# Summary of Previous Deliverable

In the previous proof of concept deliverable for the chatbot integrated ticketing system, several key steps were carried out.

1. **Setting up the Development Environment**

* Installation of Python on local machine.
* Set up Flask for creating a simple web server.
* Utilisation of Requests library for handling of HTTP requests.

1. **Developing the Web Server**

* Created a simple Flask server with a single endpoint, “/webhook”, to manage incoming POST requests.
* Configured the endpoint to expect JSON data containing a user message and to respond with a placeholder message.

1. **Established Internet Connectivity**

* Used Ngrook to expose the local PC to the internet, specifically targeting port 5000.
* Followed the command “Ngrook http 5000” to set up this connection.

1. **Creating the Basic Chatbot**

* Developed a simple chatbot that can connect to the webhook endpoint.
* Ensured that the chatbot could simulate interaction with an IT ticketing system, demonstrating the basic connection needed to transfer ticketing information.

1. **Testing the Proof of Concept**

* Ran the chatbot (chatbot.py) with an input to test its functionality.
* Verified the interaction between the chatbot and the Flask web server through the “/webhook” endpoint.

1. **Finalising the Proof of Concept**

* Committed and pushed the working proof of concept to Azure DevOps for version control and further development.

The Proof of concept successfully demonstrated the fundamental feasibility of connecting a chatbot to a webhook endpoint, simulating interactions with an IT ticketing system. This laid the groundwork for further development and integration with real-world ticketing systems, leveraging technologies like Zendesk, Python and associated natural language processing libraries. The project aligns with current trends in AI-driven solutions for enhancing organisational processes and improving user experience in IT support.