WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

MediFlow: MCP-Based Healthcare Workflow Orchestration System

Shivakumara Naik[1]*, Veeresh Devadhar[1], Prajwal Horatti [1], Vishnu K V[1], Sumana C[2]

[1] Undergraduate Student, Department of Computer Science and Engineering, G. M. Institute of Technology, Davangere, Karnataka, India - 577006 (affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India)

[2] Tutor, Department of Computer Science and Engineering, G. M. Institute of Technology, Davangere, Karnataka, India - 577006

[*] e-mail: shivakumaranaik17@gmail.com

ABSTRACT:

Healthcare institutions manage various activities such as patient registration, appointment scheduling and medical-record management. Performing these activities manually can cause delays, errors and difficulties in coordination.This paper presents MediFlow, an MCP-based healthcare workflow orchestration system that uses Artificial Intelligence, Large Language Models (LLMs) and multi-agent systems to automate administrative healthcare processes. The system consists of specialized agents for patient registration, appointment scheduling, medical-record management and data validation. Model Context Protocol (MCP) enables communication between AI agents and external tools.The system uses React for the frontend, Python and FastAPI for backend operations, Supabase for database management and Redis for shared workflow context. A centralized dashboard helps monitor patient information, appointments and workflow status. MediFlow aims to improve administrative efficiency and coordination while supporting healthcare professionals in routine operations..

KeywordsMCP, Artificial Intelligence, Multi-Agent Systems, Healthcare Automation, LLM, FastAPI, Digital Healthcare.

1. Introduction:

Healthcare institutions handle multiple administrative activities, including patient registration, appointment scheduling, medical-record management and laboratory-report processing. Managing these activities through separate systems can lead to delays, repeated data entry and communication gaps.Artificial Intelligence and Large Language Models provide opportunities to automate routine operations. However, AI systems also require a mechanism to interact with external tools and services.MediFlow addresses this challenge by combining LLMs, Model Context Protocol and multi-agent AI. It uses specialized agents to perform different healthcare tasks and coordinates their execution through a central orchestrator.The system provides a centralized dashboard to monitor patient details, appointments, workflow progress and task status. Its main purpose is to simplify healthcare administration and improve coordination between different processes.

Problem Statement:

Traditional healthcare workflows often depend on manual coordination and disconnected applications. This can result in delayed appointments, duplicate data entry, inconsistent patient information and difficulties in tracking ongoing activities.

There is a need for an intelligent system that can understand user requests, distribute tasks among specialized agents, connect with external tools and monitor workflow execution. MediFlow aims to address these challenges through MCP-based integration and multi-agent workflow automation.

Objectives:

1. To develop an intelligent healthcare workflow automation system.
2. To implement specialized AI agents for different healthcare activities.
3. To integrate external tools using Model Context Protocol.
4. To automate patient registration and appointment scheduling.
5. To manage authorized patient records and laboratory reports.

Scope of the Project:

MediFlow focuses on automating selected administrative healthcare processes. It supports patient registration, appointment scheduling, medical-record management and workflow validation.

The system uses AI agents to coordinate tasks and MCP to connect with external tools. It also provides a dashboard for monitoring workflow execution and patient-related information.

The project is intended to assist healthcare professionals with administrative activities. It does not replace doctors or make independent clinical decisions.

2. Literature Survey:

1 Healthcare Information Systems

Healthcare Information Systems support patient registration, appointment management and medical-record storage. However, separate applications can create difficulties in coordinating different healthcare activities.

2 Artificial Intelligence in Healthcare

Artificial Intelligence helps process information, understand user requests and automate repetitive tasks. Large Language Models provide natural-language understanding, which can be used to simplify healthcare administration.

3 Multi-Agent Systems

Multi-agent systems divide complex tasks among specialized agents. Each agent performs a specific responsibility and communicates with other agents to complete the overall workflow.

4 Model Context Protocol

Model Context Protocol provides a standardized mechanism for connecting AI applications with external tools and resources. It enables agents to discover and use supported tools through a common interface.

5 Research Gap

Existing systems often manage healthcare activities separately. MediFlow aims to integrate multiple administrative workflows using MCP, specialized AI agents, shared context and centralized monitoring.

3. Methodology:

MediFlow follows a multi-agent workflow architecture. Initially, the user submits a request through the React interface as shown in fig 1.

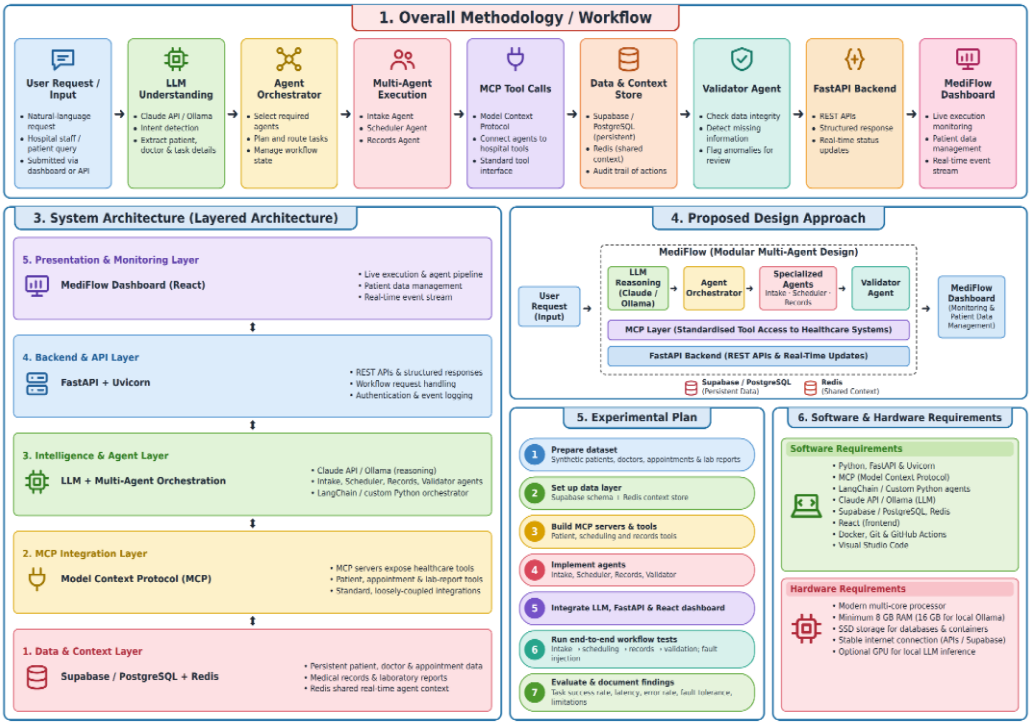


Fig 1 MediFlow system workflow and architecture

System Architecture:

The system consists of a React-based frontend, an AI orchestration layer, specialized agents and an MCP integration layer. FastAPI manages backend communication. Supabase with PostgreSQL stores persistent data, while Redis maintains shared context and workflow status. The dashboard displays patient information and workflow progress.

The technology stack is listed in Table 2.

Table 2 Software technology stack of MEDI FLOW

Layer	Technology	Purpose
Frontend	React	Building interactive user interfaces and dashboard
Backend	Python	Implementing backend logic and AI agent operations
API Framework	FastAPI	Creating REST APIs and handling frontend-backend communication
Server	Uvicorn	Running the FastAPI application
Protocol	MCP	Connecting AI agents with external tools and services
AI	LLM	Understanding natural-language requests and generating responses
Database Platform	Supabase	Providing database services, authentication and backend services
Database	PostgreSQL	Storing and managing structured patient and appointment data
Context Store	Redis	Maintaining shared agent context and temporary workflow state

The MediFlow architecture consists of interconnected layers responsible for user interaction, AI processing, task coordination, tool integration, backend communication and data management.

The presentation layer uses React to provide the user interface and dashboard. The AI processing layer uses an LLM to understand natural-language requests. The orchestration layer selects and coordinates specialized agents. The agent layer contains Intake, Scheduler, Records and Validator Agents, each responsible for a specific healthcare workflow.

The integration layer uses MCP to connect agents with authorized external tools and services. FastAPI manages backend operations and communication between system components. Supabase with PostgreSQL stores persistent patient information, while Redis maintains shared context and temporary workflow information. The dashboard displays workflow status, patient details, appointments and recent activities.

Multi-Agent Architecture

Intake Agent: The Intake Agent handles patient registration. It collects the required patient details, checks for missing information and creates or updates patient records through authorized tools.

Scheduler Agent: The Scheduler Agent manages appointment-related activities. It checks doctor availability, identifies suitable appointment slots, handles scheduling conflicts and processes booking requests.

Records Agent: The Records Agent manages authorized patient information. It retrieves patient history, updates medical records and accesses laboratory reports according to the user's permissions.

Validator Agent: The Validator Agent checks the information produced by other agents. It identifies missing fields, detects inconsistencies and flags workflow errors before completion.

MCP Integration

MCP acts as the standardized communication mechanism between the AI agents and external healthcare tools. MCP servers expose available tools and resources, allowing the connected AI application to discover and invoke supported operations.

For example, when a patient requests an appointment, the Scheduler Agent can use an MCP-exposed availability-checking tool and an appointment-booking tool. The result is returned to the agent and passed to the workflow orchestrator for further processing.

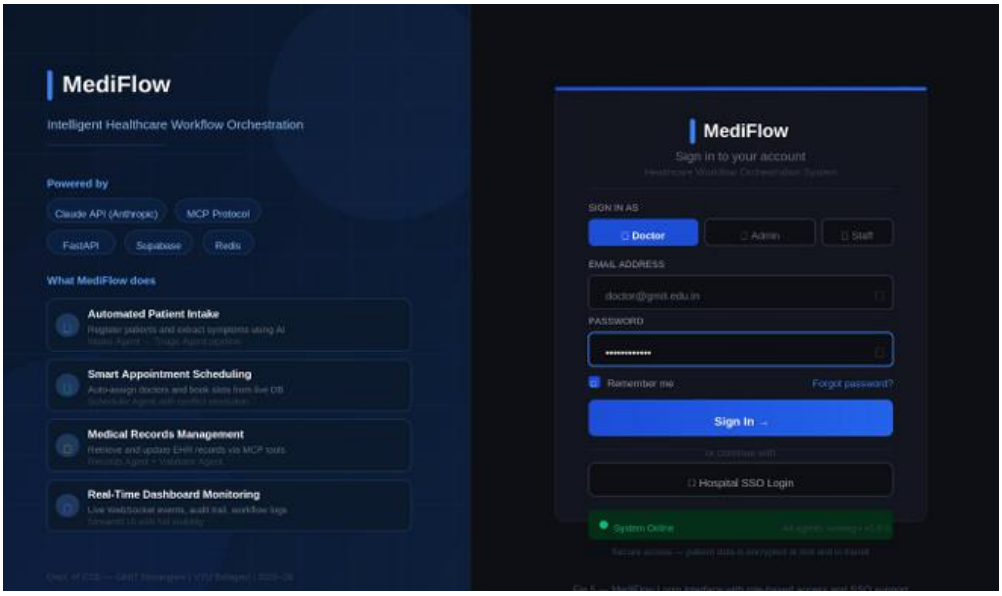
MCP does not replace the underlying APIs or databases. It provides a common protocol through which AI applications can interact with supported external services.

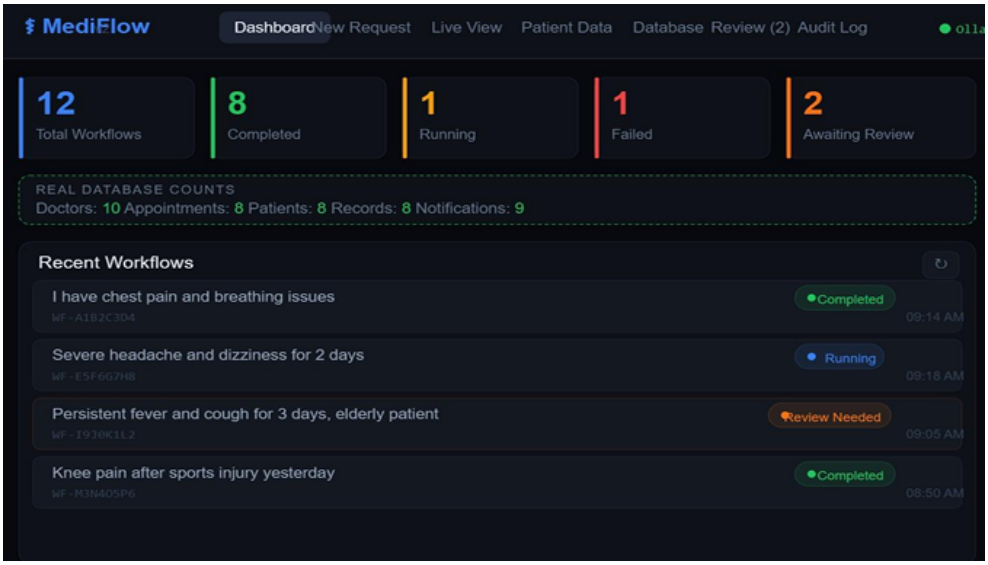
4. Results

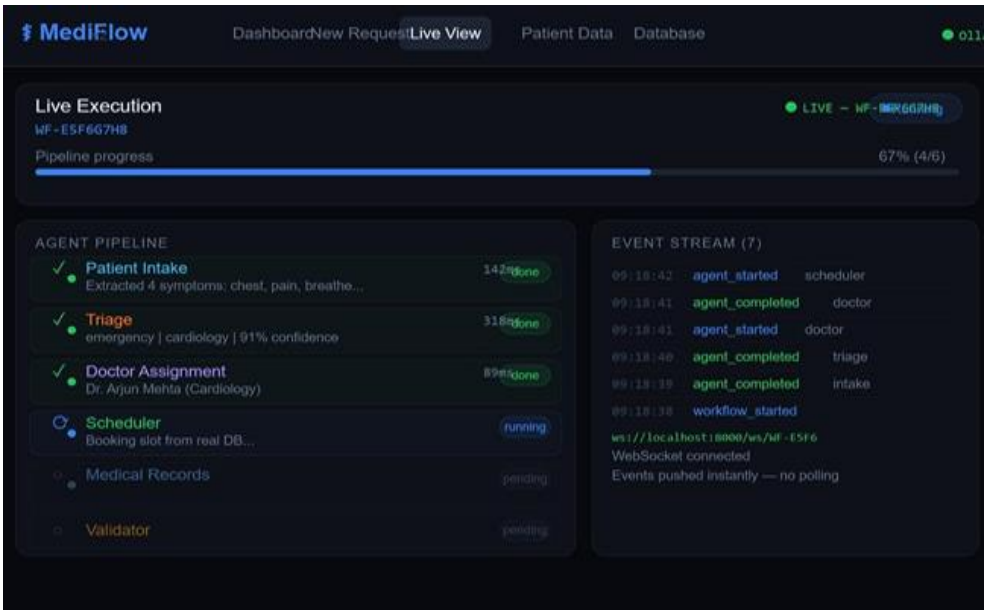
The MediFlow prototype provides a centralized platform for coordinating selected healthcare administrative workflows. It integrates specialized agents for patient registration, appointment scheduling, medical-record management and validation.

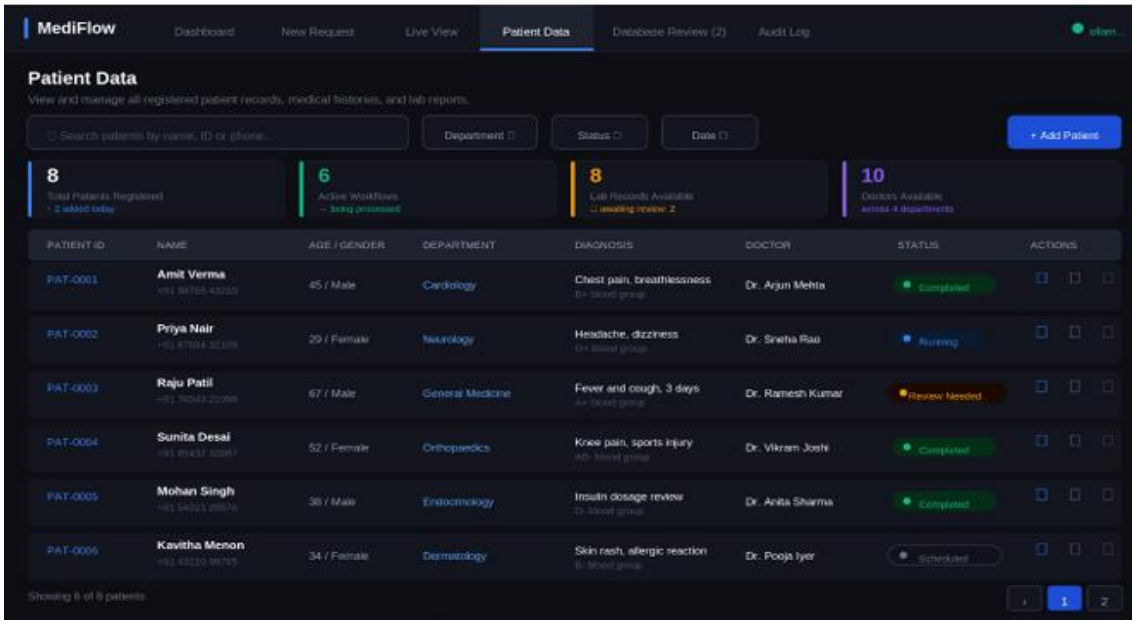
The dashboard provides visibility into patient information, workflow progress, completed tasks and pending activities. The system can be evaluated through workflow completion, data consistency, tool integration and error-handling tests. Quantitative performance results should be added after actual system testing.

Output Images :


MediFlow
Intelligent Healthcare Workflow Orchestration
Powered by
Claude API (Anthropic)
MCP Protocol
FastAPI
Supabase
Redis
What MediFlow does
Automated Patient Intake
Smart Appointment Scheduling
Medical Records Management
Real-Time Dashboard Monitoring
MediFlow
Sign in to your account
SIGN IN AS
Doctor
Admin
Staff
EMAIL ADDRESS
doctor@gmit.edu.in
PASSWORD
Remember me
Forgot password?
Sign In
Hospital SSO Login
System Online


MediFlow
Dashboard
New Request
Live View
Patient Data
Database
Review (2)
Audit Log
12
Total Workflows
8
Completed
1
Running
1
Failed
2
Awaiting Review
REAL DATABASE COUNTS
Doctors: 10 Appointments: 8 Patients: 8 Records: 8 Notifications: 9
Recent Workflows
I have chest pain and breathing issues
Completed
09:14 AM
Severe headache and dizziness for 2 days
Running
09:18 AM
Persistent fever and cough for 3 days, elderly patient
Review Needed
09:05 AM
Knee pain after sports injury yesterday
Completed
08:50 AM


MediFlow
Dashboard
New Request
Live View
Patient Data
Database
Live Execution
WF-E5F6G7H8
Pipeline progress
67% (4/6)
AGENT PIPELINE
Patient Intake
Extracted 4 symptoms: chest, pain, breathe...
done
Triage
emergency | cardiology | 91% confidence
done
Doctor Assignment
Dr. Arjun Mehta (Cardiology)
done
Scheduler
Booking slot from real DB...
running
Medical Records
pending
Validator
pending
EVENT STREAM (7)
agent_started scheduler
agent_completed doctor
agent_started doctor
agent_completed triage
agent_completed intake
workflow_started
ws://localhost:8000/ws/WF-E5F6
WebSocket connected
Events pushed instantly — no polling



5. Conclusion

MediFlow presents an MCP-based healthcare workflow orchestration system using Artificial Intelligence and multi-agent technology. It coordinates patient registration, appointment scheduling, medical-record management and data validation through specialized agents.

By integrating MCP, FastAPI, React, Supabase and Redis, the system provides a centralized approach to managing healthcare administrative workflows. It aims to reduce manual coordination, improve information management and provide better visibility into workflow execution.

REFERENCES:

Research Papers:

[1] S. Tripathi, R. Sukumaran, and T. S. Cook, "Efficient Healthcare with Large Language Models: Optimizing Clinical Workflow and Enhancing Patient Care," *Journal of the American Medical Informatics Association*, vol. 31, no. 6, pp. 1436–1440, 2024. DOI: 10.1093/jamia/ocad258.

[2] J. Qiu et al., "LLM-Based Agentic Systems in Medicine and Healthcare," *Nature Machine Intelligence*, vol. 6, pp. 1418–1420, 2024. DOI: 10.1038/s42256-024-00944-1.

[3] D. Wang and S. Zhang, "Large Language Models in Medical and Healthcare Fields: Applications, Advances, and Challenges," *Artificial Intelligence Review*, vol. 57, article 299, 2024. DOI: 10.1007/s10462-024-10921-0.

[4] H. G. Pandey, A. Amod, and S. Kumar, "Advancing Healthcare Automation: Multi-Agent System for Medical Necessity Justification," *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pp. 39–49, 2024. DOI: 10.18653/v1/2024.bionlp-1.4.

[5] L. Wang et al., "A Survey on Large Language Model Based Autonomous Agents," *Frontiers of Computer Science*, vol. 18, article 186345, 2024. DOI: 10.1007/s11704-024-40231-1.

[6] N. Mehandru et al., "Evaluating Large Language Models as Agents in the Clinic," *npj Digital Medicine*, vol. 7, article 84, 2024.

[7] Z. Al Nazi and W. Peng, "Large Language Models in Healthcare and Medical Domain: A Review," *Informatics*, vol. 11, no. 3, article 57, 2024. DOI: 10.3390/informatics11030057.

[8] V. Ayyagari, "Model Context Protocol for Agentic AI: Enabling Contextual Interoperability Across Systems," *International Journal of Computational and Experimental Science and Engineering*, vol. 11, no. 3, pp. 6072–6082, 2025. DOI: 10.22399/ijcesen.3678.