AGENT AI PROJECT

MACHINE FAULT DIAGNOSIS AGENT

Presented By:
1. P. SHIAMILEE

edunet foundation

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

Example: Currently Machine has become fault while it was working fine. This agent helps detect faults in machines like lathes, mills or pumps based on vibrations, temperature, or unusual noises. It can answer: “Why is my CNC machine vibrating too much?” or “What could cause overheating in a hydraulic pump?”. It suggests basic maintenance actions and safety precautions.

PROPOSED SOLUTION

- The proposed system aims to address the challenge of detecting faults in industrial machines (lathes, mills or pumps) in real time to reduce downtime and to optimize maintenance. This involves leveraging sensor data, analytics and machine learning techniques to accurately identify fault patterns. The solution will consist of the following components:
- Data Collection:
  - Present multiple-choice options (e.g., type of machine: Lathe, Milling, CNC) using a digression list to narrow down machine type.
  - Gather historical machine data, including sensor readings (e.g., vibration, temperature, pressure), operational status, and maintenance logs.
- Data Preprocessing:
  - Design the agent to collect structured user inputs through multiple choice questions (eg. Machine-type, fault symptoms) to reduce ambiguity.
  - Mapped each response directly to the internal decision rules to identify possible fault types without the need for additional data transformation.
- Agentic AI:
  - The principles which Agentic AI is by enabling the agent to autonomously guide users through a fault diagnosis workflow, ask the right questions, and make decisions based on the response.
  - This makes the system more interactive, explainable, and efficient, aligning with Agentic AI design where the agent functions as a task solving assistant.
- Deployment:
  - Publish the agent to IBM Watsonx Assistant or embed it in a web interface used by factory technicians.
  - Ensure accessibility through Desktop/tablet devices in control rooms or workshop floors.
- Evaluation:
  - Use Agent Lab's built in Analytics to track most common fault categories and its solution.
  - Refine the agent flow based on feedback and retrain intents as needed.

SYSTEM APPROACH

The "System Approach" section outlines the overall strategy and methodology for developing and implementing the rental bike prediction system. Here's a suggested structure for this section:

- The System uses a conversational flow – based design in IBM Agent Lab, where users interact with the agent to report machine type and symptoms through guided inputs.
- The agent applies rule-based decision approach using stored variables and conditions to identify likely fault types and provide relevant troubleshooting suggestions.

ALGORITHM & DEPLOYMENT

- In the Algorithm section, describe the machine learning algorithm chosen for predicting bike counts. Here's an example structure for this section:
- **Algorithm Selection:**
  - This system does not use traditional ML algorithm but follows a rule based decision system built using IBM Agent Lab.
- **Data Input:**
  - It uses user – reported machine types (lathes, mills or pumps) and fault symptoms (vibrating, noise, etc) and these are stored in Agentic lab variables.
- **Training Process:**
  - As a rule-based system, there is no training phase. The logic was manually designed by domain experts who mapped symptoms to possible faults. However, the agent can be improved iteratively by analyzing session logs and feedback.
- **Prediction Process:**
  - Based on user inputs, the agent analyses to match symptoms to a fault category (e.g., misalignment, bearing wear). The agent then responds with relevant advice or troubleshooting steps.

RESULT

The developed Machine Fault Detection Agent successfully guides users through a structured conversational flow to identify likely machine faults based on input symptoms and machine type. It provides relevant fault suggestions and troubleshooting steps in real time, offering a user-friendly and efficient solution without requiring manual diagnosis or coding.

CONCLUSION

- In conclusion, the Machine Fault Detection Agent developed using IBM Cloud Agent Lab offers a practical and interactive solution for diagnosing common machine issues in real time. By leveraging a rule-based, agentic AI approach, the system simplifies fault identification, enhances technician support, and reduces downtime. Its no-code structure ensures ease of deployment and adaptability, making it a valuable foundation for future integration with sensor data, machine learning, and advanced analytics.

FUTURE SCOPE

- In the future, the system can be enhanced by integrating real-time IoT sensor data to enable automatic fault detection without user input. Machine learning models can be introduced to analyze patterns in historical fault data, improving accuracy and enabling predictive maintenance. Additionally, multilingual support, voice interaction can further expand the agent's usability and impact in industrial environments.

REFERENCES

1. Zhang, W., Yang, D., & Wang, H. (2019). Data-Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey. *IEEE Systems Journal, 13(3), 2213–2227.Provided background on rule-based vs. data-driven fault detection methods.*

2. Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015). Machine Learning for Predictive Maintenance: A Multiple Classifier Approach. *IEEE Transactions on Industrial Informatics, 11(3), 812–820.Offered insights into categorizing machine faults based on symptoms*.

3. Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th ed.). Pearson. *Used for understanding the agentic AI principles applied in dialogue-based systems.*

4. IBM Cloud Agent Lab Overview. https://www.ibm.com/cloud/watsonx/agent
Provided the foundational environment for building the no-code fault detection agent.

5. IBM Watsonx Assistant Documentation. https://www.ibm.com/docs/en/watsonx-asst
Used to understand agent configuration, input handling, and deployment in IBM Agent Lab.

IBM CERTIFICATIONS

- Screenshot/ credly certificate(getting started with AI)

edunet foundation

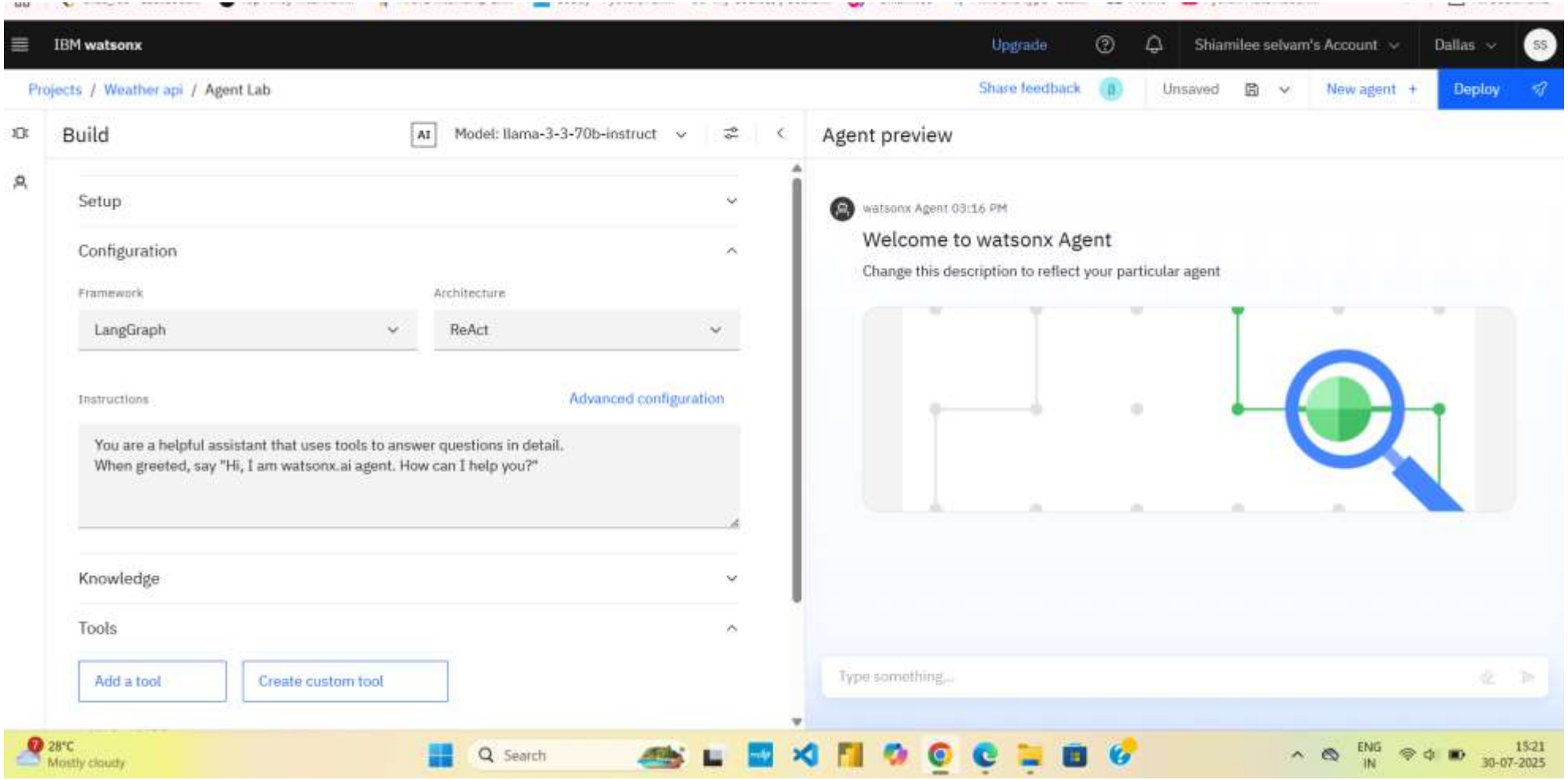
IBM CERTIFICATIONS

- Screenshot/ credly certificate(Journey to Cloud)

IBM CERTIFICATIONS

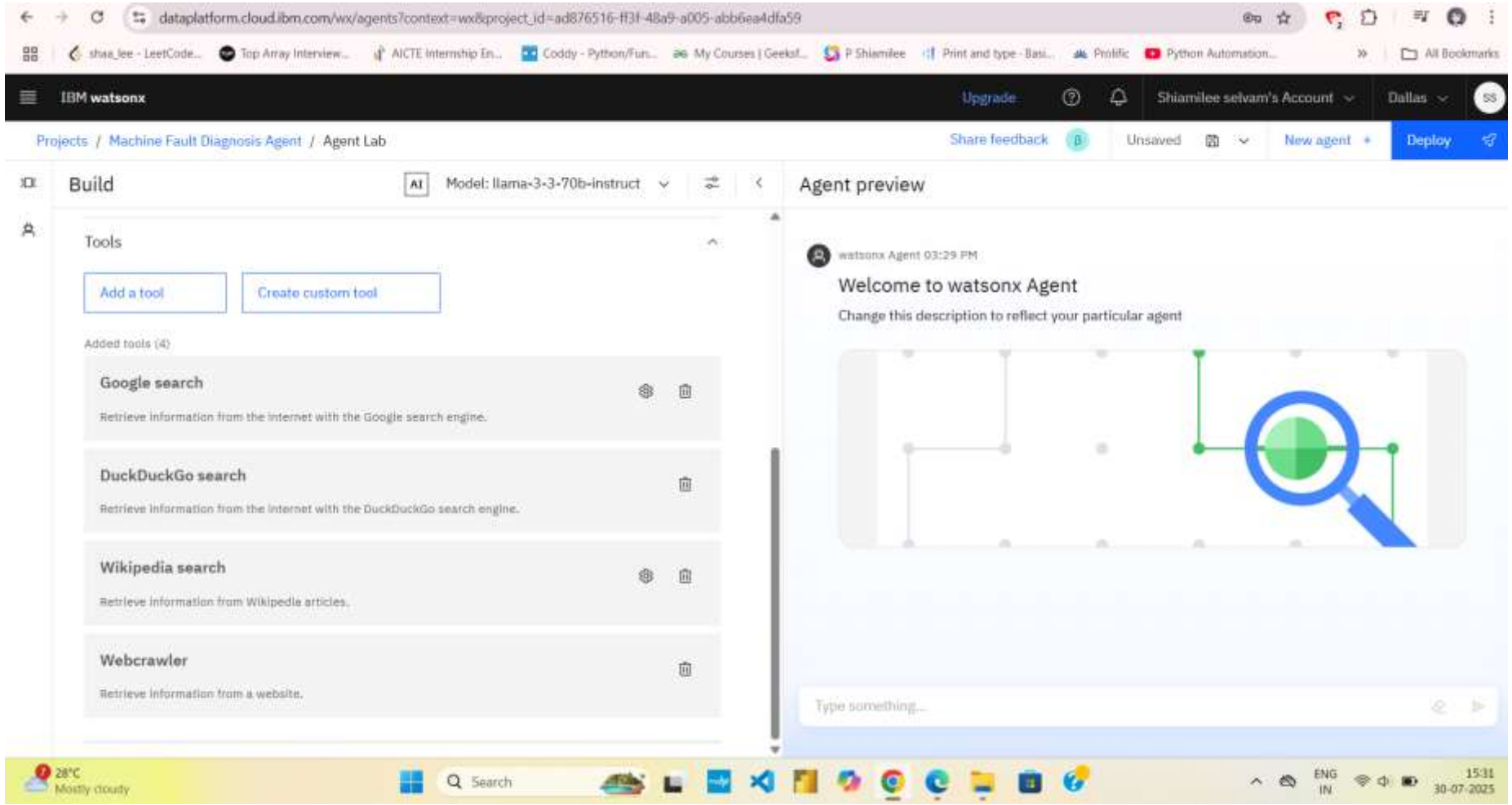
- Screenshot/ credly certificate(RAG Lab)

IBM PROJECT SCREENSHOTS – SETTING UP A AGENT LAB



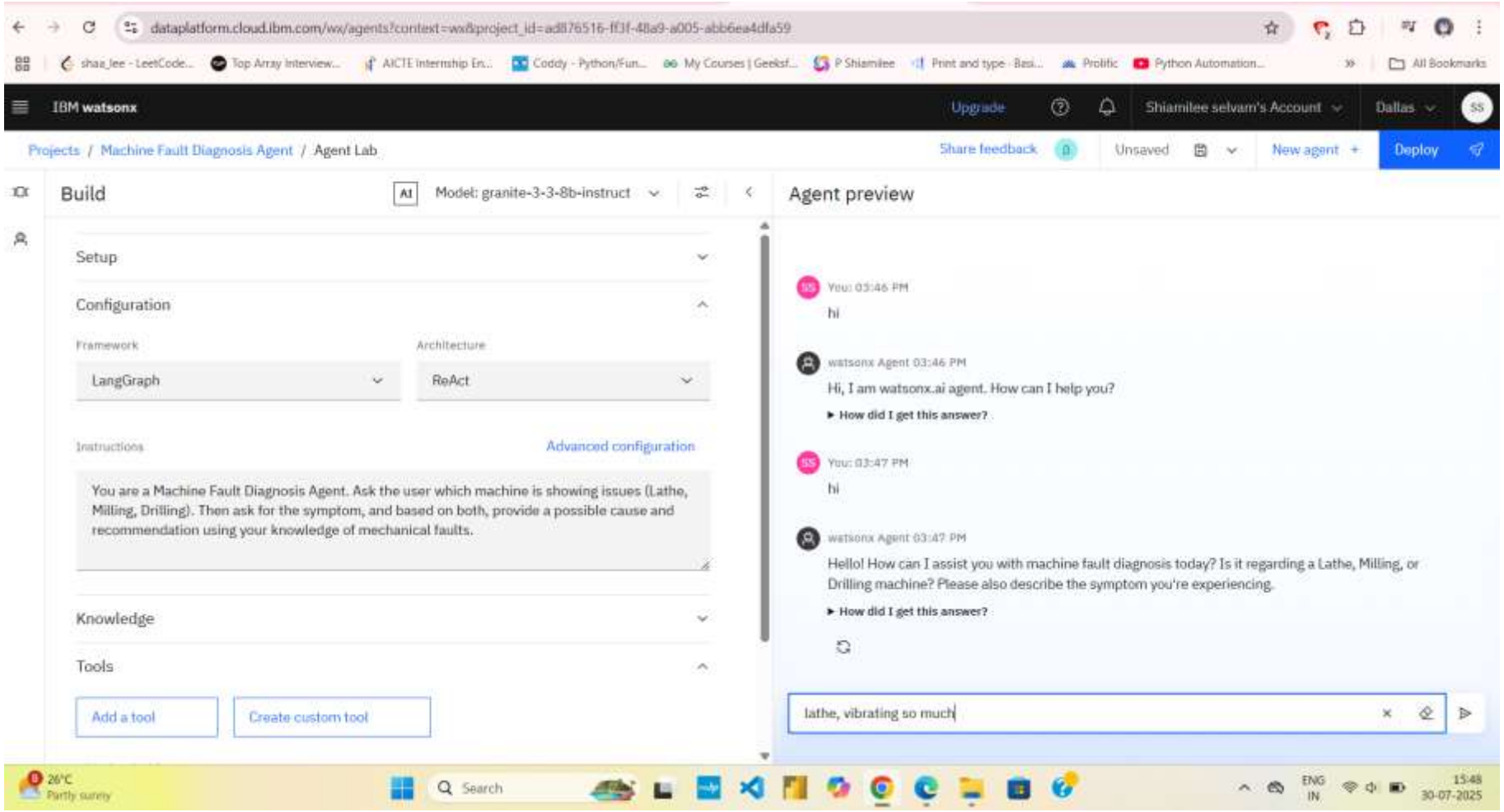
edunet foundation

UPDATING THE TOOLS



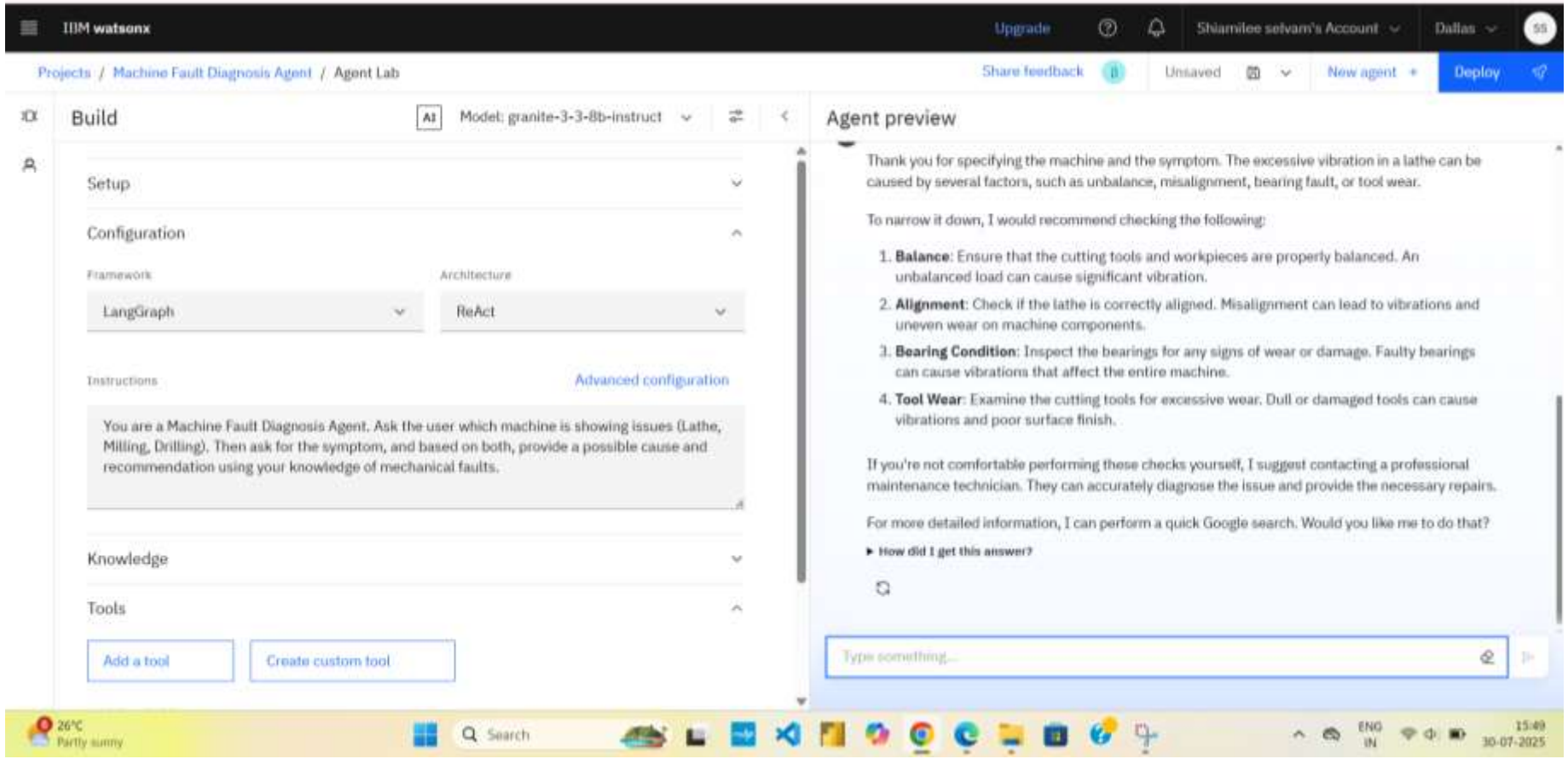
edunet foundation

INSTRUCTING THE AGENT & ENTERING PROMPT AS USER



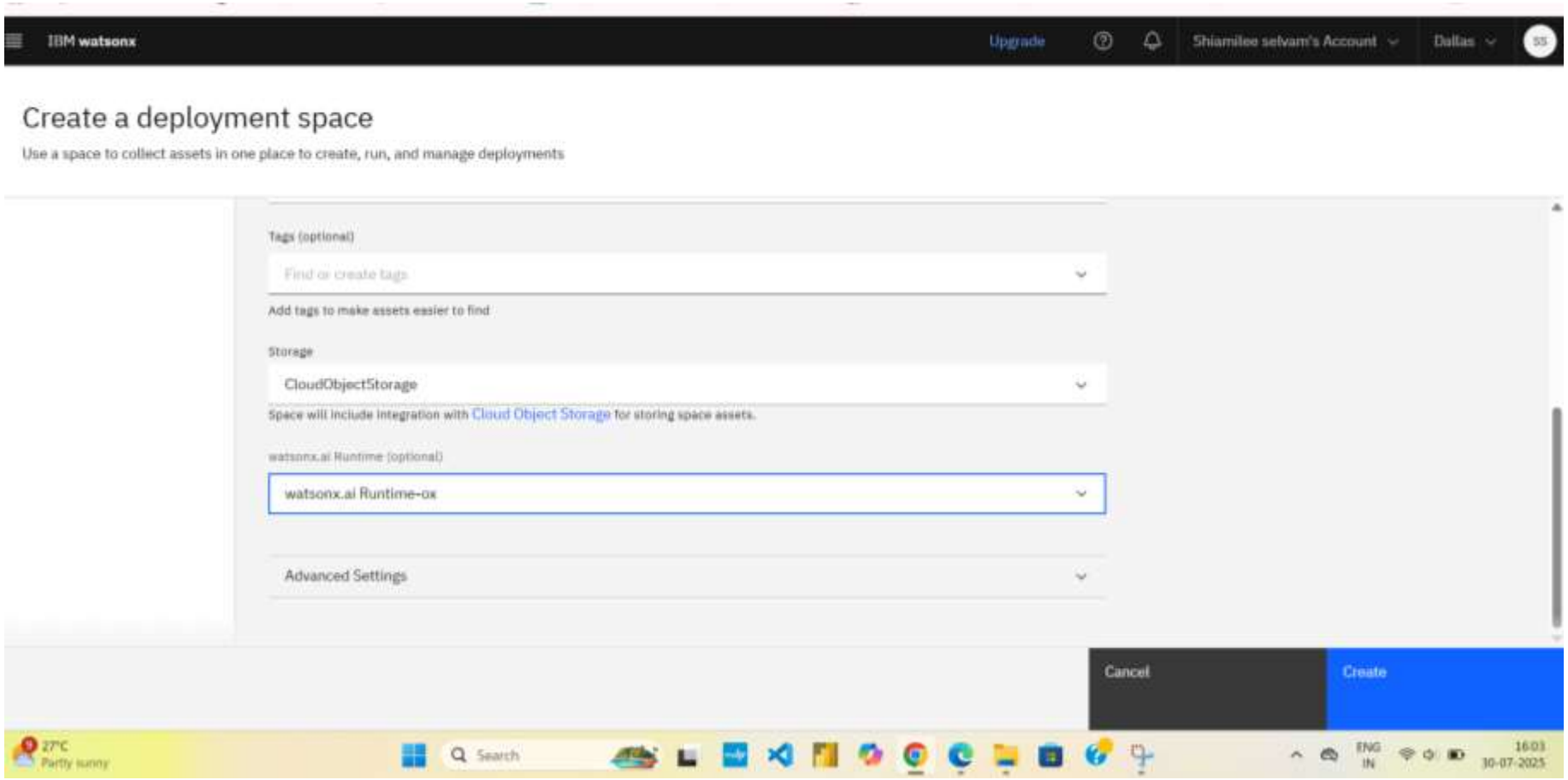
edunet foundation

SOLUTION



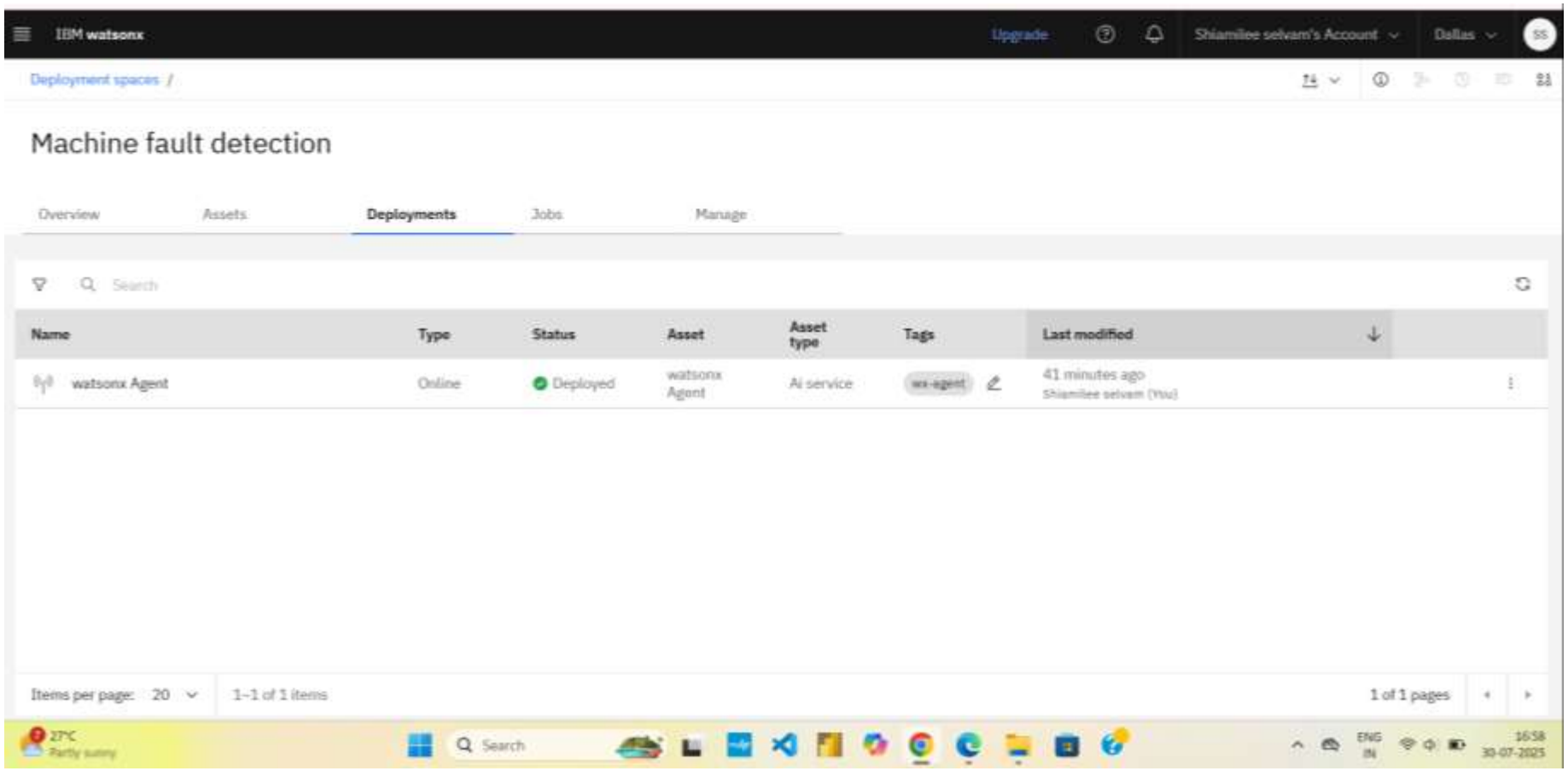
edunet foundation

DEPLOYMENT



edunet foundation

SUCCESSFULLY DEPLOYED



edunet foundation

THANK YOU

edunet foundation