INFORMATICS INSTITUTE OF TECHNOLOGY

In Collaboration with

UNIVERSITY OF WESTMINSTER

University of Westminster, Coat of Arms

NotionFL: An Explainable Mediator for Trustworthy Cross-Silo Federated Learning

A dissertation by

Mr. Ratheshan Sathiyamoorthy

w1809947 – 20200170

Supervised by

Ms. Krishnakripa Jayakumar

April 2024

Submitted in partial fulfillment of the requirements for the BSc (Hons) Computer Science degree at the University of Westminster

ABSTRACT

In recent years, cross-silo Federated Learning (FL) has gained significant attention in both industry and academia due to its ability to preserve privacy. Essentially, cross-silo FL promotes organizations to collaborate on Machine Learning (ML) tasks while keeping their data secure. However, these organizations in the same domain can be competitive in nature which makes them hesitant to collaborate and affects the client cooperation. Additionally, since these environments lack a designated server and involve competitive entities, they may worsen issues related to trust and interpretability. Unfortunately, these issues align with the inherent "black-box" nature of FL, which can result in low client cooperation, biased decision making, and decreased effectiveness of the cross-silo FL system.

To address this research gap, the author proposes a novel architectural flow that aims to improve the trustworthiness and interpretability in cross-silo FL environments. This includes integrating Trustworthy AI principles and creating explainable mediator like mechanism that can provide human-interpretable explanations of the FL server's decision-making process. The explainable mechanism will utilize Shapley Values model agnostic explainer approach by customizing it specifically to interpret the FL Workflows. Furthermore, the author was inspired to develop this proposed system as an agnostic framework which can be used by different organizations in various use cases.

Based on the initial results, it's clear that the proposed solution effectively addresses the challenges of trustworthiness and explainability in cross-silo FL environments. The integration of Trustworthy AI principles and the explainable mechanism enhances the client's trust by fostering effective client cooperation and system transparency in FL workflows. This research contributes significantly to the field by providing a unique solution tailored for cross-silo FL, and exhibited to the professionals who all agreed that the project results justify and addresses the problem.

Subject Descriptors

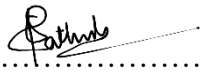
- Computing methodologies → Machine learning → Machine learning approaches
- Security and privacy

Keywords: Federated Learning, Cross-silo Federated Learning, Trustworthy AI, Explainable AI, Privacy-Preserving Machine Learning, Differential Privacy, Shapley Value

DECLARATION

I, Ratheshan Sathiyamoorthy, hereby affirm that this thesis is the result of my independent research efforts. No part of this dissertation has previously been submitted for any diploma, degree, or similar qualification. All information sourced from external, reliable sources has been properly recognized and cited.

Student Name: Ratheshan Sathiyamoorthy
Registration No: w1809947 / 20200170

Signature:  **Date**: 04/04/2024

ACKNOWLEDGEMENT

Completing this research project was both challenging and fulfilling, demanding extensive hours of research, documentation, development, and testing over eight months. I extend my deepest gratitude to all who supported me on this journey.

Firstly, I want to thank my supervisor, Ms. Krishanakripa, for her unwavering support, guidance, and encouragement. Her expertise was invaluable in ensuring the competency of my research. I also appreciate our module leader, Mr. Guhanathan Poravi, for his continuous advice and insights since the project's initiation. Thanks to all lecturers who shared their knowledge over the years, contributing to the completion of this project. I'm also grateful to the domain experts and evaluators for their feedback, which helped refine my work.

Lastly, heartfelt thanks to my friends, family, and siblings, who have been my unwavering support throughout the entire university life, especially during this final year project. Their encouragement and belief in me were instrumental in achieving this milestone.

TABLE OF CONTENTS

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ML Machine Learning

AI Artificial Intelligence

FL Federated Learning

XAI Explainable Artificial Intelligence

IoT Internet of Things

TAI Trustworthy Artificial Intelligence

TFL Trustworthy Federated Learning

SV Shapley Values

LR Literature Review

DP Differential Privacy

UI User Interface

GPU Graphics Processing Unit

DL Deep Learning

DFD Data Flow Diagram

CHAPTER 01: INTRODUCTION

1.1 Chapter Overview

The research project intends to present a novel approach to address the limitations in the current methodologies within the emerging global paradigm of Federated Learning (FL). This chapter introduces the reader with an overview of the problem domain, research gap, research aims, objectives, and contributions. It also highlights the novelty of the research and outlines essential proofs to justify the research gap. Finally, the research questions were developed along with the discussion of potential challenges, making the chapter serve as a comprehensive guide.

1.2 Problem Domain

1.2.1 Federated Learning

Over the years, Artificial Intelligence (AI) and Machine Learning (ML) have shown their true potential in every industry and influenced daily lives. However, these advancements ended up increasing privacy and security concerns, resulting in the creation of new laws and regulations against the traditional ML workflows (Asad, Moustafa and Ito, 2021). These data preservation concerns in traditional ML approaches have eventually led to the adoption of the novel paradigm known as Federated Learning (FL).

In a FL setting, many clients collaboratively train a model under the orchestration of a central server while keeping their raw data secured (Kairouz et al., 2021). The centralized server sets the model parameters in the global model and starts the initial training process. Each client receives the global model and trains the model with their local training data. Once the training is over, the client sends the model updates to the central server, and it aggregates the received model updates to the global model. After the aggregation, the newly obtained global model is sent back to the peers again and this process will be repeated until the model converges (Haffar, Sánchez and Domingo-Ferrer, 2023). Since data never leaves the peers' devices, data privacy is preserved and unlike traditional ML settings, only the encrypted model updates are shared instead of the raw training data. Although FL empowers with certain benefits from its counterparts, it also falls short in many challenges that arise due to the exponential domain growth.

1.2.2 Categorization of Federated Learning

FL can be categorized into two main different types based on the participating clients and training scale: cross-device and cross-silo setting. The **cross-device** FL setting is a powerful technique that allows clients with small data amounts to train a global model while keeping their data private (Tariq et al., 2023). These clients can be mobile or IoT devices. However, edge devices face numerous challenges such as limited computation, connectivity, and communication (Kairouz et al., 2021). Hardware limitations often prevent clients from participating in long-term training, which compromises their accountability and makes them vulnerable to anomalies and attacks (Kairouz et al., 2021).

In **cross-silo** FL, the number of participants are small compared to cross-device and the clients are typically companies or organizations with the same goal of collaboratively training a global model (Huang, Huang and Liu, 2022). For instance, multiple hospitals might train a diagnostic model and different banks can collaborate together to train a fraud detection model (Kairouz et al., 2021).

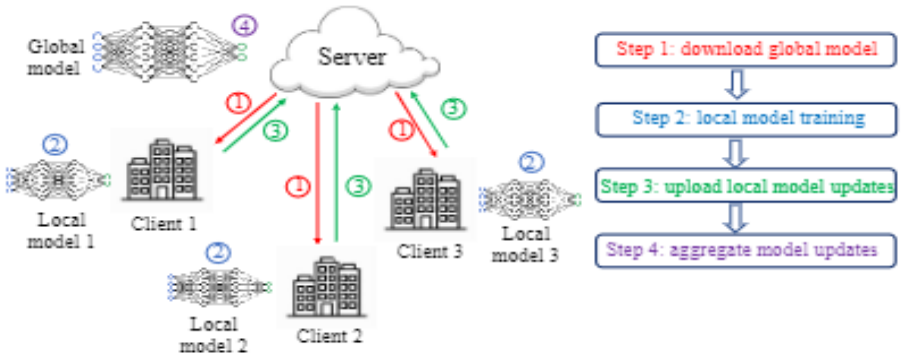


Figure 1:Cross-silo Federated Learning Process (Huang, Huang and Liu, 2022)

It is usually utilized within organizations with the same business goals while sharing incentives to train a global model with their own data (Kairouz et al., 2021). Cross-silo clients are more engaging and reliable throughout the training process due to having good connectivity and high-performance hardware resources. Since there isn't a proper central server or administrator, cross-silo FL setting often relies on a trusted third-party actor for carrying out the FL tasks which can eventually cause different issues (Q Li et al., 2023).

1.2.3 Effectiveness and Efficiency of Federated Learning

Effectiveness and efficiency of FL is often concerned with training a model using statistical and system heterogeneity of **client participation** and the **data quality** (Huang, Huang and Liu, 2022). Typically, cross-device settings are unlikely to form long-term cooperation due to the client resource constraints but, organizations in cross-silo usually have long-term strategic plans and goals between them to ensure continuous participation (Ratnayake, Chen and Ding, 2023). However, cross-silo clients may also become business competitors from the same domain which makes them hesitant to collaborate due to biased and untrustworthy systems. Therefore, incentive mechanisms and trustworthy systems were established to influence active client collaboration for better effectiveness (Kairouz et al., 2021; S'anchez et al., 2023).

1.2.4 Trustworthy Federated Learning

Similar to data privacy, trustworthiness has become a critical aspect influencing AI systems and drawing legal, ethical, social laws and regulations (S'anchez et al., 2023). Trustworthy AI is an emerging concept in Responsible AI that considers robustness, interpretability, explainability, fairness, privacy, and accountability as key pillars to create reliable AI systems (S'anchez et al., 2023; Tariq et al., 2023).

As a promising AI framework, FL provides robust privacy preservation but struggles to achieve trustworthiness in crucial areas of FL workflow (Lo et al., 2023; S'anchez et al., 2023). For instance, untrustworthy contribution evaluation could potentially foster clients to leave the training impacting the client selection, reliability, and fair reward distribution (Tariq et al., 2023). Similarly, untrustworthy secure aggregation might lead to vulnerabilities, underscoring the need for reliable servers (Tariq et al., 2023). Further, enhancing **explainability** in the context of trustworthiness can address the bias and promote fairness in ML systems (Tariq et al., 2023). However, lack of explainability in FL systems is a major setback that fails to determine if clients are treated fairly, receive explanations on server decisions, and allow client debugging (Kairouz et al., 2021; Tariq et al., 2023). This highlights the need for research focused on developing mechanisms that enhance explainability and trust in FL systems, encouraging long-term client engagement.

1.3 Problem Definition

In recent years cross-silo FL has received significant attention in popular industries, organizations, and in the field of academia. As defined already, the effectiveness of the FL system critically depends on the active participation of the clients but, the untrustworthy architecture and the competing nature of clients in cross-silo FL makes it risk of collapsing. These participants are often business competitors with conflicting interests, potentially hesitating to leave the training and affecting the effectiveness of the FL (Kairouz et al., 2021).

Additionally, since FL itself is a 'Black-box', the lack of **explainability** leads to less trustworthy systems with biased decisions, resulting in low client cooperation (Kairouz et al., 2021; Bashir et al., 2023; Li et al., 2023; Tariq et al., 2023). Moreover, the absence of mechanisms to comprehend the FL server's decision-making process, hinders the trust between the parties and with the system (Tariq et al., 2023). In a typical FL system, the central server must ensure the client's trust using fair and unbiased contribution evaluation, secure aggregation, and privacy preservation. Given that, cross-silo FL not having a designated server, unlike in cross-device settings, leads to unreliability and vulnerability to anomalies (Li, He and Song, 2021). Further, existing works on trust and explainability largely focused on cross-device FL, leaving a valiant gap to address in cross-silo settings. Considering the differences in architecture and behavior of the clients, makes it challenging to apply previous studies in cross-silo FL, and highlights the importance of the problem (Zhan et al., 2022). Hence, it is critical to investigate this problem to ensure the effectiveness of the cross-silo FL systems and environments.

1.3.1 Problem Statement

Cross-silo FL systems being untrustworthy and uninterpretable with the competing nature of clients affects the long-term participation, eventually impacting the overall effectiveness of the FL environment.

1.4 Research Motivation

The increasing popularity of AI in recent times has raised the number of concerns about AI establishment and data privacy. The European Union's General Data Protection Regulation (GDPR) laws, and European Commission's Ethics Guidelines for Trustworthy AI are few

prominent data protection regulations which have emerged to address these issues (Sánchez et al., 2023). These laws are strictly formed to insist the AI technologies to be more transparent, and interpretable while preserving data privacy and security (Haffar, Sánchez and Domingo-Ferrer, 2023). Violating these laws and regulations can hinder the adoption of AI technologies which creates researchers to find alternatives to tackle these issues.

Even though FL has huge potential to outperform other technologies, its shortcomings in explainability and trustworthiness raises compliance issues with AI regulations and opens doors for novel research opportunities to work on these aspects. Further since FL is still in its early stages, there are massive research areas to be discovered and challenges to overcome. Therefore, these factors served as the main driving forces for the author to learn, conduct and contribute research on this emerging domain, and pursue this current research project.

1.5 Research Gap

The existing literature on trustworthy and explainable FL approaches has highlighted a common limitation to consider, that these studies do not consider the cross-silo approach and have not systematically studied on how an explainable mechanism would execute under this setting. These crucial limitations are summarized as follows.

- **Lack of Trustworthiness in Cross-silo FL architecture**

Trust plays was significant role in fostering client reliability and cooperation within FL systems. However, due to differences in client behavior and architectural distinctions within cross-silo settings (refer **Section 1.2.2**), have made this approach untrustworthy and not applicable to previous works in cross-device settings (Kairouz et al., 2021). Hence, it is critical to implement trustworthy architecture in cross-silo FL scenarios to foster client cooperation between the clients to an effective FL system and to ensure compliance with AI data protection regulations and laws.

- **Lack of Explainability in FL**

Explainability is a key requirement to encourage trust in AI systems, yet current FL research lacks human interpretable explanations and makes it difficult to interpret the server decisions on multi-objectives: contribution evaluation, secure aggregation, and privacy

preservation (Bárcena et al., 2022; Huang, Huang and Liu, 2022; Tariq et al., 2023). Moreover, current approaches mostly focus on global and local model predictions with lack of explanations or debugging options on the FL training process. This makes the system less reliable among the clients and prone to affect their long-term participation.

Since cross-silo FL has started using widely around industries, a comprehensive approach is vitally important to bridge these noteworthy gaps and facilitate the adoption of a trustworthy, explainable FL architecture.

1.6 Contribution to the Body of Knowledge

This research makes significant contributions to both the research and problem domain in the field of cross-silo FL, addressing critical gaps in the existing literature and practice. The author tries to solve the gaps by investigating the current cross-silo FL architectures, systems, and the client behaviors. The past handful of approaches have primarily focused on trustworthy FL architectures in cross-device FL settings, leaving a notable gap in the cross-silo setting. Furthermore, prior work on enhancing explainability in FL systems has been limited to focusing on explaining the global and local model predictions and utilizing explanations in evaluation steps. Hence, the author proposes a **novel trustworthy architecture** designed using Trustworthy AI principles and incorporating a **novel interpretable mechanism** that explains and visualizes the server decisions and processes while ensuring fair and unbiased systems. Additionally, this contribution of a work will act as a unique mediator-like framework for cross-silo FL setting, facilitating effective FL experiences for service administrators and clients ultimately helping industries like healthcare and finance with their future adoption.

Since, FL has been the hotspot of academia and involved in significant development of AI due to the rising privacy concerns. Cross-silo setting in FL has elevated the privacy aspect to a bigger scale by getting adopted in multiple industries. However, research in cross-silo setting is considerably lower than cross-device setting, which creates huge opportunities for the researchers to boost this sub-domain. Hence, the author's contribution will assist the cross-silo FL domain to enhance its explainability and trustworthiness aspects while ensuring an effective architectural flow. Further, this explainable mechanism would eventually lead to the future adoption of new frameworks and libraries that could interpret complex FL systems with less intervention. Finally,

it is safe to say that this trustworthy architectural contribution will serve as the first of its kind in a cross-silo FL setting and create a pathway for future research.

Given the fundamental nature of this research, it is important to note that the contributions to both the problem domain and research domain are closely aligned making it pure research.

1.7 Research Challenges

Cross-silo FL setting is a relatively new area of research with few studies published. To achieve the identified gap, the author confronts some of the expected challenges that could hinder the research project and presented it below.

1.7.1 Handling Data Quality and Structure

In cross-silo Federated Learning, the challenge is twofold: navigating the varied data quality from diverse entities and assessing each client's contribution without direct data access. Addressing data heterogeneity is crucial to maintain an effective and unbiased FL model, while innovatively evaluating client data upholds the integrity of contribution assessments, all within the privacy constraints of FL. This balancing act is key to the system’s success.

1.7.2 Creation of Trustworthy Architecture

Crafting a truly trustworthy architecture in cross-silo Federated Learning requires the careful integration of key principles such as fairness, privacy, explainability, robustness and accountability into the very essence of the system. The primary challenge is to develop approaches that effectively implement these principles, ensuring that the architecture is not only theoretically compliant but also practically reliable. It's crucial to balance these elements without compromising the efficiency of FL workflows. This demands creative strategies to seamlessly blend these important aspects, ensuring a well-balanced and efficient FL system.

1.7.3 Visual representation of explanations

Addressing Federated Learning's 'black-box' nature involves crafting methods for clear, interpretable results, particularly in visualizing server decisions about client contributions, secure aggregation, and privacy preservation. Developing straightforward and insightful visualizations is

crucial to enhance client understanding and trust in the FL system, posing a unique challenge in transforming complex processes into comprehensible information.

1.7.4 Securing privacy and performance in Interpretable FL

Finding the right balance between making the system highly explainable, maintaining its performance, and ensuring privacy protection is a vital challenge. Enhanced explainability can potentially risk exposing client data and model parameters, thus conflicting with FL's core principle of privacy preservation. It is essential to innovate new methods that uphold security and privacy while seamlessly integrating interpretability into the system. This requires a thoughtful approach to ensure that enhancing transparency does not compromise privacy, thus maintaining a careful balance between clarity and data protection in FL systems.

1.8 Research Questions

RQ1: How can consistent client cooperation be fostered in cross-silo FL settings despite having conflicts, competitive nature, and system untrustworthiness?

RQ2: How can a novel architectural design be developed and implemented to enhance trustworthiness in cross-silo FL systems, while ensuring the effectiveness of FL workflows?

RQ3: Which interpretable and explainable techniques are currently employed in FL, and where do they offer scope for enhancement in system trustworthiness?

RQ4: What strategies can be utilized to develop and integrate the proposed explainable mechanism within the trustworthy cross-silo FL architecture?

RQ5: How can the outcomes of the explainable visualizations and results be communicated to FL system clients and sever administrators in a manner that upholds privacy standards?

1.9 Research Aim

The aim of this project is to research, design, develop, and evaluate a solution that employs trustworthiness and interpretability in cross-silo Federated Learning environments by introducing

a novel trustworthy architecture and a trusted mediator to deliver visually interpretable explanations amongst the competing nature of clients.

To make it clear, this research aims to design a new trustworthy Federated Learning architecture in cross-silo FL setting which incorporates an explainable mechanism to foster trust between the clients with competing interests. The proposed system will act as a mediator between the clients and server and contain an explainable model to interpret and explain the FL workflows. The human-interpretable explanations will help both the clients and the server administrators to understand the FL's black box-like workflows and improve the system to be more trustworthy, fair, and unbiased while maintaining privacy. Hence, the successful achievement of the aim will maintain the long-term participation of the clients amidst conflicting and competing interests, ensuring an effective FL environment.

1.10 Research Objectives

The objectives of this research study are presented below and mapped to the apt research questions and learning outcomes.

Table 1:Research Methodology

Research Objectives	Explanation	Learning Outcome	Research Questions
Problem Identification	**RO1:** To conduct a study on the problem domain, Federated Learning. **RO2:** To analyze and identify the research gap within the subdomain of FL.	LO1, LO2	-
Literature Review	**RO3:** To conduct an in-depth literature study on the problem domain FL. **RO4:** To critically evaluate the past works on trustworthy and interpretability-based FL systems.	LO4, LO6	RQ1, RQ2, RQ3, RQ4, RQ5

	RO5: To explore novel ways to address the problem of creating reliable FL architectures and interpretable methods to propose the project implementation. **RO6:** To review, analyze, and discover the existing limitations, best methodologies, frameworks, datasets, and evaluation metrics for the identified FL research gap and the proposed prototype.		
Data gathering and requirements analysis	**RO7**: To identify FL domain and technical experts to discuss, validate and receive feedback on the research gap, research challenges and objectives. **RO8**: To analyze the feedback from the FL experts and identify the requirements for the project solution. **RO9**: To gather requirements on how to design and implement the trustworthy cross-silo FL architecture and explainability mechanism within the FL system. **RO10**: To gather necessary requirements to figure out the user expectations from the proposed FL system. **RO11:** To plan out the functional and nonfunctional requirements of the proposed FL system.	LO3	RQ2, RQ3
Design	**RO12:** To design the novel cross-silo FL architecture to foster trust. **RO13:** To design the explainable visualization mechanism inside the cross-silo FL system. **RO14:** To design methods and algorithms to perform FL workflows and incorporate trustworthy principles.	LO5, LO7	RQ4, RQ5

Implementation/Development	**RO14:** To develop the proposed trustworthy FL architecture. **RO15:** To develop the simulation to demonstrate the explainable mechanism inside the trustworthy cross-silo setting. **RO16:** To implement the core functionalities of the proposed FL system with the proper GUI.	LO5, LO7	RQ1, RQ2, RQ3, RQ4, RQ5
Testing and Evaluation	**RO17:** To design and develop a suitable test plan to test the important features of the prototype. **RO18:** To evaluate the proposed approach against the past works, using the evaluation criteria mentioned on LR and reviews with the domain experts.	LO8, LO9	RQ1, RQ2, RQ3
Project completion and Publishing	**RO19:** To produce the thesis to mark the completion of the research objectives and research aim. **RO20:** To publish the survey review paper using the LR and to publish a research paper on the completion of this research study of NotionFL.	LO8, LO9	

1.11 Chapter Summary

This chapter started off with the background study on the problem domain and proceeds with further discussion on the untrustworthy nature and explainability factors of FL systems. The chapter also outlined the novel research gap, aims, questions, contributions, and presented the significance of the research clearly with justifications. Finally, the chapter concludes by aligning the project's research objectives with the research questions and specified learning outcomes, adhering to the research module guidelines.

CHAPTER 2: LITERATURE REVIEW

2.1 Chapter Overview

This chapter explores the problem area in detail by providing a perceptive examination of many aspects within the discipline. It provides background information for a thorough analysis of previous research, highlighting the gaps, limitations and advancements of current methodologies technologies and studies that motivates future studies to improve the explainability and trustworthiness in cross-silo FL. Furthermore, it also describes the suggested prototype design to overcome current constraints and outlines the evaluation and benchmarking that can be applied.

2.2 Concept Map

The concept map portrays the sections covered in the literature review with the existing solutions, potential technologies, testing, and evaluation metrics for the proposed solution. The concept map is given in **Appendix A** for better visualization.

2.3 Problem Domain

2.3.1 The Rise of Cross-Silo Federated Learning

As previously discussed in **Section 1.2.1**, FL is a continuous learning paradigm used in ML for enabling collaborative training of models by engaging multiple data silos in a more privacy-preserving way (Tan et al., 2022). This has enabled industries around the world to recognize FL as a great solution to their data privacy dilemmas. Typically, a FL process starts off with a server orchestrating the training process by iterating the following steps until the target is achieved or the model convergence has been achieved (Kairouz et al., 2021).

Federated Learning Training Process:

1. **Client selection**: The server or the training initiator will select a set of clients who are eligible for the training.

2. **Broadcast**: The server sends the global model with the weights to the selected clients.

3. **Client computation**: The clients train the downloaded model with their local data.

5. **Aggregation**: The clients send the trained model updates to the server. The client's local data will not be sent to the server; it will stay within the peers.

6. **Model update**: The server updates the global model using the model updates sent by the clients and starts the next round.

Since an introduction to the two main categorizations has been given in **Section 1.2.2**; this section explores the cross-silo setting in-depth with the real-world examples and applications from the existing work highlighting the theoretical and applied breakthroughs.

Cross-silo FL is a unique trend in the FL landscape. It involves fewer but larger clients and is becoming popular in various industries due to its ability to handle complex datasets while focusing on data privacy. Cross-silo FL has proven to have real-world applications, for instance, an insurance service Swiss Re, and a private bank WeBank collaboratively performed a federated analysis of using their private data in the financial risk prediction of reinsurance (FedAI.org, 2020; Kairouz et al., 2021; Huang, Huang and Liu, 2022). Moreover, multiple industries have started to adopt and invest cross-silo FL in real-world scenarios like discovery of pharmaceuticals, electronic health records, smart manufacturing, and more (Kairouz et al., 2021).

2.3.2 Data Distribution in Federated Learning

To fully understand the diverse applications and structures of FL, it is essential to explore its two primary models: Horizontal Federated Learning (HFL) and Vertical Federated Learning (VFL). These models represent distinct ways of handling and distributing data among clients, each tailored to specific types of data sets and scenarios.

I. Horizontal Federated Learning

HFL is designed for domains in which datasets from several clients share a similar feature space, but the sample sizes of the data vary (Mothukuri et al., 2021). Although their example ID spaces differ, clients choose similar characteristics in this case for example Google utilized this setting for their android updates and language models.

In healthcare, HFL facilitates speech disorder diagnosis by merging diverse voice data into a unified speech recognition model (Tariq et al., 2023) and aids in medical imaging to detect cancer cells, ensuring data privacy through secure updates (Mothukuri et al., 2021). From a technical

perspective, HFL architecture often assumes the clients are honest and the server is secure but curious, which means the setting mostly considers client-server centralized architecture where the server manages FL functionalities and discovers the participant data while preserving data privacy (McMahan et al., 2016; Yang et al., 2019; Ratnayake, Chen and Ding, 2023). HFL is widely used in cross-device settings due to utilization in edge devices and IoT devices. Figure 2.1 displays the data distribution of two clients in HFL.

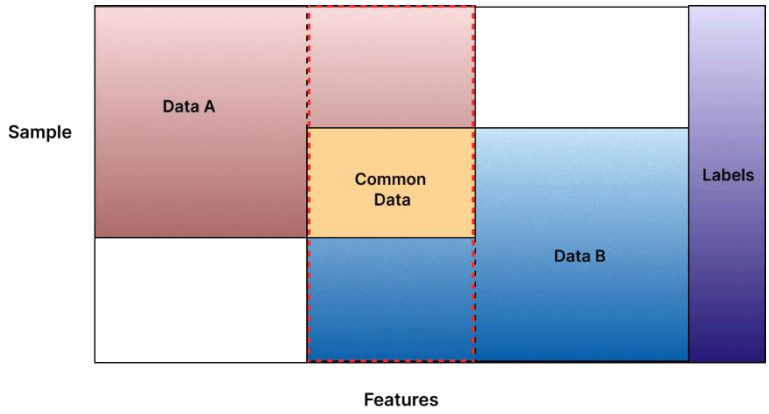


Figure 2: Horizontal FL (Self-Composed)

II. Vertical Federated Learning

VFL finds useful in situations where ML models have different feature spaces across domains but share a common sample ID space. The partnership between Swiss Re and WeBank, covered in **Section 2.3.1**, is a notable example of this use case. In this scenario, the sample ID space is shared by the insurance and finance domains, but their feature spaces are different, where they can operate in the same city but have different data about their customers.

Typically, VFL architecture clients are assessed as honest but curious about each other and they consist of clients with overlapping data samples and a collaborator where the collaborator can be a participating client or a third-party entity (Ratnayake, Chen and Ding, 2023). Techniques like entity alignment and encryption are two main functionalities in VFL which are used to handle data sample overlapping during the local training process (Aledhari et al., 2020; Neto et al., 2023; Tariq et al., 2023). Further, even though the implementation of VFL is more complex than HFL, VFL is more likely to match real-world scenarios and starting to widely adopt in cross-silo FL settings. Figure 2.2 displays the data distribution of the clients in FL.

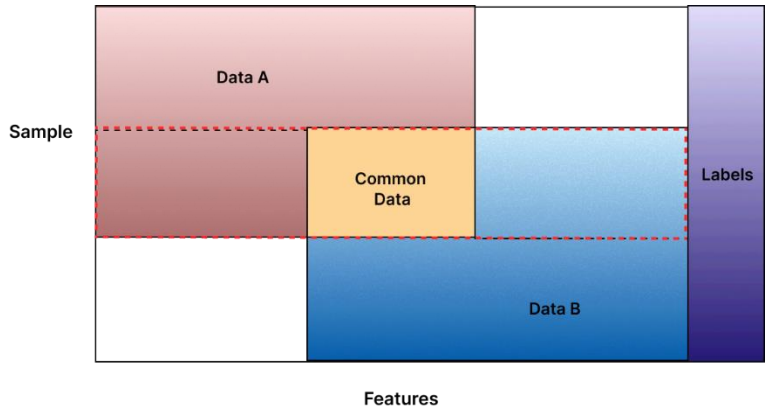


Figure 3: Vertical FL (Self-Composed)

2.3.3 Trustworthiness in Federated Learning

The trustworthiness of a system is always considered on the level of "trust" that makes the system "worthy" of being relied on. On top of data privacy, trustworthiness also becomes a significant aspect influencing AI systems and demanded by the society. In recent years Trustworthy AI has become an emerging concept within the Responsible AI paradigm which was introduced due to the prevailing legal, ethical, social laws and regulations (S'anchez et al., 2023; Tariq et al., 2023; Zhang et al., 2023).

I. The Role of Trust on FL systems

In FL environments, trust can be addressed as a node's confidence in another node's capability of performing tasks and functions as a reliable partner. Trustworthiness, a measurable reflection of this trust, varies based on the entity's actions, particularly in aspects like data security, accuracy, authenticity, and processing capabilities (Tariq et al., 2023).

Client Selection is a crucial stage in FL to influence trustworthiness due to the fact of different selection methods can affect the fairness, potentially leading to a biased global model training (Tariq et al., 2023). Furthermore, the variance in clients' connections or thresholds can cause imbalances in the training process, creating bias and making it a notable problem (Tariq et al., 2023). The contribution evaluation, vital in determining each client's impact on the global model, helps in making informed client selections and fair reward allocations without exposing local data (Tariq et al., 2023). Reputation systems further enhance trust by tracking past client contributions to inform future selections and rewards while fostering trustworthiness in FL systems

(Tariq et al., 2023). S'anchez et al. (2023) also discovered importance of trustworthiness in FL for ensuring privacy constraints, maintaining model integrity, secure aggregation, and fostering client participation, while being accountable. By endorsing these principles FL can influence the potential for collaborative and privacy-preserving approach across numerous domains while sustaining highest level of trust and privacy (S'anchez et al., 2023).

II. Challenges of Trust in Cross-silo FL Setting

Cross-silo FL setting can be differentiated in many ways when compared to cross-device setting which was explained in sections **1.2.2** & **2.3.1**. In FL systems the server or the training initiator is the one who coordinates and manages the FL models, clients, and tasks throughout the training process. But in general cross-silo settings usually won't have a fixed centralized server like cross-device setting because, it mostly utilizes Vertical FL (VFL) data partitioning as explained in **Section 2.3.2**, which assesses the clients as honest participants and appoints the collaborator as one of the clients itself or look for a third-party server (Kairouz et al., 2021). Since there is no reliable server and it is hard to find a compromising third-party server, the cross-silo setting becomes untrustworthy among the clients and within the system and prone to anomalies (Kairouz et al., 2021; Li, He and Song, 2021).

Further, the cross-silo clients are usually business competitors with competing mindsets which makes them hesitant to collaborate and unreliable for long-term participation (Kairouz et al., 2021; Huang, Huang and Liu, 2022). Moreover, this affects the overall effectiveness of the FL system and makes it an untrustworthy environment.

2.3.5 Explainability in Federated Learning

I. The Imperative of Explainability

The advancements in AI, ML, and Deep Learning (DL) have dramatically transformed human lives worldwide, yet most of the technologies used and produced in the AI ecosystems are complex and non-explicit, making them as black-boxes with the mystery of inner workings (Haffar, Sánchez and Domingo-Ferrer, 2023; Kusiak, 2023). This uncertainty creates huge need for clear explanations and understanding of these systems. The facet of explainability always

remains at the forefront of the Trustworthy AI that pairs with Responsible AI which was proposed after the GDPR acts and regulations (Bárcena et al., 2022; S'anchez et al., 2023; Tariq et al., 2023).

Explainability remains as one of the main ethical principles to be followed and respected by the trustworthy AI guidelines since it encourages trust and fairness in the AI systems by producing human interpretable explanations and justifications to the users within the systems (European Commission, 2019). Consequently, academia and industries have started placing their attention on another branch of AI known as **Explainable Artificial Intelligence** (XAI). XAI allows the AI systems to be more transparent while making their decisions and behavior more understandable to the users with practical deployment abilities on a large scale (Corcuera Bárcena et al., 2022; Ungersböck et al., 2023). FL researchers, (Wang, (2019), Bárcena et al. (2022), Chen et al. (2022), Li et al. (2022, 2023), Tan et al. (2022), and many others have also identified the need for Explainability and Interpretability in the FL systems since modern FL systems have adopted to use complex DL models which makes it hard for humans to understand the inner workings of the models and encourages them to leverage XAI.

II. Taxonomy of XAI

XAI aims to improve model comprehension while maintaining a high level of performance and enables trust in choices generated by ML systems (Adadi and Berrada, 2018). According to (Barredo Arrieta et al., 2020), XAI intends to develop a suite of ML models that allow people to understand, interpret, and trust predictions while retaining prediction accuracy. In the context of XAI literature, Explainability and Interpretability are often misused interchangeably, but there are notable differences in these terms. Barredo Arrieta et al. (2020) define interpretability as the "ability to explain or to provide meaning in understandable terms to a human". On the other hand, explainability is referred to as, "an active feature of a model that denotes any action or procedure conducted by a model to clarify its internal functioning". To make systems accessible to people, interpretability alone is insufficient, therefore inclusion of explainability is also vital. However, in this study, the words ***interpretability and explainability shall be used to indicate the same definition***. In General, explainability in AI can be divided into two categories like below.

Post-hoc: these techniques target the models that aren't capable of being interpretable by design by creating a separate model to explain its decisions.).

Ante-hoc/ transparent/ inherent: these techniques already own interpretable capabilities from the start which makes it easy to understand on a modular level.

Moreover, these explainable techniques can be also classified into model agnostic and model specific techniques. Model specific methods are applicable only for a single type or class of algorithms. Model agnostic methods are not tied to a specific type of ML model and can be used in any ML model. In terms of the scope of these techniques providing explanations can be divided into local and global explanations. Global interpretation explains the whole logic of the model and follows with the entire reasonings. While local interpretation explains only a specific decision or a single prediction in the model with the reasoning.

III. Lack of Explainability in FL

FL meets the privacy demands of AI systems, yet its explainability aspects require deeper exploration, essential for aligning with Trustworthy AI principles (Tan et al., 2022). Current FL models often rely on complex, non-linear foundations, posing challenges for stakeholders to comprehend the core functionalities and decision-making processes (Li et al., 2023). Since FL itself a black-box model, the absence of transparency and explainability is troublesome for the collaborators affected by the decisions and for the developers or training initiatives who train the models.

Building on the transparency issue, Wei et al. (2019) noted that FL servers' lack of transparency can confuse users and collaborators, especially those with limited technical knowledge, thereby reducing trust and impeding the technology's broader adoption. Tariq et al. (2023) addressed this by highlighting the absence of mechanisms to ensure fair treatment of clients, a factor that could deter their future involvement. They suggest that enhancing explainability can counteract biases and foster fairness. Similarly, Bashir et al. (2023) pointed out that the 'black box' nature of FL models can obscure understanding of end-users in sensitive sectors like healthcare and finance, where such unclear outputs, predictions or conclusions from the models could lead to critical failures. Hence it is crucial to design explainability mechanisms to offer a complete understanding of the server’s decision-making process while also ensuring that these mechanisms align with FL's primary goal of maintaining privacy preservation.

IV. Federated Explainable AI (FedXAI)

In recent times, the combination of FL and XAI area has been increasing attention in the academia and it has also been recognized as one of the key innovation approaches by the EU Innovation Radar (Corcuera Bárcena et al., 2023). Fed-XAI's goal is to develop methodological and technical solutions that, on the one hand, exploit the FL approach for privacy preservation while jointly training ML/AI models that ensure themselves are adequately explainable (Bárcena et al., 2022). There has been a wide range of adoption in FL-based XAI studies where authors have studied across industries and evolved around the following aspects, Global model prediction explanations, Local client model prediction explanations, Model feature importance explanations, Explainable anomaly detection/intrusion detection, and Visual Analytics (Wang, 2019; Bárcena et al., 2022; Corcuera Bárcena et al., 2023; Guo et al., 2023).

Since FedXAI paradigm is still in its early stages of research, there are still many unexplored aspects left to be unfolded like, FL server explanations on training periods, Debugging on server's functionalities and many more.

2.3.6 Proposed Architecture

The Figure below presents the system overview architecture of the proposed FL system. This architecture includes the FL server, local training, and the explainable mechanism as the core modules and highlights the key workflows.

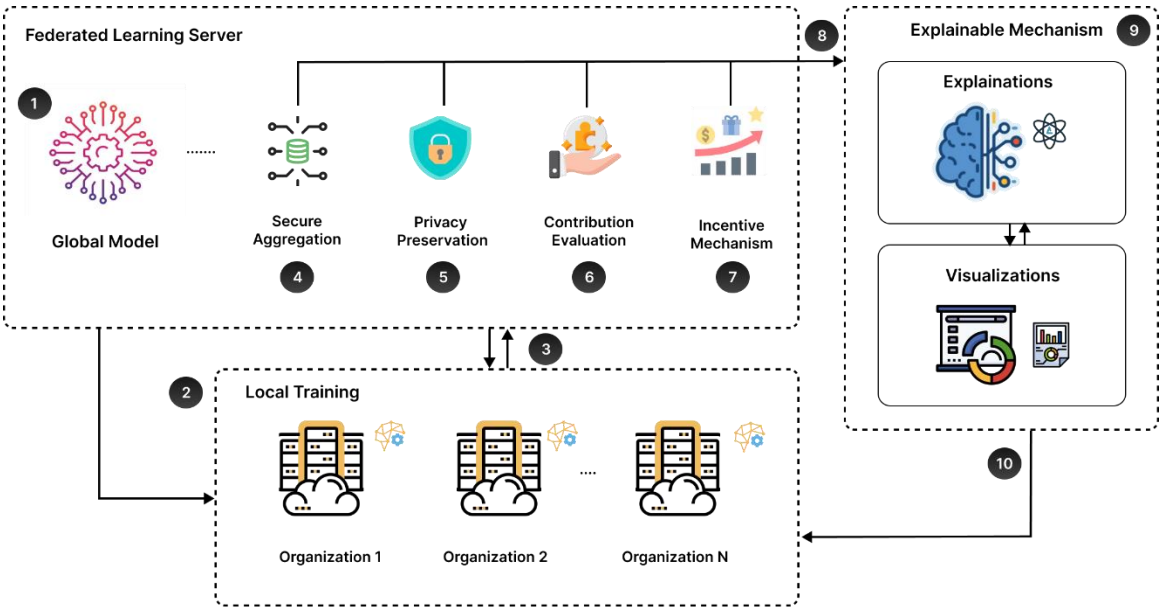


Figure 4: Proposed Architecture (Self Composed)

The above architecture illustrates several key steps that revolve around the flow of the architecture. The author considers a global model ***Mg*** and a fixed set of organizations as the clients ***C = {0, 1, …., N}*** and each client ***Ci*** contains its own local data for training ***Di***. The client contribution of each module is denoted as ***Ei***.

Firstly, the FL server initiates the process by building a Global Model to solve a particular problem.

Step 01: The clients/organizations receive the global model from the server.

Step 02: The clients build their own local model and start training with their local data.

Step 03: Once the training is over (after model convergence), the clients send their local model updates to the server.

Step 04: The client model updates have been sent to the privacy preservation techniques to prevent data leakage and maintain data privacy before aggregating.

Step 05: The client model updates will be received by the aggregator, and it securely aggregates the model parameters into the global model.

Step 06: The encrypted client updates will be sent to the contribution evaluation mechanism to calculate the client contribution.

Step 07: The client contribution evaluation results will be passed onto the incentive mechanism to perform incentive calculations.

Step 08: Collects all the logs, model updates, metrics in each main section in FL server (aggregation, privacy preservation, contribution evaluation, and incentive mechanism) to send it back to the explainable mechanism for thorough evaluation process.

Step 09: The explainable mechanism, monitors, creates and evaluates FL processes and stores the interpretable visualizations and insights.

Step 10: Dashboard visualization helps clients and server initiators to evaluate their work throughout the FL process in each FL process by providing explanations, interpretations and debugging opportunities.

2.4 Existing work

A considerable volume of research is being conducted on trustworthy FL across various settings, recognizing its vital role in AI systems. The study under review has been concisely evaluated and summarized in a table, highlighting its key strengths and weaknesses.

2.4.1 Trustworthy Federated Learning Approaches

Majority of the previous work chooses a certain set of criteria or aspects related to trustworthy AI to evaluate the trust assessments in their proposed work. Tariq et al. (2023) introduces a novel concept of trustworthy FL architecture that encompasses all the characteristics of trustworthiness in the FL process by utilizing three main phases, interpretability, fairness and security and privacy. Each phase covers different aspects related to the FL process while collectively ensuring trustworthiness, data integrity, fairness, security, and privacy throughout the FL system. This study stands as the first comprehensive survey on trustworthy FL, delving into key FL components and identifying limitations in current systems.

A similar study from S'anchez et al. (2023) also examines the existing FL works to evaluate the importance of trustworthiness in FL systems and introduces a novel taxonomy for computing trustworthiness of FL models. Their work addresses the scarcity of comprehensive trustworthy FL research, particularly in cross-silo contexts. the author also tested their FederatedTrust algorithm using the FEMNIST dataset, evaluating trust across multiple pillars. Despite its innovative approach, it encounters issues like data leakage, resource consumption, and scalability, along with device limitations and the need for regulatory compliance. Another fascinating study from Zhang et al. (2023) explores trustworthy FL in different development stages using the core aspects: privacy, robustness and security and presents a comprehensive survey highlighting the general picture, open problems, existing gaps, and a roadmap to develop FL in a trustworthy manner.

The Blockchain technology is another common approach that was used for FL architectures based on trustworthy FL systems. Even though blockchain technology creates additional complexity in FL systems it also enables fair, trustworthy, and accountable architecture with no single point of failure since there's no central server in a decentralized setting. Yang et al. (2022) introduces created a decentralized blockchain-based FL system to enhance trustworthy AI training

and reduce latency, using the PBFT consensus protocol for improved wireless blockchain FL communication and a TD3-based algorithm for efficient resource management. However, it faces challenges like complexity, data leakage risks, scalability issues, and a specific focus on cross-device settings, posing hurdles for practical deployment.

A similar work named TrustFed by Rehman et al. (2021) also adopts Blockchain technologies like ethereum and smart contracts to propose a novel protocol to detect outliers in the FL training and removes it before the aggregation process. The framework is more focused on cross-device edge devices setup, and it uses ethereum smart contracts to incentivize the clients in the system to maintain reputation. Fairness was considered as the main aspect in this study to make the FL system trustworthy among participants and the performance comparison suggests that TrustFed achieves better results in identifying and removing outliers if most of the workers/clients within the training are honest. Moreover, these studies are more focused on individual aspects of achieving trust within the FL system rather than attaining the overall trustworthiness in the architecture.

Lo et al. (2023) explored a blockchain-based trustworthy FL architecture for COVID-19 x-ray detection, integrating smart contracts to address accountability and fairness. They used a smart contract-based registry for data model provenance to ensure accountability, and a weighted fair data sampler algorithm to improve fairness. However, this approach introduces complexity to the architecture and poses challenges like connectivity and latency issues. Hsu et al. (2022) contributed a unique study enhancing the robustness and trustworthiness of FL system moderators. Focusing on securing client trust, they utilized lightweight cryptographic tools for secure, verifiable computations.

I. Taxonomy of Existing Trustworthy FL Architectures

Table 2: Taxonomy of Existing Trustworthy FL Architecture

Existing Work	FL Architecture	Trustworthy AI Principles	FL Processes

	Cross-device setting	Cross-silo setting	Vertical FL	Horizontal FL	Explainability	Fairness	Privacy	Robustness	Accountability	Contribution evaluation	Secure Aggregation	Privacy and security	Incentive mechanism
(Tariq et al., 2023)	X			X	X	X	X			X	X	X	X
(S'anchez et al., 2023)	X				X	X	X	X	X				
(Yang et al., 2022)	X										X	X	
(Zhang et al., 2023)	X						X	X			X	X	
(Lo et al., 2021)	X			X		X			X		X		
(Rehman et al., 2021)	X					X	X			X	X	X	X
(Hsu et al., 2022)							X	X			X	X	
(Bao et al., 2019)						X				X	X	X	X
NotionFL		**X**	**X**		**X**	**X**	**X**	**X**	**X**	**X**	**X**	**X**	

II. Summary of Existing Trustworthy FL Architectures

The summary of the previous approaches in trustworthy FL systems and architecture has been outlined in a tabular form for better readability and can be viewed in **Appendix B**.

2.4.2 Explainable Federated Learning Approaches

Explainability and interpretability are central to trustworthy AI, yet Federated Learning (FL) systems, often seen as black boxes, struggle with these aspects. This lack of interpretability

hinders stakeholder understanding and trust in FL technology. Recent research efforts are focusing on enhancing FL systems' explainability and architectures.

i. Explainable FL Taxonomies

Li et al. (2023) introduced a comprehensive taxonomy for interpretable FL, cataloging significant past works that enable FL models to clarify predictions, aid in debugging, and explain client contributions. The taxonomy encompasses phases like client, sample, and feature selection, alongside model optimization, and contribution assessment. However, a notable limitation is its focus on interpreting individual FL training stages, which doesn't fully address the overall FL architecture's reliability.

Similarly, Bárcena et al. (2022) explored federated explainable AI models (FED-XAI), utilizing FL to balance privacy in ML/AI models with necessary explainability. The study highlights major challenges, including data leakage, performance degradation, and privacy management with large-scale data. It notes the often-competing nature of model performance and explainability (see, Figure 2.2) Complex, high-performance solutions tend to lack interpretability, while simpler, interpretable models like decision trees may not perform as robustly. This has made researchers find new solutions and ML models that can perform well under explainability constraints.

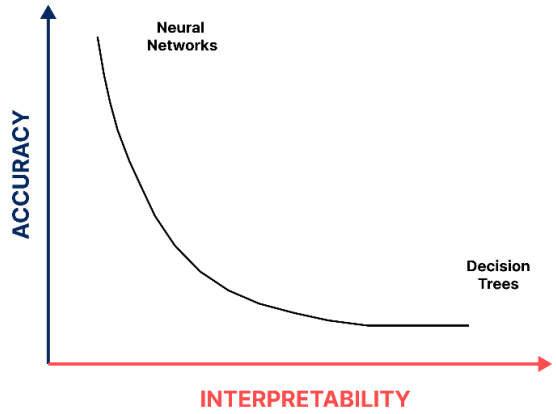


Figure 5: Trade-off between model interpretability and performance (Self-compassed)

ii. Global and Local Model Explanation in FL

Ben Saad, Brik and Ksentini. (2022) introduced a novel XAI-powered framework, focusing on the prediction of network slice performance indicators. The authors built a deep learning model in Federated way to predict key performance indicators of network slices with the use of linear and non-linear XAI models. These developed on top of the FL models to enhance the trust and credibility, transparency and explanations of the FL-based decisions while managing privacy. The system utilized these XAI models to generate local, global, and feature important based explanations related to the predictions and visualizes them on a graphical user interface.

Corcuera Bárcena et al. (2022) introduced an interpretable FL model using Takagi-Sugeno-Kang Fuzzy rule Based Systems (TSK-FRBS), with a strategy for learning global models while preserving data privacy. Despite achieving accuracy, its performance was less efficient compared to centralized models. Expanding on this, their Corcuera Bárcena et al. (2023) study applied similar XAI-FL integration in B5G/6G vehicular networks, focusing on video streaming quality. However, it faced performance challenges against centralized architectures.

Renda et al. (2022) explored a FL-based XAI system in 6G networks and automated vehicle networking, predicting, and explaining Quality of Experience (QoE) for car manufacturers and network operators. This study, focused on edge devices and cross-device FL settings, was limited to global model QoE predictions. Similarly, Huong et al. (2022) developed 'FedeX', a lightweight, FL-based system for anomaly detection in industrial controls, which excelled in learning speed and analysis but encountered privacy and security challenges, especially in edge cloud data transmission.

Haffar, Sánchez and Domingo-Ferrer (2023) utilized random forest decision trees in their FL model to identify key features in erroneous predictions, enhancing model explainability in anomaly detection. Chen et al. (2022) introduced an explainable vertical FL framework, incorporating credibility assessments and counterfactual explanations, focusing on feature importance and providing credible insights into the selection process. Finally, Wang (2019) in a vertical FL study, balanced interpretability and privacy using Shapley values, offering detailed feature importance explanations to clients and moderators, showing promise for real-world application while preserving performance and privacy.

iii. Explainable FL Visualizations

Ungersböck et al. (2023) presented a pioneering approach to elucidate the FL workflows through a life cycle dashboard visualizing the server information from the FL systems (Figure 2. 4). The authors aim to improve the transparency and explainability of the system for all types of users and industries. Despite its generality and applicability, the study was predominantly focused on edge devices (cross-device) in industrial settings and exhibited limitations in user interactions and debugging capabilities. Further another notable gap was the absence of explanations on underperforming clients within the system.

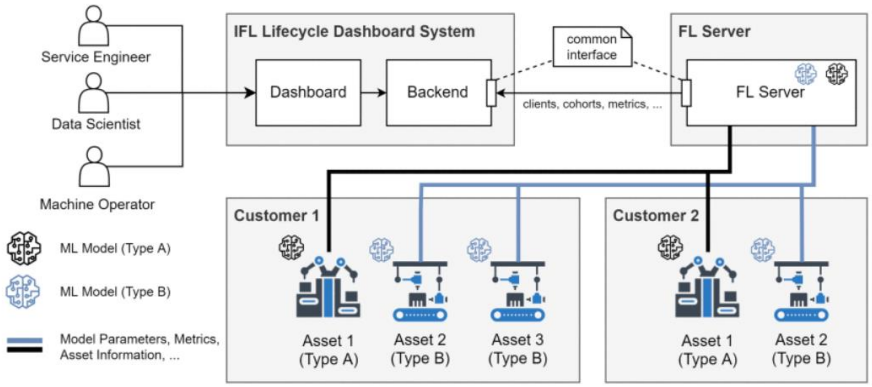


Figure 6: Life cycle Dashboard Architecture Ungersböck et al. (2023)

Wei et al. (2019) developed a multi-agent visualizations system, utilizing a racing game scenario for educational purposes to interpret the FL's inner workings. This novel approach served to make the complex processes of FL more accessible and transparent to users, particularly in an educational context. FATEboard is an interactive dashboard-like visualization component created by the popular FL library FATE which was dedicated for explaining Vertical FL processes. While it efficiently records and summarizes the FL process logs, it falls short to provide detailed analysis on anomaly detection.

Guo et al. (2023) developed an interactive visualization tool to help data owners comprehend privacy aspects in FL systems, focusing mainly on horizontal FL scenarios, which limits its application range. Similarly, Li et al. (2022) created HFLens, a visual analytics tool tailored for horizontal FL, offering comparative insights from an architectural overview to individual client analysis. While HFLens provides detailed HFL process analysis, it lacks in areas

like hardware utilization, client configuration, and advanced visualizations, and its focus is predominantly on horizontal FL, indicating a research inclination towards this specific setting.

iv. Summary of Existing Explainable FL Works

The existing approaches in explainable FL which were analyzed deeply in above sections are outlined in tabular form for better readability and can be viewed in **Appendix C**.

2.5 Technological Review

This subsection examines techniques applicable for designing and implementing explainable mechanisms in Trustworthy FL architecture, covering both widely used methods in past research and those selected for this project.

2.5.1 Trustworthy Federated Learning Framework

i. Federated Learning Frameworks

TensorFlow Federated (TFF) (Bonawitz et al., 2019) is an open-source framework with large-scale simulations for FL environments. This framework enables users to experiment with existing FL algorithms along with their data and models using the advanced API services (FL API & Core API) provided by them. Although it allows deep learning models on decentralized data, it also falls short to provide aggregation strategies, vertical and hybrid data partitioning, privacy mechanisms except differential privacy and multi node deployment options.

Recent popular open-source framework, **PySyft** (Ryffel et al., 2018) is also an open-source Python framework built using PyTorch for privacy preserving deep learning-based FL. It recommends users to simulate FL systems inside virtual environments using package managers and facilitates implementing complex private and secure FL settings.

Federated AI Technology Enabler (FATE) (Liu et al., 2021), is one of the world's first open-source FL platform project done by the Webank and the Linux foundation. This framework promotes advanced security and privacy in implementing FL architectures while enabling data partition techniques, aggregation techniques, industry-graded secure protocols, and algorithms. **Flower** (Beutel et al., 2020) is a new comprehensive open-source FL framework facilitating

scalable FL experiments with custom secure aggregation strategies. **IBM Federated Learning** (Ludwig et al., 2020) is an IBM licensed FL framework written in Python.

ii. Trustworthy FL Framework

Even though all the above frameworks resemble general architectures in FL, it does not consider trustworthiness and the significance of trustworthy AI regulations. A detailed discussion of this proposed architecture will be presented in **Section 6.5.3**.

2.5.2 Explainability Techniques for Federated Learning

i. Interpretability Methods

LIME (Ribeiro, Singh, and Guestrin, 2016) is an explanation technique that interprets machine learning model predictions using a simpler, understandable model. It explains how changes in features influence predictions. Haffar, Sánchez, and Domingo-Ferrer (2023) utilized LIME for explaining attack predictions in FL models via Random Forest, while Ben Saad, Brik, and Ksentini (2022) employed it for local interpretation of KPI latency predictions.

Shapley Additive Explanations (SHAP) (Lundberg and Lee, 2017), derived from game theory, assesses each feature's marginal contribution by considering all possible combinations in a model. SHAP equitably allocates predictions across features. However, its application in deep learning models is ***time-intensive*** and dependent on specific model structures. Notably, Ben Saad, Brik, and Ksentini (2022) implemented SHAP for local feature importance in an XAI model. Similarly, Huong et al. (2022) employed it for explainable anomaly detection in FL, and Wang (2019) used it to highlight feature significance, demonstrating SHAP's wide application in various FL studies.

The RuleFit algorithm, an interpretable tool, elucidates features using decision rules and combines them with dataset features to create a linear model. In FL research, Ben Saad, Brik, and Ksentini (2022) applied RuleFit to a KPI latency prediction model, generating new rules for feature importance. Additionally, the Partial Dependence Plot (PDP) method visualizes the relationship between features and predicted outcomes in learning models. Ben Saad, Brik, and Ksentini (2022)

used PDP in FL to explore the connection between features and target labels, aiding in visual interpretation of predictions.

ii. Visualization Techniques

This section reviews two of the most significant visualization tools in FL, as identified in the author's research. **FATEBoard**, a visualization tool from the FATE Federated Learning framework (Liu et al., 2021), offers comprehensive visuals on FL training jobs, displaying task status, model outputs, log metrics, and dataset details. However, it lacks in-depth analysis of anomalies. Wei et al. (2019) used FATEBoard in a multi-agent system for horizontal FL visualization, while Li et al. (2022) introduced HFLens, a tool for horizontal FL that provides detailed views of communication rounds and client interactions. **NVIDIA FLARE** (Roth et al., 2022) is another FL visualization tool, focusing on model training and validation through TensorBoard.

2.5.3 Techniques for Contribution Evaluation.

i. Self-reported/Text-Based

Self-report evaluation in FL involves clients reporting their contributions, such as computational resources and dataset sizes, to the model owner. However, the challenge lies in verifying these reports, as dataset size doesn't directly correlate with model accuracy, making this approach less effective for quality-based reward systems (Zeng et al., 2021). Additionally, this method can be applied in test-based evaluations, where the parameter server generates a test dataset, and client contributions are assessed based on their performance on this dataset to fairly distribute rewards (Huang et al., 2020).

ii. Utility Game Based

The Shapley Value (SV) technique in FL assigns a unique evaluation profile to each client, based on their mean marginal contribution to model training. It compares client data combinations to assess the uniqueness of each participant's contribution fairly. Kairouz et al. (2021) note that computing SV is resource-intensive, posing challenges for regular FL systems. To address this, alternatives like MonteCarlo Shapley value and Gradient-based Shapley value have been

proposed. SV has gained popularity in FL research for contribution evaluation and interpretability, with various SV-based models being adapted for different use cases.

iii. Influence and Reputation Based

In FL, a client's reputation is calculated based on their task-specific performance and past contributions (Huang et al., 2020). Liu et al. (2022) transformed historical contributions into reputation scores for evaluating client input and reward distribution. The TrustFed framework by Rehman et al. (2021) tracks client activities via blockchain for credible contribution assessments. Xu and Lyu (2020) designed a framework assessing contributions through a reputation mechanism, comparing uploaded and aggregated gradients for similarity.

2.5.4 Techniques for Secure Aggregation

Secure aggregation is critical for preserving privacy and maintain data integrity in FL systems especially the choice of aggregation, impacts the communication efficiency, privacy, and robustness of the FL systems (Pillutla, Kakade and Harchaoui, 2022). Author has examined the major techniques used on previous literature and highlighted the selected technique for this project.

i. Consensus based Secure Aggregation

The goal of this technique is to achieve agreement among multiple FL participants using consensus algorithms. While it improves data integrity and system stability, computational overhead and scalability issues are possible drawbacks (Li et al., 2021). It is critical for FL applications that require high data correctness and integrity, but it may be inefficient in large-scale, dynamic FL situations and vulnerable to attacks like, central server attacks, Byzantine attacks, and backdoor attacks.

ii. Blockchain and Smart Contract Based

Integrating blockchain and smart contracts into FL improves security and transparency by assuring data integrity and participant responsibility. However, this approach can introduce new complexities and result in higher computational and operational costs (Rahman et al., 2020). This technique will be hard to utilize in cross-device setting and will be suitable for decentralized cross-silo setting since the computational resources could handle the additional complexity.

iii. Gradient based Secure Aggregation

This solution improves FL's privacy and security by focusing on safe model update aggregation (gradients). It successfully eliminates data leaking during model training, but it may result in higher communication overhead and delayed convergence. This technique is usually ideal for processing sensitive data but may require tuning to improve efficiency and decrease resource utilization.

2.5.5 Techniques for Privacy and Security

i. Secure Multi-Party Computation (SMPC)

Secure multi-party computation (MPC) is a cryptographic approach used in FL to encrypt data, ensuring privacy by only sharing necessary information with clients. Kairouz et al. (2021) describe MPC more as a field comprising various secure computation technologies with different encryption schemes, often implemented on finite fields due to challenges in representing real numbers. This method is also adapted for secure aggregation.

ii. Homomorphic Encryption (HE)

Homomorphic encryption (HE) is a cryptographic method that allows computation on encrypted data without needing to decrypt it, maintaining data privacy (Kairouz et al., 2021). While it enables complex computations securely, it incurs significant computational costs. In FL, clients send encrypted data to the server without risking decryption, necessitating frequent renewal of secret keys held by a trusted external party.

iii. Differential Privacy (DP)

Perturbation techniques in ML training involve adding noise to data or mapping instances to compute statistical differences from the original. Differential Privacy (DP), a specific perturbation method, masks user contributions by adding noise to model parameters before aggregation, often at the cost of accuracy (Geyer, Klein, and Nabi, 2017). DP is categorized into local and global, with global DP less revealing of dataset information leading to its broader adoption in FL. According to Dwork, (2006) the definition of differential privacy is as follows; A randomized function K provides ε-differential privacy, which ensures the chance of identifying

whether any individual's data was utilized in a dataset is limited, by a factor mathematically bound by ε (epsilon), a small constant. This is achieved by adding controlled noise to the output, which is proportionate to the data's sensitivity (the maximum change to the output that could occur by altering any single individual's data).

$$\mathbf{Pr\,[K(D1) \in S] \leq e^{\epsilon} \times Pr\,[K(D2) \in S]}$$

In essence, ε-differential privacy ensures that the inclusion or exclusion of any individual data point in a dataset has a negligible impact on the output of a specific function, K. This minimal effect is crucial for protecting the privacy of individuals in the dataset, defining a core requirement for a system to be regarded as offering ε-differential privacy.

2.5.6 Evaluation and Benchmarking

A multi-faceted evaluation strategy is to assess the effectiveness of the trustworthy AI principles implemented in the system along with the explainable mechanism. Therefore, the process can be categorized into qualitative and quantitative measures.

i. Quantitative Evaluation

Privacy-Preserving and fairness Metrics: Using privacy metrics such as differential privacy helps to quantify the level of privacy maintained while explaining the workflows. Implementing statistical tests to measure bias or fairness across different demographics or data sets. Further, to assess the FL training effectiveness the model evaluation metrics such as accuracy, precision, F1-score, and recall were utilized.

ii. Qualitative Evaluation

- **Expert Interviews**: Conduct structured interviews with domain experts to assess the perceived trustworthiness of the system and provided explainability within the system. Gather insights on the architecture's explainability, fairness, and accountability.
- **User Studies**: Engage with end-users through interviews or surveys to evaluate the practicality and understandability of the system's decisions.
- **Feedback Loop**: Establishing a feedback loop mechanism to access initial users who can provide direct feedback on the system, will be used for iterative improvements.

iii. Benchmarking

- **Comparison with Baselines**: Comparing this project with different baseline models in similar domains to highlight the advancements in trustworthiness and explainability.
- **Adherence to Standards**: Evaluate how well the architecture aligns with industry standards for Trustworthy AI, such as the guidelines set by the European Union.

2.6 Chapter Summary

This chapter offers a concise overview of the latest developments in TFL architectures, with a keen focus on identifying existing research gaps. It delves into various aspects of this complex domain, presenting a comprehensive comparison of past approaches to pinpoint the most effective strategies for this project. The chapter concludes by exploring diverse methodologies that will be utilized for evaluating the outcomes and effectiveness of the proposed FL system, ensuring a thorough assessment of the project's advancements and contributions to the field.

CHAPTER 3: METHODOLOGY

3.1 Chapter Overview

This chapter presents the methodologies selected for research, development, and project management in this work. Each methodology is discussed in-depth with justifications behind to support why each component is selected over the other counterparts for this study.

3.2 Research Methodology

The selection of the scientific research methodologies were selected from the Saunders Research Onion Model (Saunders, Lewis and Thornhill, 2006) and presented below with reasonings.

Table 3: Research Methodology

Research Philosophy	The **Pragmatism** approach was chosen as the research philosophy among positivism, realism and interpretivism. Pragmatism was chosen because this research project requires various kinds of both qualitative and quantitative methodologies for the evaluation and to achieve the research goal comprehensively.
Research Approach	Among deductive and inductive, the **Deductive** approach was chosen as the research approach because this research project aims to apply existing theories and the results will be examined by qualitative and quantitative methods. Inductive approach was not chosen because the author will not be creating new theories or generalizations using specific data.
Research Strategy	The **experimental** based research strategy was chosen as the research strategy of this project where it will help the author to carry out the research and answer the research questions while the project is involved in a controlled environment to exhibit the prototype.

Research Choice	This research will incorporate the **mixed-method** over the mono method and multi-method as the research choice due to the usage of qualitative and quantitative approaches. Mono method only considers qualitative or quantitative approaches and multi-methods are usually considered when two or more projects were conducted at the same time.
Time Horizon	The **longitudinal method** was chosen as the time horizon over the cross-sectional method because the data collection for the whole project would be done in different intervals.
Techniques and procedures	The author will be using the interviews, questionnaires, documentations, research papers, articles, books, etc as the techniques and procedures of this project.

3.3 Development Methodology

The **Agile** Software Development life cycle was chosen over the waterfall, spiral, iterative and prototype models since this research project needs an iterative trial and error approach to reach the final solution. The ability to handle new changes, and efficient resource utilization makes it more suitable to the project. Also, it helps to review the solution in each iteration to mitigate the errors in advance, but the continuity of new changes might make the project estimation overdue, still, it is not significant to this project.

3.4 Project Management Methodology

The **Prince2** approach was chosen by the author as the project management methodology over other methodologies like Scrum, Kanban, and Waterfall since it helps the author to design and develop the prototype in a controlled environment. Flexibility on changing requirements, effective risk management, focus on quality (deliverables), and integration with the agile life cycle model are some of the key advantages of Prince2 over other methodologies.

3.4.1 Project Plan

The project schedule was managed and visualized using GANTT chart, please refer to the **Appendix D** for the detailed visualization.

3.4.2 Deliverables and Milestones

Potential deliverables and deadlines for the project have been given in Table.

Table 4:Project Deliverables and dates

Deliverables	Date
Project Proposal Document The initial project proposal of the research thesis.	19/10/2023
Literature Review Document An in-depth review of existing works and solutions on the domain.	13/11/2023
Software Requirement Specification The requirements to be satisfied while developing the research prototype.	27/11/2023
Proof of Concept - Prototype Core part of the research project	15/02/2024
Project Specification Design and Prototype (PSDP) The document specifying project specifications, design, and the prototype.	15/02/2024
Final Thesis Final thesis submission with all the project documentations combined.	04/04/2024

3.5 Resource Requirements

Based on the project objectives and functionalities the resources required to complete the project have been listed below with the categorizations.

Table 5: Resource Requirements

Requirement	Tool	Justification
Software	Operating system (Windows/ macOS/ Linux)	To perform the development of the project and for research documentation and study purposes. First preference will be windows, since it supports FL related technologies far better than MacOS.
	Python	The Python programming language was chosen over other languages because it is an all-purpose language and especially goes well with data science projects. It is also a great addition to support the development of Federated Learning (FL) systems and libraries.
	PyTorch	Among the popular libraries for FL PyTorch was chosen over the TensorFlow Federated due to its popularity, good community, and great support to the other tools.
	React JS	The primary programming language chosen for the frontend development of the proposed system.
	Visual Studio Code	IDE selection due to being a versatile code editor for developing applications on both front and backend.
	GitHub	Utilized for the version controlling and backing up the project code base in an online repository.
	Zotero	For the author's convenience this application was chosen to manage the research papers and use as reference management software.
	Google Drive	Online repository to store the project files on the cloud as backup storage.

	MSOffice Package	Thesis documentation making, spreadsheets for planning and presentation making.
Hardware	Core i5/i7 Gen 8+ or Ryzen 5/7 new Gen processors	Minimum requirements to smoothly run the FL project environment with multiple clients and training rounds. To manage the bigger computational power needs for effective collaborative model training.
	16 GB RAM	
	12 GB NVIDIA Tesla K80 GPU	To execute machine learning tasks with fewer hurdles since GPUs are more capable of running the ML than CPU.
	64GB Disk Space	Author's choice of minimum requirement to manage and store project data, resources, models, and results.
Data	MNIST Dataset CIFAR 10 Dataset	Popular open-source FL datasets that will be used to simulate the model training in the proposed system as the organization's local datasets.
Skills	ML libraries and model creation	Mandatory data science knowledge and skills needed for the NotionFL project implementation using FL and XAI.
	FL training and models	
	Explainable AI models	
	UI/UX skills	To effectively design and illustrate diagrams and prototype wireframes.

3.6 Risk Management

The risk management planning for the research has been given in Table below.

Table 6: Risk Management

Risk Item	Severity	Frequency	Mitigation Plan

Lack of knowledge in the problem domain	High	High	Taking the online resources to study. Talking to domain experts.
Learning curve of the technologies, frameworks and theories of the problem domain which gets some time to adapt	High	Medium	Constantly taking good resources to review and reaching out to other developers in the domain for help and feedback
The use of new libraries with new domains which requires adaptations and support (XAI with FL)	High	Medium	Staying touch with the libraries and with the community to be alerted with new updates
Lack of resources on the existing works in the sub-domain	Medium	Low	Adapting to related domain works and trying out the same in the problem domain
Data loss and software failures	Medium	Low	Using online repositories and taking regular backups
Hardware failures	Medium	Low	Physical storage backup devices

3.7 Chapter Summary

This chapter explored the research methodology, development methodology and other methodologies that were chosen for this project's implementation by the author and examined the project management strategies along with the project timeline and crucial requirements. Finally, the chapter concluded with the potential risks that could hinder the project accompanied by the possible management approaches.

CHAPTER 4: SOFTWARE REQUIREMENTS SPECIFICATION

4.1 Chapter Overview

The purpose of this chapter is to investigate the requirements elicitation models in order to successfully maximize the data collection for the project's successful development. The chapter first starts off with identifying the stakeholders, and then the selected elicitation methods are used on important stakeholders followed by the data analysis. Further the gathered data will be utilized to construct use case diagrams and with the use case descriptions. Finally, the chapter concludes with the identification of functional and non-functional requirements of the system.

4.2 Rich Picture Diagram

The rich picture diagram provides a detailed overview of the system's structure, its process and interactions with other systems that surround it. The figure below portrays the rich picture diagram of the proposed NotionFL system which depicts the defined structure, process, and issues.

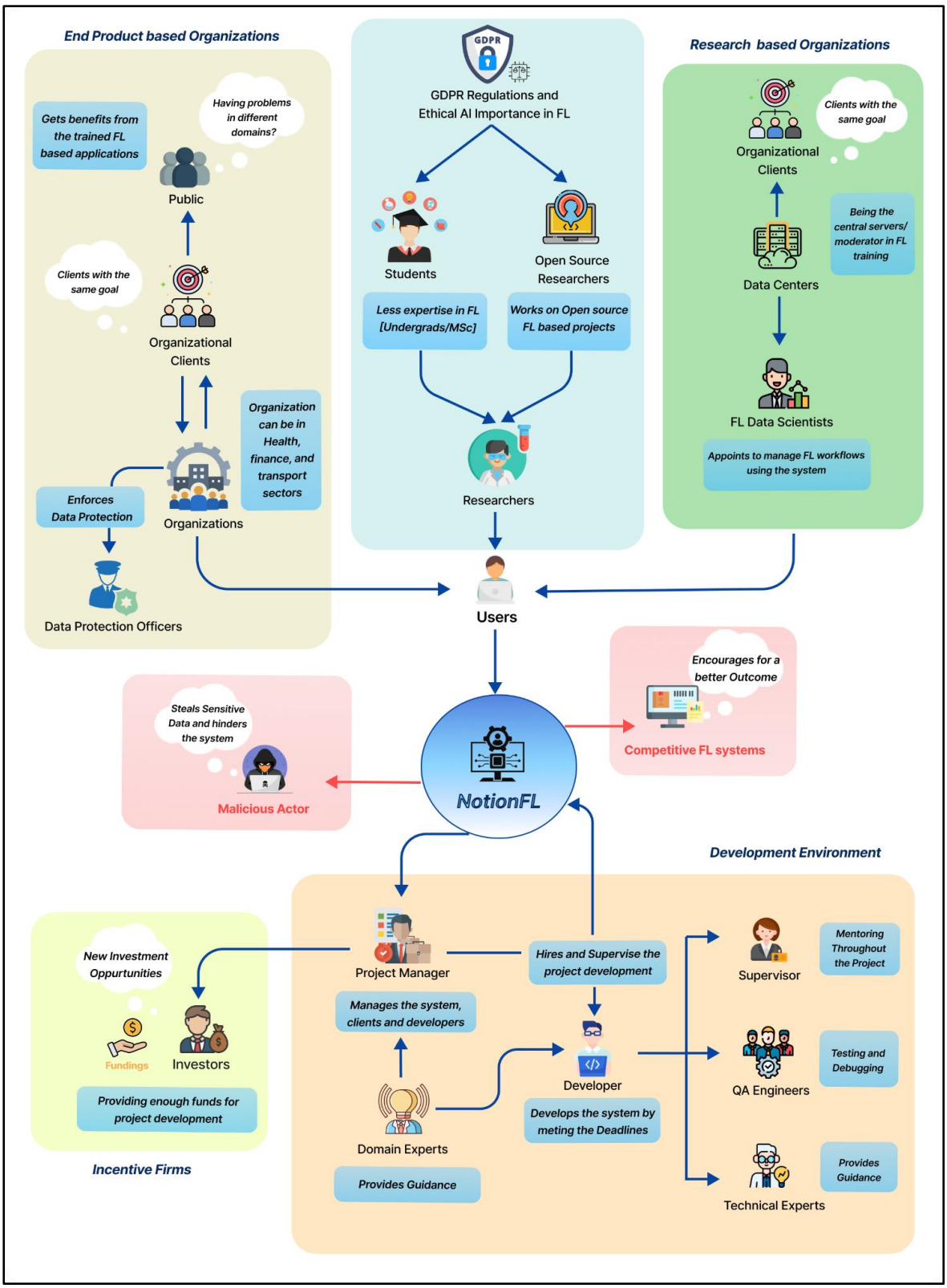


Figure 7: Rich Picture Diagram (Self Composed)

4.3 Stakeholder Analysis

The proposed NotionFL system's stakeholder onion model is presented, detailing the key stakeholders and their detailed discussions.

4.3.1 Stakeholder Onion Model

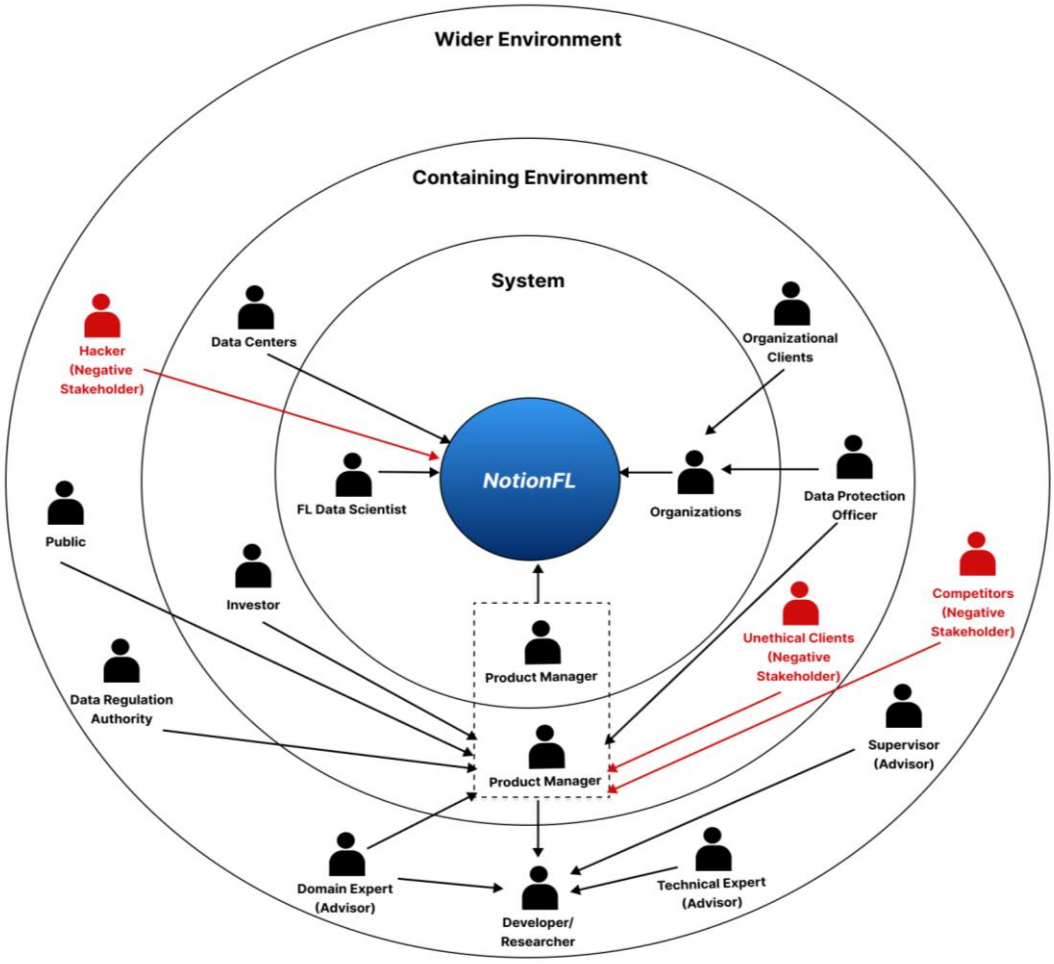


Figure 8: Stakeholder Onion Model (Self Composed)

4.3.2 Stakeholder Analysis and Description

Table 7:Stakeholder Viewpoint Descriptions

Stakeholder	Role	Benefits
The Project Stakeholders		

FL Data Scientist	Normal Operator	Interacts with the system to create effective FL training Environments.
Organization	Normal Operator / Functional Beneficiary	Use the system to collaboratively train a model to provide enhanced services for their clients.
Product Manager	Maintenance Operator / Support Operator	System manager will work as a system administrator to maintain the system, oversee all requirements & prioritizes them, and also hires developers.
Containing Environment Stakeholders		
Organizational Clients	Functional Beneficiary	Benefits from the services provided by the organizations using the system.
Unethical Clients	Negative Stakeholder	Refuse or remove from training the model using their local dataset but try to benefit from the training results of other organizations.
Investors	Financial Beneficiary	Provides funding for the development of the system and benefits from the profit generated through the system.
Data Protection Officers	Functional Beneficiary /Regulator	Hired by the Organizations or the Moderators to oversee the data privacy and security of the system.
Wider Environment Stakeholders		
Technical Expert	Regulator/ Advisor	Gives proper guidance to the developer or the researcher to achieve their goal.
Domain Expert	Regulator/ Advisor	Gives domain knowledge and guidance on the Federated Learning and XAI platforms.
Supervisor	Regulator/ Advisor	Gives guidance throughout the entire project and wants the researcher to

		complete the project successfully.
Developer/Researcher	Financial Beneficiary	Develops and maintains the system for the system manager and clients.
Data Protection Authority	Regulator	Government and Private data protection authorities that oversee the application to ensure the data protection laws are obeyed with investigation and corrective capabilities.
Competitors (similar systems)	Negative Stakeholder	Competes with the system by building or duplicating the system to cater other organizations.
Hackers	Negative Stakeholder	Malicious actors who attack the system for data exploitation and break the system.
Public	Social Beneficiary	Gets benefited from better quality of life services resulting from the organizations after the use of the system.

4.4 Requirement Elicitation Methodologies

In order to gather software requirements for this project, the author went through a set of available elicitation methodologies and decided on techniques such as LR, interview, prototyping and brainstorming. The justifications for the selected elicitation methods are summarized in a tabular version below along with its **strengths** and **weaknesses** described.

Table 8:Requirement Elicitation Methodologies

Method 01: Literature Review

LR is used to aid authors to establish knowledge and discover recent advancements in the field of FL. LR on previous work helps to figure out and validate research gaps established. The field of Federated Learning (FL) is constantly explored and evolved by the researchers to bridge the gaps to use this intriguing field in real world applications. Hence, the research papers serve as a good resource to identify the strengths and weaknesses in existing approaches and help to review the limitations to benefit the research. However, LR on any subject requires a significant amount of time and effort to go through all the credible published work in the field. Furthermore, there is only a slight or no stakeholder involvement to obtain precise requirements.
Method 02: One-on-one Interviews
The project involves the cross-silo Federated Learning subdomain to bridge the gap to the existing problems. One-on-one interviews were chosen as the main elicitation method since it's a rich tool that helps author to analyze the feasibility and validity of the research from different expert's (Federated Learning) perception. This technique gives the flexibility to gather requirements on both parties since they will be engaged in deep discussions on what requirements are needed to complete the end product. Although this method is time consuming and costlier in terms of the preparations and executions, but the data that gathered using this approach will greatly benefit the research more than any other methods.
Method 03: Prototyping
Prototyping approach will be a good asset to the research's design and development phases since it may help author to evaluate the wireframes and early prototypes and get feedback to improve it while shaping towards the end goal of the research project. Because the essence of this project is study, trial and error are the greatest choice for arriving at the optimum answer. However, it's unavoidable that the researcher might have to spend a significant amount of time on prototyping the proposed system since the domain and technical aspects are quite new.
Method 04: Brainstorming

Brainstorming was employed throughout the research to gather requirements while thinking and planning for the things that the author doesn't have much prior knowledge of. Since research is a creative process which gets an ample amount of time to formulate research aim, designing and implementing brainstorming helps to identify new innovative ways to approach the goals. However, since brainstorming is self-employed and unmoderated, the decisions and steps that author take will not be as accurate as possible. Moreover, every self-employed decision and steps should have to be evaluated by the experts for further improvements.

4.5 Data Analysis and Presentation of the Outcome

4.5.1 Analysis of Literature Review Findings

Table 9: Analysis of LR findings

Findings	Citation
The fulfillment of trustworthy AI principles in an AI service ensures the trustworthiness which also adopts into the creation of Trustworthy FL architecture.	(Tariq et al., 2023)
The competitor mindset of the clients in cross-silo settings always makes it necessary for a trusted third-party central server to achieve fair, unbiased, and effective FL training.	(Kairouz et al., 2021)
The amount of literature study went through for cross-silo setting far less than cross-device setting which makes it even harder to implement new innovative ideas to accommodate cross-silo architecture limitations.	(Kairouz et al., 2021)
The proposed system should have human understandable explanations on inner workings of the FL server and client and produce better visualizations with debugging capabilities	(A Li et al., 2023)

The proposed explainable mechanism should not compromise model performance or privacy to accomplish its transparency and visualizing goals.	(Corcuera Bárcena et al., 2022)
Communication overhead is always a common threat in any given FL environment; thus, it's critical to optimize the rounds for training.	(Huang, Huang and Liu, 2022)
The proposed system should not be vulnerable to any malicious actors or activities in any given point in the training or testing periods.	(Kairouz et al., 2021)
Any proposed FL mechanism should always ensure privacy and security of data preservation as their main goal even when it has a different set of goals to complete.	(Tariq et al., 2023), (S'anchez et al., 2023)

4.5.2 Analysis of Interview Findings

Interviewing the domain and technical experts was chosen as the main requirements gathering process of this research project and several experts were identified and requested to do the interviews. Among the interviewees, there were PHD and master's students in Federated Learning, lecturers and researchers who are experts in Machine Learning, Trustworthy AI and privacy-preserving domains in general. Most of the interviews were done using online questionnaire form due to the busy schedules of the interviewees and the complete interview questions with the explanations has been added to the **Appendix E**.

Table 10:Interview findings in Thematic Analysis

Codes	Theme	Analysis
Regulations and Trustworthy AI, Effectiveness of FL, Data privacy,	Research gap and project scope	All the interviewees accepted that it is important to ensure trustworthiness and explainability in a typical AI system like FL due to the importance of the upcoming regulations. And further, they're highlighted how author should approach his wide and novel research gap and

Transparency and accountability, Explainability in Federated Learning		appreciated for choosing such interesting research gaps. Hence, they all approved that the research gap will perfectly align with the cross-silo FL sub-domain and mentioned that this will surely benefit the overall subdomain to grow on a positive note and establish new researchers and research ideas.
Explainable Mechanism, XAI, Interpretability	Features and main components of the prototype	All the interviewees accepted that the ability to interpret black boxes by the explainable AI platform will surely be a key asset for this research and finding effective ways to implement the mechanism under the FL setting and mentioned to be cautious always of data privacy. They further approved the other selections of techniques and encouraged the author to work on the XAI domain to introduce a novel approach specific to FL.
Privacy and security, data privacy, Communication	Potential Drawbacks and challenges	After considering the prototype ideation with aims and objectives the interviewees have highlighted potential drawbacks that could hinder the system. Data privacy and security was the main aspect which was talked about mostly since the project involves explaining and interpret the FL workflows. Communication overhead, and performance issues have also been highlighted as important challenges within their conversations.
Metrics, privacy preserving techniques,	Testing and Evaluation	The interviewees pointed out several ways to test the proposed system, testing agnostic abilities to work on various models and datasets, they've also mentioned to use LIME and SHAP metrics find the accuracy trade-off for explainability and FL convergence, privacy preserving techniques for assessing the robustness and trustworthiness of the system.

4.5.3 Analysis of Prototyping Findings

Table 11:Findings through Prototyping

Criteria	Findings
Trustworthiness in cross-silo FL architecture	The author identified that its far more difficult and requires extensive amount of research and time to implement the chosen trustworthy AI principles inside system using a FL framework (PySyft and Flower), therefore, author chooses to create the FL system from scratch using the Torch library. Implementing agnostic abilities to work well with different models and dataset was tried by conducting a series of trial and errors sessions with different prototypes, and finally the author decided to work with the PyTorch based models and datasets. Further, initial prototyping of differential privacy made author realize that it needs knowledge on privacy policies and budgets.
Explainability mechanism inside the system	After the in-depth LR review author chosen to use the SHAPLEY additive values as the explainable technique inside the system, where at first glance the development was due to the compatibility with torch library. However, the brainstorming and LR findings helped to achieve the desired goals in implementing the explainable mechanism.
Verifying computational resource consumption for the proposed system	The author conducted the prototype development and testing using an Intel Core i7 (12th Gen) Laptop with 16GB ram and concluded that, at some point specifically when the Shapley values explanation and visualization generation, the computer gets less smooth but apart from that it can be proven that the proposed system can run on other similar general computers.

4.5.4 Analysis of Brainstorming Findings

Throughout the research period, regular brainstorming sessions were conducted by the author aligning with each chapter in the thesis. These sessions were strictly limited to 45 - 60 minutes and during the session the raw ideas and considerations were noted down using the pen and paper without any initial filtering or editing. This unrestricted approach allowed author to come up with broad varieties of concepts and thoughts to emerge from their own domain and technical knowledge around the problem domain. The key findings and crucial points that were selected for further designing and development of the project are outlined at **Appendix F**.

4.6 Summary of Findings

Table 12:Summary of the Requirements Gathering Findings

Findings	Literature Review	Interviews	Prototyping	Brainstorming
Validating the research gap of the project	X	X		X
Determining the potential techniques, methods and metrics for the successful completion of the project	X	X	X	X
The architecture has to be novel and incorporate trustworthy AI principles to ensures the overall trustworthiness and explainability and adapt to the cross-silo setting.	X	X		X
Consideration and confirmation on the selection of techniques for the explainable mechanism inside the architecture.	X	X	X	
The system should maintain the performance amidst the other processes to resemble an effective FL environment.	X		X	X
The implementation of the methods and algorithms have to be	X	X	X	X

agnostic as possible to ensure the overall system is agnostic to specific organizations with different use cases.				
The identification of Testing and evaluation methodologies along with the potential drawbacks and scope deviations.	X	X		

4.7 Context Diagram

The figure below describes the context diagram, also known as a level 0 Data Flow Diagram (DFD), illustrating the data interactions of the proposed system with its broader environment. It serves as an essential tool for understanding the fundamental processes and data exchanges significant to the functioning of the proposed system.

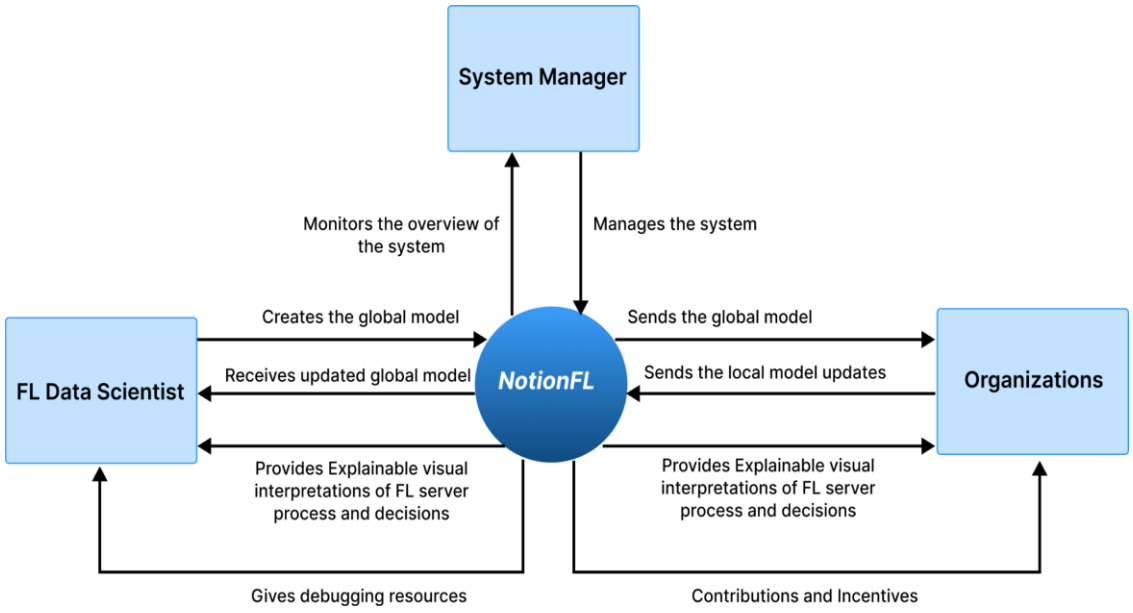


Figure 9:Context Diagram (Self-Composed)

4.8 Use Case Diagram

Figure below showcases the use case diagram of the system while the use case description is given in the next section.

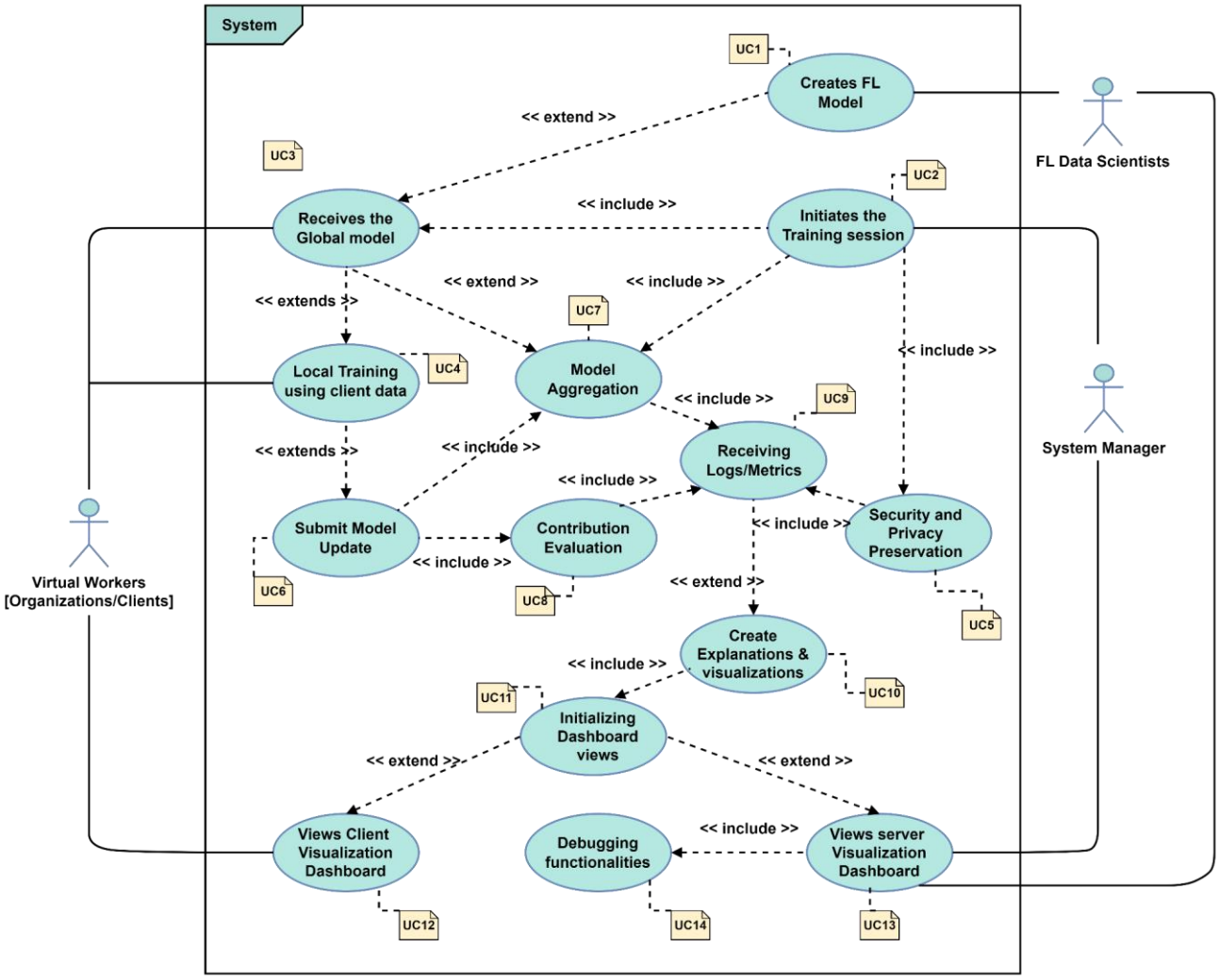


Figure 10: Use case Diagram (self-composed)

4.9 Use Case Descriptions

The significant use cases of the proposed system are described below with comprehensive descriptions and other sub use cases ones have moved to **Appendix G.**

Table 13: Use case description for Local training using client data

Use Case Name	**Local Training using Client Data**	**ID 04**
Description	The clients will start the local model training after receiving the global model using their own dataset.	

Participating actors	Client (Organizations)
Preconditions	Receive Global model
Extended use cases	• Submit model updates. • Receive global model
Included use cases	None
Main flow	• Receives global model. • Dataset Preprocessing • Model Training for certain rounds
Alternative flow	None
Exceptional Flows	• Unable to receive a global model from the server due to communication issues. • Unable to perform the Model training due to low resources. • Unable to perform the training due to lack of datasets.
Post conditions	Submit Model updates

Table 14:Use case description for creating explanation and visualizations.

Use Case Name	**Create Explanations and Visualizations**	**ID 10**
Description	The explainable mechanism generates and creates explanations and visualizations based on the logs, data and metrics received from the FL workflows.	
Participating actors	Client (Organizations), FL data scientists	
Preconditions	Storing the data/metrics/logs received from the FL workflows	
Extended use cases	Initialize dashboard views	
Included use cases	None	

• **Main flow**	• Access metrics/data/logs from the FL workflows (contribution evaluation, secure aggregation, privacy preservation etc.). • Generate explanations on certain data related to the workflows. • Generate visualization based on the data and explanation for the workflows.
Alternative flow	None
Exceptional Flows	• Unable to access data/metrics or logs from the storage due to communication faults. • Unable to perform the explanation generation due to unsupported data formats and errored data. • Unable to store the visualization and explanations due to communication difficulties.
Post conditions	Store the explanations and visualizations on local database

4.10 Requirements with Prioritization

The essential requirements of the proposed NotionFL system have been categorized using the MoSCoW principle and highlighted in the tabular data below.

Table 15: Summary of 'MoSCoW' Prioritization levels

Priority Level	Description
Must have (M)	These requirements are so crucial for developing and successfully completing the Minimum Viable Product (MVP) of the project.
Should have (S)	The requirements are important but not essential or required to develop and complete the MVP of the project.
Could have (C)	These requirements are desirable but not crucial or essential for the development and completion of the MVP.
Will not have (W)	These requirements are identified as the unimportant or not requiring completing the MVP of the project inside the timeline.

4.10.1 Functional Requirement

Functional requirements of the proposed project are moved to **Appendix H.**

4.10.2 Non-Functional Requirement

Non-Functional requirements of the proposed project are moved to **Appendix I.**

4.11 Chapter Summary

This chapter utilized different requirement elicitation techniques to explore various requirements for the preliminary research of the project. The rich picture diagram, stakeholder diagram, use case diagram and the context diagram were presented in the chapter to support the requirements elicitation process. The complete analysis of the requirements was discussed in detail along with the reasons for the finalized decisions and finally the functional and non-functional requirements were formed for the proposed project.

CHAPTER 05: SOCIAL, LEGAL, ETHICAL, & PROFESSIONAL ISSUES

5.1 Chapter Overview

This chapter examines the social, legal, ethical, and professional challenges inherent in the project, detailing mitigation strategies in line with the 'University of Westminster Code of Practice', 'Code of Research Good Practice', and the 'BCS Code of Conduct'.

5.2 SLEP issues and Mitigations

Table 16:SLEP issues and mitigations

Social	Legal
• One-on-one interviews and interview questionnaire were conducted with consents and provided necessary evidence and documents to prevent communication conflicts and maintains integrity. • The project is free from religious, political, ethical, or emotional biases.	• All academic materials and graphics obtained for documenting the research thesis have been well cited. • The languages, libraries, frameworks, and tools utilized to develop the project were licensed under open-source or education-based.
Ethical	**Professional**
• Data collected for this research project is solely accessible by the researcher with restricted access for others. • The project does not involve, gather, or store any organizational or private data.	• This research project adheres to both academic and industry standards. • This research project is not falsified in any manner to benefit the author. • The project software developed in highest standard of code quality with MIT license.

5.3 Chapter Summary

This chapter identifies and addresses potential social, legal, ethical, and professional challenges in each section, outlining appropriate mitigation strategies.

CHAPTER 06: DESIGN

6.1 Chapter Overview

This chapter introduces and discusses design conclusions of the proposed system, which will be supported by various diagrams like the high-level system architecture diagram, class diagram, and sequence diagrams. All the design decisions that were made during this chapter was based on the requirements-gathering phase. Additionally, it includes design strategies, and goals that were laid out to explain the selected design paradigms as well as the system user interfaces.

6.2 Design Goals

The stakeholder viewpoints, non-functional requirements and project objectives were utilized to form the design goals of the system. Important design goals and their explanations are outlined below in Table.

Table 17: Design Goals of the Proposed System

Design Goals	Description
Reliability	The contribution evaluation results generated by the proposed system should be reliable and with explanations, that way clients can be able to trust the system.
Performance	The implementation of the core functionalities should not have a negative influence on the system's performance. The system must maintain its performance with different training customizations.
Flexibility	The core architecture of the proposed system and the core algorithms and functionalities must be adaptable to accommodate any FL configurations.
Transparency	The system should explain and interpret the important FL workflows and processes with proper explanations and visualizations with proper design components.

Reusability	The proposed system should be designed as a framework that can be used as a novel cross-silo framework. The core functionalities of the framework should be built in a way that could be reusable for adapting to different use cases in future FL systems.

6.3 System Architecture Design

6.3.1 System Architecture Diagram

The project NotionFL adopts a three-tier layered software architecture and visualizes below in illustration figure.

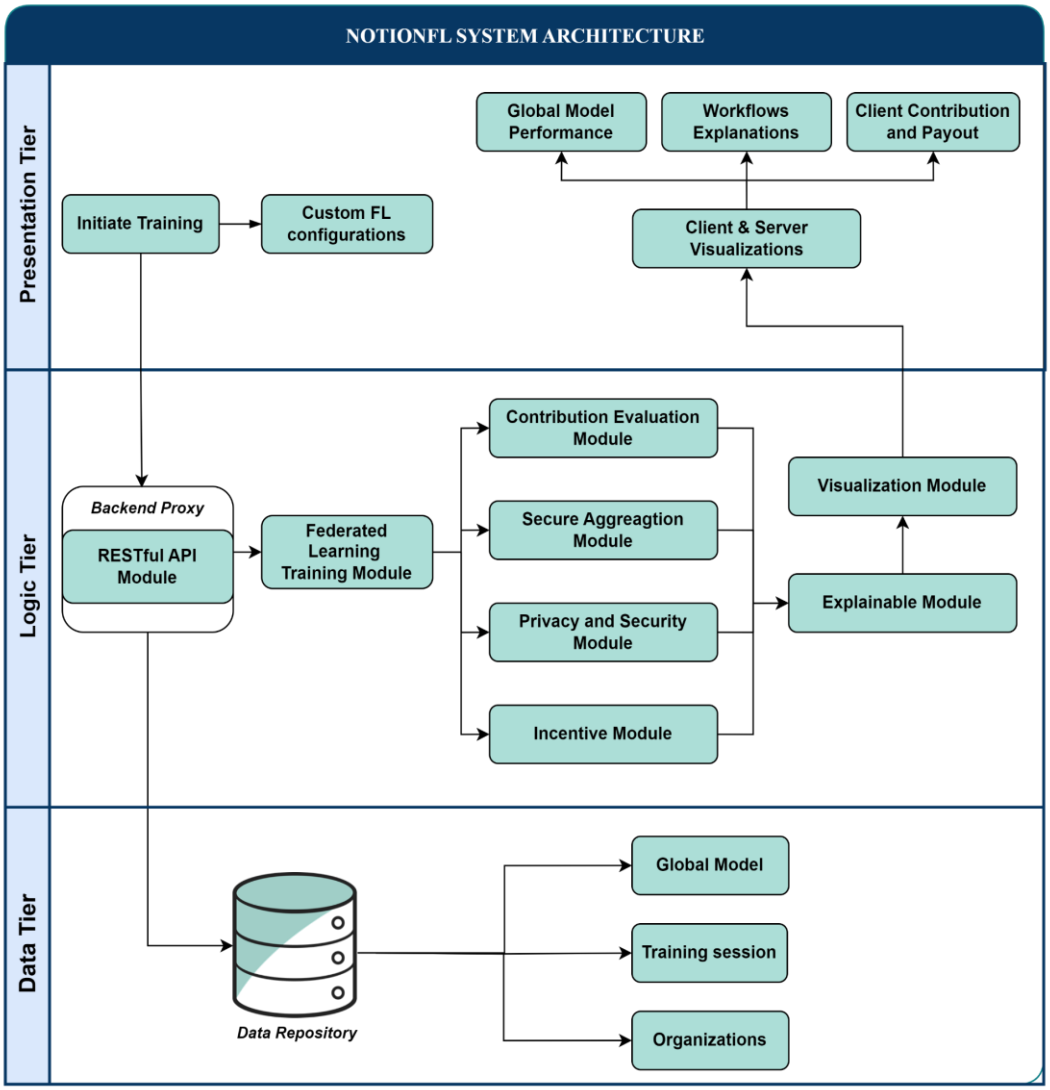


Figure 11:System Architecture Diagram (self-composed)

6.3.2 Discussion of System Architecture Layers

For the proposed NotionFL system, a N-tier architecture diagram was designed on three tier architecture concepts (presentation, logic, and data tier) to ensure the overall reliability of the system. Above three tiers along with its modules have been explained in detail below.

Table 18: Summary of System Architecture

Layer	Components	Description
Presentation Layer	Initiate Training	The user interface via system manager initiates the training.
	Custom FL Configurations	The interface responsible for enabling custom configuration settings for data scientists in FL training.
	Client/Server Visualizations	The two interface views responsible for client and server explainable visualizations related to the training and results.
	Global model Performance	The interface presents the results, statistics about the global model's performance and recent updates.
	Client contribution and Payout	The interface responsible to provide organizations (clients) with their contributions and associated payouts.
Logic Layer	RESTful API module	The module in charge of connecting the front and backend of the application with endpoints while performing important tasks and handling the database.
	FL training Module	The module in charge for managing the Federated learning training among the clients.
	Contribution Evaluation Module	This module is responsible for evaluating the client's contribution based on their received model updates
	Secure Aggregation Module	This module is responsible for aggregating the local model updates to the global model.

	Privacy and Security Module	This module oversees preserving privacy and security inside the system.
	Incentive Module	This module is in charge for the incentive allocation of the clients based on their contributions
	Explainable Module	This module is responsible for receiving the logs and metrics of the FL workflows to create explanations.
	Visualization Module	The module is in charge of creating visualization based on the explanations for both client and server views.
Data Layer	Training Sessions	Storing each training session with vital information such as clients' contributions, round inference time, logs and metrics related to the workflows and anomalies.
	Global Model	Storing the versions of global models for the final use case and evaluation.
	Organizations	Storing the organization(clients) details and their training history.

6.4 System Design

6.4.1 Choice of Design Paradigm

Design paradigms in general help developers to visualize and plan a solution to a specific problem. These paradigms are usually chosen based on the system's nature and behavior and two of the most popular design paradigms are:

1. Structured System Analysis and Design (SSADM)
2. Object-Oriented Analysis and Design (OOAD)

The OOAD paradigm explores the behavior and interaction of the real-world entities and maps it to the system, which is quite ideal for the projects that have changes in the user requirements. However, since this project already laid out its requirements and intends to design a

new novel approach on cross-silo FL setting to accomplish the research gaps in existing literature, **SSADM** was the ideal selection for this project. Furthermore, SSADM helps to break down the project into smaller modules and aids during the requirement analysis process while being straightforward to understand.

6.5 Detailed Design Diagrams

6.5.1 Component Diagram

The illustration below highlights the insignificant components and modules of the proposed cross-silo FL system with their relationships and interactions. The diagram has been moved to **Appendix J**.

6.5.2 Data Flow Diagram

The illustration below visualizes the level 1 Data Flow Diagrams (DFD) of the proposed cross-silo FL system, which further highlights the main components of the level 0 DFD diagram drawn for the proposed system on the SRS chapter. The diagram has been moved to **Appendix K**.

The Level 1 data flow diagram of the proposed system consists of 5 main data processes, which have been highlighted below.

1. Create Global Model
2. Federated Learning Training
3. Data Collector Mechanism
4. Explainable Mechanism
5. Visualize Dashboard

Out of these five major processes, the author has further filtered three utmost important processes and illustrated their level 2 diagram below.

The illustration below visualizes the Level 2 DFD diagram of the Federated Learning training sessions.

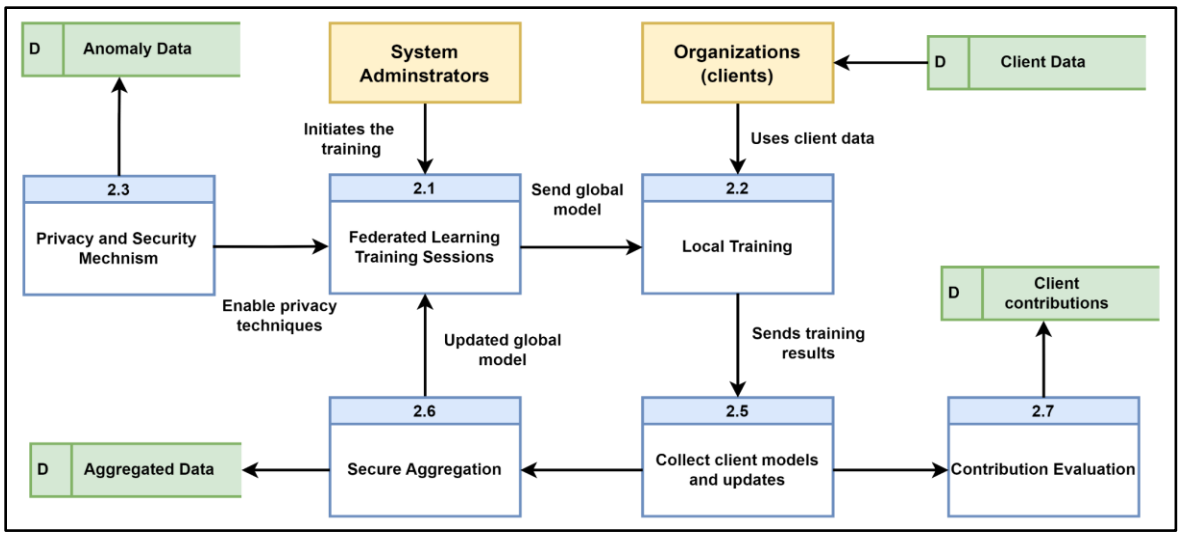


Figure 12: FL training Process - Data Flow Diagram - Level 2 (self-composed)

Next, the illustration below visualizes the level 2 DFD diagram of the Explainable mechanism.

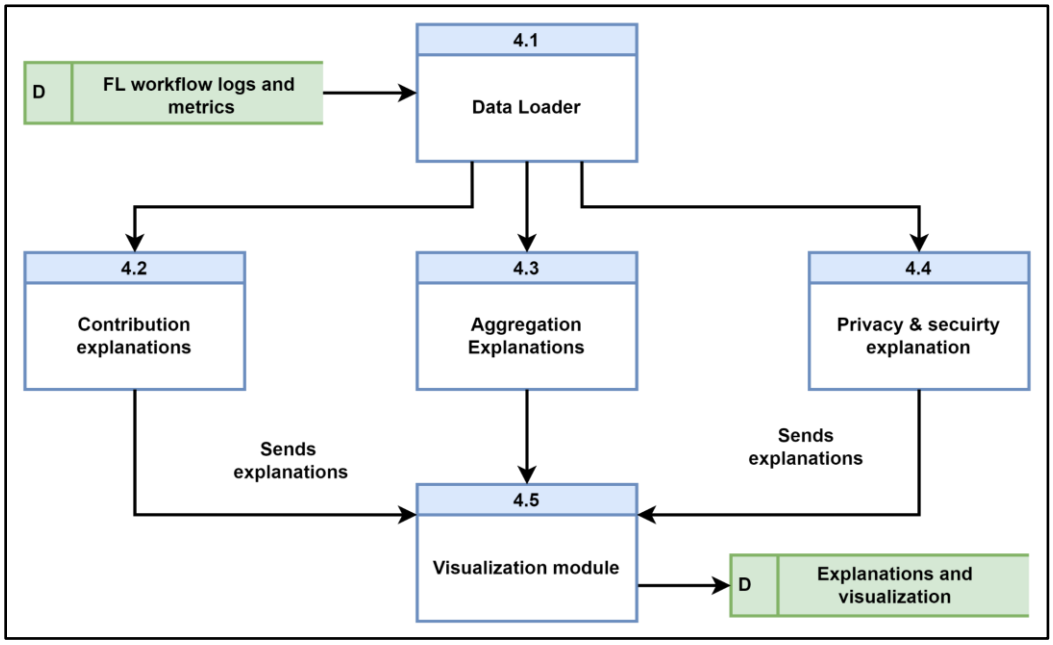


Figure 13: Explainable Mechanism - Data Flow Diagram - Level 2 (self-composed)

Finally, the illustration below presents the Dashboards visualization process of the proposed system.

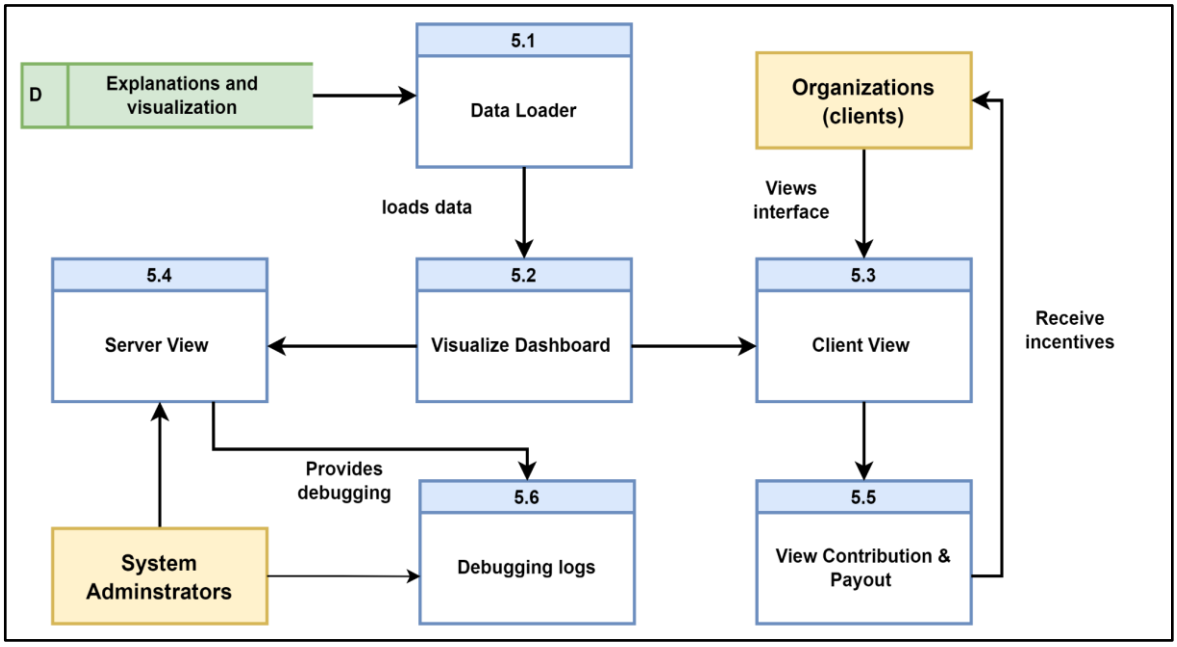


Figure 14: Dashboard Visualization - Data Flow Diagram - Level 2 (self-composed)

6.5.3 Trustworthy Cross-silo Architecture Design

As depicted before in the first chapter and in chapter two LR, the author introduces a new novel architecture for cross-silo Federated Learning architecture using trustworthy AI principles to tackle the identified research gaps. Since the project research gaps involve the trust and interpretability of cross-silo FL systems, the creation of new architecture would greatly help the author to accomplish his research objectives and aims.

The architecture, as conceptualized, addresses the critical aspects of the trust and interpretability, which are essential to the effective cross-silo FL system. To illustrate the architecture in detail, the author has presented a high-level architecture diagram with various components of the systems and their interconnections providing a clear overview on how the **Trustworthy AI principles** have been embedded within the framework.

The trustworthy AI principles that have been chosen are ***explainability, fairness, privacy, robustness, and accountability.***

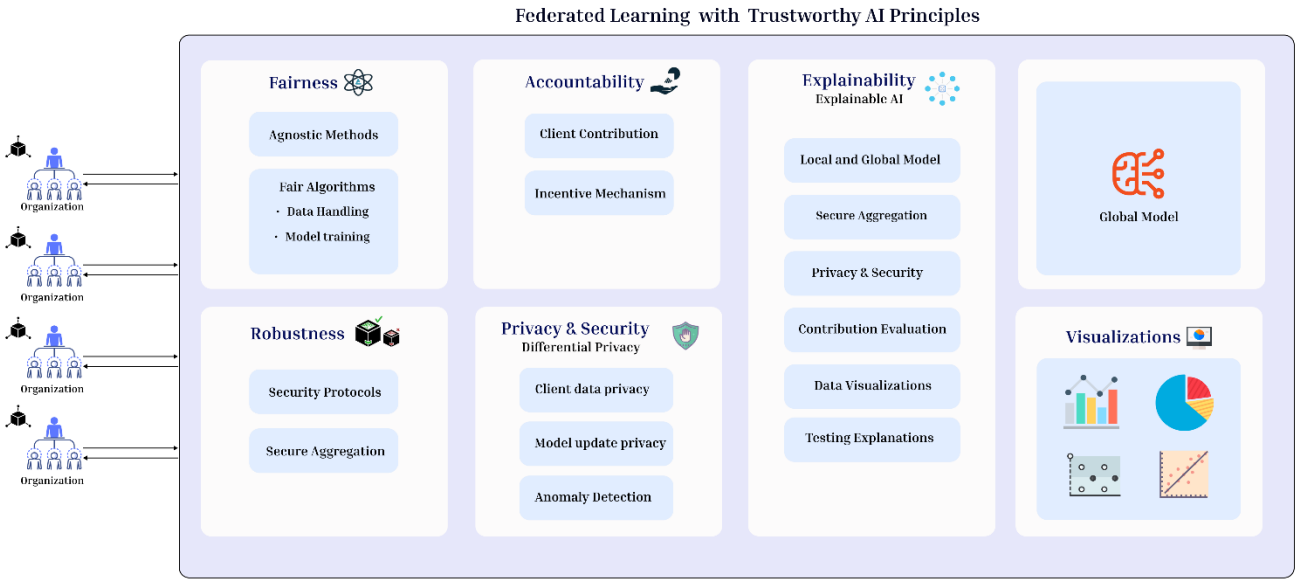


Figure 15: Trustworthy Cross-silo FL Architecture

Explainability: The **XAI mechanism** has been used as the main technique to interpret the FL workflows as well to create visualizations to enhance the overall transparency of the FL architecture providing clear insights into the decision making.

Fairness: The **agnostic methods** along with algorithms ensures fairness of the system by managing equitable data handling, unbiased model training, and fair explanations with justifications helps to tackle potential biases with the architecture.

Privacy: The architecture leverages the advanced crypto graphical methods like **differential privacy** to safeguard the entire user data, model updates and the overall system performance, aligning with robust privacy preservation methodologies.

Robustness: To ensure the security of the system architecture against the potential anomalies and attacks, the **robust security protocols** and **secure aggregation mechanisms** are employed within the architecture to ensure resilience of the system.

Accountability: This aspect involves in clear implementation of the client contribution evaluation and incentive mechanism to assess the contributions within the FL process to be more accountable. Each of the above components have been tailored to meet the specific set of requirements and

constraints within the cross-silo FL setting, which was derived from the research insights. This structured approach is crucial to effectively integrate the trustworthy AI principles within the architecture to ensure the overall trustworthiness and the effectiveness of the FL environment.

6.5.4 System Process Flow Chart

The illustration below describes the **Activity Diagram** of the proposed NotionFL system. It visualizes the developers' viewpoint of the general application workflow of the prototype system, with input, processes, and outputs.

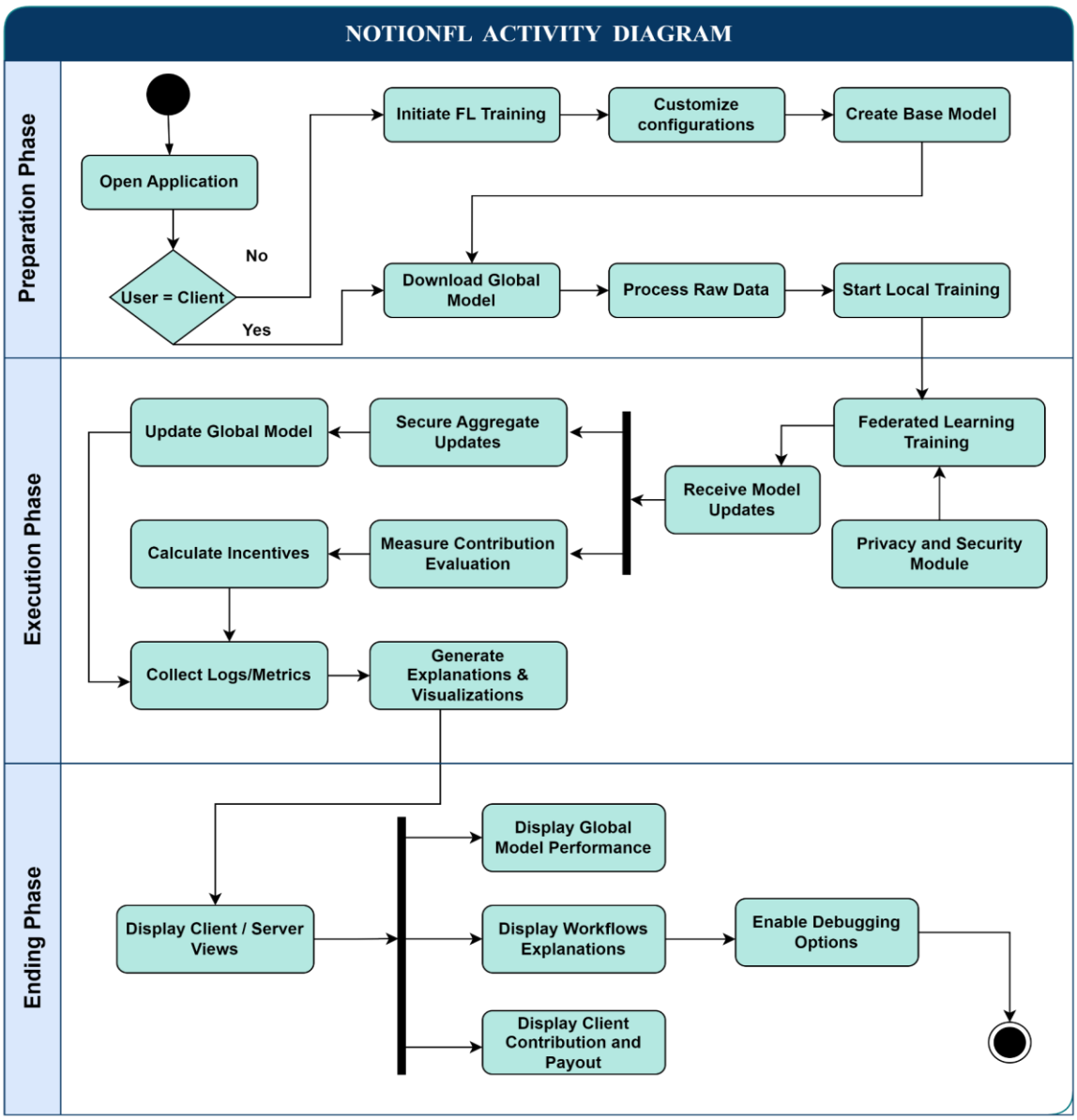


Figure 16: Activity Diagram of the Proposed System (self-composed)

6.5.6 User Interface Design

6.5.6.1 Low level Fidelity Wireframe Diagram

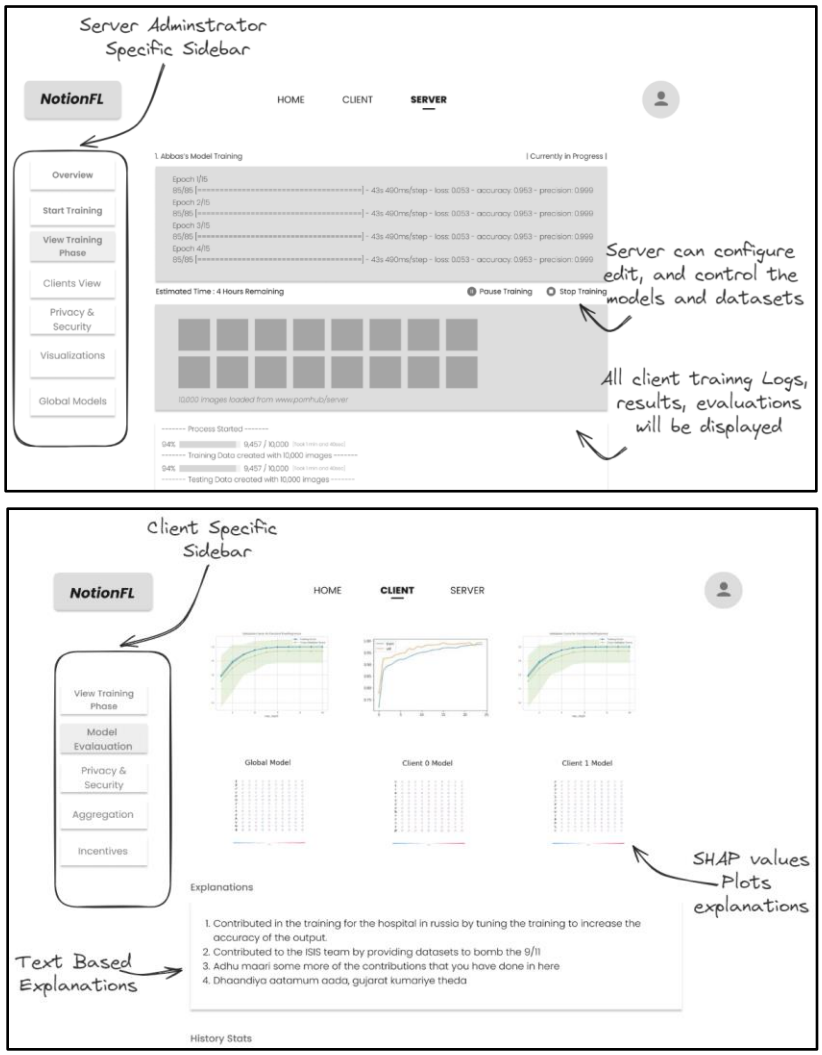


Figure 17: Low fidelity wireframes for the Proposed System (self-composed)

6.6 Chapter Summary

This chapter highlights the design components of the project, where it begins with the chosen design goals and high-level architecture diagram. Further it discusses the design techniques that were selected along with different design diagrams as the explanations to structure the proposed project. Further, the novel designing and creations to the project were presented along with a detailed analysis. Lastly, the chapter concludes with the UI design of the proposed system.

CHAPTER 07: IMPLEMENTATION

7.1 Chapter Overview

This chapter discusses the initial implementation of the proposed cross-silo system, starting off with a selection of technology stacks, programming languages, frameworks, libraries, etc. It also provides detailed explanations on the chosen technologies and discusses how each will be utilized for the development. Lastly, chapter presents the development of core functionalities of the proposed system along with its user interfaces.

7.2 Technology Selection

7.2.1 Technology Stack

The figure below represents the technology stack selected for the implementation of the proposed system. The diagram consists of five layers of different technologies associated with the proposed system.

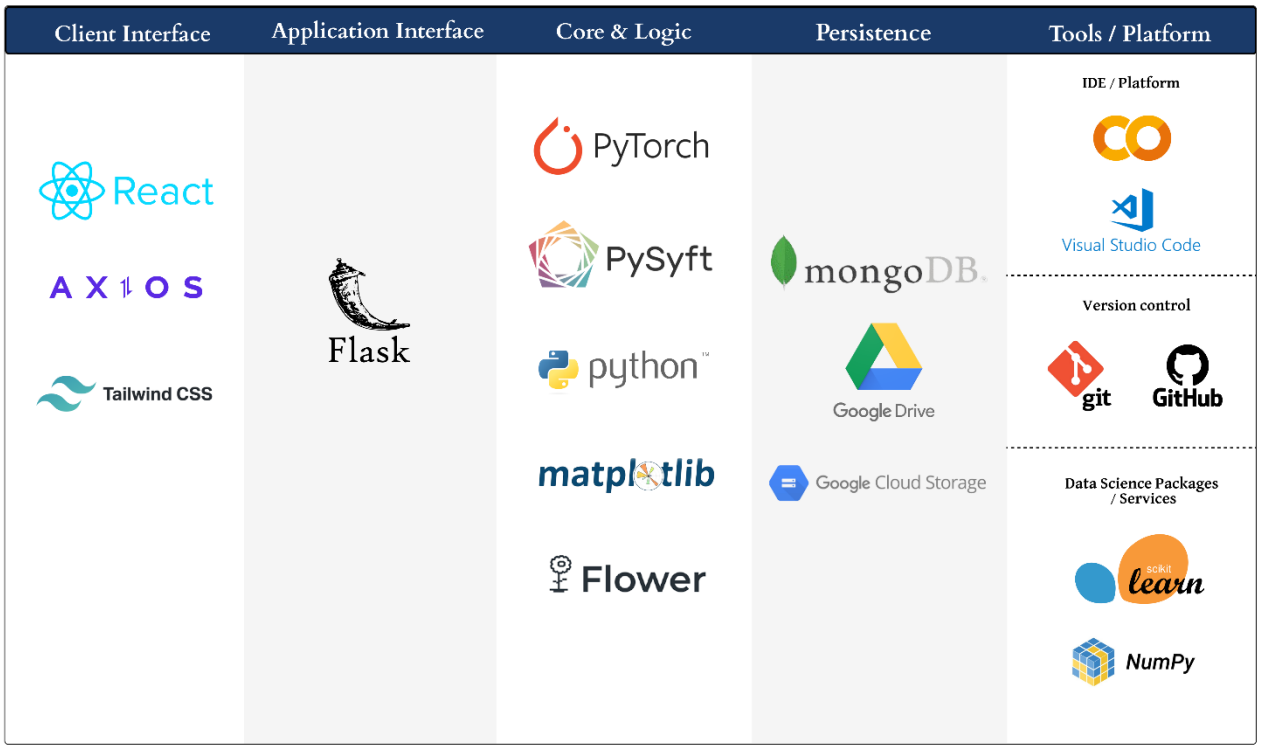


Figure 18:Technology Stack (Self-Composed)

7.2.2 Data-set Selection

The proposed system is designed to operate with any dataset that reflects FL training since it is intended to be agnostic for datasets, however, selecting the proper dataset was crucial to simulate the local training behavior of the organizations in the proposed system. The author selected popular datasets used among FL community from the PyTorch library, such as **MNIST** and **CIFAR10**, with the goal of experimenting with new architecture and mechanisms. These datasets are well-suited to cross-silo FL simulations due to their sizes and formats, which replicate real-world situations and facilitate comprehension of FL systems.

7.2.3 Development Frameworks

The technological review in the LR chapter provides specifics on the development framework that was selected after a great deal of research on well-known FL frameworks. The table below outlines the main arguments for the selection of PyTorch's **'PySyft'** framework for FL simulation atop the new cross-silo architecture after it underwent a successful review.

Table 19:Development Framework and Justifications

Development Framework	Rationale For Selection
Pysyft	As a deep learning python package, this framework ensures both security and privacy. By default it includes differential privacy, multiparty computation, and encrypted computation techniques. The research community is a great advantage over the other frameworks, with over 10,000+ members, who can provide you with support at any time. The OpenMined platform also provides you with tailored courses for better understanding of the framework with a well-organized documentation.
Flask	Since the FL framework is based on python language, the Flask microservice framework has been chosen as the development framework to build the server application programming interface (APIs). Being a well-known python framework with a strong documentation support it justifies the author’s selection for this project.

7.2.4 Programming Languages

The Python programming language has been chosen to develop the proposed system since the selected FL frameworks are based on python language. Python is one of the go to programming languages for data science and ML related projects due to having a wide range of support with the libraries and technologies.

Table 20: Selection of Programming Languages and justifications

Programming Languages	Rationale For Selection
Python	Having a wide range of support to the existing ML libraries and techniques with better support around the communities and forums. Further it was also selected due to the selection of FL development framework 'Pysyft'.
JavaScript	For the front-end development of the proposed system, React JS has been chosen to make the user interface interactive with rich user experiences.

7.2.5 Selection of Libraries

7.2.5.1 Selection of ML/DL Libraries

Table 21: Selection of ML/DL Libraries and justifications

Libraries	Rationale For Selection
PyTorch	Since the author choose PySyft as the FL development framework which is wrapped under the deep learning library PyTorch. Hence, the PyTorch library was chosen for the development of the FL framework as it uses PyTorch's native syntax and packages.
Matplotlib	This python-based package was chosen to create interactive visualizations and insights on the user interface of the proposed system to explain the FL workflows. It also will be utilized for the visualization of client's datasets in testing phases.

7.2.5.2 Selection of XAI Library

Table 22: Selection of XAI Libraries and justifications

Libraries	Rationale For Selection
SHAPLEY Values	Existing works outlined SV as an agnostic explainer model with being helpful for contribution evaluation use case, fair distribution payouts while being robust for evaluations. Since its enhanced with interpretability abilities with privacy preserving features and transparency it aligns perfectly with the project requirements to develop an explainable mechanism under a trustworthy FL system. Further, the detailed documentation and the strong community made author to select this library for better help.

7.2.5.3 Selection of Front-end Libraries

Table 23: Selection of Front-end Libraries and justifications

Libraries	Rationale For Selection
React	Among the frontend frameworks like Angular, Vue and React, react was chosen over others due to being most popular among developers and research communities. Reacts component-based architecture and state management capabilities can be crucial for development and testing of FL environments in real-time. Compared to other frameworks, having vast community and documentation support, it allows authors to troubleshoot errors related to ML or FL development.
Tailwind CSS	It is considered as one of the best UI libraries for React and other frontend frameworks due to having an extensive amount of UI elements with easy installation and integration. Since the proposed FL system consists of a dashboard with minimal designs utilizing this UI library can be vital.

7.2.6 IDE

Since the proposed project involves a Full-stack development phase where python language is involved in the server side and react has been chosen as the front-end framework, the author explored a common IDE which suits both. These requirements immediately eliminated language specific IDEs and left the path to IDEs like atom, sublime, and visual studio code. VS Code has been chosen with certain reasons and those rationales have been highlighted below.

Table 24: Selection of IDEs and justifications

IDE	Rationale For Selection
Visual Studio Code	Having an extensive number of libraries, languages, and extensions support, aligning with Reach and Python language support. Being able to build industry standard full-stack applications with version control and debugging support with a better user interface. Supports to build and test the proposed FL simulation system with a more streamlined and efficient development environment.
Google Collab	As the Federated learning involved in resource intensive tasks like handling multiple ML model training etc., the author selected to use Google Collaboratory due to having free GPU access. Since it supports Python in default, perfectly aligns with the project's selection of programming languages and frameworks. It was considered over the other IDEs like Jupitar notebook, Kaggle, and Data lore due to having massive storage support for google drive, giving the author more storage space.

7.2.7 Selection of Persistence Service

The proposed project involves storing data like training session logs, FL workflow metrics and visualization explanations where the author intended to use **MongoDB** Cloud storage over the other options like Google Firebase, Azure storage and Amazon S3. Since the Project's frontend framework React and Python based server supports well with MongoDB storage. Furthermore, a

well document-oriented storage structure of nested data can be suitable for FL workflows data storage. Alongside, **Google Cloud Storage** will also be utilized for storing the training results, visualizations, models and logs since it has high storage capability than Free-tier MongoDB atlas.

7.2.8 Summary of Technology Selection

Table 25: Summary of Technology Selections

	Component	Tools
Server	Programming Language	Python
	Microservice framework	Flask
Logic/Model building	DL Library	PyTorch & PySyf
	Other Libraries	Matplotlib, NumPy, Scikit Learn
	XAI Library	SHAP
Client side	Front-end Library	React with Tailwind
Persistence	Data Storage	MongoDB Atlas, Firebase & Google Drive
Services	Version control	Git & GitHub
	IDE	Visual Studio Code
	GPU runtime environment	Google Collaboratory

7.3 Implementation of the Core Functionality

As previously mentioned on chapter 1, project NotionFL will consist of a novel FL architecture with core FL functionalities implemented with different technologies to incorporate Trustworthy AI principles. The core components of the proposed system have been highlighted below with detailed explanations.

7.3.1 Setting up Federated Learning Environment

i. Federated Learning Training Configuration

This configuration file is the starting point of the FL training which consists of the basic arguments needed for the FL environments to work accordingly as configured by the server administrator. These arguments can also be seen utilized on certain FL workflows like, ***noise_multiplier*** and ***clip_threshold*** can be used for differential privacy module where it allows to adjust the privacy preservation limits. Likewise, training initiator also known as the central server can specify specific arguments to configure the FL workflows to work differently.

NotionFL-BE > config.yml > ...

```
...
batch_size: 64
clip_threshold: 1.0
device: cpu
epochs: 10
eval_every_n_rounds: 1
fl_rounds: 2
learning_rate: 0.01
noise_multiplier: 0.1
num_clients: 4
incentive_pool: 10000
```

Figure 19: FL Configuration File (Self-Composed)

ii. FL Training Module

The training loop in the NotionFL system will consist of several object classes like, server, clients, data loaders, data collector and explainable mechanism. These object instances will be required to be initialized at the start of the FL training to be able to use across different places in the FL training. Each objects have their own built-in methods which they will utilize for their assigned work in a structured manner to effectively train the model.

```python
def main(training_id, config):
    train_loader, test_loader = get_data_loaders(config['dataset'], batch_size=config['batch_size'])
    client_data_loaders = split_client_data(train_loader.dataset, num_clients=config['num_clients'], batch_size=config['batch_size']
    if config['dataset'] == 'MNIST':
        global_model = MNISTModel().to(config['device'])
    elif config['dataset'] == 'CIFAR10':
        global_model = CIFAR10Model().to(config['device'])

    server = FLServer(global_model)
    clients = [FLClient(i, global_model, client_data_loaders[f'client_{i}'], test_loader, config['device'])
                                                                for i in range(config['num_clients'])]
    for round in range(config['fl_rounds']):
        client_updates = []
        for client in clients:
            model_updates = client.train_and_get_updates(config['epochs'], config['learning_rate'])
            client_updates.append(model_updates)
            client.evaluate(round)
        aggregated_state_dict = perform_fedavg_aggregation(global_model.state_dict(), client_updates)
        global_model.load_state_dict(aggregated_state_dict)

        accuracy = server.evaluate_global_model(test_loader, config['device'])
        print(f"Round {round + 1}/{config['fl_rounds']}: Global Model Accuracy: {accuracy}")

if __name__ == "__main__":
    training_id = sys.argv[1]
    config = {
        'dataset': 'MNIST',  # or 'CIFAR10'
        'num_clients': 5,
        'epochs': 3,
        'batch_size': 64,
        'learning_rate': 0.01,
        'fl_rounds': 10,
        'device': 'cuda' if torch.cuda.is_available() else 'cpu'
    }
    main(training_id, config)
```

Figure 20: Simplified FL Training Loop (self-Composed)

The above Figure presents the simplified version of the main training loop of the project NotionFL where it follows a certain set of processes for a round of training. The main flow of the training loop is outlined below.

- Retrieving the global model from the previous round or initializing a new model.
- Create and initialize clients(organizations) as the virtual workers.
- Load client data loaders and distribute the global model to the virtual workers.
- Clients training the models locally and sending the model updates.
- Before aggregation differential privacy is applied to the model parameters.
- Aggregation receives the model updates and aggregates to the global model, if there's another FL round global model will be loaded again by the client, and it starts the training.
- Once the FL rounds were finished, the contribution evaluation mechanism will calculate the client contributions along with the incentives.
- Data collector mechanism stores data and metrics of every FL processes.
- Explainable mechanism will use the data collector to retrieve required data and creates visualizations and explanation on FL processes.

iii. FL Data Loaders Module

This module sets up data handling for FL training and evaluation, where a central dataset is split into several parts according to the given number of clients and distributes unevenly to reflect the real-world setting. The **MNIST** and **CIFAR-10** datasets are prepared with suitable transformations and split into batches with for efficient processing during training.

```
def get_data_loaders(dataset_name, batch_size=64, train_shuffle=True, test_shuffle=True):
    if dataset_name == 'MNIST':
        trainset = datasets.MNIST(root='./data', train=True, download=True, transform=mnist_transform)
        testset = datasets.MNIST(root='./data', train=False, download=True, transform=mnist_transform)
    elif dataset_name == 'CIFAR10':
        trainset = datasets.CIFAR10(root='./data', train=True, download=True, transform=cifar_transform)
        testset = datasets.CIFAR10(root='./data', train=False, download=True, transform=cifar_transform)
    else:
        raise ValueError(f"Dataset {dataset_name} not supported.")

    # Create data loaders
    train_loader = DataLoader(trainset, batch_size=batch_size, shuffle=train_shuffle)
    test_loader = DataLoader(testset, batch_size=batch_size, shuffle=test_shuffle)

    return train_loader, test_loader

def split_client_data(dataset, num_clients, batch_size=64):
    total_data_points = len(dataset)
    indices = torch.randperm(total_data_points).tolist()
    # Calculate split sizes for each client
    split_sizes = [total_data_points // num_clients + (1 if i < total_data_points % num_clients else 0) for i in range(num_clients)]
    client_data_loaders = {}
    index = 0

    for i in range(num_clients):
        client_size = split_sizes[i]
        client_indices = indices[index:index+client_size]
        client_data_loaders[f'client_{i}'] = DataLoader(dataset, batch_size=batch_size, sampler=SubsetRandomSampler(client_indices))
        index += client_size

    return client_data_loaders
```

Figure 21:FL Data Loader and Data Splitter (Self-Composed)

7.3.2. Secure Aggregation Module

The Secure Aggregation in this project takes quite a new way of aggregating the clients model update to the global model by accessing the state dictionaries of the client models and aggregates into the global model's state dictionary using the Fed Averaging. The functionality iterated over each parameter in the global model, calculating the average parameters value across all the client models and updates it accordingly. Finally, it returns the updated global model's state dictionary. The function also provides insights into the computational resources used, enabling analysis on this FL process.

```python
def perform_fedavg_aggregation(global_state_dict, client_state_dicts, client_weights):
    if not client_state_dicts:
        raise ValueError("No client model state_dicts provided for aggregation.")

    start_time = time.time()
    start_memory = psutil.virtual_memory()

    aggregated_state_dict = copy.deepcopy(global_state_dict)

    # Aggregate each parameter
    for key in aggregated_state_dict.keys():
        aggregated_state_dict[key] = sum(client_state_dict[key] * weight
                for client_state_dict, weight in zip(client_state_dicts, client_weights)) / sum(client_weights)

    end_time = time.time()
    end_memory = psutil.virtual_memory()
    memory_used = start_memory.used - end_memory.used

    time_overheads = {
        'aggregation_time': end_time - start_time,
        'computational_resources': {
            'cpu_usage': os.cpu_count(),
            'memory_usage': memory_used,
        }
    }

    return aggregated_state_dict, time_overheads
```

Figure 22: Secure Aggregation Module (self-composed)

7.3.3 Contribution Evaluation Module

The contribution of the clients has been calculated based on their model update's performance, which was evaluated using the subsets of clients. The code iterates over the subsets of clients, evaluating each client's model performance with or without the client updates. System calculates the **Shapley Values** (SV) for each client based on its marginal contribution to the aggregated model's performance considering the permutations of the client subsets. Advantages of this novel implementation listed below,

- **Balanced Weighting:** contributions are weighted based on subset size, ensuring a balanced approach to calculating each client's input.
- **Consistency over rounds:** averages contributions over multiple FL rounds, offering a stable view of a client's impact throughout the training.
- **Visualization of contribution** supports creating a plot to visualize the contributions, enhancing transparency.

```python
def calculate_shapley_values(total_rounds, num_clients, client_models, global_models, model_evaluation_func, averaging_func, device):
    shapley_values = {client_id: 0 for client_id in range(num_clients)}

    for round_num in range(total_rounds):
        for client_id in range(num_clients):
            for subset_size in range(1, num_clients + 1):
                for subset in itertools.combinations(range(num_clients), subset_size):
                    if client_id not in subset:
                        continue

                    # Retrieve models for each round in the current subset
                    subset_models = [client_models[other_client_id][round_num]
                                     for other_client_id in subset if other_client_id in
                                         client_models and round_num in client_models[other_client_id]]

                    if not subset_models:
                        continue

                    # Calculate the contribution of the client
                    value_with = model_evaluation_func(averaging_func(subset_models))

                    # Evaluate without the current client
                    if len(subset) > 1:
                        subset_models_without_client = [s for i, s in enumerate(subset_models) if subset[i] != client_id]
                        value_without = model_evaluation_func(averaging_func(subset_models_without_client))
                    else:
                        base_model = global_models[round_num]
                        value_without = model_evaluation_func(base_model)

                    # Update Shapley values
                    weight = (math.factorial(subset_size - 1) * math.factorial(num_clients - subset_size)) / math.factorial(num_clients)
                    shapley_values[client_id] += weight * (value_with - value_without)

    # Normalize Shapley values across all rounds
    for client_id in shapley_values:
        shapley_values[client_id] /= total_rounds

    shapley_plot = create_shapley_value_plot(shapley_values)
    return shapley_values, shapley_plot
```

Figure 23: Contribution Evaluation Module (Self-Composed)

7.3.4 Privacy Module

The project utilizes encryption method, **Differential Privacy** as the mechanism to prevent data privacy issues and secure the client data. Here this function applies differential privacy by getting the gradients of the model parameters and clipping them (limiting them from threshold value) and adding **Gaussian Noise**. Further, it iterates over the parameters tensors and calculates the gradient norm, adds noise and records noise statistics. Finally, it returns the updated gradient model with gaussian noise which limits the adversary to directly extract sensitive information.

- **Clip threshold** limits each gradient's L2 norm to a maximum threshold.
- **Noise Multiplier** Controls the amount of noise added, directly influencing the privacy-accuracy trade-off.

```python
def apply_differential_privacy(model_parameters, clip_threshold, noise_multiplier, device):
    """
    Apply gradient clipping and add Gaussian noise for differential privacy.

    :param model_parameters: A list of parameter tensors from the model.
    :param clip_threshold: The maximum L2 norm of the gradients.
    :param noise_multiplier: The amount of noise to add (related to the privacy budget).
    :param device: The device on which to perform the calculations.
    :return: Dictionary containing noise statistics and computation time.
    """
    start_time = time.time()
    noise_stats = []

    for p in model_parameters:
        if p.grad is not None:
            grad_norm = p.grad.norm(2)
            clip_coef = min(1, clip_threshold / (grad_norm + 1e-6))
            p.grad.data = p.grad.data * clip_coef

            noise = torch.normal(0, noise_multiplier * clip_threshold,
                                 p.grad.data.size()).to(device)
            p.grad.data += noise
            noise_stats.append({'mean': 0, 'std': noise_multiplier *
                    clip_threshold, 'variance': (noise_multiplier * clip_threshold)**2})

    computation_time = time.time() - start_time
    return {'noise_stats': noise_stats, 'computation_time': computation_time}
```

Figure 24: Privacy Module - Differential Privacy (Self-Composed)

7.3.5 Explainable Mechanism Module

The explainable mechanism module is the one responsible for all the interpretations and visualizations of the FL modules and workflows. It utilizes the **SHAP** values, an agnostic **XAI** library to explain the FL workflows using different data that it receives from the workflows.

```python
You, 9 hours ago | 1 author (You)
class FederatedXAI:
    def __init__(self, data_collector_path, device, global_model, server, training_id):
        self.data_collector_path = data_collector_path
        self.device = device
        self.global_model = global_model
        self.server = server
        self.training_id = training_id
        self.file_handler = FileHandler()

    def explain_client_model(self, client_model_state, client_id, test_loader): ...

    def ex_global_model(self, model_state, test_loader): ...

    def explain_global_model(self, test_loader): ...

    def compare_models(self, round_num, num_clients): ...

    # Needs to be fixed
    def explain_combined_models(self, num_clients, data_loader): ...

    def explain_aggregation(self, pre_aggregated_state, post_aggregated_state, data_loader, round_num): ...

    def explain_privacy_impact(self, client_id, round_num, test_loader, privacy_params): ...

    def load_model(self, client_id, round_num, suffix): ...

    def generate_incentive_explanation(self, shapley_values, incentives): ...
```

Figure 25: Explainable Mechanism Module (Self-composed)

The mechanism facilitates the explanations and evaluation of client and global models, compares explanations, analyzes the impacts of differential privacy and secure aggregation, and generates explanations on contribution evaluation and incentive allocation. Being an agnostic model, SHAP values library suits well within the system requirements and aids the author to interpret the FL workflows for both server administrators and clients. The author was intended to present this explainable mechanism as python package for the users to utilize on their FL systems in future.

7.4 User Interface

The proposed system's main server-based user interfaces are outlined below and other user interfaces of the proposed NotionFL system are attached at **Appendix L** section**.**

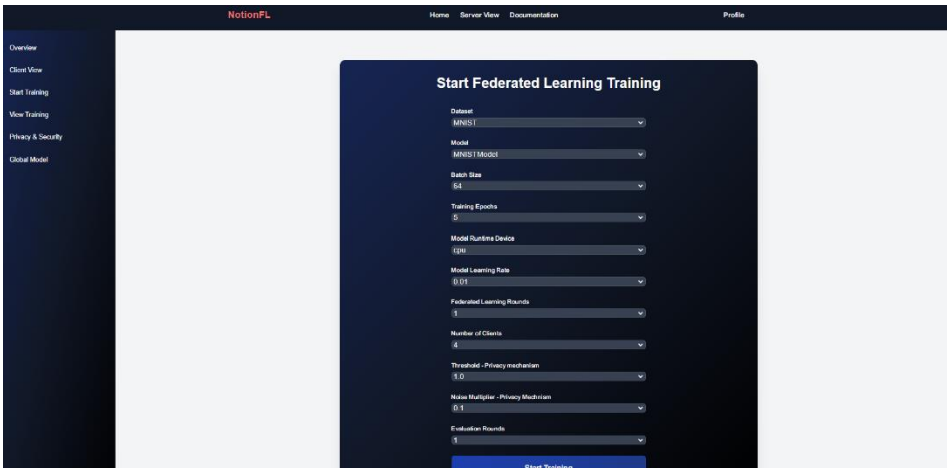


Figure 26: Server FL training starting screen.

7.5 Chapter Summary

This chapter is dedicated to outline the technical stacks including different technologies selected for the proposed system's prototype development. Along with the discussion of the technologies, the author also justifies and gives reasons why these are chosen specific to the project. Moreover, chapter dives into the implementations of the core functionalities of the proposed systems accompanied by the code snippets and detailed explanations. Further the user interfaces of the developed system have been presented with final outputs.

CHAPTER 08: TESTING

8.1 Chapter Overview

The main purpose of this chapter is to provide a comprehensive analysis of the testing methodologies used to evaluate the effectiveness of the new FL. Starting with the test criteria the chapter presents system testing, functional and non-functional testing, module, and integration testing with the discussion of benchmarking and testing limitations.

8.2 Objectives and Goals of Testing

The primary goal of software testing is to confirm that the system is operating according to the specified requirements. Here are the key objectives of the testing process for NotionFL.

- Validate that all models within the NotionFL system are performing as intended and have undergone thorough testing to achieve the best possible outcomes.
- Ensure that the system meets the "Must have" and "Should have" functional requirements, as identified through the MoSCoW technique.
- Ensure that the system meets the critical non-functional requirements.
- Identify potential areas for improvement and bug fixes in the system.

8.3 Testing criteria

The proposed system employs two different testing methodologies which were employed in aiming for decreasing the gap between the expected results and actual results.

Table 26: Testing criteria

Criteria	Values
Functional Testing Strategy	Trustworthy architecture testing, Explainable mechanism testing, FL modules and Integration Testing, API Testing
Non-functional Testing Strategy	Performance Testing, Accuracy Testing, Usability Testing,
Test Platforms	Chrome/browsers, Postman
Testing Techniques	Black Box, White Box, Grounded Evaluation

8.4 Model Testing & Evaluation

8.4.1 Testing Setup

The testing arrangements for the proposed system is highlighted below. The MNIST and CIFAR10 datasets were chosen for the testing along with their models since these datasets can achieve higher performance on low number of epochs. At first place, these models were integrated within the system to test the agnostic ability of the FL architecture to perform under various PyTorch based models and datasets. Therefore, author used different training configurations to test the FL architecture modules, whether they are robust enough to handle both datasets while ensuring the overall system performance. Various FL training configurations that were used for the testing of the novel architecture is summarized below.

Table 27: Testing cases for FL training configurations

Testing Case	FL Training Configurations
TC1	Batch_size:64 epochs:3 fl_rounds:1 num_clients:2 learning_rate:0.01 Clip_threshold:1.0 noise_multiplier:0.1
TC2	Batch_size:64 epochs:5 fl_rounds:1 num_clients:3 learning_rate:0.01 Clip_threshold:1.5 noise_multiplier:0.2
TC3	Batch_size:64 epochs:5 fl_rounds:2 num_clients:5 learning_rate:0.01 Clip_threshold:2.0 noise_multiplier:0.3

The results of the test cases were divided based on the FL architecture modules and presented below with separates sections.

8.2.2 FL Architecture Components Testing Results

The novel architecture consists of several important workflows and their test results have been outlined below and with more justifications on the **Appendix M**.

A. Privacy: Differential Privacy Testing Results

For every testing session, DP was evaluated using on how its impacting the global model performance since it applies certain levels of noise on the models every time.

Table 28:Differential Privacy Test Results

Criteria				Results		
Test case	**Model**	**Clip threshold**	**Noise multiplier**	**Global model Accuracy**	**Computational overhead**	**DP Impact Level on Global Model Performance**
TC1	MNIST	1.0	0.1	0.880	0.01255 sec	Negative
TC2	MNIST	1.5	0.2	0.744	0.04075 sec	Negative
TC3	MNIST	2.0	0.3	0.098	0.01051 sec	Negative

Test case DP results for CIFAR10 datasets are attached to **APPENDIX M.**

B. Robustness: Secure Aggregation Testing Results

Table 29:Secure Aggregation Test Results

Test case	Model	Aggregation Test Results				
		Variance before aggregation	**Pre-aggregation accuracy**	**Post-aggregation accuracy**	**Performance difference**	**Aggregation time (sec)**
TC1	MNIST	2.04345	0.9683	0.8807	-0.0876	0.01833
TC2	MNIST	2.32557	0.9769	0.7439	-0.2329	0.01159
TC3	MNIST	1.60914	0.9727	0.0974	-0.8753	0.01252

Test case aggregation results for CIFAR10 datasets are attached to **APPENDIX M.**

C. Accountability: Contribution Evaluation Testing Results

Table 30: Contribution Evaluation Test Results

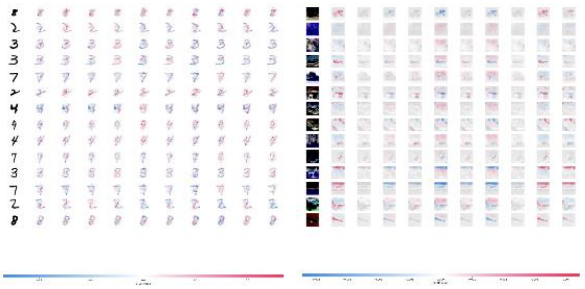
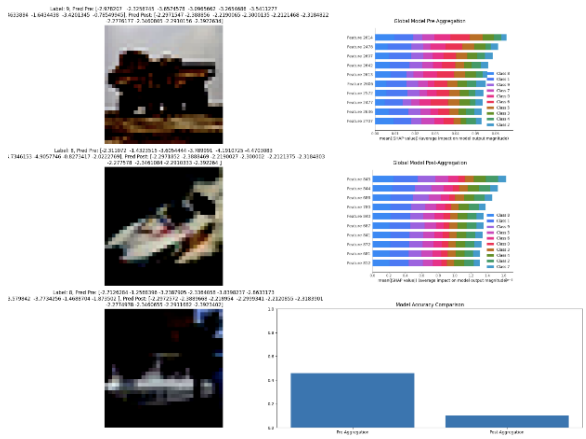
Test Case	Model	Contribution Evaluation Results			
		Number of clients	**Training session Pool money**	**Contribution Scores (Shapley Values)**	**Allocated Incentives**
TC1	MNIST	2	10000	Client 1: 0.033499 Client 2: 0.049100	Client 1: $4055.69 Client 2: $5944.31
TC2	MNIST	3	10000	Client 1: 0.065433	Client 1: $ 2890.16

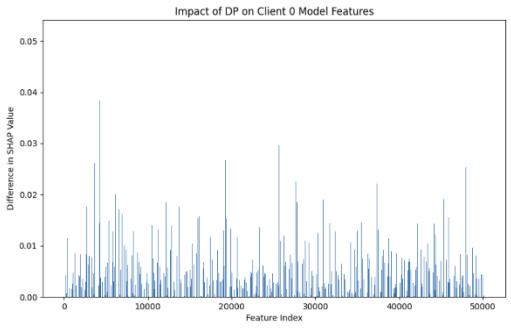
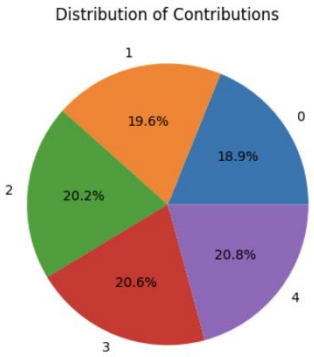
				Client 2: 0.077533 Client 3: 0.083433	Client 2: $ 3424.62 Client 3: $ 3685.22
TC3	MNIST	5	10000	Client 1: 0.169461 Client 2: 0.169461 Client 3: 0.174956 Client 4: 0.178115 Client 5: 0.179905	Client 1: $ 1888.09 Client 2: $ 1956.95 Client 3: $ 2020.51 Client 4: $ 2056.88 Client 5: $ 2077.57

Test case contribution evaluation results for CIFAR10 datasets are attached to **APPENDIX M.**

E. Explainability: XAI Mechanism Testing Results

Table 31: Explainable Mechanism Test Results

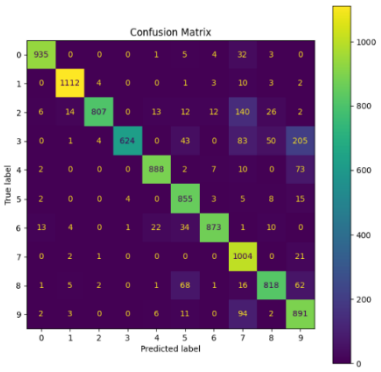
FL Workflow	Short Explanation of the Result	Generated Result
Client, Global Model Evaluation	A matrix of SV heatmaps, illustrating the feature importance for a set of images evaluated using the model.	
Secure Aggregation	The plot compares feature importance and model accuracy before and after aggregation, highlighting changes in model performance.	

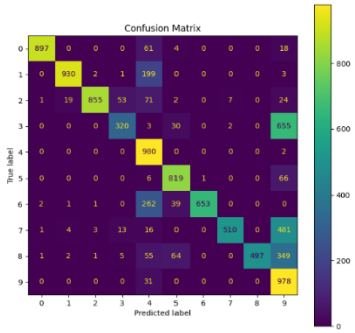
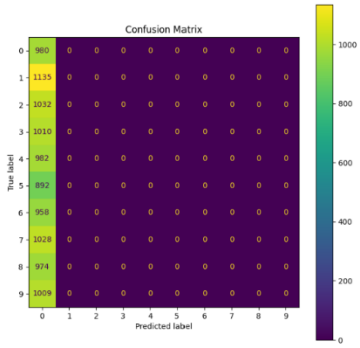
Differential Privacy	Variation in SV across client model features, illustrating the differential impact of privacy-preserving measures.	
Contribution Evaluation	Distribution of contributions among five clients using their SVs.	

8.2.2 Global Model Test Results

For every test case defined on the testing criteria the final global model evaluation has been given.

Table 32: Global model Results

Test case	Model	Global Model Test Results					
		Average loss	Accuracy	Precision	Recall	F1 Score	Confusion Matrix
TC1	MNIST	2.241	0.880	0.899	0.880	0.880	

TC2	MNIST	2.284	0.744	0.859	0.744	0.751	
TC3	MNIST	2.302	0.098	0.001	0.098	0.017	

Global model evaluation results for CIFAR10 dataset are attached to **Appendix M.**

8.5 Benchmark Discussion

For the benchmarking of this proposed project two approaches were conducted as identified from the chapter 2. First approach comparing the proposed system with baseline models were done during **model testing and evaluation**, while the second approach adhering the industry standards for Trustworthy AI guidelines has been performed with the use of comprehensive evaluation of the TAI principles inclusion in the project NotionFL. This process was a combined evaluation where both the author and selected experts have participated to provide a balanced and objective analysis. The evaluation encompassed an in-depth assessment of how NotionFL adheres to the selected TAI principles, ensuring a fair comparison and an extensive understanding of its alignment with industry standards and attached to **Appendix N.**

8.6 Functional Requirement Testing

To evaluate whether NotionFL met its intended functionalities, black-box testing was utilized. This approach aligned with the functional requirements detailed in **Chapter 4**. And the results of the functional requirement testing are attached on **Appendix N -11**.

8.7 Module Integration & Testing

The systems core FL functionalities can be categorized into 8 modules and their integration testing results have been outlined below.

Table 33: Module and Integration Testing

Module	Input Action	Expected Results	Actual Results	Status
Client Generation	Clients	Initialize clients with global model assigned	Initializes virtual workers for training	PASS
Dataset Loaders	Datasets	Splitting dataset according to the clients and load clients with data	Splits data for clients and load the data	PASS
Server Initializing	Server, Model	Initializing server with the ml model	Server initialized with the selection of ML model	PASS
Training module	Model configurations	Iterates over the clients for the given FL rounds	Starts training and iterates over the clients and FL rounds to train the model	PASS
Data collector	Model results and DB instance	Collects data from entire FL workflows and explains the calls	Gets initialized and collects training data and results	PASS
Secure Aggregation	Model Updates	Get the client model updates and aggregates to the global model	Performs secure aggregation using gradients and now.	PASS
Explainable mechanism	Models, model updates, model	Collects the data from the FL workflows and creates explanations and visualizations	Utilizes Shapley values and generates explanations and visualizations	PASS
Contribution evaluation	Model updates	Perform contributions evaluation and computes incentives	Calculates contribution evaluation and	PASS

			incentives using model updates and evaluations	

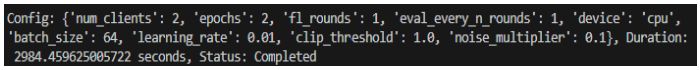
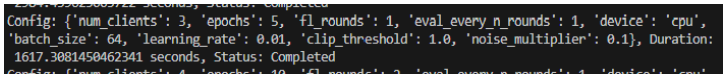
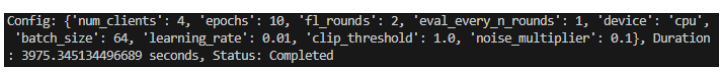
8.8 Non-Functional Requirement Testing

8.8.1 Performance Testing

i. Core FL Performance Testing

The proposed system's core FL workflow put to performance testing, to evaluate the system requirements and computational needs and resources. The MNIST dataset with three different training configurations used to monitor the amount of time it takes the system to complete the trainings within author's personal computer.

Table 34: Performance Testing Results

Configuration	Test Results
{'num_clients': 2, 'epochs': 2, 'fl_rounds': 1, 'eval_every_n_rounds': 1, 'device': 'cpu', 'batch_size': 64, 'learning_rate': 0.01, 'clip_threshold': 1.0, 'noise_multiplier': 0.1}	Duration: 2984.45964504722 seconds  Since, the number of clients, epoch and FL rounds are smaller made the entire duration lesser compared to other two configurations.
{'num_clients': 3, 'epochs': 5, 'fl_rounds': 1, 'eval_every_n_rounds': 1, 'device': 'cpu', 'batch_size': 64, 'learning_rate': 0.01, 'clip_threshold': 1.0, 'noise_multiplier': 0.1},	Duration: 1617.3081450462341 seconds  As epochs and clients have increased this time, it took lesser time than previous. This needs additional analysis to check the received results.
{'num_clients': 4, 'epochs': 10, 'fl_rounds': 2, 'eval_every_n_rounds': 1, 'device': 'cpu',	Duration: 3975.345134496689 seconds 

`'batch_size': 64,`
`'learning_rate': 0.01,`
`'clip_threshold': 1.0,`
`'noise_multiplier': 0.1}`
	This setting takes three time more duration to complete than the last time due to increment of FL rounds, number of clients and training epochs.

Testing revealed longer-than-expected training times likely due to using SV for explanation and visualizations. The limited resources (no dedicated GPU) of author's computer might have also impacted the performance and it has been highlighted below with the resource intensive plots.

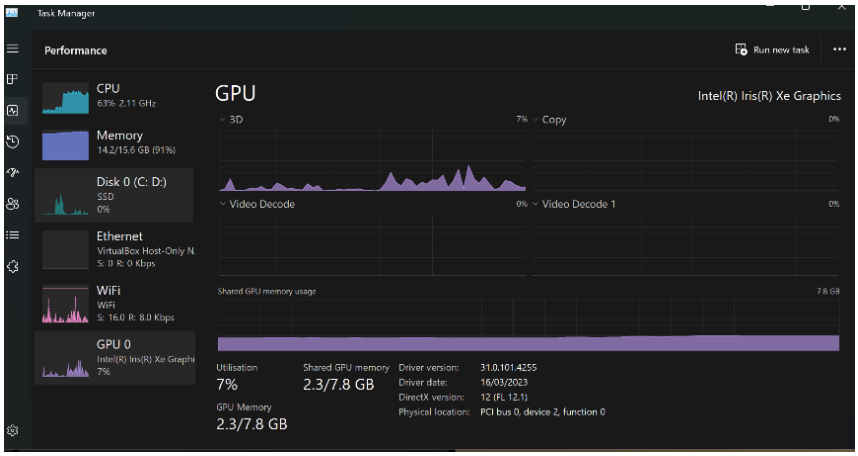


Figure 27: Resource Intensiveness Due to Local Training

8.8.2 Privacy & Security Testing

This non-functional testing, privacy and security were conducted to assess the differential privacy mechanism and secure aggregation mechanism which were implemented inside the proposed project. The results of the testing were presented in the **Appendix O.**

8.8.3 Accuracy Testing

This non-functional requirement test focused on the system's ability to results accurate global models. This aspect was successfully evaluated, as detailed in **Section 8.2.2.**

8.8.4 Usability Testing

To evaluate the application's user-friendliness, an external group of evaluators without the knowledge of the FL domain were contacted. The usability feedback and the results of the evaluator discussions are outlined in the **Chapter 9** section.

8.9 Limitation of Testing Process

The current evaluation of the proposed NotionFL application has limitations due to the **controlled nature** of the testing environment. Real-world data and scenarios were not incorporated, potentially affecting the generalizability of the results. Additionally, the **absence of a baseline method** hinders a direct comparison to establish NotionFL's effectiveness and performance. Furthermore, the **lack of statistically validated metrics** for the system's modules and FL workflows makes it challenging to objectively assess overall performance. **Limited resources** also played a role. The training process relied on the author's personal device, potentially restricting the system's performance capabilities. **Time constraints** further restricted testing, preventing benchmarks on datasets beyond MNIST and CIFAR-10.

8.10 Chapter Summary

This chapter delves into the various testing methodologies employed to comprehensively evaluate the proposed system. The objective was to meet the effective results in achieving the project's specific goals and contributing towards the research gap within the project.

CHAPTER 09: EVALUATION

9.1 Chapter Overview

This chapter dives into the evaluation process of the project NotionFL, beginning with a comprehensive self-evaluation by the author. Subsequently, it explores the selection of industry and domain experts for their valuable evaluations of the project in all aspects. Finally, chapter concludes with an examination of the limitations inherent in the evaluation process.

9.2 Evaluation Methodology and Approach

The study utilizes a mixed-method approach and undergoes evaluation employing both quantitative and qualitative methodologies. As previous chapter focused on quantitative assessments encompassing metrics such as accuracy, performance, and benchmarking, the primary objective of this chapter lies in the qualitative evaluation of the system. This approach draws insights from the domain and industry experts. These experts were given with comprehensive documentation, video presentation, and a live demonstration of the proposed system on an experimental setup, allowing thorough discussion and analysis of the project.

9.3 Evaluation Criteria

Table 35: Evaluation Criteria

Criteria	Objectives
Problem background, and problem novelty	To determine if the problem is serious enough for a solution to bring value.
Research scope and complexity	To guarantee that the study is sufficiently broad, deep, and complex for a bachelor's level research.
Design and development decisions of the proposed solution	To analyze the theoretical and technical choices made during the development of the proposed solution design, as well as to establish the solution's distinctiveness.
Proposed Trustworthy cross-silo FL architecture	To assess the proposed Trustworthy FL architecture and its workflows such as secure aggregation, privacy preservation, contribution evaluation, and incentive mechanism.

Proposed Explainable mechanism	To assess the proposed explainable mechanism and its core function to interpret the FL workflows.
Limitations and future work	To identify potential areas for improvement that may be addressed in future attempts.

9.4 Self-Evaluation

The complete self-evaluation results have been added to the **Appendix P** section.

9.5 Selection of the Evaluators

The selection of the evaluator for the NotionFL system are shown in the **Appendix Q** section.

9.6 Evaluation Result

9.6.1 Qualitative Evaluation Result Analysis

The expert feedback was analyzed to identify codes that meets the evaluation criteria, and a thematic analysis was conducted and presented on **Appendix R**. The Supporting evidence also provided in the **Appendix S.**

Table 36: Themes identified by conducting thematic analysis.

Themes	Codes
Research gap and problem	Tackled Problem, Strategical and critical, current limitations, influence/impact
Novelty of the research	Unique, in-depth
Scope, complexity, & depth	Undergraduate, complex, Broder, challenge, overcome
Architecture	Federated Learning, resources, explainability, security
Results	Performance, computational resource, effectiveness
Suggestions	More results, research & review paper

9.6.2 Quantitative Evaluation Result Analysis

Building upon the quantitative evaluation of the UI/UX, author below delves into a qualitative analysis of the feedback provided by the experts regarding the NotionFL system.

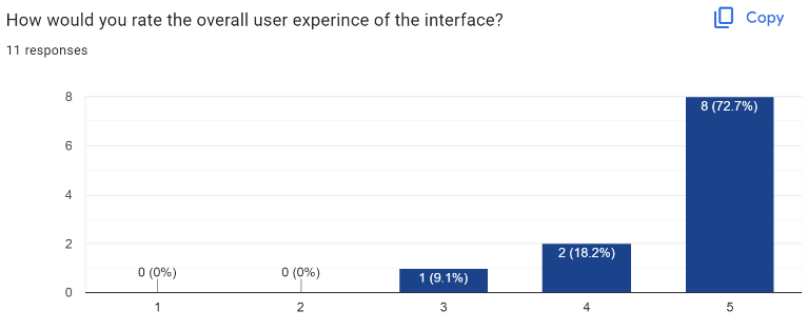


Figure 28: Quantitative Analysis of UI/UX of Project NotionFL

9.7 Limitations of Evaluation

The evaluators for this project were primarily academics, not industry professionals, which is notable since FL is more common in research than in industry. This difference in context means the evaluators might not be fully versed in the real-world applications of cross-silo FL, especially regarding the need for trust and explainability. Also, the technical aspects of the system, like XAI, SV, and DP, might exceed some evaluators' expertise. A major challenge in evaluation was demonstrating the entire training process due to its duration, which can be at least 20 minutes even for less complex sessions, constrained by computational resources. To address this, the training sessions were pre-run, and results were presented to evaluators with evidence from actual data.

9.8 Evaluation of Functional Requirements

The Evaluation of Functional Requirements have added under the **Appendix T.**

9.9 Evaluation of Non-Functional Requirements

IM – Implemented | **NIM** – Not Implemented

The Evaluation of Non-Functional Requirement have added under the **Appendix U**.

9.10 Chapter Summary

This chapter provides an evaluation of research methodologies utilized and their underlying rationales. The assessments during the evaluation process were conducted utilizing predetermined criteria, and the evaluators' opinions were thematically examined and well presented. Furthermore, the evaluation process also covered an assessment of functional and non-functional requirements which were developed during the software requirement specification process.

CHAPTER 10: CONCLUSION

10.1 Chapter Overview

This chapter intends to present the project deviation that took place from the initial project proposal stage, along with the current initial test results of the prototype and the future improvements that planned to achieve before the final prototype exhibition. The chapter concludes with the video demonstration of the current progress of the project with the project implementation code link to the GitHub repository.

10.2 Achievements of Research Aims & Objectives

10.2.1 Achievements of Aim

The aim of this project is to research, design, develop, and evaluate a solution that employs trustworthiness and interpretability in cross-silo Federated Learning environments by introducing a novel trustworthy architecture and a trusted mediator to deliver visually interpretable explanations amongst the competing nature of clients.

The research successfully accomplished the aim of implementing a novel trustworthy architecture along with the explainable mechanism that that fosters trust within the cross-silo FL environments while interpreting the FL processes. The testing and evaluation chapter provides necessary evidence to support the effectiveness of the proposed solution.

10.2.1 Achievements of Objectives

Table 37:Achievements of Research Objectives

Research Objective	Learning Outcomes	Status
Problem Domain: Conduct a thorough study of the relevant research topics.	**LO1, LO4, LO5**	Completed
Literature Review: In-depth review and analysis of current literature and research on the problem domain.	**LO1, LO4, LO5**	Completed
Requirements Analysis: Gather user requirements and analyze them critically.	**LO3, LO6, LO7**	Completed

Design: Designing a comprehensive architecture for the proposed FL system.	**LO2, LO5, LO7**	Completed
Development: Development of the proposed FL architecture based on the designs.	**LO1, LO5, LO6, LO7**	Completed
Testing & Evaluation: Performing testing and evaluation of the FL system to validate the results.	**LO5, LO6, LO7, LO8**	Completed

10.3 Utilization of Knowledge from the Degree Program

Table 38: Utilization of Knowledge of Degree Program

Modules	Utilized knowledge
Software development I, II & Object-Oriented Programming	These modules introduced the basic programming skills, designing of UML along with the software designing and testing to the author to get started with application development.
Advanced-client-side development, Web Design & Development, Database System, Client-server architecture	The knowledge gained from these modules assisted author in designing, developing, and testing of full stack / client-server architecture application with database connectivity of the proposed system
Software Development Group Project	The knowledge gained from this module helped author to significantly progress with writing the thesis document and it also guided the process to complete the research project successfully.

10.4 Use of Existing Skills

- **Industry experience as Software Engineer**: The author gained significant experience during his internship at IFS R&D Internationals and used those skills to complete the project within the timeline.
- **Application development & data science**: The author started learning new concepts and theories while managing to complete small personal projects before starting of the final

year to brush up the understanding and gaining new skills using platforms like YouTube and LinkedIn Learning.

10.5 Use of New Skills

- **Federated Learning**: Author was inexperienced with the chosen problem domain and had to study from scratch by reading research papers and other academic resources to conduct the research and improve the current state of cross-silo FL setting.
- **Data Protection Laws and Regulations**: The author gained a fair understanding on the data protection laws and regulations under AI systems and specifically learned Trustworthy AI regulations to get comfortable to design the novel FL architecture.
- **Explainable AI**: The author was unfamiliar with the explainable ai domain and considerable amount of time and effort was needed to get the understanding to successfully implement the explainable mechanism within the FL environment.
- **LaTeX:** The author learnt LaTeX scripting language to prepare the research papers and presented a survey paper using this language and overleaf online platform.

10.6 Achievement of Learning Outcomes

Table 39: Achievement of Learning Outcomes

What has been Learned	LOs
The process of research project has made the author familiar with the importance of following research methodologies and improving decision-making abilities by examining the collected data.	**LO1, LO2, LO4**
The author has conducted a comprehensive literature review on the FL domain and acquired knowledge on many FL related technologies.	**LO4, LO5**
The author gained knowledge of the formal techniques used for data collection, all while considering the significance of social, legal, and ethical factors.	**LO3, LO6, LO4**
Documenting the research at every stage helped the author to improve their academic writing abilities.	**LO8**

In the research's prototyping phase, experimentation was conducted through trial and error. The author developed their problem-solving abilities by identifying problems, finding solutions, and achieving their desired results.	**LO5, LO7**

10.7 Problems and Challenges Faced

Table 40: Achievement of Learning Outcomes

Problems/Challenges	Solution
Project Scope	The author was initially drawn to the broad scope of federated learning, particularly the challenges of trustworthiness architectures and lack of explainability. However, to achieve a focused and achievable research project, the author narrowed the investigation to the subdomain of cross-silo FL with the trustworthy architecture and explainability.
Learning Curve	Studying several aspects of the field, particularly ones that involve cross-silo architecture and integration of explainable ai was a challenging task due to the lack of regulation and absence of any up-to-date standards known to the author. Moreover, developing a system as proposed is a considerable difficulty as the internet was filled with numerous articles containing outdated code.
Trial and Error	The study is centered on a process of experimentation and refinement, requiring the author to carry out numerous rounds of training to enable the system to generate outcomes.
Resource constraints	The author had to opt for lengthier training due to the imposed time constraints for GPU training on Google Collaboratory. In addition, the author had to operate the system locally on their laptop, which did not have sufficient capacity to carry out training sessions smoothly.

10.8 Deviations

The author has proposed not to support the incentive mechanism during the initial project proposal since the system already involved in other main FL workflows. However, incentive mechanisms being a crucial part of cross-silo FL settings made author to include this requirement in the project. Further, there won’t be any FL framework used and instead author developed the

FL system from scratch using the PyTorch library simulating a novel cross-silo FL setting. Due to architecture complexity the application was not hosted online, instead it was connected with a real-time cloud database. Finally, the application was built as a web application with limited features to fulfill the University of Westminster requirement for the final year project.

10.9 Limitations of the Research

Major limitations of the research are highlighted below.

- The proposed trustworthy architecture was not evaluated properly due to the lack of proper mechanism that can quantity trust measures in FL systems.
- Only a limited number of explanations and visualization were created during the training process to explain the server process on FL workflows.
- An assumption was made that the server administrators are legit since they are the ones who manages the entire FL training process and clients.
- The results of this project heavily rely on an experimental setup, which means that real-world outcomes may be lacking.
- The application heavily relies on computational resources and has a worst time complexity O(n!) due to the integration of SV and XAI methods for explanation and visualization.
- The current development environment inherently limits the system's ability to fully guarantee privacy preservation. This outlines the absence of robust security protocols typically implemented in production deployments.

10.10 Future Enhancements

- Potential direction for future research is to create a trust measuring mechanism for cross-silo FL system to measure actual metrics in the systems.
- Another enhancement can be making the system more agnostic with other ML libraries, model, and dataset instead of only agnostic for PyTorch library-based model and datasets.
- Explore other explainable techniques like transformers, and Rule Fit algorithms to compare with the SV and reduce the time complexity.
- Investigating secure multi-party computation for secure aggregation integration while making the privacy preservation more robust than the gradient descents.

- Exploring how does blockchain and smart contracts can be utilized for enhanced accountability and trust between the clients and the server administrators.
- Frontend application needs work on the static pages and backend application needs more computational power to handle the FL workflows. This can be leveraged by deploying application on high-performance cloud-based servers.

10.11 Achievement of the Contribution to Body of knowledge

10.11.1 Problem and Research Domain Contribution

The author has contributed significantly to both the research and problem domain in the field of cross-silo FL by introducing two new additions.

- **Novel Trustworthy Cross-silo FL Architecture**: The author introduced a novel architecture in the domain of cross-silo as an enhancement to the current workflow, which will help to foster trust and explainability while broadening the use of cross-silo FL in industrial settings.
- **Explainable FL mechanism**: This novel approach explains and visualizes the server decisions (FL workflows) and processes while ensuring fair and unbiased systems. This paves the way for research into more robust, transparent cross-disciplinary FL systems.

10.11.2 Technical Contribution

- **NotionFL**: The mediator like system which developed to exhibit the proposed TFL architecture, and the explainable mechanism is considered as a field of software engineering.
- **ExplainableFL**: A Python package for Explainable FL using Shapley Values.

10.12 Concluding Remarks

The author has proposed a novel trustworthy cross-silo architecture featuring an explainable mechanism designed to foster explainability and trust between the clients and servers within cross-silo FL. Addressing the inherent ‘competing nature’ of clients in cross-silo domain, the research architecture involves in incorporating various Trustworthy AI principles to adhere with trustworthy AI regulations. Out of the principles that have been chosen, the author seeks for industry level technologies to implement these within the FL workflows. The resulting study was

very well-received, with positive feedback from all quarters. The explanations and insights provided within the workflows were noteworthy and are expected to be introducing an innovative and state-of-the-art flow in cross-silo FL settings. Further, this project has set the stage for future exploration in this the field of trust and explainability. Finally, completing this project was the author's most challenging and rewarding learning experience and potentially this work will be made open-sourced to ensure wider research.

REFERENCES

A Survey on Securing Federated Learning: Analysis of Applications, Attacks, Challenges, and Trends. (no date). Available from https://ieeexplore.ieee.org/document/10107622?denied= [Accessed 6 November 2023].

A Survey on Securing Federated Learning: Analysis of Applications, Attacks, Challenges, and Trends | IEEE Journals & Magazine | IEEE Xplore. (no date). Available from https://ieeexplore.ieee.org/document/10107622?denied= [Accessed 6 November 2023].

Adadi, A. and Berrada, M. (2018). Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access*, 6, 52138–52160. Available from https://doi.org/10.1109/ACCESS.2018.2870052.

Aledhari, M. et al. (2020). Federated Learning: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Access*, 8, 140699–140725. Available from https://doi.org/10.1109/ACCESS.2020.3013541.

Asad, M., Moustafa, A. and It, T. (no date). Federated Learning Versus Classical Machine Learning: A Convergence Compariso.

Bao, X. et al. (2019a). FLChain: A Blockchain for Auditable Federated Learning with Trust and Incentive. *2019 5th International Conference on Big Data Computing and Communications (BIGCOM)*. August 2019. QingDao, China: IEEE, 151–159. Available from https://doi.org/10.1109/BIGCOM.2019.00030 [Accessed 11 December 2023].

Bao, X. et al. (2019b). FLChain: A Blockchain for Auditable Federated Learning with Trust and Incentive. *2019 5th International Conference on Big Data Computing and Communications (BIGCOM)*. August 2019. 151–159. Available from https://doi.org/10.1109/BIGCOM.2019.00030 [Accessed 11 December 2023].

Bárcena, J.L.C. et al. (2022). Fed-XAI: Federated Learning of Explainable Artificial Intelligence Models. 2022. Available from https://www.semanticscholar.org/paper/Fed-XAI%3A-Federated-Learning-of-Explainable-Models-B%C3%A1rcena-Daole/205587a6ffe4c0a993e930a2d47b81a6c2d58d45 [Accessed 28 September 2023].

Barredo Arrieta, A. et al. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58, 82–115. Available from https://doi.org/10.1016/j.inffus.2019.12.012.

Bashir, A.K. et al. (2023). A Survey on Federated Learning for the Healthcare Metaverse: Concepts, Applications, Challenges, and Future Directions. Available from https://doi.org/10.48550/ARXIV.2304.00524 [Accessed 23 August 2023].

Ben Saad, S., Brik, B. and Ksentini, A. (2022). A Trust and Explainable Federated Deep Learning Framework in Zero Touch B5G Networks. *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. December 2022. 1037–1042. Available from https://doi.org/10.1109/GLOBECOM48099.2022.10001371 [Accessed 27 September 2023].

Beutel, D.J. et al. (2020). Flower: A Friendly Federated Learning Research Framework. 28 July 2020. Available from https://www.semanticscholar.org/paper/Flower%3A-A-Friendly-Federated-Learning-Research-Beutel-Topal/04bc6eb6bbbedd83b094a989c821393a26682bf5 [Accessed 15 December 2023].

Bhagoji, A. et al. (2018). Analyzing Federated Learning through an Adversarial Lens. 29 November 2018. Available from https://www.semanticscholar.org/paper/Analyzing-Federated-Learning-through-an-Adversarial-Bhagoji-Chakraborty/6c66108edb9af0533309055e7b2ecb8922db03d8 [Accessed 4 December 2023].

Bonawitz, K. et al. (2019). Towards Federated Learning at Scale: System Design. *ArXiv*. Available from https://www.semanticscholar.org/paper/Towards-Federated-Learning-at-Scale%3A-System-Design-Bonawitz-Eichner/79cf9462a583e1889781868cbf8c31e43b36dd2f [Accessed 15 December 2023].

Chatila, R. et al. (2021). Trustworthy AI. In: Braunschweig, B. and Ghallab, M. (eds.). *Reflections on Artificial Intelligence for Humanity*. Lecture Notes in Computer Science. Cham: Springer International Publishing, 13–39. Available from https://doi.org/10.1007/978-3-030-69128-8_2 [Accessed 10 November 2023].

Chen, P. et al. (2022). EVFL: An explainable vertical federated learning for data-oriented Artificial Intelligence systems. *Journal of Systems Architecture*, 126, 102474. Available from https://doi.org/10.1016/j.sysarc.2022.102474.

Corcuera Bárcena, J.L. et al. (2022). An Approach to Federated Learning of Explainable Fuzzy Regression Models. *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. July 2022. 1–8. Available from https://doi.org/10.1109/FUZZ-IEEE55066.2022.9882881 [Accessed 29 September 2023].

Corcuera Bárcena, J.L. et al. (2023). Enabling federated learning of explainable AI models within beyond-5G/6G networks. *Computer Communications*, 210, 356–375. Available from https://doi.org/10.1016/j.comcom.2023.07.039.

Dwork, C. (2006). Differential Privacy. In: Bugliesi, M. Preneel, B. Sassone, V. et al. (eds.). *Automata, Languages and Programming*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 1–12. Available from https://doi.org/10.1007/11787006_1 [Accessed 22 December 2023].

Ethics guidelines for trustworthy AI | Shaping Europe's digital future. (2019a). Available from https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai [Accessed 10 November 2023].

Fan, Z. et al. (2022). Fair and efficient contribution valuation for vertical federated learning. *ArXiv*. Available from https://www.semanticscholar.org/paper/2f11e998e9ac38eca2a55a6e9b2dfa74e35a551e [Accessed 17 August 2023].

FedAI.org. (2020). WeBank and Swiss Re signed Cooperation MoU. *FedAI.org*. Available from https://www.fedai.org/news/webank-and-swiss-re-signed-cooperation-mou/ [Accessed 5 November 2023].

Fiosina, J. (2021). Explainable Federated Learning for Taxi Travel Time Prediction: *Proceedings of the 7th International Conference on Vehicle Technology and Intelligent Transport Systems*. 2021. Online Streaming, --- Select a Country ---: SCITEPRESS - Science and Technology Publications, 670–677. Available from https://doi.org/10.5220/0010485606700677 [Accessed 4 November 2023].

Geyer, R.C., Klein, T. and Nabi, M. (2017). Differentially Private Federated Learning: A Client Level Perspective. *ArXiv*. Available from https://www.semanticscholar.org/paper/Differentially-Private-Federated-Learning%3A-A-Client-Geyer-Klein/b1e538dbf538fd9fdf5f5870c5b7416ae08c9882 [Accessed 17 December 2023].

Guo, Y. et al. (2023). Seeing is believing: Towards interactive visual exploration of data privacy in federated learning. *Information Processing & Management*, 60 (2), 103162. Available from https://doi.org/10.1016/j.ipm.2022.103162.

Haffar, R., Sánchez, D. and Domingo-Ferrer, J. (2023). Explaining predictions and attacks in federated learning via random forests. *Applied Intelligence*, 53 (1), 169–185. Available from https://doi.org/10.1007/s10489-022-03435-1.

Hsu, C.-F. et al. (2022). FedTrust: Towards Building Secure Robust and Trustworthy Moderators for Federated Learning. *2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (MIPR)*. August 2022. 318–323. Available from https://doi.org/10.1109/MIPR54900.2022.00063.

Huang, C., Huang, J. and Liu, X. (2022). Cross-Silo Federated Learning: Challenges and Opportunities. Available from http://arxiv.org/abs/2206.12949 [Accessed 15 August 2023].

Huang, J. et al. (2020). An Exploratory Analysis on Users' Contributions in Federated Learning. *2020 Second IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, 20–29. Available from https://doi.org/10.1109/TPS-ISA50397.2020.00014.

Huong, T.T. et al. (2022). Federated Learning-Based Explainable Anomaly Detection for Industrial Control Systems. *IEEE Access*, 10, 53854–53872. Available from https://doi.org/10.1109/ACCESS.2022.3173288.

Jia, B. et al. (2022). Blockchain-Enabled Federated Learning Data Protection Aggregation Scheme With Differential Privacy and Homomorphic Encryption in IIoT. *IEEE Transactions on Industrial Informatics*, 18 (6), 4049–4058. Available from https://doi.org/10.1109/TII.2021.3085960.

Kairouz, P. et al. (2021). Advances and Open Problems in Federated Learning. *Foundations and Trends® in Machine Learning*, 14 (1–2), 1–210. Available from https://doi.org/10.1561/2200000083.

kb:google_docs_citations_unlinked [Zotero Documentation]. (no date). Available from https://www.zotero.org/support/kb/google_docs_citations_unlinked [Accessed 15 February 2024].

Kholod, I. et al. (2020). Open-Source Federated Learning Frameworks for IoT: A Comparative Review and Analysis. *Sensors*, 21 (1), 167. Available from https://doi.org/10.3390/s21010167.

Kusiak, A. (2023). Federated explainable artificial intelligence (fXAI): a digital manufacturing perspective. *International Journal of Production Research*, 0 (0), 1–12. Available from https://doi.org/10.1080/00207543.2023.2238083.

Kusiak, A. (2023). Federated explainable artificial intelligence (fXAI): a digital manufacturing perspective. *International Journal of Production Research*, 0 (0), 1–12. Available from https://doi.org/10.1080/00207543.2023.2238083.

Li, A. et al. (2023). Towards Interpretable Federated Learning. Available from https://doi.org/10.48550/ARXIV.2302.13473 [Accessed 17 August 2023].

Li, Q., He, B. and Song, D. (2021). Practical One-Shot Federated Learning for Cross-Silo Setting. Available from http://arxiv.org/abs/2010.01017 [Accessed 8 September 2023].

Li, Q. et al. (2022). Inspecting the Running Process of Horizontal Federated Learning via Visual Analytics. *IEEE Transactions on Visualization and Computer Graphics*, 28 (12), 4085–4100. Available from https://doi.org/10.1109/TVCG.2021.3074010.

Li, Q. et al. (2023). A Survey on Federated Learning Systems: Vision, Hype and Reality for Data Privacy and Protection. *IEEE Transactions on Knowledge and Data Engineering*, 35 (4), 3347–3366. Available from https://doi.org/10.1109/TKDE.2021.3124599.

Li, Y. et al. (2021). A Blockchain-Based Decentralized Federated Learning Framework with Committee Consensus. *IEEE Network*, 35 (1), 234–241. Available from https://doi.org/10.1109/MNET.011.2000263.

Liu, Y. et al. (2021). FATE: An Industrial Grade Platform for Collaborative Learning With Data Protection. *J. Mach. Learn. Res.* Available from https://www.semanticscholar.org/paper/FATE%3A-An-Industrial-Grade-Platform-for-Learning-Liu-Fan/3d73e21af71bde8dc7984bd72f7077fb691b2523 [Accessed 15 December 2023].

Liu, Z. et al. (2022). Contribution-Aware Federated Learning for Smart Healthcare. *Proceedings of the AAAI Conference on Artificial Intelligence*. 28 June 2022. 12396–12404. Available from https://doi.org/10.1609/aaai.v36i11.21505 [Accessed 18 December 2023].

Lo, S.K. et al. (2021). Blockchain-based Trustworthy Federated Learning Architecture. *ArXiv*. Available from https://www.semanticscholar.org/paper/ad52dadd52f614339607b56661ecb64a807e875e [Accessed 30 October 2023].

Lo, S.K. et al. (2023). Toward Trustworthy AI: Blockchain-Based Architecture Design for Accountability and Fairness of Federated Learning Systems. *IEEE Internet of Things Journal*, 10 (4), 3276–3284. Available from https://doi.org/10.1109/JIOT.2022.3144450.

Ludwig, H. et al. (2020). IBM Federated Learning: an Enterprise Framework White Paper V0.1. *ArXiv*. Available from https://www.semanticscholar.org/paper/IBM-Federated-Learning%3A-an-Enterprise-Framework-Ludwig-Baracaldo/6e9fb743a27d4d471d14b578f84cd0c57e9d3e55 [Accessed 15 December 2023].

Lundberg, S.M. and Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. 22 May 2017. Available from https://www.semanticscholar.org/paper/A-Unified-Approach-to-Interpreting-Model-Lundberg-Lee/442e10a3c6640ded9408622005e3c2a8906ce4c2 [Accessed 16 December 2023].

Mahawaga Arachchige, P.C. et al. (2020). Local Differential Privacy for Deep Learning. *IEEE Internet of Things Journal*, 7 (7), 5827–5842. Available from https://doi.org/10.1109/JIOT.2019.2952146.

McMahan, H.B. et al. (2016). Communication-Efficient Learning of Deep Networks from Decentralized Data. Available from http://arxiv.org/abs/1602.05629 [Accessed 6 November 2023].

mnist · Datasets at Hugging Face. (2022). Available from https://huggingface.co/datasets/mnist [Accessed 28 January 2024].

Mothukuri, V. et al. (2021). A survey on security and privacy of federated learning. *Future Generation Computer Systems*, 115, 619–640. Available from https://doi.org/10.1016/j.future.2020.10.007.

Mugunthan, V., Rahman, R. and Kagal, L. (2020). BlockFLow: An Accountable and Privacy-Preserving Solution for Federated Learning. *ArXiv*. Available from https://www.semanticscholar.org/paper/BlockFLow%3A-An-Accountable-and-Privacy-Preserving-Mugunthan-Rahman/77e26b6a326d83f3871673082cface18a2a1d554 [Accessed 22 December 2023].

Neto, H.N.C. et al. (2023). A Survey on Securing Federated Learning: Analysis of Applications, Attacks, Challenges, and Trends. *IEEE Access*, 11, 41928–41953. Available from https://doi.org/10.1109/ACCESS.2023.3269980.

Papers with Code - Gradient-based learning applied to document recognition. (no date). Available from https://paperswithcode.com/paper/gradient-based-learning-applied-to-document [Accessed 28 January 2024].

Papers with Code - MNIST Dataset. (no date). Available from https://paperswithcode.com/dataset/mnist [Accessed 28 January 2024].

Pillutla, K., Kakade, S.M. and Harchaoui, Z. (2022). Robust Aggregation for Federated Learning. *IEEE Transactions on Signal Processing*, 70, 1142–1154. Available from https://doi.org/10.1109/TSP.2022.3153135.

Radley-Gardner, O., Beale, H. and Zimmermann, R. (eds.). (2016). *Fundamental Texts On European Private Law*. Hart Publishing. Available from https://doi.org/10.5040/9781782258674 [Accessed 7 November 2023].

Rahman, M.A. et al. (2020). Secure and Provenance Enhanced Internet of Health Things Framework: A Blockchain Managed Federated Learning Approach. *IEEE Access*, 8, 205071–205087. Available from https://doi.org/10.1109/ACCESS.2020.3037474.

Ratnayake, H., Chen, L. and Ding, X. (2023). A review of federated learning: taxonomy, privacy and future directions. *Journal of Intelligent Information Systems*. Available from https://doi.org/10.1007/s10844-023-00797-x [Accessed 21 September 2023].

Regulation: Regulation (EU) 2016/679 of the European... - Google Scholar. (no date). Available from https://scholar.google.com/scholar_lookup?title=Regulation%20%28EU%29%202016%2F679%20of%20the%20European%20Parliament%20and%20of%20the%20Council%20of%2027%20april%202016%20on%20the%20protection%20of%20natural%20persons%20with%20regard%20to%20the%20processing%20of%20personal%20data%20and%20on%20the%20free%20movement%20of%20such%20data%2C%20and%20repealing%20Directive%2095%2F46&journal=Official%20J%20Eur%20Union%20%28OJ%29&volume=59&issue=1-88&publication_year=2016&author=Regulation%2CGDP [Accessed 7 November 2023].

Rehman, M.H. ur et al. (2021). TrustFed: A Framework for Fair and Trustworthy Cross-Device Federated Learning in IIoT. *IEEE Transactions on Industrial Informatics*, 17 (12), 8485–8494. Available from https://doi.org/10.1109/TII.2021.3075706.

Renda, A. et al. (2022). Federated Learning of Explainable AI Models in 6G Systems: Towards Secure and Automated Vehicle Networking. *Information*, 13 (8), 395. Available from https://doi.org/10.3390/info13080395.

Ribeiro, M.T., Singh, S. and Guestrin, C. (2016). 'Why Should I Trust You?': Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. Available from https://doi.org/10.1145/2939672.2939778.

Roth, H.R. et al. (2022). NVIDIA FLARE: Federated Learning from Simulation to Real-World. Available from https://doi.org/10.48550/ARXIV.2210.13291 [Accessed 17 December 2023].

Ryffel, T. et al. (2018). A generic framework for privacy preserving deep learning. *ArXiv*. Available from https://www.semanticscholar.org/paper/76c6d39edecdb943ce0f68f5a44e6608db96e383 [Accessed 15 December 2023].

Samek, W., Wiegand, T. and Müller, K. (2017). Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. *ArXiv*. Available from https://www.semanticscholar.org/paper/58e0ca33ae3068fee7005f339bf6c444fc17d55f [Accessed 10 November 2023].

S'anchez, P.M.S. et al. (2023). FederatedTrust: A Solution for Trustworthy Federated Learning. *ArXiv*. Available from https://doi.org/10.48550/arXiv.2302.09844 [Accessed 18 August 2023].

Sathya, S.S. et al. (2018). A Review of Homomorphic Encryption Libraries for Secure Computation. *ArXiv*. Available from https://www.semanticscholar.org/paper/A-Review-of-Homomorphic-Encryption-Libraries-for-Sathya-Vepakomma/27b7f8655f4fb365154f4fb9865a3f797c913eaa [Accessed 18 December 2023].

Saunders, M., Lewis, P. and Thornhill, A. (2006). Research Methods for Business Students. 26 October 2006. Available from https://www.semanticscholar.org/paper/Research-Methods-for-Business-Students-Saunders-Lewis/bef028d7d7fcb4705c24451304b089f4912920d4 [Accessed 28 August 2023].

Tan, A.Z. et al. (2022). Towards Personalized Federated Learning. *IEEE Transactions on Neural Networks and Learning Systems*, 1–17. Available from https://doi.org/10.1109/TNNLS.2022.3160699.

Tang, M. and Wong, V.W.S. (2021). An Incentive Mechanism for Cross-Silo Federated Learning: A Public Goods Perspective. *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*. May 2021. 1–10. Available from https://doi.org/10.1109/INFOCOM42981.2021.9488705.

TensorFlow Federated. (no date). Available from https://www.tensorflow.org/federated [Accessed 15 December 2023].

Ungersböck, M. et al. (2023a). Explainable Federated Learning: A Lifecycle Dashboard for Industrial Settings. *IEEE Pervasive Computing*, 22 (1), 19–28. Available from https://doi.org/10.1109/MPRV.2022.3229166.

Usage. (2023). Available from https://github.com/FederatedAI/FATE-Board [Accessed 16 December 2023].

Wang, G. (2019). Interpret Federated Learning with Shapley Values. *ArXiv*. Available from https://www.semanticscholar.org/paper/34aa8d1271e0ee3526b3ebd78def20d36390e0ef [Accessed 4 November 2023].

Wei, X. et al. (2019b). Multi-Agent Visualization for Explaining Federated Learning. *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. August 2019. Macao, China: International Joint Conferences on Artificial Intelligence Organization, 6572–6574. Available from https://doi.org/10.24963/ijcai.2019/960 [Accessed 10 November 2023].

Xu, R. et al. (2022). DeTrust-FL: Privacy-Preserving Federated Learning in Decentralized Trust Setting. *2022 IEEE 15th International Conference on Cloud Computing (CLOUD)*, 417–426. Available from https://doi.org/10.1109/CLOUD55607.2022.00065.

Xu, X. and Lyu, L. (2020). A Reputation Mechanism Is All You Need: Collaborative Fairness and Adversarial Robustness in Federated Learning. 20 November 2020. Available from https://www.semanticscholar.org/paper/A-Reputation-Mechanism-Is-All-You-Need%3A-Fairness-in-Xu-Lyu/329734fdbb35faab89e14eb9b105a665d7a5f079 [Accessed 18 December 2023].

Yang, Q. et al. (2019). Federated Machine Learning: Concept and Applications. *ACM Transactions on Intelligent Systems and Technology*, 10 (2), 1–19. Available from https://doi.org/10.1145/3298981.

Yang, Z. et al. (2022). Trustworthy Federated Learning via Blockchain. *IEEE Internet of Things Journal*. Available from https://doi.org/10.1109/JIOT.2022.3201117 [Accessed 18 August 2023].

Yin, X., Zhu, Y. and Hu, J. (2022a). A Comprehensive Survey of Privacy-preserving Federated Learning: A Taxonomy, Review, and Future Directions. *ACM Computing Surveys*, 54 (6), 1–36. Available from https://doi.org/10.1145/3460427.

Zeng, R. et al. (2021). A Comprehensive Survey of Incentive Mechanism for Federated Learning. *ArXiv*. Available from https://www.semanticscholar.org/paper/A-Comprehensive-Survey-of-Incentive-Mechanism-for-Zeng-Zeng/46fa2ef11417b75250945f98a538742de8ea99c4 [Accessed 17 December 2023].

Zhan, Y. et al. (2022). A Survey of Incentive Mechanism Design for Federated Learning. *IEEE Transactions on Emerging Topics in Computing*, 10 (2), 1035–1044. Available from https://doi.org/10.1109/TETC.2021.3063517.

Zhang, Y. et al. (2023a). A Survey of Trustworthy Federated Learning with Perspectives on Security, Robustness and Privacy. *Companion Proceedings of the ACM Web Conference 2023*, 1167–1176. Available from https://doi.org/10.1145/3543873.3587681.

Zhou, J. et al. (2022). A Differentially Private Federated Learning Model against Poisoning Attacks in Edge Computing. *IEEE Transactions on Dependable and Secure Computing*, 1–1. Available from https://doi.org/10.1109/TDSC.2022.3168556.

APPENDIX

APPENDIX A: Concept Map

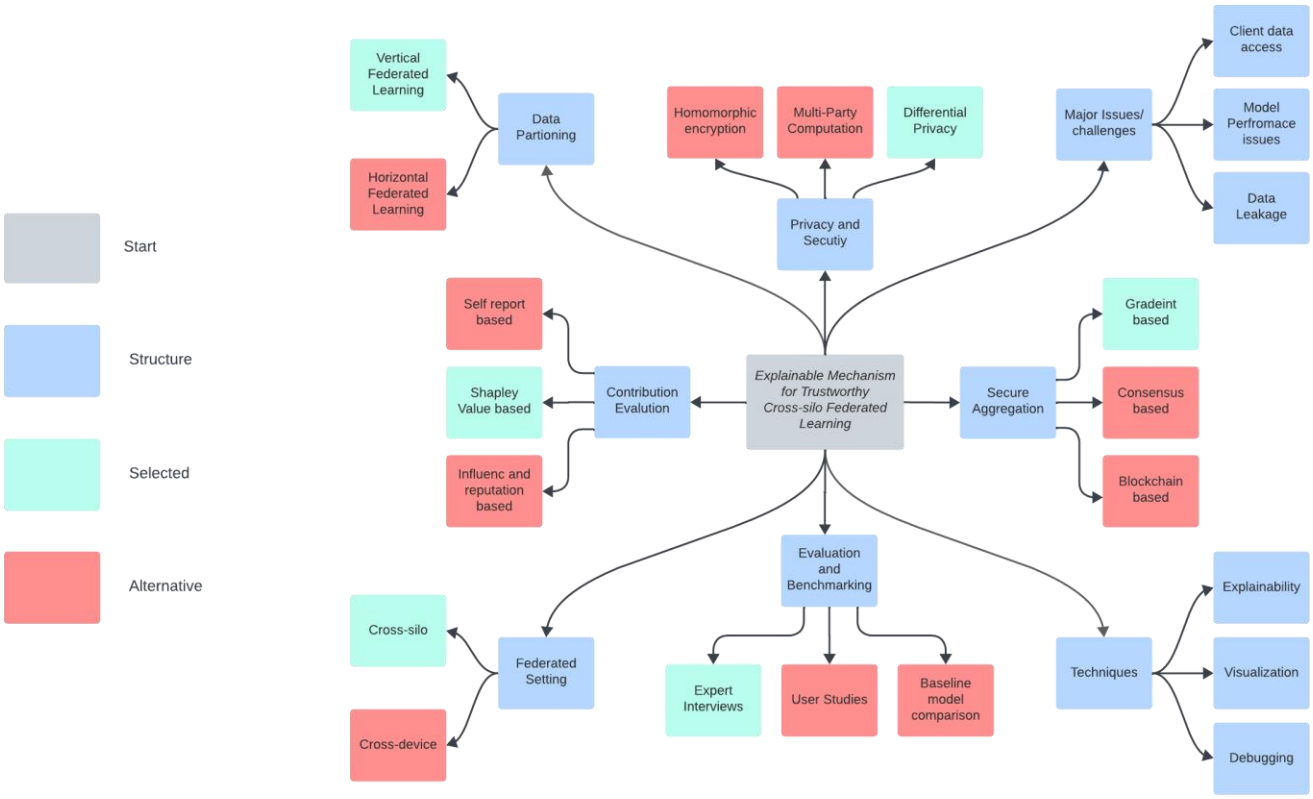


Figure 29: concept map

APPENDIX B: Summary of Existing Trustworthy FL Architecture

Table 41: Summary of Existing Trustworthy FL Architectures

Citation	(Tariq et al., 2023)
Summary	Proposes a novel trustworthy Federated Learning architecture using trustworthy AI principles as the main pillars. This architecture was based on explainability, fairness and privacy representing different levels of trust in FL workflows. Further study also presented a comprehensive survey on existing trust assessment works in FL to foster future directions on Trustworthy FL.
Improvements	First research to encompass all dimensions of trustworthy FL aspects. Provided a well-structured survey on Trustworthy FL exploring key components and existing solutions.
Limitations	No practical implementation or simulation has been done yet on creating or testing the architecture.
Citation	**(S'anchez et al., 2023)**
Summary	Paper presents a novel taxonomy on creating an algorithm to compute trustworthiness of the FL models using the federatedScope framework. Further it also examines existing work to evaluate the proposed scenario.
Improvements	FederatedTrust: Algorithm to compute trustworthiness in FL models.
Limitations	Overall system lacking trust and explainability. Lacking multi-objective optimization on FL trust.
Citation	**(Yang et al., 2022)**
Summary	Proposes a decentralized blockchain based trustworthy FL architecture.

	The architecture uses secure aggregation methods to resist malicious devices and supports latency AI services under consensus protocols.
Improvements	Deployed a practical Byzantine fault tolerance consensus protocol with high effectiveness and low energy consumption.
Limitations	It was focused to cross-device FL setting
Citation	**(Zhang et al., 2023)**
Summary	Study presents a comprehensive road map like architecture for building trustworthy FL systems. IT also provides a survey reviewing recent improvements in TFL regarding security, robustness and privacy aspects.
Improvements	Utilizes three trustworthy AI aspects to create the architecture. Provides a novel survey with existing advances and future trends in TFL.
Limitations	Consists of less adaptive aggregation and model performance degenerations issues. Lacks for a standard benchmark to verify the trust and robust ability of the architecture.
Citation	**(Lo et al., 2021)**
Summary	Proposes a blockchain based trustworthy FL architecture using smart contracts to enable accountability and fairness in covid-19 X-ray detection use case.
Improvements	Study applied two trustworthy AI aspects: accountability and fairness. Enabled accountability using smart contract-based data model registry. Introduced a weighted data sampler algorithm to enhance the fairness of the architecture.

Limitations	Smart contracts make the system more complex and blockchain creates new connectivity and latency issues.
Citation	**(Rehman et al., 2021)**
Summary	Presents a trust enabled blockchain based cross-device FL framework named TrustFed, to detect outliers in the FL training process before the aggregation process.
Improvements	Utilizes blockchain with ethereum and smart contracts to create a novel protocol to detect the outliers.
Limitations	Study focused only in cross-device FL setting. Creates additional complexity due to the blockchain use case.

APPENDIX C: Summary of Existing Explainable FL Works

Table 42: Table: Summary of Existing Explainable FL Works

Citation	**(Ben Saad, Brik and Ksentini, 2022)**
Summary	Proposes a novel XAI-powered framework to explain FL based operation decisions by building a deep learning FL model to predict key performance indicators of network slices. Framework leverages multiple XAI approaches like Rulefit, LIME, SHAPE and PDP to explain and interpret a latency prediction FL model.
Improvements	Predicts positive and negative impact of the features of the FL model. Framework presented both local and global model prediction explanations.
Limitations	It only explains the features of both global and local models.
Citation	**(Huong et al., 2022)**
Summary	Proposes a XAI based anomaly detection architecture called FedeX to be used in Industrial control systems with liquid storage dataset infrastructure.
Improvements	Shapley Additive XAI models were used to explain the detected anomalies in the proposed system. Multimodal optimization using FedVAE-SVDD
Limitations	It only focuses on the edge devices and cross-device FL setting. Evaluation of the system was only done in terms of performance and detection capability.
Citation	**(Wang, 2019)**

Summary	Study presents interpretable FL models to explain the models using Shapley Values while preserving data privacy in vertical FL settings.
Improvements	Shapley values help to reveal detailed feature importance of the host and guest features in separate views. Unified feature importance techniques give a holistic contribution of each feature and helps to identify the contribution of each guest from their data.
Limitations	Only based on the Vertical FL setting. Explanations are only for the feature importance of the FL model.
Citation	**(Chen et al., 2022)**
Summary	Proposes an explainable vertical federated learning (EVFL) framework, including the credibility assessment strategy, the federated counterfactual explanation, and the importance rate (IR) metric.
Improvements	Implements a credibility assessment strategy for choosing uncertain queries. Provided a federated counterfactual explanation (CFCE) for Vertica FL models. Designed an importance rate (IR) metric method for evaluating the feature importance.
Limitations	Limitation of data hindered the evaluation of the proposed system. Focused only for Vertical FL architecture.
Citation	**(Guo et al., 2023)**
Summary	Study proposes an interactive visualization system for privacy

	interpretation on FL systems for clients who participated in the system.
Improvements	Presents privacy preservation explanations to help FL data owners in data privacy and security. System provides client-oriented visualizations to interpret privacy concerns.
Limitations	Focused on Horizontal FL architectures. Explanations are only based on privacy preservation. Trade-off between privacy protection and inference accuracy.
Citation	**(Ungersböck et al., 2023)**
Summary	Proposes a life cycle dashboard visualizing and explaining server information from FL systems. The system presents generic and applicable explanations for all types of users in different industries.
Improvements	Demonstrates explanations on Horizontal FL workflows using visualizations. System adopts an industrial setting and hopes to get utilized in real world use cases.
Limitations	Focused to edge devices - cross-device setting. System having limited interaction options for the clients. Lack of finding underperforming clients and contribution evaluation.
Citation	**(Li et al., 2022)**
Summary	Proposed a novel FL visualizing tool using HFL privacy-preserving protocol and designed an exploratory visual analytics system for the Horizontal FL process called HFLens.
Improvements	System supports visual interpretation at the overview, communication rounds, and client instance levels on the FL process. Visualizations include features to support fine-grained analysis on Horizontal FL workflows.

Limitations	Focused on HFL setting and applicable to cross-device settings. Lack of privacy preservation and differentiable visualizations.

APPENDIX D: GANNT Chart

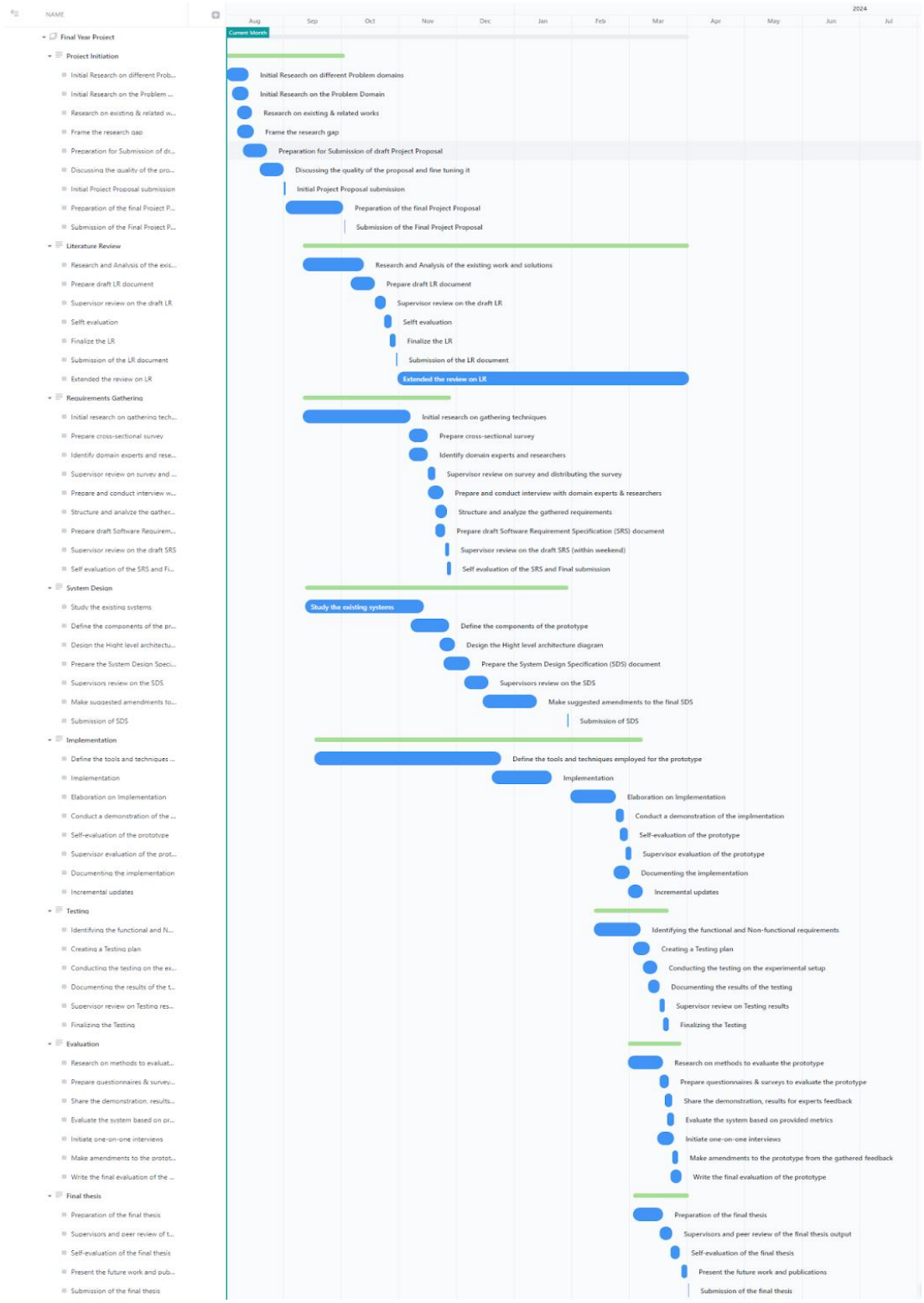


Figure 30: Gannt Chart

APPENDIX E: Interview Questionnaire

I Interview Questions and Reasoning

Table 43: Interview Questions

Questions	Reasoning	RQ
Research Gap / Aim / Objectives		
To begin our discussion, I'd like to hear your perspective on the importance of trustworthiness and explainability in Federated Learning systems, especially with GDPR, Ethical AI and Trustworthy AI regulations. How do these regulations influence the design and implementation of such systems in your experience?	To validate the research gap from experts' point of view	RQ3
Do you think that making federated learning systems more understandable and trustworthy will lead to better cooperation among clients, especially in competitive settings like a group of organizations in cross-silo Federated Learning?	To understand the key factors influencing client cooperation for better cross-silo FL systems	RQ1
Considering data sensitivity in Federated Learning, how can we manage data privacy and data access while performing secure data aggregation, contribution assessment and data visualizations?	To validate the importance of the data privacy and identify how to securely access the client data for evaluation of the system	RQ2
Do you believe the new ML paradigm Explainable AI (XAI) platform is suitable for enhancing clarity in Federated Learning systems without compromising privacy and security? If yes, how can XAI be effectively utilized?	To validate the technological selections and understanding the applicability on the domain	RQ4

What sort of technical difficulties do you think we might face when trying to make AI systems more understandable, and are there any design principles or best practices to consider when creating a secure and transparent Federated Learning system that works across different organizations?	To identify technical challenges and feasibility considerations for the proposed system	RQ3, RQ4
Beyond differential privacy, homomorphic encryption, and Multi party computation what other strategies could enhance privacy and security in a Federated Learning system like mine enhancing trust and interpretability?	To Tackle technical difficulties on data privacy and security and find new solutions to the proposed system	RQ3, RQ4
In AI systems, we often face a trade-off between balancing transparency and model performance due to two contradicting values. How do you suggest we ensure a Federated Learning system is both easy to understand and performs well, without one affecting the other?	To Tackle technical difficulties and find new novel solution to the proposed system	RQ3, RQ4
Designing / User Interface / View		
On a system design point of view, How can we design user interfaces to effectively explain FL workflows and model results while protecting privacy? Also, could you share some UI/UX design best practices for making complex machine learning processes accessible to a varied audience?	To understand effective UI/UX workflows regarding complex AI systems like FL and get introduced to new design practices.	RQ5

Based on your experience, do you think showing federated learning processes in real-time helps users understand them better? Also, what key design features would you focus on to make a system easy to use and accessible for everyone?	To validate the explainable visualization part of the proposed system and identify new design considerations in user interface.	RQ5
What essential features should the user interface of my Federated Learning system have to meet the needs of various users like data scientists, organizations, and non-technical users? Also, regarding the server's role in FL training, what key workflows or lifecycle methods should be clearly included in the system's design?	To find the requirements of the proposed visualizations interface of the system.	RQ4, RQ5
Considering the abstract nature of Federated Learning (FL), which real-world domain (healthcare, transportation, finance) do you think would be most effective for simulating and demonstrating the capabilities of my proposed FL system? and why? [Personally, I thought of taking healthcare industry]	To find a suitable use case to test the proposed system in a more real-world setting.	RQ1, RQ5
What evaluation and benchmarking methods would you recommend for assessing the explainability and trustworthiness aspects of my proposed FL system? could you suggest any specific metrics or standards that are crucial for measuring the success of the system in these areas?	To find relevant evaluation and benchmarking techniques and ideas for the proposed system	**RQ2, RQ3**
As an expert in this field, what final piece of advice or insight would you offer for someone undertaking a project like mine in Federated Learning? Additionally, upon completion of my prototype, would you be open	To receive overall feedback and insights on the current proposed ide to implement the system in coming future.	

to provide your valuable feedback on how it's been implemented and how well it effectively meets the requirements?		

APPENDIX F: Brainstorming Analysis

Table 44: Analysis of the brainstorming findings

Criteria	Findings
Privacy	The interpretability of an AI system always brings privacy issues along with it, likewise federated learning in general is vulnerable to data leaking, theft and variations of attacks like, poisoning attack, backdoor attack, and differential privacy attacks. Therefore, the proposed system should always maintain its primary goal of data privacy.
Efficiency	As FL already has certain issues and vulnerabilities related to its performance and privacy, the author realized that the prototype of the project should not affect the overall performance and results of the system which is to ultimately train a better global model.
Uniqueness	After successful literature review and surveying, the author found out that there aren't any FL frameworks focusing on vertical data partitioning supporting organizations in cross-silo FL setting, with trustworthy architecture and visual interpretations. This allows the author to work on the implementation building a proof concept as a framework or a toolkit narrowing the research gap and helping to grow this domain.
Fairness	The proposed FL system should compute fair contribution evaluations and incentive percentages for the clients without any biases to maintain trust among other clients while ensuring trustworthiness among the system. Since

	fairness is a part of trustworthy AI principles this component is a must to include in the proposed system.
Robustness	The implementation of the prototype should be flexible for industries that could potentially utilize this system in their FL architectures, since this system is supposed to be agnostic to other datasets, ML models and use cases.

APPENDIX G: Use Case Description

Table 45: Use case description for receiving data/logs and metrics.

Use Case Name	Receiving Data/Logs and Metrics	ID 09
Description	The data collector mechanism receives or collects data/metrics/logs from the FL workflows.	
Participating actors	Client (Organizations)	
Preconditions	Completing Secure Aggregation, Contribution Evaluation, Privacy Preservation and other FL workflows.	
Extended use cases	Create Explanations/Visualizations	
Included use cases	None	
Main flow	Await till the main FL workflows finish their functionalities then evoke data collection functions. Receives metrics/data/logs from the FL workflows (contribution evaluation, secure aggregation, privacy preservation etc. Stores the collected data to generate and create explanations and visualizations.	
Alternative flow	None	
Exceptional Flows	Unable to receive data/metrics or logs from FL workflows due to communication problems, runtime errors and function vulnerabilities. Unable to perform the data collection due to unsupported data, results, and logs. Unable to store the collected data due to different results and data format.	
Post conditions	Store data, metrics, and logs	

APPENDIX H: Functional Requirements

Table 46: Functional Requirements

FR ID	Requirement	Priority Level	Use case
FR1	Organizational clients have to be assigned to a virtual worker.	M	UC1
FR2	Virtual Workers (organizations/clients) should be able to perform local training using their raw data.	M	UC4
FR3	The system must perform secure aggregation using the model updates provided by the virtual workers after the local training.	M	UC7
FR4	The system should be able to evaluate client's model updates to calculate the contribution evaluation measure.	M	UC8
FR5	The system must have the security and privacy mechanisms running behind while the training phase is ongoing.	M	UC5
FR6	The data collector mechanism should collect the logs/metrics and data related to the FL workflows.	M	UC9
FR7	Explainable mechanisms should generate interpretable visualizations and explanations using the metrics collected by the data collector mechanism.	M	UC10
FR8	The system administrators should be able to initiate the FL training.	M	UC2
FR9	The clients and the system administrators should be able to view their correspondent user interfaces with relevant visualization related to the training phase.	M	UC12, UC13
FR10	The system should provide functionality for debugging during the FL process through the corresponding user dashboards.	S	UC14

FR11	The system must have the ability for training initiators to customize the FL training.	S	UC1, UC2
FR12	The system users are able to view the evaluation of the global models.	S	UC11, UC12, UC13
FR13	The system should be able to give an option to the users to download specific logs and metrics for additional analysis without compromising the privacy violation.	W	UC12, UC13
FR14	The training initiators or server administrators could be able to download the global models.	W	UC12, UC13
FR15	Distributing incentives based on the contribution measure of each organization.	C	UC12

APPENDIX I: Non-Functional Requirements

Table 47: Non-Functional Requirements

NFR ID	Requirements	Description	Priority
NFR1	Performance	The system should not suffer from performance issues compared to general cross-silo Federated learning systems.	M
NFR2	Privacy and Security	The system must have robust security measures to protect the sensitive data during the transmission and processing, adhering to the industry standard encryption techniques and protocols.	M
NFR3	Accuracy	The system must achieve its main objectives accurately without any compromission.	S
NFR4	Interoperability	The system must be compatible with various data formats and machine learning models to support a wide range of use cases.	S
NFR5	Scalability	The system should have to be scalable to accommodate the increasing number of virtual workers without the degradation in performance by deploying it to cloud.	C
NFR6	Usability	The user interface of the proposed system should manage visualizing all the necessary FL tasks with friendly user experience without complicating the interface.	C

APPENDIX J: Component Diagram

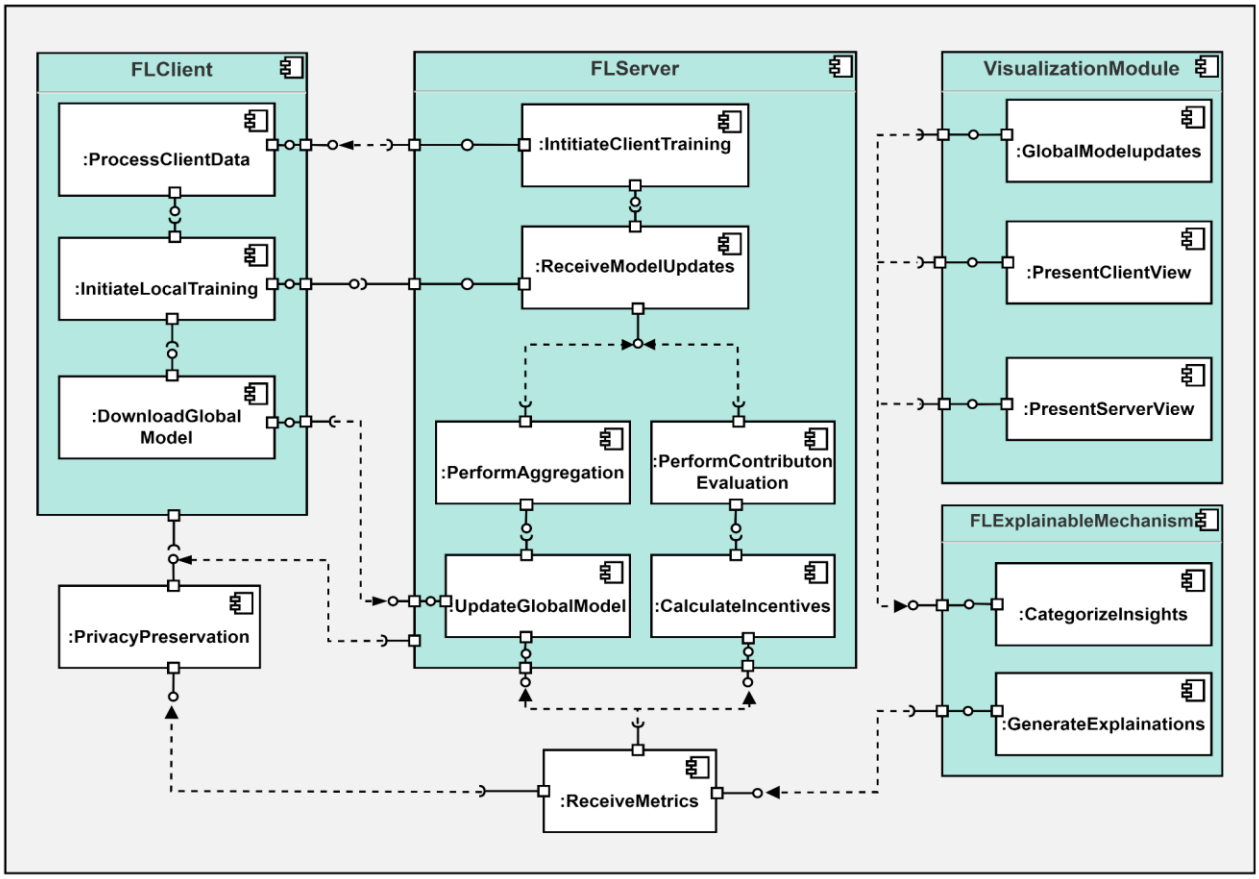


Figure 31: component Diagram

APPENDIX K: Dataflow Diagram

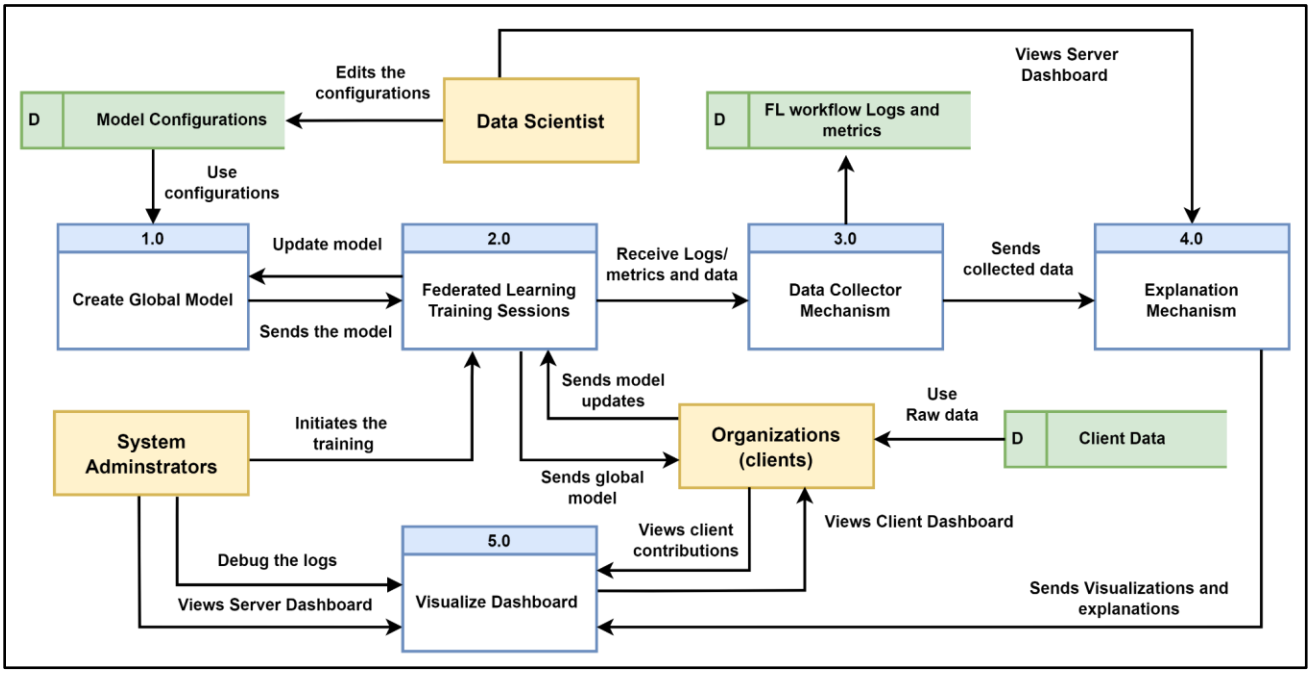


Figure 32: DFD Diagram

APPENDIX L: User Interfaces

i. Landing Page

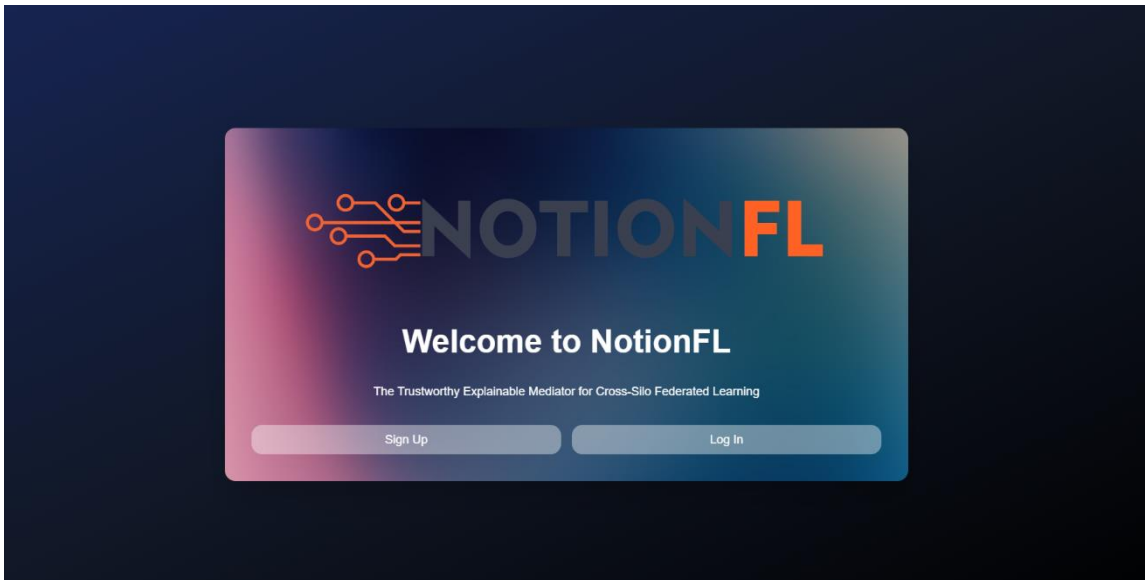


Figure 33: Landing Page

ii. User Signup Page

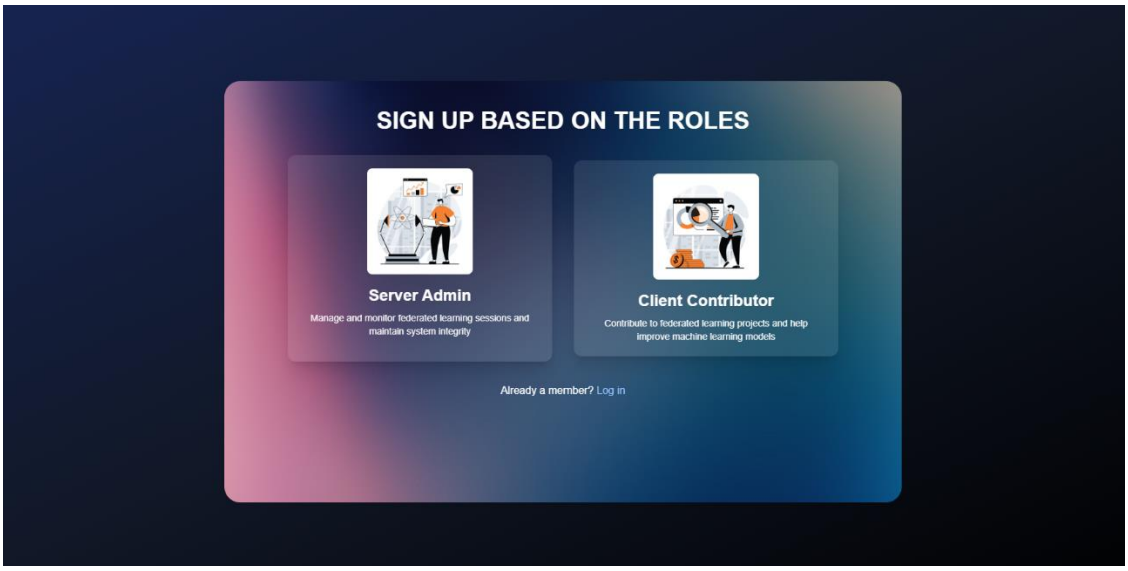


Figure 34: User SignUp Page

iii. Privacy Explanation: Server Page

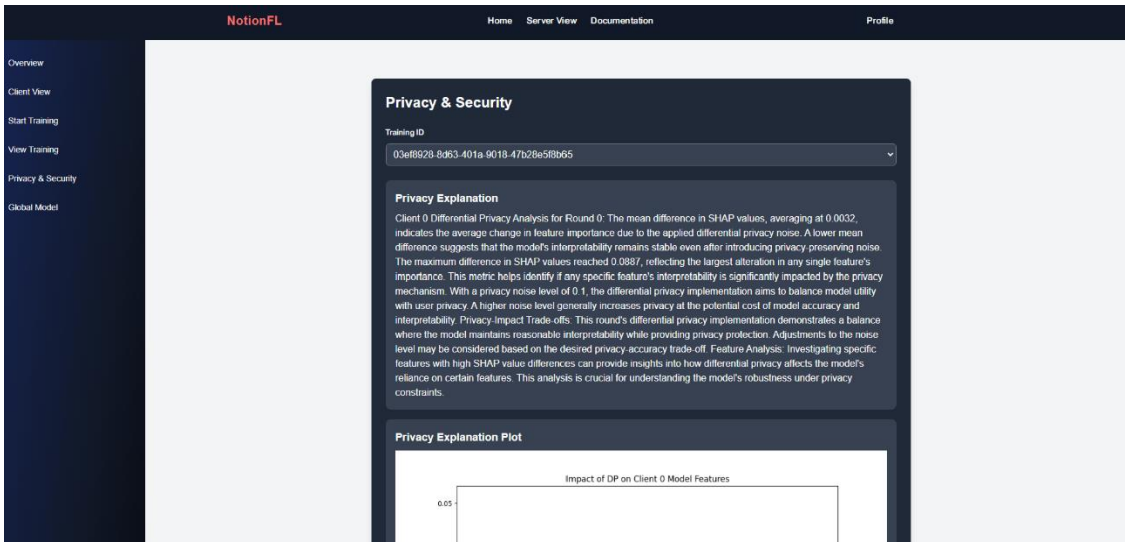


Figure 35: Privacy Explanation Page

iv. Global Model Evaluation: Server Page

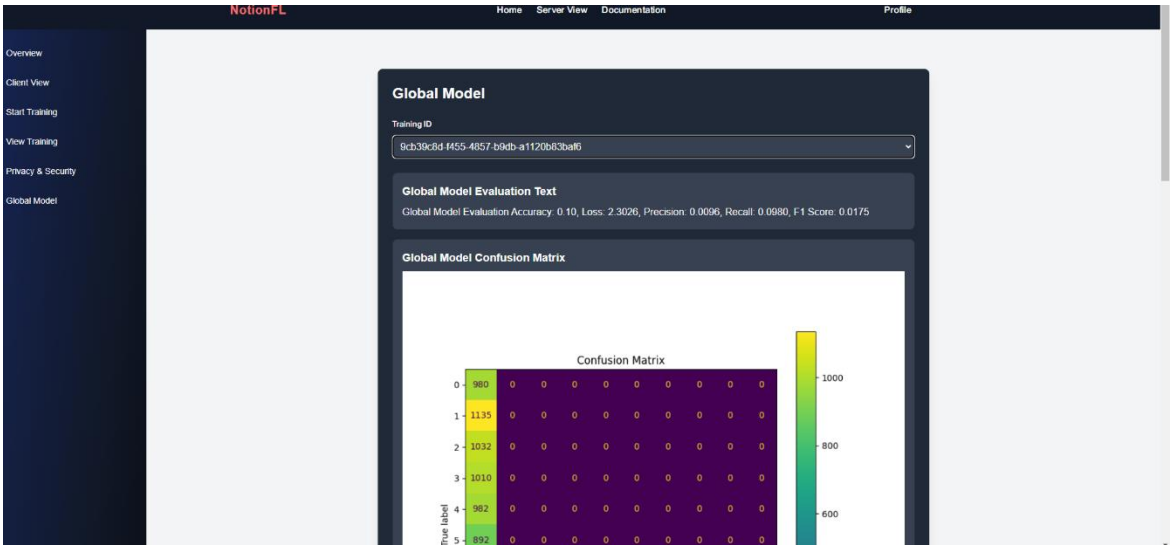


Figure 36: Global model evaluation Page

v. Model Evaluation Page: Client

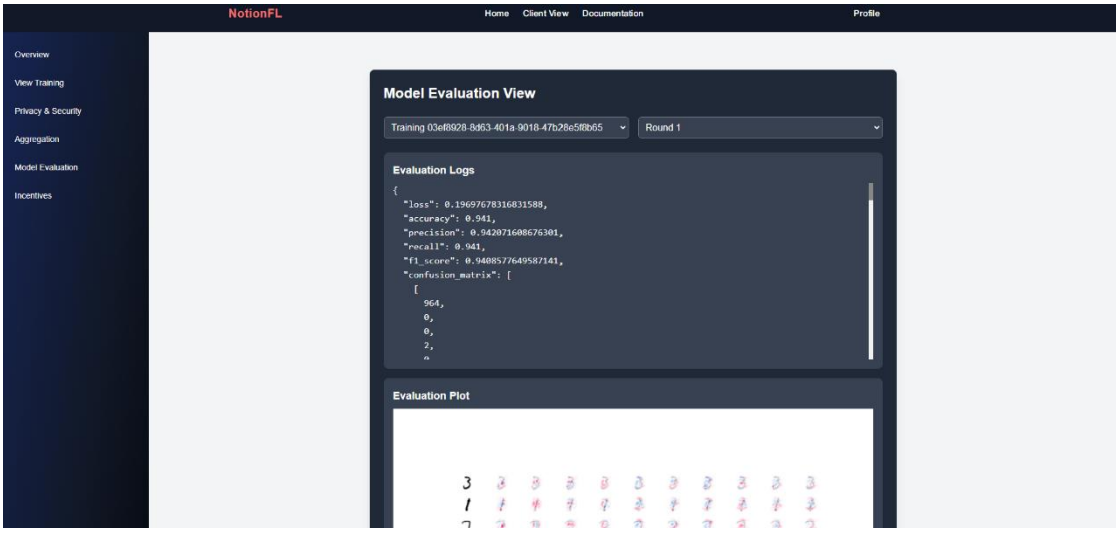


Figure 37:Client Model Evaluation Page

vi. Client Contribution Evaluation Page

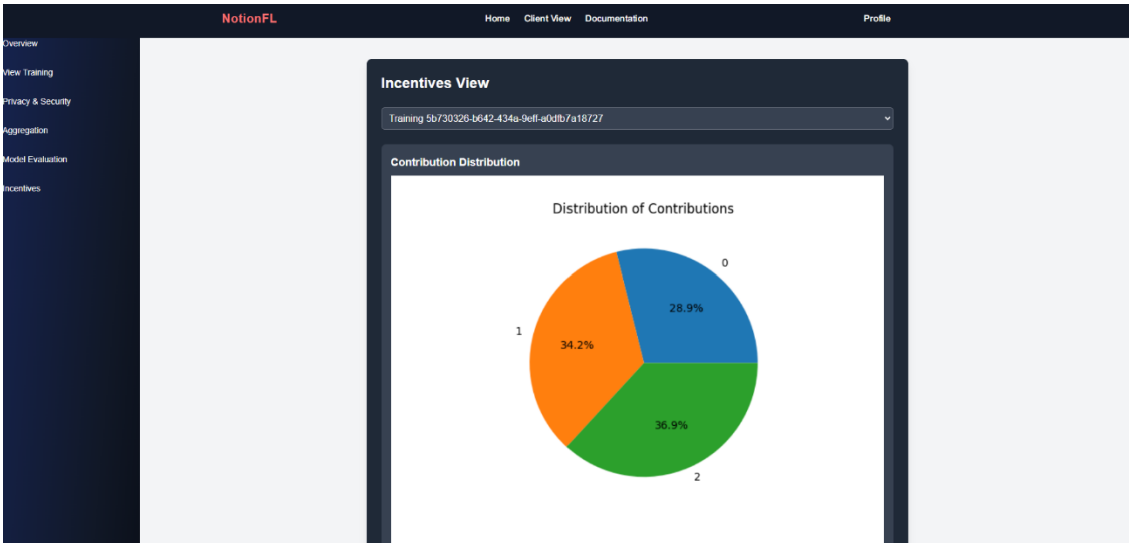


Figure 38:Client Contribution Page

vii. Secure Aggregation Page

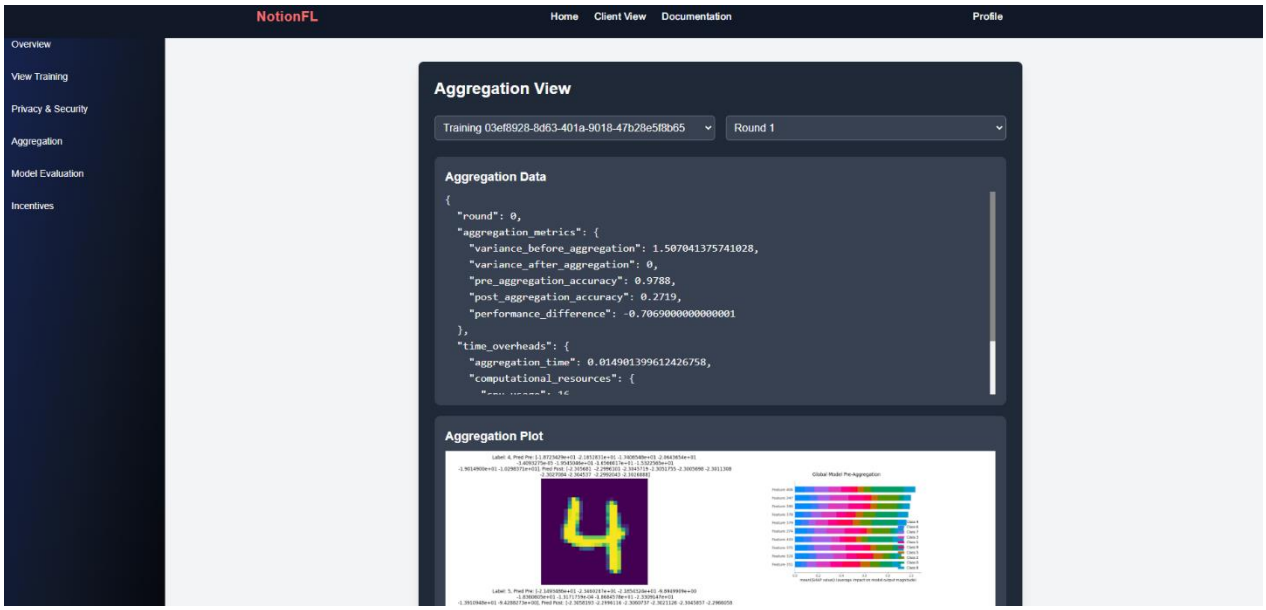


Figure 39: Secure Aggregation explanation page

APPENDIX M: Test cases for Model Integration Testing

A. Privacy: Differential Privacy Testing Results

For every testing session, DP was evaluated using on how its impacting the global model performance since it applies certain levels of noise on the models every time.

Table 48: Differential Privacy Test Results

Criteria				Results		
Test case	**Model**	**Clip threshold**	**Noise multiplier**	**Global model Accuracy**	**Computational overhead**	**DP Impact Level on Global Model Performance**
TC1	CIFAR10	1.0	0.1	0.1100	0.01255 sec	Negative
TC2	CIFAR10	1.5	0.2	0.1000	0.04075 sec	Negative
TC3	CIFAR10	2.0	0.3	0.5271	0.01051 sec	Negative

B. Robustness: Secure Aggregation Testing Results

Table 49:Secure Aggregation Test Results

Test case	Model	Aggregation Test Results				
		Variance before aggregation	**Pre-aggregation accuracy**	**Post-aggregation accuracy**	**Performance difference**	**Aggregation time (sec)**
TC1	CIFAR10	0.69050	0.3470	0.1000	-0.2469	0.03165
TC2	CIFAR10	1.19510	0.4427	0.1000	-0.3427	0.07541
TC3	CIFAR10	1.60914	0.9727	0.0974	-0.8753	0.01252

C. Accountability: Contribution Evaluation Testing Results

Table 50: Contribution Evaluation Test Results

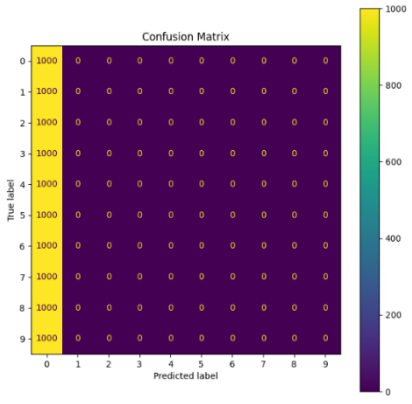
	Model	Contribution Evaluation Results

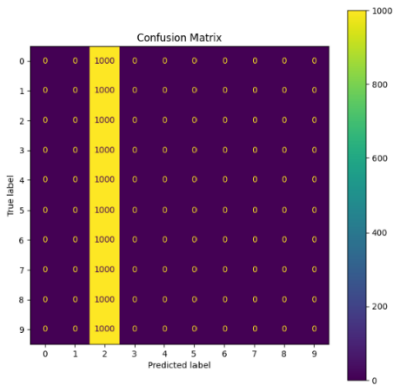
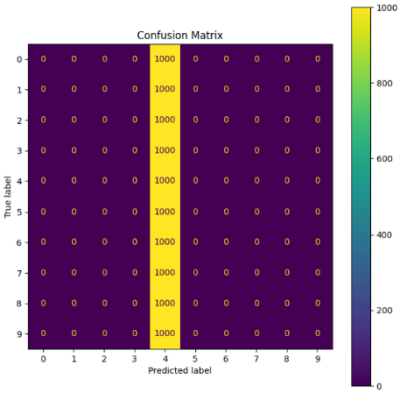
Test Case		Number of clients	Training session Pool money	Contribution Scores (Shapley Values)	Allocated Incentives
TC1	CIFAR10	2	10000	Client 1: 0.030600 Client 2: 0.166400	Client 1: $1553.30 Client 2: $8446.70
TC2	CIFAR10	3	10000	Client 1: 0.028716 Client 2: 0.098116 Client 3: 0.015786	Client 1: $ 1008.66 Client 2: $ 3446.32 Client 3: $ 5545.02
TC3	CIFAR10	5	10000	Client 1: 0.169461 Client 2: 0.169461 Client 3: 0.174956 Client 4: 0.178115 Client 5: 0.179905	Client 1: $ 1888.09 Client 2: $ 1956.95 Client 3: $ 2020.51 Client 4: $ 2056.88 Client 5: $ 2077.57

Global Model Test Results

For every test case defined on the testing criteria the final global model evaluation has been given.

Table 51: Global Model Test Results

Test case	Model	Global Model Test Results					
		Average loss	Accuracy	Precision	Recall	F1 Score	Confusion Matrix
TC1	CIFAR10	2.3045	0.10	0.0100	0.1000	0.0182	

TC2	CIFAR10	2.3035	0.100	0.0100	0.1000	0.0182	
TC3	CIFAR10	2.3028	0.100	0.0100	0.1000	0.0182	

APPENDIX N: Benchmarking Results

Project architecture adhering Trustworthy AI Regulation set up by European Union.

Table 52: Benchmark Discussion

TAI Principle	Self-Evaluation	Expert Evaluation	Compliance Level	Remarks
P1: Fairness	Model and Dataset agnostic ability	Recognizing efforts in the model agnosticism and	Medium	Strength in model adaptation requires further proof of fairness

		suggesting further datasets.		across diverse datasets.
P2: Privacy	Security and privacy preservation using differential privacy	Implementation of the differential privacy needs robust testing and advise on scalability.	High	Differential privacy well-implemented but consider scalability.
P3: Accountability	Contribution evaluation and incentive mechanism	Appreciates the contribution evaluation implementations SV values and motivates to study more on marginal contribution values	Medium	Contribution tracking is good, incentive system could be more robust.
P4: Robustness	Secure Aggregation using FedAvg	Positive feedback on the FedAvg mechanism implementation. Which also gives help on adversarial conditions.	High	Reven though robust yet requires adversarial testing for resilience.
P5: Explainability	Explainable mechanism with use of XAI	High praise for the implementation of explainable mechanism interpreting FL workflows.	High	Reinforcing the practical explainability

APPENDIX N-II : Functional Requirement Testing

Test Case	Action	Expected Outcome	Actual Outcome	Status
FT1	Clients assigning as a virtual worker.	Creating FL clients as virtual workers.	Creating FL clients as virtual workers.	PASS
FT2	Virtual workers performing training using unique data.	Clients performing FL training using their own allocated data.	Clients performing FL training using their own allocated data.	PASS
FT3	Performing secure aggregation.	System performs aggregation using client updates.	System performs aggregation using client updates.	PASS
FT4	Measuring client contributions.	System measures client contribution using Shapley values.	System measures client contribution using Shapley values.	PASS
FT5	Preserving privacy using privacy mechanisms.	System preserves privacy by applying differential privacy to the model parameters.	System preserves privacy by applying differential privacy to the model parameters.	PASS
FT6	Creating explanations and visualizations using explainable mechanism.	Explainable mechanism generating explanations and visualizations of FL workflows results.	Explainable mechanism generating explanations and visualizations of FL workflows results.	PASS
FT7	System administrator can start the training with custom configurations	Saving FL training configurations to start training.	Saving FL training configurations to start training.	PASS

FT8	Client and server admins might be able to view their corresponding pages	Displaying different dashboard pages based on the user role.	Displaying different dashboard pages based on the user role.	PASS
FT9	Providing debugging opportunities.	Displaying training and evaluation logs and results.	Displaying training and evaluation logs and results.	PASS
FT10	Server admin can download the final trained global model.	Storing the global model at cloud storage to download at any time.	Storing the global model at cloud storage to download at any time.	PASS
FT11	Server admin should be able to view evaluation of the global model.	Generating and presenting global model evaluation results as a detailed report.	Generating and presenting global model evaluation results as a detailed report.	PASS
FT12	Clients can view their contribution assessment with incentives.	Calculating and presenting the client contributions and the allocated incentives.	Calculating and presenting the client contributions and the allocated incentives.	PASS

APPENDIX O: Security Testing

```
PS D:\Rathe\Final Year\FYP\Implementation\NotionFL\notionfl-be\tests> python DP_attack_test.py
2024-04-10 19:54:27,168 - INFO - Testing differential privacy effectiveness against model poisoning.2024-04-10 19:54:27,169
2024-04-10 19:55:52,986 - INFO - Epoch 1/3, Loss: 1.6316
2024-04-10 19:57:29,026 - INFO - Epoch 2/3, Loss: 0.6153
2024-04-10 19:59:01,147 - INFO - Epoch 3/3, Loss: 0.4685
2024-04-10 19:59:04,550 - INFO - Applying differential privacy to client 0.

2024-04-10 19:59:04,602 - INFO - Training client 1.
2024-04-10 19:59:04,605 - INFO -
Training client 1 model...
2024-04-10 20:01:01,481 - INFO - Epoch 1/3, Loss: 1.6302
2024-04-10 20:02:38,225 - INFO - Epoch 2/3, Loss: 0.6136
2024-04-10 20:04:58,781 - INFO - Epoch 3/3, Loss: 0.4632
2024-04-10 20:05:01,706 - INFO - Applying differential privacy to client 1.

2024-04-10 20:05:01,751 - INFO - Training client 2.
2024-04-10 20:05:01,765 - INFO -
Training client 2 model...
2024-04-10 20:07:48,441 - INFO - Epoch 1/3, Loss: 1.6269
2024-04-10 20:11:45,491 - INFO - Epoch 2/3, Loss: 0.6164
2024-04-10 20:14:33,594 - INFO - Epoch 3/3, Loss: 0.4676
2024-04-10 20:14:43,075 - INFO - Applying differential privacy to client 2.
2024-04-10 20:14:43,112 - INFO - Training client 3.
2024-04-10 20:14:43,132 - INFO -
Training client 3 model...
2024-04-10 20:16:57,114 - INFO - Epoch 1/3, Loss: 1.6350
2024-04-10 20:20:06,443 - INFO - Epoch 2/3, Loss: 0.6160
2024-04-10 20:21:56,615 - INFO - Epoch 3/3, Loss: 0.4628
2024-04-10 20:21:59,719 - INFO - Applying differential privacy to client 3.
2024-04-10 20:21:59,757 - INFO - Training client 4.
2024-04-10 20:21:59,767 - INFO -
Training client 4 model...
2024-04-10 20:24:33,879 - INFO - Epoch 1/3, Loss: 1.6291
2024-04-10 20:27:09,856 - INFO - Epoch 2/3, Loss: 0.6123
2024-04-10 20:28:41,551 - INFO - Epoch 3/3, Loss: 0.4641
2024-04-10 20:28:44,105 - INFO - Applying differential privacy to client 4.
2024-04-10 20:28:44,141 - INFO - Simulating a malicious client with inverted gradients.
2024-04-10 20:31:09,496 - INFO - Performing secure aggregation with client updates.
2024-04-10 20:31:09,609 - INFO - Aggregation time: 0.09958791732788086s
 DP and handling malicious client: 0.5397
 DP and handling malicious client: 0.5397
F
```

```
----------------------------------------------------------------------
Ran 1 test in 490.305s

FAILED (failures=1)
PS D:\Rathe\Final Year\FYP\Implementation\NotionFL\notionfl-be\tests> python DP_attack_test.py
2024-04-10 19:35:46,137 - INFO - Testing differential privacy effectiveness against model poisoning.
2024-04-10 19:35:46,138 - INFO - Training client 0.
2024-04-10 19:35:46,141 - INFO -
Training client 0 model...
2024-04-10 19:36:38,943 - INFO - Epoch 1/3, Loss: 1.2850
2024-04-10 19:37:53,877 - INFO - Epoch 2/3, Loss: 0.5419
2024-04-10 19:39:17,351 - INFO - Epoch 3/3, Loss: 0.4320
2024-04-10 19:39:21,177 - INFO - Applying differential privacy to client 0.
2024-04-10 19:39:21,191 - INFO - Training client 1.
2024-04-10 19:39:21,192 - INFO -
Training client 1 model...
2024-04-10 19:40:35,270 - INFO - Epoch 1/3, Loss: 1.2854
2024-04-10 19:41:39,341 - INFO - Epoch 2/3, Loss: 0.5411
2024-04-10 19:42:59,064 - INFO - Epoch 3/3, Loss: 0.4350
2024-04-10 19:43:03,572 - INFO - Applying differential privacy to client 1.
2024-04-10 19:43:03,604 - INFO - Training client 2.
2024-04-10 19:43:03,607 - INFO -
Training client 2 model...
2024-04-10 19:44:58,205 - INFO - Epoch 1/3, Loss: 1.2792
2024-04-10 19:46:56,984 - INFO - Epoch 2/3, Loss: 0.5421
2024-04-10 19:47:59,737 - INFO - Epoch 3/3, Loss: 0.4320
2024-04-10 19:48:01,754 - INFO - Applying differential privacy to client 2.
2024-04-10 19:48:01,780 - INFO - Simulating a malicious client with inverted gradients.
2024-04-10 19:49:23,736 - INFO - Performing secure aggregation with client updates.
2024-04-10 19:49:23,760 - INFO - Aggregation time: 0.01589655876159668s
2024-04-10 19:49:30,825 - INFO - Final global model accuracy after applying DP and handling malicious client: 0.4782
F
======================================================================
FAIL: test_dp_effectiveness_against_poisoning (__main__.TestFLProcessWithDP)
```

Figure 40: DP model poisoning test

APPENDIX P: Self Evaluation

Table 53: Self Evaluation

Criteria	Rationale
Problem background, and problem novelty	The author's fascination towards the FL domain helped in discovering the research gap. As experts acknowledge the problem exists in the domain, the author's effort is to answer an ongoing open issue in the field of FL, which might be great addition to the domain. In that regard, the author believes this project and its results are noteworthy.
Research scope and complexity	Initially, the author had a limited scope due to the lack of confidence and knowledge in the domain. But later, once the author started to get comfortable with the domain, author managed to extend the scope further to make the research scope well defined with depth within the allocated timeframe.
Design and development decisions of the proposed solution	The development of the proposed system underwent iterative cycles, involving in repeated phases of design, development, and improvements. These iterations were critical in identifying the most suitable design for the specific scenario. This ultimately resulted in identifying innovative methods to create the NotionFL application.
Proposed Trustworthy cross-silo FL architecture	The novel trustworthy cross-silo FL was initially not planned to use trustworthy AI principles, and however the extensive research on LR made author to take up this approach. Technology selection was perfectly aligned with the core principles of Trustworthy AI. The comprehensive. The subsequent testing phase validated the effectiveness of this strategy in maintaining ethical standards and fostering trust in the proposed solution.
Proposed Explainable mechanism	The proposed explainable mechanism introduces the novel usage of shapely values to interpret the FL workflows. The mechanism receives different data types from the workflows and performs unique ways to generate explanations and visualizations. The test results indicated the

	enhancements of transparency in FL workflows and how helpful will it be to understand these processes of server eventually fostering trust.
Limitations and future work	Reflecting on the project, notable limitations include the absence of quantifiable trust metrics and limited explanations. The system also assumes server administrator legitimacy and relies heavily on computational resources. Future work can focus on developing trust measurement tools, enhancing agnostic ability, and exploring more efficient techniques to enhance both trustworthiness and explainability in real-world scenarios.

APPENDIX Q: Selection of Evaluators

Table 54: Selection of Evaluators

ID	Name	Position	Affiliation
EV1	Mr. Pubudhu Indrasiri	PhD Candidate AI/ML/DL/FL	Deakin University Melborne, Australia
EV2	Mr. Anuradha Wishmantha	DevOps Engineer, Software Architect	IFS R&D Internationals
EV3	Mr. Kanchana Ratnayake	Senior Information Security Analyst	WSO2
EV4	Mr. Imanshu Jayasinghe	Senior Software Engineer	Enpal GmbH, Berlin Germany
EV5	Mr. Sajana Chathuranga	Senior DevOps Engineer	IFS R&D Internationals
EV6	Mr. Akassharjun Shanmugarajah	Software Engineer, FL Researcher	Circles Life Sri Lanka
EV7	Mr. Sandeesh Croos	Associate Software Engineer specializing in AI/ML applications	Techlabs Global

EV8	Mr. Bhavaneetharan	Associate Software Engineer specializing in AI/ML applications	Techlabs Global
EV9	Mr. Abdul Basith	Software Engineer, Visiting Lecturer	Noon, IIT
EV10	Mr. Sadir Omer	Network Engineer	HSO, London, UK
EV11	Mr. Prashanna Sathiyamoorthy	Cyber security Analyst, master's student	University of Bedfordshire, UK
EV12	Mr. Thisaru Wickramasekara	Associate DevOps Engineer	IFS R&D Internationals

APPENDIX R: Qualitative Evaluation Result Analysis

Table 55: Qualitative evaluation

Criteria	Theme	Summary of the Feedback
Problem background and Problem novelty	Research gap, problem, depth, and complexity	Initial impression of the project and problem background are both quite impressive, given the unique and complex nature of the domain. Specifically, the project aims to build trust in federated learning, which is a challenge due to its distributed nature and complexity. One evaluator stated that he is so proud about the author to take up such impressive idea and understanding and having knowledge to understand in BSC level.
The project depth and scope	Novelty, Scope, Depth	The idea of using an explainable mediator is novel and critical in the FL space. This project showcases a comprehensive exploration of FL and is described as complex, broader than most undergraduate projects, and a challenge that was well managed and overcome. This project strikes as exciting but could also become very complex. One evaluator stated that idea of using an explainable mediator within FL is quite new and fit into the FL, where trust matters a lot. Further several evaluators also think novelty and relevance the project is critically addressing the research gap

Design and development of the proposed solution, solution, solution novelty	Architecture novelty, scope	The project offers a suitable level of technical difficulty for undergraduate research. It is complex and well thought-out, bringing valuable contributions to FL, focusing on resources, explainability, and security aspects of the system. One evaluator praised the suggested solution for its uniqueness and contribution to the Body of Knowledge (BoK), meeting desired requirements.
Analysis of the results	Results	Based on the feedback received, the proposed solution, which prioritizes transparency and can be easily understood, effectively meets the project's objectives, resulting in a clearer and more trusted FL. The approach has been highly praised for its adherence to TAI principles and its potential for real-world implementation, representing a noteworthy accomplishment for an undergraduate project. While the test results are encouraging and compelling, and the project is deemed relevant to the field, additional opportunities for growth exist. Specifically, a more comprehensive focus on practical application and continued advancement of system capabilities could substantially enhance the project's practical feasibility and industry acceptance.
Limitations and future improvements	suggestions	It's great to see a focus on trust in federated learning, as this will have a big impact on the field moving forward. Praise for the project's

		comprehensive nature indicates a successful end-to-end application, signaling robust execution and practical deployment.

APPENDIX S: Evidence for Expert Analysis Evaluation

Table 56:expert analysis evidence

Evaluator ID	Q1. What are your initial impressions on the project, the background and problem?
EV1	The project looks at building trust in federated learning with an explainable mediator. The background is well laid out, highlighting the challenges of trust in cross-silo federated learning.
EV2	excellent. I'm feeling proud that this guy is having such a talent/knowledge in BSC level
EV3	This project strikes as exciting, by focusing on the trustworthiness and interpretability in Federated Learning, particularly in environments where clients might have competing interests, it addresses a significant challenge
EV4	Notion FL appears to be a promising solution addressing the complexities of Federated Learning in diverse client environments. The project's focus on transparency, fairness, and privacy preservation suggests a comprehensive approach to overcoming the challenges associated with FL. The background provided establishes a clear understanding of the problem space, and the project aims to bridge the gap by providing a trustworthy mediator for FL processes.
EV5	This project looks promising and address a significant challenge which is a critical and timely issue in the field of AI and ML.
EV6	I was really impressed by the concept and the complexity of the project; it does address latest and emerging problem when it comes to ML

EV7	The project addresses issues in federated learning regarding trustworthiness and interpretability, especially in cross-silo environments. It acknowledges the competitive nature of organizations within the same domain, which affects collaboration and client cooperation.
EV8	I think trust is a real problem in cross silo federated learning and XAI would be a good starting point towards it
EV9	It is an interesting area of study, and one of the problems ingrained into machine learning is the challenging task of explaining the inner behaviour of the system.
EV10	The project demonstrates a strong understanding of the Federated Learning (FL) landscape, particularly within cross-silo environments. Overall, the project's initial impressions suggest a thorough understanding of the challenges and nuances within cross-silo FL
EV11	The Notion FL project is tackling a critical and novel problem within the realm of Federated Learning (FL), particularly in the cross-silo FL domain. The project's focus on enhancing trustworthiness and interpretability in this complex setting is commendable. By incorporating Responsible AI and Trustworthy AI principles, the team aims to create a comprehensive cross-silo FL architecture that can foster trust and cooperation among competing entities. Addressing the 'Black-box' nature of FL systems through improved interpretability and explainability is also a crucial aspect of this ambitious endeavour. Overall, the Notion FL project demonstrates a deep understanding of the current challenges in the field and is poised to make significant advancements in ethical and practical FL applications.

Table 57: Q2- Experts analysis evidence

Evaluator ID	**Q2. Is the depth of the research sufficient for undergraduate research?**
EV1	It's good enough for undergraduates to understand the problem and the solution.
EV2	more than enough

EV3	The re-search appears to be quite- extensive, which is impre-ssive for an undergraduate proje-ct
EV4	Yes
EV5	It looks very much sufficient for undergraduate level.
EV6	Yes, very much.
EV7	The depth of the research very high for undergraduate level, as it explores a specific problem in federated learning, proposes a solution, and provides initial test results.
EV8	Yes
EV9	Yes, utilizing relatively latest studied areas and methodologies, and attempting to incorporate/consider them in real-world scenarios that could benefit from them, while identifying the challenges that may be faced and proactively addressing them, is truly commendable.
EV10	-
EV11	"The depth of research for the NotionFL project appears to be sufficient for an undergraduate research endeavor. The project tackles a complex and multi-faceted problem within the field of Federated Learning (FL), which is indicative of a substantial research effort.

Table 58: Q3: experts evaluation evidence

Evaluator ID	Q3. Is the depth of the research sufficient for undergraduate research?
EV1	It's good enough for undergraduates to understand the problem and the solution.
EV2	more than enough
EV3	The re-search appears to be quite- extensive, which is impre-ssive for an undergraduate proje-ct
EV4	Yes
EV5	It looks very much sufficient for undergraduate level.
EV6	Yes, very much.
EV7	The depth of the research very high for undergraduate level, as it explores a specific problem in federated learning, proposes a solution, and provides initial test results.
EV8	Yes
EV9	Yes, utilizing relatively latest studied areas and methodologies, and attempting to incorporate/consider them in real-world scenarios that could benefit from them, while identifying the challenges that may be faced and proactively addressing them, is truly commendable.
EV10	
EV11	"The depth of research for the NotionFL project appears to be sufficient for an undergraduate research endeavor. The project tackles a complex and multi-faceted problem within the field of Federated Learning (FL), which is indicative of a substantial research effort.

Table 59: Q4: Experts evaluation evidence

Evaluator ID	Q4. From your perspective, are there any ways the system could be further enhanced?
EV1	May could focus on optimizing the performance of the explainable mediator and exploring additional ways to improve transparency and interpretability in federated learning.
EV2	yes. He has already considered about privacy, secure aggregation, incentive mechanisms for potential clients and etc. Apart from that, its good to add the security aspect to this. What if the clients will send the corrupted updates to the central server? how can we select the reliable clients? these are the two other main things to be focused.
EV3	
EV4	UI/UX Improvements: Enhancing the user interface to make it more intuitive and user-friendly could improve the overall user experience. Clear and concise visualizations, interactive features, and guided walkthroughs can help users navigate the system more efficiently.
EV5	Enhancement of privacy and security aspects of this approach is most important.
EV6	Enhancements would be based on the initial requirement of the system, If it's to be released in to the industry, i believe having more robust UI/UX and Access controls might enhance the system more.
EV7	To enhance the system, future iterations could explore additional techniques for improving interpretability and trustworthiness
EV8	"Can the explainable AI results expose the client data by any chance? If so that could use some improvements

EV9	
EV10	the scope of this project could be further refined"
EV11	

APPENDIX T: Evaluation Functional Requirements

IM – Implemented | **NIM** – Not Implemented

Table 60: Evaluation of functional Requirements

FR ID	Requirement	Priority Level	Status
FR1	Organizational clients have to be assigned to a virtual worker.	M	IM
FR2	Virtual Workers (organizations/clients) should be able to perform local training using their raw data.	M	IM
FR3	The system must perform secure aggregation using the model updates provided by the virtual workers after the local training.	M	IM
FR4	The system should be able to evaluate client's model updates to calculate the contribution evaluation measure.	M	IM
FR5	The system must have the security and privacy mechanisms running behind while the training phase is ongoing.	M	IM
FR6	The data collector mechanism should collect the logs/metrics and data related to the FL workflows.	M	IM
FR7	Explainable mechanisms should generate interpretable visualizations and explanations using the metrics collected by the data collector mechanism.	M	IM
FR8	The system administrators should be able to initiate the FL training.	M	IM

FR9	The clients and the system administrators should be able to view their correspondent user interfaces with relevant visualization related to the training phase.	M	IM
FR10	The system should provide functionality for debugging during the FL process through the corresponding user dashboards.	S	IM
FR11	The system must have the ability for training initiators to customize the FL training.	S	IM
FR12	The server admin can view the evaluation of the global models.	S	IM
FR13	The system should be able to give an option to the users to download specific logs and metrics for additional analysis without compromising the privacy violation.	W	NIM
FR14	The training initiators or server administrators could be able to download the global models.	W	NIM
FR15	Distributing incentives based on the contribution measure of each organization.	C	IM

APPENDIX: U: Evaluation non-functional requirements.

IM – Implemented | **NIM** – Not Implemented

Table 61:Evaluation of non- functional Requirements

NFR ID	Description	Priority Level	Status
NFR1	The system should not suffer from performance issues compared to general cross-silo Federated learning systems.	M	IM
NFR1	The system must have robust security measures to protect the sensitive data during the transmission and processing, adhering to the industry standard encryption techniques and protocols.	M	IM
NFR1	The system must achieve its main objectives accurately without any compromission.	S	IM
NFR1	The system must be compatible with various data formats and machine learning models to support a wide range of use cases.	S	IM
NFR1	The system should have to be scalable to accommodate the increasing number of virtual workers without the degradation in performance by deploying it to cloud.	C	NIM
NFR1	The user interface of the proposed system should manage visualizing all the necessary FL tasks with friendly user experience without complicating the interface.	C	IM