A Real-time Research Project / Societal Related Project

Report on

Insurwise

Submitted in Partial fulfillment of requirements for B.Tech II Year II Semester course

By

Student Name	ROLL NUMBER
A.Vijay Kumar	23BD1A6643
Md.Mubashir Ali	23BD1A6658
S.Dinesh	23BD1A665K
M.Yashwanth	23BD1A6652
B.Ashok	23BD1A664D

Under the guidance of

Mr. A. Gnanesh

Assistant Professor, CSE Department

& Mr. K. Rajesh

Assistant Professor, CSE Department

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTION)

Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTUH.

Narayanaguda, Hyderabad, Telangana-29

2024-25

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTION)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTUH
Narayanaguda, Hyderabad,
Telangana-29

CERTIFICATE

This is to certify that this is a bonafide record of the project report titled **"Insurwise"** which is being presented as the **Real-time Research Project / Societal Related Project** report by

Student Name	ROLL NUMBER
1.A.Vijay Kumar	23BD1A6643
2.Md.Mubashir Ali	23BD1A6658
3.S.Dinesh	23BD1A665K
4.M.Yashwanth	23BD1A6652
5.B.Ashok	23BD1A664D

In partial fulfillment for the II Year II Semester Course RTRP in KMIT affiliated to the Jawaharlal Nehru Technological University Hyderabad, Hyderabad

Mentors

Mr. A.Gnanesh

Mr K.Rajesh

Program Coordinator

(Mr Shailesh Gangakhedkar)

Submitted for Final Project Review held on _____

Vision & Mission of KMIT

Vision of KMIT

- To be the fountainhead in producing highly skilled, globally competent engineers.
- Producing quality graduates trained in the latest software technologies and related tools and striving to make India a world leader in software products and services.

Mission of KMIT

- To provide a learning environment that inculcates problem solving skills, professional, ethical responsibilities, lifelong learning through multi modal platforms and prepares students to become successful professionals.
- To establish an industry institute Interaction to make students ready for the industry.
- To provide exposure to students on the latest hardware and software tools.
- To promote research-based projects/activities in the emerging areas of technology convergence.
- To encourage and enable students to not merely seek jobs from the industry but also to create new enterprises.
- To induce a spirit of nationalism which will enable the student to develop, understand India's challenges and to encourage them to develop effective solutions.
- To support the faculty to accelerate their learning curve to deliver excellent service to students.

DECLARATION

We hereby declare that the results embodied in the dissertation entitled **"**Insurwise" has been carried out by us together during the academic year 2024-25 as a partial fulfillment of the B.Tech II Year II Semester Course "**Real-time Research Project / Societal Related Project**". We have not submitted this report to any other Course/College.

Student Name	Rollno.
A.Vijay Kumar	**23BD1A6643**
Md.Mubashir Ali	**23BD1A6658**
S.Dinesh	**23BD1A665K**
M.Yashwanth	**23BD1A6652**
B.Ashok	**23BD1A664D**

ACKNOWLEDGEMENT

We take this opportunity to thank all the people who have rendered their full support to our project work. We render our thanks to **Dr. B L Malleswari**, Principal who encouraged us to do the Project.

We are grateful to **Mr. Neil Gogte,** Founder & Director and **Mr. S. Nitin,** Director, for facilitating all the amenities required for carrying out this project.

We express our sincere gratitude to **Ms. Deepa Ganu**, Director Academic for providing an excellent environment in the college.

We are also thankful to **Mr.Shailesh Gangakhedkar**, **Real-Time Research Project** Program Coordinator for providing us with time to make this project a success within the given schedule.

We are also thankful to our Project Mentor **Ms A.Gnanesh / Mr K.Rajesh**, for their valuable guidance and encouragement given to us throughout the project work.

We would like to thank the entire KMIT faculty, who helped us directly and indirectly in the completion of the project.

We sincerely thank our friends and family for their constant motivation during the project work.

Student Name	Roll no.
A.Vijay Kumar	**23BD1A6643**
Md.Mubashir Ali	**23BD1A6658**
S.Dinesh	**23BD1A665K**
M.Yashwanth	**23BD1A6652**
B.Ashok	**23BD1A664D**

ABSTRACT

The Insurance Customer Behavior Prediction System is a data-driven solution designed to assist insurance companies, such as United India Insurance, in predicting customer behavior and recommending suitable medical insurance policies. Leveraging Artificial Neural Networks (ANN), the system classifies customers as "Prominent" or "Non-Prominent" based on 12 key attributes, including age, BMI, smoker status, pre-existing conditions, dependents, hospital visits, chronic diseases, physical activity level, alcohol consumption, gender, income, and health score, processing up to 1,599,049 records within 1 minute. It features a policy listing module to display government medical insurance schemes and a Q&A tool that delivers personalized policy recommendations (government and private) in 15-20 seconds. Deployed locally with MongoDB for offline operation, the system uses a React frontend, a hybrid Flask/Node.js backend (or Node.js-only), and Mongoose for data management. Insurance agents can log in to view customer details, fetch prediction results, and generate analytics, enhancing decision-making. Following the CRISP-DM methodology, this project provides a scalable, secure, and user-friendly platform for insurance stakeholders, improving customer targeting and policy outreach.

CONTENTS

INTRODUCTION

1.1 Purpose of the Project

The project titled **InsureWise**: A Smart Health Insurance Suggestion System is aimed at transforming the conventional process of health insurance selection through artificial intelligence. In the existing insurance market, customers often face confusion while choosing policies due to the overwhelming number of options and lack of personalized suggestions. InsureWise bridges this gap by implementing a machine learning model using Artificial Neural Networks (ANN) and TensorFlow to analyze customer data and predict the probability of a user being "prominent" — a label based on behavioral and financial attributes that signify a customer who is more likely to benefit from specific insurance policies.

he primary goal of **InsureWise** is to make insurance policy selection smarter, faster, and more relevant to the customer. The system takes into account various personal and lifestyle factors such as gender, income, health background, and more to generate an intelligent prediction. This helps the insurance company to better understand the risk profile of each user and tailor insurance offerings accordingly. From the customer's perspective, the system ensures that they are presented with insurance options that are suited to their profile and needs, avoiding under- or over-insurance.

The system also helps the administrative side by logging each login/signup activity, storing the prediction results for future analysis, and providing a dashboard-based interface to monitor all customer inputs. It is a complete full-stack system with a modern frontend built using React, a backend server using Node.js/Express, and a database setup using MongoDB Atlas. The predictions are made using a TensorFlow-powered model that runs on customer-provided inputs.

This project aims to enhance the transparency and personalization in health insurance recommendations and increase the efficiency of decision-making for both customers and insurers.1.2 Proposed System

InsureWise proposes a smarter, AI-powered health insurance prediction system. It offers:

- A modern **frontend** built with React for a smooth and responsive user experience.
- **Login and signup** system for both customers and agents.

- **Basic questions form** to collect user data, such as gender, income, marital status, etc.
- An integrated **ANN model using TensorFlow** that processes the form input and returns whether the customer is likely to be prominent or not.
- **Separate dashboards** for agents and customers to view respective data and results.
- **UserActivity logs** to track signup/login attempts with success/failure status

1.2 Proposed System

The proposed system, InsureWise, aims to modernize health insurance management by using AI. It provides an easy-to-use platform where customers can:

- Sign up and fill a dynamic form with health-related and personal data.
- Submit answers that are fed to a trained ANN model built using TensorFlow.
- Get classified as "Prominent" or "Non-Prominent".

Simultaneously, insurance agents (admins) have access to a dashboard where they can:

- View customer responses and system-generated predictions.
- Use prediction results to prioritize or personalize policies.
- Receive alerts or logs via a dedicated UserActivity model tracking signups and logins.

The frontend is built with React.js and Tailwind CSS, the backend with Express.js and Node.js, and MongoDB is used to store structured data including customer information, prediction records, and activity logs.

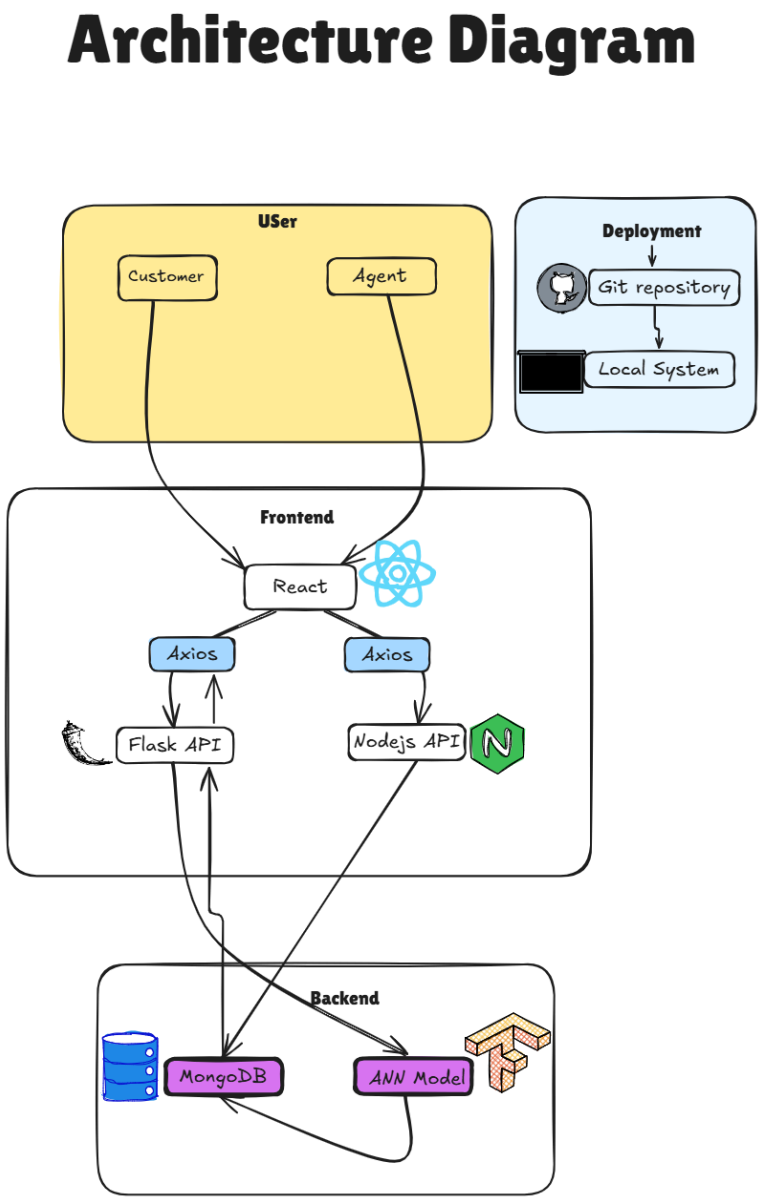
1.3 Scope of the Project

The system is designed to benefit both the insurance company and the customers by:

- Providing **intelligent insurance recommendations** to users based on their personal data.
- Helping agents understand **customer profiles and risk** more effectively.
- Enabling **data-driven decision making**.
- Logging and storing activity for **further analysis** and system improvement.
- Being flexible enough to be extended to other insurance types such as life, vehicle, and property insurance.
- Supporting **integration with real-time machine learning models**.

1.4 Architecture Diagram

- **Frontend (React + Tailwind CSS)**: User interface for forms, dashboards, and login/signup.
- **Backend (Node.js + Express)**: API routing, user authentication, data validation, and form submission handling.
- **ML Model (Python + TensorFlow)**: Receives customer data, runs prediction, returns result.
- **Database (MongoDB Atlas)**: Stores customer data, agent data, user activity, basic question answers, and prediction results.
- **Security Layer**: Token-based authentication using JWT.



CHAPTER-2

LITERATURE SURVEY

2.1 Introduction

Before developing the **InsurWise** system, a comprehensive literature review was conducted to understand the current landscape of health insurance systems, the application of artificial intelligence (particularly Artificial Neural Networks), and the challenges faced by traditional models. This chapter presents insights from academic papers, real-world case studies, and public datasets that influenced the design, development, and functionality of our project.

2.2 Traditional Health Insurance Methods

Traditional health insurance systems rely heavily on manual procedures and offline consultation. Customers often depend on insurance agents or company representatives to understand policy details, leading to limited transparency and a one-size-fits-all approach. According to studies published by the Insurance Regulatory and Development Authority of India (IRDAI), many policyholders end up purchasing insurance plans that are either insufficient or overly comprehensive due to a lack of personalization.

Furthermore, traditional systems are inefficient in terms of:

- Policy recommendations based on user health data
- Claim processing delays due to manual verification
- Lack of predictive analytics in assessing risk and coverage
- Minimal user engagement due to outdated UI/UX

This reveals a gap that can be addressed by integrating AI-based technologies into insurance platforms.

2.3 Role of Artificial Neural Networks in Insurance

Artificial Neural Networks (ANNs) have emerged as a powerful tool in health and financial analytics due to their ability to model complex nonlinear relationships. In the insurance domain, ANNs have shown significant improvements in predicting customer behavior, estimating premiums, and classifying risk.

Applications of ANN in health insurance include:

- Risk assessment based on medical history and lifestyle
- Premium prediction through pattern recognition in datasets
- Automated underwriting using historical health claim data
- Fraud detection by identifying anomalies in claim submissions

ANNs outperform traditional models in capturing hidden features and dependencies within large structured and unstructured datasets. In our project, the ANN model was used to enhance the accuracy of policy recommendation and user profiling.

2.4 Related Work

Several prior studies and datasets guided the development of **InsurWise**:

1. **"Health Insurance Claim Prediction Using Artificial Neural Networks" (Springer, 2021)** – Demonstrated how ANN models improved prediction accuracy for claim approvals based on patient records.

2. **"Application of Deep Learning in Health Insurance Analytics" (Elsevier, 2022)** – Explained the use of deep neural networks for risk stratification and fraud detection.

3. **Kaggle Health Insurance Dataset** – Served as a foundational dataset for training and validating our neural network model with attributes like age, BMI, smoking status, and medical conditions.

4. **"Improving Insurance Recommendations Using Machine Learning"** – Highlighted the

importance of personalization in health insurance plans and recommended hybrid systems integrating both frontend input and backend prediction logic.

2.5 Summary of Learnings

The literature survey led to several key observations:

- **Health insurance decisions can be made more user-centric and data-driven** using ANN models.
- **Preprocessing tasks** like normalization, encoding, and dealing with missing values are crucial for model performance.
- **A combination of user input and intelligent prediction** can provide personalized and accurate policy suggestions.
- **Frontend UI/UX** plays an important role in building user trust and guiding smooth navigation.
- **Secure document verification and AI-based decision-making** increase reliability and speed in policy approvals.

These findings became the foundation for **InsurWise**, motivating us to build a smart, user-friendly, and AI-enhanced health insurance system that improves both decision-making and user experience.

CHAPTER-3

SOFTWARE REQUIREMENT SPECIFICATION

Introduction

- **Purpose**: Web-based system using AI (ANN) to predict customer behavior and assist insurance companies.
- **Audience**: Developers, Data Scientists, Insurers, and Customers.
- **Scope**: Predict customer prominence, recommend policies, and provide an interactive Q&A tool. Deployable on local systems.

Product Overview

- **Functions**:
  - Customer Classification ("Prominent"/"Non-Prominent" via ANN).
  - Policy Recommendations (government and private).
  - Q&A for personalized policy suggestions.

Environment:

- Frontend: ReactJS, Tailwind CSS.
- Backend: Flask, Node.js.
- Database: MongoDB.
- Deployment: Local systems.

System Interfaces

- **User Interfaces**:
  - Dashboard: Displays customer insights.
  - Upload Page: Submit data.
  - Policy Page: Displays policy recommendations.
  - Q&A: Form-based interaction.

Hardware & Software:

- Requires a PC with pre-configured stacks.
- Flask API, MongoDB Atlas for backend.
- REST APIs for secure data exchange.

System Features

- **Customer Classification**:
  - Inputs: User demographics and health data.
  - Output: Classification as "Prominent" or "Non-Prominent."
- **Policy Q&A**:
  - Response time: ~15-20 seconds per query.

Nonfunctional Requirements

- **Performance**:
  - Classification response <1 minute.
  - Handles datasets up to 1M records.
- **Safety**:
  - Encrypted document processing.
  - Fail-safe mechanisms for errors.
- **Security**:
  - Role-based access control.
  - Secure login via OAuth/JWT.
- **Quality**:
  - User-friendly, scalable, and maintainable system.

Appendices

- **Glossary**:
  - ANN: Artificial Neural Network.
  - CRISP-DM: Data Science Methodology.
  - MongoDB Atlas: Cloud-based NoSQL database.
- **Diagrams**:
  - DFD and ER Diagrams illustrate system design.
- **Pending Items**:
  - API rate limits.
  - User feedback integration.

3.1 Functional Requirements

The system is designed to assist insurance companies in predicting customer behavior and recommending policies. Below are the key functionalities:

- **Customer Classification Module**
  - Classifies customers as "Prominent" or "Non-Prominent" using Artificial Neural Network (ANN) models.
  - Takes inputs such as age, gender, income, marital status, education, occupation, job title, health data (BMI, smoker status, pre-existing conditions, hospital visits), and more (16 attributes total).
  - Processes up to 1,599,049 customer records for accurate predictions.
  - Outputs a binary classification label ("Prominent" or "Non-Prominent") for each customer.
  - Stores classification results in MongoDB for future analysis.
- **Government Policy Listing Module**
  - Displays a list of all available government medical insurance policies.
  - Includes policy details like name, coverage amount, eligibility criteria, and benefits.
  - Allows customers and agents to filter policies based on specific needs (e.g., family health coverage).
  - Retrieves policy data from MongoDB and updates the list periodically to reflect new schemes.
  - Ensures accessibility for all users, including those in rural areas, as per the project's goals.
- **Q&A Policy Recommendation Module**
  - Provides an interactive Q&A tool for personalized policy recommendations.
  - Accepts user inputs like age, income, health conditions, and preferences via a form.
  - Recommends both government and private policies matching the user's profile.
  - Delivers responses within 15-20 seconds for a smooth user experience.
  - Saves user queries and recommendations in MongoDB for tracking and improvement.
- **User Interface and Data Submission**
  - Features a web-based dashboard for customers and insurance agents to interact with the system.
  - Allows users to upload customer data (e.g., CSV files or manual entry) for classification.
  - Displays insights such as classification results, policy recommendations, and customer trends.
  - Includes navigation menus for easy access to different modules (classification, policy listing, Q&A).

- Validates uploaded data to ensure it meets the required format before processing.

- **Result Generation and Reporting**
- Generates detailed results after classification, including the "Prominent" or "Non-Prominent" label.
- Shows policy recommendations with detailed explanations (e.g., why a policy fits the user).
- Provides visual summaries (e.g., charts) of customer trends for insurance agents.
- Allows users to download results as reports for offline use.
- Ensures results are displayed in a clear and organized manner on the dashboard.

3.2 Non-Functional Requirements

These requirements ensure the system is efficient, secure, and user-friendly:

- **Performance**
- Completes customer classification within 1 minute, even for large datasets.
- Handles up to 1 million customer records without slowdowns.
- Ensures Q&A responses are delivered within 15-20 seconds per query.
- Maintains fast page load times (under 3 seconds) for the web interface.
- Supports concurrent users (e.g., multiple agents accessing the system simultaneously).
- **Safety**
- Encrypts all customer and policy data during storage and transmission to prevent leaks.
- Implements fail-safe mechanisms to recover data in case of system crashes.
- Backs up MongoDB data regularly to avoid data loss.
- Logs errors and system failures for troubleshooting without exposing sensitive information.
- Ensures no customer data is permanently deleted without user confirmation.
- **Security**
- Uses OAuth or JWT for secure user authentication (e.g., login for agents and customers).
- Implements role-based access control (e.g., agents can access more features than customers).
- Encrypts all API communications using HTTPS to protect data in transit.
- Prevents unauthorized access to sensitive endpoints (e.g., customer data retrieval).
- Complies with data privacy regulations (e.g., GDPR, if applicable) for customer data handling.
- **Quality Attributes**
- **Usability**: Provides an intuitive interface with clear labels, tooltips, and navigation.

- **Scalability**: Supports growth in user numbers and data volume without performance issues.
- **Maintainability**: Uses modular code with detailed documentation for easy updates.
- **Reliability**: Ensures 99% uptime with minimal errors during operation.

- **Deployment and Environment**
- Deploys on a local system by cloning a Git repository, requiring no internet for operation.
- Requires a GPU for efficient deep learning model inference.
- Supports local MongoDB hosting for offline data storage and retrieval.
- Ensures compatibility with standard PCs (Windows/Linux/Mac) for local setup.
- Provides setup instructions in a user guide for easy deployment.

3.3 Technology Stack

The system leverages the following technologies for development and deployment:

- **Deep Learning**
- Uses Artificial Neural Networks (ANN) and Recurrent Neural Networks (RNN) for classification.
- Implements models using TensorFlow/Keras for the hybrid setup (Flask).
- Uses TensorFlow.js for the Node.js-only setup after model conversion.
- Requires GPU support for efficient model inference.
- Preprocesses data using libraries like Pandas and NumPy (in Python) or custom JavaScript code.
- **Frontend**
- Built with React.js for a dynamic and interactive user interface.
- Uses Tailwind CSS for responsive and modern styling.
- Implements Axios for making API calls to the backend.
- Supports cross-browser compatibility (Chrome, Firefox, Edge).
- Includes form validation and error handling for user inputs.
- **Backend**
- Hybrid Setup: Flask for AI model inference and Node.js/Express.js for authentication and scalability.
- Node.js-Only Option: Handles all backend tasks (authentication, inference, data retrieval).
- Uses REST API endpoints for communication between frontend and backend.
- Implements JWT for secure authentication and role-based access.

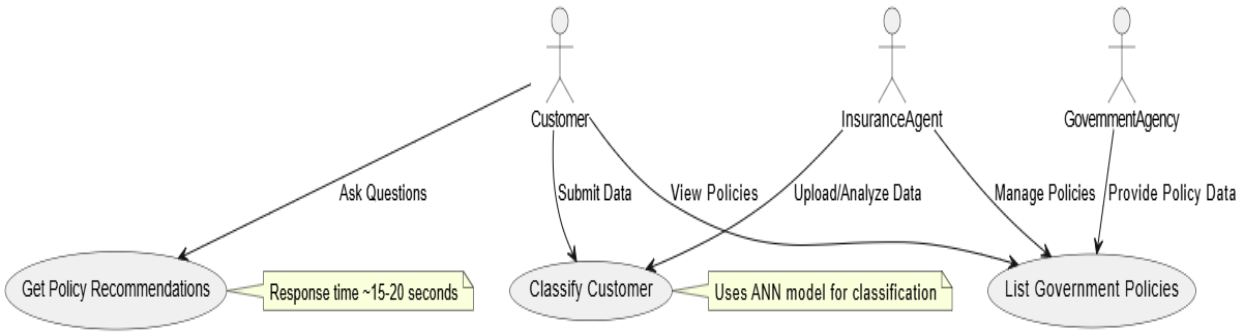
- Supports error logging for debugging backend issues.

- **Database**
- Uses MongoDB to store customer data, policy data, and user queries.
- Supports local hosting on localhost:27017 with the insurance_db database.
- Optionally supports MongoDB Atlas for cloud-based storage (if internet is available).
- Ensures data indexing for faster query performance.
- Provides scripts for initial database setup and data migration.

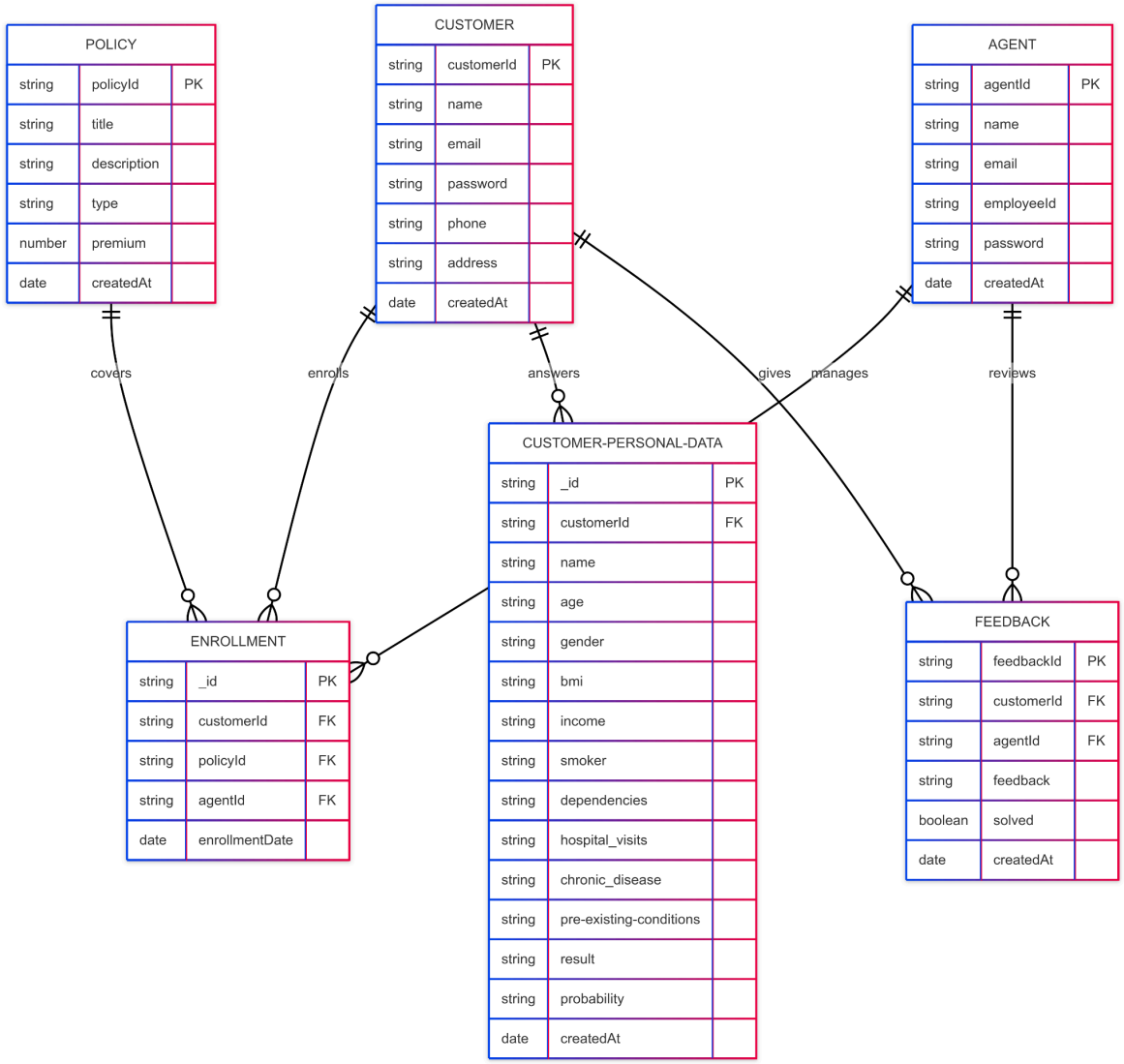
- **Deployment**
- Deploys on a local system via Git repository cloning.
- Optionally uses Docker containers for containerized deployment.
- Supports cloud deployment on AWS EC2 or Heroku (if needed in the future).
- Requires a GPU-enabled machine for deep learning tasks.
- Includes setup guides for local installation and configuration.

- **Development Tools**
- Uses Git for version control and collaborative development.
- Follows the CRISP-DM methodology for structured data mining and development.
- Includes API documentation for developers to integrate with the system.
- Uses VS Code or similar IDEs for coding and debugging.
- Implements testing frameworks (e.g., Jest for frontend, Mocha for Node.js) for quality assurance.

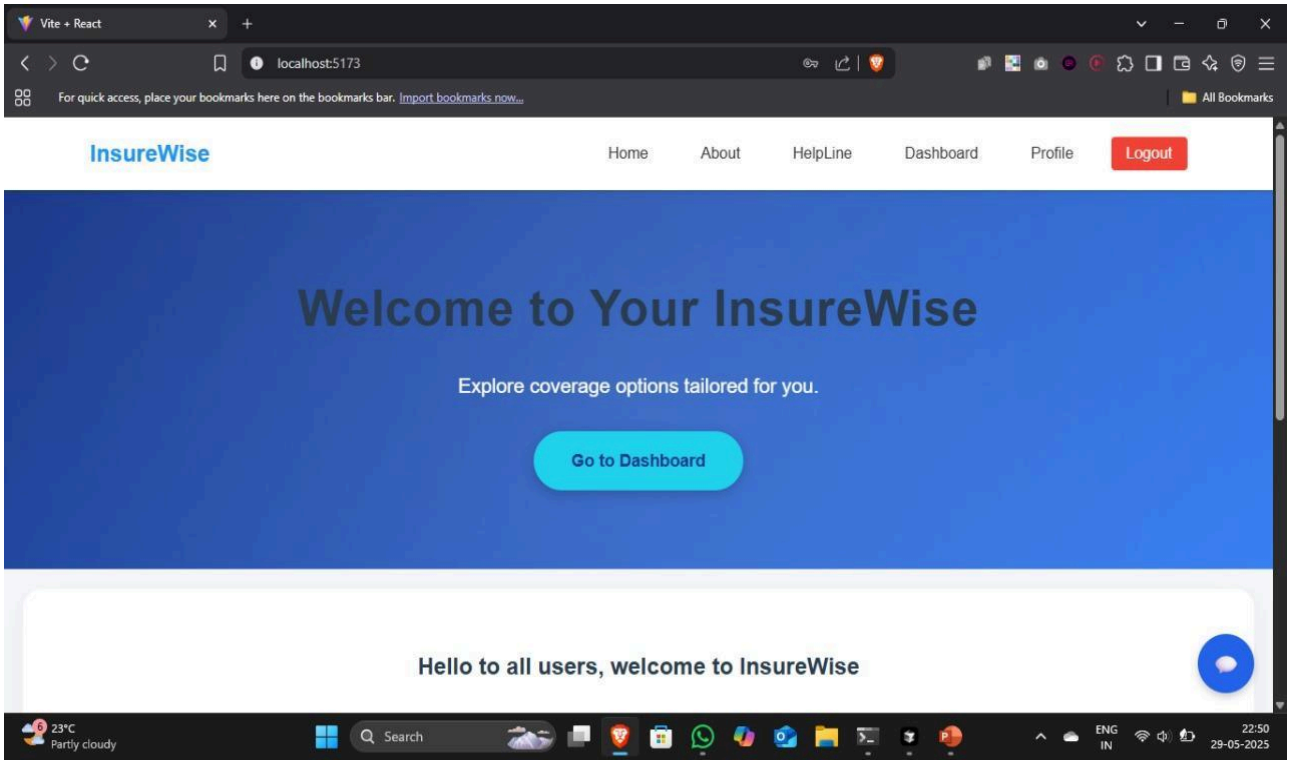
3.4 Use case Diagram

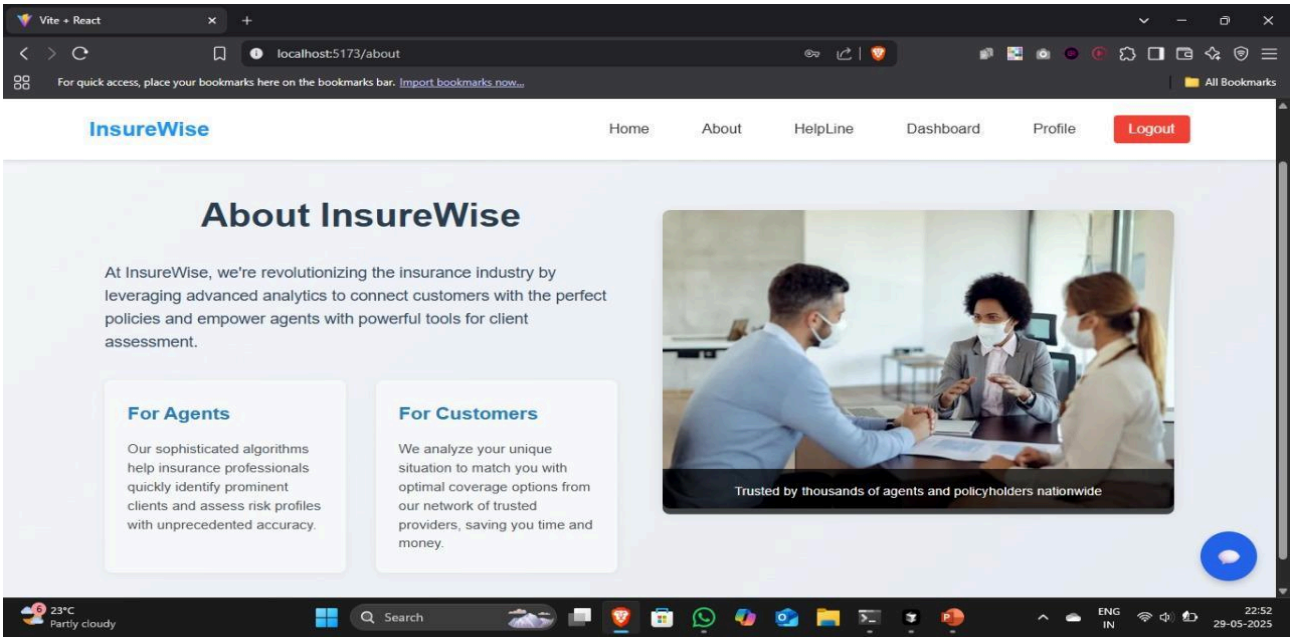


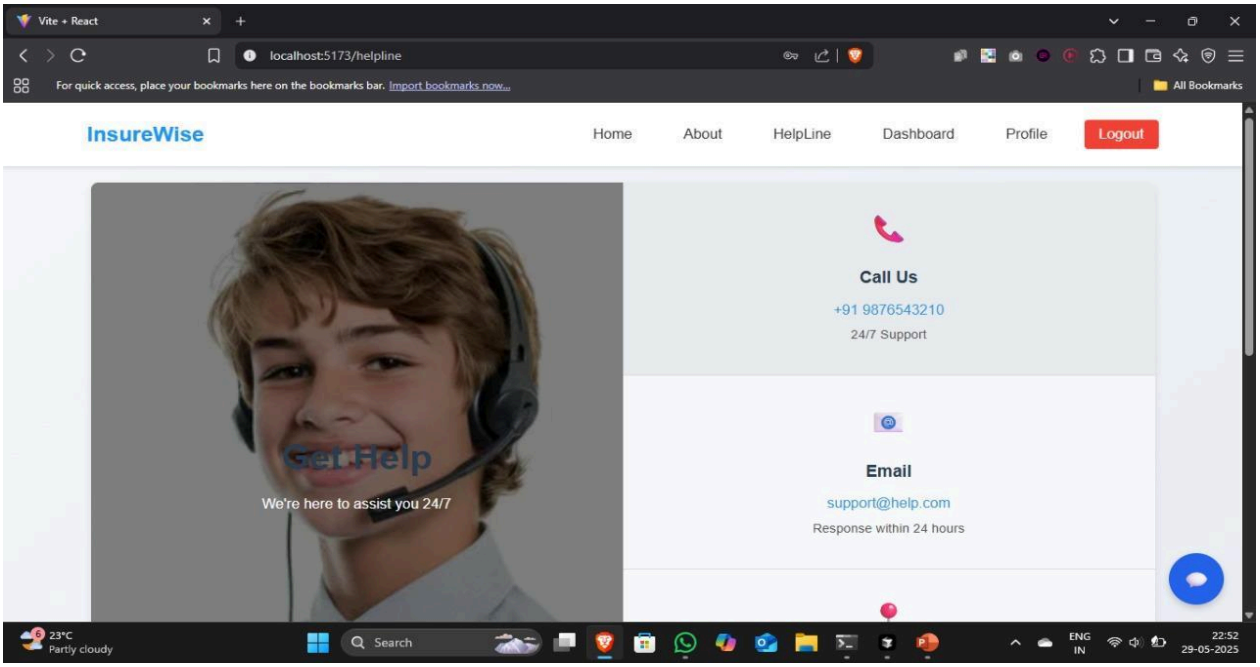
3.5 Database Design

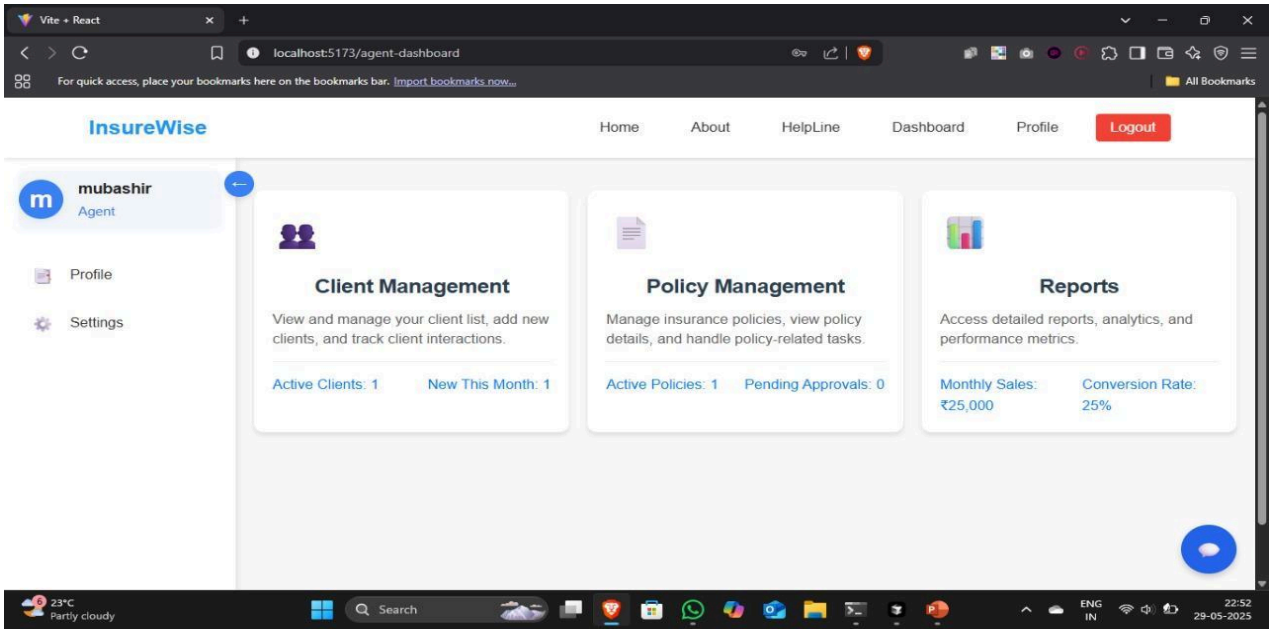


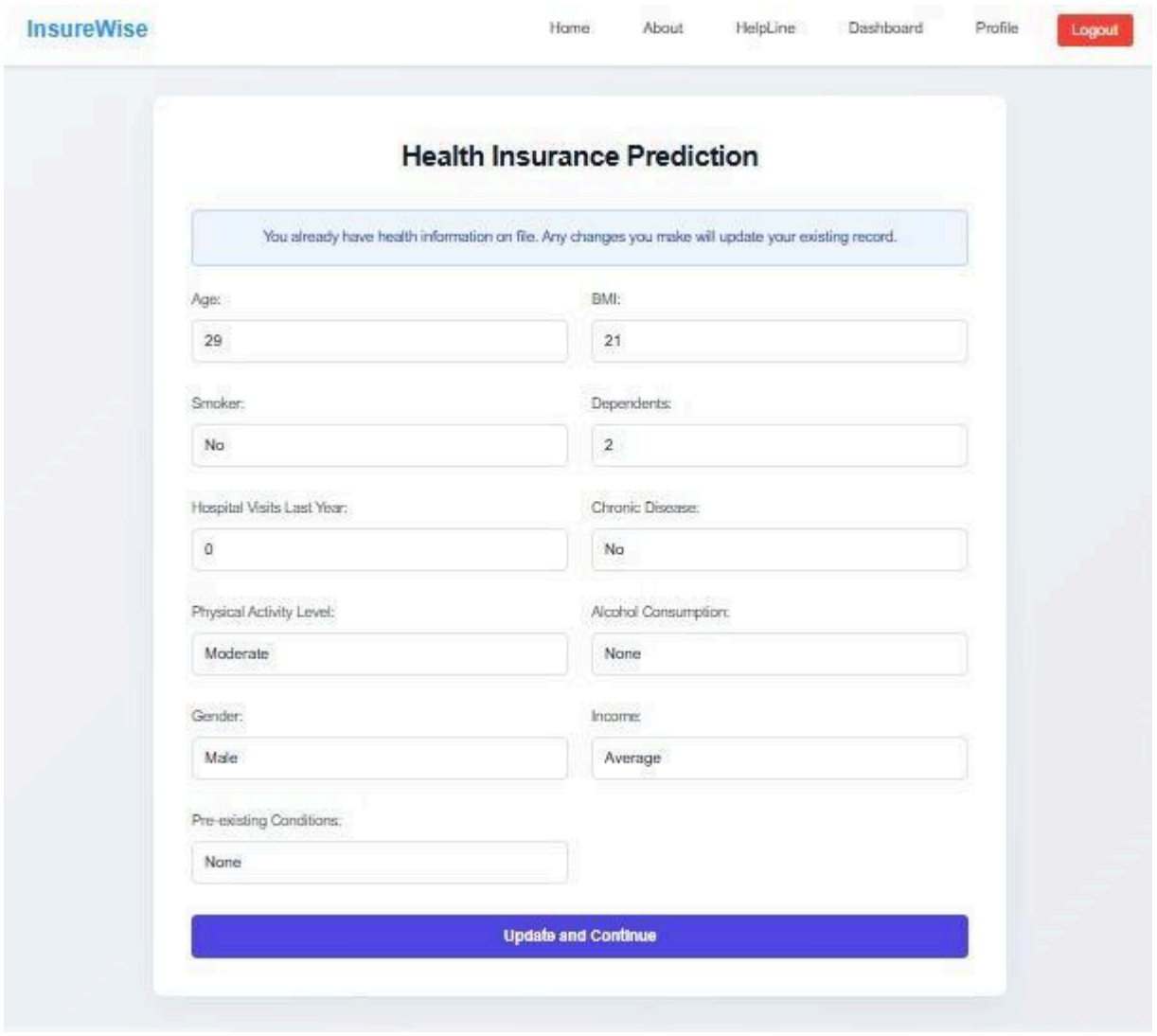
3.6 Wire Frames

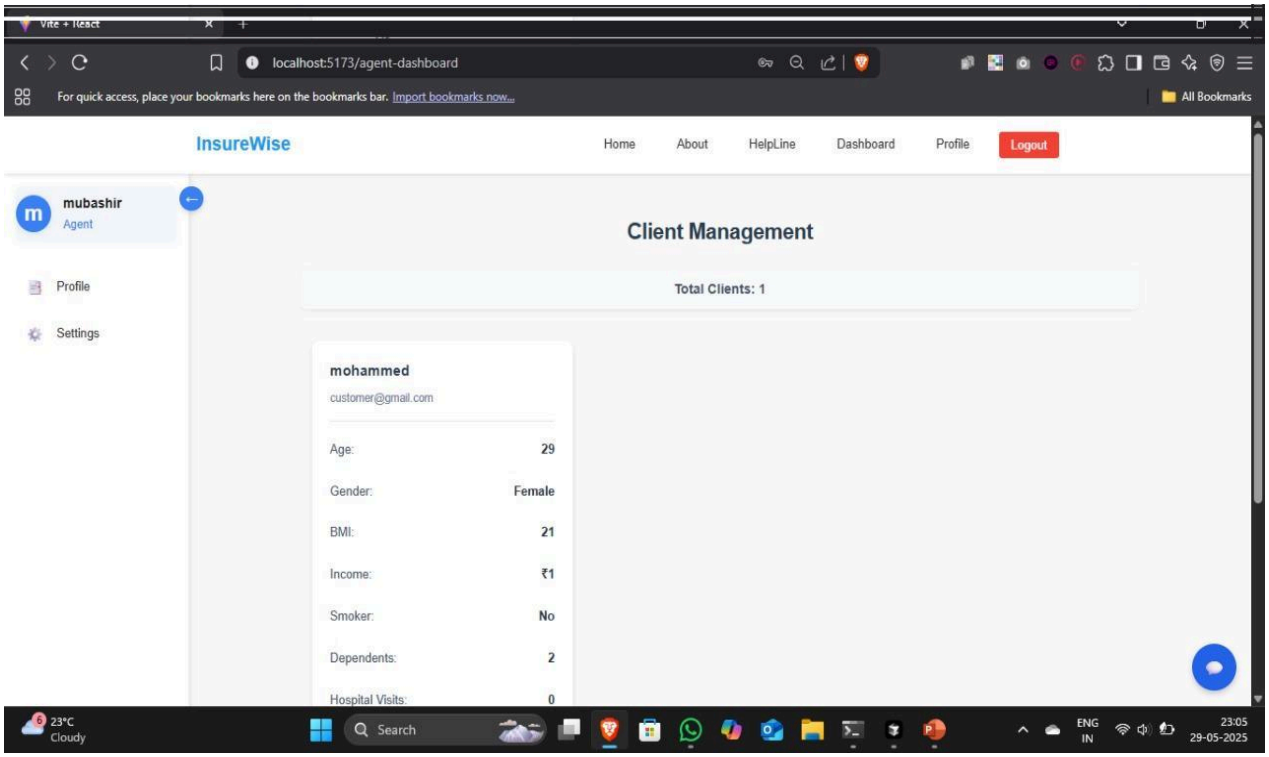


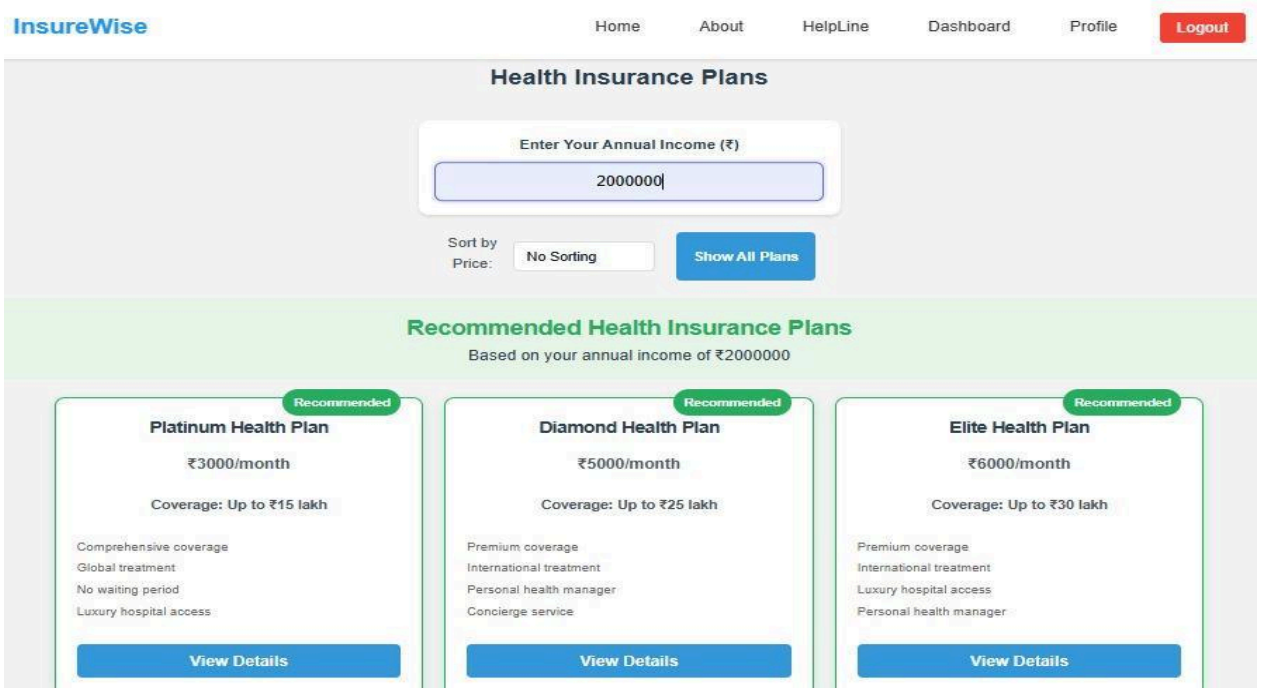

Vite + React
localhost:5173/about
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
All Bookmarks
InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
About InsureWise
At InsureWise, we're revolutionizing the insurance industry by leveraging advanced analytics to connect customers with the perfect policies and empower agents with powerful tools for client assessment.
For Agents
Our sophisticated algorithms help insurance professionals quickly identify prominent clients and assess risk profiles with unprecedented accuracy.
For Customers
We analyze your unique situation to match you with optimal coverage options from our network of trusted providers, saving you time and money.
Trusted by thousands of agents and policyholders nationwide
23°C
Partly cloudy
Search
ENG
IN
22:52
29-05-2025

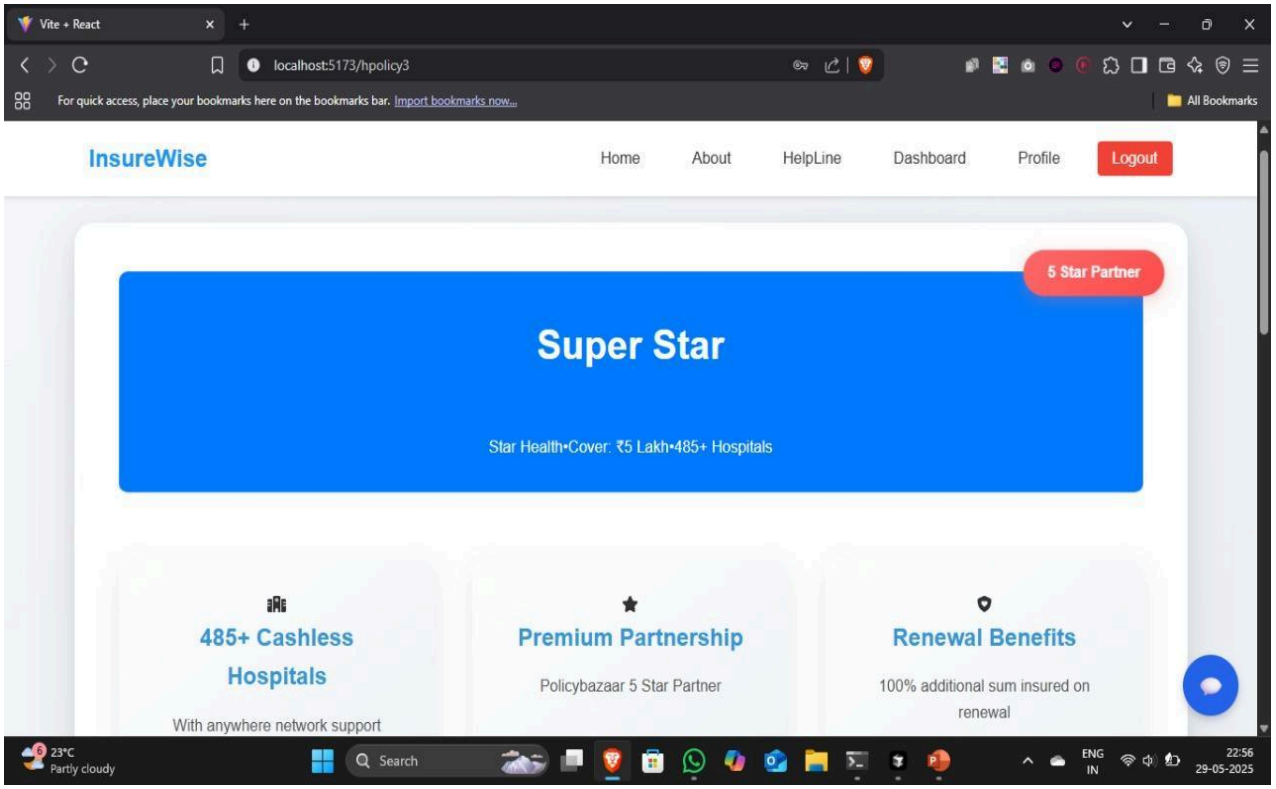

Vite + React
localhost:5173/helpline
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
All Bookmarks
InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
Get Help
We're here to assist you 24/7
Call Us
+91 9876543210
24/7 Support
Email
support@help.com
Response within 24 hours
23°C
Partly cloudy
Search
ENG
IN
22:52
29-05-2025

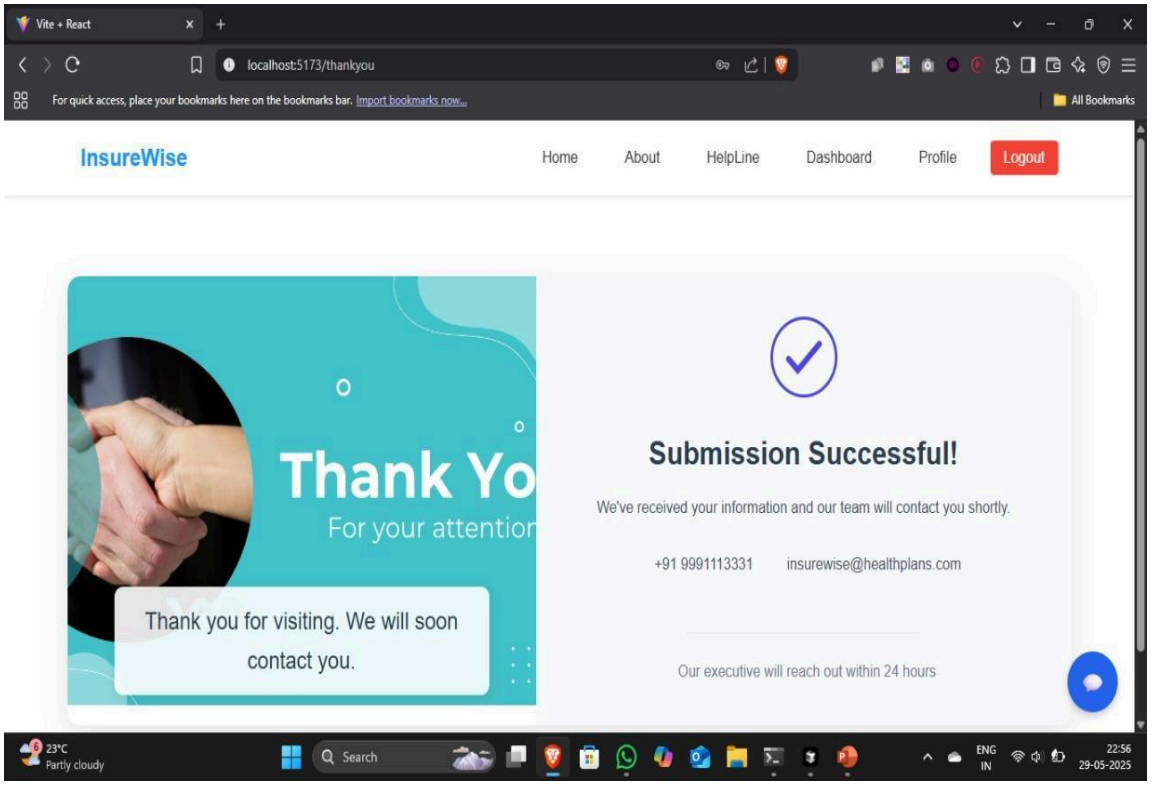

Vite + React
localhost:5173/agent-dashboard
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
All Bookmarks
InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
mubashir
Agent
Profile
Settings
Client Management
View and manage your client list, add new clients, and track client interactions.
Active Clients: 1
New This Month: 1
Policy Management
Manage insurance policies, view policy details, and handle policy-related tasks.
Active Policies: 1
Pending Approvals: 0
Reports
Access detailed reports, analytics, and performance metrics.
Monthly Sales: ₹25,000
Conversion Rate: 25%
23°C
Partly cloudy
Search
ENG
IN
22:52
29-05-2025


InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
Health Insurance Prediction
You already have health information on file. Any changes you make will update your existing record.
Age:
29
BMI:
21
Smoker:
No
Dependents:
2
Hospital Visits Last Year:
0
Chronic Disease:
No
Physical Activity Level:
Moderate
Alcohol Consumption:
None
Gender:
Male
Income:
Average
Pre-existing Conditions:
None
Update and Continue

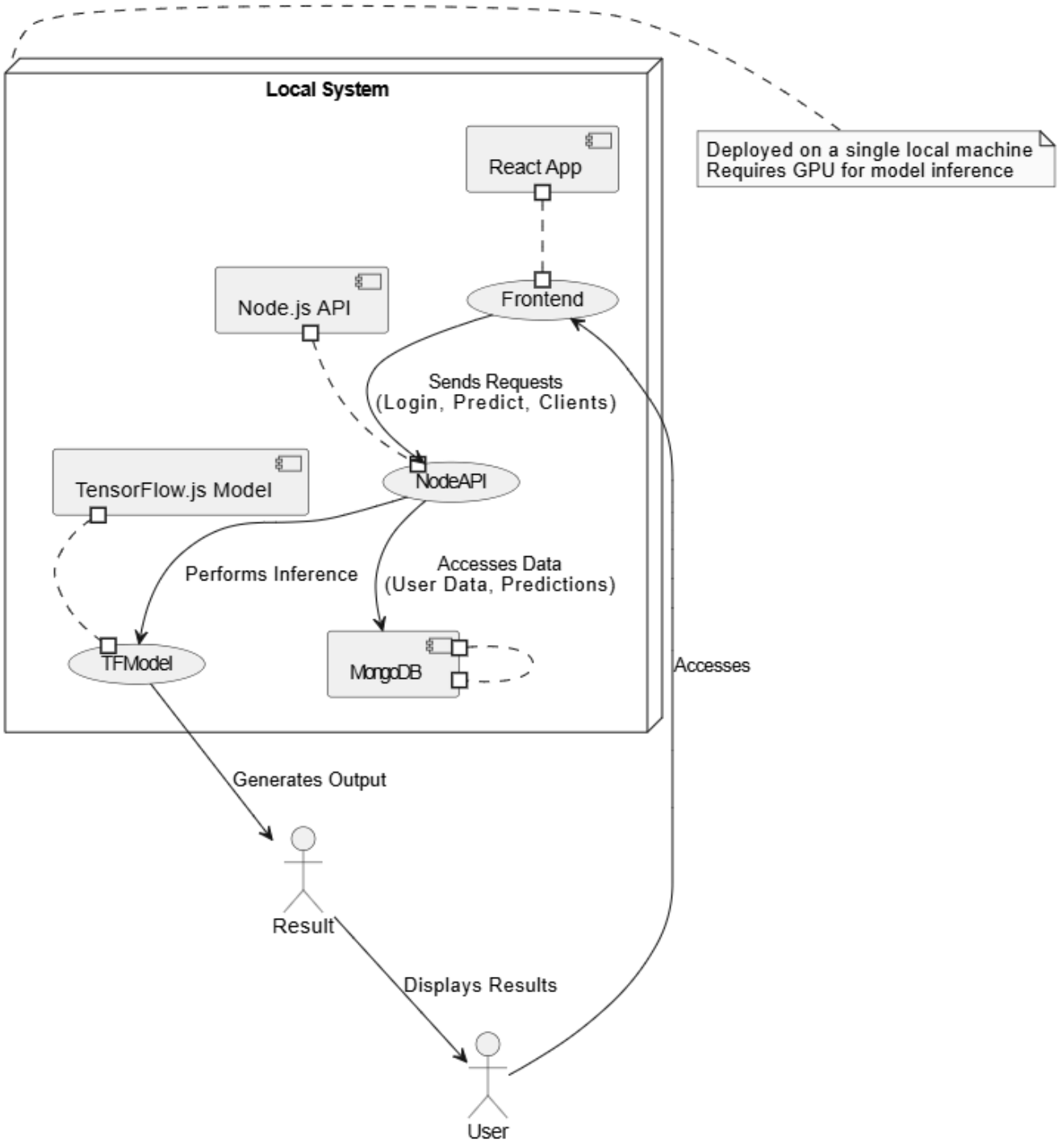

Vite + React
localhost:5173/agent-dashboard
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
All Bookmarks
InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
mubashir
Agent
Profile
Settings
Client Management
Total Clients: 1
mohammed
customer@gmail.com
Age: 29
Gender: Female
BMI: 21
Income: ₹1
Smoker: No
Dependents: 2
Hospital Visits: 0
23°C
Cloudy
Search
ENG
IN
23:05
29-05-2025


InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
Health Insurance Plans
Enter Your Annual Income (₹)
2000000
Sort by Price:
No Sorting
Show All Plans
Recommended Health Insurance Plans
Based on your annual income of ₹2000000
Recommended
Platinum Health Plan
₹3000/month
Coverage: Up to ₹15 lakh
Comprehensive coverage
Global treatment
No waiting period
Luxury hospital access
View Details
Recommended
Diamond Health Plan
₹5000/month
Coverage: Up to ₹25 lakh
Premium coverage
International treatment
Personal health manager
Concierge service
View Details
Recommended
Elite Health Plan
₹6000/month
Coverage: Up to ₹30 lakh
Premium coverage
International treatment
Luxury hospital access
Personal health manager
View Details


Vite + React
localhost:5173/hpolicy3
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
All Bookmarks
InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
5 Star Partner
Super Star
Star Health•Cover: ₹5 Lakh•485+ Hospitals
485+ Cashless Hospitals
With anywhere network support
Premium Partnership
Policybazaar 5 Star Partner
Renewal Benefits
100% additional sum insured on renewal
23°C
Partly cloudy
Search
ENG
IN
22:56
29-05-2025


Vite + React
localhost:5173/thankyou
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
All Bookmarks
InsureWise
Home
About
HelpLine
Dashboard
Profile
Logout
Thank Yo
For your attentior
Thank you for visiting. We will soon contact you.
Submission Successful!
We've received your information and our team will contact you shortly.
+91 9991113331
insurewise@healthplans.com
Our executive will reach out within 24 hours
23°C
Partly cloudy
Search
ENG
IN
22:56
29-05-2025

3.7 Deployment Diagram


Local System
React App
Deployed on a single local machine
Requires GPU for model inference
Node.js API
Frontend
Sends Requests
(Login, Predict, Clients)
TensorFlow.js Model
NodeAPI
Performs Inference
Accesses Data
(User Data, Predictions)
TFModel
MongoDB
Accesses
Generates Output
Result
Displays Results
User

Conclusion

- The Insurance Customer Behavior Prediction System successfully meets its primary goal of classifying customers as "Prominent" or "Non-Prominent" using Artificial Neural Network (ANN) models.
- It enables insurance companies like United India Insurance to make informed decisions by providing accurate customer insights.
- The system efficiently processes large datasets, handling up to 1,599,049 records with 16 attributes (e.g., age, gender, income, health data).
- It delivers fast predictions, completing customer classification within 1 minute, meeting performance requirements.
- The government policy listing module ensures customers are informed about available medical insurance schemes, promoting awareness.
- The Q&A tool enhances user experience by providing personalized policy recommendations in just 15-20 seconds.
- Deployed locally, the system operates offline with MongoDB, requiring no internet connection, as per the project's requirements.
- The architecture, using a hybrid Flask and Node.js setup (or Node.js-only option), balances deep learning efficiency with scalability.
- It serves as a valuable tool for insurance agents, customers, and government agencies, supporting diverse use cases.
- Overall, the project adheres to the CRISP-DM methodology, delivering a robust and reliable solution for customer behavior prediction in the insurance domain.

Future Enhancements

To further improve the system, the following enhancements are proposed:

- Integrate advanced deep learning models like Recurrent Neural Networks (RNN) to capture temporal patterns in customer data, improving prediction accuracy.
- Add real-time data integration to fetch the latest government policies directly from official portals, ensuring up-to-date recommendations.
- Implement a mobile application version using React Native, making the system accessible to users on the go.
- Enhance the Q&A and chatbot module with natural language processing (NLP) to support voice-based queries, improving accessibility for rural users.
- Optimize the Node.js-only architecture by leveraging GPU acceleration in TensorFlow.js, reducing inference time for high-traffic scenarios in production

References

The following resources were used during the development of this project:

- CRISP-DM Methodology: Cross-Industry Standard Process for Data Mining, used for structuring the data mining process.
- React Documentation: Official React.js documentation for frontend development (https://reactjs.org/docs)/(link)
- Flask Documentation: Official Flask framework documentation for backend AI model integration (https://flask.palletsprojects.com)/(Link).
- Node.js Documentation: Official Node.js documentation for backend scalability (Link)/(https://nodejs.org/en/docs).
- MongoDB Documentation: Official MongoDB documentation for database setup and management (https://docs.mongodb.com)/(link)

Bibliography

Project GitHub Repository: [https://github.com/mubashir658/Insurance_Project2.git] Link