You're referring to a newer feature called **"Goose"** and its integration with **DeepSeek LLM**, using **Ollama API** to enable remote task execution through the **"tool calling"** feature. Let's break down what this means and how it would work in your scenario.

1. What is the Goose AI Agent?

The **Goose** AI agent is a new tool within the Ollama ecosystem that allows AI models (like **DeepSeek LLM**) to call and interact with external tools or resources from a remote environment. This is often referred to as **tool calling**. In this case, the Goose agent is designed to execute **tasks remotely** based on instructions sent by a server, while being able to use the power of an AI model like **DeepSeek** to carry out tasks on a local machine.

Key Features of Goose:

- **Remote Execution**: Goose is able to execute tasks locally on the remote machine.
- **Tool Integration**: It integrates with tools or systems on the local workstation, such as interacting with file systems, executing shell commands, creating users, and more.
- **Server-Client Interaction**: A server running DeepSeek LLM (hosted via Ollama API) can send tasks to the Goose agent on the remote workstation, allowing the agent to perform various local tasks.

2. Tool Calling Feature

The **"tool calling"** feature in Goose allows AI agents to interact with the underlying operating system, local programs, and services directly. Essentially, it expands the capabilities of the LLM beyond just answering questions—giving it the ability to execute tasks such as:

- Running scripts or shell commands
- Managing system resources (creating users, managing files)
- Accessing and modifying local databases
- Interacting with applications installed on the workstation

This means that you could **send a command to the Goose agent** that is running on a **remote workstation**, and the Goose agent will carry out that command locally.

3. How Does It Work with DeepSeek LLM and Ollama API?

The workflow could be structured as follows:

- **Step 1: Server Setup (DeepSeek LLM & Ollama API)**:
    - You have a server that is hosting **DeepSeek LLM** (via Ollama API). This server is responsible for handling complex AI queries or higher-level tasks.
    - The Ollama API is configured to interact with both the LLM and any tools or agents (such as Goose) on the remote workstations.
- **Step 2: Create and Push the Goose Agent**:

- On the remote workstation, you would use the ollama create and ollama push commands to deploy the **Goose agent**.
- Example command:
- ollama create michaelneale/deepseek-r1-goose -f Modelfile
- ollama push michaelneale/deepseek-r1-goose

This will push the Goose agent to your remote workstation.

- **Step 3: Task Sending**:
  - The server (hosting DeepSeek LLM) will now be able to send instructions or tasks to the **Goose agent** on the remote workstation.
  - The tasks could be anything that the Goose agent can execute locally, such as creating a user, fetching files, running scripts, etc.
- **Step 4: Remote Execution**:
  - The Goose agent receives the tasks and executes them on the remote workstation.
  - The results of those tasks (e.g., command output, user creation success, file modification) are sent back to the server or reported back to the AI system.

4. Example Use Case: Running Goose on a Remote Workstation

Let's break it down with a concrete example:

- **Scenario**: You want to automate user creation on multiple remote workstations. You have a server running **DeepSeek LLM** and the **Ollama API**, and you want to use the Goose agent to execute the task of creating a user on a remote machine.

Workflow:

1. **Server Query**:
   - The user sends a request to the server (DeepSeek LLM via Ollama API) asking to create a new user on the remote workstation.
2. **DeepSeek Response**:
   - The server (running DeepSeek LLM) processes the request and identifies that it requires executing a task on a remote machine.
3. **Task Delegation to Goose**:
   - The server sends a task to the **Goose agent** running on the **remote workstation**: "Create a new user named john_doe."
4. **Goose Agent Execution**:

- The Goose agent on the remote machine receives the task and executes the required command, such as:
- sudo useradd john_doe
- The agent then sends back a response indicating success or failure.

5. **Server Response**:
    - The server processes the result of the task (e.g., "User john_doe created successfully") and sends it back to the user or other systems that requested the task.

5. Key Benefits of Using Goose with DeepSeek LLM

- **Remote Task Execution**: With Goose, you can have the LLM send tasks to be executed on remote workstations. This is powerful for managing multiple systems or automating administrative tasks across a network of machines.
- **Flexibility**: The Goose agent can execute a wide range of local system tasks (e.g., managing files, users, running commands) based on the LLM's instructions. This makes it suitable for automating administrative and operational tasks.
- **Separation of Concerns**: DeepSeek LLM handles the higher-level AI tasks (natural language understanding, processing), while Goose handles the local execution of tasks, allowing you to separate the AI's decision-making from the actual execution of commands.

6. Practical Considerations

- **Security**: You must ensure that the remote workstations where Goose agents are deployed are properly secured. Since Goose can execute local system tasks, you should implement authentication, encryption, and other security measures to prevent unauthorized access.
- **Error Handling**: Ensure that there is a robust mechanism for handling failures (e.g., if a task cannot be completed on the remote workstation, the server should be notified with appropriate error messages).
- **Scalability**: If you have many workstations, managing all the Goose agents and making sure the server sends tasks to the right agent can get complex. Consider building a management layer for scaling the system across multiple machines.

Summary:

- Yes, with the **Goose AI agent** and **DeepSeek LLM**, you could run the Goose agent on **remote workstations**, and have it execute tasks (like system administration tasks) that are sent from a **server** hosting the **DeepSeek LLM** and **Ollama API**.
- The **Goose agent** on each remote workstation listens for commands from the server and executes local tasks like user creation, running shell commands, managing files, etc.

- The server, using **DeepSeek LLM**, processes high-level AI queries and then delegates task execution to the Goose agents on remote workstations using the **tool calling** feature.

Let me know if you'd like more detail on specific parts of the setup!