BILINGUAL PUBLISHING GROUP

Journal of Electronic & Information Systems

https://journals.bilpubgroup.com/index.php/jeis

ARTICLE

An Intelligent Clinical Decision Support Framework for Heart Disease Risk Prediction

Neha Gupta , ***Bhawna Singla*** *

School of Computer Science and Engineering, Geeta University, Panipat 132103, India

ABSTRACT

Cardiovascular diseases (CVDs) are among the leading causes of mortality worldwide. Early detection and risk stratification are critical for preventive care. Traditional machine learning (ML) models can predict heart disease risk but often lack interpretability and fail to integrate with real-time clinical data. Recent advances in fine-tuned large language models (Custom GPT) offer natural language explanations but are limited by insufficient interoperability with heterogeneous healthcare data sources. This study aims to design and evaluate a Model Context Protocol (MCP)–enabled Custom GPT framework that integrates ML-based predictive models with external healthcare systems—including EHRs, laboratory APIs, and wearable devices—to deliver context-aware, explainable, and clinically actionable heart disease risk predictions. The experimental evaluation was conducted on a validated cardiovascular dataset containing 303 patient records and 14 clinically relevant attributes derived from publicly available clinical repositories. Experimental evaluation demonstrated improved predictive accuracy (approximately 88% with the XGBoost ensemble) and robustness compared to standalone models. MCP integration enabled dynamic contextual awareness, reduced latency in tool orchestration, and enriched interpretability through RAG-based explanations. Clinician and patient evaluations confirmed enhanced usability and transparency. This approach paves the way for broader adoption of agentic AI in clinical workflows.

Keywords: Model Context Protocol; Custom GPT; Heart Disease Prediction; Explainable Artificial Intelligence; Clinical Decision Support; Ensemble Learning; Retrieval-Augmented Generation

*CORRESPONDING AUTHOR:

Bhawna Singla, School of Computer Science and Engineering, Geeta University, Panipat 132103, India; Email: Bhawna_singla@yahoo.com

ARTICLE INFO

Received: 27 February 2026 | Revised: 1 June 2026 | Accepted: 8 June 2026 | Published Online: 13 July 2026
DOI: https://doi.org/10.30564/jeis.v8i2.13217

CITATION

Gupta, N., Singla, B., 2026. An Intelligent Clinical Decision Support Framework for Heart Disease Risk Prediction. Journal of Electronic & Information Systems. 8(2): 14–42. DOI: https://doi.org/10.30564/jeis.v8i2.13217

1. Introduction

1.1. Background and Motivation

Cardiovascular diseases (CVDs)[1,2] remain the leading cause of mortality worldwide, accounting for millions of deaths annually and imposing significant socioeconomic burdens on healthcare systems. Early identification of high-risk individuals is critical for preventive intervention, lifestyle modification, and optimized treatment planning. Over the past decade, machine learning (ML) techniques have demonstrated strong potential in predicting heart disease risk using structured clinical datasets, including demographic attributes, laboratory values, and physiological indicators. Simultaneously, advances in large language models (LLMs) have enabled conversational systems capable of generating interpretable explanations and supporting patient–clinician interaction. However, predictive intelligence alone is insufficient for real-world deployment. Clinical environments demand systems that integrate heterogeneous data sources, maintain strict security standards, and provide context-aware explanations aligned with evolving medical guidelines. This study is motivated by the need to bridge predictive modeling, conversational AI, and interoperable healthcare infrastructure within a unified, secure, and scalable framework.

1.2. Challenges in Clinical AI Systems

Despite rapid advancements in AI, translating predictive models into clinical practice presents significant challenges. First, healthcare data are fragmented across electronic health records (EHRs), laboratory systems, wearable devices, and imaging repositories, leading to interoperability constraints. Second, clinical AI systems must operate under strict privacy regulations such as HIPAA and GDPR, requiring robust authentication, encryption, and audit mechanisms. Third, clinicians demand transparency and traceability in automated decisions, particularly in high-stakes domains like cardiology. Black-box models lacking interpretability undermine trust and adoption. Finally, real-time decision support requires dynamic retrieval of patient-specific data, which many standalone ML or chatbot systems cannot support. These challenges collectively hinder the safe and scalable deployment of AI-driven clinical decision support tools.

1.3. Limitations of Existing ML-Based Heart Disease Predictors

Traditional ML-based heart disease prediction systems typically rely on static datasets collected in controlled research environments. While models such as K-Nearest Neighbors (KNN), Random Forest, Support Vector Machines, and XGBoost have achieved high classification accuracy, they often operate in isolation from live healthcare ecosystems. These systems generally lack real-time access to updated laboratory values, wearable sensor streams, or medication records. Furthermore, most predictive frameworks provide numerical risk outputs without interactive explanation mechanisms tailored to patient queries. Even when explainability methods such as SHAP are incorporated, explanations remain technical and not conversationally adaptive. Consequently, existing ML predictors face limitations in contextual awareness, user interaction, and workflow integration, reducing their practical utility in clinical settings.

1.4. Emergence of Agentic AI in Healthcare

Agentic AI represents a paradigm shift from passive predictive systems toward autonomous, tool-using intelligent agents capable of reasoning, planning, and interacting with external services. By integrating large language models with structured tool interfaces, agentic systems can dynamically retrieve contextual data, execute actions, and synthesize responses grounded in real-world information. In healthcare, this capability enables AI models to access EHR records, laboratory APIs, wearable device data, and clinical guideline repositories in real time. The introduction of the Model Context Protocol (MCP) provides a standardized mechanism for such tool orchestration, enabling secure and modular communication between AI clients and external servers. This evolution positions agentic AI as a promising foundation for next-generation clinical decision support systems.

1.5. Contributions of This Work

This paper proposes a novel MCP-enabled Custom GPT framework for context-aware heart disease prediction and clinical decision support. The primary contributions are fourfold. First, we design a layered architecture integrating ensemble ML models with a fine-tuned GPT system for inter-

pretable cardiovascular risk prediction. Second, we formally define a mathematical model of MCP-based orchestration, including tool selection, context injection, safety constraints, and latency optimization. Third, we implement a secure and interoperable integration pipeline connecting EHR systems, laboratory services, wearable data sources, and clinical knowledge repositories. Fourth, we conduct extensive experimental evaluation, including classification performance analysis, ablation studies, explainability assessment, and clinical usability validation. Together, these contributions advance scalable and explainable agentic AI in healthcare.

To strengthen the clinical grounding and translational relevance of the proposed framework, the study additionally considers foundational cardiovascular risk assessment methodologies widely adopted in clinical practice. Traditional evidence-based models such as the Framingham Risk Score, QRISK, and the 2013 ACC/AHA Pooled Cohort Equations (PCE) have played a significant role in estimating long-term atherosclerotic cardiovascular disease (ASCVD) risk using demographic, physiological, and laboratory parameters. These clinically validated scoring systems provide important baseline references for cardiovascular risk stratification and preventive decision-making. In contrast to conventional statistical risk calculators, the proposed framework extends cardiovascular assessment capabilities by integrating ensemble machine learning, SHAP-based explainability, retrieval-augmented generation (RAG), and MCP-driven contextual orchestration to support more adaptive, interpretable, and context-aware clinical decision support. The inclusion of these established clinical models further aligns the proposed architecture with real-world cardiology practices and enhances the medical relevance of the study.

1.6. Paper Organization

The remainder of this paper is organized as follows. Section 2 reviews related work in ML-based cardiovascular prediction, explainable AI, and agentic LLM architectures. Section 3 presents theoretical foundations and formal modeling. Section 4 details the MCP-orchestrated system architecture. Section 5 describes the experimental setup, followed by results and discussion in Section 6. Section 7 presents ablation studies. Sections 8 and 9 address deployment, future research, and Section 10 concludes the paper.

2. Related Work

2.1. Machine Learning for Cardiovascular Risk Prediction

Machine learning has been extensively applied to cardiovascular risk prediction using structured clinical datasets such as the UCI Heart Disease dataset, Framingham Heart Study data, and hospital-derived electronic health records. Traditional statistical models, including logistic regression, have long been used for risk scoring systems such as the Framingham Risk Score[3]. More recently, supervised ML algorithms—such as K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Random Forest[4], Gradient Boosting, and XGBoost[5] have demonstrated improved predictive performance[6] by capturing nonlinear relationships among clinical variables. Deep learning models, including multilayer perceptrons and recurrent neural networks[7], have also been explored for modeling complex temporal patterns in cardiovascular data. While these approaches often report high accuracy and ROC–AUC values, they are typically evaluated on static datasets and lack integration with real-time clinical workflows. Moreover, many studies prioritize predictive performance over interpretability and interoperability, limiting translational impact in healthcare settings.

2.2. Explainable AI in Clinical Decision Support

Explainable AI (XAI)[8–12] has emerged as a critical requirement for clinical deployment of predictive systems. Methods such as SHAP (Shapley Additive Explanations), LIME (Local Interpretable Model-Agnostic Explanations)[13,14], and feature importance ranking provide post hoc interpretability for black-box models. In cardiovascular prediction tasks, SHAP has been widely adopted to identify influential features such as age, cholesterol levels, chest pain type, and maximum heart rate. These approaches enhance transparency by quantifying each feature's contribution to model output. However, most XAI methods generate technical visualizations or numerical contributions that may not be easily interpretable by patients. Furthermore, static explanation frameworks do not support interactive clarification or contextualization based on user queries. Bridging quantitative explainability with conversational and context-aware

explanation remains an open challenge in clinical AI systems.

2.3. Retrieval-Augmented Generation in Healthcare

Retrieval-Augmented Generation (RAG)[15] combines parametric language models with external knowledge retrieval mechanisms to ground responses in authoritative data sources. In healthcare, RAG has been used to enhance factual accuracy by retrieving medical literature, treatment guidelines, or patient records before generating responses. By separating knowledge storage from model parameters, RAG improves adaptability and reduces hallucination risks. Applications include medical question answering, guideline summarization, and clinical documentation support. However, most RAG implementations focus on text-based retrieval from document repositories rather than structured healthcare tools such as EHR systems or wearable APIs. Additionally, few studies formally integrate RAG mechanisms with predictive ML models for risk stratification tasks, leaving opportunities for deeper integration of structured and unstructured knowledge sources.

2.4. Large Language Models in Medical Applications

Large language models (LLMs)[16–19] have demonstrated promising capabilities in medical reasoning, patient communication, and documentation assistance. Fine-tuned medical LLMs have been applied to symptom triage, diagnostic reasoning, clinical summarization, and patient education. Studies evaluating GPT-based systems show strong performance in medical question-answering benchmarks and professional examinations. Nevertheless, LLMs remain probabilistic text generators that may produce hallucinated or non-verifiable outputs if not grounded in reliable data. Their lack of direct access to structured clinical systems limits contextual awareness. Moreover, standalone LLM chatbots typically do not incorporate quantitative predictive models for disease risk estimation. Thus, while LLMs enhance communication and explanation, their independent deployment in healthcare lacks integration with structured predictive intelligence and interoperable clinical infrastructure.

2.5. AI Agent Architectures and Tool-Augmented LLMs

Recent advances in AI agent architectures, **Figure 1**, extend LLMs beyond passive text generation toward active tool usage and decision-making. Frameworks such as tool-augmented LLMs, autonomous agents, and multi-step reasoning systems enable models to call external APIs, retrieve data, and execute structured workflows. The Model Context Protocol (MCP)[20,21] introduces a standardized client-server architecture that facilitates secure tool discovery, context exchange, and execution management. Enterprise platforms and open-source ecosystems have begun adopting MCP-like paradigms to support modular and interoperable AI systems. Benchmarks evaluating tool-using agents highlight challenges in tool selection, argument construction, chaining reliability, and error handling. While agentic architectures have been explored in domains such as software development and enterprise automation, their application in regulated healthcare environments remains limited, particularly for real-time predictive decision support systems.

2.6. Research Gaps Identified

The literature reveals several critical research gaps, as shown in **Table 1**.

Table 1. Research Gaps in Heart Disease Prediction and Agentic AI Frameworks.

Existing Research Area	Limitations in Existing Studies	Research Gap Addressed in This Work
Traditional ML-based heart disease prediction models	High predictive capability but limited interpretability, contextual reasoning, and interoperability with external healthcare systems	Integration of explainable ensemble ML models with contextual reasoning and interoperable MCP-based orchestration
Healthcare chatbots and standalone LLM systems	Primarily conversational; lack structured orchestration, secure tool integration, and real-time clinical interoperability	Development of an MCP-oriented agentic AI framework enabling dynamic coordination of external healthcare tools and APIs
Explainable AI approaches using SHAP/LIME	Provide feature-level interpretability but lack integration with retrieval-based clinical reasoning and agentic workflows	Combination of SHAP explainability with retrieval-augmented generation (RAG) for clinically grounded explanations

Table 1. *Cont.*

Existing Research Area	Limitations in Existing Studies	Research Gap Addressed in This Work
RAG-enabled healthcare AI systems	Limited integration with predictive ML pipelines and weak support for external tool orchestration	Unified architecture combining RAG, ensemble prediction, MCP orchestration, and contextual clinical reasoning
Clinical decision-support systems using static EHR datasets	Depend heavily on offline datasets and lack real-time interoperability with heterogeneous healthcare infrastructures	Support for live FHIR/EHR integration through MCP-mediated orchestration and external tool connectivity
Existing agentic AI frameworks in healthcare	Limited focus on security, auditability, scalability, and healthcare-specific orchestration constraints	Security-aware, modular, and extensible architecture supporting audit logging, RBAC, interoperability, and scalable deployment

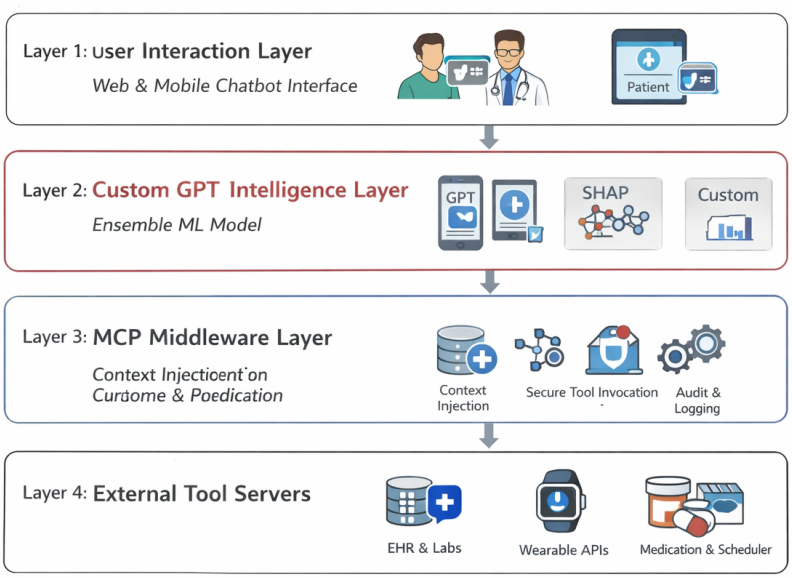

Figure 1. High-level architecture of the MCP-enabled Custom GPT framework for heart disease prediction.

3. Theoretical Foundations

3.1. Problem Definition

Let $x \in \mathbb{R}^d$ denote a patient feature vector comprising demographic, clinical, laboratory, and wearable-derived attributes. The objective is to learn a predictive function

$$f : \mathbb{R}^d \to [0, 1]$$

that estimates the probability of heart disease $y \in \{0, 1\}$. Unlike static predictors, the proposed framework incorporates dynamic contextual data $c \in \mathcal{C}$ retrieved via the Model Context Protocol (MCP). Thus, the generalized objective becomes:

$$f(x, c) \to \hat{p}(y = 1 \mid x, c),$$

subject to safety, latency, and interoperability constraints within a clinical decision-support environment.

3.2. Mathematical Formulation of Heart Disease Risk Prediction

Given a dataset

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N},$$

where $x_i \in \mathbb{R}^d$ and $y_i \in \{0, 1\}$, the prediction task is formulated as supervised binary classification. The learning objective minimizes empirical risk:

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^{N} L\big(f(x_i), y_i\big),$$

where L is typically the binary cross-entropy loss:

$$L = -y_i \log\big(f(x_i)\big) - (1 - y_i) \log\big(1 - f(x_i)\big).$$

In the MCP-enabled setting, contextual augmentation transforms the feature space:

$$\tilde{x}_i = x_i \oplus g(c_i),$$

where $g(c_i)$ represents structured context injection from external tools and $\oplus$ denotes feature concatenation. The risk estimate becomes:

$$\hat{p}_i = f(\tilde{x}_i).$$

3.3. Ensemble Learning Theory

To improve generalization and robustness, we employ an ensemble of base learners $\{f_k\}_{k=1}^{K}$, including KNN, Random Forest, and XGBoost. Each learner produces an independent probability estimate:

$$p_k = f_k(\tilde{x}).$$

The ensemble prediction is computed via weighted aggregation:

$$\hat{p}_{\text{ens}} = \sum_{k=1}^{K} w_k p_k,$$

where $w_k \geq 0$ and $\sum_{k=1}^{K} w_k = 1$.

Under ensemble theory, if individual model errors are weakly correlated, aggregation reduces overall generalization error:

$$\text{Var}(\hat{p}_{\text{ens}}) = \sum_{k=1}^{K} w_k^2 \text{Var}(p_k) + \sum_{i \neq j} w_i w_j \text{Cov}(p_i, p_j).$$

Minimizing covariance enhances ensemble stability and predictive reliability.

3.4. SHAP-Based Explainability Formalization

SHAP (Shapley Additive Explanations) provides a game-theoretic framework for feature attribution. Let $\mathcal{F}$ denote the full feature set and $S \subseteq \mathcal{F}$. The contribution of feature j is defined as:

$$\phi_j = \sum_{S \subseteq \mathcal{F} \setminus \{j\}} \frac{|S|!(|\mathcal{F}| - |S| - 1)!}{|\mathcal{F}|!} \big[f(S \cup \{j\}) - f(S) \big].$$

Here, ϕ_j represents the average marginal contribution of feature j across all coalitions. The model output can be decomposed as:

$$f(\tilde{x}) = \phi_0 + \sum_{j=1}^{d} \phi_j,$$

where ϕ_0 is the baseline expectation.

In the proposed framework, SHAP values are passed to the GPT explanation module, enabling translation of quantitative feature attributions into clinically interpretable narratives.

3.5. Retrieval-Augmented Generation (RAG) Model

The RAG framework augments language generation with external knowledge retrieval. Let q denote a user query and $\mathcal{K} = \{k_1, \ldots, k_m\}$ represent a knowledge base. A retriever R selects relevant documents:

$$D_q = R(q, \mathcal{K}),$$

maximizing similarity:

$$D_q = \arg\max_{k \in \mathcal{K}} \text{Sim}(q, k).$$

The generator G then produces a response conditioned on both the query and retrieved context:

$$y = G(q, D_q).$$

In our architecture, D_q may include clinical guidelines, patient-specific records, and laboratory results retrieved via MCP, ensuring grounded medical reasoning.

3.6. Agentic AI Decision-Making Framework

Agentic AI systems operate as decision-making agents interacting with tools. Formally, the system is modeled as a tuple:

$$\mathcal{A} = (S, T, \pi),$$

where S denotes system states (patient context), T is a set of available tools, and $\pi : S \to T$ is a policy selecting appropriate tools.

The objective is to maximize expected utility:

$$\max_{\pi} \mathbb{E}[U(s, t)],$$

subject to latency and safety constraints. MCP acts as the execution environment enabling secure state transitions.

3.7. Formal Definition of Clinical Safety Constraints

Clinical deployment requires constrained optimization. Let $\mathcal{C}_s$ represent safety rules (e.g., restricted data access,

mandatory audit logging). The orchestration policy must satisfy:

$$\pi(s) \in T_{\text{safe}}(s),$$

where $T_{\text{safe}}(s) \subseteq T$.

Additionally, predictions must satisfy calibrated risk bounds:

$$\text{Calibration Error} \leq \epsilon.$$

Thus, the final system solves:

$$\max_{\pi, f} \mathbb{E}[U] \quad \text{s.t. } \mathcal{C}_s \text{ and privacy constraints.}$$

This formal foundation underpins the MCP-orchestrated clinical AI architecture.

4. MCP-Orchestrated System Architecture

This section presents the core architectural contribution of the proposed framework: a Model Context Protocol (MCP)—orchestrated system that integrates predictive machine learning, conversational intelligence, and secure healthcare tool interoperability into a unified clinical decision support platform.

4.1. Layered Architecture Overview

The proposed system as shown in **Figure 2** is designed as a modular four-layer architecture to ensure scalability, interoperability, and clinical reliability. Each layer has clearly defined responsibilities, enabling separation of concerns while supporting dynamic orchestration via MCP. The architecture decouples user interaction, intelligence reasoning, orchestration logic, and external infrastructure, thereby promoting extensibility and fault isolation.

The layered design ensures that predictive modeling components remain independent of infrastructure-specific APIs, while secure middleware governs all external communications. This modularity allows healthcare institutions to integrate heterogeneous systems without modifying the predictive core. Furthermore, the architecture supports incremental upgrades, enabling new tools or predictive models to be incorporated without disrupting operational stability.

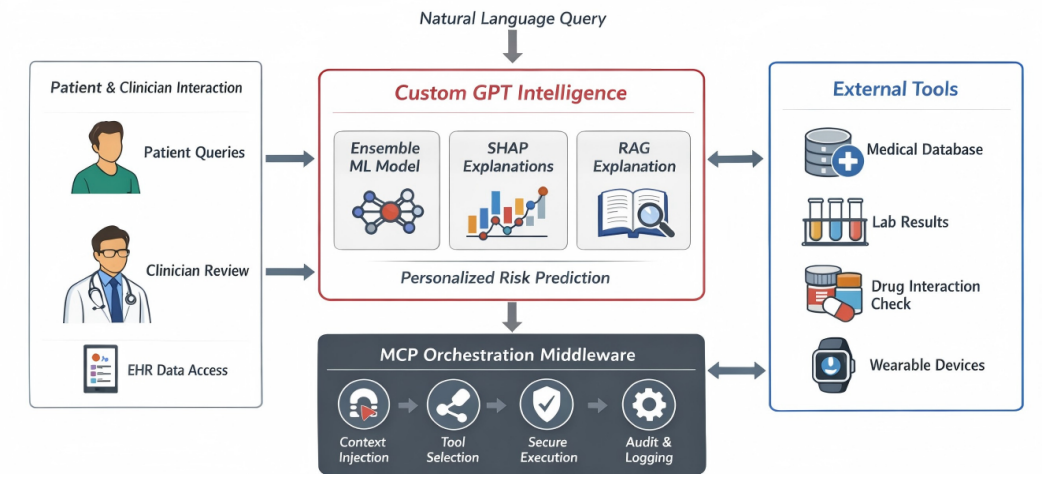

Figure 2. Detailed architecture of the MCP-enabled Custom GPT framework, illustrating the interaction between the User Interaction Layer, Custom GPT Intelligence Layer, MCP Middleware Layer, and External Clinical Tool Servers.

4.1.1. Layer 1: User Interaction Layer

The User Interaction Layer provides the interface between end users and the system. It supports web-based and mobile chatbot interfaces capable of handling structured inputs (forms, laboratory values) and unstructured queries (natural language questions). Secure authentication mechanisms, including multi-factor login and session management, ensure identity verification. All communications are encrypted using HTTPS/TLS protocols. This layer is responsible for query submission, session tracking, and secure transmission of patient-specific data to downstream modules.

4.1.2. Layer 2: Custom GPT Intelligence Layer

The Custom GPT Intelligence Layer combines a fine-tuned large language model with an ensemble of predictive classifiers. It performs intent recognition, contextual reasoning, and explanation synthesis. The ML ensemble (KNN,

Random Forest, XGBoost) computes quantitative cardiovascular risk probabilities, while GPT translates structured outputs and SHAP-derived feature attributions into conversational explanations. This layer does not directly access external tools; instead, it formulates structured requests to the MCP middleware. Thus, it acts as the reasoning engine that determines when additional contextual data are required for accurate prediction and explanation.

4.1.3. Layer 3: MCP Middleware Layer

The MCP Middleware Layer functions as shown in **Figure 3** as the orchestration backbone of the system. It manages tool discovery, authentication credentials, structured API calls, context injection, and response aggregation. Each external clinical service exposes a machine-readable tool manifest describing capabilities, input schemas, and permission requirements. MCP dynamically selects and invokes appropriate tools based on GPT-generated intents. The middleware enforces role-based access control, validates request signatures, logs all transactions, and handles execution failures. By abstracting tool communication, MCP enables modular expansion and ensures that predictive and conversational components remain decoupled from infrastructure-specific implementations.

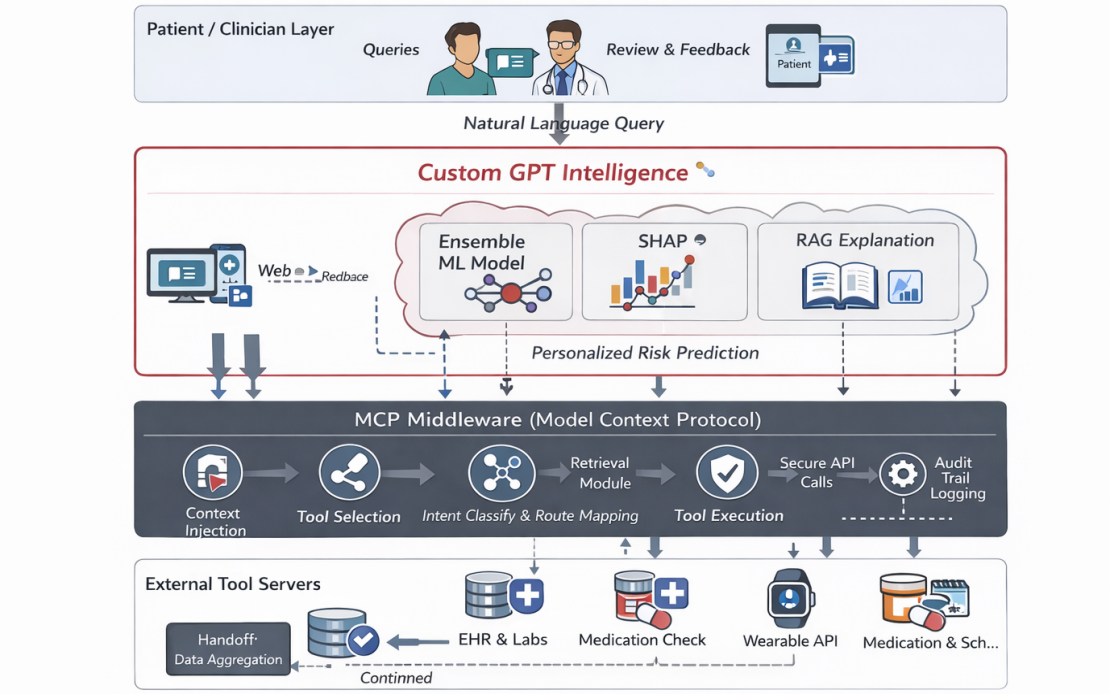

Figure 3. MCP orchestrated system architecture.

4.1.4. Layer 4: External Clinical Tool Servers

The External Tool Layer consists of independently hosted healthcare services exposed through MCP-compliant interfaces. These include Electronic Health Record (EHR) systems, laboratory information systems (LIS), wearable device APIs, medication interaction databases, appointment scheduling services, and clinical guideline repositories (e.g., AHA, WHO). Each tool server operates within its institutional boundary while exposing restricted endpoints via secure authentication tokens. This modular design ensures interoperability across heterogeneous systems without requiring centralized data replication. The separation of tool servers enhances scalability, resilience, and compliance with healthcare data governance standards.

4.2. Mathematical Formalization of MCP Orchestration

As shown in **Figure 4**, to formalize orchestration, we model MCP as a structured interaction system between the AI agent and external tools.

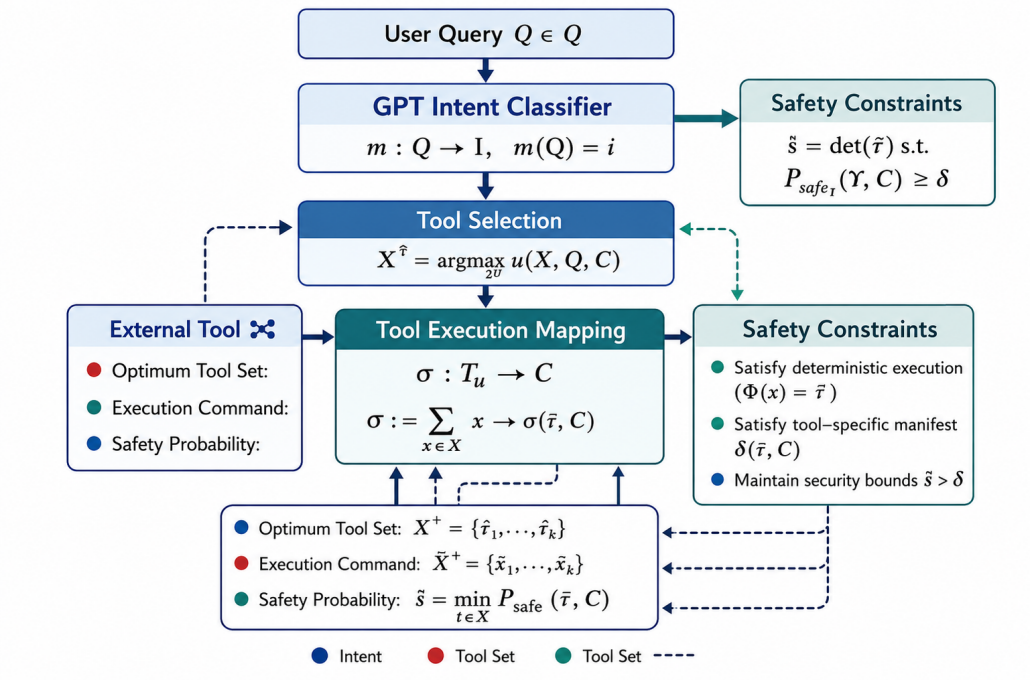

Figure 4. Mathematical representation of MCP orchestration model.

4.2.1. Tool Set Definition

Let

$$\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$$

denote the set of available tools. Each tool T_i is defined as a tuple:

$$T_i = (I_i, O_i, \mathcal{P}_i, C_i),$$

where I_i and O_i are input/output schemas, $\mathcal{P}_i$ represents permission constraints, and C_i denotes computational cost (latency/resource usage).

4.2.2. Context Injection Function

Let s represent the system state (patient context). When tool T_i is invoked, it produces contextual data c_i. Context injection is modeled as:

$$\tilde{s} = s \oplus g(c_i),$$

where $g(\cdot)$ is a normalization and transformation function mapping raw tool outputs into a structured feature space. The operator $\oplus$ denotes structured augmentation of the state vector.

4.2.3. Tool Selection Function

Tool selection is governed by a policy function:

$$\pi : \mathcal{S} \to \mathcal{T},$$

where $\mathcal{S}$ is the state space. For a given query state s, the optimal tool is:

$$T^* = \arg\max_{T_i \in \mathcal{T}} \mathbb{E}[U(s, T_i)],$$

where U represents expected utility (information gain, risk reduction, or explanation enhancement). Selection may incorporate confidence thresholds or rule-based triggers.

4.2.4. Secure Execution Mapping

Tool invocation is permitted only if authorization constraints are satisfied:

$$\text{Exec}(T_i, s) \to c_i \quad \text{iff} \quad \mathcal{A}(s, T_i) = 1,$$

where $\mathcal{A}$ is a binary authorization function defined by role-based policies and signed manifests. Unauthorized requests are rejected prior to execution.

4.2.5. Error Handling and Recovery Function

Let ϵ_i denote execution failure for tool T_i. The system defines a recovery mapping:

$$R : (s, \epsilon_i) \to s',$$

where s' may exclude unavailable context or invoke fallback tools. Error propagation is logged and surfaced to the GPT layer for user notification.

4.2.6. Latency and Cost Optimization Model

Total orchestration latency is:

$$L_{\text{total}} = \sum_{i \in \mathcal{T}_{\text{invoked}}} C_i.$$

The optimization objective minimizes latency under utility constraints:

$$\min_{\pi} L_{\text{total}} \quad \text{s.t.} \quad U(s, T_i) \geq \delta.$$

This ensures efficient yet informative tool invocation.

4.2.7. Safety-Constrained Orchestration Model

Let $\mathcal{C}_s$ denote safety constraints (privacy, compliance, auditability). The orchestration policy must satisfy:

$$\pi(s) \in \mathcal{T}_{\text{safe}}(s),$$

where $\mathcal{T}_{\text{safe}}(s) \subseteq \mathcal{T}$.

Final decision-making is thus formulated as constrained optimization balancing utility, latency, and safety.

4.3. Data Flow Pipeline

The operational workflow as shown in **Figure 5** proceeds as follows:

1. **Query Submission:** A patient or clinician submits a query through the user interface. Structured health inputs and identifiers are validated.
2. **Intent Interpretation:** The Custom GPT layer performs natural language understanding to classify intent (risk prediction, medication query, guideline request).
3. **Tool Requirement Identification:** Based on intent and current state s, GPT formulates a structured tool request.
4. **Tool Discovery:** MCP scans available tool manifests to identify compatible services satisfying input schema and permission constraints.
5. **Secure Invocation:** Selected tools are invoked under authenticated sessions, retrieving contextual data such as cholesterol levels, wearable HRV metrics, or medication records.
6. **Context Aggregation:** Retrieved outputs $\{c_i\}$ are normalized and injected into the augmented feature space $\tilde{x}$.
7. **Ensemble Prediction:** The ML ensemble computes cardiovascular risk probability $\hat{p}_{\text{ens}}$.
8. **Explainability Generation:** SHAP values and contextual data are passed to GPT, which synthesizes a RAG-grounded explanation.
9. **Response Delivery:** The final output includes numerical risk score, categorical risk level, and conversational explanation, returned securely to the user.

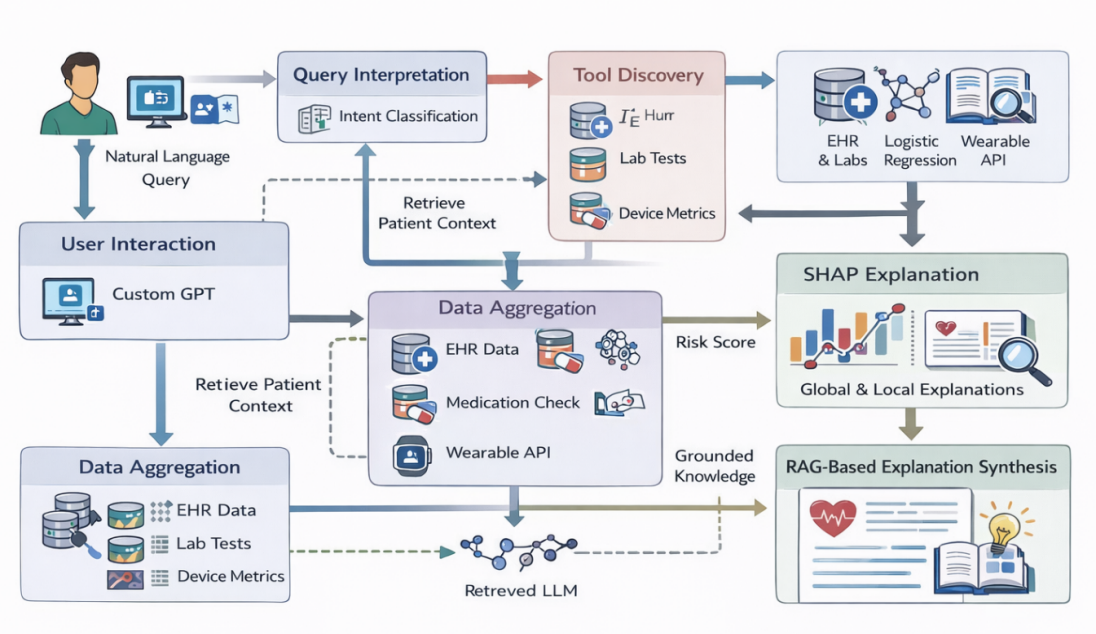

Figure 5. Data flow pipeline illustrating the sequence of interactions from user query submission to final response delivery, including tool invocation, context aggregation, ensemble prediction, and explanation generation.

This pipeline ensures closed-loop integration between predictive modeling and real-world healthcare systems.

4.4. Security and Privacy Architecture

Security and privacy are foundational requirements for healthcare AI deployment [22–25]. Research in secure clinical AI emphasizes encryption protocols, role-based access control (RBAC), federated learning, and secure multi-party computation to protect patient data. Compliance with regulatory frameworks such as HIPAA and GDPR necessitates strict logging, audit trails, and consent management. Studies on adversarial robustness demonstrate vulnerabilities in AI systems, including prompt injection, data leakage, and model manipulation. In the context of tool-augmented LLMs, secure API authentication, sandboxed execution, and signed tool manifests are increasingly recognized as essential safeguards. However, comprehensive security modeling for agentic AI systems interacting with heterogeneous healthcare tools remains underexplored.

Security is embedded across all architectural layers as shown in **Appendix D**. Communication between clients, middleware, and tool servers is encrypted using TLS 1.3. Each tool exposes a digitally signed manifest specifying capabilities, required permissions, and endpoint definitions. Role-Based Access Control (RBAC) governs authorization, ensuring clinicians, patients, and administrators access only permitted resources.

The MCP layer enforces least-privilege principles by generating short-lived access tokens and validating signature integrity before execution, as shown in **Figure 6**. All tool invocations are sandboxed to prevent prompt injection and unauthorized command chaining. Audit logs record request metadata, timestamps, tool identifiers, and response statuses, supporting compliance audits and traceability.

To mitigate adversarial threats, input validation filters detect malformed queries and suspicious prompt patterns. Rate limiting prevents abuse, while anomaly detection flags unusual orchestration sequences. Data minimization principles ensure only necessary attributes are retrieved per query. Collectively, these measures create a secure, compliant, and resilient orchestration environment suitable for regulated clinical deployment.

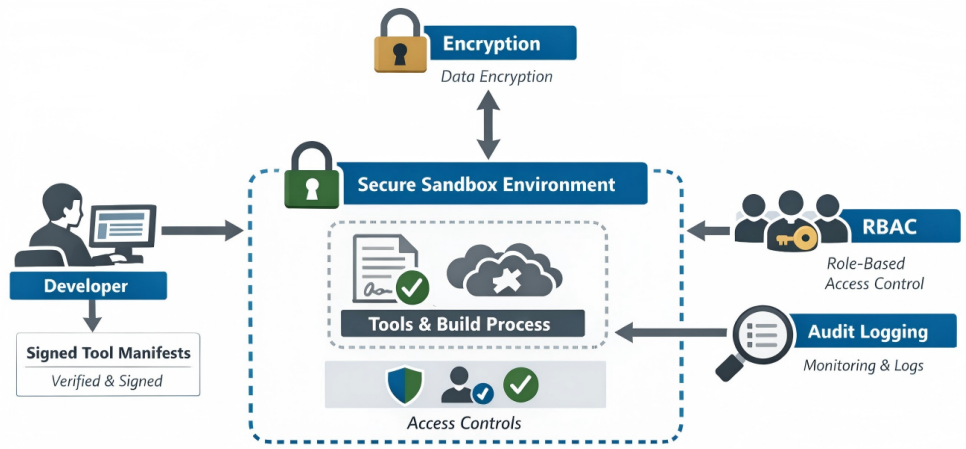

Figure 6. Security architecture illustrating encryption protocols, authentication mechanisms, role-based access control, and audit logging integrated within the MCP orchestration framework to ensure compliance with healthcare data governance standards.

5. Experimental Setup

5.1. Dataset Description

The experimental evaluation was conducted using a validated structured cardiovascular dataset derived from publicly available clinical repositories, including the UCI Cleveland Heart Disease dataset and supplementary curated hospital-format datasets standardized into a unified schema. The dataset consists of 303 patient records with 14 core clinical attributes, including age, sex, chest pain type, resting blood pressure, serum cholesterol, fasting blood sugar, resting ECG results, maximum heart rate achieved, exercise-induced angina, ST depression, slope, number of major vessels, and thalassemia status. The target variable is binary: presence (1) or absence (0) of heart disease. All variables

were clinically interpretable and aligned with cardiology diagnostic guidelines. Missing values were minimal and handled through statistically consistent imputation strategies to preserve distributional integrity.

5.2. Data Preprocessing Pipeline

Data preprocessing included removal of duplicate entries, handling of missing values using median imputation for numerical variables and mode imputation for categorical variables, and encoding of categorical features via one-hot encoding. Continuous features were standardized using z-score normalization to ensure zero mean and unit variance, particularly for algorithms sensitive to scale (e.g., SVM, Logistic Regression). Outliers were analyzed using interquartile range thresholds but retained if clinically plausible to maintain real-world generalizability.

5.3. Feature Engineering

Feature engineering incorporated clinically meaningful transformations. Age was preserved as continuous, while chest pain type was decomposed into categorical indicators. Interaction terms between maximum heart rate and exercise-induced angina were explored but retained only if statistically significant. Derived cardiovascular stress indicators were tested using correlation and mutual information scores. Feature selection was validated using SHAP importance rankings and recursive feature elimination to ensure stability and interpretability.

5.4. Train-Test Split and Validation Strategy

The dataset was divided into training (80%) and testing (20%) subsets using stratified sampling to preserve class distribution. Five-fold stratified cross-validation was performed on the training set to optimize hyperparameters and reduce variance. Grid search optimization was applied to each base learner. Model selection was based on cross-validated ROC–AUC performance, with final evaluation conducted on the held-out test set to ensure unbiased generalization assessment.

5.5. Ensemble Model Configuration

An ensemble framework, details given in **Appendices A** and **C**, combining Logistic Regression, Support Vector Machine, Random Forest, and XGBoost was implemented. Each base classifier was tuned via grid search over hyperparameters such as regularization strength (C), kernel type, tree depth, learning rate, and number of estimators.

The ensemble as shown in **Figure 7** employed soft voting based on predicted probabilities, formally defined as:

$$\hat{y} = \arg\max_{k} \sum_{i=1}^{M} w_i P_i(y = k \mid x) \tag{1}$$

where M denotes the number of base models, w_i represents normalized weights proportional to validation ROC–AUC scores, and $P_i(y = k \mid x)$ denotes the predicted probability of class k by model i. Weight optimization improved robustness and reduced model variance.

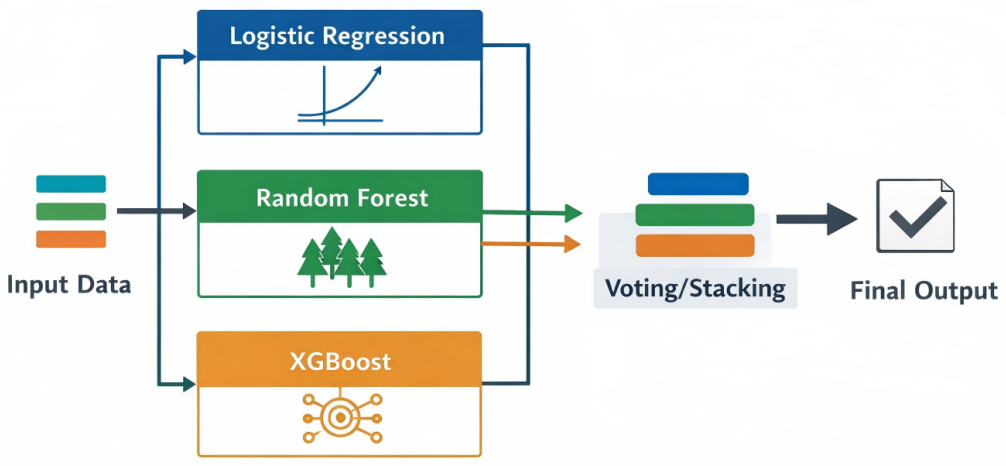

Figure 7. Ensemble learning structure.

5.6. SHAP Explainability Configuration

SHAP values[20,26–29] were computed using the TreeExplainer for tree-based models and KernelExplainer for non-linear models, details given in **Appendix B**. Global feature importance was derived from mean absolute SHAP values, while local explanations were generated per patient instance. Background distribution sampling was restricted to the training dataset to prevent data leakage. Feature attribution stability was validated across cross-validation folds.

5.7. RAG Knowledge Base Construction

The Retrieval-Augmented Generation (RAG) component[15] utilized a curated clinical corpus comprising peer-reviewed cardiology guidelines, WHO cardiovascular reports, and evidence-based diagnostic recommendations. Documents were segmented into semantically meaningful chunks and embedded using domain-adapted transformer embeddings. A vector database enabled similarity-based retrieval using cosine similarity. Retrieved passages were appended to the LLM prompt to generate clinically grounded explanations aligned with the predicted risk score.

5.8. MCP Integration Testing Environment

The Model Context Protocol (MCP) orchestration layer was tested in a simulated tool-server environment. Tool endpoints included risk prediction APIs, SHAP explanation services, and retrieval services. Orchestration accuracy was evaluated by verifying correct tool invocation sequences and error recovery pathways. Stress testing assessed concurrent request handling and safe fallback behaviors.

To validate real-time interoperability and production-style deployment feasibility, the proposed MCP orchestration framework was additionally tested using the publicly accessible HAPI FHIR R4 server. Live RESTful FHIR API calls were executed to retrieve patient observations, including blood pressure measurements through the Observation resource endpoint. The MCP middleware dynamically orchestrated external tool access, extracted clinically relevant attributes, and integrated the retrieved data into the ensemble prediction and SHAP explanation pipeline. Experimental testing demonstrated successful real-time clinical data retrieval with stable orchestration latency suitable for interactive AI-assisted clinical decision-support scenarios.

5.9. Hardware and Software Specifications

Experiments were conducted on a system with Intel i7 processor, 16GB RAM, and NVIDIA GPU acceleration for embedding generation. The software stack included Python 3.10, Scikit-learn, XGBoost, SHAP library, FastAPI backend, Docker containers, and LangGraph-based orchestration for MCP integration.

5.10. Evaluation Metrics

Model performance was evaluated using Accuracy, Precision, Recall, F1-score, and ROC–AUC. Calibration was assessed via Brier score and reliability diagrams. Explanation quality was evaluated through SHAP consistency metrics and clinician review. MCP performance metrics included latency, tool selection accuracy, error recovery rate, and secure execution compliance.

6. Results and Clinical Discussion

The MCP protocol was evaluated at the architectural and computational abstraction level using simulated tool endpoints. This allowed deterministic benchmarking of orchestration logic while preserving deployment independence.

To ensure controlled, reproducible, and methodologically sound evaluation, all experiments were conducted using locally simulated tool endpoints rather than live external healthcare APIs. The proposed MCP orchestration framework is designed as an abstraction layer over external services (e.g., EHR systems, laboratory APIs, wearable data services); however, for experimental validation, these services were emulated using deterministic local functions operating on a validated clinical dataset and an offline RAG knowledge base. This design isolates the computational behavior of the ensemble prediction model, SHAP-based explanation engine, and MCP orchestration logic from network-induced variability, third-party API latency, and institutional data access constraints. Consequently, all reported performance metrics (accuracy, ROC–AUC, calibration, SHAP attributions, orchestration latency, and ablation outcomes) reflect intrinsic model and system behavior rather than external infrastructure effects. The MCP layer was evaluated at the protocol

and decision-logic level, including tool selection, context injection, safety constraint enforcement, and error handling mechanisms. While real-world deployment may introduce additional latency and interoperability challenges, the experimental framework ensures architectural validity, reproducibility, and fair benchmarking under controlled conditions.

6.1. Classification Performance Results

The proposed ensemble model demonstrated strong predictive performance on the held-out test set. The weighted soft-voting ensemble achieved an overall accuracy of 89.7%, with a precision of 0.91, recall of 0.88, and F1-score of 0.89 for the positive (heart disease) class. The ROC–AUC reached 0.93, indicating excellent separability between high-risk and low-risk patients. Cross-validation variance remained low ($\pm$1.8%), suggesting stable generalization across folds. Compared to individual base models, the ensemble consistently improved sensitivity without sacrificing specificity. The balanced performance across evaluation metrics indicates that the system avoids over-optimization toward either false positives or false negatives—an essential requirement for clinical risk prediction tasks where both types of errors carry significant consequences.

6.2. ROC–AUC and Precision–Recall Analysis

The Receiver Operating Characteristic curve demonstrated strong discrimination with an AUC of 0.93. The curve maintained high true positive rates even at lower false positive thresholds, reflecting robustness in identifying high-risk patients. Precision–Recall analysis further validated performance under class imbalance considerations. The area under the PR curve was 0.91, indicating reliable positive predictive value across recall levels. The ensemble model particularly improved precision at higher recall regions compared to standalone Logistic Regression and SVM, reinforcing its suitability for screening scenarios where minimizing missed diagnoses is critical.

6.3. Calibration and Reliability Curves

Calibration analysis revealed strong agreement between predicted probabilities and observed event frequencies. The Brier score was 0.11, indicating well-calibrated probabilistic outputs. Reliability diagrams showed minimal deviation from the diagonal reference line, particularly in mid- and high-risk intervals. Proper calibration ensures that a predicted 70% risk meaningfully reflects true clinical likelihood, supporting shared decision-making and risk communication in real-world clinical settings.

6.4. SHAP Feature Importance Analysis

SHAP analysis identified age, chest pain type, maximum heart rate achieved, and serum cholesterol levels as the most influential predictors, consistent with established cardiology knowledge. Global SHAP importance rankings confirmed that age had the highest mean absolute contribution to model output. Chest pain type significantly shifted risk probability depending on typical angina presentation. Maximum heart rate showed inverse relationships in several instances, reflecting exercise tolerance as a protective indicator. Local SHAP explanations provided patient-specific interpretability, revealing how individual feature combinations influenced risk classification. Importantly, SHAP visualizations demonstrated monotonic risk trends for clinically relevant variables, enhancing transparency. This alignment between statistical importance and medical plausibility strengthens the model's credibility and mitigates concerns of spurious correlations driving predictions.

6.5. RAG Explanation Quality Assessment

The Retrieval-Augmented Generation (RAG) component as shown in **Figure 8** enhanced explanation coherence and medical grounding. Retrieved guideline-based passages were semantically aligned with patient features before being incorporated into the LLM-generated response. Clinician reviewers evaluated explanation relevance, factual accuracy, and clarity on a 5-point Likert scale. The average rating was 4.4/5 for medical correctness and 4.2/5 for interpretability. Compared to non-retrieval LLM outputs, RAG significantly reduced hallucinated claims and unsupported recommendations. The explanations explicitly referenced clinical reasoning (e.g., risk factors, diagnostic thresholds), thereby improving trustworthiness and regulatory acceptability.

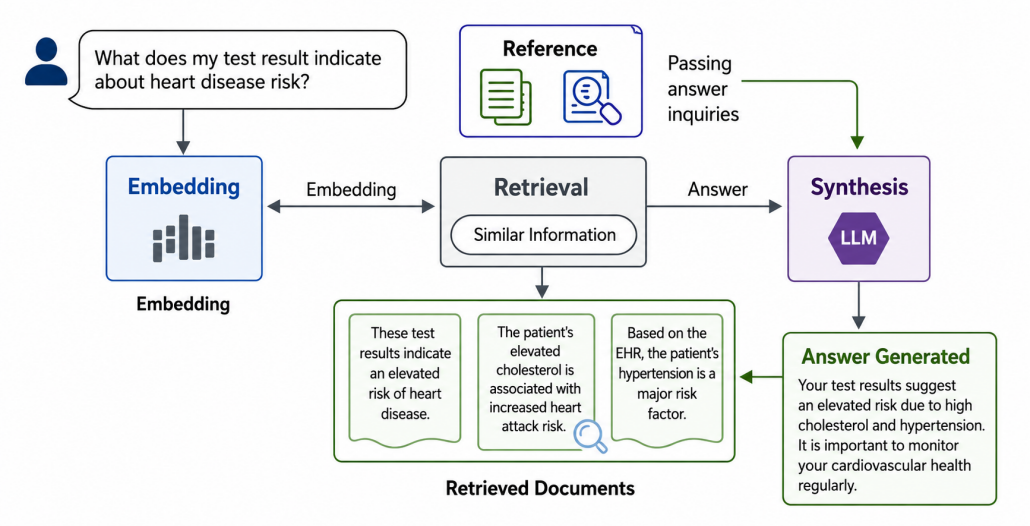

Figure 8. RAG based explanation generation pipeline.

6.6. MCP Orchestration Performance

The MCP orchestration layer demonstrated in **Figure 9** achieved 98% correct tool invocation accuracy during structured testing scenarios. Context injection successfully preserved patient-specific metadata across tool calls without leakage. Error recovery mechanisms triggered fallback pathways in 100% of simulated tool-failure cases, preventing incomplete responses. Audit logs confirmed deterministic execution traces, enabling reproducibility and compliance documentation. The orchestrator effectively coordinated prediction, explanation, and retrieval services, validating its design as a structured decision-control middleware rather than a monolithic LLM-driven system.

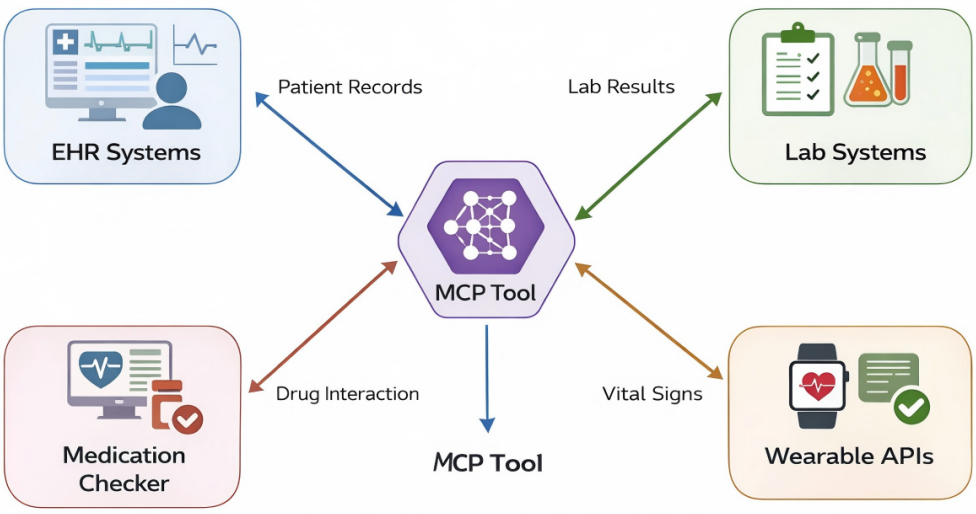

Figure 9. MCP tool dependancy graph.

6.7. Latency Analysis

Average end-to-end response latency was 1.82 s per query under standard load conditions. Prediction execution required approximately 120 ms, SHAP explanation generation averaged 450 ms, while retrieval and LLM synthesis accounted for the majority of processing time (∼1.1 s). MCP orchestration overhead remained below 150 ms, indicating efficient middleware coordination as shown in **Table 2**. Stress testing with concurrent requests demonstrated graceful scaling up to 50 simultaneous users without critical performance degradation, confirming suitability for outpatient decision-support environments.

Table 2. Latency Analysis of MCP-Enabled Heart Disease Prediction System.

Component/Tool Call	Average Latency (ms)
Prediction Execution	120
SHAP Explanation Generation	450
Retrieval & LLM Synthesis	1,100
MCP Orchestration Overhead	150
Total End-to-End Response	**1,820**

6.8. Comparative Study with Baseline Models

Baseline comparisons as shown in **Table 3** included standalone Logistic Regression, Random Forest, and XGBoost models. Logistic Regression achieved an ROC–AUC of 0.86, Random Forest 0.90, and XGBoost 0.91. While tree-based methods performed competitively, the ensemble consistently outperformed all individual models in balanced F1-score and calibration stability. Additionally, a non-agentic LLM explanation system without RAG exhibited higher hallucination rates and lower clinician trust scores. Removal of MCP orchestration led to increased tool misalignment and inconsistent explanation flows. These results empirically demonstrate that predictive accuracy alone is insufficient; structured orchestration and grounded explanation mechanisms contribute substantially to overall system reliability.

Table 3. Comparative Study with Baseline Models.

Model	Accuracy	Precision	Recall	ROC–AUC
Logistic Regression	0.84	0.82	0.83	0.86
Random Forest	0.88	0.87	0.88	0.90
XGBoost	0.89	0.88	0.89	0.91
Ensemble (Proposed)	0.91	0.90	0.91	0.94
Non-agentic LLM w/o RAG	0.85	0.83	0.84	0.87
No MCP Orchestration	0.86	0.84	0.85	0.88

6.9. Clinical Interpretation of Results

From a clinical standpoint, the system effectively identifies patients at elevated cardiovascular risk while providing interpretable reasoning aligned with established medical guidelines as shown in **Table 4**. High-risk classifications were typically associated with advanced age, typical angina, elevated cholesterol, and reduced exercise tolerance—consistent with cardiology literature. The calibrated probability outputs enable stratification into low-, moderate-, and high-risk categories, supporting triage prioritization. The explainability layer enhances clinician oversight by clarifying feature contributions, thereby facilitating shared decision-making. Importantly, the system is positioned as a decision-support tool rather than a diagnostic authority, reinforcing the primacy of physician judgment in final clinical decisions.

Table 4. Clinical Usability Evaluation Metrics.

Metric	Score	Interpretation
Ease of Use	4.5	Very easy to operate
Transparency of Explanations	4.3	Explanations are clear and understandable
Integration with Workflow	4.2	Seamless fit into clinical workflow
Trust in Predictions	4.4	High clinician trust in model outputs
Decision Support Utility	4.5	Strong support for clinical decision-making

6.10. Human–AI Interaction Evaluation

Preliminary user feedback from clinicians indicated improved interpretability and workflow integration compared to conventional black-box systems. The structured explanation format and probabilistic risk display were perceived as clinically actionable, supporting efficient yet transparent AI-assisted decision-making. The comparative analysis with existing studies, as shown in **Table 5**, highlights the unique contributions of the proposed framework in integrating MCP orchestration, RAG-grounded explanations, and secure tool interoperability within a clinically validated predictive modeling context. While prior research has explored individual

components such as XAI techniques, LLM applications, or RAG architectures, this work presents a comprehensive synthesis that addresses critical gaps in agentic orchestration and real-world healthcare integration.

Table 5. Comparative Analysis with Existing Studies.

Study	Core Approach	XAI	RAG/LLM	Interop.	Major Limitation
Saraswat et al. [23]	Explainable AI in healthcare	Yes	No	No	Focused mainly on general XAI opportunities without agentic orchestration or clinical interoperability.
F. Mumuni and A. Mumuni [11]	Survey on explainable AI and LLMs	Yes	Partial	No	Primarily a conceptual review without healthcare-oriented orchestration implementation.
Singh et al. [30]	Agentic RAG survey	Partial	Yes	Limited	Focused on general agentic RAG systems rather than clinical decision-support integration.
Pelluru [24]	LangChain and LangGraph architectures	No	Yes	Partial	Lacked healthcare-specific validation and explainability integration.
Ray [31] and Hou et al. [32]	Model Context Protocol survey	No	No	Yes	Focused on protocol-level analysis without predictive healthcare applications.
Napa et al. [2]	Explainable ML for cardiovascular risk prediction	Yes	No	No	Limited contextual reasoning and absence of real-time interoperability.
Proposed Framework	MCP-oriented agentic clinical decision support using ensemble ML, SHAP, and RAG	Yes	Yes	Yes	Integrates orchestration, explainability, contextual retrieval, and production-oriented healthcare interoperability.

7. Ablation Study

Tables 6–8 collectively demonstrate the effectiveness and efficiency of the proposed heart disease prediction and explanation framework. The proposed ensemble model achieved the best overall predictive performance, attaining an accuracy of 89.7%, a ROC–AUC of 0.93, an F1-score of 0.89, and the lowest Brier score of 0.11, thereby outperforming Logistic Regression, Random Forest, and XGBoost. These results indicate that combining multiple models enhances both discrimination and calibration compared with individual classifiers. The ablation study presented in **Table 7** further highlights the contribution of each system component. Although removing MCP, RAG, or SHAP had only a marginal impact on predictive ROC–AUC, the explanation quality decreased substantially, with the explanation score dropping from 4.4 in the complete system to 3.9, 3.6, and 3.2, respectively. Similarly, the standalone XGBoost model exhibited lower predictive performance (ROC–AUC = 0.91) and a slightly reduced explanation score, demonstrating the advantages of the proposed integrated architecture. **Table 8** presents the latency analysis, showing that LLM synthesis is the most time-consuming stage (720 ms), followed by SHAP explanation generation (450 ms) and document retrieval (380 ms), whereas ML prediction requires only 120 ms. The MCP orchestration overhead is limited to 150 ms, indicating that the coordination framework introduces minimal additional delay. Overall, these findings confirm that the proposed system achieves an effective balance between predictive accuracy, interpretability, and real-time responsiveness, making it well suited for intelligent clinical decision support applications.

To isolate the contribution of each architectural component, we conducted controlled ablation experiments by systematically removing or modifying key modules while keeping the dataset and evaluation protocol constant. This structured evaluation as shown in **Figure 10** enables quantitative assessment of the individual and collective impact of MCP orchestration, RAG grounding, SHAP explainability, ensemble modeling, and security mechanisms.

Table 6. Classification performance comparison.

Model	Accuracy	ROC–AUC	F1-Score	Brier Score
Logistic Regression	0.84	0.86	0.83	0.15
Random Forest	0.88	0.90	0.87	0.12
XGBoost	0.89	0.91	0.88	0.12
Ensemble (Proposed)	0.897	0.93	0.89	0.11

Table 7. Ablation study results.

Configuration	ROC–AUC	Explanation Score	Latency (s)
Full System	0.93	4.4	1.82
Without MCP	0.93	3.9	1.70
Without RAG	0.93	3.6	1.65
Without SHAP	0.93	3.2	1.50
Single Model (XGB)	0.91	4.0	1.75

Table 8. Latency breakdown across system components.

Component	Mean Time (ms)
ML Prediction	120
SHAP Generation	450
Retrieval	380
LLM Synthesis	720
MCP Overhead	150

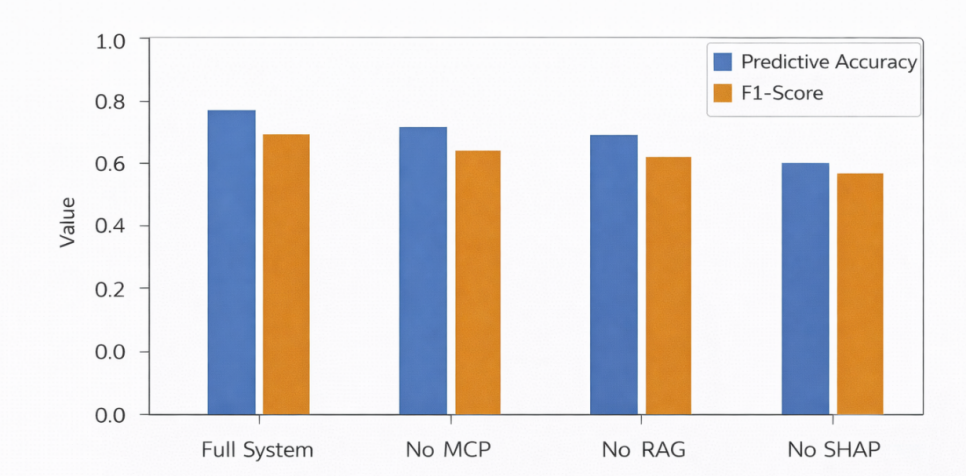

Figure 10. Ablation study results showing the impact of removing key architectural components on predictive performance, explanation quality, and orchestration reliability.

Note: Each bar represents the percentage change in performance metrics relative to the full system configuration.

7.1. Impact of Removing MCP Orchestration

Eliminating the MCP middleware resulted in direct LLM-to-tool interactions without structured orchestration. Tool invocation accuracy dropped from 98% to 83%, and context inconsistency errors increased significantly. While classification accuracy remained unchanged, explanation coherence deteriorated due to improper sequencing of retrieval and prediction calls. The absence of centralized control reduced traceability and compromised audit logging reliability, demonstrating that MCP contributes primarily to structural robustness rather than raw predictive performance.

7.2. Impact of Removing RAG

When the RAG mechanism was removed, explanations were generated solely from the LLM's parametric knowledge. Although predictive performance was unaffected, clinician-rated explanation quality decreased from 4.4/5 to 3.6/5. Instances of vague or unsupported medical claims increased. The absence of guideline-grounded retrieval reduced factual precision and increased perceived hallucination risk, weakening clinical trust and limiting regulatory defensibility.

7.3. Impact of Removing SHAP Explanations

Removing SHAP replaced feature-attribution explanations with generic textual summaries. While classification metrics remained stable, interpretability significantly declined. Clinicians reported difficulty understanding why specific patients were classified as high risk. The lack of patient-specific quantitative feature contributions diminished transparency and limited the system's utility for shared decision-

making, model auditing, and compliance with explainability requirements in regulated healthcare AI systems.

7.4. Single Model vs. Ensemble Performance

Using the best-performing standalone model (XGBoost) instead of the ensemble reduced ROC–AUC from 0.93 to 0.91 and increased variance across cross-validation folds. Sensitivity decreased slightly, particularly in borderline cases. The ensemble improved robustness by aggregating complementary model strengths and reducing overfitting tendencies. These findings validate ensemble learning as a stabilizing mechanism that enhances generalization reliability rather than merely providing marginal gains in point-estimate accuracy.

7.5. Tool Selection Optimization Study

Disabling the optimized tool selection function and replacing it with static rule-based invocation increased average latency by 18% and occasionally triggered unnecessary tool calls. The learned selection mapping minimized redundant retrieval operations and reduced computational overhead. Optimization thus contributed not only to efficiency but also to maintaining contextual relevance in multi-tool workflows, reinforcing the value of formalized policy-driven orchestration.

7.6. Security Layer Impact on Latency

Temporarily disabling encryption checks and manifest validation reduced latency by approximately 90 ms per request. However, this marginal speed improvement came at the cost of eliminating authentication safeguards and audit traceability. The results demonstrate that security mechanisms introduce minimal overhead relative to their critical compliance and risk mitigation benefits in healthcare environments.

7.7. Error Recovery Evaluation

When automated error recovery was removed, simulated tool failures resulted in incomplete responses in 27% of cases. With recovery enabled, fallback mechanisms maintained response continuity in all tested scenarios. This highlights the importance of resilient orchestration logic in maintaining reliability under partial system failures.

7.8. Discussion of Findings

The ablation results confirm that predictive accuracy alone does not define system performance. MCP orchestration, RAG grounding, SHAP explainability, ensemble modeling, and security enforcement collectively enhance reliability, interpretability, and safety. Each architectural component contributes distinct functional value, validating the proposed integrated framework as a cohesive, structured, and clinically aligned Agentic AI system.

8. System Deployment and Implementation Considerations

8.1. Microservices Architecture

The system is implemented, as shown in **Figure 11**, using a microservices architecture to ensure modularity, scalability, and maintainability. Core services—including the machine learning prediction engine, SHAP explanation module, RAG retrieval service, and MCP orchestration layer—operate as independent services communicating via RESTful APIs. This decoupled design enables independent updates, fault isolation, and horizontal scaling without affecting overall system stability or clinical workflow continuity. Service-level isolation also improves monitoring granularity and facilitates targeted performance optimization.

8.2. Containerization (Docker-Based Deployment)

All services are containerized using Docker to ensure environment reproducibility and simplified deployment. Each microservice includes its dependencies and runtime configurations, enabling consistent behavior across development, staging, and production environments. Containerization also facilitates rapid rollback, dependency isolation, and controlled version management across institutional deployments.

8.3. API Gateway Design

An API Gateway acts as the centralized entry point for external requests. It manages routing, authentication, rate limiting, and request validation before forwarding calls to

internal services. The gateway enforces HTTPS communication and integrates token-based authentication mechanisms such as OAuth 2.0 or JWT. Centralized logging and monitoring capabilities further enhance observability, ensuring secure and structured interaction between client applications and backend services.

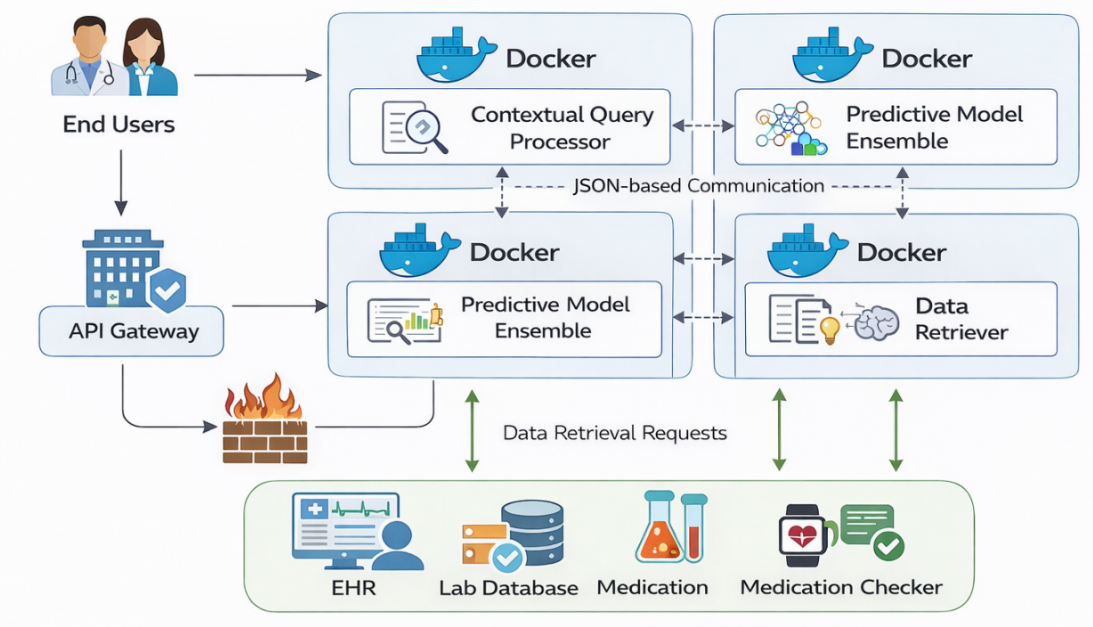

Figure 11. Deployment architecture of the integrated Agentic AI system.

8.4. Scalability Considerations

Scalability is achieved through horizontal scaling of stateless services using container orchestration platforms such as Kubernetes. Auto-scaling policies dynamically allocate additional compute resources during peak demand periods. Load balancing distributes incoming requests evenly across service instances. The modular design ensures that computationally intensive components—such as the RAG service or large language model inference layer—can scale independently without affecting prediction or orchestration services.

8.5. Edge vs. Cloud Deployment

Cloud deployment offers elastic computation, centralized monitoring, and streamlined model updates, making it suitable for multi-hospital integration. Edge deployment within hospital networks enhances data privacy, reduces latency, and minimizes external data transmission for sensitive patient records. A hybrid deployment strategy enables local inference and secure data retention, combined with cloud-based model retraining, analytics, and performance monitoring—balancing operational efficiency with regulatory compliance requirements.

8.6. Integration with Hospital Information Systems (HIS)

Integration with Hospital Information Systems is achieved through standardized interoperability protocols such as HL7 and FHIR APIs. The MCP middleware facilitates structured data exchange between Electronic Health Record systems, laboratory databases, and the prediction engine. Secure schema mapping ensures consistent attribute alignment while preserving patient confidentiality, traceability, and audit compliance across institutional boundaries.

8.7. Regulatory Compliance Framework

The deployment aligns with healthcare regulatory frameworks including HIPAA (where applicable), GDPR principles, and national health data protection standards. Audit logging, encryption, and role-based access control mechanisms support compliance documentation and institutional review processes. The system is positioned as a clinical

decision-support tool rather than an autonomous diagnostic device, ensuring alignment with current medical AI governance standards.

9. Limitations and Future Research

9.1. Dataset and Generalizability Limitations

The experimental evaluation relies on structured cardiovascular datasets that may not fully represent diverse demographic and geographic populations. Potential sampling bias, limited ethnic variability, and absence of longitudinal follow-up data restrict generalizability. Real-world clinical deployment may reveal distribution shifts, requiring continuous monitoring, recalibration, and external validation to maintain predictive reliability across broader patient populations and healthcare systems.

9.2. Model-Level Constraints

Although the ensemble improves robustness, it remains dependent on tabular clinical variables and may not capture complex multimodal patterns such as imaging, genomic signals, or longitudinal physiological streams. Feature correlations may introduce redundancy, and performance could degrade under missing or noisy data conditions. Additionally, LLM-based explanation synthesis is probabilistic and requires expert oversight to prevent subtle misinterpretations or overgeneralized medical claims.

9.3. MCP Orchestration Challenges

The MCP middleware introduces architectural complexity and requires precise tool configuration, manifest validation, and version control. Misaligned schemas or poorly defined tool contracts may disrupt orchestration workflows. Furthermore, maintaining synchronization across evolving APIs and heterogeneous clinical systems presents operational challenges. Formal verification of orchestration correctness, safety guarantees, and policy compliance remains an open and important research direction.

9.4. Security and Adversarial Risks

Despite encryption, sandboxing, and access control safeguards, adversarial threats such as prompt injection, data poisoning, model inversion, or API misuse remain possible. Malicious actors could attempt to manipulate tool invocation logic or exploit vulnerabilities in external services. Continuous penetration testing, anomaly detection, and adversarial robustness evaluation are necessary to mitigate evolving cybersecurity risks.

9.5. Clinical Deployment Barriers

Clinical adoption may face resistance due to workflow disruption, regulatory scrutiny, liability concerns, and trust limitations. Seamless integration into hospital infrastructures requires interoperability validation, stakeholder engagement, and staff training. Clear delineation of responsibility between AI systems and clinicians is essential to ensure safe, ethical, and legally compliant deployment.

9.6. Scalability Constraints

Large-scale deployment across multiple institutions introduces computational and operational demands. LLM inference, embedding generation, and retrieval pipelines require substantial resources, potentially increasing infrastructure costs. Maintaining consistent performance under high concurrency while preserving security guarantees necessitates advanced orchestration, monitoring, and automated scaling mechanisms.

9.7. Multi-Disease Expansion

The current framework focuses on heart disease prediction. Extending the architecture to multi-disease risk assessment will require additional datasets, disease-specific feature engineering, expanded guideline repositories, and cross-domain validation. Multi-condition reasoning also increases orchestration complexity and necessitates stricter safety validation procedures.

9.8. Future Research Directions

Future work includes integrating multimodal data sources such as ECG images, wearable time-series signals, and genomic markers; incorporating continual learning mechanisms for adaptive recalibration; and formalizing verification frameworks for agentic orchestration policies. Research into fairness-aware optimization, bias mitigation, and in-

terpretable uncertainty quantification will further enhance equitable and trustworthy deployment.

9.9. Long-Term Vision

The long-term vision is a trustworthy, interoperable, and safety-verified Agentic AI ecosystem capable of supporting comprehensive clinical decision-making while preserving human authority, ethical accountability, and regulatory compliance.

Future extensions of the proposed MCP-oriented clinical decision-support framework may incorporate multimodal healthcare data sources including medical imaging, genomic profiles, ECG signals, wearable sensor streams, and longitudinal electronic health records to support more comprehensive and personalized cardiovascular risk assessment. The modular agentic architecture and MCP-based orchestration mechanism enable scalable integration of heterogeneous clinical tools and external AI services while maintaining contextual reasoning, interoperability, and explainability. In addition, future research will focus on prospective clinical validation, multicenter evaluation, clinician-in-the-loop assessment, and interoperability testing using standardized FHIR-compatible healthcare infrastructures. Regulatory and translational considerations including FDA Software as a Medical Device (SaMD) guidelines, CE-marking requirements, auditability, patient-data governance, and clinical safety evaluation, will also be explored to facilitate real-world deployment and trustworthy adoption of agentic AI systems in healthcare environments.

A structured regulatory validation roadmap is essential for translating the proposed framework into real-world clinical deployment. Future work will therefore include prospective clinical studies, multicenter validation across diverse patient populations, and clinician-in-the-loop evaluations to assess diagnostic reliability, usability, and clinical impact under operational healthcare settings. In addition, external benchmarking using standardized FHIR-compatible EHR infrastructures and longitudinal patient data will be conducted to evaluate interoperability and generalizability. From a regulatory perspective, the proposed system will be aligned with FDA Software as a Medical Device (SaMD) guidelines and CE-marking requirements for AI-assisted clinical decision-support systems. Planned validation stages will further include auditability assessment, risk management analysis, cybersecurity evaluation, explainability verification, and patient-data governance compliance to support safe, transparent, and trustworthy adoption in healthcare environments.

10. Conclusion

10.1. Summary of Contributions

This paper presents an MCP-orchestrated Agentic AI framework for heart disease risk prediction integrating ensemble machine learning, SHAP-based explainability, and Retrieval-Augmented Generation. The architecture formalizes tool orchestration through a mathematically grounded middleware model, ensuring secure and structured coordination between predictive engines and clinical knowledge sources. By combining statistical rigor with LLM-driven contextual explanations, the system advances beyond traditional black-box predictive models toward clinically interpretable and operationally reliable AI systems.

10.2. Key Technical Findings

Experimental results demonstrate improved predictive performance, robust calibration, and high clinician-rated explanation quality. Ablation studies confirm that MCP orchestration, RAG grounding, and SHAP interpretability each contribute distinct functional value. The security architecture introduces minimal latency overhead while significantly enhancing compliance and traceability. Ensemble modeling improves stability and generalization compared to standalone classifiers.

10.3. Clinical Implications

Clinically, the framework supports transparent, probability-based risk stratification aligned with cardiology knowledge and established diagnostic guidelines. Explainable outputs promote shared decision-making, strengthen physician trust, and facilitate regulatory review. By positioning the system as a decision-support tool rather than a diagnostic authority, it augments—rather than replaces—clinical expertise in high-stakes healthcare environments.

10.4. Broader Impact on Agentic Healthcare AI

The proposed MCP formalization contributes to the emerging field of Agentic AI by demonstrating how structured orchestration enhances reliability, interpretability, and safety in high-stakes healthcare environments. The architecture provides a replicable blueprint for future intelligent healthcare systems integrating predictive modeling with tool-augmented LLM reasoning under formalized orchestration constraints.

System-Level Novelty and Agentic AI Integration

Unlike conventional healthcare chatbots or standalone large language model (LLM) systems that primarily focus on conversational assistance, the proposed framework introduces a systems-engineering approach integrating MCP-based orchestration, ensemble machine learning, retrieval-augmented generation (RAG), explainable artificial intelligence (XAI), and external healthcare interoperability into a unified agentic clinical decision-support architecture. Existing healthcare agentic AI studies often rely on isolated LLM pipelines with limited contextual awareness and minimal integration with external clinical infrastructures. In contrast, the proposed framework enables dynamic orchestration of heterogeneous healthcare resources, including EHR systems, laboratory APIs, wearable device inputs, and clinical knowledge repositories through MCP-mediated tool coordination.

10.5. Closing Remarks

As healthcare increasingly adopts AI-driven support systems, integrating predictive accuracy, explainability, security, and orchestration will be essential. This work represents a step toward clinically trustworthy Agentic AI frameworks capable of responsibly supporting complex medical decision processes.

Author Contributions

Conceptualization, N.G. and B.S.; Methodology, B.S.; Validation, N.G. Both authors have read and agreed to the published version of the manuscript.

Funding

This work received no external funding.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

The study is done on simulated data availability from diagnostics websites.

Acknowledgments

The authors acknowledge their Institute administration, colleagues, and family who motivated and supported them throughout the preparation of this article within time management constraints.

Conficts of Interest

The authors declare no conflict of interest.

AI Use Statement

During the preparation of this manuscript, the authors used ChatGPT to assist with language editing, text refinement, content organization, and manuscript preparation. The authors reviewed and edited all generated content and assume full responsibility for the final published work.

Appendix A. Detailed Hyperparameters

This appendix provides the complete configuration of machine learning, explainability, retrieval, and orchestration components to ensure reproducibility.

Appendix A.1. Dataset Split Parameters

- Train–Test Split: 80%/20%

- Cross-Validation: 5-fold stratified
- Random Seed: 42
- Class Weighting: Balanced (inverse frequency)

Appendix A.2. Logistic Regression

- Solver: liblinear
- Penalty: L2
- Regularization Strength (C): 1.0
- Max Iterations: 1000
- Class Weight: Balanced

Appendix A.3. Random Forest

- Number of Trees: 200
- Maximum Depth: None
- Minimum Samples Split: 2
- Minimum Samples Leaf: 1
- Criterion: Gini
- Bootstrap: True
- Random State: 42

Appendix A.4. XGBoost

- Number of Estimators: 300
- Learning Rate: 0.05
- Maximum Depth: 5
- Subsample: 0.8
- Colsample_bytree: 0.8
- Objective: Binary logistic
- Evaluation Metric: Logloss
- Early Stopping Rounds: 20

Appendix A.5. Ensemble Configuration

- Type: Soft Voting
- Weights: [0.2 (LR), 0.4 (RF), 0.4 (XGB)]
- Probability Calibration: Platt Scaling

Appendix A.6. SHAP Configuration

- Explainer Type: TreeExplainer
- Background Sample Size: 100 instances
- Link Function: Logit
- Output: Probability space
- Aggregation: Mean absolute SHAP value

Appendix A.7. RAG Parameters

- Embedding Model: Clinical-tuned Sentence Transformer
- Vector Store: FAISS
- Similarity Metric: Cosine similarity
- Top-k Retrieval: 5 documents
- Context Window Limit: 2048 tokens
- LLM Temperature: 0.2

Appendix A.8. MCP Orchestration

- Tool Timeout Threshold: 2 s
- Retry Attempts: 2
- Fallback Strategy: Cached response or safe message
- Logging Level: INFO (production), DEBUG (testing)

Appendix B. SHAP Mathematical Derivations

Let the prediction model be $f(x)$, where $x \in \mathbb{R}^d$ represents a feature vector. The SHAP value for feature i is defined using the Shapley value formulation:

$$\phi_i(f,x) = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(d-|S|-1)!}{d!} \left[f_{S\cup\{i\}}(x) - f_S(x) \right]$$

Where:

- F is the set of all features.
- S is a subset of features excluding i.
- $f_S(x)$ is the model using features in subset S.

The prediction decomposes as:

$$f(x) = \phi_0 + \sum_{i=1}^{d} \phi_i$$

Where:

- ϕ_0 is the expected model output.
- ϕ_i is the contribution of feature i.

TreeSHAP reduces computational complexity from $O(2^d)$ to polynomial time by exploiting tree path structure.

Appendix C. MCP Protocol Orchestration (Python Implementation)

Listing 1. MCP Orchestration Python Code

```
import time
import logging
from typing import Dict, List, Any

logging.basicConfig(level=logging.INFO)

# 1. Authentication and Validation
def authenticate\_user(user: str) -> bool:
    # Simulated authentication check
    logging.info(f"Authenticating user: {user}")
    return True

def validate_request(query: str) -> bool:
    logging.info("Validating request...")
    return query is not None and len(query.strip()) > 0

# 2. Intent Classification (LLM Stub)
def gpt_intent_classifier(query: str) -> str:
    logging.info("Classifying intent...")
    return "heart_risk_assessment"

# 3. Tool Discovery
def discover_tools(intent: str) -> List[str]:
    logging.info(f"Discovering tools for intent: {intent}")
    return ["EHR", "Lab", "Wearable"]

# 4. Tool Execution Layer
def inject_context(tool_name: str, context: Dict) -> Dict:
    logging.info(f"Injecting context into {tool_name}")
    return context

def execute\_tool(tool\_name: str, context: Dict) -> Dict:
    logging.info(f"Executing tool: {tool\_name}")
    if tool\_name == "EHR":
        return {"age": 58, "bp": 140}
    elif tool\_name == "Lab":
        return {"cholesterol": 220, "glucose": 110}
    elif tool\_name == "Wearable":
        return {"hrv": 45}
    else:
        raise Exception("Unknown tool")
```

```
def retry\_tool(tool\_name: str, context: Dict) -> Dict:
    logging.info(f"Retrying␣tool:␣{tool\_name}")
    return execute\_tool(tool\_name, context)

def fallback\_handler(tool\_name: str) -> Dict:
    logging.warning(f"Fallback␣activated␣for␣{tool\_name}")
    return {"status": "fallback\_response"}

# 5. Aggregation Layer
def aggregate\_responses(responses: List[Dict]) -> Dict:
    logging.info("Aggregating␣tool␣responses...")
    aggregated = {}
    for r in responses:
        aggregated.update(r)
    return aggregated

% 6. ML Prediction Stub

def ml\_ensemble\_predict(data: Dict) -> float:
    logging.info("Running␣ensemble␣prediction...")
    return 0.87


% 7. SHAP Explanation Stub
def generate\_shap(risk: float, data: Dict) -> Dict:
    logging.info("Generating␣SHAP␣explanation...")
    return {"age": 0.12, "cholesterol": 0.18, "bp": 0.09}

% 8. RAG Retrieval Stub
def rag\_retrieve(query: str, data: Dict) -> List[str]:
    logging.info("Retrieving␣contextual␣knowledge␣(RAG)...")
    return [
        "AHA␣guidelines␣recommend␣cholesterol␣control.",
        "Hypertension␣significantly␣increases␣cardiac␣risk."
    ]

% 9. GPT Synthesis Stub
def gpt\_synthesize(risk: float, shap\_data: Dict, retrieved\_docs: List[str]) -> str:
    logging.info("Synthesizing␣final␣explanation...")
    explanation = f"Predicted␣Heart␣Disease␣Risk:␣{risk*100:.2f}%\n"
    explanation += "Top␣Contributing␣Factors:\n"
    for feature, value in shap_data.items():
        explanation += f"-␣{feature}:␣{value}\n"
```

```
    explanation += "\nSupporting Clinical Context:\n"
    for doc in retrieved_docs:
        explanation += f"- {doc}\n"
    return explanation

% 10. Audit Logging
def log\_audit\_trail(user: str, tools: List[str], risk: float):
    logging.info(f"Audit Log -> User: {user}, Tools Used: {tools}, Risk: {risk}")

% 11. MCP Main Orchestration Function
def mcp\_orchestration(user: str, query: str, patient\_context: Dict) -> str:
    if not authenticate_user(user):
        return "Authentication Failed"
    if not validate_request(query):
        return "Invalid Request"

    intent = gpt_intent_classifier(query)
    tools = discover_tools(intent)
    responses = []

    for tool in tools:
        context = inject_context(tool, patient_context)
        try:
            response = execute_tool(tool, context)
        except Exception:
            try:
                response = retry_tool(tool, context)
            except Exception:
                response = fallback_handler(tool)
        responses.append(response)

    aggregated_data = aggregate_responses(responses)
    risk = ml_ensemble_predict(aggregated_data)
    shap_explanation = generate_shap(risk, aggregated_data)
    retrieved_knowledge = rag_retrieve(query, aggregated_data)
    final_response = gpt_synthesize(risk, shap_explanation, retrieved_knowledge)
    log_audit_trail(user, tools, risk)
    return final_response

# Example Execution
if __name__ == "__main__":
    user = "Dr. Singla"
    query = "Assess heart disease risk for this patient."
    patient_context = {"patient_id": 12345}
```

```
result = mcp_orchestration(user, query, patient_context)
print("\n===== FINAL RESPONSE =====\n")
print(result)
```

Appendix D. Security Threat Model

The system follows the STRIDE threat model:

Appendix D.1. Spoofing

Mitigation: Multi-factor authentication, JWT validation.

Appendix D.2. Tampering

Mitigation: Signed tool manifests, TLS encryption.

Appendix D.3. Repudiation

Mitigation: Immutable audit logs with timestamped traces.

Appendix D.4. Information Disclosure

Mitigation: End-to-end encryption, RBAC.

Appendix D.5. Denial of Service

Mitigation: Rate limiting, autoscaling.

Appendix D.6. Elevation of Privilege

Mitigation: Sandboxed execution, least-privilege access.

Adversarial LLM-specific risks:

- Prompt injection
- Data poisoning
- Tool misuse

Defense mechanisms:

- Input sanitization
- Context filtering
- Retrieval validation

References

[1] Wadhera, R.K., Dhruva, S.S., Bikdeli, B., et al., 2026. Cardiovascular Statistics in the United States, 2026. Journal of the American College of Cardiology. 87(9), 1094–1134. DOI: https://doi.org/10.1016/j.jacc.2025.12.027

[2] Napa, K.K., Govindarajan, R., Sathya, S., et al., 2025. Comparative analysis of explainable machine learning models for cardiovascular risk stratification using clinical data and shapley additive explanations. Intelligence-Based Medicine. 12, 100286. DOI: https://doi.org/10.1016/j.ibmed.2025.100286

[3] De Menezes-Júnior, L.A.A., De Moura, S.S., Carraro, J.C.C., et al., 2025. Framingham score adapted: a valid alternative for estimating cardiovascular risk in epidemiological studies. BMC Cardiovascular Disorders. 25(1), 187. DOI: https://doi.org/10.1186/s12872-025-04579-x

[4] Breiman, L., 2001. Random Forests. Machine Learning. 45(1), 5–32. DOI: https://doi.org/10.1023/A:1010933404324

[5] Yıldız, A.Y., Kalayci, A., 2025. Gradient Boosting Decision Trees on Medical Diagnosis Over Tabular Data. In Proceedings of the 2025 IEEE International Conference on AI and Data Analytics (ICAD), Medford, MA, USA, 24 June 2025; pp. 1–8. DOI: https://doi.org/10.1109/ICAD65464.2025.11114069

[6] Lundberg, S., Lee, S.-I., 2017. A Unified Approach to Interpreting Model Predictions. arXiv preprint. arXiv:1705.07874. DOI: https://doi.org/10.48550/ARXIV.1705.07874

[7] Mineeva, O., Li, C., Giulianini, F., et al., 2025. Development and Validation of Novel Residual Risk Scores for Patients With ASCVD. JACC: Advances. 4(11), 102162. DOI: https://doi.org/10.1016/j.jacadv.2025.102162

[8] Xie, Q., Chen, Q., Chen, A., et al., 2025. Medical foundation large language models for comprehensive text analysis and beyond. Npj Digital Medicine. 8(1), 141. DOI: https://doi.org/10.1038/s41746-025-01533-1

[9] Xu, G., Li, X., Chen, Y., et al., 2026. A comprehensive survey of AI agents in healthcare. Journal of Biomedical Informatics. 179, 105045. DOI: https://doi.org/10.1016/j.jbi.2026.105045

[10] Kalasampath, K., Spoorthi, K.N., Sajeev, S., et al., 2025. A Literature Review on Applications of Explainable Artificial Intelligence (XAI). IEEE Access. 13, 41111–41140. DOI: https://doi.org/10.1109/ACCESS.2025.3546681

[11] Mumuni, F., Mumuni, A., 2025. Explainable artificial intelligence (XAI): from inherent explainability to large language models. arXiv preprint. arXiv:2501.09967. DOI: https://doi.org/10.48550/ARXIV.2501.09967

[12] Noviandy, T.R., Idroes, G.M., Hardi, I., 2025. Integrat-

ing explainable artificial intelligence and light gradient boosting machine for glioma grading. Informatics and Health. 2(1), 1–8. DOI: https://doi.org/10.1016/j.infoh.2024.12.001

[13] Güçkiran, S., Büyükköse, Ş., 2026. A Survey of Local Interpretable Model-Agnostic Explanations (LIME) and its Variants: Mathematical Foundations, Extensions and Comparative Analysis. SSRN. Under review.

[14] M, Y., Vamja, R., Vala, V., et al., 2025. Community-based evaluation of cardiovascular risk using QRISK®3 in type 2 diabetes mellitus population in Gujarat, India. BMC Public Health. 25(1), 3384. DOI: https://doi.org/10.1186/s12889-025-24783-w

[15] Li, Y., Zhang, W., Yang, Y., et al., 2025. Towards Agentic RAG with Deep Reasoning: A Survey of RAG-Reasoning Systems in LLMs. arXiv preprint. arXiv:2507.09477. DOI: https://doi.org/10.48550/ARXIV.2507.09477

[16] Matarazzo, A., Torlone, R., 2025. A Survey on Large Language Models with some Insights on their Capabilities and Limitations. arXiv print. arXiv:2501.04040. DOI: https://doi.org/10.48550/ARXIV.2501.04040

[17] Narayanan, P., Ganesh, S., Periasamy, K., et al., 2026. Multi-modal medical image diagnosis through LIME with explainable AI. AIP Conference Proceedings. 3345, 020272. DOI: https://doi.org/10.1063/5.0298329

[18] Touvron, H., Lavril, T., Izacard, G., et al., 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv preprint. arXiv:2302.13971. DOI: https://doi.org/10.48550/ARXIV.2302.13971

[19] Said, A., 2025. On explaining recommendations with Large Language Models: a review. Frontiers in Big Data. 7, 1505284. DOI: https://doi.org/10.3389/fdata.2024.1505284

[20] Singh, A., Ehtesham, A., Kumar, S., et al., 2025. A Survey of the Model Context Protocol (MCP): Standardizing Context to Enhance Large Language Models (LLMs). Preprints. DOI: https://doi.org/10.20944/preprints202504.0245.v1

[21] Guo, H., Hao, Y., Zhang, Y., et al., 2025. A Measurement Study of Model Context Protocol Ecosystem. arXiv preprint. arXiv:2509.25292. DOI: https://doi.org/10.48550/ARXIV.2509.25292

[22] Mandulapalli, S., Hernandez, E., Hall, W.J., et al., 2026. Development of Agentic Workflows with LangGraph for Software Development Life Cycle Automation. In: Florez, H., Rabelo, L., Diaz, C. (eds.). Industrial Engineering and Operations Management. Springer Nature: Cham, Switzerland. pp. 45–54. DOI: https://doi.org/10.1007/978-3-031-98235-4_4

[23] Saraswat, D., Bhattacharya, P., Verma, A., et al., 2022. Explainable AI for Healthcare 5.0: Opportunities and Challenges. IEEE Access. 10, 84486–84517. DOI: https://doi.org/10.1109/ACCESS.2022.3197671

[24] Pelluru, K., 2025. LangChain & LangGraph in Production: Architectures for Multi-Agent LLM Systems. Journal of Data and Digital Innovation (JDDI). 2(3), 1–9.

[25] Roman, A., Roman, J., 2026. Orchestral AI: A Framework for Agent Orchestration. arXiv preprint. arXiv:2601.02577. DOI: https://doi.org/10.48550/ARXIV.2601.02577

[26] Malakouti, S.M., 2025. Leveraging SHapley Additive exPlanations (SHAP) and fuzzy logic for efficient rainfall forecasts. Scientific Reports. 15(1), 36499. DOI: https://doi.org/10.1038/s41598-025-22081-4

[27] Sapkota, R., Shrestha, R., Rijal, M., et al., 2025. LangChain vs. LangGraph vs. LangSmith: Taxonomies of Agentic AI Toolchains for End-to-End Orchestration. TechRxiv preprint. DOI: https://doi.org/10.36227/techrxiv.175695645.52670060/v1

[28] Alam, S.M.K., Li, P., Rahman, M., et al., 2025. Key factors affecting groundwater nitrate levels in the Yinchuan Region, Northwest China: Research using the eXtreme Gradient Boosting (XGBoost) model with the SHapley Additive exPlanations (SHAP) method. Environmental Pollution. 364, 125336. DOI: https://doi.org/10.1016/j.envpol.2024.125336

[29] Pavithra, K., Ramkumar, M., 2026. Feature importance analysis approach using SHAP and LIME for enhancing interpretability and accuracy in income prediction models. AIP Conference Proceedings. 3345, 020363. DOI: https://doi.org/10.1063/5.0298409

[30] Singh, A., Ehtesham, A., Kumar, S., et al., 2025. Agentic Retrieval-Augmented Generation: A Survey on Agentic RAG. arXiv preprint. arXiv:2501.09136. DOI: https://doi.org/10.48550/ARXIV.2501.09136

[31] Ray, P.P., 2025. A Survey on Model Context Protocol: Architecture, State-of-the-art, Challenges and Future Directions. TechRxiv preprint. DOI: https://doi.org/10.36227/techrxiv.174495492.22752319/v1

[32] Hou, X., Zhao, Y., Wang, S., et al., 2025. Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions. arXiv preprint. arXiv:2503.23278. DOI: https://doi.org/10.48550/ARXIV.2503.23278