



# Chatbot Design: Part 3



## Academic Advisor Chatbot

The chatbot is designed to support programme leaders and academic advisors in managing advising tasks efficiently. It serves as a quick-access tool for retrieving important programme information, guidelines, and consultation details—right when they're needed.

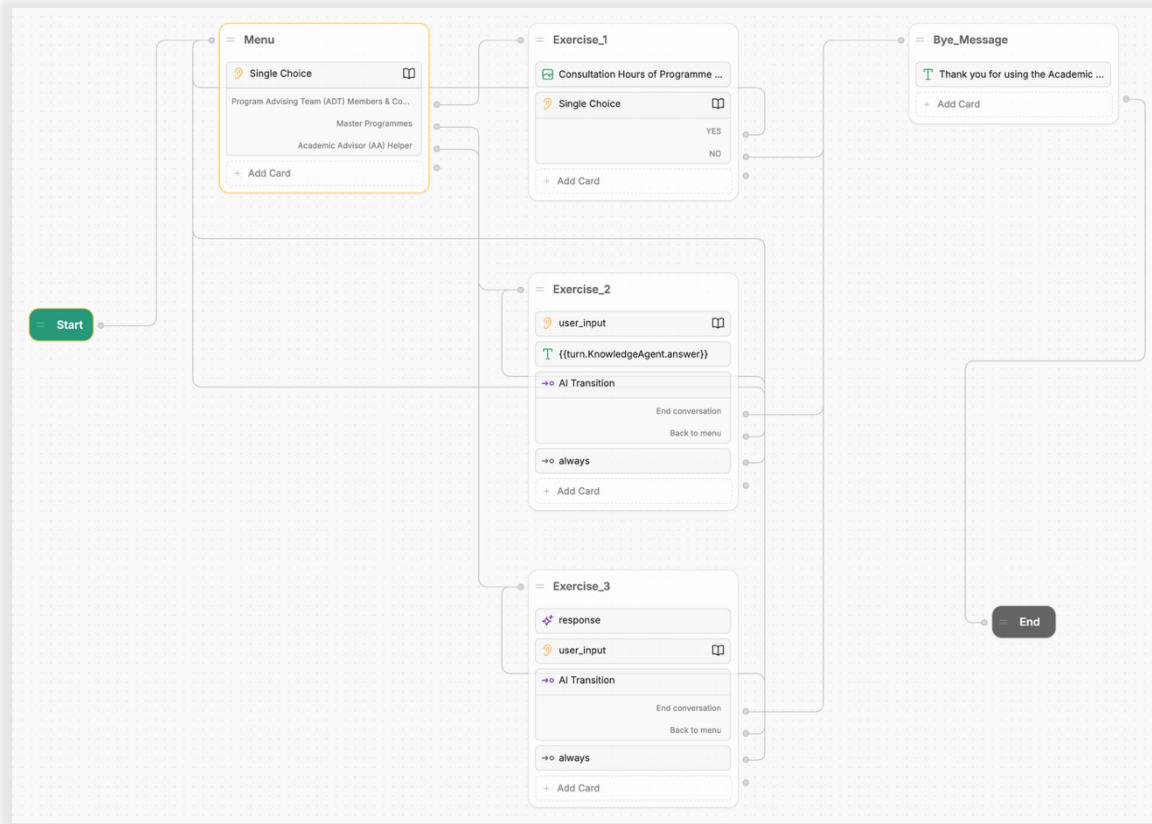
## Key Features

- Main Menu Navigation – Quickly access different advisor resources from a single interface.
- Consultation Hours Display – Instantly view the Programme Advising Team's schedule.
- Integrated Knowledge Bases – Search through curated resources like programme guidelines and study patterns.
- Conversational Memory – Understands context and remembers prior queries for natural follow-up questions.
- AI Transitions – Smoothly navigate between topics or return to the main menu based on user intent.



Want to learn Botpress?  
Check out my notes and tutorials here!

[https://linktr.ee/21kaw.botpress\\_tutorial](https://linktr.ee/21kaw.botpress_tutorial)



## Knowledge Base

### What is a Knowledge Base in Botpress?

In Botpress, a Knowledge Base is a central repository of information that your chatbot can access to answer user queries. It can store text from documents, website content, or other uploaded resources, enabling the bot to provide factual, consistent, and up-to-date responses without having to hard-code every answer into the flow.

### Why Use a Knowledge Base?

Instead of manually scripting every possible reply, the Knowledge Base allows you to simply upload relevant materials—such as PDFs, web pages, or plain text—and let the chatbot retrieve answers automatically. This is especially powerful for supporting dynamic, content-rich topics like academic advising, programme details, or institutional guidelines.



## How it Works?

When the bot receives a question, the Knowledge Agent searches the specified Knowledge Base for relevant information. It then returns the best-matching response, which can be shown to the user directly or processed through an AI Task for summarisation, rephrasing, or formatting.

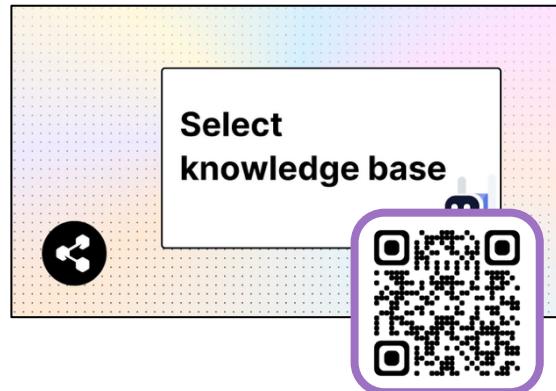


## Video Tutorial

### Select Knowledge Base

Learn how to choose a specific knowledge base for an AI card, ensuring your bot answers from the right set of documents or resources.

🔗 <https://youtu.be/DmOHpeswL6E>



### Integrate AI Task and Knowledge Base

See how to combine AI Tasks with a knowledge base to perform custom actions—like summarizing or formatting answers—based on the retrieved data.

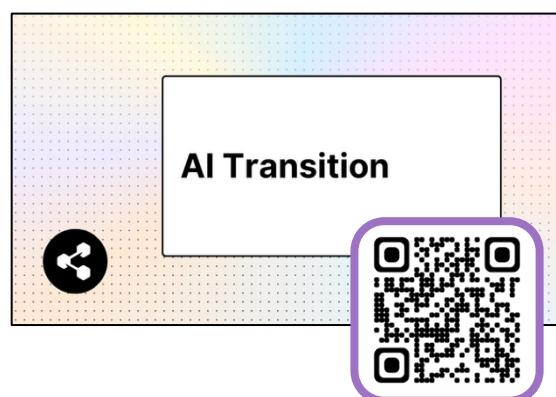
🔗 <https://youtu.be/z4dLcUffF8d0>



### AI Transition

Discover how to route conversations intelligently by using AI to interpret user input and automatically jump to the right node or flow. Meanwhile, our chatbot here focus on routed assessment and evaluation-based feedbacks.

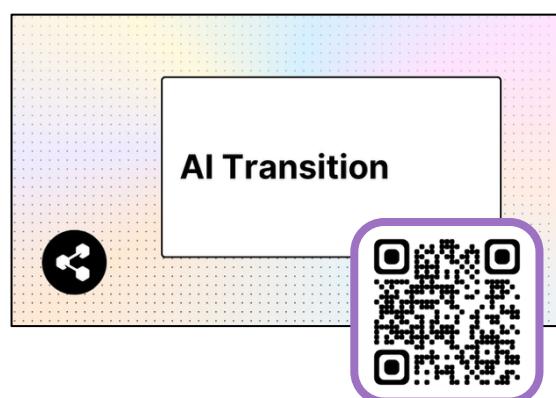
🔗 <https://youtu.be/ylgTqCorKlw>



### Create an AI Chatbot with Conversational Memory like ChatGPT

Understand how to make your chatbot remember previous messages, enabling context-aware conversations and natural follow-up interactions.

🔗 <https://youtu.be/p1-iBvEuTzA>





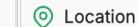
# Components

## ➤ Send Messages / Capture Information



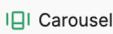
### Comment Card

- Purpose:** Allows adding internal notes or remarks within the chatbot workflow.
- Use Case:** Used for documentation purposes to help developers or collaborators understand specific parts of the workflow.



### Text/Image/Audio/Video/File/Location

- Purpose:** Displays multimedia content, including text, images, audio, video, files, or location data, to enhance user interaction.
- Use Case:** Can be used for delivering course-related images, lecture recordings, downloadable study materials, or location-based information.



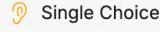
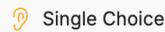
### Carousel Card

- Purpose:** Presents multiple cards in a horizontal scrollable format, allowing users to browse and select content.
- Use Case:** Used for displaying multiple course modules, recommended textbooks, or learning resources in an interactive format.



### Person Name / Email Address / Raw Input

- Purpose:** Captures user-provided text input, such as a name, email address, or any free-text response.
- Use Case:**
  - Person Name:** Collects a user's name for personalization.
  - Email Address:** Captures the user's email for follow-ups or authentication.
  - Raw Input:** Accepts open-ended text responses for flexible user input.

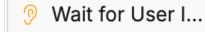


I have a question

End conversation

### Inputs: Single Choice Card

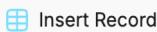
- Purpose:** Presents single/multiple-choice options for user navigation.
- Use Case:** Allows users to select answer, such as "A. Increase productivity," "B. Climate focused investments," or "C. Digital Innovation."



### Wait for User Input Card

- Purpose:** Temporarily pauses the chatbot flow until the user provides additional input.
- Use Case:** Ideal for scenarios where the user needs time before continuing, such as confirming readiness or responding to an open-ended question.

## ➤ Execute



### Table: Insert Record

- Purpose:** Stores user-provided information into a database or external system.
- Use Case:** Used to log student details, chatbot interactions, or form submissions for tracking and data management.



### Execute code

- Purpose:** Allows custom JavaScript code execution within the chatbot flow to handle advanced logic or data manipulation.
- Use Case:** Typically used for advanced scoring calculations, conditional logic, or to manage complex interactions and decisions within the chatbot workflow.

## ➤ Nodes

### Start

#### Start Node

- **Purpose:** Kicks off the chatbot conversation and initializes the workflow.
- **Use Case:** Includes an initial greeting, menu introduction, or chatbot instructions to guide users into the conversation flow.

### End

#### End Node

- **Purpose:** Marks the end of a chatbot interaction, terminating the conversation flow.
- **Use Case:** Used when the chatbot has completed its task, ensuring a natural exit for the user or providing an option to restart the conversation.



#### Linking the Nodes

- **Purpose:** Connects different nodes to create a logical flow.
- **Use Case:** Ensures smooth transitions between menu options, user inputs, and responses, allowing structured navigation.

## ➤ Flow Logic

### →○ Expression

#### Expression (Condition)

- **Purpose:** Enables the use of dynamic expressions to control chatbot logic and process data.
- **Use Case:** Used to create conditional responses, personalize user interactions, or automate chatbot decisions based on user input.

## ➤ AI

### ❖ AI Task

#### AI Task

- **Purpose:** Executes a specific AI-powered action, such as generating a custom response, summarizing a conversation, or transforming user input based on the provided prompt and context.
- **Use Case:** Ideal for scenarios where the bot needs to process information dynamically—e.g., maintaining conversational memory, drafting tailored replies, or performing complex data extraction before continuing the flow.

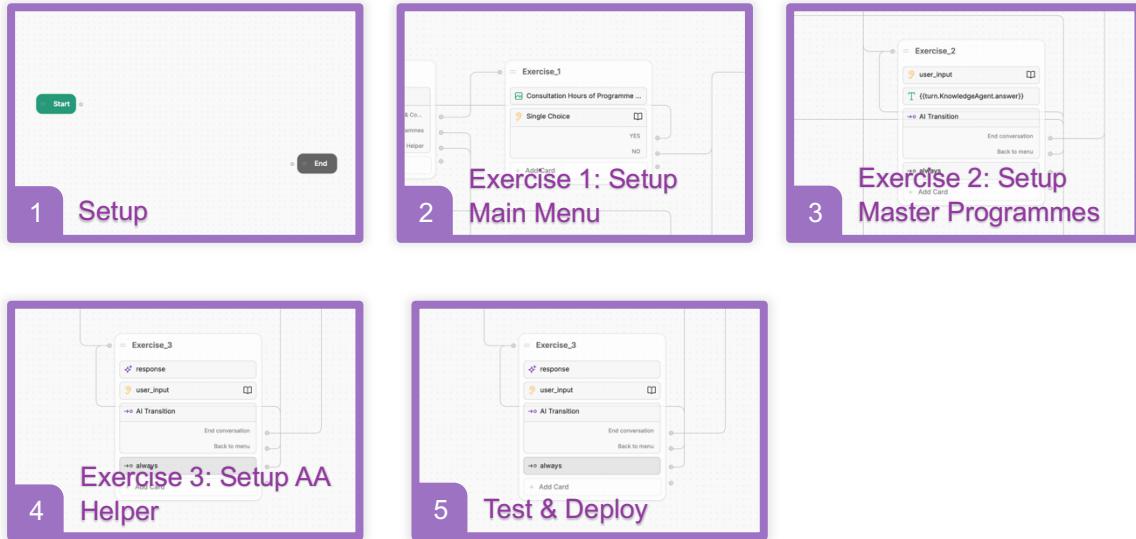
NEW

### →○ AI Transition

#### AI Transition

- **Purpose:** Uses AI-based logic to interpret user input and automatically routes to appropriate nodes based on specific keywords or intents.
- **Use Case:** Helps create flexible conversational flows, especially useful in waiting scenarios, to recognize when users express readiness, agreement, or specific commands.

## Quick Look



The workflow consists of five main parts:

### Exercise 1 – Create Main Menu (Recap from Chatbot Design: Part 1 – Flow-based)

- 1-A: Build Main Menu Node
- 1-B: Build Exercise\_1 Node
- 1-C: Build Bye\_Message Node
- 1-D: Connect the Nodes

### Exercise 2 – Master Programmes (Recap from Chatbot Design: Part 2 – Knowledge Base)

- 2-A: Create Master Programmes Knowledge Base
- 2-B: Set Knowledge Agent Mode
- 2-C: Build Exercise\_2 Node

### Exercise 3 – Academic Advisor Helper (Advanced Techniques)

- 3-A: Create AA Helper Knowledge Base
- 3-B: Build Exercise\_3 Node

### Test and Deploy

- **Follow the first guideline (Setting-up Course Support Chatbot on Botpress) for detailed steps on initial setup.**

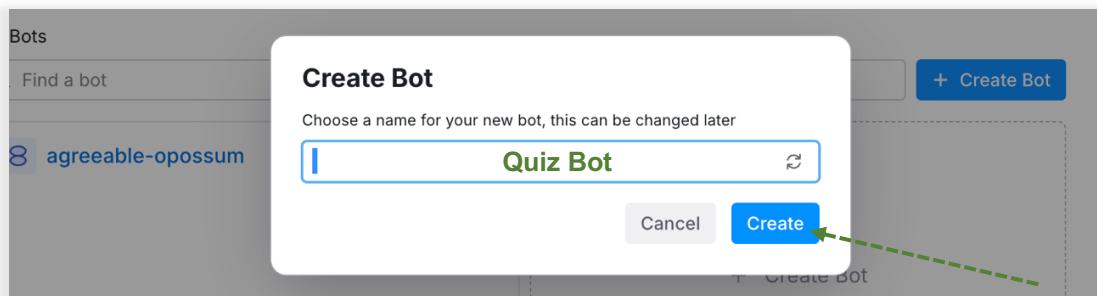


# Setup

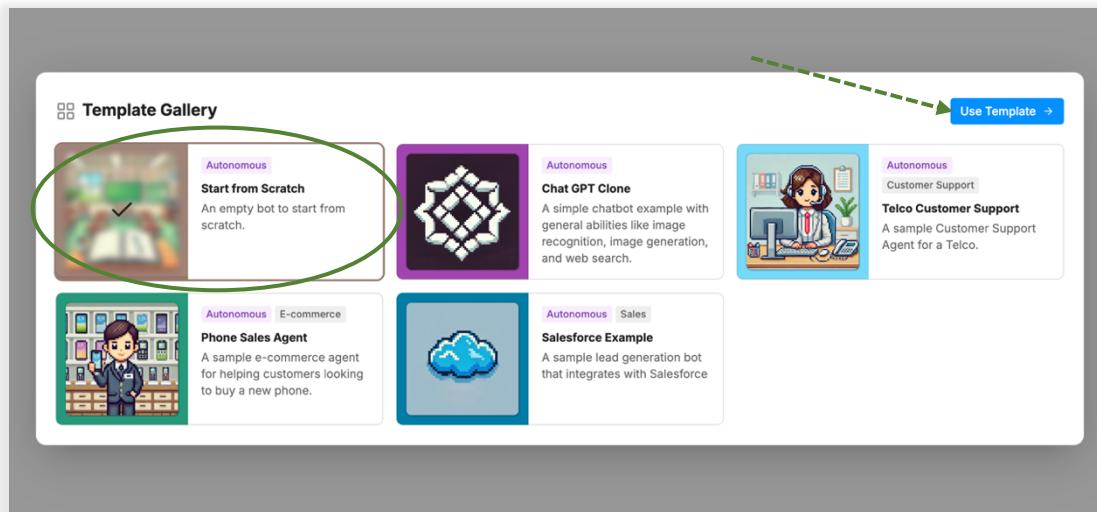
In this section, we'll guide you through creating a new bot project in Botpress Studio. You'll set up a new bot from scratch, and configure it, so you can start building the interaction flow.

## Start a New Bot

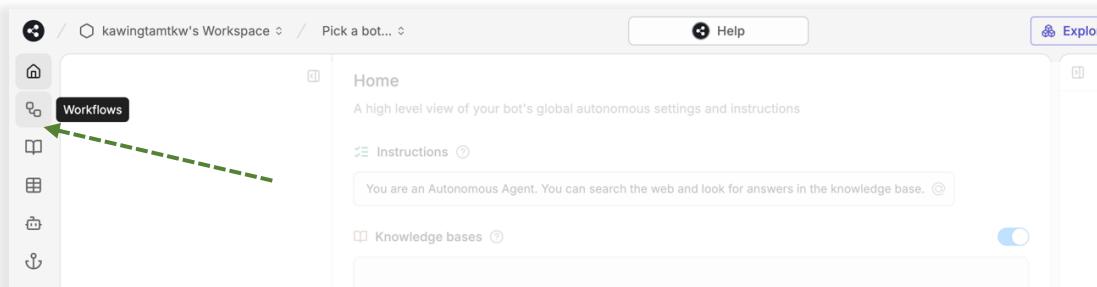
1. Open **Botpress**  [app.botpress.cloud](https://app.botpress.cloud) and create a new bot named **Quiz Bot**. Then, click on **Open in Studio**  **Open in Studio**.



2. From the **Template Gallery**, select **Start from Scratch**. Then click on **Use Template**  **Use Template**.

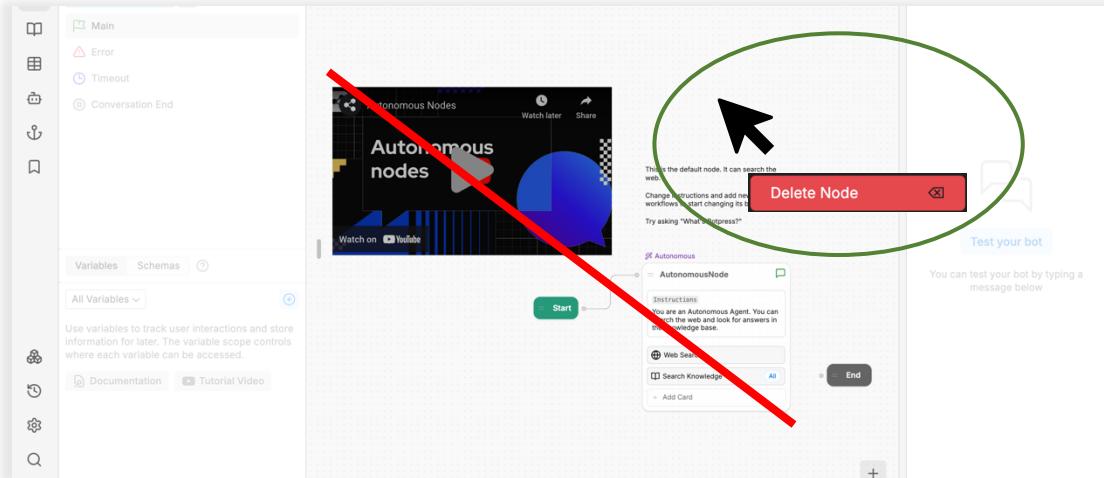


3. From the left menu, click on  **Workflows**.



## Remove Unnecessary Nodes

1. After the bot is generated, remove any unnecessary default nodes **Start** and **End** to maintain a clean workspace.



## Ensure Start and End Nodes Remain

1. The **Start** node will be the entry point of the bot, triggering when a user initiates a conversation.
2. The **End** node will terminate the conversation gracefully after responses are provided.



### End of Setup

Great job! You've created the bot. Let's move on to build the bot flow!



# Exercise 1 – Create Main Menu (Recap from Chatbot Design: Part 1 – Flow-based)

In this section, you'll design the chatbot's starting menu and build the complete flow for "Program Advising Team (ADT) Members & Consultation Hours".

- **What you'll learn:** How to create a Single Choice menu, link options to nodes, and display a table using an Image/Text card.
- **What the bot can do:** Present users with a clear menu, show the consultation hours table, and allow them to choose whether to return to the menu.

## 1-A: Build Main Menu Node

1. Rename the default node to Menu for clarity.
2. Inside Menu Node:

### 2.1. Add a Single Choice card

Question to ask the user: "Make a choice from the list below" (default)

Choices:

**Program Advising Team (ADT) Members & Consultation Hours**  
**Master Programmes**  
**Academic Advisor (AA) Helper**

The screenshot shows the Botpress interface with a 'Single Choice' card added to a 'Menu' node. The card displays three choices: 'Program Advising Team (ADT) Members & Consultation Hours', 'Master Programmes', and 'Academic Advisor (AA) Helper'. The 'Choices' section on the right lists these three items. The 'Included Knowledge Bases' section at the bottom shows 'All KBS selected'.

## 1-B: Build Exercise\_1 Node

3. Create a new node: **Exercise\_1**

4. Inside **Exercise\_1**:

4.1. Add an **Image** card:

**Image:** (Upload “Consultation Hours of Programme Advising Team Members.png”)

**Title:** Consultation Hours of Programme Advising Team Members

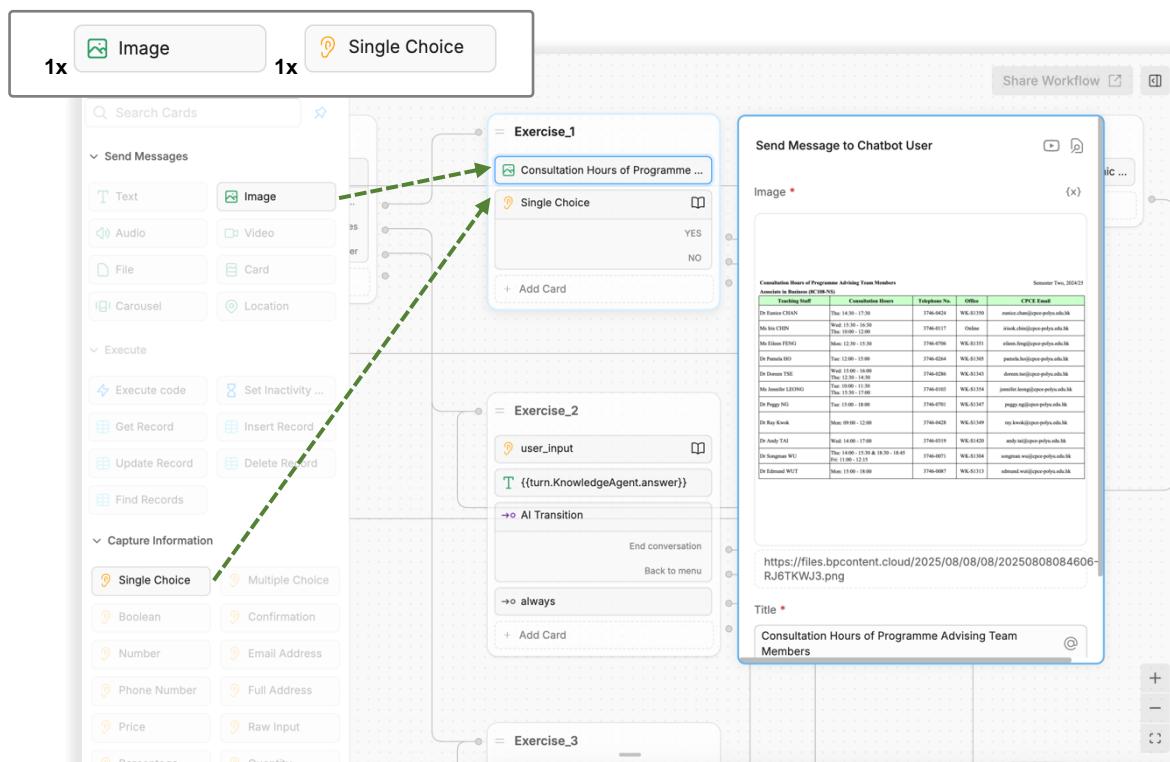
4.2. Add a **Single Choice** card

**Question to ask the user:** “Any further question?”

**Choices:**

**YES**

**NO**



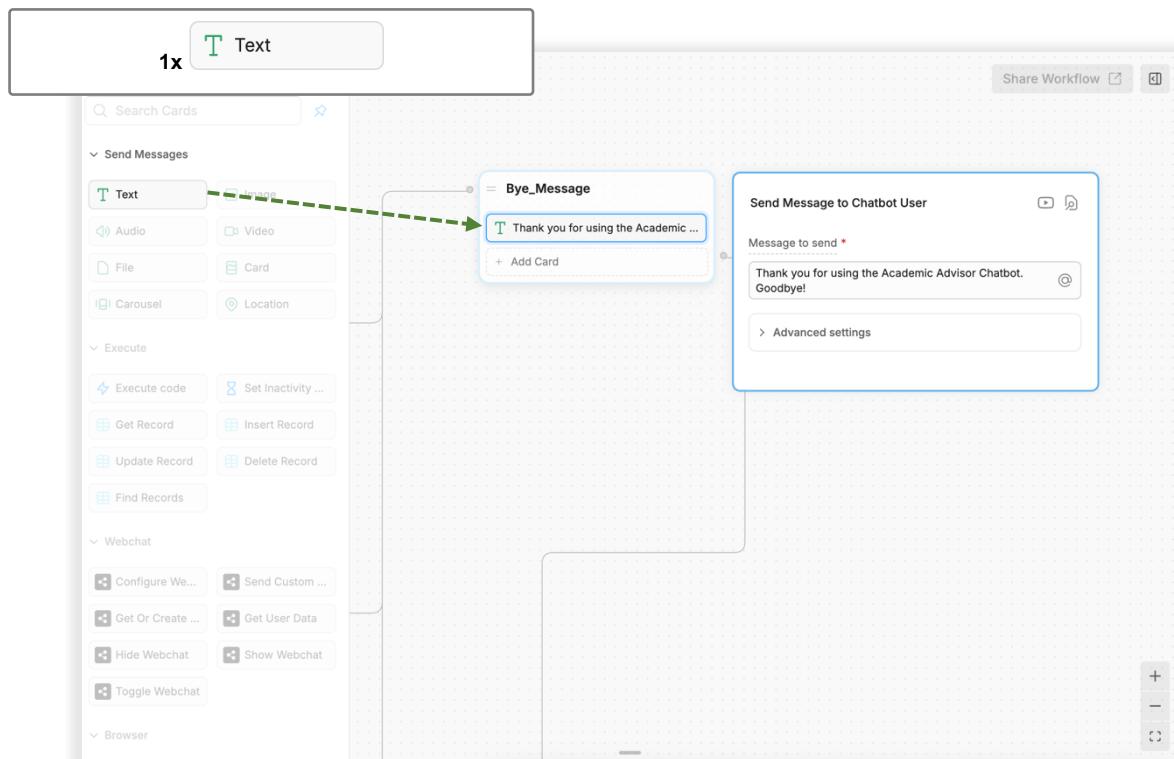
## 1-C: Build Bye\_Message Node

5. Create a new node: Bye\_Message

6. Inside Bye\_Message:

6.1. Add an **Text** card:

**Message to send:** Thank you for using the Academic Advisor Chatbot. Goodbye!



## 1-D: Connect the Nodes

7. Connect the end of option **Program Advising Team (ADT) Members & Consultation Hours** to node `Exercise_1`.
8. Connect the end of option **YES** to start of node `Menu`.
9. Connect the end of option **NO** to the start of node `Bye_Message`

### End of Exercise 1

💡 Great job! You've created the bot and set up the basic menu navigation and first function. Let's move on to integrating the Master Programmes knowledge base! 📚



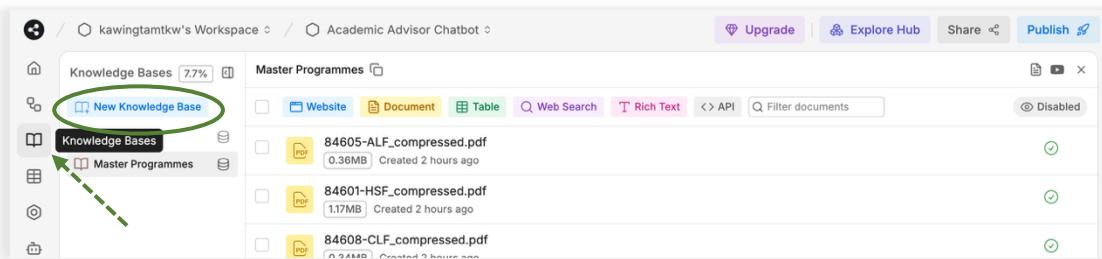
## Exercise 2 – Master Programmes (Recap from Chatbot Design: Part 2 – Knowledge Base)

In this section, you'll connect the bot to a Master Programmes knowledge base and set it to answer only when triggered.

- **What you'll learn:** How to add a website and PDFs to a knowledge base, configure the Knowledge Agent to “Answer Manually”, and display answers in a node. Besides, you will learn how to specify a knowledge base and use AI Transition.
- **What the bot can do:** Provide accurate information about Master Programmes from official SPEED resources.

### 2-A: Create Master Programmes Knowledge Base

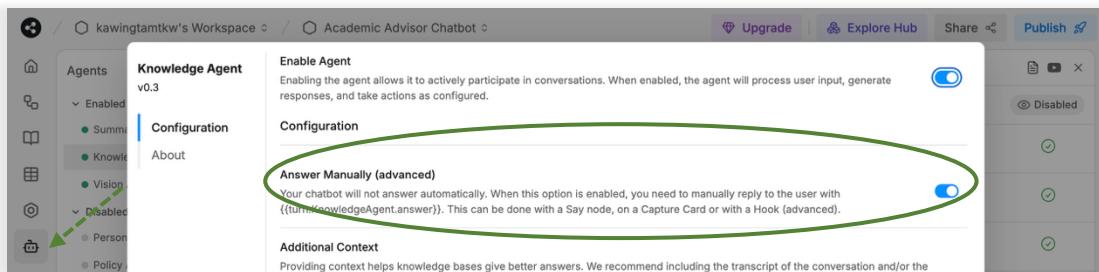
1. Go to **Knowledge Bases** in the left menu.
2. Click **New Knowledge Base**, name it **Master Programmes**.



3. Click **Add Source → Website**.
4. Paste the SPEED Master Programmes link: <https://www.speed-polyu.edu.hk/programmes/postgraduate-programmes/>
5. Click **Add Source → Document** **Document** and upload the Master Programmes PDF leaflets.

### 2-B: Set Knowledge Agent Mode

6. Go to **Agents** → **Knowledge Agent**.
7. Check the box of **Answer Manually**.



## 2-C: Build Exercise\_2 Node

8. Create a new node: **Exercise\_2**

9. Inside **Exercise\_2**:

9.1. Add a **Raw Input** card:

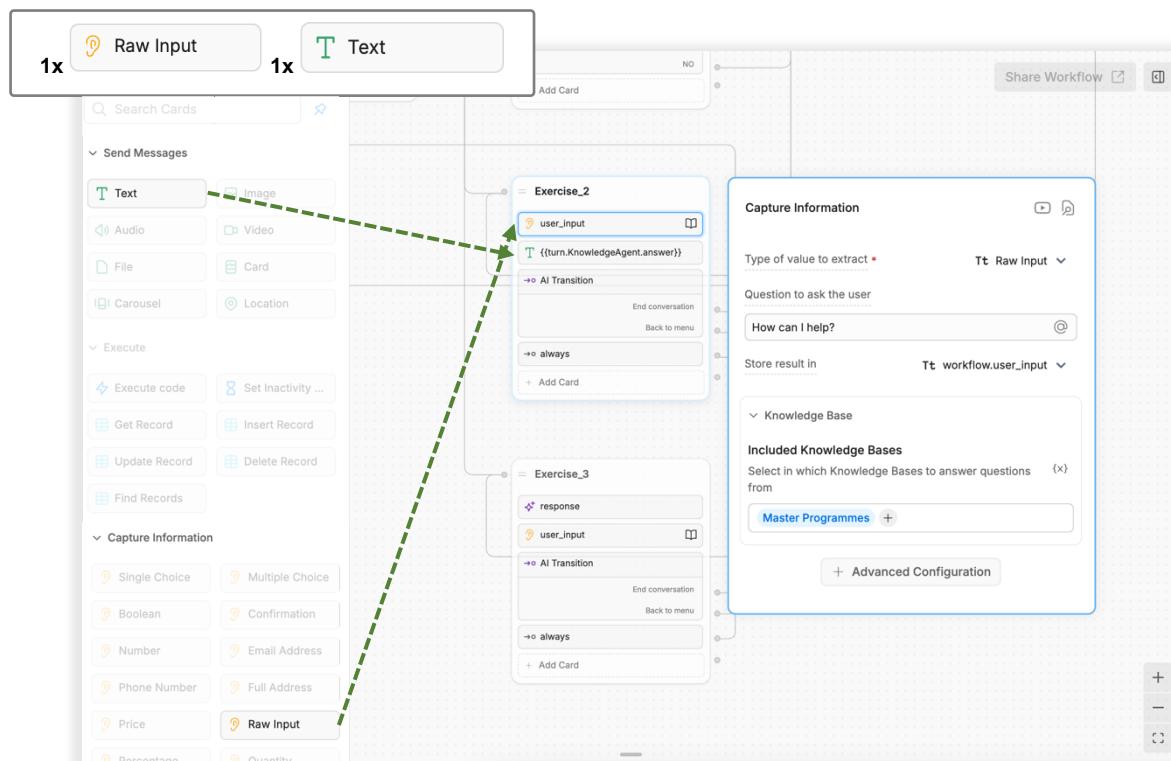
Question to ask the user: How can I help?

Store result in: **user\_input**

**Knowledge Base > Included Knowledge Bases: Master Programmes**

9.2. Add a **Text** card:

Message to send: `{{turn.KnowledgeAgent.answer}}`



## Guidelines on Botpress Chatbot Design: Part 3

### 9.3. Add an **AI Transition** card:

Text to categorize: **@workflow.user\_input**

Categories:

**End conversation**

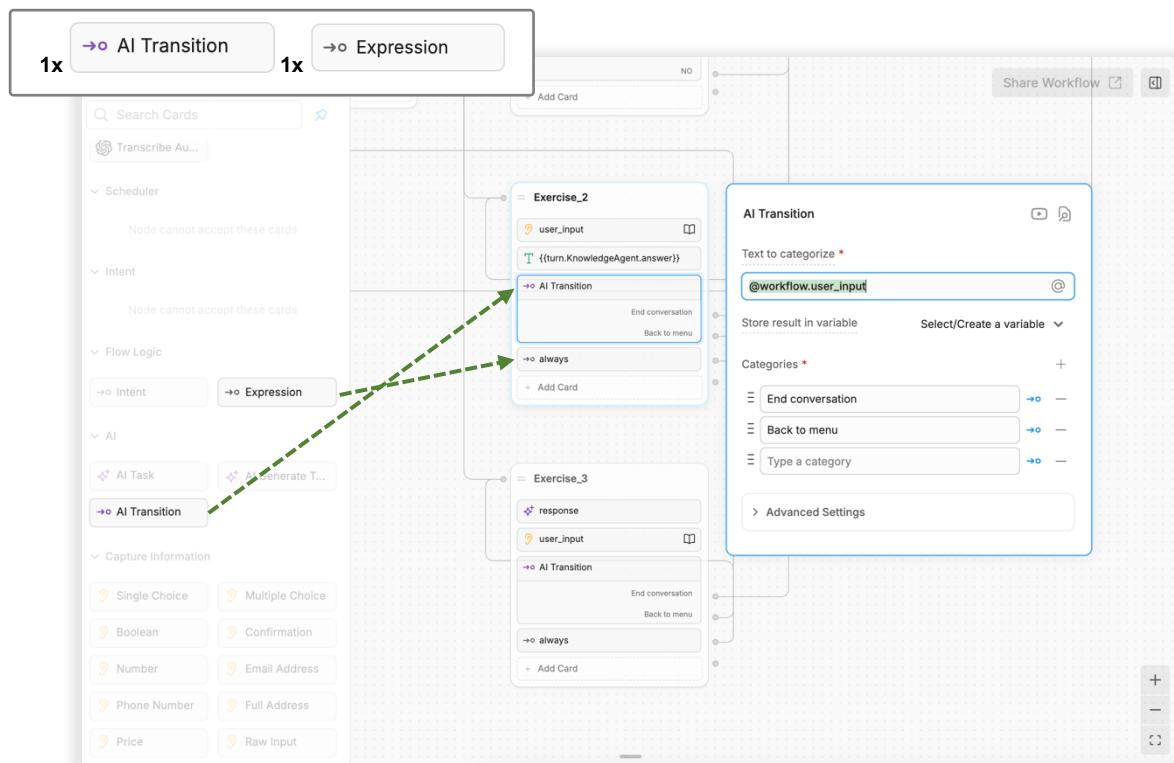
**Back to menu**

9.4. Connect the end of **End conversation** to the start of node **Bye\_Message**.

9.5. Connect the end of **Back to menu** to the start of node **Menu**.

### 9.6. Add an **Expression** card:

Label: always (default)



10. Connect the end of **always** to the start of the node **Exercise\_2** itself.

11. Go back to the **Menu** node, and connect the end of **Master Programme** to the start of node **Exercise\_2**.

## End of Exercise 2

Well done! Your bot can now answer Master Programmes queries from curated sources.

Next, let's add the Academic Advisor Helper with conversational memory. 🧠



# Exercise 3 – Academic Advisor Helper

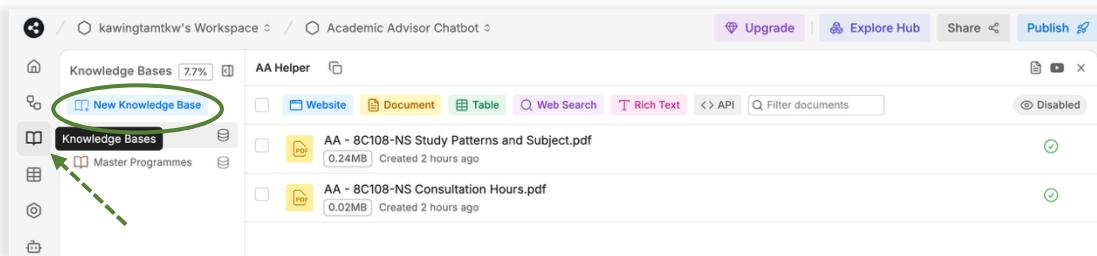
## (Advanced Techniques)

*In this section, you'll create a second knowledge base for AA Helper and enable conversational memory for context-aware responses.*

- **What you'll learn:** How to use AI Task, and create a chatbot with conversational memory like ChatGPT.
- **What the bot can do:** Answer advising-related questions using the AA Helper KB, remember context for follow-up questions, and intelligently navigate based on user intent.

### 3-A: Create AA Helper Knowledge Base

1. Go to  **Knowledge Bases** in the left menu.
2. Click  **New Knowledge Base**, name it **AA Helper**.



3. Click **Add Source** →  **Document** and upload the two PDFs starting with **AA\_**.

## 3-B: Build Exercise\_3 Node

1. Create a new node: **Exercise\_3**
2. Inside **Exercise\_3**:

- 2.1. Add a **AI Task** card:

### Task Instructions:

You are a helpful assistant. I will give you a transcript between assistant and user.

response1  
Tt workflow.response1

You must come up with a new "assistant" response to the last user response.

- The message must be friendly as possible.
- Ask questions.
- Respond in 50 words or less.

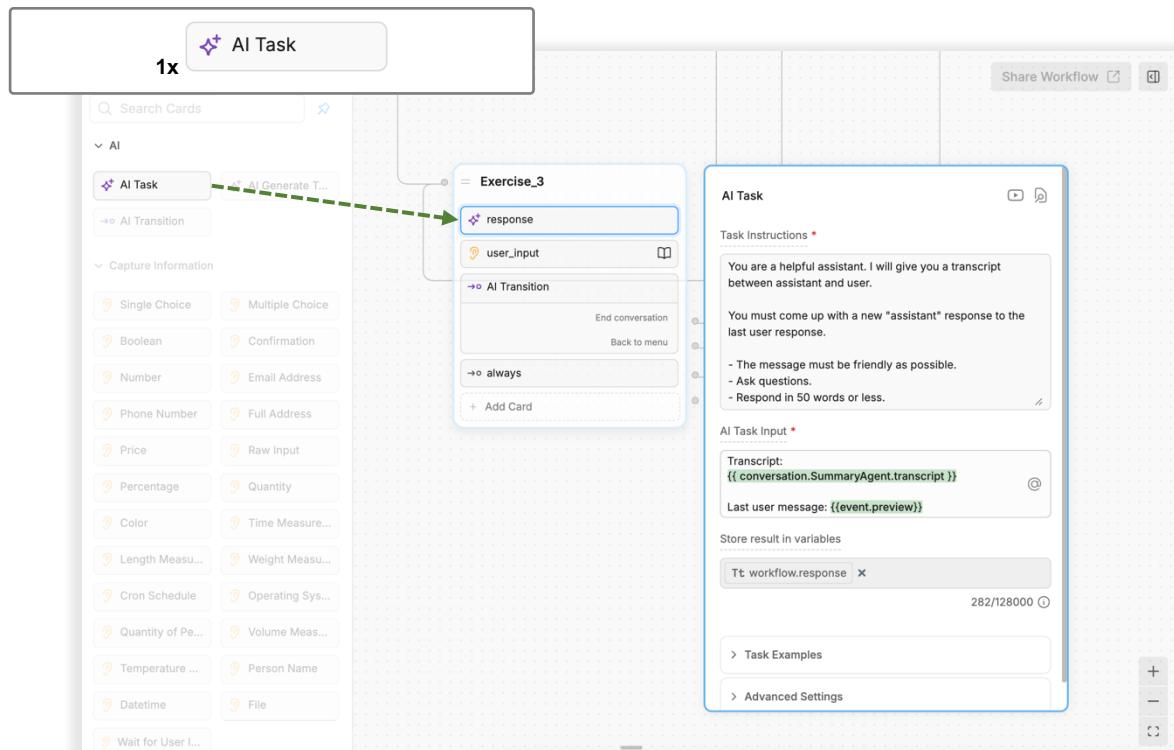
### AI Task Input:

#### Transcript:

`{{ conversation.SummaryAgent.transcript }}`

Last user message: `{{event.preview}}`

Store result in: `response`



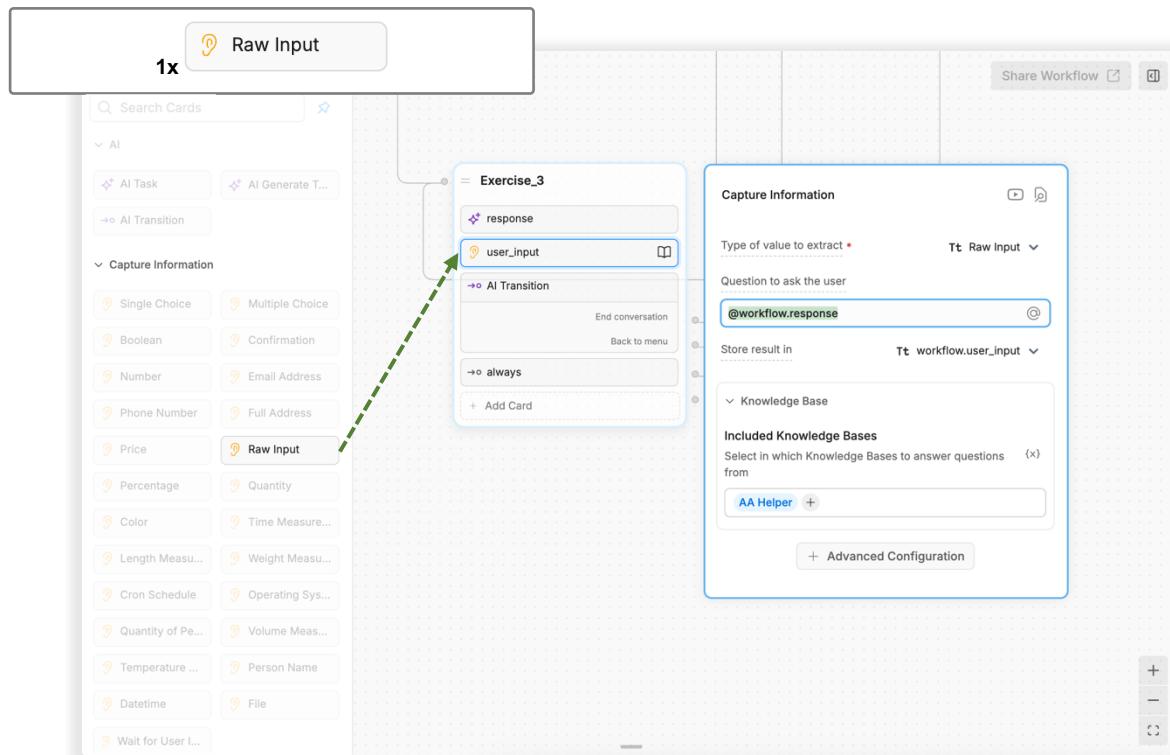
## Guidelines on Botpress Chatbot Design: Part 3

### 2.2. Add a Raw Input card:

Question to ask the user: `@workflow.response`

Store result in: `user_input`

Knowledge Base > Included Knowledge Bases: AA Helper



## Guidelines on Botpress Chatbot Design: Part 3

### 2.3. Add a **AI Transition** card:

Question to ask the user: @workflow.response

Categories:

**End conversation** (→ to node Bye\_Message)

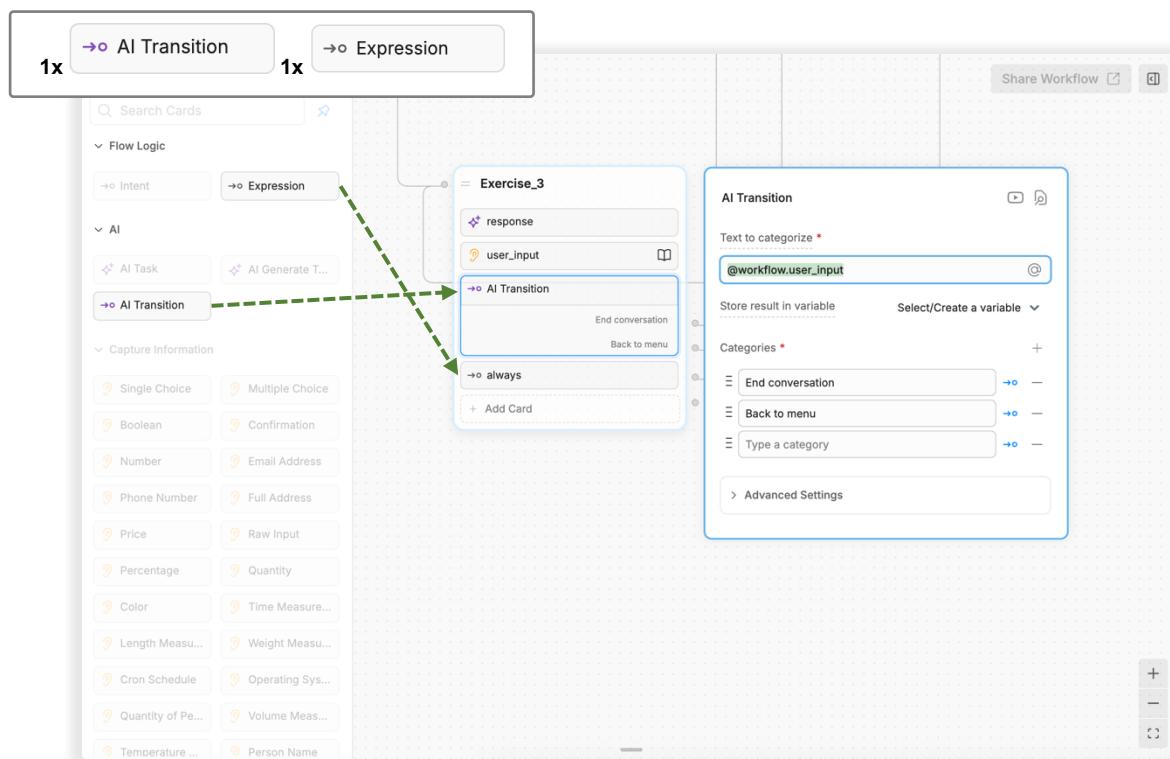
**Back to menu** (→ to node Menu)

2.4. Connect the end of **End conversation** to the start of node Bye\_Message.

2.5. Connect the end of **Back to menu** to the start of node Menu.

### 2.6. Add an **Expression** card:

Label: always (default)



3. Connect the end of **always** to the start of the node **Exercise\_3** itself.

4. Go back to the **Menu** node, and connect the end of **Academic Advisor (AA) Helper** to the start of node **Exercise\_3**.

### End of Exercise 3

✨ Excellent work! You've now built a multi-knowledge base chatbot with conversational memory.  
Test it in the Emulator to see your bot in action! 🚀





## Test & Deploy

In this section, you'll test the bot's entire flow using the Botpress Emulator to ensure it works as intended.

- **What you'll learn:** How to simulate conversations, validate logic, and identify any issues before deployment.
- **What the bot can do:** Provide a smooth and accurate experience for users, from start to finish.

### Run the Emulator

1. Open Botpress and click on the **Test Emulator** button.
2. Start the conversation in the emulator and observe how the flow works from start to finish, checking for any errors or unexpected behaviour.

### Validate Functionality

1. Try different inputs to confirm that each knowledge base responds correctly.
2. Check that conversational memory works in the AA Helper.

### Check Navigation

1. Ensure that "Back to menu" and "End conversation" options work smoothly.

### Confirm Completion

1. Verify that the bot ends the conversation cleanly at the End Node.

#### End of Section

Great work! Your bot is now fully functional and ready. 🎉



**Improving the Student Learning Experience by Helping Teachers Develop and Utilise Chatbots**  
**Project No: QESS/04/2023**

**Documentation by Ka Wing TAM and Team QESS Chatbot**

**HOTLINE** (852) 3746-0065

**EMAIL** qesschatbots@cpce-polyu.edu.hk



**Facebook** – QESS Chatbots  
[https://www.facebook.com/  
profile.php?id=61558478435727](https://www.facebook.com/profile.php?id=61558478435727)



**Instagram**  
@qesschatbots

*Any opinions, findings, conclusions or recommendations expressed in this material (or by members of the project team) do not reflect the views of the Government of the Hong Kong Special Administrative Region, the Education Bureau, any member in the Committee on Self-financing Post-secondary Education (CSPE) and its Sub-committee on Support Measures, and the Secretariat of the CSPE and its Sub-committee on Support Measures.*