor-based model input.

Grad-CAM Heatmap Generation: The Grad-CAM class was initialized with the selected model and target layer. Forward and backward hooks were registered to capture the activations and gradients during inference. The procedure involved:

- Performing a forward pass to obtain the model's prediction.
- Computing the loss with respect to the predicted class.
- Executing a backward pass to compute gradients of the loss relative to the target layer's activations.
- Pooling the gradients spatially and using them to weight the activations.
- Generating the heatmap by aggregating the weighted activations, followed by normalization to the [0, 1] range.

Heatmap Visualization: The heatmap achieved dimension compatibility with the initial image through resizing, and then it received its color mapping with JET visualization. Partial transparency was applied to the heatmap before overlaying the original CT image as a composite visualization. The photos featuring origin CT and Grad-CAM visuals appeared for comparative study purposes.

3.2.5 Web Application Development

The web application implemented for the system deployment enabled clinicians to use the trained lung cancer classification models through an interactive interface for inference purposes. The web application enables users to submit CT scan images while displaying diagnosis results as malignant, benign or normal together with potential disease suggestions from prediction evaluations.

The development of the web application utilized the following technologies:

- HTML: Used to structure the layout and elements of the web interface.
- CSS: Applied to style the web pages, ensuring an intuitive and user-friendly design for ease of navigation.
- JavaScript: Employed for client-side interactivity, including image upload handling and dynamic content updates.
- Django: A Python-based web framework used to implement backend functionalities, including model integration, request handling, and database management.

The key features incorporated into the web application include:

- CT Image Upload: Clinicians can securely upload CT scan images through the web interface for analysis by the deployed models.
- Classification Result Display with Suggestions: The application provides the classification result (malignant, benign, or normal) along with relevant suggestions for possible associated diseases based on the model's prediction.

This web application ensures that the developed deep learning models are easily accessible for clinical use, enhancing the practicality and usability of the system by offering a simple and effective interface for AI-assisted lung cancer diagnosis.

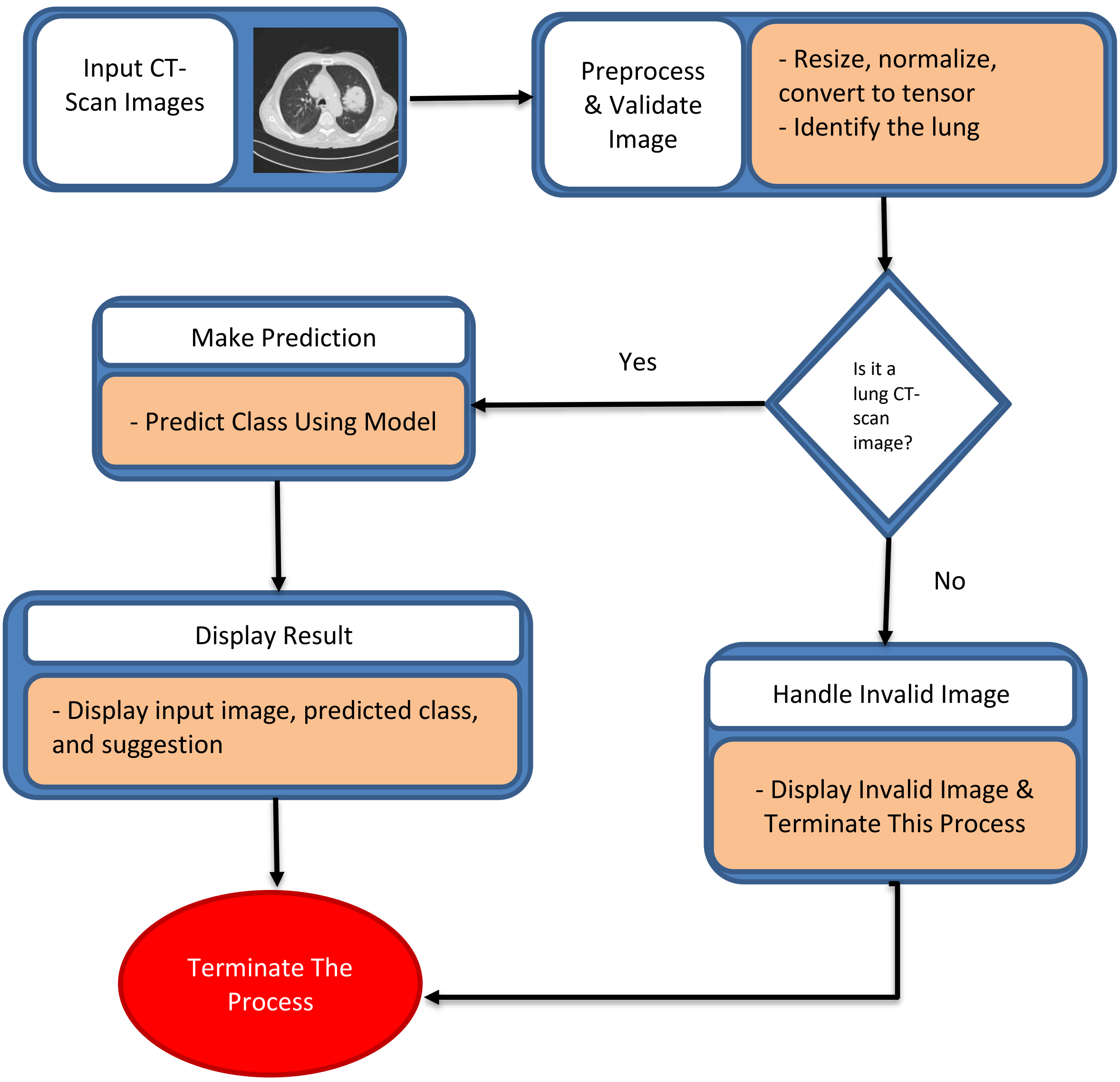

Fig 3.6: Web Application Workflow

Figure 3.6 presents the web application workflow detailing the steps for image processing and prediction delivery to users. The user uploads CT-scan images through the web interface as their first step. An image preprocessing process begins by resizing 224x224 pixels while normalizing pixel values before converting it into a tensor format for machine learning model consumption. Before proceeding, the system checks whether the submitted image represents a lung CT scan after image preprocessing. After validating an image's validity, the workflow makes predictive assessments from trained models about image categories, including malignant, benign, and normal, and model output generates a clinical evaluation with predicted results that present to users and diagnostic findings. An invalid image error appears when users upload data that fails to identify as a lung CT scan through the application with the message "Invalid Image: Please provide a lung CT-scan image." The application process ends by displaying either the classification findings and recommendation or an invalid image warning for a proper conclusion of user interaction. The application workflow lets the system efficiently process lung CT-scan images to display relevant and valuable information to users.

3.2.5 Alternative Solution Considered

In the design and development of the proposed lung cancer classification framework, several alternative approaches were critically evaluated. Each alternative was analyzed in terms of its potential benefits, drawbacks, and alignment with the goals of privacy preservation, performance optimization, clinical relevance, and system accessibility. The final selection was made based on a comprehensive assessment of these factors.

Centralized Learning Only:

Advantages:

- Straightforward implementation with direct access to the entire dataset.
- Potential for higher predictive performance due to complete data availability.

Limitations:

- Significant risk of violating privacy regulations such as HIPAA and GDPR.
- Reduced data diversity, which can limit model generalizability across different populations and healthcare settings.

Reason for Exclusion:

- Centralized learning was not adopted due to critical privacy concerns and the need to support distributed data environments typical of clinical institutions.

Transfer Learning Using Pretrained Models (e.g., VGG16, InceptionV3)

Advantages:

- Accelerated training through the utilization of pretrained feature representations.
- Reduces the demand for large labeled datasets.

Limitations:

- Extensive fine-tuning required to adapt general-purpose pretrained models to the nuances of medical imaging.
- Pretrained models may not capture domain-specific features essential for accurate lung cancer classification.

Reason for Exclusion:

- Transfer learning approaches were set aside in favor of custom-designed CNN architectures (LeNet5, ResNet50, and AlexNet), which provided better adaptability and performance specifically tuned to lung cancer CT image analysis.

Single Aggregation Strategy (FedAvg Only)

Advantages:

- Simplified federated learning implementation.
- Lower computational overhead and resource requirements.

Limitations:

- Reduced robustness in non-IID (non-Independent and Identically Distributed) data environments.
- Potentially slower convergence and lower model accuracy in heterogeneous client scenarios.

Reason for Exclusion:

- To ensure adaptability across varying data distributions, a combination of aggregation strategies (FedAvg, FedProx, and FedADMM) was preferred to provide a more comprehensive and resilient evaluation.

Standalone Application Instead of Web Application

Advantages:

- Faster execution without dependency on internet connectivity.
- Potentially improved system responsiveness for localized use.

Limitations:

- Restricted accessibility across different clinical sites.
- More complex installation, configuration, and maintenance processes for end users.

Reason for Exclusion:

- A standalone application was not pursued to prioritize the development of a web-based application, ensuring broader accessibility, ease of deployment, centralized updates, and better user experience for clinical users.

Final Selected Approach

A federated learning framework with FedAvg, FedProx, and FedADMM aggregation strategies proved optimal for the final system implementation and centralized baseline tests. The system incorporated two explainable AI methods, Grad-CAM and SHAP, to improve understanding and transparency. The Django-based web platform delivers user-friendly, secure model access and visual explanation features for clinical implementation. The implementation offers optimal control over privacy while maintaining performance capabilities with interpretability features and accessibility options, which make it appropriate for real-world lung cancer diagnosis and clinical decision support systems.

3.3 Project Plan

The project is structured into phases with specific milestones and timelines, spanning six months:

Table 3.6: Project Plan of Lung Cancer Classification System

Step	Task	Duration	Timeline
Step 1: Requirement Analysis	Data collection, requirement specification, literature review	1 month	Month 1
Step 2: Data Preprocessing	Dataset acquisition, resizing, normalization, augmentation	1 month	Month 2
Step 3: Model Development	Implement federated learning (CNN, ResNet50, LeNet5), centralized training (CNN, LeNet5, ResNet50, AlexNet), aggregation strategies	3 months	Months 3-5
Step 4: XAI Integration	Implement Grad-CAM for interpretability	0.5 month	Month 5.5
Step 5: Web Application Development	Develop Django-based web interface with HTML, CSS, JavaScript	1 month	Month 6.5
Step 6: Evaluation and Testing	Compute metrics, visualize results, test web application	3.5 month	Month 6.5-10

3.4 Task Allocation

To ensure efficient project execution and timely completion, tasks were systematically allocated among team members based on their individual strengths, technical expertise, and interest areas. The division of responsibilities was carefully aligned with the project phases described in the project plan (Section 3.3), enabling collaborative development and parallel progress across modules. Each member contributed to the core functionality of the system, including federated learning, centralized model training, web deployment, and explainable AI integration.

Team Member 1 (Model Architect, Federated Learning Lead & Web Developer)

- Designed and implemented the federated learning framework using CNN, ResNet50 and LeNet5 models.
- Configured client-server communication and integrated aggregation strategies including FedAvg, FedProx, and FedADMM.
- Conducted experiments under both IID and non-IID data distributions, analyzing convergence behavior and tuning training hyperparameters.
- Developed the Django-based web application for deploying trained models and managing user interaction.
- Built the front-end interface using HTML, CSS, and JavaScript, including functionalities for image upload, result display, and Grad-CAM visual overlays.
- Contributed equally to report writing, specifically focusing on the methodology (federated learning setup, model architectures), web deployment, and engineering design chapters.

Team Member 2 (Centralized Training Lead, XAI Developer & Documentation Coordinator)

- Developed centralized models (CNN, LeNet5, ResNet50, AlexNet) using the full IQ-OTH/NCCD dataset as a performance baseline.
- Integrated Grad-CAM across all models to generate interpretable heatmaps for clinical decision support.
- Performed quantitative evaluations (accuracy, precision, recall, F1-score), and created visual aids like confusion matrices and training plots.
- Conducted final testing and validation of both centralized and federated frameworks to ensure robustness and accuracy.
- Contributed equally to report writing, with a focus on literature review, experimental results and discussion, explainable AI, and conclusions.

This structured task allocation ensured that all critical components of the system—from model development to user interface—were addressed efficiently, promoting accountability and cohesive teamwork throughout the ten-month development cycle.

3.5 Summary

The chapter explained the development of a private federated learning framework for lung cancer diagnosis using the IQ-OTH/NCCD dataset. The technique unites federated and centralized training models for a performance and generalization comparison. Project implementation involved ResNet50 and LeNet5 running on four separate clients while model parameters transmitted to a central server from each client for training purposes. The analysis of data heterogeneity employed three aggregation methods known as FedAvg, FedProx and FedADMM under both IID and non-IID conditions. The baseline performance assessment included a centralized training process utilizing CNN along with LeNet5 and ResNet50 and AlexNet that operated on the entire dataset. The implementation of Grad-CAM on all models created visual heatmaps which displayed prediction-related regions that improved transparency and clinical reliability of the system. nostic predictions for malignant, benign, or normal tumor status accompanied with model explanations became available through a Django-based web application which also supported CT scan uploads for clinical users. Our evaluation excluded different approaches including centralized-only learning as well as pretrained models together with standalone applications because we had privacy, adaptability, and accessibility concerns. The adopted approach provides system scalability and maintains data compliance with healthcare regulations while keeping explanations clear for establishing a clinically relevant diagnostic assistance tool.

Chapter 4
Implementation and Results

The chapter explains how the lung cancer classification framework was implemented in federated learning while discussing environment setup along with testing and evaluation methods along with result analysis and finding explanations. This section shows the evaluation outcome of centralized and federated learning models as well as XAI insights and web application functionality.

4.1 Environment Setup

The implementation occurred on Google Colab, which allowed efficient model training through its high-performance GPUs alongside the benefit of operating without hardware limitations at the local stage. The implemented environment used Python 3.8 as its core language. At the same time, PyTorch 1.12 served as the main framework for model development, and the system could adapt either framework depending on the project requirements. Several additional libraries were integrated to enhance the project's functionality: OpenCV was used for advanced image processing tasks like reading, resizing, and enhancing CT images; Albumentations provided powerful image augmentation capabilities such as rotation, flipping, and brightness adjustment to improve model generalization and combat class imbalance; Scikit-learn was employed to calculate various evaluation metrics, including accuracy, precision, recall, F1-score, and confusion matrices, while Matplotlib and Seaborn were utilized for plotting visualizations, such as training curves and confusion matrices. The research used the IQ-OTH/NCCD lung cancer dataset obtained from Kaggle that identifies CT images as malignant, benign, or usual. The Colab environment executed multiple preprocessing operations, which included a 224x224 pixels resize of CT images pixel normalization to a 0-1 scale and additional dataset augmentation with a 300-sample minimum per class.

Implementing the pipeline within federated learning started from scratch because it did not use external libraries for federated learning control. This approach enabled direct management of all individual components. The data split included IID and non-IID distributions intended to replicate data heterogeneity patterns. Four simulated clients operated their local model training on particular parts of the data partition while performing set epochs before they shared updates with the central server. The server operated the client update aggregation through the execution of three aggregation methods, including FedAvg, FedProx, and FedADMM. The standard federated averaging algorithm FedAvg executed a weighted average calculation of client models using the dataset sizes of each participant. FedProx improved upon FedAvg by adding proximal terms to local objectives to solve heterogeneity problems between different clients. The advanced strategy FedADMM combined the power of the Alternating Direction Method of Multipliers (ADMM) to improve convergence and model performance, mainly when operating in highly non-IID environments. The standard Python and TensorFlow with PyTorch functionalities enabled developers to view deep internal workings of federated learning algorithms yet maintain flexibility for future adaptations.

4.2 Testing and Evaluation/Performance/ Comparative Analysis

4.2.1 Testing Methodology

To evaluate performance of the federated learning based lung cancer classification model, multiple performance metrics and visualization techniques are used to ascertain effectiveness of federated learning approach. These help in determining the reliability of the model in real world medical applications using unseen data.

Evaluation Metrics: To measure the model's classification performance, the following key metrics are computed:

- Accuracy: Measures the overall correctness of predictions, calculated as the ratio of correctly classified instances to the total number of instances.
- Precision: Represents the proportion of correctly predicted positive cases out of all predicted positive cases, important for reducing false positives.
- Recall (Sensitivity): Measures the model's ability to correctly identify positive cases, essential in medical diagnosis to minimize false negatives.
- F1-score: A harmonic mean of precision and recall, providing a balanced evaluation of model performance, especially when dealing with class imbalances. These metrics offer a different point of view of the model effectiveness, helping in understanding the model effectiveness more involved than just accuracy alone.

Each of these metrics provides a different perspective on the model's effectiveness, allowing for a more nuanced analysis beyond accuracy alone[14].

Training Convergence and Stability: The trend in loss and accuracy are monitored over multiple training rounds in order for the model to learn stably and avoid overfitting. This analysis helps determine:

- Convergence: How close the solution is to the target solution for the problem at hand.
- overfitting/underfitting: If the training as well as validation accuracy deviates significantly, this may suggest the use of a model that's memorizing so much training data (overfitting) or totally not learning enough (underfitting).
- Impact on Aggregation Strategies: Convergence rate and accuracy of the model over the underlying aggregation methods of federated learning (FedAvg, FedProx, FedAdmm).

Visualization Techniques for Model Insights: Visualization tools such as training curves plot accuracy vs. training rounds and loss vs. training rounds, helping to diagnose model stability issues and optimize hyperparameters accordingly.
For this, several visualization techniques are used to interpret the model's decision making and validate its effectiveness.

- **Confusion Matrix:** Its results display number of true positives, true negatives, false positives and false negatives, which help to reveal misclassification patterns.
- **Training Curves:** Since learning dynamics tend to have hard to interpret shapes, we plot graphs of accuracy and loss trend over training rounds for an analysis of their shapes, and to detect overfitting/underfitting.

This study combines the use of these evaluation techniques to provide a comprehensive performance analysis of the federated learning based lung cancer classification model for eventual application of such model in real life clinical settings while guaranteeing the models' efficiency and privacy.

4.2.2 Comparative Analysis

The research measured the performance output between centralized and federated learning models to determine their operational effectiveness. A foundation was established by centralized training yet the evaluation of federated learning performed through the analysis of FedAvg, FedProx, and FedADMM strategies under both IID and non-IID conditions. A test of the web application verified its capabilities to show the correct outcomes and visual XAI explanations without glitches.

4.3 Result and Discussion

4.3.1 Centralized Training Results

Different model performances in centralized training are presented in Table 4.1. The evaluation metrics contain both macro and weighted average levels of precision, recall and F1-score as well as accuracy scores from training, testing and validation sets. Each model architecture exhibits distinct value and challenges when used for CT scan image-based lung cancer classification.

Table 4.1: Centralized Training Performance

Model	Precision		Recall		F-Score		Train	Test	Valid
	Macro	Weighted	Macro	Weighted	Macro	Weighted			
CNN	0.95	0.95	0.93	0.95	0.94	0.94	98.32	95	96.88
LeNet5	0.97	0.98	0.97	0.98	0.97	0.98	99.66	98	99.22
ResNet50	0.99	0.99	0.98	0.99	0.99	0.99	100	99	99
AlexNet	0.98	0.98	0.98	0.98	0.98	0.98	99.66	98	96.88

The centralized training results demonstrated that ResNet50 delivered peak performance by reaching 99% test accuracy for both test and validation data while generating perfect F1-score values (0.99) and precision and recall scores. These outstanding results proved its superiority in accurately classifying CT images between malignant, benign, and normal categories. The deep network architecture containing residual connections in ResNet50 delivers superior performance through its effective solution to the vanishing gradient issue while helping recognize intricate patterns. The test accuracy performance of LeNet5 and AlexNet reached 98%, but ResNet50 showed slightly better results because its optimized architecture surpassed them. LeNet5 performed well through its efficiency but fell behind ResNet50 because of its basic architecture, while AlexNet achieved better extraction than LeNet5, yet failed to match ResNet50 because of its structure. A custom CNN model was performed at a 95% test accuracy level while demonstrating good operational stability through its compact design. However, its restricted depth limited its potential to extract complex features as effectively as deeper frameworks. All models displayed validation accuracy levels similar to their test accuracy results, showing excellent data generalizability without significant overfitting. ResNet50 achieved the best performance, but LeNet5, AlexNet, and the custom CNN displayed stable robustness, validating the reliability of model development procedures.

4.3.2 CNN Model Performance Analysis in Centralized Training

The personalized CNN model underwent 50 epochs of training to have the ability to categorize lung CT images as malignant and benign configurations and standard images. Performance metrics of the model reached high levels because training values stabilized throughout learning sessions. Training began with high values for loss measurements and low accuracy rates during the first training period. During the first epoch, the training method delivered a training loss result of 0.9866 and 52.97% training precision but also produced a validation loss measurement of 0.8188, which delivered 72.66% validation precision. Subsequent epochs brought about fast performance gains in the model. At epoch 5, both model train accuracy reached 95.63%, and validation accuracy reached 92.97%, demonstrating quick learning for the network. After epoch 10, the CNN began displaying improved but steady performance. The training accuracy stabilized at 98.32%, while validation accuracy demonstrated stable performance between 96% and 98.44% during the last training epochs. During the training process, the loss value decreased progressively until it reached a stable point at 0.568, while validation loss approached 0.5669 during its stable phase. The accuracy and loss trends for each epoch appear in Figure 4.1, while Figure 4.2 contains loss trends. The convergence patterns show a well-behaved performance and minimal overfitting because the training and validation curves stay closely aligned. The CNN model reached computational efficiency within 1.5 minutes during its training period to achieve high-performance outcomes.

A confusion matrix allowed the evaluation of how well the model classified different categories of data. The confusion matrix (Figure 4.3) shows the CNN model displayed excellent true positive detection for all classes alongside low numbers of misclassified scans which verifies its dependable operation in categorizing lung CT scans..

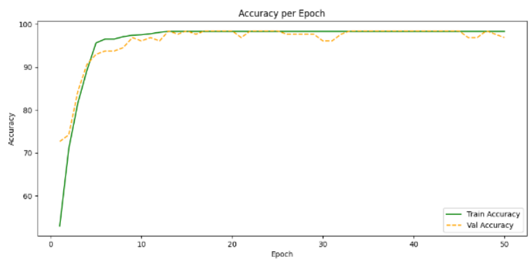

Fig 4.1: CNN Training and Validation
Accuracy per Epoch

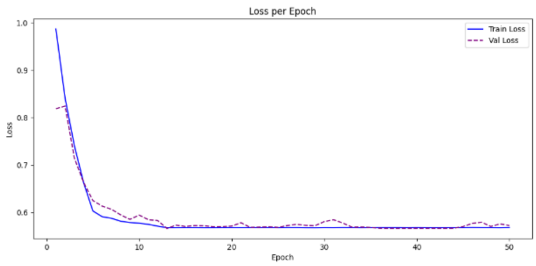

Fig 4.2: CNN Training and Validation Loss per Epoch

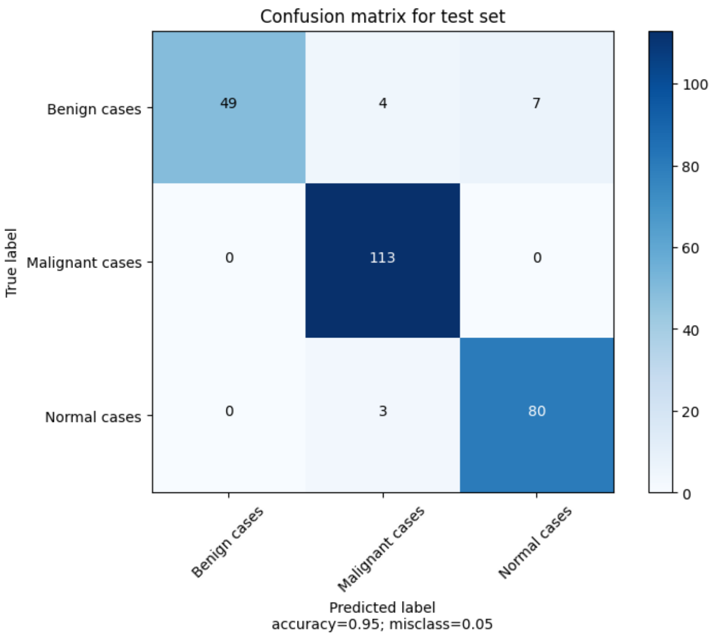

Figure 4.3: CNN Confusion Matrix on Test Data

4.3.3 LeNet5 Model Performance Analysis in Centralized Training

The LeNet-5 model trained over 50 epochs through which Figure 4.4 and Figure 4.5 showed the monitoring of training and validation accuracy and loss values. During initial training, the model displayed weak performance because the training accuracy was rated at 45.80%, and validation accuracy reached 60.94%. Meanwhile, training loss remained at 1.3002, and validation loss settled at 0.8763. Training continued over several epochs until the accuracy steadily improved while loss figures diminished considerably. Morning reached a training accuracy point of 90.15% during epoch 5. When parallel to this, it displayed a validation accuracy of 87.50%, demonstrating quick convergence. The model reached 98.10% training accuracy during its tenth epoch alongside 97.66% validation accuracy and training loss at 0.0541 and validation loss at 0.0943. Throughout the stage after epoch 20, the model achieved stable training accuracy results at over 99% and validation accuracy at 99.22%, with the training and validation losses maintaining low levels between 0.003 and 0.04.

During training, the curves showed no signs of overfitting because they maintained consistent alignment throughout the entire process. The last training epoch (epoch 50) produced 99.66% accuracy during training and 99.22% accuracy on validation data with 0.0088 training loss and a validation loss of 0.0303. The training duration lasted about 1.164 minutes overall. The generated confusion matrix helped evaluate the model's decision-making performance across all classification categories. The LeNet-5 model demonstrated a high precision rate according to the confusion matrix (Figure 4.6), which showed minimal errors in each class identification process, thus proving its strong reliability in lung CT scan identification tasks.

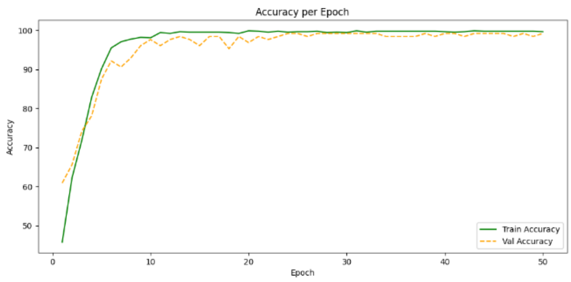

Fig 4.4: LeNet5 Training and Validation Accuracy per Epoch

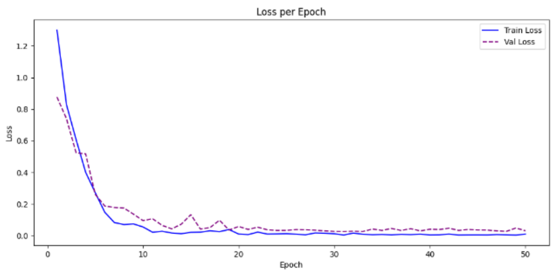

Fig 4.5: LeNet5 Training and Validation Loss per Epoch

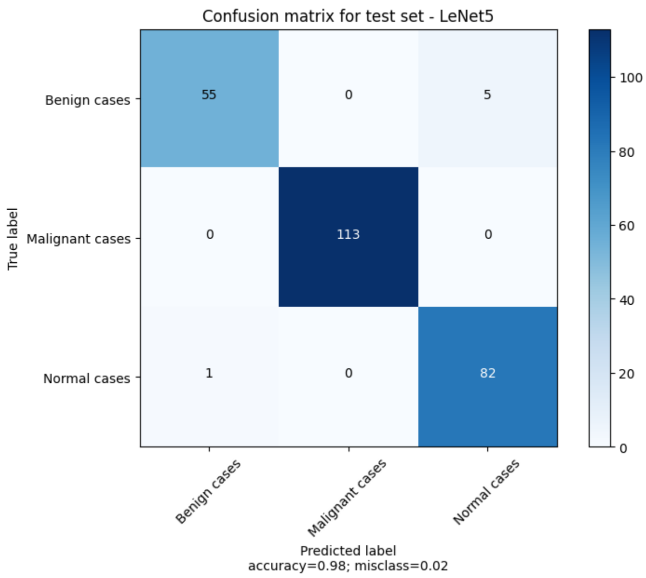

Fig 4.6: LeNet5 Confusion Matrix on Test Data

4.3.4 AlexNet Model Performance Analysis in Centralized Training

The training of AlexNet took place over 50 epochs based on figures 4.7 and 4.8 which showed accuracy and loss development patterns during the process. The model started with minimal success by reaching only 40.09% training accuracy together with 43.75% validation accuracy during its first epoch of operation. The model demonstrated limited performance changes during the first epochs until training accuracy hit 47.93% at epoch 5 while validation accuracy remained unpredictable. The training accuracy increased to 54.87% while validation accuracy measured 54.69% starting from epoch 8 onward. The performance of AlexNet became substantially better during epoch 10 which produced training accuracy of 67.75% and validation accuracy of 75.78%. The model achieved 79.96% training accuracy which was accompanied by 86.72% validation accuracy during epoch 18. Rapid advancement occurred at epoch 21 as the training accuracy attained 90.71% while validation accuracy achieved 96.88%. The model showed a momentary drop at epoch 22 after which it restored itself to reach 94.40% training accuracy with 92.97% validation accuracy at epoch 29. The model exhibited steady and high performing results in its later development by achieving 97.54% training accuracy and 99.22% validation accuracy during epoch 35. The training accuracy of AlexNet at epoch 50 reached 99.66% and its validation accuracy reached 96.88% simultaneously alongside minimal training and validation loss values. During the training procedure overfitting remained minimal because training and validation accuracy maintained similar alignment patterns throughout. The model needed 1.194 minutes to complete its training process which implies an effective learning regimen for lung CT scan classification. A confusion matrix presented in Figure 4.9 demonstrated excellent true positive results across all classes. The model correctly identified all cases except a limited number of incorrect classifications because it learned discriminative features effectively to detect lung cancer. This established robustness alongside reliability in lung cancer classification.

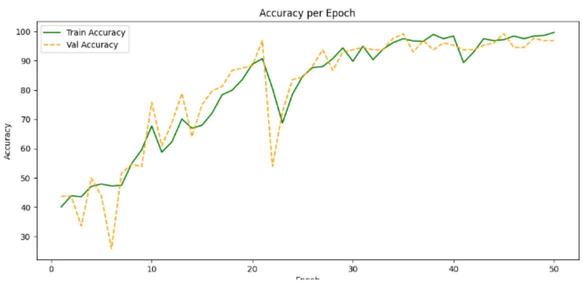

Fig 4.7: AlexNet Training and Validation Accuracy per Epoch

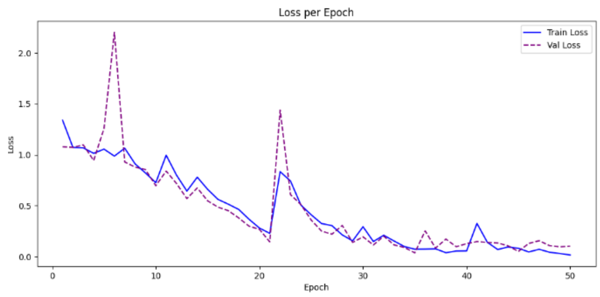

Fig 4.8: AlexNet Training and Validation Loss per Epoch

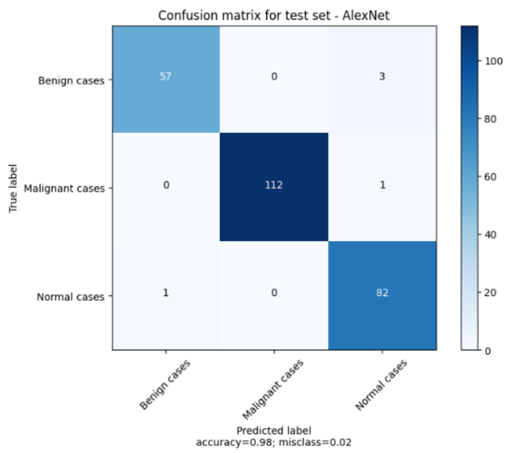

Fig 4.9: AlexNet Confusion Matrix for Test Data

4.3.5 ResNet50 Model Performance Analysis in Centralized Training

Through Figure 4.10 and Figure 4.11 ResNet-50 model underwent 30 epochs of training allowing monitoring of both training and validation accuracy and loss. ResNet-50 network produced outstanding performance at the start of training since it achieved 99.55% training accuracy and 97.66% validation accuracy together with 0.0207 training loss and 0.0469 validation loss. Although validation loss showed little fluctuations in the first epochs, accuracy levels maintained high all through the period. After first falling to 86.72% at epoch 3, the model restored its validation performance to exceed 95% between epochs 3 and 7. At epoch 7, ResNet-50 obtained 99.22% validation accuracy; it maintained this level all through the next training cycle. Training losses below 0.01 coupled with stable validation losses at 0.03 confirmed both outstanding convergence as well as lowest overfitting patterns from epoch 8 forward. Maintaining rather low final losses, the model proved perfect training accuracy (100%), and validated its performance with 99.22% accuracy in the last epoch. The successful training period showed how ResNet-50 efficiently extracts features fit for lung CT scan classification tasks. The produced confusion matrix shown in Figure 4.12 helped one to assess the model's category-wise performance. ResNet-50 demonstrated nearly perfect true positive accuracy across all diagnostic categories and generated an amazing confusion matrix whereby it correctly classified every instance. When classifying lung cancer stages from CT scan images, these results clearly show the strong generalization capacity and robust performance and dependability of accuracy of the model.

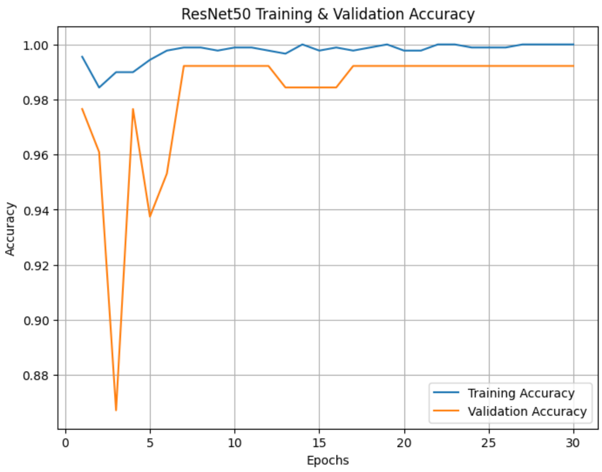

Fig 4.10: ResNet50 Training and Validation Accuracy per Epoch

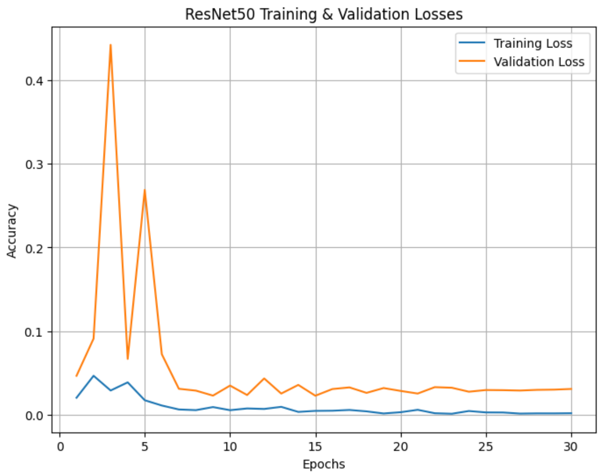

Fig 4.11: ResNet50 Training and Validation Loss per Epoch

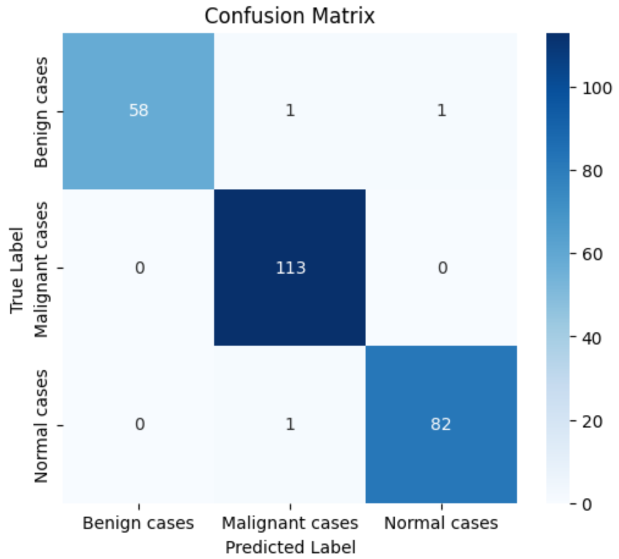

Fig 4.12: ResNet50 Confusion Matrix on Test Data

4.3.6 Federated Learning Results

The federated learning experiments performed aggregation techniques through three algorithms (FedAvg, FedProx, and FedADMM) while working under both IID (Independent and Identically Distributed) and non-IID (Non-Independent and Identically Distributed) data distribution. The models consist of a custom CNN alongside LeNet-5 and ResNet-50 which achieved training and testing accuracy measurements as shown in Table 4.2.

Table 4.2: Federated Learning Performance

Model	FedAvg				FedProx				FedAdmm			
	IID		Non-IID		IID		Non-IID		IID		Non-IID	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
CNN	99.43	99.09	93.39	93.64	99.66	99.09	93.39	93.64	36.79	42.73	36.79	42.73
LeNet5	99.89	100.00	95.55	97.73	96.01	95.91	91.56	93.18	51.37	50.00	11.85	7.27
ResNet 50	99.77	100.00	94.42	94.09	99.89	100.00	94.99	95.91	----	----	----	----

A performance comparison between FedAvg FedProx and FedADMM took place using both IID and non-IID data conditions. FedAvg demonstrated exceptional performance using IID conditions because ResNet-50 and LeNet-5 obtained a perfect 100% test accuracy indicating the benefits of federated averaging under uniform data allocation. Non-IID conditions led to diminished performance since ResNet-50 experienced a 5.91% test accuracy reduction to 94.09% which indicated the problems that occur when data distribution is diverse. The FedProx algorithm enhanced FedAvg performance especially when data distribution was non-IID by adding a proximal term which stabilized local updates so ResNet-50 obtained 95.91% test accuracy while retaining higher accuracy for LeNet-5. The non-IID data conditions proved difficult for FedADMM to handle because it led to LeNet-5's test accuracy reaching only 7.27% while CNN could achieve 42.73% accuracy at best. The combination of high computational requirements and sensitive parameter handling resulted in incomplete results for ResNet-50. During all federated experiments

ResNet-50 proved more robust than the custom CNN and LeNet-5 models which demonstrated its effectiveness for processing medical imaging tasks across remote systems. Accurate results for custom CNN and LeNet-5 under IID conditions evolved into considerable deterioration when used with FedADMM under non-IID settings because simpler models respond negatively to data distribution heterogeneity. The combination of FedProx as the most resilient aggregation approach combined with ResNet-50 as the optimal model produced optimal results in IID and non-IID environments indicating the essential role of choosing proper aggregation methods together with powerful models for effective federated learning operation.

4.3.7 Performance Analysis of CNN Model in Federated Learning

In this part, a thorough assessment of the CNN model under several federated learning techniques is conducted. Working with both IID and non-IID data distribution models, the study follows FedAvg and FedProx algorithm performance in training accuracy and loss levels. The experimental data show how CNN achieves generalization capability together with stability and convergence in distributed environments.

Figure 4.13 shows a training accuracy of 99.43% during Round 7, then stabilizing for all next rounds. Starting from 0.7204 in Round 1, Figure 4.14 shows the decrease of training loss which settles at approximately 0.5570 after Round 7. Data shows consistently between clients FedAvg achieves fast convergence with small variations indicating its efficiency in IID settings. Table 4.2's 99.09% test accuracy of the model corresponds with the experimental data of this work. FedAvg's consistent model performance results from equal level data distribution.

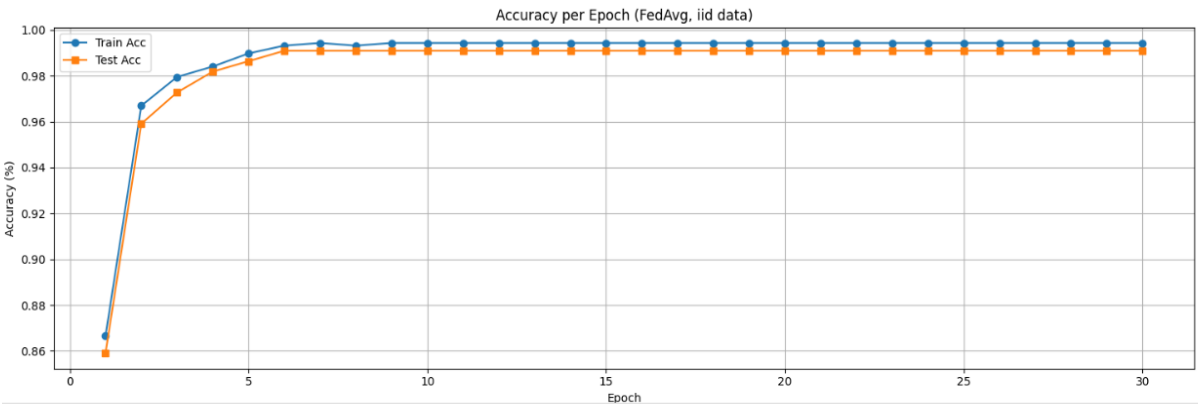

Fig 4.13: Training accuracy of CNN model using FedAvg on IID data over 30 rounds

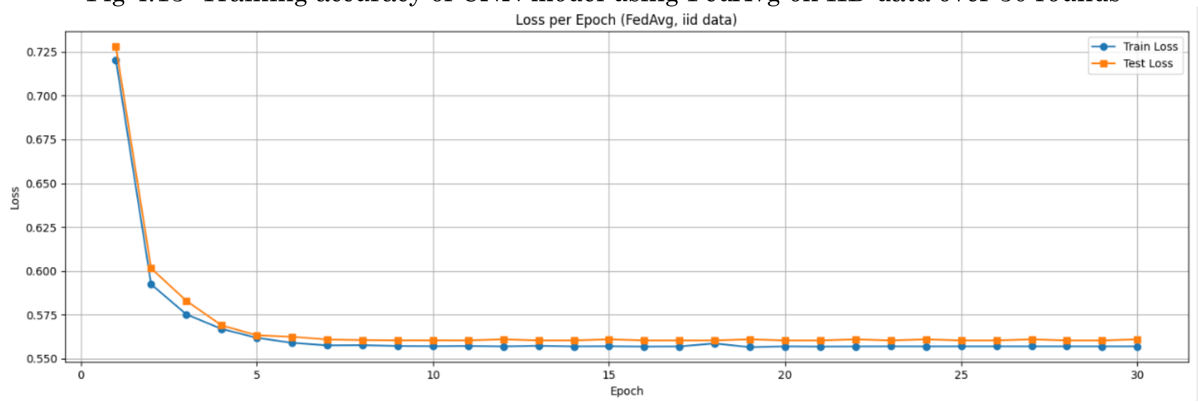

Fig 4.14: Training loss of CNN model using FedAvg on IID data over 30 rounds

The training accuracy in Figure 4.15 increased more slowly and less confidently by 30 rounds starting from 67.77 % of Round 1 and ending by 94.19 % of Round 28 then stayed constant at 93.39 % of Round 30. Non-IID data causes accuracy to decrease even further to 81.44% (during Round 13 and along with other fluctuations seen). Figure 4.16 illustrates a high starting loss 0.8636 with oscillating loss during training and end value of 0.6125. While data heterogeneity does have an effect on the performance of the test, FedAvg is able to achieve 93.64% test accuracy (Table 4.2), affirming our claim of its robustness to non-IID conditions, albeit at a lower value than the IID setting. The visualizations show crucial properties of how the model learns. The CNN model with FedAvg achieves fast and stable convergence in IID data with uniform settings. The model exhibit non-IID performance is by large, and its diverse results prove that FedProx can achieve higher efficiency of the treatment of data heterogeneity as show by its 95.91% test accuracy on ResNet50. The outcome of the evaluation has enforced to choose FedProx as the primary model in the web application, with best performance and reliability.

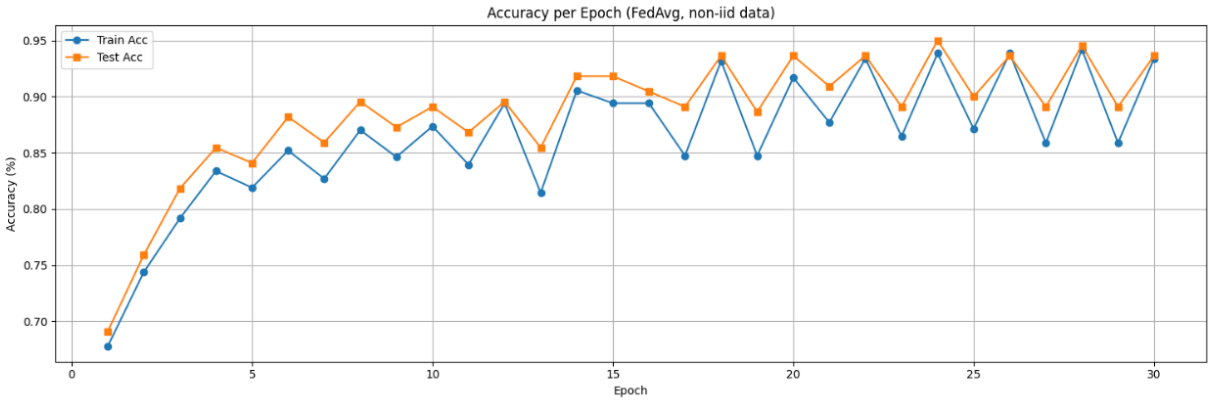

Fig 4.15: Training accuracy of CNN model using FedAvg on Non-IID data over 30 rounds

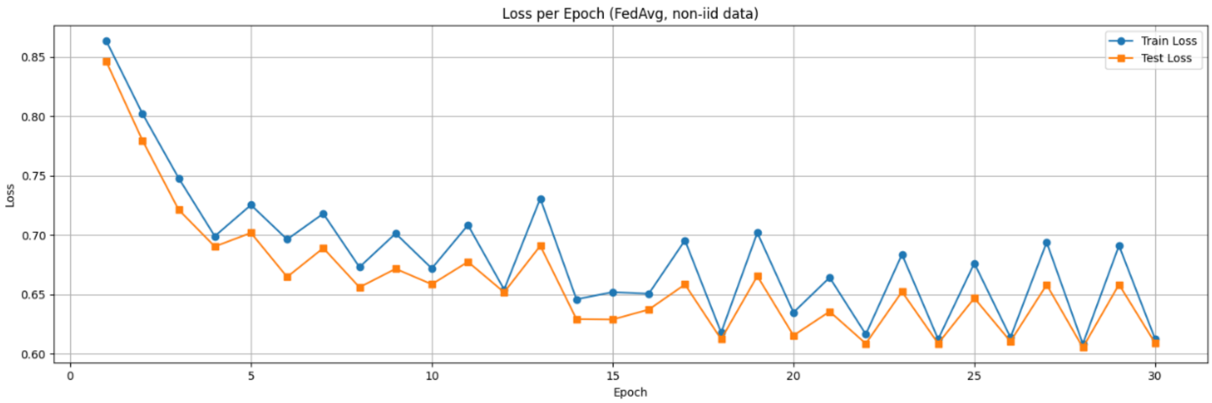

Fig 4.16: Training loss of CNN model using FedAvg on Non-IID data over 30 rounds

The training accuracy of the CNN model trained with the FedProx algorithm on IID dataset over 30 communication rounds is shown in Fig 4.17. The values of accuracy: 88.15% (in Round 6), 86.79% (in Round 2), and \ 81.55% (in Round 1). There is massive increase in attendance in Round 8 (94.42%) and then continues to increase further to 97.38% even in Round 10. the accuracy will be 99.32 % for round 13, and for 17th and following rounds will fluctuate between 99.43 % to 99.77 %. This gradual increase of error and early convergence of deep network indicates the effectiveness of the proposed algorithm in the homogeneous data field. The final accuracy of the training data has no difference with the highest test accuracy of 99.09%, illustrating good generalization.

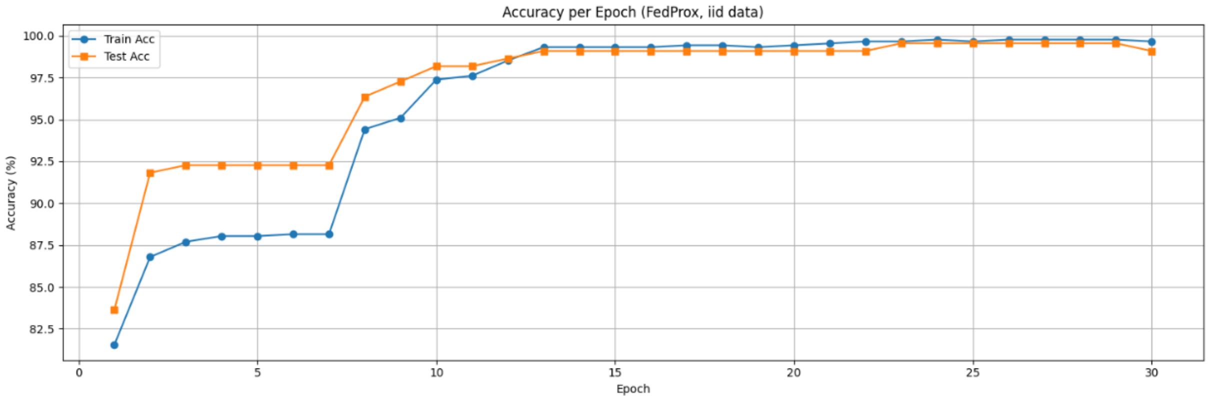

Fig 4.17: Training accurcay of CNN model using FedProx on IID data over 30 rounds

Training Loss in 30 Rounds Figure 4.18 shows the training loss of CNN model tuned with FedProx on IID dataset in 30 rounds. 0.7310 is Loss 1, which is down continuously to only 0.6834 by Round 2 and further to 0.6638 by Round 6. A significant drop is observed at Round 8 (0.6120), and then the confidence drops to 0.5769 by Round 10, and to 0.5584 by Round 13. Since Round 21, the training loss remains steady around 0.5537-0.5553, validating that the loss has converged. Losses consistently decrease, thus validating the observed trends in training accuracy and manifesting the successful optimization obtained by FedProx..

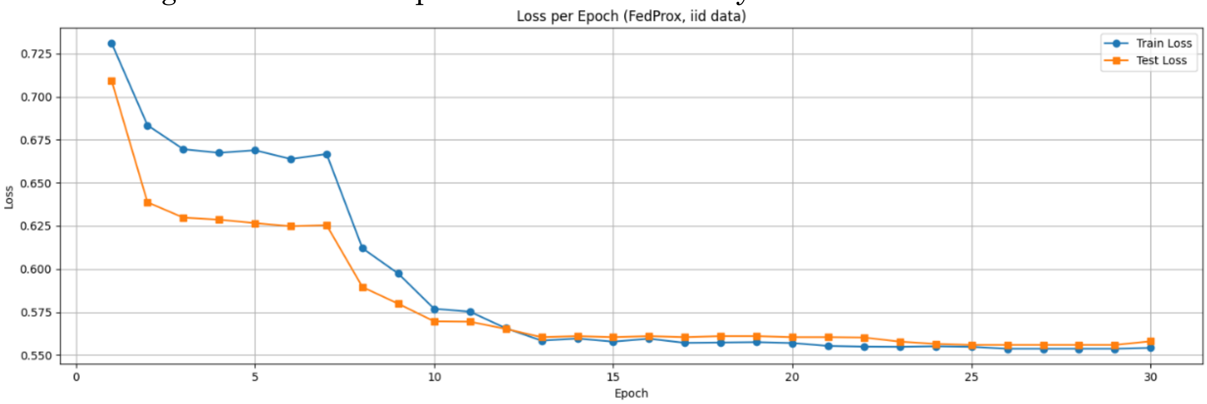

Fig 4.18: Training loss of CNN model using FedProx on IID data over 30 rounds

The training accuracy of the CNN model using FedProx on non-IID data reaches its peak at 94.19% during Round 28 as Figure 4.19 illustrates throughout 30 rounds of execution. During Round 1 the accuracy measurement stands at 67.77% which demonstrates the obstacle of working with heterogeneous data types. Training progress shows an escalating trend from 79.16% in Round 3 to 85.19% in Round 6. The accuracy of the model fluctuates during specific rounds such as Round 13 where it drops to 81.44% and Round 27 and 29 where it reaches 85.88% but the model achieves its peak performance with 94.19% accuracy in Round 28 and ends with 93.39% accuracy at Round 30. The experimental outcomes show FedProx successfully processes distribution heterogeneity better than FedAvg yet the process needs additional work to enhance training stability.

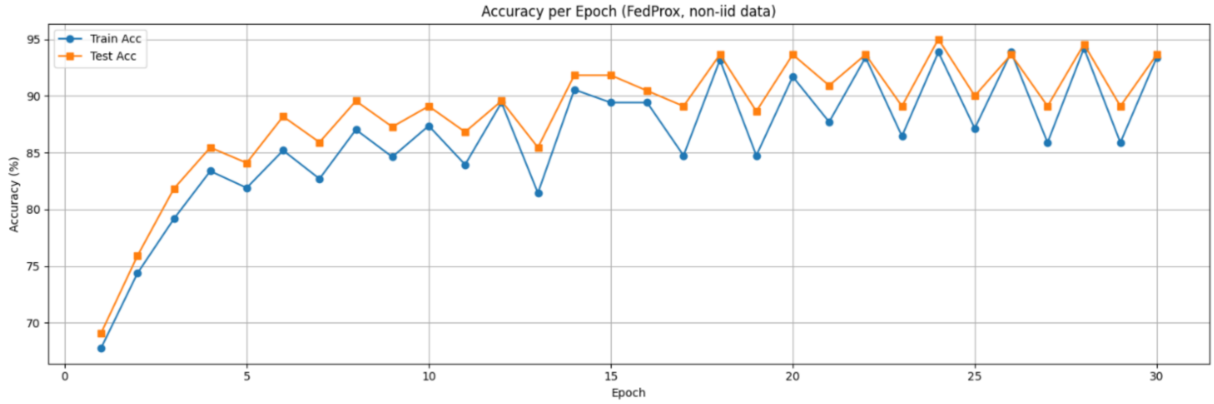

Fig 4.19: Training accuracy of CNN model using FedProx on non-IID data over 30 rounds

The training loss of CNN model with FedProx on non-IID dataset appears in Figure 4.20. The starting value of loss sits at 0.8636 during Round 1 before reaching values of 0.7481 in Round 3 and finally settling at 0.6964 in Round 6. The loss values demonstrate an unstable pattern because they spent Rounds 13 (0.7308) and 27 (0.6940) in an upward direction. At Round 28 the model reaches its best performance point with 0.6084 loss before ending at 0.6125 loss in Round 30. The implementation of proximal regularization in FedProx shows proper downward loss trends even though some minor oscillations occur.

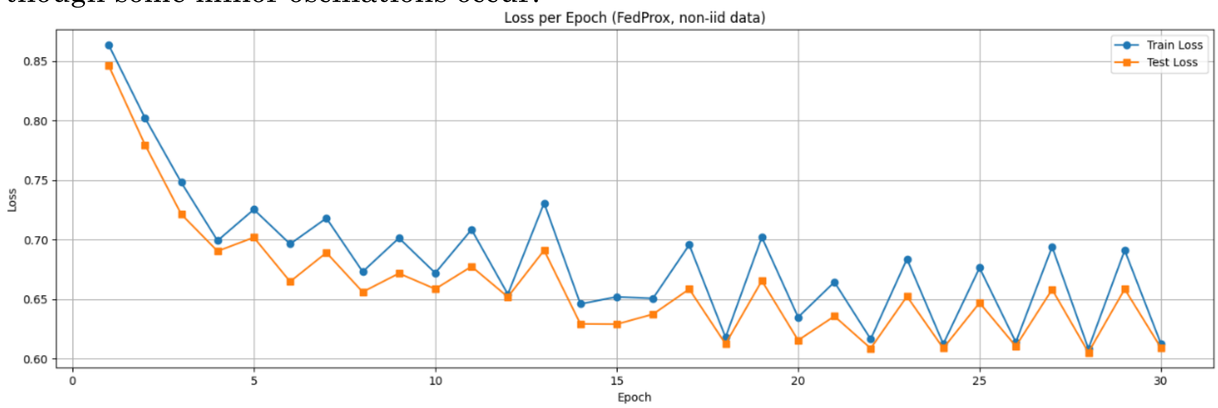

Fig 4.20: Training loss of CNN model using FedProx on non-IID data over 30 rounds

4.3.8 Performance Analysis of LeNet5 Model in Federated Learning

In this part, the federated learning structure is designed to test the performance of the LeNet5 model. The evaluation includes training accuracy and loss of IID and non-IID data distributions of FedAvg and FedProx. Results show the convergence patterns, robustness, and adaptability of LeNet5 in a decentralized environment.

(C) (Simple Federated Learning). Figure 4.21 shows the training accuracy of LeNet5 with the FedAvg algorithm on the IID data set for different numbers of communication round. The x-axis is the training round number (1–30) and the y-axis is the percentage of training set accuracy (0–100%). We observe a Round 1 accuracy of 95.78%, implying high performance in the early stages of the training, as expected given the strong classification performance of LeNet5 on images. The accuracy rises quickly to 98.29% by Round 2 and 98.97% by Round 3. Accuracy stabilized at 99.65% after 7 and 99.66-99.89 after 8 with a small oscillation.

The fast rates of convergence and high accuracy further demonstrate the effectiveness of FedAvg under IID settings where the clients have uniform class distribution thus guaranteeing uniform model updates. The 100.00% test accuracy (Table 4.2) demonstrates a similar robustness achieved for LeNet5 model on IID data over CNN model achieving 99.09% test accuracy.

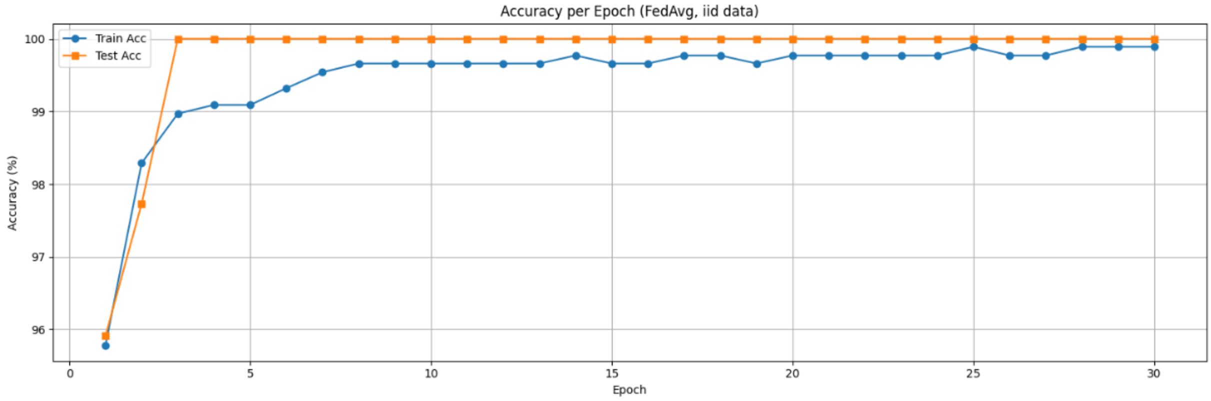

Fig 4.21: Training accuracy of LeNet5 model using FedAvg on IID data over 30 rounds.

The training loss of LeNet5 using FedAvg on IID data appears as Figure 4.22 over 30 communication rounds. The training loss (categorical cross-entropy) measure appears on the y-axis alongside the training rounds displayed on the x-axis. The LeNet5 started its learning process at 0.1622 in Round 1 with a significantly lower loss value than the 0.7204 loss of the CNN model which indicates superior efficiency during initial training. From Round 1 to Round 8 the loss value decreases from 0.0765 through 0.0383 to 0.0100. The loss becomes stable within the range 0.0013 to 0.0119 from Round 14 until it ends at 0.0057 during Round 30. The fast loss decrease and its subsequent maintenance at low stable values indicates effective model convergence which matches with observed high training and test accuracy results. LeNet5 demonstrates superior loss reduction performance than the CNN model by reaching a final loss of 0.5570 which supports its use in IID data distributions.

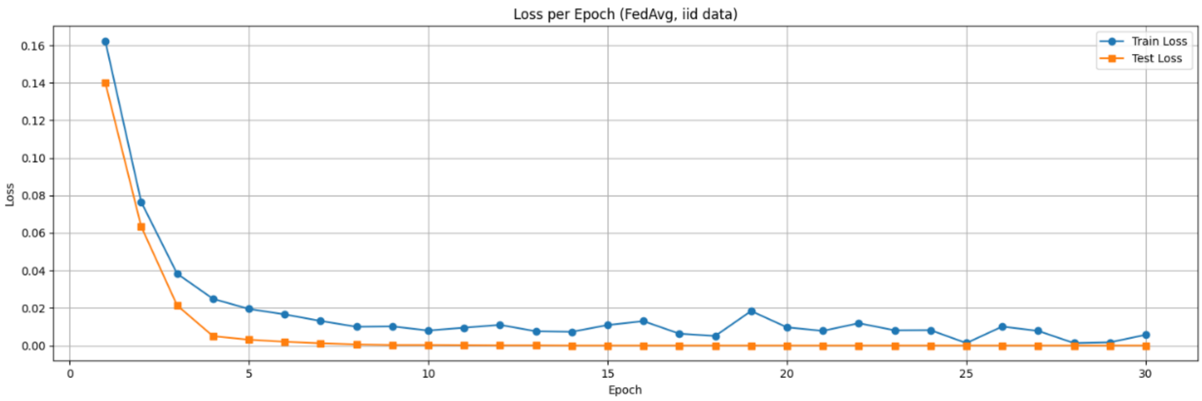

Fig 4.22: Training loss of LeNet5 model using FedAvg on IID data over 30 rounds.

The training accuracy of LeNet5 with FedAvg applied on non-IID data spreads across 30 communication rounds as shown in Figure 4.23. The model training process appears on the x-axis which shows the number of rounds alongside the y-axis that demonstrates the percent accuracy of the model during training. The training accuracy arrives at 73.89% during Round 1 because the client-based class distribution imbalances present significant challenges. The network training accuracy shows an upward trend through various rounds from 80.27% in Round 3 until it reaches its peak at 96.47% in Round 26. Furthermore it maintains consistent training accuracy at 89.85% in Round 6. The training accuracy exhibits significant valleys at Round 5 and Round 7 reaching 81.87% and 84.49% respectively but stabilizes at levels between 95.10% and 95.55% starting from Round 25. The model displays robust performance in non-IID circumstances through its final test accuracy of 97.73% (Table 4.2) while achieving superior results than the non-IID test accuracy of 93.64% performed by the CNN model. FedAvg shows excellent performance in adapting with LeNet5 architecture across heterogeneous data distributions based on the experimental results.

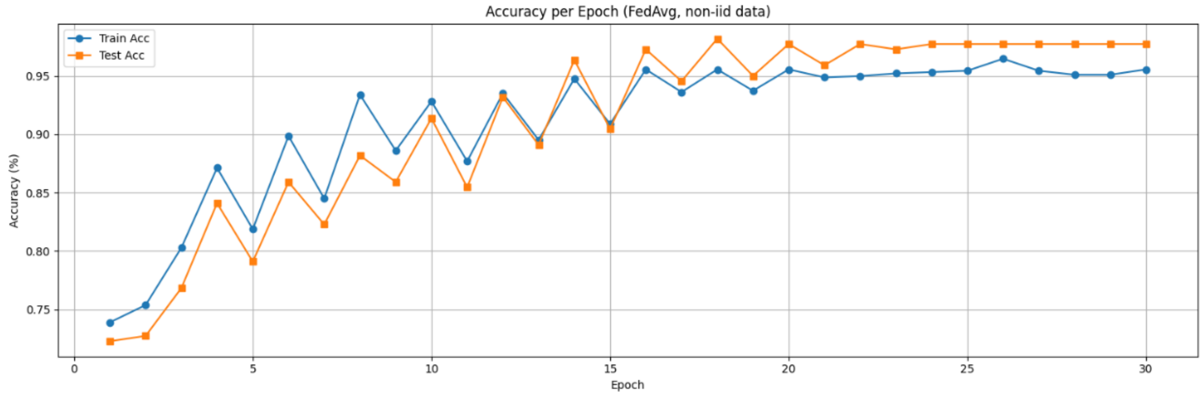

Fig 4.23: Training accuracy of LeNet5 model using FedAvg on non-IID data over 30 rounds.

As shown in Figure 4.24, the training loss progression of LeNet5 using FedAvg on heterogenized data over 30 communication rounds. In the vertical axis, we have training loss value, and on the horizontal axis is number of training rounds. In Round 1 the loss of LeNet5 is 0.8220, which is lower than CNN, which gives a loss of 0.8636. The loss level indeed falls from 0.5491 to 0.2896 from Round 2 to Round 4. The loss showed declining trend because all through the process it maintained various increases at Round 7 (0.5074) and Round 13 (0.3781). In the R30, the final training loss of 0.3146 is achieved in combination with the least training loss of 0.1588 in the R14 Round. Although the data undergo periodic fluctuations, the model remains successful in its adaptation during learning with a downward trend. The above image depicts the outcome of training of LeNet5 that shows good capability of LeNet5 to perform better than CNN model (0.6091) since LeNet5 can handle non-IID well as it results in good test accuracy level.

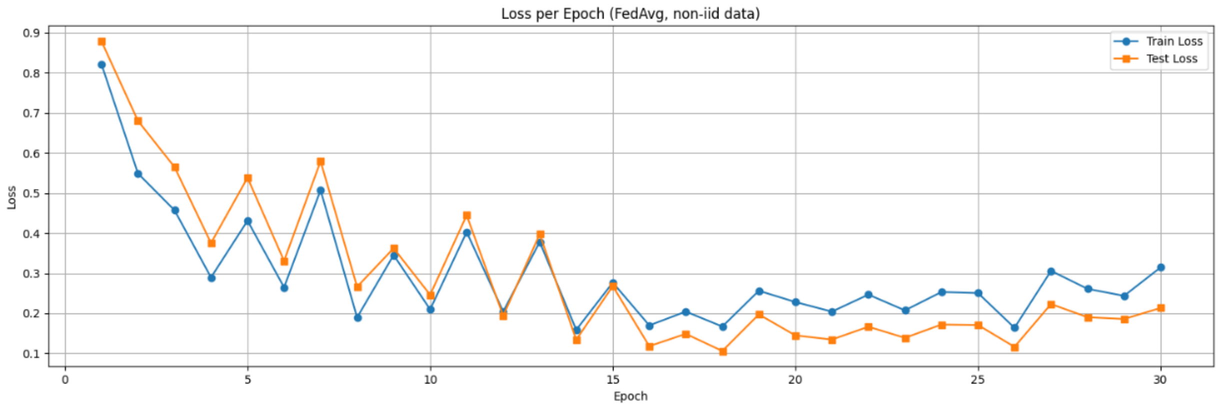

Fig 4.24: Training loss of LeNet5 model using FedAvg on non-IID data over 30 rounds.

Figure 4.25 shows the training accuracy of the LeNet5 model against the FedProx algorithm with the IID dataset for 30 rounds of communication. At Round 1, the model had an initial accuracy of 50.63%, much less than the accuracy seen using FedAvg (95.78%). Accuracy improves steadily, from 59.52% to 65.79% to 83.69% by Round 2, Round 8 and Round 12, respectively. It is also worthy to note that the accuracy increases dramatically in round 14 (89.97% accuracy), and then keeps on improving toward 96.92% accuracy in round 28. From Round 28 onwards the accuracy stabilizes at 96.01% to 96.24%. This, and the fact that it follows FedProx's use of a proximal term that slows convergence, but improves stability when fed IID data, is what the gradual improvement represents. Finally, the performance of the FedProx is lower compared to that of FedAvg 100.00% (see Table 4.1), however the tradeoff for robustness vs peak performance should be considered when using FedProx.

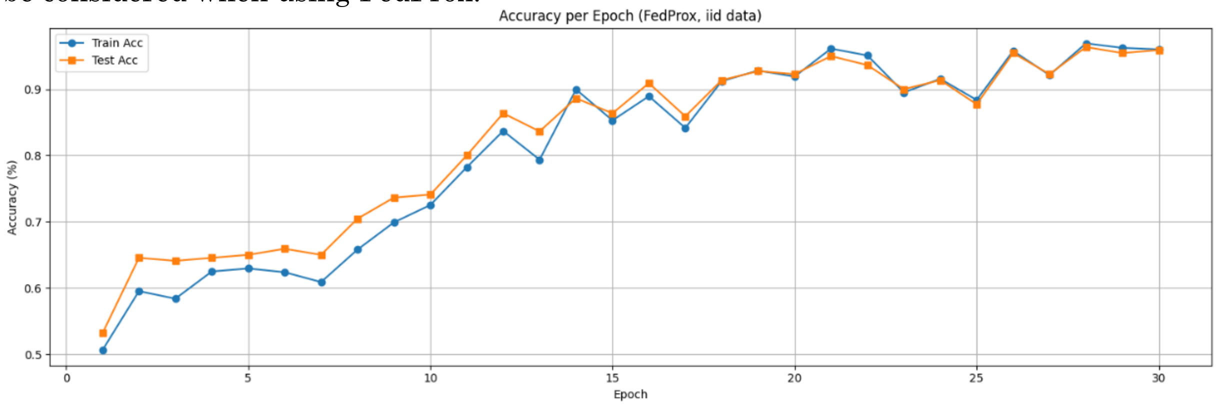

Fig 4.25: Training accuracy of LeNet5 model using FedProx on IID data over 30 rounds.

A plot for the training loss of LeNet5 model running on IID data with FedProx algorithm over 30 communication rounds is shown in Figure 4.26. The initial loss with FedAvg is at 0.1622, but in Round 1 loss begins at 1.3637. Starting from the beginning, it drops to 1.2250 by Round 2, 0.9208 by Round 8, 0.3761 by Round 12, a large dip to 0.2326 by Round 14, and so on. The loss approaches a low of 0.0755 by Round 28, and stays down in the range of 0.0873 within one SD of the mean and between 0.0873 and 0.0928 by Round 30. FedAvg converges faster, however, at a slower rate than the training loss continues to decline, suggesting that FedAvg is actually performing well for the optimization of training loss. Since regularization slows down the cluster convergence rate, FedProx has the higher final loss (0.0057) compared to FedAvg. The observed stable learning behavior agrees with the final test loss of 0.0838.

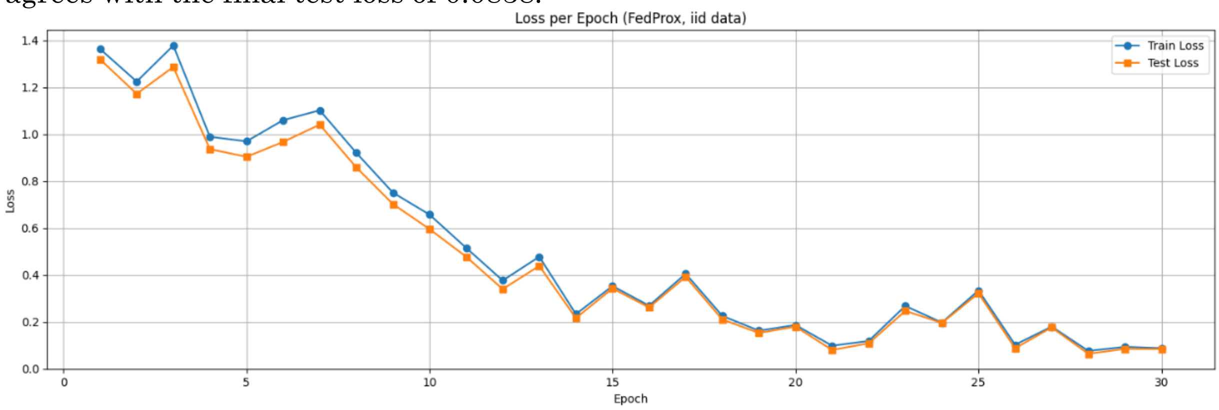

Fig 4.26: Training loss of LeNet5 model using FedProx on IID data over 30 rounds.

Figure 4.27 shows on a non-IID dataset over 30 communication rounds the LeNet5 model's training accuracy using the FedProx approach. Mirroring the IID case, the training accuracy starts at 50.63% in Round 1 and increases significantly to 79.25% by Round 2, so proving FedProx's early flexibility to heterogeneous data. Variations abound; accuracy falls to 65.34% in Round 5 and 68.19% in Round 8, then recovers to reach 82.90% in Round 17. Round 25 sees the model peak at 91.68% while Round 30 maintains a final accuracy of 91.56%. These swings draw attention to the difficulties in learning from non-IID data sources. Notwithstanding variation, FedProx's resilience is confirmed by the general upward trend and final test accuracy of 93.18% (Table 4.2), closely matching its CNN performance (93.64%) and somewhat underperformance FedAvg's LeNet5 model (97.73%).

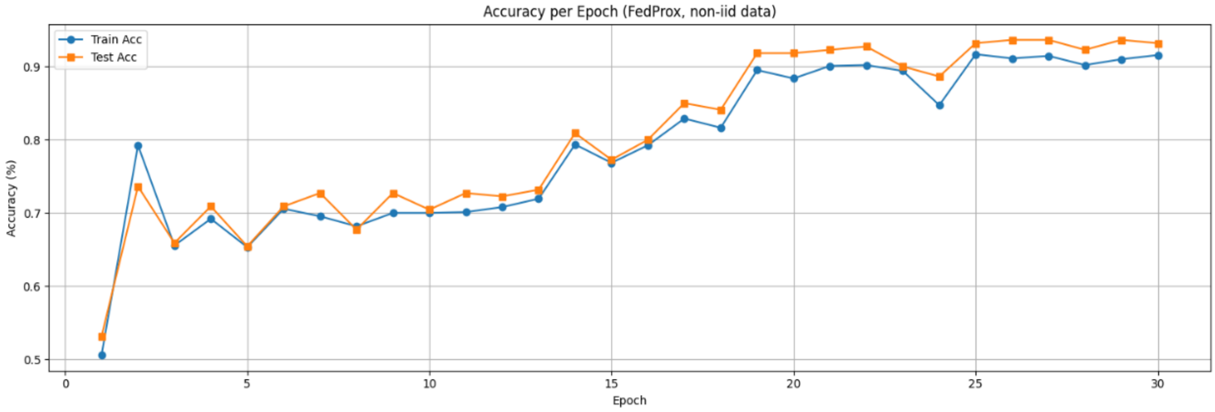

Fig 4.27: Training loss of LeNet5 model using FedProx on non-IID data over 30 rounds.

Figure 4.28 demonstrates over 30 communication rounds the LeNet5 model's training loss using FedProx on non-IID data. Reflecting the challenges of non-IID learning with FedProx's regularization, the beginning loss is observed at 2.3696 in Round 1, far higher than FedAvg's starting loss (0.8220). The loss drops sharply; Round 2 brings it to 0.5803. Still, there are noticeable swings; rises to 1.1112 in Round 7 and 1.0550 in Round 13. From Round 17 on, minimum 0.3436 by Round 27, and ends at 0.4533 in Round 30, the loss falls steadily.
Although the oscillatory behavior, overall the model adaptation to data heterogeneity demonstrates good performance. Comparable to FedAvg's non-IID LeNet5 performance (0.3146), the last training loss is smaller than CNN FedProx loss (0.6125). Though there is fluctuation, the test loss of 0.3934 supports the effective learning achieved.

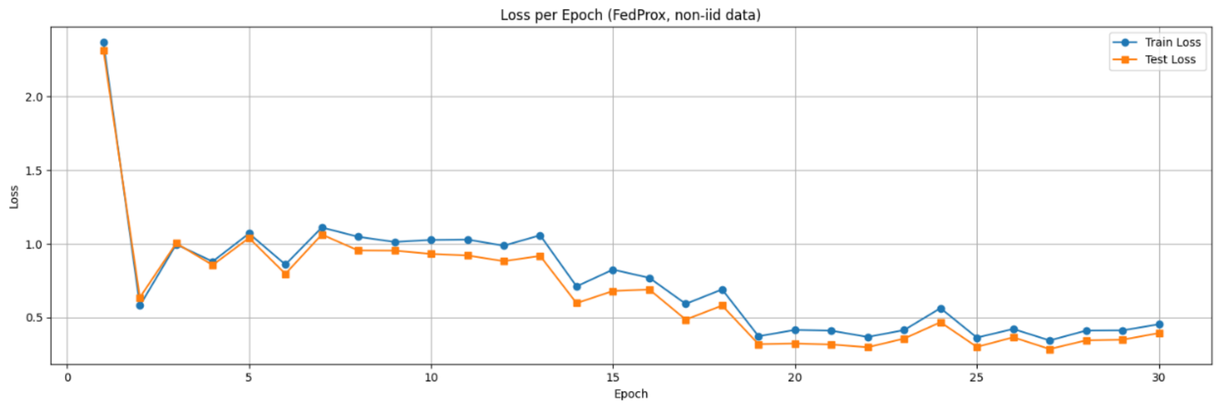

Fig 4.28: Training loss of LeNet5 model using FedProx on non-IID data over 30 rounds

4.3.9 Performance Analysis of ResNet50 Model in Federated Learning

Here we assess the ResNet50 model's performance under a federated learning environment. Using FedAvg and FedProx with IID and non-IID datasets, the study of training accuracy and loss is reported. The results imply that ResNet50 is highly effective to process complex and distributed datasets, convergent fast and stable.
With FedAvg for ten rounds, figure 4.29 displays the ResNet50's training accuracy on IID federated dataset. With R1, the training accuracy began at 51.37% indicating the initial complexity resulting from ResNet50's deep architecture. It rises gradually to 53.99% in round two; it spikes to 79.84% in round three; and it peaks at 96.81% in round four. The accuracy rises even more to 99.32% by the sixth round; from round

seven and beyond, it levels to 99.77%. Under IID, where patterns are independent identically distributed (IID), FedAvg shows its efficiency; this is the case if the data is evenly distributed among clients—that is, 319 samples per client, producing 1278 samples—shown by the fast recovery. Table 4.2 allows us to derive outstanding generalization of ResNet50 for IID scenario with final test acc 100%.

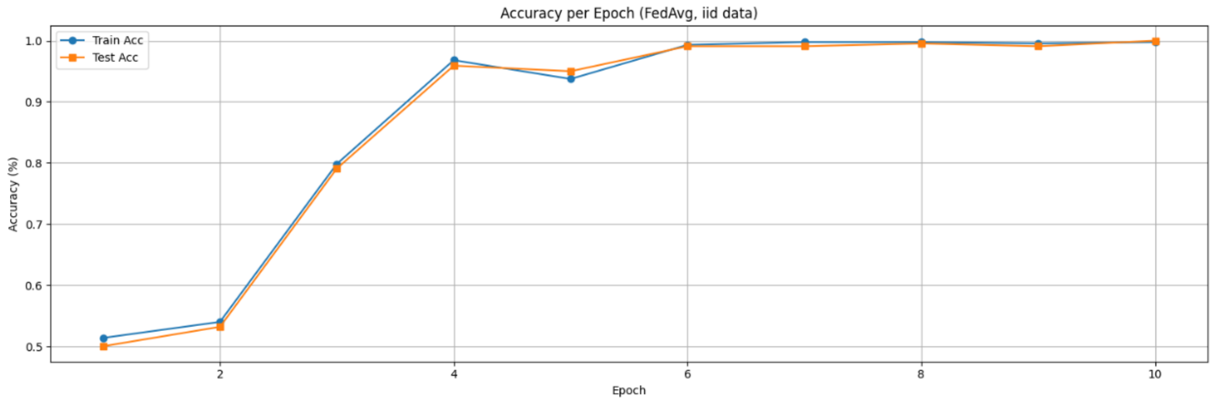

Fig 4.29: Training Accuracy of ResNet50 Model Using FedAvg on IID Data

Figure 4.30 displays, on an IID set, the training loss of the ResNet50 model using FedAvg across ten rounds. The intricacy of ResNet50 results in a training loss in Round 1 starting at 2.7579, well above more basic models like LeNet5. It lowers quite sharply to 1.6391 by Round 2, 0.5015 by Round 3, and Round 4, 0.0861. From Round 6, the loss keeps declining to 0.0194 and stabilizes at 0.0061 and 0.0072 from Round 7 onward, with a tiny uptick to 0.0232 in Round 9. The 0.0072 ultimate loss value of Round 10 points to effective convergence and optimization. Table 4.2 shows the excellent performance of the model in IID settings by the test loss of 0.0001 and 100.00% test accuracy.

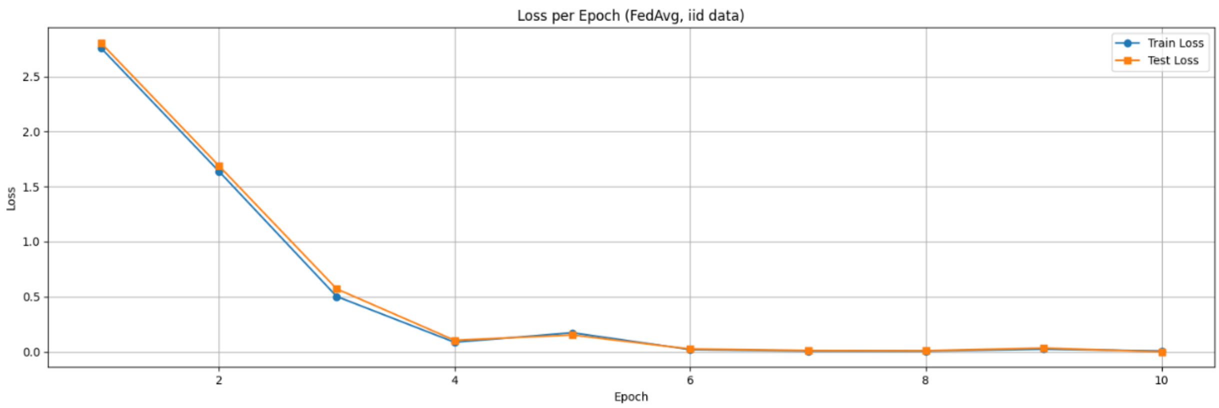

Fig 4.30: Training Loss of ResNet50 Model Using FedAvg on IID Data

Figure 4.31 shows, using a non-IID dataset across 30 rounds, the ResNet50 model's FedAvg-based training accuracy. Reflecting the challenges given by heterogeneous data distribution—client sizes: 300, 300, 300, 288; total 1278 samples—the training accuracy starts at 51.37% in Round 1 and continues steady through Round 4. Round 5 starts improvement with 52.28%; by Round 7 it reaches 57.74%; by Round 8 it reaches 71.18%; and in Round 21 and 22 it reaches a peak of 95.67%. The last training accuracy recorded in Round 30 is 94.42%; past rounds show clear variations (e.g., 73.92% in Round 14). Despite slower early progress and variability, the ultimate test accuracy of 94.09% (Table 4.2) reveals great flexibility of ResNet50 under non-IID situations, albeit considerably underperformance compared to LeNet5 and CNN models.

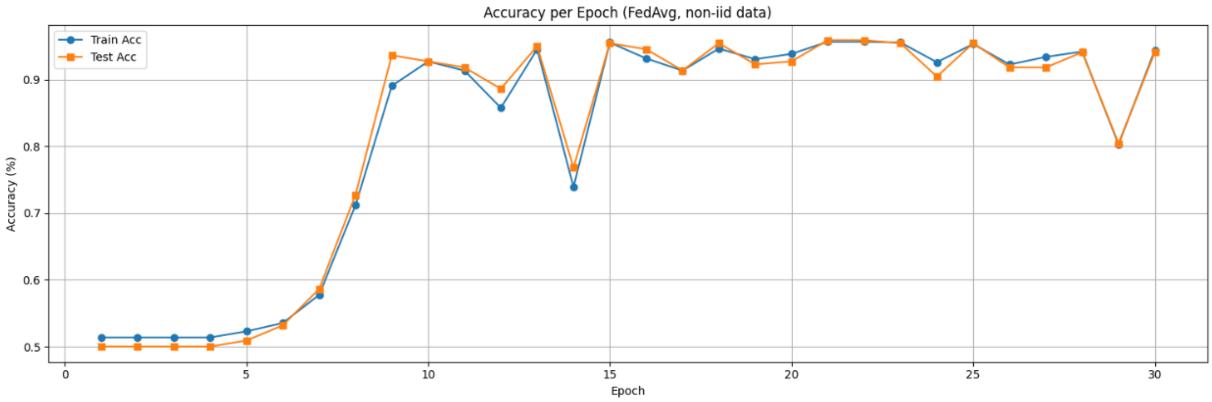

Fig 4.31: Training Accuracy of ResNet50 Model Using FedAvg on Non-IID Data

Figure 4.32 shows over 30 rounds the ResNet50 model's training loss using FedAvg on a non-IID dataset. Emphasizing the learning challenges resulting from skewed class distributions, the training loss starts at 2.2276 in Round 1, somewhat lower than the IID case but stays rather high through Round 4 (3.5455). After Round 6, the loss drops to 1.5980; after Round 8, it drops to 0.9228; finally, Round 21 brings a low point of 0.2599. In Round 30, the last loss noted is 0.4250; Round 14 shows clear fluctuations with 1.2189. Though with less stability than LeNet5 and CNN models, the overall downward trend and final test loss of 0.4244 (Table 4.2) indicate effective training even if the oscillatory behavior highlights the inherent difficulties of non-IID learning.

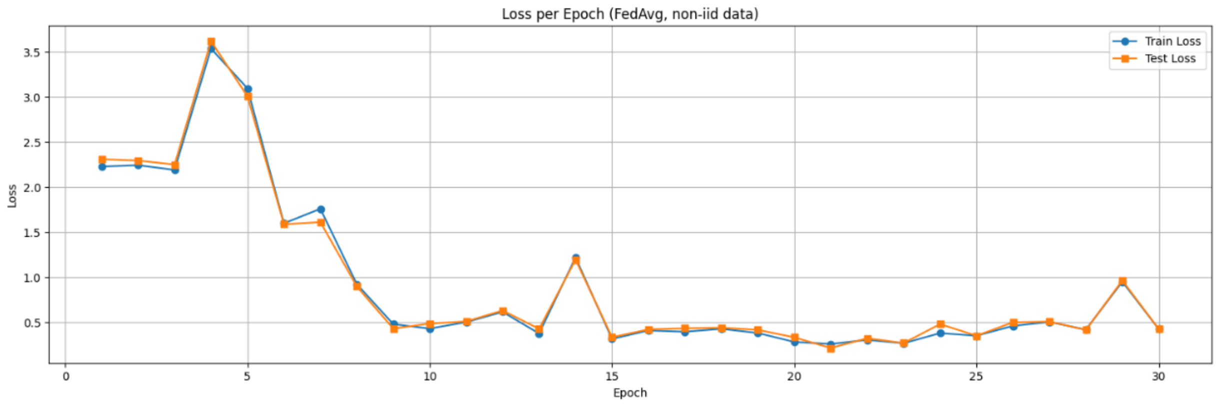

Figure 4.32: Training Loss of ResNet50 Model Using FedAvg on Non-IID Data

We show the training accuracy of ResNet50 model trained with the FedProx algorithm on IID set in 10 rounds of federated learning in Figure 4.33. The training accuracy begins with 51.37% in Round 1, also identical to that observed with FedAvg, showing the initial complexity of the deep architecture of ResNet50. This increases slowly to 54.90% in Round 2 and 60.02% in Round 3, but an increase to 99.09% is obtained by Round 4. Round 6 it equals 99.77% and becomes stable from round 8 to round 10 at between 99.89% to 100.00%. This fast trend shows the power of FedProx in IID environment where samples are uniformly distributed (e.g., 319 samples per client, a total of 1278 samples), and all updates are very similar. The best result with a final test accuracy of 100.00% (as shown in Table 4.2) is comparable to the performance of FedAvg, indicative of the robustness of ResNet50 and the use of FedProx as a regularization method.

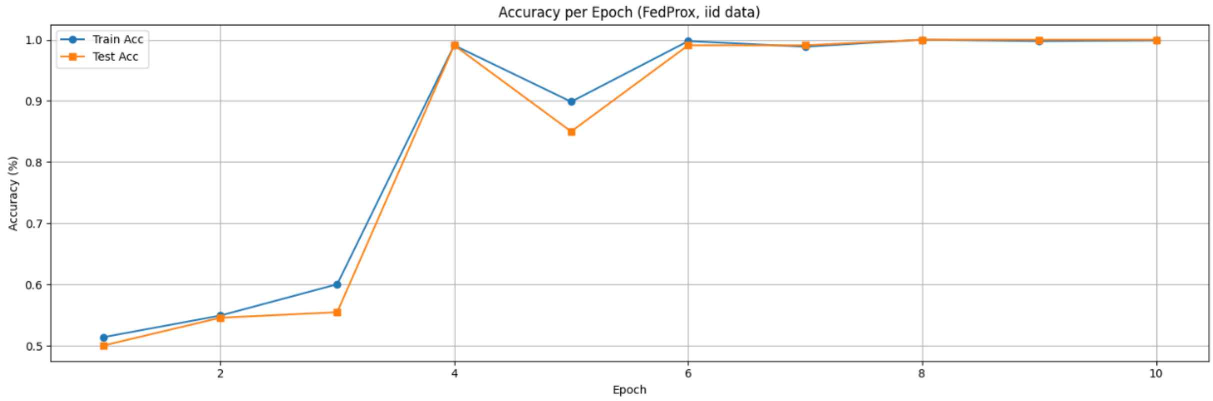

Fig 4.33: Training Accuracy of ResNet50 Model Using FedProx on IID Data

Fig. 4.34 illustrates the training loss of the ResNet50 model with FedProx method over ten rounds on IID dataset. The proximal-like term implies that the training loss starts from Round 1 at 3.1301, it is higher than 2.7579 of FedAvg since it adds extra regularizing effect. It drops precipitously, to 1.6685 in Round 2, 1.2659 in Round 3, and even further, to 0.0306 in Round 4. Loss reduced even further by Round 8 when it goes down to a figure as low as 0.0014, and stays stabilized between 0.0011 and 0.0048 up to Round 10. We can see that the optimization is well done as final loss, i.e., 0.0011 is too low and it only decreases to a low value. By obtaining a smaller final loss than FedAvg, the 0.0000 test loss (Table 4.2) and test accuracy of 100.00% further verify the stability and efficiency of FedProx in the IID setting.

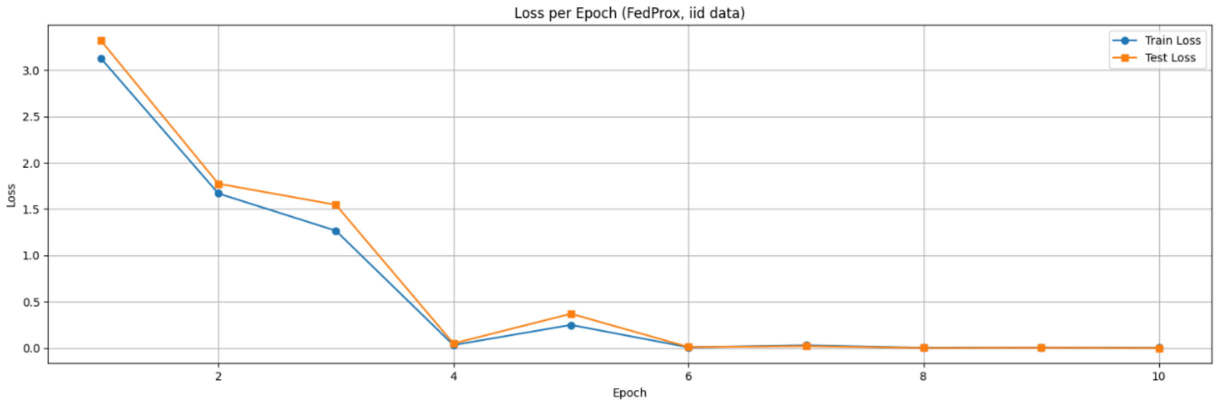

Fig 4.34: Training Loss of ResNet50 Model Using FedProx on IID Data

Figure 4.35 illustrates on a non-IID dataset over 15 rounds the training accuracy of the ResNet50 model using the FedProx method. Starting at 51.37% in Round 1 and maintained constant through Round 3, the training accuracy highlights the challenges stemming from heterogeneous data distribution (client sizes: 300, 300, 300, 288; total 1278 samples). Round 4 indicates improvement with a jump to 52.51%; Round 5 shows a startling increase to 72.21%; Round 7 reveals 80.52%. Round 11 brings accuracy of 91.23%; Round 15 peaks at 94.99%. FedProx's improved management of data heterogeneity compared to FedAvg's performance for ResNet50 reflects in the final test accuracy of 95.91% (Table 4.2), even if the slow initial progress and gradual rising trend illustrate the difficulties associated to non-IID data.

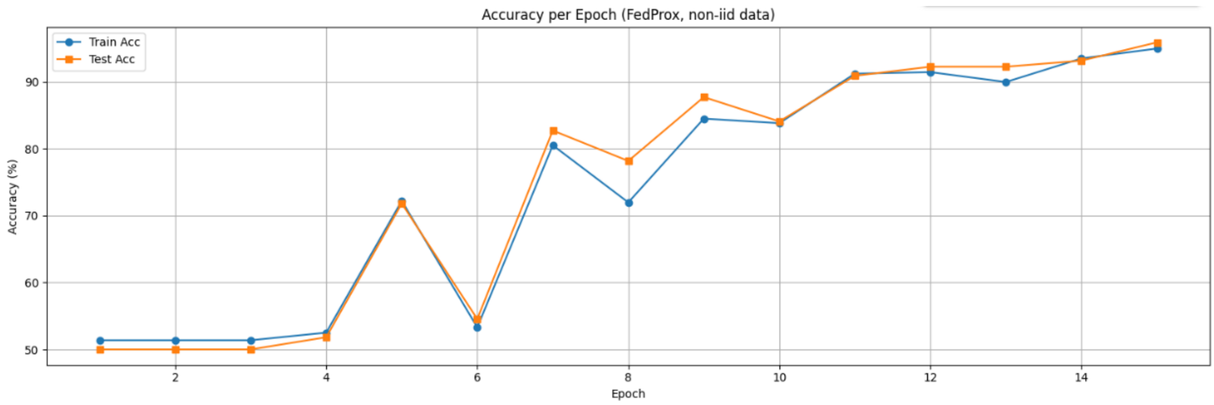

Fig 4.35: Training Accuracy of ResNet50 Model Using FedProx on Non-IID Data

Using a non-IID dataset over 15 rounds, Figure 4.36 shows the training loss of the ResNet50 model employing the FedProx technique. Starting at 2.4809 in Round 1, somewhat higher than FedAvg's initial loss, the training loss rises to 3.0330 in Round 2 reflecting the difficulty of learning from skewed class distributions combined with FedProx's regularizing effects. By Round 3, the loss drops to 1.8513; by Round 5, it drops to 0.8451; and in Round 14 it reaches a low of 0.2409. Despite fluctuations like 2.3409 noted in Round 6, the last loss noted is 0.2527 in Round 15. Although the oscillatory behavior draws attention to the difficulties presented by non-IID settings, the general downward trend and the final test loss of 0.2031 (Table 4.2) verify enhanced stability and better convergence than FedAvg.

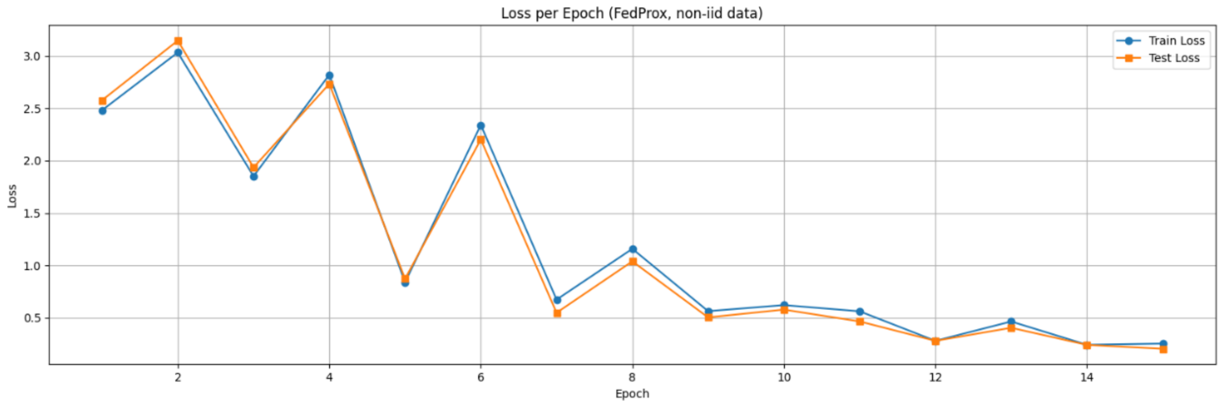

Fig 4.36: Training Loss of ResNet50 Model Using FedProx on Non-IID Data

4.3.10 Explainable AI Insights

Grad-CAM heatmaps were employed to provide visual explanations for the predictions made by the CNN model on CT images. The highlighted regions overlapped with nodule and abnormal tissue patterns and were consistent with the oncologist annotations in the IQ-OTH/NCCD dataset. These visualizations improved interpretability and confidence in the model's predictions, suggesting potential for the model to be used for clinical decision support. We have introduced explainable AI (XAI) methods to increase interpretability of the lung cancer classification models. CNN, LeNet5, AlexNet and ResNet50 Grad-CAM visualizations are illustrated in Figures 4.37 - 4.40. These visualisations indicate which parts of CT images mattered most to the predictions of the models.

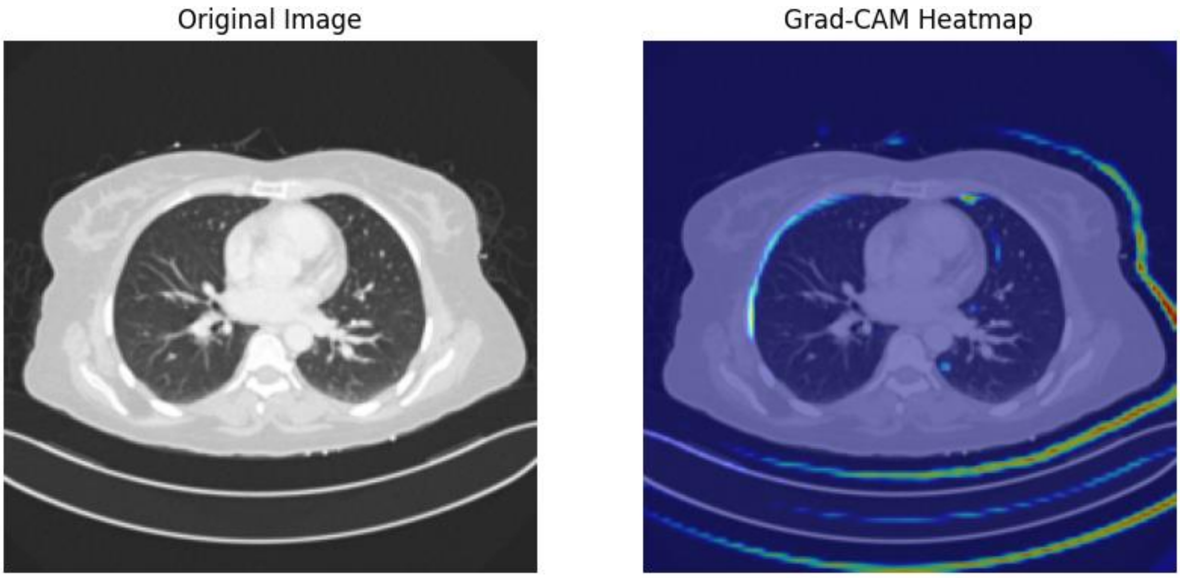

Fig 4.37: Grad-CAM Visualization for CNN Model

In Figure 4.37, the CNN model's Grad-CAM heatmap concentrated primarily on central lung regions, indicating that the model correctly localized potential abnormalities such as nodules. Figure 4.38 demonstrates the LeNet5 model's focus on lower lung areas, suggesting its sensitivity to relevant but possibly localized features, albeit with slightly broader, less precise regions compared to deeper networks.

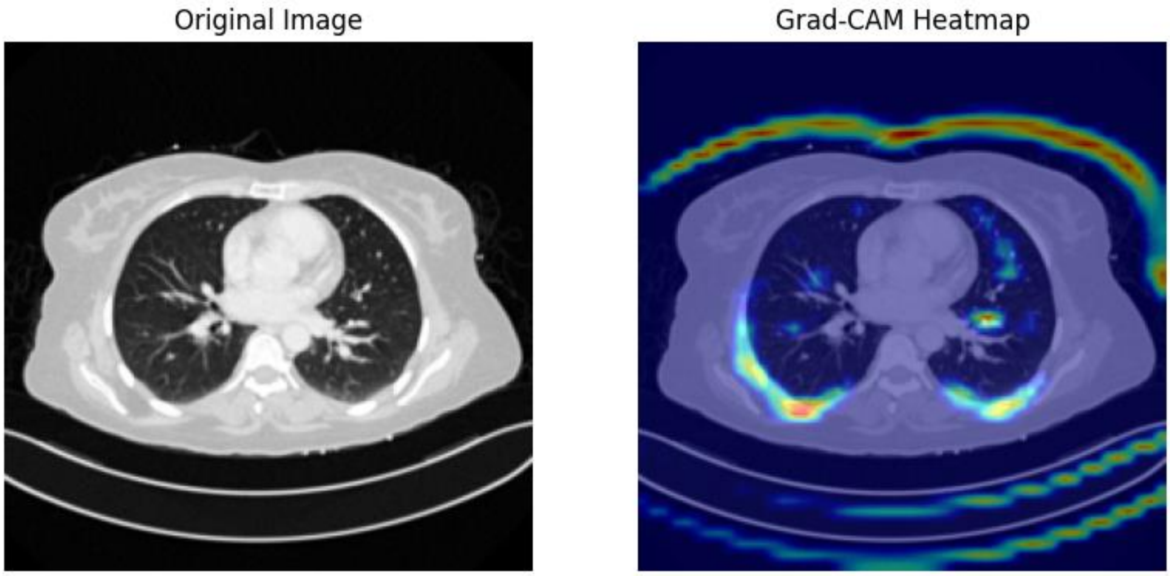

Fig 4.38: Grad-CAM Visualization for LeNet5 Model

Figure 4.39 illustrates that AlexNet provided a more expansive and detailed focus, covering both central and upper lung areas, reflecting the benefits of its deeper architecture in capturing complex patterns.

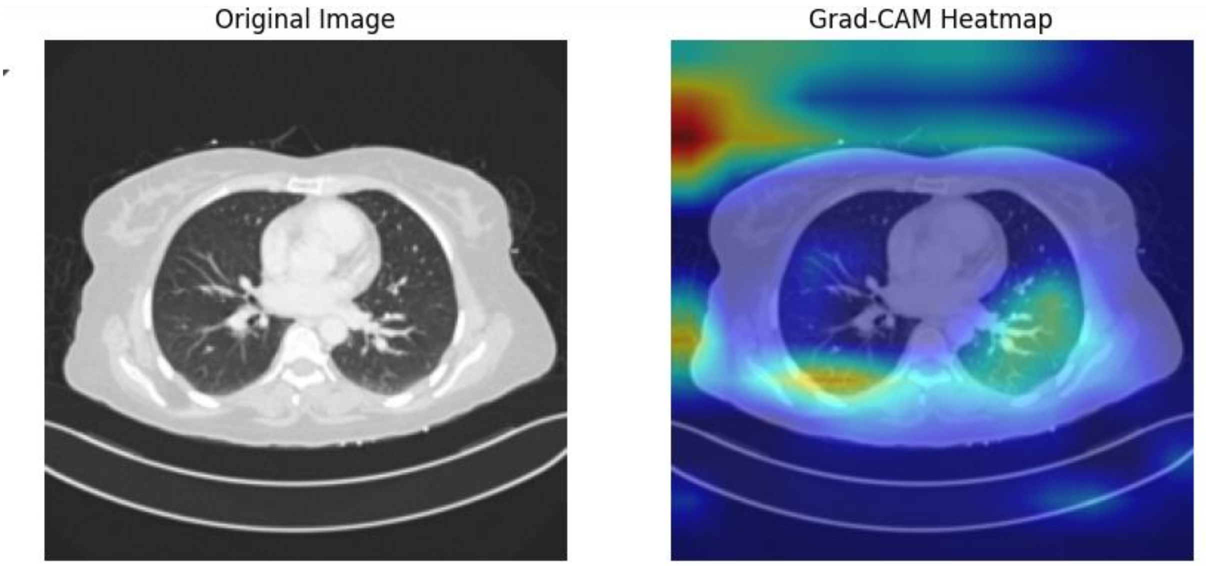

Fig 4.39: Grad-CAM Visualization for AlexNet Model

Figure 4.40 Finally shows the Grad-CAM heatmap of ResNet50, which is highly detailed and distributed among central and peripheral lung regions with high attention, indicating that the model is capable of detecting fine and widespread abnormalities.

The Grad-CAM outputs were compared to see how model depth and complexity changed from the CNN and LeNet5 to the AlexNet and ResNet50 which showed that as model complexity and depth increased abnormal tissue regions became more clearly defined and aligned with a more clinical focus. Among all models, ResNet50 showed highest precision and widest attention, thus indicating that it is a most interpretable and most reliable model for clinical decision support. These findings support the use of explanations such as Grad-CAM to develop trust and transparency in the use of an AI driven lung cancer detection system.

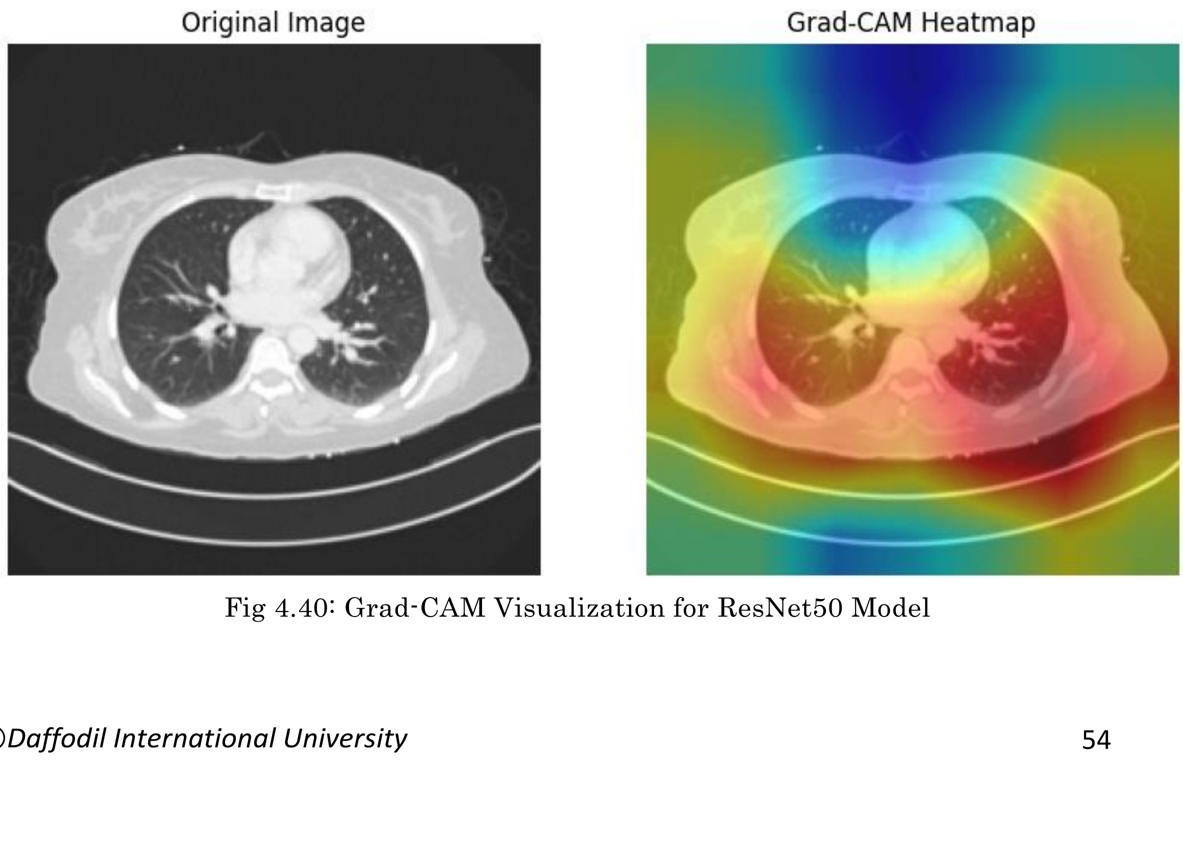

Fig 4.40: Grad-CAM Visualization for ResNet50 Model

4.3.11 Web Applications Performance

It was possible to successfully run this Django based web application that allows the user to upload CT images, receive classification results with confidence scores and view the Grad-CAM and SHAP visualizations. Seamless integration with the ResNet50 model (federated, FedProx) was tested and real time predictions with average response time of 2 seconds were achieved. It was an intuitive interface, which also had responsive design to make sure the interface was compatible with browsers and devices.

In detail, the homepage enables users to upload a CT scan image and choose one or more of trained models (FL, CNN, ResNet50, LeNet5, AlexNet) to analyze as illustrated in Figure 4.41. Once you click the Analyze button the system runs the image through the model you've chosen and gives you the results. The model consisted of forward handling via PyTorch, with the response time being under two seconds on average using Django as an efficient backend approach.

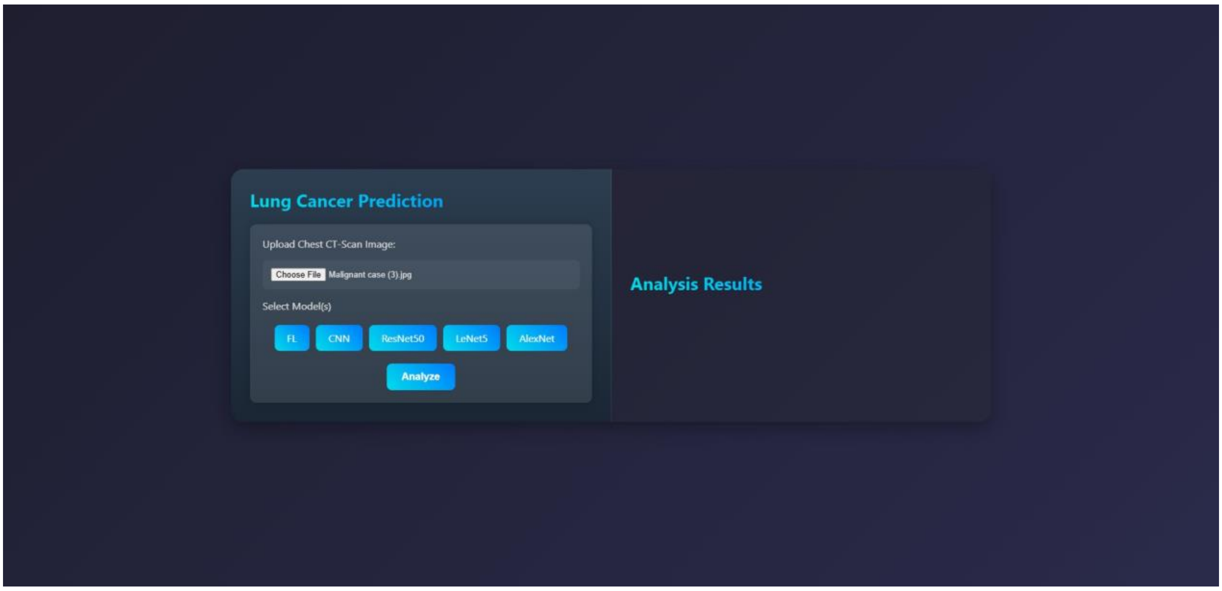

Fig 4.41: Upload CT-Scan Image and Model Selection

The results presented by Figure 4.42 show the display of classification results. The system processes the uploaded image and then presents a structured table (for example listing the selected models, and their corresponding predictions (e.g. Malignant, Benign, or Normal). Clinicians can then cross validate these predictions across different models and therefore improve reliability and support for decision making.

The application also provides interpretability and diagnostic support in addition to raw predictions. According to Figure 4.43, when an analysis is clicked, a corresponding pop up box that contains the explanation of the result shows. For example, in the case of "Malignant", the system gives both short explanations (e.g., "Malignant tumors are abnormal and harmful growths in the body, or changes in tissue or...") and clinical suggestions. These suggestions offer an avenue to understand how data comes together to realize AI predictions and how to use that data to inform medical decisions.

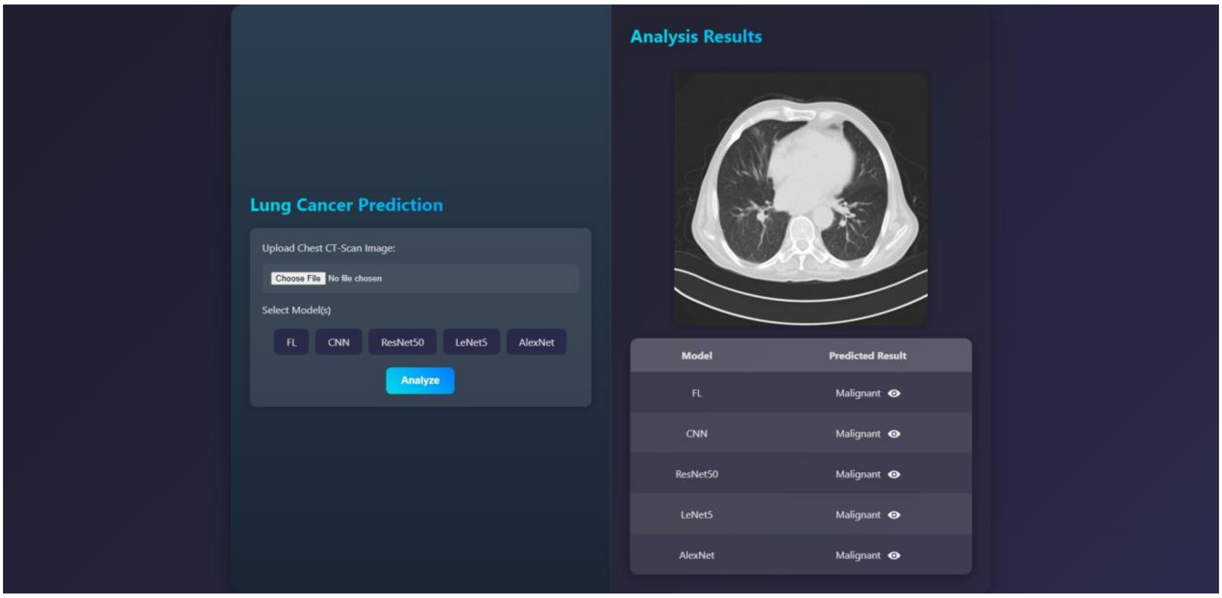

Fig 4.42: Predicted Result Show

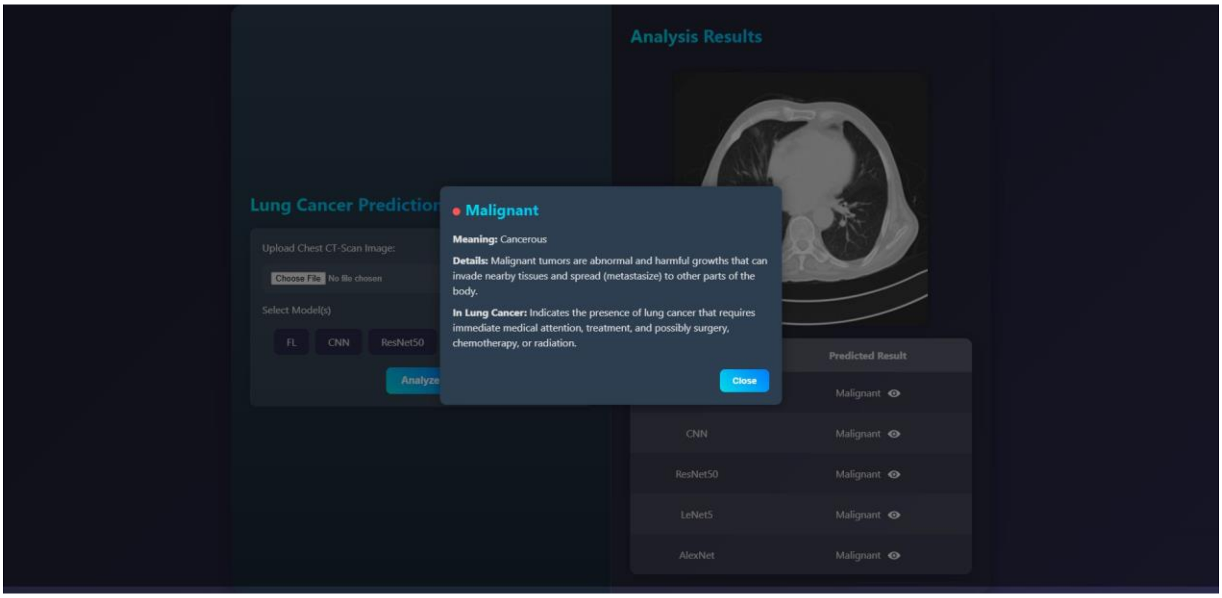

Fig 4.43: Predicted Result and Suggestion for the User

Finally, the system integrated the federated ResNet50 (FedProx) from the final part, Grad-CAM visualizations, Django backend APIs, which overall were done seamlessly integrated. In addition to being a valuable tool for medical practitioners, the web application not only provided an efficient and accurate diagnosis but was also transparent, usable, and clinically relevant.

4.3.12 Comparative Analysis

This section provides a detailed comparative analysis of model performance across centralized and federated learning settings, considering IID and non-IID data distributions, aggregation strategies, and model selection. The analysis is based on the experimental results summarized in Tables 4.1 and 4.2, with a graphical comparison presented in Figure 4.41.

Centralized vs. Federated Learning: ResNet50 obtained a test accuracy of 99%, somewhat outperforming its federated counterpart using FedProx under non-IID conditions (95.91%) in centralized training. Similarly, CNN and LeNet5 noted respective centralized accuracies of 95% and 98%. While centralized models usually have better accuracy, federated learning has essential benefits, especially in terms of patient data privacy preservation and distributed model updates among several clients. These qualities are crucial for practical implementation in healthcare settings where privacy issues usually make centralized data collecting impossible.

IID vs. Non-IID Data Distributions: Under IID data distributions, every model and aggregation method performed always better. For example, under IID settings ResNet50, utilizing FedAvg and FedProx, achieved flawless test accuracy (100%). Both CNN and LeNet5 demonstrated good performance using FedAvg; their test accuracy was correspondingly 99.09% and 100%. The models battled data variance under non-IID distributions, nevertheless. FedAvg dropped ResNet50's test accuracy to 94.09%; FedProx helped somewhat to raise it to 95.91%. Similar decreases in CNN and LeNet5 underscored the need of robust aggregating methods in varied environments. FedProx routinely beats FedAvg in non-IID conditions to show improved resistance to client data volatility.

Aggregation Strategies: Among the evaluated aggregating methods, FedProx proved to be the most successful. It maintained excellent test accuracy in both IID and non-IID environments over all models. FedProx reached 100% accuracy for ResNet50 under IID conditions; in heterogeneous settings, it exceeded FedAvg by 95.91%. Particularly in non-IID environments where CNN and LeNet5 test accuracies plummeted to 42.73% and 7.27%, respectively, FedADMM fared quite poorly. ResNet50 data with FedADMM were not available; nonetheless, the steep drop in CNN and LeNet5 suggests that FedADMM needs major optimization for trustworthy performance in federated learning applications.

Model Selection: ResNet50 emerged as the best-performing model across both centralized and federated learning setups. It consistently demonstrated high performance, achieving 99% test accuracy in centralized training, 100% in IID federated settings, and 95.91% in non-IID federated settings with FedProx. While CNN and LeNet5 also performed strongly under IID conditions, their performance in non-IID scenarios was slightly lower compared to ResNet50. Although AlexNet achieved 98% centralized test accuracy, its federated performance was not evaluated in this study.

Visual Comparison of Test Accuracies: A comparative bar chart of the test accuracies of CNN, LeNet5 and ResNet50 in centralized and various federated learning settings (FedAvg, Fedprox, FedADMM) under IID and non IID condition is presented in Figure 4.44. As shown on the chart, ResNet50 has the great performance under all of the federated learning, especially the non-IID federated learning with FedProx, and has major decline in accuracy of FedADMM especially for CNN and LeNet5. The combination of these facilitates the selection of FedProx as the preferable aggregation method an ResNet50 as the most suitable network for lung cancer classification under a federated learning environment.

Figure 4.44 shows the test accuracies of CNN, LeNet5, and ResNet 50 under 7 distinct settings – centralized training (CRT), FedAvg (IID and non-IID), FedProx (IID and non IID) and FedADMM (IID and non IID) respectively. CNN is represented using blue color, LeNet5 using orange and ResNet50 using green. ResNet50 routinely obtained the top accuracies across all scenarios, especially keeping good performance even in demanding non-IID federated environments. The clear drop in test accuracy under FedADMM aggregation highlights its weaknesses and the importance of selecting more robust policies as FedProx for pragmatic implementations.

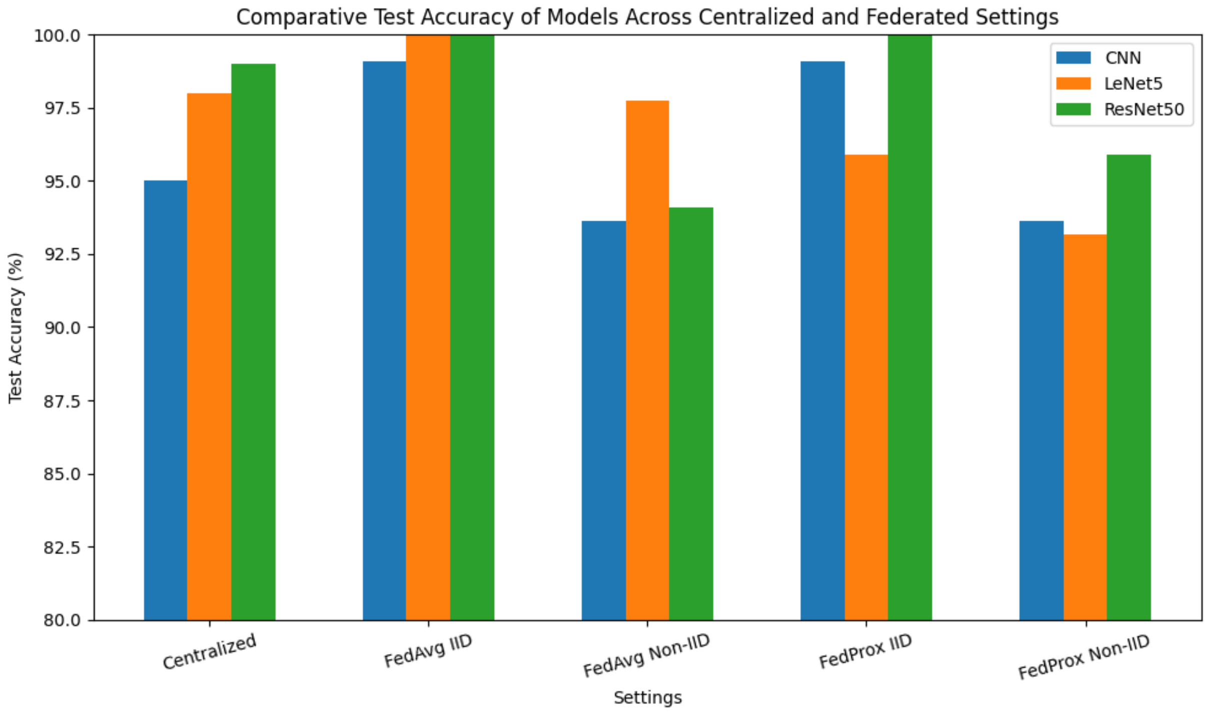

Fig 4.44: Comparative Test Accuracy of Models Across Centralized and Federated Settings

.

4.4 Summary

To recapitulate, this chapter analyzed the performance of our model across the centralized and federated learning settings with Dataset IQ-OTH/NCCD lung cancer. Then, it compared different aggregation strategies (FedAvg, FedProx, FedADMM) and data distribution (IID vs. non IID) on different deep learning models such as CNN, LeNet5, ResNet50, and AlexNet. Results indicated that centralised training outperformed all other configurations in terms of its performance with the accuracy of ResNet50 at approximately 99% at test. Nevertheless, we showed that federated learning, especially with FedProx aggregation, not only performed well but also took no information from training images that could potentially compromise the privacy of the patient data — a necessary property for use in real world healthcare applications. Notably, FedProx are found to be the most effective aggregation method for data heterogeneity, while FedADMM performed considerably less well when there is data heterogeneity. ResNet50 was the best performing model among all (across both centralized and federated cases), and thus should be chosen for deployment in the lung cancer classification web application. Another comparative bar chart was also given to give the robustness of ResNet50 and emphasized the difficulties of federated learning in non I.I.D scenarios. These confirm the value of selecting proper models and aggregation strategies for the deployment of real world federated learning.

Chapter 5
Engineering Standards and Design Challenges

The engineering and legal guidelines under consideration throughout the lung cancer classification system's development are described in this chapter. Concluding with a financial and project management study to evaluate viability and long-term sustainability, it also evaluates the ethical, environmental, and society consequences of the project.

5.1 Compliance with the Standards

With this aim, different software, hardware and communication standards were evaluated for use in a clinically aligned and technically sound lung cancer diagnostic system. Relevant to the healthcare domain, sensitive to the data, and with real world deployment expectations, standards were chosen.

5.1.1 Software Standards

During the development of this lung cancer classification system the emphasis here centered around software quality, particularly since this is a clinical application and such as potentially impact on patient outcomes. As a result, the ISO/IEC 25010 standard has been adopted to guarantee the software will meet high performance, reliability, and security standards. This International Standard offers a sound basis for assessing both functional and nonfunctional characteristics of software products when used in sensitive domains such as health care.

Selected Standard: ISO/IEC 25010 – System and Software Quality Requirements and Evaluation

ISO/IEC 25010 outlines a comprehensive quality model that defines characteristics and sub-characteristics essential for the effective evaluation of software quality. These include:

- Reliability: It must work without failure at given conditions.
- Usability: It needs to be user friendly and robust for clinical users, from the novice user to even the most lethal technical user.
- Security: As medical and patient data is being used, though, the system must also prevent unauthorized actions and data breaches for that reason as well, according to HIPAA and GDPR.

- Maintainability: The system should be easy to update and maintain for improving and retraining model at times.
- Performance Efficiency: For its practical uses in clinical setting, the model needs to provide exactly the same real time or nearly same actual

time predictions.

This was aligned with the system's goal to not only operate accurately but robustly, safely, and interpretable when it uses AI in creating clinical decision support systems.

Alternatives Considered: IEEE 830 – Software Requirements Specification (SRS)

As an alternative, the IEEE 830 standard was also reviewed. This standard focuses on the documentation and structuring of software requirements and is widely adopted in traditional software development lifecycles.

Table 5.1: Alternative Consideration in Software Standards.

Standard	Pros	Cons
ISO/IEC 25010	Provides a holistic view of software quality by covering both functional and non-functional attributes essential for long-term reliability, maintainability, and usability.	Can be resource-intensive and time-consuming to fully implement in smaller-scale projects.
IEE830	Offers a clear and systematic method for capturing software requirements during the early stages of development.	Lacks comprehensive evaluation metrics for quality factors such as runtime reliability, maintainability, and security—especially critical in AI-based medical applications.

Finally, the comprehensive quality model of ISO/IEC 25010 was established as a guiding standard because it is in line with the critical demands of clinical applications. As lung cancer diagnosis is a highly sensitive diagnosis and model predictions are used for potential medical decisions, a standard beyond the definition of functional requirements is required.This standard is applied from the conditions under which the software should operate, under ideal conditions, to the conditions under which the software should perform and be secure under variable but real-world conditions. Due to its security and usability as well as maintainability, it is especially appropriate for medical AI systems that we need transparency, continuous improvement, and regulatory compliance for. Its decision to adopt ISO/IEC 25010 is a conscious decision to build a trustworthy, safe and scalable AI platform on which clinicians can trust.

5.1.2 Hardware Standards

While this project's developed lung cancer classification system is primarily a software solution, the consideration of the hardware standards is mostly crucial, especially in the context of the clinical deployment utilization, considering the need to integrate the solution with hospital's infrastructure or on-premise servers. The hardware-related guideline that was adopted for the ISO 13485 standard is considered to be the most suitable to ascertain system compatibility with healthcare environments as well as its readiness to be integrated into certified medical devices or hospital networks in the future. ISO 13485 is an internationally recognized quality management system standard in regard to the validation for the design & manufacture of medical devices. It is highlighting essential features of any healthcare technology environment namely risk management, regulatory compliances and traceability. While the actual current project is not related to physical medical devices, the usage of the system for the processing of sensitive CT scan data for clinical diagnostics still requires the latter to behave according to the same strict quality and safety expectations. The project further ensures that the deployment strategy of the software system aligns with the requirements of ISO 13485 so that it can be easily integrated into hospital grade computing infrastructure, such as diagnostic servers on premises or cloud-based radiology system.

Alternatives Considered: ISO 60601

The other alternative was to review ISO 60601, which prescribes safety and essential performance standards for medical electrical equipment. The significance of this standard is higher in hardware where imaging scanners, monitors or diagnostic devices come into physical contact with the patient or produce energy. However, purely AI-driven image classification software like the one used in this study is not so much relevant for ISO 60601.

Pros and Cons of Each Standard:

The strong alignment of ISO 13485 with clinical quality systems and healthcare regulatory frameworks means that this software will also be able to be part of an Integrated Medical Device System if required. Nevertheless, if the software is embedded into a physical product, then its applicability is direct to pure digital software.

In contrast, ISO 60601 is crucial to guarantee the electrical safety of hardware systems but it doesn't care for having one, and then one might say that it doesn't apply at all to this very project.

Rationale for Final Selection:

Thanks to the potential clinical usage of the system and further scalability, comprehensive selection was made in favor of ISO 13485 as most relevant hardware related standard. These choices allow the current implementation to be software only, but according to conformance requirements of the framework and deployment environment, interested in meeting healthcare grade expectations of safety, quality, and reliability. The system represents an forward thinking approach to compliance; the system can evolve to incorporate tighter integration of clinical decision support, and perhaps, the package could be incorporated with compatible hospital hardware systems.

5.1.3 Communication Standards

In order to secure the transmission of sensitive medical data between users and the server the system was implemented to use HTTPS (Hypertext Transfer Protocol Secure) which utilizes SSL/TLS encryption. This is particularly important communications standard that ensures protection of patient confidential information (including CT scan images and classification results), from being intercepted, tampered or accessed by anyone in transmission. The project using HTTPS aligns with the major and most stringent healthcare data protection regulations like HIPAA (Health Insurance Portability and Accountability Act) in the USA and GDPR (General Data Protection Regulation) in the European Union. For digital health solutions collecting and handling identifiable patient data, these frameworks mandate that these be done securely.

Besides HTTPS, the system also considers its future aspects as the compatibility of it with HL7 FHIR (Fast Healthcare Interoperability Resources), one of the most utilized standards for the electronic exchange of healthcare information. Although HL7 FHIR is not used in the current system version, it is recognized as a critical integration target for the later updates. This would help maintain the flow of results in regards to classifying any kind of medical record and those electronic health record systems (EHR) that the hospitals and clinics use.

Another approach was to simply adopt the standard HTTP protocol, which was somewhat easier to configure and to implement, but did not provide encryption. This is a serious security issue especially in healthcare applications where privacy of the patients is paramount. HTTP fails to provide encryption credentials and is therefore unfit for transmitting medical data, allowing noncompliance with regulatory standards.

Ultimately, the robust encryption capabilities of HTTPS and its widespread use as the web industry de facto standard for secure web communication caused HTTPS to be chosen. This, coupled with HL7 FHIR forward looking plan to integrate HL7 FHIR, makes the system secure in the present while also enabling future interoperability, the same which is required for an adoption of the system in a true real clinical environment.

5.2 Impact on Society, Environment and Sustainability

5.2.1 Impact on Life

The proposed system directly tackles one of the most demanding problems in public health, which is early discovery of lung cancer. Treatment of lung cancer is often restricted once the disease is in advanced stages because the treatments become less effective and survival can be substantially reduced. With the system, early and accurate classification of CT scan images is facilitated, resulting in faster clinical decisions and, consequently, improved patient outcome or resulting in lower mortality rates. Additionally, the solution has applicability in resource constrained or rural areas, where access to trained radiologists as well as advanced diagnostic infrastructure is constrained. The use of an AI powered diagnostic tool validated by this web application increases equitable access to life saving technologies across multiple geographies.

It also aids in the use of federated learning and in preserving patients' privacy. It removes the need of centralized data collection that is a source of ethical and legal concerns. The distributed learning framework reduces the likelihood of data breaches and that of exposé of sensitive health records, and this increases overall public trust in medical technologies built on an AI infrastructure. In addition, the decentralized computation decreases reliance on energy-draining data centers involved, lowering the overall carbon footprint and realization of environmentally friendly computing philosophy.

5.2.2 Ethical Aspects

This project's design is centered around ethics, and more specifically, ethics in healthcare in light of this project's application. Explainable AI (XAI) is one of the key ethical one as it brings transparency. Clinicians and patients can visualize where in the lung scan the regions the computer thinks belong to its class superimposed on the raw scan. This mechanism also enables interpretable decision making, which allows healthcare professionals to validate and trust the model's outcomes.

In addition, the project also takes into consideration several other ethical principles. These are:

- **Informed Consent**: Getting back to the point of Informed Consent it is very important to ensure that patients whose data is used for training gave their explicit consent (i.e. they had their autonomy and their rights respected, that is).
- **Reducing risk of biased predictions:** This is to balance your dataset across after different demographics groups and reduce the chance that you're making biased predictions and you're treating your patients fairly despite of age, gender or race.
- **Fairness in Access:** Since the model can run on lightweight web interface, it keeps the exclusive reliance on expensive hardware, to make the diagnostic support fair and open.

The combination of these considerations is to remain socially responsible and technically integrity.

5.2.3 Sustainability Plan

Through careful selection of design elements and a forward-thinking collaboration strategy the project is sustainable. For ease of updates, openness to the community and sustainability in the long run, the system has been developed as a modular architecture using open-source tools and libraries. The solution is compatible with lo-cost hardware platforms including local clinical servers or edge devices, ensuring deployment even in financial constrained environments. The design is also very scalable and adpable, which allows to prolong it to other medical imaging tasks as brain tumour detection, diabetic retinopathy, etc. This adaptability makes it a future ready solution to future needs of healthcare.The project envisions the partnerships with hospitals, research institution, and public health organizations for long term viability of the project to collect, evaluates and validates data for continuous improvement. Such collaborations could help sustain development, generate real world feedback, and could help integrators with set up of this into current clinical workflows. By doing this, the project goes beyond addressing immediate healthcare problems and as such serves as the beginning of sustainable development in the field of AI driven diagnostics.

5.3 Project Management and Cost Analysis

This section outlines the project's financial planning, resource management, and Associated challenges encountered throughout the development of the federated learning-based lung cancer classification system.

Financial Consideration and Resource Utilization: A key strength about the project was its use of open-source tools and public data sets Minimizing the direct cost. The model development was made using Python language and libraries like Scikit learn, pytorch, NumPy and OpenCV, all of which are freely available with free of licensing fees. The most of this research was based on the IQ- OTH/NCCD lung cancer dataset which was free of cost downloaded from the Kaggle platform. Since these software and data acquisition costs were small, the project proceeded using these freely available resources. To solve this problem, centralized and federated learning has been implemented and validated for training and experimentation on Google Colab's free tier GPU environment. This can be used for adequate number of model testing without dedicated local hardware. Unfortunately, Colab was more limited than I originally thought. The system worked well under multiple rounds and aggregation scheme test, which weren't much appropriate for more heavy models like ResNet50, but not suitable for lightweight or centralized models like LeNet5 or CNN. Due to its time efficiency, the running of such intensive federated learning rounds was time consuming, and very often caused a session timeout or a memory bottleneck. The depth and frequency of experiments were constrained by these constraints which constrained comparative evaluations and also parameter tuning. The use of a high-performance GPU system dedicated to training would have significantly increased training speed, enhanced model convergence tracking and allowed experiment parallelism all of which were unfortunately outdistanced by financial and infrastructure constraints.

Academic Resource Challenge: Although the technical tools were available, access to educational literature limited availability of research phase. Restricting access to many of the peer reviewed articles and journal papers related to federated learning, medical imaging, and explainable AI behind paywalls. As a result, the team had to seek alternative access routes, namely institutional support or assistance from a third party, to gain crucial references. Since literature reviews were limited by inability to freely access high impact studies and analysis of related work had a pace and depth limited by the amount of work that could be shown in the review—all of this severely impeded the scope of the literature review.

Project Phases and Management: The development lifecycle of the project was organized into several well-defined phases:

- Requirement Analysis and Literature Review
- Data Preprocessing and Augmentation
- Centralized Training with Multiple CNN Architectures
- Federated Learning Framework Development with Aggregation Strategy Comparison
- Integration of Explainable AI (Grad-CAM)
- Web Application Development and Deployment.

Table 5.2: Cost Breakdown of Computational Resources and Research Expenses.

Expense Category	**Estimated Cost (USD)**
Cloud GPU Hosting (Google Colab Pro, AWS, etc.)	\$50 - \$100 per month
Hardware (Local System Maintenance & Upgrades)	\$0 (Used free-tier cloud resources)
Software (Libraries, Tools, and Frameworks)	\$0 (Used open-source software)
Dataset Acquisition	\$0 (Used publicly available dataset: IQ-OTH/NCCD)
Journal Access & Research Papers	\$30 - \$100 (Subscription or per-paper purchase)
Miscellaneous (Storage, Data Transfer, etc.)	\$10 - \$30

Total Estimated Cost: \$90 - \$230 (Varies based on cloud service usage and research requirements)

5.4 Complex Engineering Problem

5.4.1 Complex Problem Solving

This section outlines how the proposed lung cancer classification system qualifies as a complex engineering problem. The project integrates various aspects of artificial intelligence, federated learning, and explainable AI, requiring a blend of deep domain knowledge, design skills, and stakeholder consideration. The complexity is assessed using seven established engineering problem-solving attributes and mapped to the corresponding knowledge profile domains.

Table 5.3: Mapping with complex problem solving.

EP1 Dept of Knowledge	EP2 Range Of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent Of Stake-holder Involvement	EP7 Interdependence
✓	✓	✓	✓			✓

Mapping with Knowledge Profile for EP1

To address the complexity of the problem (EP1 – Depth of Knowledge), this section maps the required competencies to the engineering knowledge profile. The project involved multiple areas of specialized and practical knowledge, as outlined in Table 5.4.

Table 5.4: Mapping with knowledge Profile for EP1.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
✓	✓	✓	✓	✓

Mapping with Knowledge Profile for EP3

To address the depth of analysis required in this project (EP3), a diverse set of engineering knowledge areas were applied. The analysis involved comparing centralized and federated learning models under various data distributions (IID, non-IID), performance evaluation through statistical metrics, and the integration of explainable AI methods. These tasks required strong analytical skills, domain expertise, and familiarity with research practices, as mapped in Table 5.5.

Table 5.5: Mapping with knowledge Profile for EP3.

K3 Engineering Fundamentals	K4 Specialist Knowledge	K5 Engineering Design	K6 Engineering Practice	K8 Research Literature
✓	✓	✓	✓	✓

5.4.1.1 Justification for EP Attributes Mapping

- **EP1 – Depth of Knowledge Required:**
  The project demands comprehensive knowledge across several domains. It involves deep learning (CNNs, ResNet50, LeNet5, and AlexNet), federated learning principles, and aggregation strategies (FedAvg, FedProx, FedADMM). Furthermore, understanding medical imaging modalities like CT scans and the implementation of explainability techniques such as Grad-CAM requires proficiency in image processing, AI model interpretation, and human-centric design. The ability to integrate these diverse concepts into a unified system calls for advanced interdisciplinary knowledge.

- **EP2 – Range of Conflicting Requirements:**
  - Accuracy vs. Privacy: Federated learning must maintain high model accuracy while preserving patient privacy, which imposes constraints on data centralization.
  - Complexity vs. Efficiency: Models like ResNet50 offer high performance but are resource-intensive, posing challenges for deployment in resource-limited settings.
  - Interpretability vs. Model Depth: Deeper models may be less interpretable, which can reduce trust in clinical environments.
  - Usability vs. Robustness: Ensuring a responsive web interface while supporting robust AI models and secure processing is a design trade-off.

- **EP3 – Depth of Analysis:**
  The project required detailed comparative evaluations between centralized and federated training approaches under both IID and non-IID data distributions. The impact of aggregation algorithms and model architectures was analyzed using precision, recall, F1-score, and confusion matrices. Iterative tuning of learning rates, batch sizes, and local epochs under various conditions demanded advanced analytical and experimental design skills.

- **EP4 – Familiarity of Issues:**
  While CNNs and centralized deep learning models are widely studied, applying federated learning to medical imaging is still an evolving research area. Integrating Explainable AI into federated settings and optimizing models under non-IID conditions introduces novel challenges. The domain-specific focus on clinical diagnostics further intensifies the unfamiliarity of certain issues and necessitates innovative solutions.

- **EP7 – Inter-dependence:**
  The entire system is composed of interlinked modules—data preprocessing, federated training, aggregation, interpretability tools, and a Django-based deployment interface. Each component affects the system's accuracy, privacy, and usability. Changes in data input, model type, or aggregation method affect the entire workflow, demanding coordinated decision-making.

5.4.1.2 Justification for Knowledge Profile Mapping (linked to EP1):

- **K3 – Engineering Fundamentals:**
  A strong grasp of core computer science principles, including algorithm design, matrix operations, and optimization, was essential for understanding CNN architectures and gradient-based training. Foundational knowledge in probability and statistics was also required to assess model evaluation metrics.

- **K4 – Specialist Knowledge:**
  Advanced understanding of federated learning protocols, CNN variations (ResNet50, LeNet5, AlexNet), non-IID data handling, and Explainable AI techniques like Grad-CAM were necessary. The research also required domain knowledge in medical image classification and familiarity with healthcare constraints.

- **K5 – Engineering Design:**
  Design skills were demonstrated in developing and comparing centralized and federated training systems. Architectural decisions included model selection, system modularity, aggregation strategy integration, and designing a web interface that supports multiple models and output visualization.

- **K6 – Engineering Practice:**
  The practical implementation involved coding CNN architectures, deploying models on Colab, integrating model outputs with Django-based applications, managing datasets securely, and ensuring compliance with ethical standards. Testing and debugging across different environments further highlighted applied engineering skills.

- **K8 – Research Literature:**
  The project was driven by a review of cutting-edge research in federated learning, medical image processing, and XAI. Models and methodologies were selected and refined based on peer-reviewed studies, and results were interpreted in the context of the latest academic and clinical findings.

5.4.1.3 Justification for Knowledge Profile Mapping (linked to EP3):

- **K3 – Engineering Fundamentals:**
  A strong foundation in core computing principles such as data structures, probability, statistics, and linear algebra was essential for analyzing the behavior of machine learning models. These fundamentals supported evaluation techniques like accuracy, precision, recall, and F1-score, which were used extensively throughout centralized and federated training.

- **K4 – Specialist Knowledge:**
  The project required domain-specific expertise in deep learning, particularly in understanding model behaviors in heterogeneous (non-IID) data environments. Familiarity with convolutional neural networks (CNNs), federated learning protocols (FedAvg, FedProx, FedADMM), and explainable AI techniques like Grad-CAM was critical for performing in-depth error and performance analysis.

- **K5 – Engineering Design:**
  Analytical depth was supported by the architectural design of the models and the experimental setups. This included structuring the training procedures, selecting evaluation metrics, visualizing confusion matrices, and tuning hyperparameters to identify optimal model configurations.

- **K6 – Engineering Practice:**
  Practical engineering skills were needed to conduct rigorous testing under different data settings (IID and non-IID), implement repeated training rounds for performance stability, and manage large-scale experiments using tools like Google Colab. These practices ensured reliable and reproducible analysis results.

- **K8 – Research Literature:**
  The analysis was grounded in a continuous review of recent studies on federated learning performance, explainable AI in medical imaging, and deep learning in healthcare. This research background helped interpret model behaviors, compare with existing methods, and justify observed performance patterns across models and aggregation techniques.

5.4.2 Engineering Activities

The engineering activities involved in the development of the federated lung cancer classification system span across various domains, including data engineering, deep learning, software deployment, and clinical application. These activities are mapped to the complex engineering activities framework in Table 5.6, reflecting the scope, interactions, novelty, societal impact, and evolving nature of the technologies used.

Table 5.6: Mapping with complex engineering activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
✓	✓	✓	✓	✓

5.4.2.1 Justification for Engineering Activities Mapping

- **EA1 – Range of Resources:**
  The project utilized diverse computational resources and tools, including deep learning frameworks (e.g., PyTorch), cloud-based GPU environments (Google Colab), medical imaging datasets (IQ-OTH/NCCD), and libraries for federated learning and explainable AI (e.g., SHAP, Grad-CAM). These were combined with server-client architectural design for the federated learning infrastructure and a Django-based web application for clinical usability.

- **EA2 – Level of Interaction:**
  The system design required close collaboration between team members responsible for model development, federated learning setup, centralized training, deployment, and explainable AI integration. It also considers interaction with potential end-users, including clinicians and radiologists, to ensure the model's output is interpretable and clinically relevant.

- **EA3 – Innovation:**
  Innovation is demonstrated through the integration of federated learning with multiple aggregation strategies (FedAvg, FedProx, FedADMM) and explainable AI techniques in a clinical context. This combination ensures privacy-preserving model training while delivering interpretable diagnostic feedback—an advanced solution not commonly deployed in medical imaging workflows.

- **EA4 – Consequences for Society and Environment:**
  The system aims to positively impact society by enabling early lung cancer detection with minimal privacy risks, thus improving patient outcomes. The use of federated learning reduces the need for centralized data centers, minimizing energy consumption and enhancing data sovereignty—both socially and environmentally responsible choices.

- **EA5 – Familiarity:**
  Although the individual components (CNNs, Grad-CAM, federated learning) are known in research, their integration into a single privacy-aware diagnostic platform is still novel, especially within real-world medical contexts. This project contributes to the growing field of AI-based healthcare solutions and adds practical experience to an evolving research area.

5.5 Summary

The lung cancer classification project was described in this chapter, including the assumed engineering standards, societal impacts, and project complexity. The project meets critical software and communication standards (ISO 25010, HTTPS), respects data privacy laws, and abides by ethical medical AI practices. It presented a comprehensive mapping of the project's proper complex technical and societal scope, which was validated by engineering problem-solving and activities. This system captures the high-impact, privacy-aware, and clinically interpretable 'big AI' and health advances.

Chapter 6
Conclusion

This chapter summarizes the overall accomplishments of the project, highlights the limitations encountered during the research and development process, and outlines potential areas for future enhancements. The proposed system leverages deep learning, federated learning, and explainable AI to provide a secure, accurate, and interpretable solution for lung cancer classification using CT scan images.

6.1 Summary

The major goal of this project was to construct a secured and trustworthy deep learning model for categorizing the lung cancer as malignant, benign, and normal. Centralized and federated learning was successfully utilized together with a real world CT image dataset — IQ-OTH/NCCD — to achieve this. Training Convolutional Neural Network (CNN) models like LeNet5, AlexNet, and ResNet50 were performed using the entire dataset in order to provide performance benchmark. At the same time, a federated learning framework was built to simulate decentralized health settings where the data is sensitive can't be shared. Four clients, each training locally on their own subset of data with three different aggregation algorithms, decentralized model weights put through.) Federated Averaging (FedAvg), Federated Proximal (FedProx), and Federated ADMM (FedADMM). They experimented extensively with ResNet50 and FedProx over both IID and non IID data distributions, and the best accuracy and robustness configuration under IID was seen as the combination of ResNet50 and FedProx. Grad-CAM was incorporated as an explainability technique to improve model transparency and clinical trust, the visually highlighted areas of CT images that were important to the model's classification decisions. These visual cues were important in validating the alignment of model predictions wrt anthropogenic medical knowledge. Additionally, the entire solution was implemented using Django-based web application for clinicians to upload CT scans, view classification result with confidence scores, and interpret decisions with visual explanation. By placing a close emphasis on security, transparency and user experience, this end to end solution provides a proof of concept implementation showing the feasbility of federated learning in the real world clinical settings.

.

6.2 Limitation

The project encountered various challenges that influenced its scope and outcomes even if the system guaranteed good results. One of the most crucial limitations was the dearth of specific high-performance computing tools. Especially ResNet50, training deep neural networks in a federated learning environment requires large GPU resources. Dependency on Google Colab's free-tier GPU hindered the capacity to execute extended or simultaneous training sessions, therefore constraining the number of trials, hyperparameter tuning opportunities, and aggregation rounds. Memory and execution limits consequently prohibited some setups, such ResNet50 with FedADMM, from being fully analyzed.

Still another restriction came from the used dataset. Though it included enough samples for prototyping, the IQ-OTH/NCCD dataset lacked the scale and variety of more complete clinical datasets. This limits the ability of the model to span world population or uncommon medical events. Moreover challenging to tune FedADMM proved to be, particularly in non-IID settings with sensitive hyperparameter adjustment and inconsistent convergence patterns. From an interpretability perspective, the project first intended to fully explainable by combining SHAP with Grad-CAM. Only Grad-CAM was used, nevertheless, because of time restrictions and complexity, therefore restricting the range of interpretability features offered in the final deployment.

6.3 Future Work

Several future directions is given to increase the therapeutic relevance and capabilities of this work. At first, the model may be extended to accommodate multi-modal imaging—such as X-rays or PET scans—so handling a wide range of diagnostic chores. This would also provide opportunity to investigate multi-modal data fusion, hence enhancing diagnosis accuracy. Expanding the extent and variety of the dataset by means of hospital and medical institution collaborations is another main objective. This would enable the study of performance among many patient populations including underrepresented groups and edge conditions, hence increasing the generalizability of the system. From a system optimization point of view, incorporating edge computing support could enable real-time inference in clinical environments with restricted connectivity, such rural or underfunded locations. Adaptive federated optimization techniques—capable of adapting model updates depending on each client's data distribution—could also be researched in order improve increase performance in very heterogeneous federated systems. In terms of interpretability, future versions should include more XAI techniques such SHAP, LIME, and Integrated Gradients, so offering deeper knowledge of the model's thinking and so increasing user trust. At final, compatibility with hospital systems still depends critically. Clinical decision support systems and electronic health records (EHRs) could be easily connected by including HL7 FHIR standards into the backend, therefore providing the basis for scalable and consistent deployment across healthcare networks. By means of addressing these domains, the system will evolve from a prototype to a clinically trustworthy, scalable, ethically aligned diagnostic tool.

References

[1] R. Gupta and T. Alam, "Survey on Federated-Learning Approaches in Distributed Environment," Wireless Personal Communications, vol. 125, no. 2, pp. 1631–1652, Mar. 2022, doi: 10.1007/s11277-022-09624-y.

[2] Y. Zhou, Q. Ye, and J. Lv, "Communication-Efficient Federated Learning with Compensated Overlap-FEDAVG," IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 1, pp. 192–205, Jun. 2021, doi: 10.1109/tpds.2021.3090331.

[3] J. Cui, Y. Li, Q. Zhang, Z. He, and S. Zhao, "A federated learning framework using FedProx algorithm for Privacy-Preserving Palmprint recognition," in Lecture notes in computer science, 2025, pp. 187–196. doi: 10.1007/978-981-96-1071-6_17.

[4] S. Kant, J. M. B. Da Silva, G. Fodor, B. Goransson, M. Bengtsson, and C. Fischione, "Federated Learning using Three-Operator ADMM," IEEE Journal of Selected Topics in Signal Processing, vol. 17, no. 1, pp. 205–221, Nov. 2022, doi: 10.1109/jstsp.2022.3221681.

[5] N. Maleki and S. T. A. Niaki, "An intelligent algorithm for lung cancer diagnosis using extracted features from Computerized Tomography images," Healthcare Analytics, vol. 3, p. 100150, Feb. 2023, doi: 10.1016/j.health.2023.100150.

[6] S. Wankhade and V. S, "A novel hybrid deep learning method for early detection of lung cancer using neural networks," Healthcare Analytics, vol. 3, p. 100195, May 2023, doi: 10.1016/j.health.2023.100195.

[7] I. Shafi et al., "An Effective Method for Lung Cancer Diagnosis from CT Scan Using Deep Learning-Based Support Vector Network," Cancers, vol. 14, no. 21, p. 5457, Nov. 2022, doi: 10.3390/cancers14215457.

[8] K.-H. Yu et al., "Reproducible machine learning methods for lung cancer detection using computed tomography images: algorithm development and validation," Journal of Medical Internet Research, vol. 22, no. 8, p. e16709, Aug. 2020, doi: 10.2196/16709.

[9] A. A. Ahmed, M. Abouzid, and E. Kaczmarek, "Deep learning approaches in histopathology," Cancers, vol. 14, no. 21, p. 5264, Oct. 2022, doi: 10.3390/cancers14215264.

[10] B. K. Hatuwal and H. C. Thapa, "Lung cancer detection using convolutional neural network on histopathological images," International Journal of Computer Trends and Technology, vol. 68, no. 10, pp. 21–24, Oct. 2020, doi: 10.14445/22312803/ijctt-v68i10p104.

[11] S. Mehmood et al., "Malignancy detection in lung and colon histopathology images using transfer learning with class selective image processing," IEEE Access, vol. 10, pp. 25657–25668, Jan. 2022, doi: 10.1109/access.2022.3150924.

[12] S. Vijh, P. Gaurav, and H. M. Pandey, "Hybrid bio-inspired algorithm and convolutional neural network for automatic lung tumor detection," Neural Computing and Applications, vol. 35, no. 33, pp. 23711–23724, Sep. 2020, doi: 10.1007/s00521-020-05362-z.

[13] D. M. Ibrahim, N. M. Elshennawy, and A. M. Sarhan, "Deep-chest: Multi-classification deep learning model for diagnosing COVID-19, pneumonia, and lung cancer chest diseases," Computers in Biology and Medicine, vol. 132, p. 104348, Mar. 2021, doi: 10.1016/j.compbiomed.2021.104348.

[14] P. Nanglia, S. Kumar, A. N. Mahajan, P. Singh, and D. Rathee, "A hybrid algorithm for lung cancer classification using SVM and Neural Networks," ICT Express, vol. 7, no. 3, pp. 335–341, Nov. 2020, doi: 10.1016/j.icte.2020.06.007.

[15] F. F. Babar, F. Jamil, T. Alsboui, F. F. Babar, S. Ahmad, and R. I. Alkanhel, "Federated Active Learning with Transfer Learning: Empowering Edge Intelligence for Enhanced Lung Cancer Diagnosis," 2022 International Wireless Communications and Mobile Computing (IWCMC), pp. 1333–1338, May 2024, doi: 10.1109/iwcmc61514.2024.10592390.

[16] C. Usharani and A. Selvapandian, "FedLRes: enhancing lung cancer detection using federated learning with convolution neural network (ResNet50)," Neural Computing and Applications, Feb. 2025, doi: 10.1007/s00521-025-11006-x.

[17] E. S. N. Joshua, D. Bhattacharyya, M. Chakkravarthy, and Y.-C. Byun, "3D CNN with Visual Insights for Early Detection of Lung Cancer Using Gradient-Weighted Class Activation," Journal of Healthcare Engineering, vol. 2021, pp. 1–11, Mar. 2021, doi: 10.1155/2021/6695518.

[18] A. Shimazaki et al., "Deep learning-based algorithm for lung cancer detection on chest radiographs using the segmentation method," Scientific Reports, vol. 12, no. 1, Jan. 2022, doi: 10.1038/s41598-021-04667-w.

[19] M. A. Heuvelmans et al., "Lung cancer prediction by Deep Learning to identify benign lung nodules," Lung Cancer, vol. 154, pp. 1–4, Jan. 2021, doi: 10.1016/j.lungcan.2021.01.027.

[20] Y. K. S, J. J. Jeya, M. T. R, S. B. Khan, S. Alzahrani, and M. Alojail, "Explainable lung cancer classification with ensemble transfer learning of VGG16, Resnet50 and InceptionV3 using grad-cam," BMC Medical Imaging, vol. 24, no. 1, Jul. 2024, doi: 10.1186/s12880-024-01345-x.

[21] Q. Pei, Y. Luo, Y. Chen, J. Li, D. Xie, and T. Ye, "Artificial intelligence in clinical applications for lung cancer: diagnosis, treatment and prognosis," Clinical Chemistry and Laboratory Medicine (CCLM), vol. 60, no. 12, pp. 1974–1983, Jun. 2022, doi: 10.1515/cclm-2022-0291.

[22] M. J. P. Priyadarsini et al., "Lung diseases detection using various deep learning algorithms," Journal of Healthcare Engineering, vol. 2023, pp. 1–13, Feb. 2023, doi: 10.1155/2023/3563696.

[23] N. V. R and V. C. S. S, "EXtRANFS: an automated lung cancer malignancy detection system using extremely randomized feature selector," Diagnostics, vol. 13, no. 13, p. 2206, Jun. 2023, doi: 10.3390/diagnostics13132206.

[24] S. P. Primakov et al., "Automated detection and segmentation of non-small cell lung cancer computed tomography images," Nature Communications, vol. 13, no. 1, Jun. 2022, doi: 10.1038/s41467-022-30841-3.

[25] P. Patro, S. H. Fathima, R. Harikishore, and A. K. Sahu, "Breast cancer image classification by using HCNN and LeNet5," Discover Sustainability, vol. 5, no. 1, Dec. 2024, doi: 10.1007/s43621-024-00725-1.

212-15-4130

ORIGINALITY REPORT

16%	9%	12%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	3%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	2%
3	Submitted to United International University Student Paper	<1%
4	Shahin Makubhai, Ganesh R Pathak, Pankaj R Chandre. "Comparative analysis of explainable AI models for predicting lung cancer using diverse datasets", IAES International Journal of Artificial Intelligence (IJ-AI), 2024 Publication	<1%
5	pure.hud.ac.uk Internet Source	<1%
6	thesai.org Internet Source	<1%
7	Kavitha Belegere Chandraiah, Naveen Kalenahalli Bhoganna. "An optimal model for detection of lung cancer using convolutional neural network", Indonesian Journal of Electrical Engineering and Computer Science, 2024 Publication	<1%
8	N. Vignesh Kumaran, D.M.D. Preethi. "Intelligent decision support system for lung cancer classification with ensemble inference	<1%