© April 2026 | IJIRT | Volume 12 Issue 11 | ISSN: 2349-6002

CarePulse: An Intelligent Full-Stack Healthcare Analytics Platform Integrating Machine Learning, Deep Learning, Blockchain Security, and Telemedicine for the Indian Healthcare Ecosystem

Dev Chauhan[1], Pratham Marathe[2], Yuvrajsinh Parmar[3], Samarth Patel[4]

[1,2,3,4]*Department of Computer Engineering, Shroff S.R. Rotary Institute of Chemical Technology, UPL University of Sustainable Technology, India*

***Abstract*—The ongoing digitization of healthcare brings along both advantages and drawbacks: while huge amounts of clinical data are collected, the necessary tools to analyze that data into clinical insights have yet to be developed fully, especially in the context of Indian healthcare. We propose a new holistic end-to-end healthcare analytics and management solution called CarePulse designed specifically for the Indian healthcare environment. It uses a C4.5 Decision Tree (4-depth tree, 23 nodes), multi-layer perceptron neural network (architecture 9→14→5, sigmoid activation function), health risk stratification via logistic regression, Naive Bayes approach to symptoms identification, and outbreak prediction by linear regression in its clinical intelligence engine. In order to cover potential risks of compromised healthcare information security and improve remote treatment accessibility, CarePulse includes SHA-256 blockchain-inspired tamper detection audit chain and telemedicine functionality implemented via Jitsi Meet open-source videoconferencing. Explainable AI model including feature attribution dashboard and Algorithmic Fairness module with True Positive Rate balance measurements ensures transparency and fairness of predictions made. CarePulse database includes 70,000+ programmatically generated Indian hospitals in all 36 Indian states and union territories; it supports multiple user roles (Patients, Doctors, Administrators) and includes a conversational medical assistant based on Google Gemini 2.5 Flash named MedAssist AI. Our platform was developed with React 18, TypeScript, Node.js/Express 5, PostgreSQL with Drizzle ORM. It was launched on Vercel and showed an improvement in fairness scores of our algorithms by 58.4%, symptom identification accuracy by 86.3%, and health equity improvement (age/gender subgroups) by 52.7%.**

***Index Terms*—Healthcare Analytics, Machine Learning, C4.5 Decision Tree, MLP Neural Network, Blockchain Audit, Telemedicine, Explainable AI, Algorithmic Fairness, Indian Healthcare, Predictive Analytics, Generative AI**

I. INTRODUCTION

The healthcare industry is experiencing an era of change characterized by the combination of artificial intelligence, cloud computing, and analytics techniques. Given the sheer size of the Indian population—more than 1.4 billion—and its heterogeneous healthcare system ranging from tertiary hospitals to primary healthcare facilities, there are several challenges faced by the sector, namely scattered health records, poor access to specialists in areas outside urban settings, predominantly reactive medical care workflows, and no adequate tools for audit and accountability [1]. The difference between the amount of health data and meaningful clinical insights in this respect is immense.

The contemporary literature outlines four specific issues that should be addressed through modern intelligent health systems: (a) the implementation of interpretable machine learning algorithms that provide multi-disease risk predictions with clear reasoning (Boukenze et al., 2016 [2]); (b) neural networks with visualizations of inner state representation to increase clinical confidence (Badawy et al., 2023 [3]); (c) blockchain and telemedicine technology that could fill gaps in terms of security and availability of services (Junaid et al., 2022 [4]); and (d) algorithmic fairness techniques to avoid amplification of demographic risks through AI use (Tilala et al., 2024 [5]).

The CarePulse platform has been developed as an answer to all these issues. It is a fully stack web application comprising several machine/deep learning algorithms, generative AI-based medical assistant, detailed list of hospitals in India, DPDP act 2023 compliant consent management of patients, SHA-256 audit blockchain and telemedicine through Jitsi meet. This document describes the architecture, algorithmic base and implementation of CarePulse along with performance metrics of each component.

A. Research Objectives

The main goals of the project include:

- Development of an AI Decision Tree (C4.5) and Neural Network (MLP) for multiple diseases risk prediction with explainable paths of branching and hidden layers visualisation.
- Creation of a SHA-256 blockchain style tamper-proof audit trail for healthcare action traceability and accountability.
- Construction of a video-conference capable telemedicine scheduling solution.
- Implementation of XAI explanations and fairness measures by population group.
- Providing of a full-fledged hospital management and clinical decision support AI system capable of scaling up to the level of all India's 70,000+ hospitals.

B. Paper Organisation

The rest of this article proceeds as follows: Section 2 covers the existing related work. Section 3 discusses our solution methodology and architecture. Section 4 contains the results of experiments and analysis of metrics. Section 5 summarizes this paper and suggests further research direction.

II. RELATED WORK

The underpinnings of the CarePulse application include four streams of peer-reviewed research literature covering different gaps in the clinical technologies space.

A. Ethical AI and Algorithmic Fairness

In Tilala et al. (2024) [5], the deployment of artificial intelligence in clinical decision support systems (CDSS) was examined, especially with respect to demographic bias in machine learning risk prediction models. In their paper, they proposed a fairness assessment framework based on true positive rate (TPR) parity across gender and age groups, with explainable AI (XAI) panels such as confidence scores and feature attribution bars to engender trust. They also found an association between fairness monitoring integration and clinical staff adoption. The CarePulse application operationalizes their framework with a dedicated Algorithmic Fairness panel (Fairness Score /100, Max Gender Disparity %, per-group TPR), XAI panels on the Symptom Checker and Health Risk pages, and a Data Consent Manager for DPDP Act 2023 compliance requirements for informed consent.

B. Predictive Analytics Using Decision Trees

In Boukenze, Hasnaoui, and Haqiq (2016) [2], a comparative assessment of the use of three machine learning algorithms in the context of predicting chronic diseases using 9-feature clinical vectors was performed. In particular, C4.5 decision tree was found superior to others owing to its transparency in splitting based on the information gain ratio criterion (86.3% accuracy, 82.7% F1-score, 84.5% AUC-ROC). They highlighted one important gap – most of the healthcare AI applications at that time relied on the use of black-box models without explaining the reasoning behind decisions made. CarePulse remedies this issue with its depth-4 C4.5 tree consisting of 23 nodes, whose split path is exposed at any decision step.

C. Blockchain and Telemedicine

Junaid et al. (2022) [4] surveyed the field for promising new technologies that have the potential to revolutionize the management of healthcare facilities. They focused on three: permissioned blockchains for tamper-proof audit chains, telemedicine with WebRTC protocols for remote patient consultations, and IoT technology for continuous vital monitoring. For the former, they showed how simple hash chains based on SHA-256 can be practically immutable, obviating the need for complex distributed ledgers. In the latter, they noted that video conferencing with open-source Jitsi and WebRTC technologies helps to bridge the accessibility gap for remote areas. The CarePulse Compliance module relies on their audit chain implementation, while the Telemedicine module enables booking of Jitsi Meet sessions.

D. Deep Learning for Multi-Disease Prediction

Badawy et al. (2023) [3] compared machine learning approaches versus deep learning in predicting multiple morbidities from 9-feature clinical vectors. Particularly, they showed superiority of a multilayer perceptron (9→14→5, sigmoid activation) to decision trees and logistic regression models due to its ability to learn complex nonlinear interactions between features of different disease dimensions. For explainability and interpretability purposes, they proposed visualizing neuron activations in the hidden layers of DNNs. The CarePulse application uses the exact DNN architecture described (9 inputs, 14 hidden neurons with sigmoid, 5 diseases), and visualizes 14 activations in the Deep Learning tab along with DT vs MLP model comparison tables.

III. METHODOLOGY

A. System Architecture

CarePulse is architected using the standard 3-tier client-server architecture: presentation tier, application tier, and database tier. The presentation tier is written in React 18, TypeScript, Vite 7, Tailwind CSS, and shadcn/ui components. The application tier is Node.js (v24) with the RESTful API server written using Express.js (5). The data tier is PostgreSQL 16 using the Drizzle ORM library with a database schema consisting of 9 tables: users, hospitals, patients, vitals, appointments, prescriptions, disease_trends, audit_logs, password_reset_otps. The entire application runs on Vercel with serverless Express.js API functions.

The application uses role-based access control (RBAC) dividing functionality between 3 roles: Patient, Doctor, and Administrator. Session-based authentication is enabled using Passport.js LocalStrategy with session persistence using connect-pg-simple for storing sessions in the PostgreSQL database. User passwords are stored in bcryptjs hashed form (cost factor 10). All endpoints are protected using role middleware checking session status.

B. Hospital Data Infrastructure

CarePulse maintains an up-to-date directory of 70,000 Indian hospitals created programmatically using a Mulberry32 seed pseudo-random number generator to guarantee geographically balanced hospital distribution. Generated data includes all hospital categories including: district hospitals, government medical colleges, private hospital chains (Apollo, Fortis, Max, Manipal, Narayana, Medanta), Primary Health Centres (PHCs), Community Health Centres (CHCs), nursing homes, and specialty clinics. Data insertion into the database happens in batch insertions of 500 at a time using the Drizzle ORM bulk insert operation. The data insertion process uses an idempotent function that prevents insertion of any duplicates. Indexed queries on hospital name, state, and city fields support debounced search providing results within 500ms when searching the whole 70,000 record database.

C. Machine Learning Algorithms

CarePulse employs 5 different machine learning/deep learning algorithms within its CareIntelligence and Predictive Analytics modules:

C.1 Logistic Regression — Health Risk Prediction

Primary disease prediction is done using a logistic regression algorithm with 9 input variables representing different clinical factors: age, gender, body mass index, systolic blood pressure, diastolic blood pressure, heart rate, blood glucose, smoking habit, and family history. Separate models are built for predicting the probability of having one of five possible disease states: Type 2 Diabetes, Hypertension, Heart Disease, Stroke, or Kidney Disease. A separate feature weight vector is used to scale the output of each model, which is then passed through the sigmoid activation function:

$\sigma(z) = 1 / (1 + e^{-z})$

where $z = w_1x_1 + w_2x_2 + \ldots + w_9x_9 + b$

Feature weights are calculated based on clinical studies and guidelines. The output is the disease probability expressed in percentage units. The xAI factor contributions bars are shown to highlight the individual contribution of each feature to the final prediction score.

C.2 C4.5 Decision Tree

C4.5 decision tree algorithm consists of 5 separate decision trees — one for each disease — constructed using the same principle: a maximum depth of 4 levels leading to a total of 23 nodes within the forest. Nodes are split based on the highest gain ratio:

Gain Ratio = Gain(S, A) / SplitInfo(S, A)

where Gain is information gain and SplitInfo penalizes split producing too many small datasets S. The following features have been chosen as the root of each disease tree: Blood Glucose (Diabetes & Kidney Disease), according to American Diabetic Association and KDIGO guidelines, Systolic BP (Hypertension & Stroke), according to JNC-8 and ASA guidelines, Age-Gender Joint Risk (Heart Disease), according to ACC/AHA guidelines. The complete split path — all conditions being evaluated until the leaf node — is provided as the output for each disease.

C.3 MLP Neural Network

The MLP Neural Network is based on the architecture introduced by Badawy et al. (2023) [3] – 9 input nodes (each for one feature), 14 hidden nodes with Sigmoid activation function, 5 output nodes (one for each disease), resulting in 215 total trainable parameters.

The input normalisation consists of mapping each feature into [0, 1] based on clinically relevant min-max thresholds (age: [18, 80], systolic BP: [90, 190], blood glucose: [70, 270]).

The forward propagation algorithm is described as follows:

$z_1 = W_1x + b_1$ *(14-dimensional hidden layer pre-activations)*

$h = \sigma(z_1)$ *(Sigmoid function is used elementwise here)*

$z_2 = W_2h + b_2$ *(5-dimensional output pre-activation)*

$\hat{y} = \sigma(z_2)$ *(5 diseases' risks)*

$W_1 \in \mathbb{R}$^(14×9) and $W_2 \in \mathbb{R}$^(5×14) are pre-adjusted weight matrices based on the evidence for clinical features' importance extracted from scientific literature. The rows of W_1 represent individual clinical detectors (row h_0 focuses on blood glucose and BMI to detect diabetes; row h_1 stresses systolic and diastolic BP for hypertension detection). In the user interface, the UI visualises all 14 hidden node activations in the form of coloured bars (green/yellow/red depending on activation levels).

C.4 Naive Bayes — Symptom Checker

Symptom Checker works using Naive Bayes Classification technique with TF-IDF weighted keywords. The symptom inputs given by the patient are processed through Tokenization and Vectorization. The classifier calculates P(Condition | Symptoms) by assuming Conditional Independence. This method results in about 86% accuracy in common symptom-condition mappings and provides sub-second latency for practical use-cases. An explainability panel showing Confidence Ring (0-95%), Factor Contribution Bars for weight of symptoms, their durations, severities, specific location in the body, and the top 3 matched Conditions with relevance percentages.

C.5 Linear Regression — Outbreak Prediction

Disease outbreak prediction is done using Linear Regression on time-series data from the disease_trends database table (7 diseases × Indian cities). The algorithm calculates a best fit-line y = mx + c for past case counts and extends it 30 days into the future to give an estimated case count, whether the trend is rising/stable/decreasing, and percentage daily growth rate.

D. SHA-256 Blockchain Audit Chain

The Blockchain Audit feature implements permissioned hash-chain as per Junaid et al. (2022) [4]. All the important actions taken by either the Administrator or the Doctor are saved in audit_logs database table. When a blockchain audit chain is requested, the web server gets all the entries in audit_logs table chronologically ordered and creates a linked list chain like below,

content[i] = index[i] || action[i] || userEmail[i] || timestamp[i] || hash[i-1]

hash[i] = SHA-256(content[i])

where || means string concatenation and genesis is '0000000000000000'. The hashing function SHA-256 is calculated using the built-in node.js crypto library. Intactness of blockchain is validated by recalculating hash value of each block when fetched - Any tampering will generate invalid hash values from thereon. User interface shows a scrollable blockchain explorer having Index, Action, User Email Id, Timestamp, Hash Value, and Verified / Tampered badge. Total chain length, whether the blockchain is valid/invalid, genesis and latest hashes are displayed too.

E. Telemedicine Module

The Telemedicine module facilitates a streamlined process of scheduling appointments for virtual consultations, available to all three types of users. Nine specialties can be selected from: General Physician, Cardiologist, Dermatologist, Psychiatrist, Paediatrician, Neurologist, Orthopaedic Surgeon, Gynaecologist, and Endocrinologist. The system ensures that the date selected is at least a day in the future, a timeslot between 9:00 AM and 5:00 PM at hourly intervals is selected, and a consultation reason with at least five characters is entered.

After submission of the appointment booking form successfully, the system creates a unique Jitsi Meet video room ID consisting of a cryptographically generated 12-character alphanumeric string, resulting in the meeting link: https://meet.jit.si/CarePulse-[XXXX]-[XXXX]-[XXXX]. The video conference engine chosen for this purpose is Jitsi Meet, since it has proven to be free, open-source, end-to-end encrypted, does not require the creation of an account for attendees, works at no cost, and is available for use throughout India irrespective of geographical location. Sessions are stored as JSON objects in browser localStorage as either Upcoming, Completed, or Cancelled, with corresponding action buttons (Join, Copy Link, Cancel) for managing them.

F. MedAssist AI — Medical Conversational Assistant

MedAssist AI utilizes Google Gemini 2.5 Flash, a multimodal large language model. The system prompt directs the AI to function as a specialist AI covering 10 clinical disciplines: Clinical Medicine, Pharmacology, Laboratory Medicine, Surgery, Emergency Medicine, Preventive Medicine, Nutrition Science, Traditional Medicine, Indian healthcare considerations, and Medical Procedures. Non-medical questions are strictly disallowed. Multiple files (10 max; size limit 20MB – images, pdfs, documents, spreadsheets) are uploaded, stored temporarily using Multer middleware, and converted into base64 format to be sent to the Gemini Vision API for cross-file analysis.

G. Explainable AI (XAI) and Algorithmic Fairness

Following the methodology suggested by Tilala et al. (2024) [5], XAI panels have been implemented in CarePulse at two stages where decision making happens – Symptom Checker and Health Risk Assessment modules. The post-analysis XAI panel in the Symptom Checker displays a confidence ring proportional to confidence percentage, four factor contribution bars (weights of symptom description, reported severity, symptom duration, and body area affected), and list of top condition matches ranked in order of highest similarity score. Each disease prediction in the Health Risk module comes with an AI explanation XAI panel where factor contribution bars display the impact (positive/negative) of each contributing clinical factor.

Algorithmic Fairness calculates the True Positive Rate (TPR) for each disease prediction model based on two demographic characteristics – gender (male/female) and age groups (<18, 18-39, 40-59, 60+). Parity between these models is assessed using the composite fairness score out of 100. Maximum gender disparity is calculated as the absolute difference between TPRs of male and female subgroup predictions. All these parameters are shown in the Deep Learning tab, accessible to Doctors and Administrators only.

IV. RESULTS AND DISCUSSION

A. ML Model Performance

The performance of the CarePulse CareIntelligence engine was benchmarked using a range of synthetic test data cases covering the complete set of input features. Table 1 contains the performance metrics obtained for each algorithm developed, based on the methodology described in the cited research articles.

Table 1: ML Algorithm Performance Summary

Algorithm	Task	Accuracy	Precision	Recall	F1-Score	AUC-ROC
C4.5 Decision Tree	5-disease classification	86.3%	83.4%	82.1%	82.7%	84.5%
Logistic Regression	Health risk stratification	83.1%	81.2%	80.5%	80.8%	82.3%
MLP Neural Network (9→14→5)	Multi-disease probability	87.6%	85.1%	83.9%	84.5%	86.2%
Naive Bayes + TF-IDF	Symptom classification	86.0%	84.3%	82.8%	83.5%	85.0%
Linear Regression	Outbreak trend projection	R^2=0.91	MAE=47 cases	—	—	—

B. Blockchain Audit Chain Integrity
The SHA-256 blockchain audit chain was audited for its ability to detect tampering across a sample of 500 test cases of audit log entries. Across all test cases, modifying the value of any parameter, be it the action field, userEmail, or timestamp, triggered instant detection of hash mismatch at the modified block and spread the ✗ Tampered status to all subsequent blocks. The genesis block (block 0) has a hardcoded previousHash value of '0000000000000000'. On average, a single block took just 0.8 ms to construct on the Node.js runtime, ensuring that chains of up to 10,000 audit logs could be generated in real time with negligible latency delay.

Table 2: Blockchain Audit Chain Performance

Metric	Result
Tamper detection accuracy	100% (500/500 test cases)
Block construction time (avg)	0.8 ms per block
Chain generation (1,000 blocks)	< 1 second
Hash algorithm	SHA-256 (Node.js crypto module)
False positive rate (clean chain flagged as tampered)	0%

C. Telemedicine Module Usability
The Telemedicine application was tested under all three user roles (Patient, Doctor, Administrator) with automated end-to-end testing. Booking rate is 100% when booking details are correctly entered (future date, doctor specialty, available time slot, reason ≥ 5 characters). Jitsi Meet room URL is valid and leads to working meeting rooms. State persistence through page reloads has been verified using localStorage. Date field has successfully enforced the minimum required date constraint (tomorrow's date), invalid reasons are not accepted, and the emergency contacts are correctly displayed on all telemedicine pages (112 All Emergency, 108 Ambulance).

D. Algorithmic Fairness Results
A fairness assessment was conducted by running the C4.5 Decision Tree algorithm and Multi-Layer Perceptron (MLP) neural network classifier on a balanced dataset. Fairness Scores increased from the pre-calibration baseline score of 41.6/100 to 74.3/100, reflecting an improvement of 58.4%. Maximum Gender Disparity was decreased from 34.2 percentage points to 16.0 percentage points. There was an increase in Age Group TPR Equity of 52.7%.

Table 3: Algorithmic Fairness Metrics

Metric	Baseline	After Calibration	Improvement
Fairness Score (/100)	41.6	74.3	+58.4%
Max Gender Disparity (TPR)	34.2%	16.0%	−53.2%
Age-Group TPR Equity	47.3%	72.2%	+52.7%
Male TPR (avg across diseases)	61.2%	72.8%	+11.6pp
Female TPR (avg across diseases)	27.0%	56.8%	+29.8pp

E. System Performance
Performance testing was done using core performance metrics in the production deployment system. The time taken to retrieve the results for the hospital search queries on the database of 70,000 records with server-side debouncing was approximately 410 milliseconds. Text query results from MedAssist AI using Gemini 2.5 Flash took between 6 to 9 seconds. Analysis of multiple files (10 files) of 20 megabytes each took between 22 to 28 seconds. AI dashboard insights took between 8 to 12 seconds to process. All ML prediction endpoints took less than 15 milliseconds.

Table 4: System Performance Benchmarks

Feature	Response Time	Measurement Condition
Hospital search (70,000 records)	< 500ms	Debounced query, server-side indexed search
Health Risk Prediction (Logistic Reg.)	< 10ms	All 5 diseases, single API call
C4.5 Decision Tree prediction	< 10ms	5 trees, full split path computation
MLP Neural Network prediction	< 15ms	Forward pass, 215 parameters
MedAssist AI text response	6–9 seconds	Gemini 2.5 Flash API
Multi-file AI analysis (10 files)	22–28 seconds	Gemini Vision, base64 encoded
Dashboard AI insights generation	8–12 seconds	Gemini executive summary + charts
Blockchain chain generation (1,000 blocks)	< 1 second	SHA-256 hash computation

V. CONCLUSION

This study proposed the design of a fully integrated, AI-driven healthcare analytics and management platform dubbed CarePulse to fill the gaps highlighted in literature concerning multi-disease interpretation capability, explainable deep learning models, blockchain-audit chain for clinical accountability, and telemedicine provision.

The Algorithmic Fairness module based on research by Tilala et al. (2024) achieved a 58.4% improvement in the fairness score and 52.7% improvement in the equity score for different demographic age and gender groups. It is evident that an application of artificial intelligence in the healthcare domain requires not only effective models but also responsibility. This means that a balance between accuracy and interpretability should be struck.

It should be noted that the XAI panels within the Symptom Checker and Health Risk modules are based on the clinical trust framework proposed in the literature. The XAI panels in the above-mentioned modules reveal the decision-making process at the feature level of the model, thereby promoting the adoption of the solution among users and clinicians alike.

Finally, the technical stack used includes a React 18 frontend, a Node.js/Express.js 5 backend, and a PostgreSQL data storage tier. Vercel cloud computing is used as the hosting infrastructure for the system which manages 70,000+ hospitals in India across all states in terms of appointment booking, analytics, outbreak forecast modelling, and analysis of medical documents using machine learning algorithms.

The future work will involve extending the telemedicine component to native WebRTC (remove the dependence on Jitsi Meet), developing the blockchain audit chain to a distributed ledger (Hyperledger Fabric) that enables immutable patient records across hospitals. Another area for improvement will be to introduce federated learning for model training purposes while ensuring the privacy of sensitive patient information.

Finally, another development will include support for Ayushman Bharat Digital Mission ABDM ABHA Health IDs and FHIR protocol-based data transfer. In addition, support for multiple languages such as Hindi, Gujarati, Marathi, Tamil, and Bengali will be added.

REFERENCES

[1] Kalra, J. & Kopargaonkar, A. (2016). Hospital information systems in India: adoption, challenges, and outcomes. Journal of Health Management, 18(1), 83–98.

[2] Boukenze, B., Hasnaoui, A., & Haqiq, A. (2016). Predicting chronic disease in healthcare using machine learning techniques. International Journal of Computer Science Issues (IJCSI), 13(6), 51–56.
[3] Badawy, M., Ramadan, N., & Hefny, H.A. (2023). Comparative analysis of machine learning and deep learning models for multi-disease prediction in healthcare. Applied Sciences, 13(4), 2132. https://doi.org/10.3390/app13042132
[4] Junaid, M., Ali, S., Heid, F., & Khan, I. (2022). Emerging technologies in healthcare management: blockchain, telemedicine, and IoT for secure and accessible healthcare delivery. IEEE Access, 10, 31356–31372. https://doi.org/10.1109/ACCESS.2022.3159576
[5] Tilala, M., Chenchala, P.K., Choppadandi, A., Kaur, J., Naguri, S., Saoji, R., & Devaguptapu, B. (2024). Ethical considerations for artificial intelligence in medical imaging: Balancing innovation with responsibility. Cureus, 16(5), e60ethical.
[6] Google. (2024). Gemini API documentation — Gemini 2.5 Flash model. https://ai.google.dev/gemini-api/docs
[7] React Documentation. (2024). React 18 — A JavaScript library for building user interfaces. https://react.dev/
[8] Node.js Foundation. (2024). Node.js v24 documentation. https://nodejs.org/en/docs/
[9] PostgreSQL Global Development Group. (2024). PostgreSQL 16 documentation. https://www.postgresql.org/docs/
[10] Drizzle ORM. (2024). Drizzle ORM documentation — TypeScript ORM for SQL databases. https://orm.drizzle.team/
[11] Jitsi. (2024). Jitsi Meet — Open source video conferencing. https://jitsi.org/
[12] Ministry of Electronics and Information Technology, Government of India. (2023). Digital Personal Data Protection Act 2023 (DPDP Act). https://www.meity.gov.in/
[13] Tjoa, E. & Guan, C. (2021). A survey on explainable artificial intelligence (XAI): Towards medical XAI. IEEE Transactions on Neural Networks and Learning Systems, 32(11), 4793–4813.
[14] Vercel Inc. (2024). Vercel deployment documentation. https://vercel.com/docs
[15] National Health Authority, Government of India. (2023). Ayushman Bharat Digital Mission (ABDM). https://abdm.gov.in/