Project Title: Health AI: Intelligent Healthcare Assistant

1.Introduction

Team ID : **NM2025TMID06823**

Team Size : 5

Team Leader : MANIMOZHI M

Team member : NIVETHA S

Team member : PRITHIKA S

Team member : RAMYA S

Team member : SIVAPRIYA K

Project Description:

HealthAI harnesses IBM Watson Machine Learning and Generative AI to provide intelligent healthcare assistance, offering users accurate medical insights. The platform includes a Patient Chat for answering health-related questions, Disease Prediction that evaluates user-reported symptoms to deliver potential condition details, Treatment Plans that provide personalized medical recommendations, and Health Analytics to visualize and monitor patient health metrics.

Pre-requisites

1. **Streamlit Framework Knowledge:** **Streamlit Documentation**
2. **IBM Watson Machine Learning:** **IBM Watson ML Documentation**
3. **Python Programming Proficiency:** **Python Documentation**

4. **Data Visualization Libraries:** [Plotly Documentation](...)

5. **Version Control with Git:** [Git Documentation](...)

6. **Development Environment Setup:** [Flask Installation Guide](...)

Model Selection and Architecture

- **Activity 1.1:** Research and select the appropriate AI model from IBM Watson for medical assistance (IBM Granite 13B Instruct v2).
- **Activity 1.2:** Setup and Access your IBM WatsonX API key.
- **Activity 1.3:** Define the architecture of the application, detailing interactions between the frontend, backend, and AI integration.
- **Activity 1.4:** Set up the development environment, installing necessary libraries and dependencies for Streamlit and IBM Watson

Core Functionalities Development

- **Activity 2.1:** Develop the core functionalities: Patient Chat, Disease Prediction, Treatment Plan Generation, and Health Analytics.
- **Activity 2.2:** Implement patient data utilities to manage and visualize health metrics.

Activity 3: App.py Development

- **Activity 3.1:** Write the main application logic in app.py, establishing functions for each feature and integrating AI responses.
- **Activity 3.2:** Create prompting strategies for the IBM Granite model to generate high-quality medical content.

Activity 4: Frontend Development

- **Activity 4.1:** Design and develop the user interface using Streamlit components, ensuring a responsive and intuitive layout.
- **Activity 4.2:** Create dynamic visualizations with Plotly to display health metrics and trends.

Activity 5: Deployment

- **Activity 5.1:** Prepare the application for deployment by configuring environment variables for API credentials.
- **Activity 5.2:** Deploy the application on a suitable hosting platform to make it accessible to users.

# Milestone 1: Model Selection and Architecture

In this milestone, we focus on selecting the appropriate AI model from IBM Watson for our medical assistance needs. This involves researching the capabilities and performance of various models, ensuring that the chosen model aligns well with our application's objectives of creating a Patient Chat system, Disease Prediction, Treatment Plan Generation, and Health Analytics.

# Research and select the appropriate AI model

1. Understand the Project Requirements: Review the specific needs of the healthcare application.

2. Explore IBM Watson ML Documentation: Examine the various models available, including their functionalities and limitations.

3. Select the Optimal Model: Choose IBM's Granite 13B Instruct v2 model for its strong performance with healthcare-related content.

Setup and Access your IBM WatsonX API key.

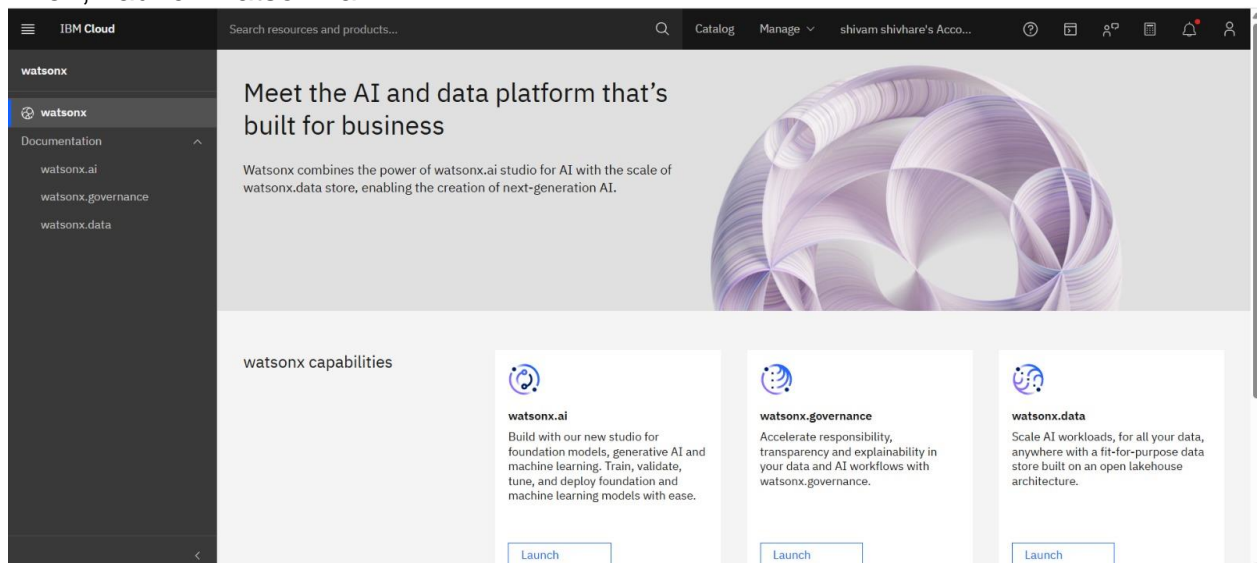1. Go to IBM WatsonX
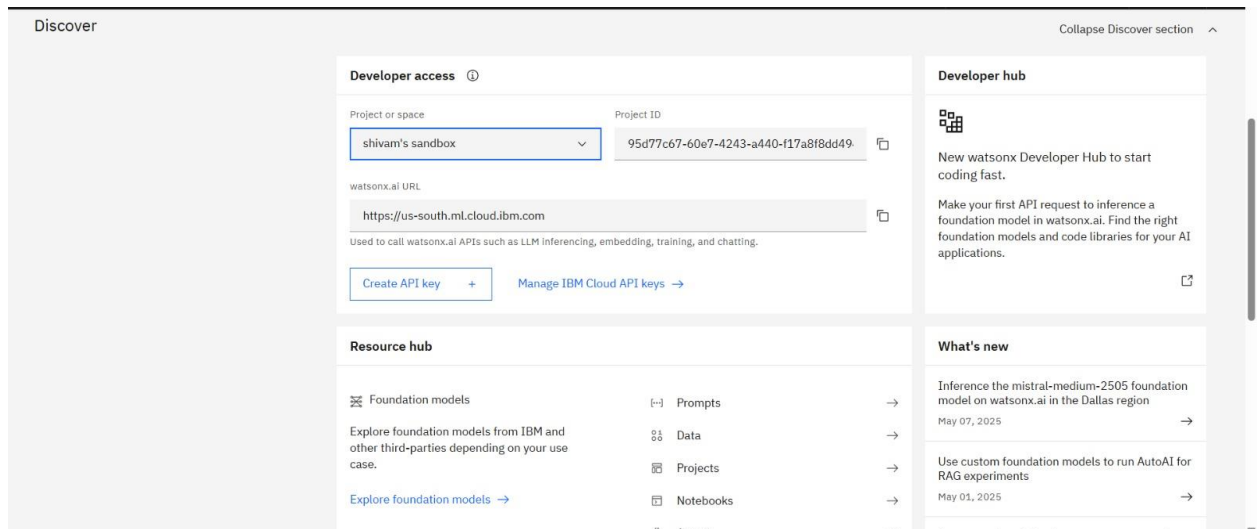2. Then, Launch watsonx.ai
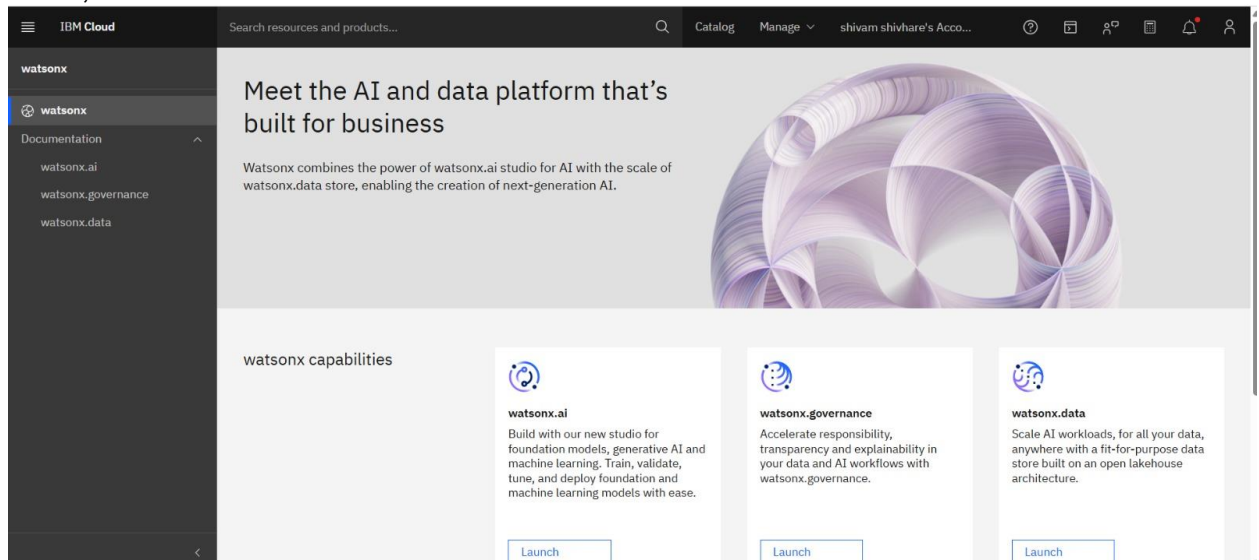
3. Here you can create your API
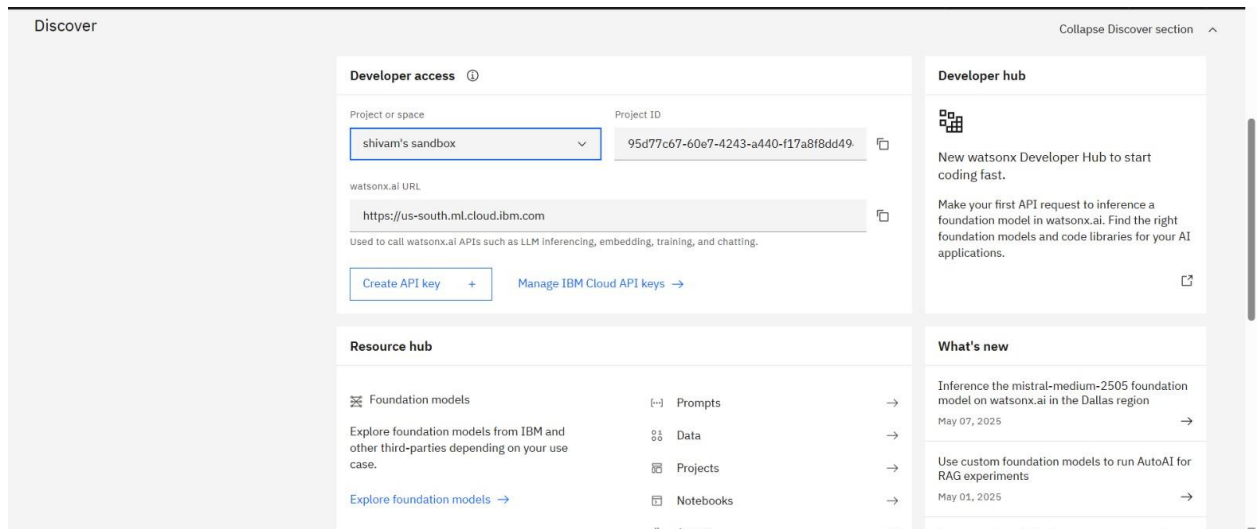
Setup and Access your IBM WatsonX API key.

1. Go to IBM WatsonX
2. Then, Launch watsonx.ai

3. Here you can create your API

Set up the development environment

1. Install Python and Pip: Ensure Python is installed along with pip for managing dependencies.
2. Create a Virtual Environment: Set up a virtual environment to isolate project dependencies.
3. **Install Required Libraries:**
    pip install streamlit pandas numpy plotly ibm-watson-machine-learning python-dotenv

4. Set Up Application Structure: Create the initial directory structure for the HealthAI application.

Design and develop the user interface

1. **Main Application Layout:**
    o Configure page title, icon, and layout preferences
    o Implement a sidebar for patient profiles and feature selection
    o Create custom CSS for enhanced visual appearance
2. **Feature-Specific Interfaces:**
    o **Patient Chat:** Chat-style interface with message history
    o **Disease Prediction:** Symptom input form and prediction display
    o **Treatment Plans:** Condition input and treatment plan output
    o **Health Analytics:** Interactive charts and metrics summary

Deployment

Milestone 6: functional testing and verify

Patient Chat Page

Description:

Here user have access to responsive healthcare communication platform enabling seamless dialogue about wellness concerns, with chronological message tracking for context retention. The system delivers intelligent, algorithmically-generated wellness guidance while providing verifiable medical insights supported by authoritative healthcare sources, creating a comprehensive virtual consultation experience.

Patient Chat output:

Disease Prediction Page:

Description: The webpage presents sophisticated diagnostic assessment tool featuring an intuitive symptom documentation section where users can detail their health concerns in depth. The interface includes a prominent analysis initiation control that, when activated, processes entered data to generate a comprehensive analysis of possible medical conditions. Results are presented with statistical probability indicators and detailed explanatory content for each potential diagnosis. The system incorporates essential medical advisories emphasizing the informational nature of the analysis and recommending professional healthcare consultation.

Disease Prediction Output:

Treatment Plans Page:

Description: This page contains medical condition entry area where users can specify their health concerns, complemented by a prominent action button for plan creation. Upon activation, the system produces comprehensive, organized therapeutic recommendations tailored to the specified condition. The interface includes essential medical advisories clarifying the informational nature of the provided guidance.

Health Analytics Dashboard:

Description: The interface features dynamic visualizations including heart rate trends, blood pressure patterns, and blood glucose fluctuations over time. A color-coded pie chart illustrates symptom occurrence frequency, while the metrics summary section provides key health indicators with directional trend markers. The dashboard is enhanced with AI-powered insights that analyze collected data to offer personalized health recommendations and observations.

Conclusion

The HealthAI project effectively demonstrates the potential of AI in revolutionizing healthcare assistance. By integrating IBM's Granite language model, the platform enables users to receive personalized health insights through Patient Chat, Disease Prediction, Treatment Plan Generation, and Health Analytics, making healthcare information more accessible.

Utilizing IBM Watson Machine Learning, the application ensures accurate health question answering, detailed disease prediction, personalized treatment recommendations, and insightful health trend analysis. The structured development process—spanning model selection, core feature implementation, backend and frontend development, and deployment—led to the creation of an interactive, user-friendly platform.

Built with Streamlit, HealthAI facilitates seamless visualization of health data and AI-generated insights, ensuring an efficient and responsive experience. This project highlights how targeted AI models and a well-structured framework can enhance healthcare accessibility. With future scalability in mind, HealthAI has the potential to expand its capabilities, incorporating more advanced diagnostics and broader medical applications.