**Tools, Technologies & Services Used**

To build this AI health agent, I used a combination of IBM’s easy-to-use AI tools and cloud services. The goal was to make something helpful, ethical, and accessible — without needing to write complex code. Here’s how I brought it all together:

**1. IBM Granite LLM – The Brain of the Agent**

At the heart of the chatbot is **IBM Granite**, a large language model that understands everyday language. When someone says something like “I have a sore throat and feel weak,” the model interprets it and gives back a helpful, easy-to-understand response — without pretending to be a doctor.

The core AI model that understands and responds like a helpful assistant.

* Interprets everyday symptom descriptions in natural language.
* Gives friendly, fact-based responses without making medical claims.
* Helps the agent feel more conversational and human-like.
* Built by IBM for responsible and ethical AI use.

**2. Watsonx.ai Studio – Where Everything Comes Together**

I used **Watsonx.ai Studio**, a no-code platform from IBM, to actually build and manage the chatbot. It gave me a simple interface to design how the agent behaves, add helpful tools, and test everything — no programming required.

The no-code platform where the agent was built and tested.

* Allowed me to build the entire agent visually, without coding.
* Provided an interface to add tools, prompts, and test responses.
* Integrated easily with IBM’s Granite LLM.
* Offers preview, save, and deploy features in just a few clicks.

**3. RAG (Retrieval-Augmented Generation) – Smarter Answers**

To make sure the agent gives better and more trustworthy advice, I used a method called **RAG**. It allows the agent to pull information from real documents (like WHO health guides or PDFs I upload) and answer questions using that content. So instead of guessing, it grounds its answers in real facts.

Used to give smarter, more grounded answers using real documents.

* Helps the AI pull info from PDFs and reliable sources like WHO.
* Reduces the chance of wrong or vague answers.
* Keeps the chatbot factual and trustworthy.
* Works especially well for health-related documents and reports.

**4. Vector Store – Letting the AI "Remember" Documents**

Whenever I upload PDFs (like a health report), the content gets turned into numbers and stored in something called a **vector store**. This lets the agent search through the documents quickly and respond based on what’s written inside — without ever storing sensitive info permanently.

Stores embedded document content so the AI can refer to it intelligently.

* Transforms uploaded PDFs into searchable data.
* Lets the AI "understand" and respond using your documents.
* Keeps data secure and doesn’t store it permanently.
* Supports use cases like reading health reports or symptom guides.

**5. IBM Cloud Lite – Hosting Everything, Free and Easy**

I used **IBM Cloud Lite** to host everything — and it’s completely free! It supports all the services I needed, like the AI model, storage, and deployment space, so I could make the project live without worrying about cost or setup.

Free cloud platform that runs everything behind the scenes.

* Hosts the AI model, vector data, and deployment.
* Completely free under the Lite plan — ideal for students.
* No setup or server configuration required.
* Makes the agent easy to access and scale later.

**6. Streamlit / Watson Assistant – How Users Talk to the AI**

To let people use the chatbot, I provided a simple interface. I either used **Watson Assistant’s built-in preview link** (just click and chat!) or built a basic **Streamlit** app. Both are friendly and work well on phones or laptops.

How users actually talk to the agent.

* **Watson Assistant** offers a ready-to-use web chat interface.
* **Streamlit** (optional) gives more control over design and layout.
* Both allow symptom input in plain language.
* Makes the chatbot accessible through phones, tablets, or PCs.

**7. Built-In NLP – Understanding Human Language Naturally**

Thanks to Granite, the chatbot doesn’t need structured inputs. People can type naturally, like they would to a friend, and it still understands. This makes it easier for anyone to use — especially those not used to tech.

What allows the agent to understand human phrasing.

* Understands phrases like “I feel weak and have a headache.”
* Doesn’t need keyword-specific inputs or structured queries.
* Built-in with Granite and enhanced by custom prompts.
* Improves accessibility for users of all backgrounds.

**Final Thoughts**

The entire system was built to be simple, ethical, and accessible. With no code, free tools, and a lot of care in the design, this project shows how technology can support real people in understanding their health better — safely and responsibly.