Unit 3 - Build your scenarios using AI

UNIT 3 - BUILD YOUR SCENARIOS USING AI

- 3.1 Unit introduction
- 3.2 Use AI in your Make scenarios

USE CASE 1

- 3.3 Use case 1: Customer service scenario
- 3.4 Use case 1: Make AI apps
- 3.5 Use case 1: Eden AI

USE CASE 2

- 3.6 Use case 2: Extract data from pictures
- 3.7 Use case 2: Make AI apps
- 3.8 Use case 2: Eden AI

USE CASE 3

END OF UNIT

Unit 3
Build your scenarios using AI

3.1 Unit Introduction

Welcome to the final unit of the Using AI in Make course! Practice: add AI to your scenarios!

In the previous unit you have learned **what you can do with AI in Make**.

Now it's time to **use AI in your scenarios**.

You will:

- build scenarios that include AI modules
- use Make AI Toolkit and Make Content Extractor

use third-party applications to give your scenario AI capabilities

Let's begin!

Continue to 3.2: Use AI in your Make scenarios

To include AI in your scenarios, you can either use Make AI apps or any other AI apps available in Make.

Now it's time to get practical!

Continue to 3.2.1: Use case menu

3.2.1 Use case menu

You have the option to build three different use cases in Make that use AI tools.

For each one, you can:

- Use the **Make AI apps**
- Use **Eden AI**, a platform that groups services from external AI providers

Note that you can build any of the use cases, regardless of your

plan, and at no cost. The Make Academy has you covered.

Here are the three use cases you can build:

Use case 1 👩‍💻

Customer service scenario

Use GenAI to reply to customer messages on Telegram, analyze sentiment, and receive notifications of negative feedback.

Theory topics included

- Sentiment analysis
- Telegram bots

Use case 2 🖼️

Extract data from pictures

Extract data from an image of a receipt, then store the data in a Google Sheet.

Theory topics included

- Unstructured vs structured data

Use case 3 📝

Social media post creation

Starting from a prompt, generate a post that will be saved in a Google Doc, ready to be shared on social media.

Theory topics included

- Google Docs Template

Click each one to get more info on the use case and the instructions to build it.

You can build just one, two or all of them!

Use case 1

Customer service scenario

3.3 USE CASE 1

Use case 2

Extract data from pictures

3.6 USE CASE 2

Use case 3

Social media post creation

3.9 USE CASE 3

Have you built all the use cases you wanted? You still have time to go back and create a quick scenario to see the power of AI combined with Make.

If you are satisfied with what you've done, click the button to continue.

Go to the last page of the course

TAKE ME THERE!

3.3 Use case 1: Customer service scenario

Remember Alicia the Astronomer from unit 1? She buys all of her instruments from *Space Oddities Tech*.

To offer personalized customer support, Major Tom, the owner of *Space Oddities Tech*, wants to build a scenario that uses AI to help him with answering customer questions.

The scenario will analyze messages from the customers and:

- 👩‍💻 Provide a professional response to the customer in the same language they used
- 🚦 Perform sentiment analysis on the message
- ☹️ If the sentiment is negative, offer a solution in the reply
- 🔔 Send a notification to Major Tom whenever it detects a negative message

This is what you will build. If you'd like to try a different one, you can return to the Use case menu by clicking the button below, otherwise let's go build it!

Go to the Use case menu

TAKE ME BACK!

Before you start, let's have a look at some key theory topics you will need to know for this use case:

1. Sentiment analysis
2. Telegram bot

1: Sentiment analysis

1

Sentiment analysis

Sentiment analysis is a technique that identifies and interprets emotions in text.

By analyzing words, phrases, and context, it determines whether the sentiment is positive, neutral, or negative.

This process **relies on AI** to understand not only the explicit meaning of the text but also the tone, intent, and subtle nuances that convey emotions.

It can be applied to various forms of text, such as customer reviews, social media posts, emails, or survey responses helping companies understand how people feel about their products, services, or brand. For example, it can analyze customer reviews to measure satisfaction or detect trends in social media discussions.

It can also help businesses make better decisions by analyzing customer opinions, improving products and services, managing brand reputation, and monitoring employee satisfaction.

Positive ★★★★★

This telescope is amazing! I was even able to see the moon with it!!

Neutral ★★★☆☆

This telescope is fine. It does its job like all the other telescopes I have.

Negative ★☆☆☆☆

This telescope is awful! I wasn't able to observe dark matter!

LLMs perform sentiment analysis by examining text patterns, word associations, and context to identify the emotional tone or attitude in the content.

Here's how it works:

Tokenization

The LLM breaks the input text into smaller pieces (tokens) like words, phrases, or subwords.

Context understanding

LLMs analyze token relationships and sentence structure to detect nuances like sarcasm or subtle emotions.

Probability distribution

The model gives probabilities to different sentiments (e.g., positive, neutral, negative) based on patterns it learned from training on labeled data.

Classification

The LLM return the sentiment with the highest probability as the output.

You will use AI to perform sentiment analysis in this scenario.

2: Telegram bot

2

Telegram bot

Telegram is an online messaging app, but unlike standard messaging apps, it allows users to create chatbots.

Telegram chatbots are programs that run within the Telegram app automating tasks, providing information and interacting with users.

For example, they can automatically respond to customer inquiries. This is what you will do in the use case.

You can create bots using Telegram's bot builder, called **BotFather** that will make you an offer you can't refuse.

In the use case, you'll see that to create a bot, you need to start a chat with BotFather and use specific commands to set up your bot.

Normally, setting up a bot requires programming knowledge to tell the bot what to do. **With Make, you can easily set up your bot without needing to code.** You'll practice this in the use case.

Once your bot is ready, you can **interact with it by chatting**, sending messages, and receiving responses.

Continue to 3.3.1: Choose how to build it

3.3.1 Choose how to build it

Now you know everything you need to build this use case. All that's left is to choose which apps you want to use. Pick your favorite, you can always come back later and try the other option too.

Click the button to see the building instructions.

Make AI apps

BUILD IT!

Eden AI

BUILD IT!

If you don't feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

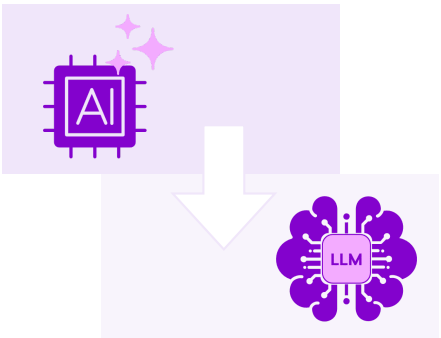
Go to the Use case menu

TAKE ME BACK!

3.4 Use case 1: Make AI apps

You have decided to build use case 1 with Make AI apps. Amazing choice!

For this use case, you will use the Make AI Toolkit.



AI provider

As you've seen in the previous section, to use the modules of this app you need to connect it to an AI provider. In the use case you will use the **Make's AI provider**.

If you're on Pro plan and above, you can also choose other AI providers if you have a connection available and want to test it. Just note that the results might be different from the ones shown here.

Model

For the LLM model, follow the best practice from the previous units: start with the smaller model and switch to a larger one if the results aren't good enough. For this use case, the Make team tested the Small model, and it works well.

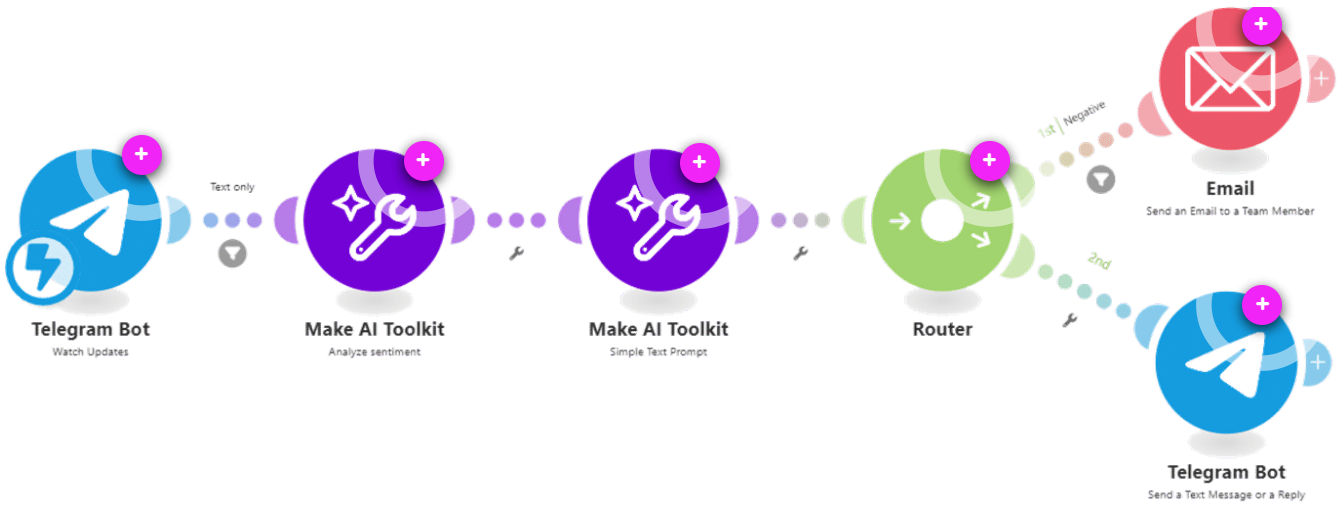
Continue to 3.4.1: Scenario overview

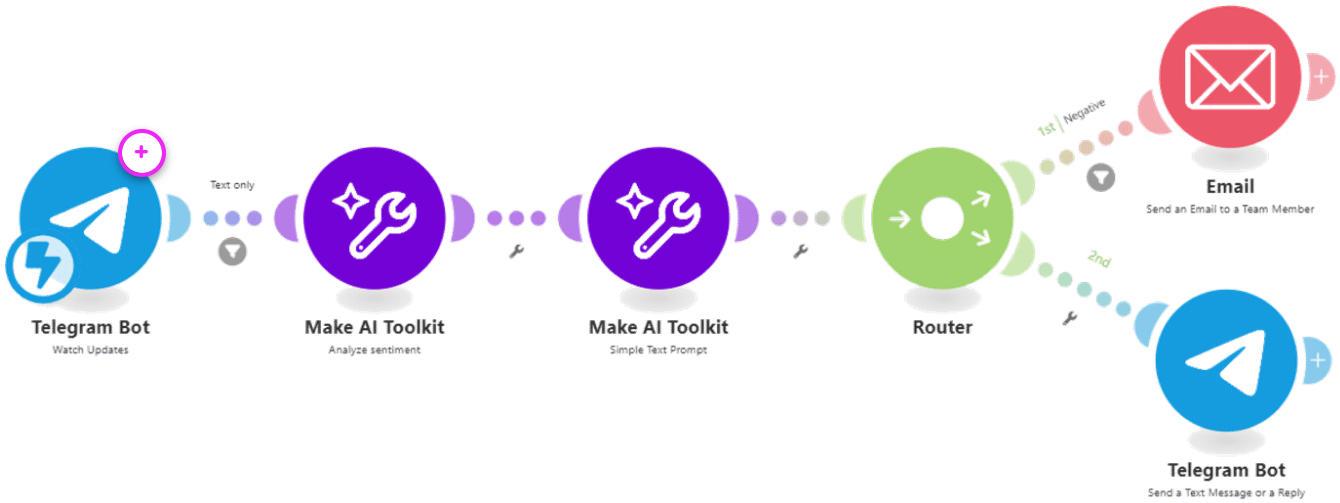
3.4.1 Scenario overview

Let's have a closer look at the scenario you will build.

Here's an overview of the scenario.

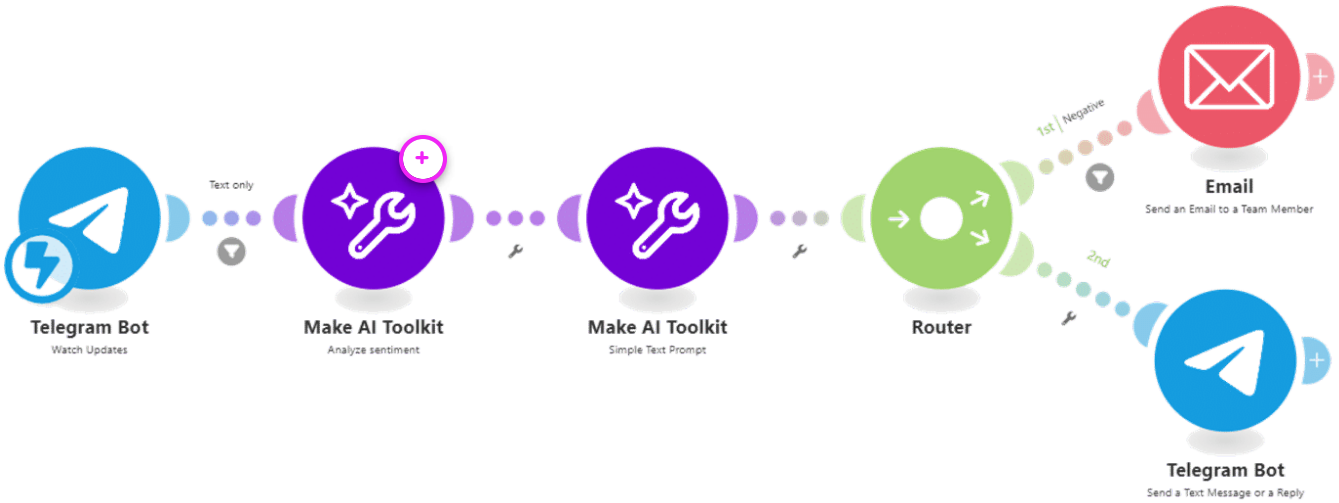
Click each + to learn more.


Telegram Bot
Watch Updates
Text only
Make AI Toolkit
Analyze sentiment
Make AI Toolkit
Simple Text Prompt
Router
1st | Negative
Email
Send an Email to a Team Member
2nd
Telegram Bot
Send a Text Message or a Reply



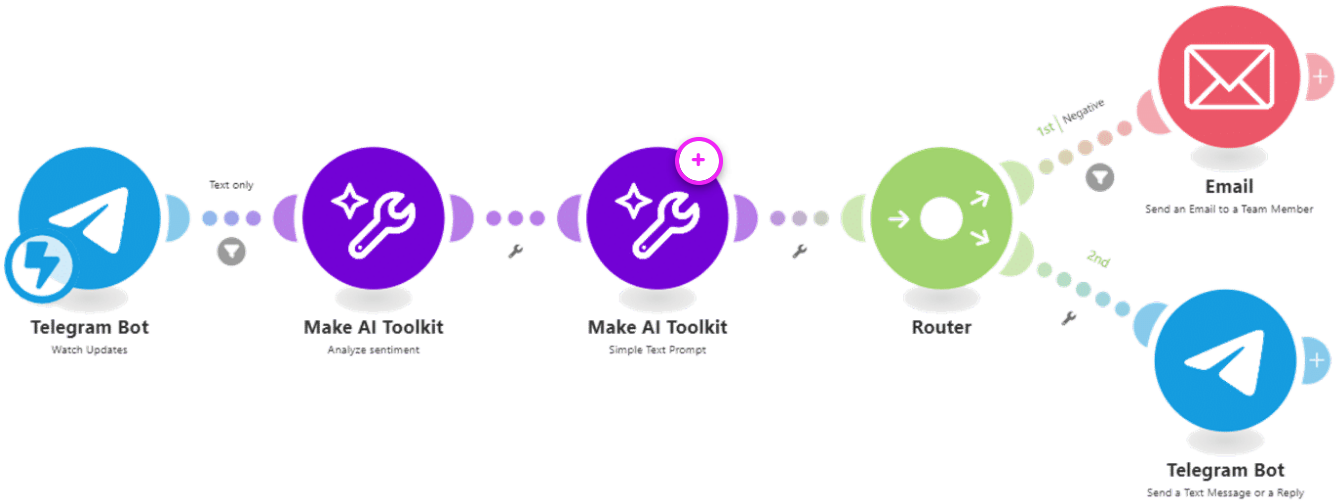
Telegram Bot>Watch Updates

Watch for new messages from customers in the Telegram chat with a customer service chatbot that you create.
When your Telegram bot receives a message, it triggers the scenario.



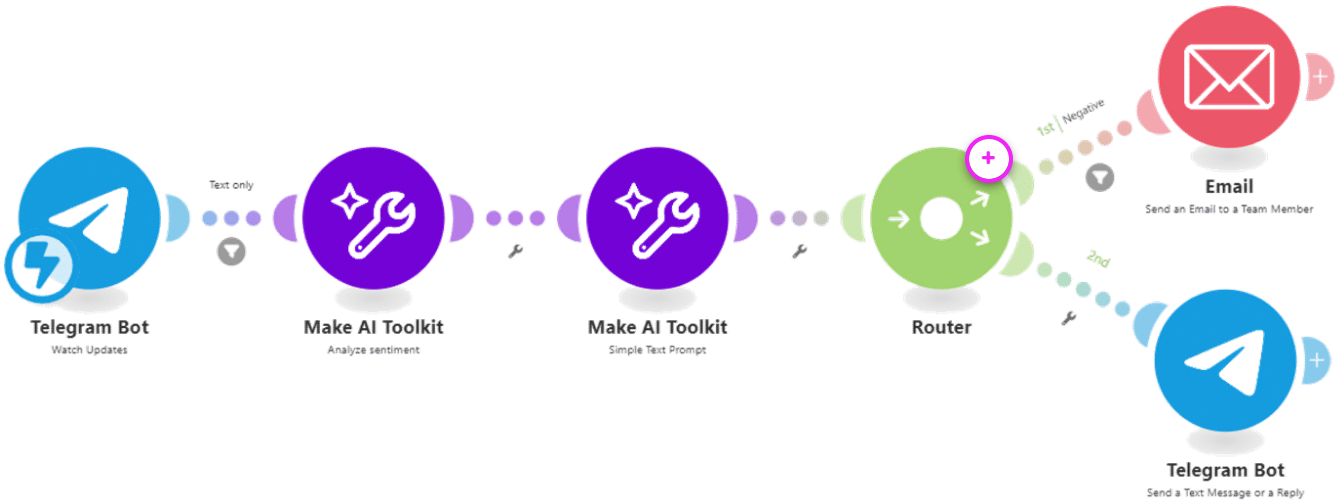
Make AI Toolkit> Analyze sentiment

Use AI to perform sentiment analysis on the text message received.
A filter is placed before it to make sure only text messages, and not images or voice notes, are passed to this module.



Make AI Toolkit> Ask anything

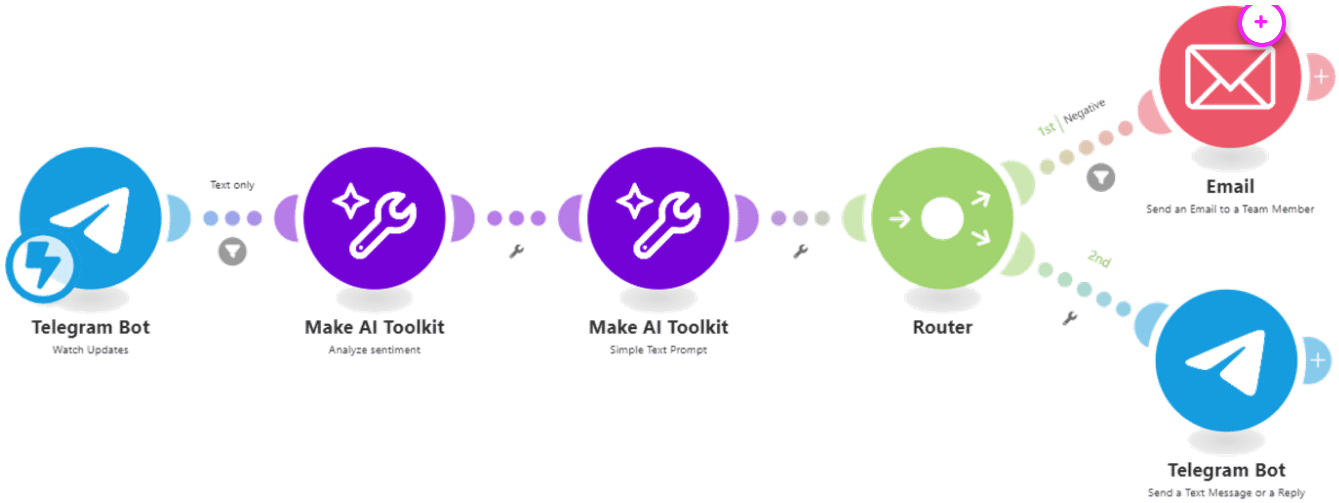
Use AI to generate text starting from a prompt. In this case the prompt contains the text message received. The prompt asks to generate a reply for the customer.



Flow control> Router

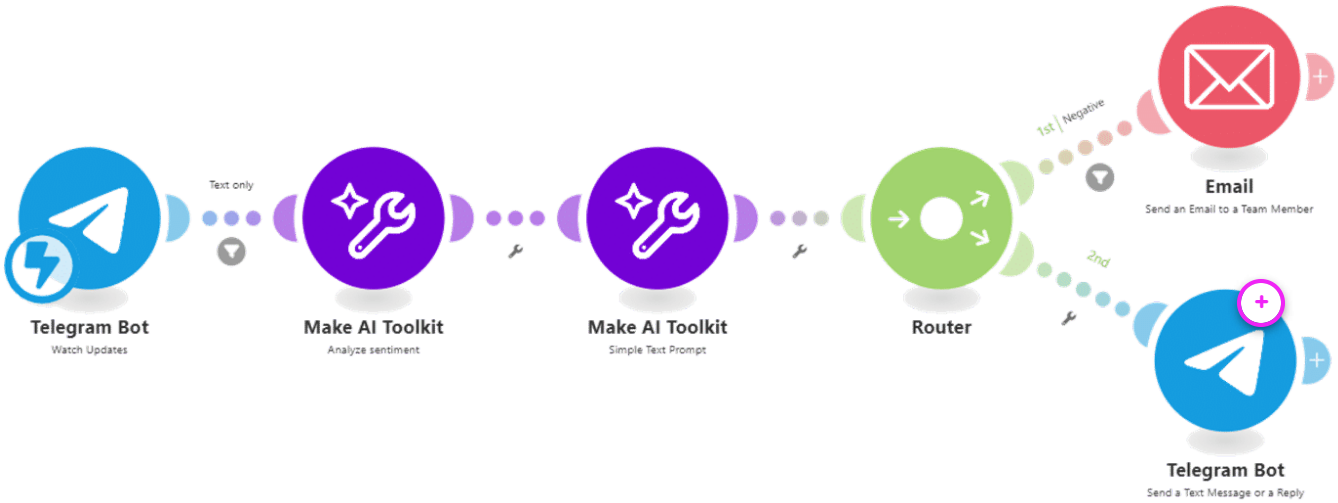
Generate two routes to handle negative messages.

- **1st | Negative**: messages with negative sentiment will pass through this route, to make sure you are notified about them.
- **2nd**: all the messages go through this route to send a response to the user. Note that it is not a fallback route and there is no filter, this means **also** negative messages will be processed by it.



Email> Send an Email to a Team Member

Send yourself an email to get notified about negative comments.
This module doesn't require any connection set up. You can easily select the email of the people in your Make Team, yourself in this case.



Telegram Bot > Send a Text Message or a Reply

Send a Telegram message. In this case, the message generated by AI in the Make AI Toolkit module is sent back to the user in the chat they initiated with the customer service chatbot.

If you don't feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

Continue to 3.4.2: Use case preparation

3.4.2 Use case preparation

Let's create the bot.

Work through each stage before you continue.

Step 1

Get Telegram

To create a Telegram bot, the first thing you need is, surprise surprise, Telegram!

If you haven't already, download and **install Telegram** on your mobile device. Once installed, you can either use it on your phone or access the **web version**.

Note that to be able to access the web version you need to have installed the Telegram app on your phone first.

Step 2

Create the bot

In Telegram, **search for BotFather** to start creating your bot. Select the one with the **blue verification badge** to make sure it is the official one.

In the BotFather chat, if it's the first time you've used BotFather, click **Start**.

Then type **/newbot** and press **Enter** to start creating your bot. The BotFather replies with detailed instructions on how to proceed.

Next, you need to name your bot, like **Make_AI** and press **Enter**.

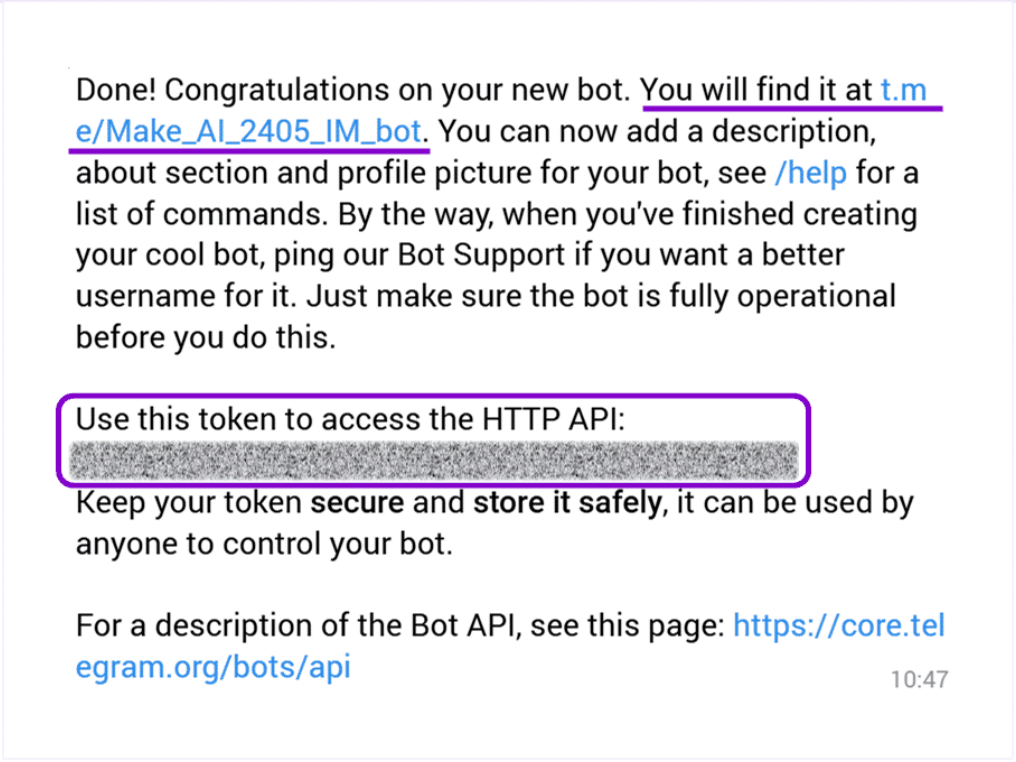
Then, you have to give your bot a unique username that ends with **bot**. For example, you can add to your bot's name today's date and your initials to create something like: **Make_AI_2401_IM_bot**.

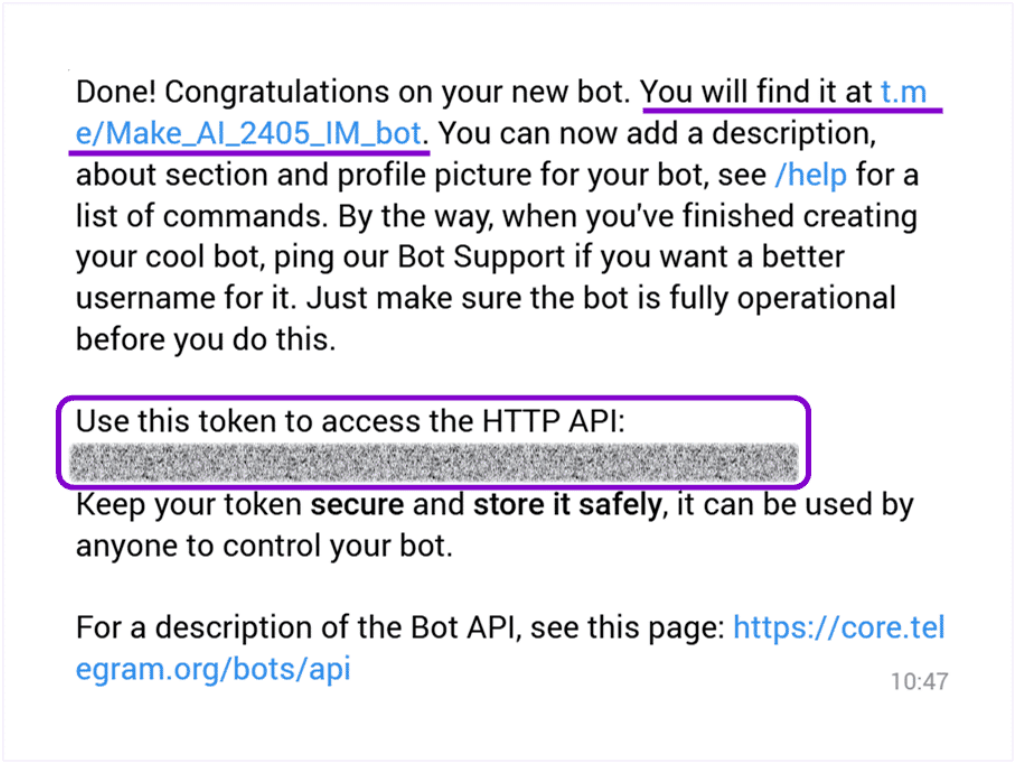
Note: If the BotFather tells you that the user name has already been taken, you can try to add a different number or variation until you find an available

one.

Once you have successfully created your bot, the BotFather will display some useful information about it.

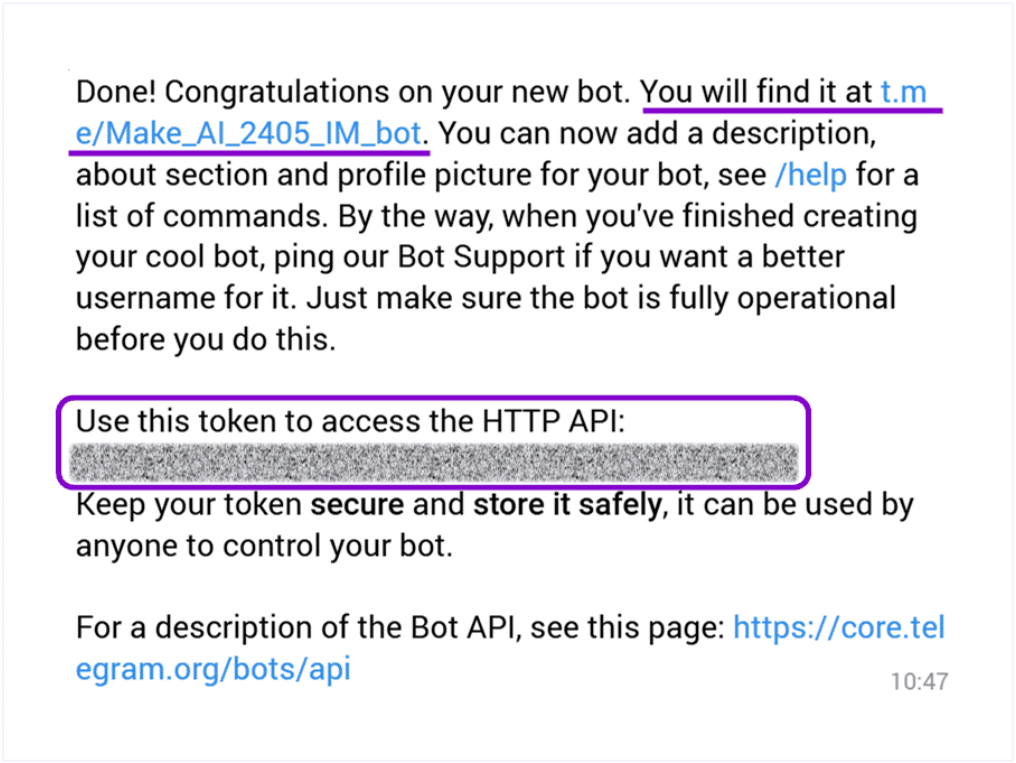
Click each + to learn more.





Link

Link where you can access the chat with your bot. If you click on it, you will be redirected to the chat with the bot that you can use to interact with it.



API token

The **API token**. It is the API key, the secret password that will allow you to interact with your bot. Use your mouse to highlight it and copy it. You will need it to set up the connection to Telegram within Make.

You will need this information when you will build your scenario. Let's do it!

Continue to 3.4.3: Build it

3.4.3 Build it

When building this scenario, you will set up a webhook in the Telegram bot module so it can detect new messages.

Webhooks

Webhooks are a technology that allows applications to communicate with each other and perform actions when something happens. In this case, the webhook will receive a a notification from Telegram when someone sends a message to your chatbot. When a message arrives, it starts the scenario.

That's everything you need to know for this use case.

Webhooks are very powerful, but can be a bit complex, so if you want to learn more, check out the course What are webhooks?

Let's build the scenario.

Step 1

Telegram Bot> Watch Updates (1/7)

1 Create the scenario

In Make, create a new scenario and call it **AI_C03_U03_UC1_MakeAI**.

2 Add the module

Add a **Telegram Bot> Watch Updates** module.

3 Create the webhook

The first step is to create a webhook. Click **Create a webhook**. Name your webhook, for example using the same name of you bot (here **Make_AI_2405_IM_bot**).

4 Create the connection to Telegram

Click **Create a connection**. Paste the **API token** that you have obtained from the **BotFather** to connect to your bot. Click **Save** to save the connection.

5 Save

Click **Save** again to save your webhook and **Save** to save your module.

Step 2

Make AI Toolkit> Analyze sentiment (2/7)

1 Add the module

To perform sentiment analysis on the message received add a **Make AI Toolkit> Analyze sentiment** module.

2 Create the connection

If you're on a Pro plan or above, click **Create a connection** to connect it to an AI provider. Choose **Make's AI provider**.

If you're on Free or Core plan, **Make's AI provider** is already selected.

Name the connection **Make_AI_Toolkit**, and click **Save**.

3 Set up the module

For the **Model**, select **Small** from the dropdown menu.

In the **Text to analyze** field map the `Message: Text` from the **Telegram** module.

4 Save

Click **Save** to save the module.

Step 3

Filter (3/7)

To make sure that only text messages are passed through, **add a filter** between these first two modules.

1 Add the filter

Click on the link between them and select **Set up a filter** to open the filter setup. Label it **Text only**.

2 First condition

In the Condition, map the Message: Text item from the **Telegram** module and select **Basic operators: Exists**.

This ensures that only text messages are passed. For example, if a photo is sent, the Message: Text item won't be present, and the message won't

be passed to the **Make AI Toolkit** module.

3 Second condition

Click **Add AND rule** to add another condition. In the top field, map the `Message: Text` item again. Then select **Text operators: Not equal to** and type **/start** in the lower field. Press **Escape** to close the suggestion window.

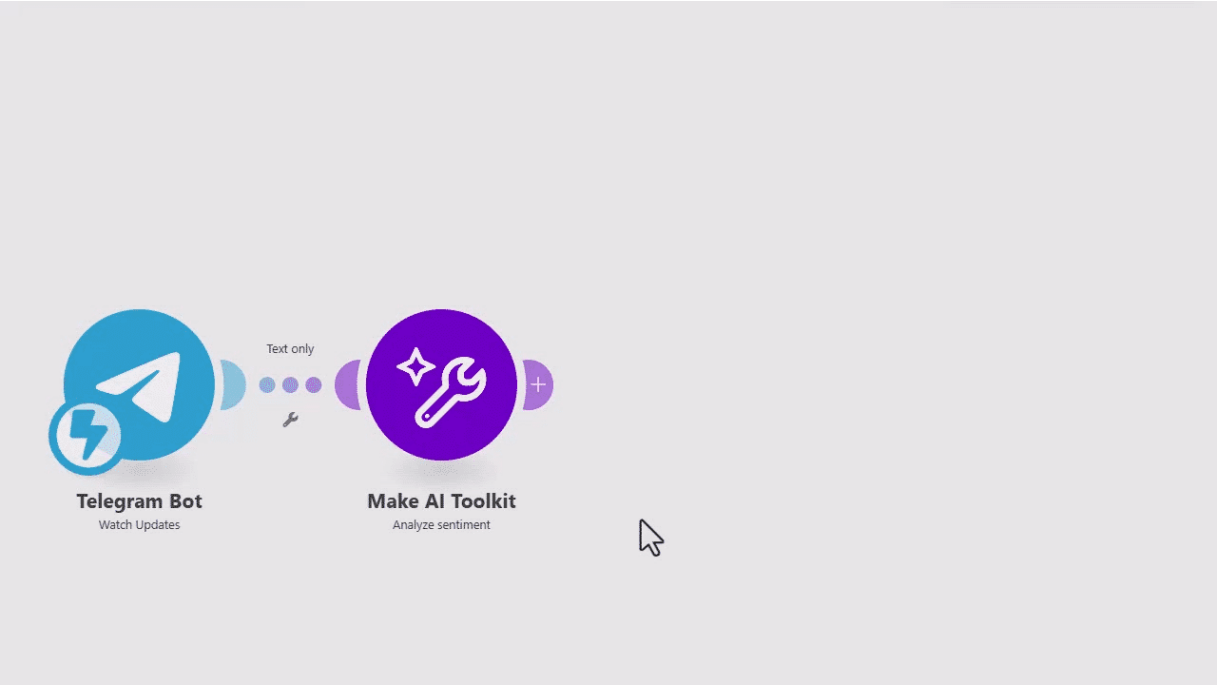
The **/start** message is sent to the chat when you first interact with your bot, and you don't want to process it. This condition ensures that the **/start** message is filtered out.

4 Save

Click **Save** to save the filter.

Step 4

Make AI Toolkit> Ask anything (4/7)



1 Add the module

Add an **Make AI Toolkit> Simple Text Prompt** module. This will generate the text that you will use to reply to Telegram messages.

For the **Connection**, make sure that the connection you created for the **Analyze sentiment** module is selected. Nothing to set up here.

2 Set up

For the **Model**, select **Small** from the dropdown menu.

In the **Text** field paste the following prompt that contains the instructions to generate a reply to the customer message:

> You received the following message from a customer: [MAP MESSAGE]. Reply to the message: - Provide a simple

response of around 20 words. - Keep a professional tone. - If the message is negative, offer a solution. - Answer in the same language of the customer.

Remove the **[MAP MESSAGE]** instruction (including the parentheses) and map the **Message: Text** from the **Telegram** module.

3 Save

Click **Save** to save the module.

Step 5

Router and Filter (5/7)

You want to get notified about negative messages, so let's create a route for this.

1 Add the module

Add a **Flow Control> Router** module.

2 Add the filter

Then click the **Filter** of the 1st route. Label it **Negative** and, in the **Condition** field, map the Sentiment item from the **Analyze sentiment** module.

Select **Text operators: Equal to (case insensitive)** and type **negative** in the lower field. Choose case insensitive to be on the safe side, in case AI returns the sentiment with different capitalization.

3 Save

Click **Save** to save the filter.

This ensures that only messages with negative sentiment will pass through this route.

Step 6

Email> Send an Email to a Team Member (6/7)

1 Add the module

In the 1st route, add an **Email> Send an Email to a Team Member** module to receive notifications about the negative messages.

2 Set up

From the dropdown menu, select your email address and enter **Negative message from a customer** in the **Subject** field.

Then, highlight, copy, and paste the following message into the **Content** field.

```
There was a negative comment from <b> [MAP FIRST and LAST NAME]</b><br><br><b>Message:</b> <i>[MAP MESSAGE]</i><br><br> <b>Response sent:</b></br> <i> [MAP REPLY]</i>
```

Time to do some mapping.

Remove the **[MAP FIRST and LAST NAME]** instruction (including the parentheses) and map the `Message: From: First Name` and `Message: From: Last Name` items from the **Telegram** module.

Repeat the process for **[MAP MESSAGE]**, mapping the `Message: Text` item from the **Telegram** module.

Do the same for **[MAP REPLY]**, mapping the `Answer` item from the **Ask anything** module.

3 Save

Click **Save** to save the module.

Note that the email message contains HTML tags for text formatting:

- **
** adds a line break in the text
- **** makes the enclosed text bold
- **** makes the enclosed text italic

Step 7

Telegram Bot> Send a Text Message or a Reply (7/7)

The final step is to send a reply to the user in the bot chat. All the messages, including the negative ones, will pass through the 2nd route, so add the module to send a reply in this route.

1 Add the module

Add a **Telegram Bot> Send a Text Message or a Reply** module.

For the **Connection**, make sure that the connection you created for the **Telegram bot** module is selected. .

2 Set up

In the **Chat ID** field, map the Message: Chat: ID item form the first **Telegram** module, which contains the chat ID with your bot. This ensures that

the replies is sent to the correct chat.

Be careful not to select **Edited Channel Post: Message ID**. If you do, your use case won't work.

In the **Text** field map the **Answer** item from the **Ask anything** module to use the reply generated by AI.

3 Save

Click **Save** to save the module and save your scenario as well.

Continue to 3.4.4: Run it

3.4.4 Run it

Time to test the use case.

Turn **ON** your scenario. In the **Schedule settings** pop up, check that **Immediately** is selected and click **Save**.

Then save your scenario as well.

Go to **Telegram** and open the chat with the bot that you created earlier.

Click **/start** to begin chatting.

Let's test three different conditions.

Click each one to learn more.

Positive message 🙂 _

In the chat with the bot, type a positive message like:

This telescope is amazing! I was even able to see the moon with it!!

Press **Enter**.

Wait for the scenario to run and see the bot's reply to your message.
Note that it might take some time for the scenario to run.
This is the text generated by your scenario!
Cool right?

Note that it might take some time for the scenario to run.
Check your inbox and you should have received the email that notifies you about this negative message.

Message in another language 💬

You can now try to type your message in another language, your mother tongue, any other language you speak, or use your friend Google Translate.

Since one of the creators of this course is Italian, you can type:

> **Che figo questo telescopio!**

Press **Enter**. Note that it might take some time for the scenario to run.

See the scenario replying to you in Italian, making you dream of pizza, spaghetti, parmesan, gelato, caffe, and all the amazing things they have over there.

Well done! You've tested your scenario. Now you know how to use AI tools in your scenario to handle customer messages.

> Your job is done here. Go back to the **Use case menu** and pick another use case.
>
> *Click the button below.*

Go to the Use case menu

TAKE ME BACK!

3.5 Use case 1: Eden AI

You have decided to build use case 1 with Eden AI. Great choice!

Now it’s time to learn what Eden AI is, how it works, and how to set up your account so you can start using it in your scenarios.

To do this, head over to the dedicated Eden AI section. You can access it from any of the three Eden AI use cases, and at the end you’ll find a button to return here and start building.

Eden AI information and set up

LEARN MORE

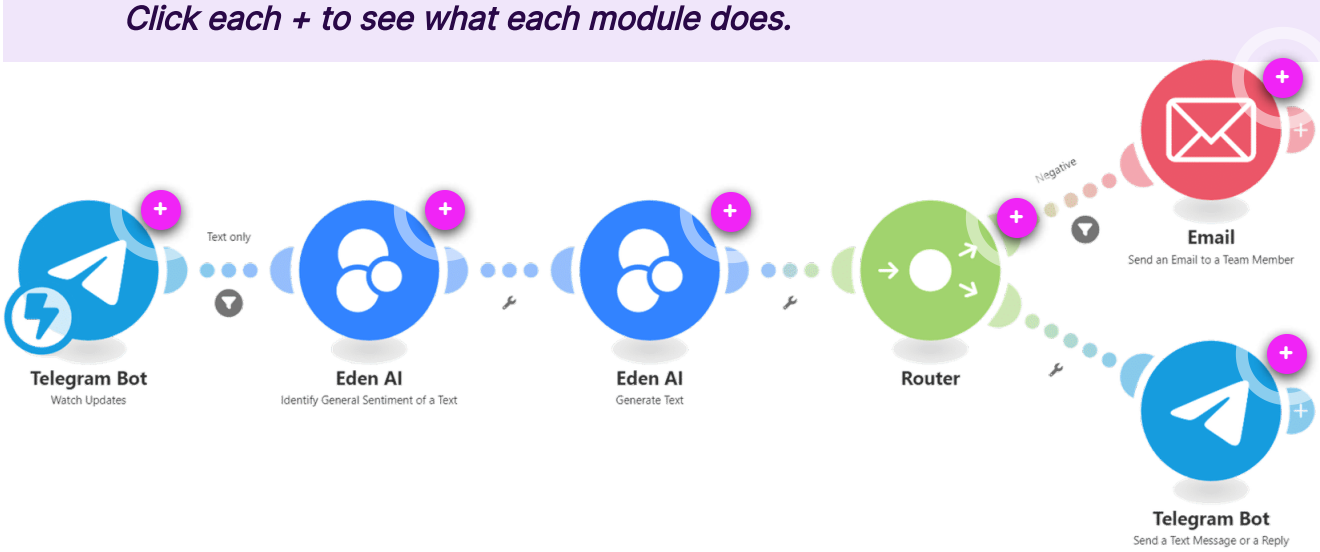
Continue to 3.5.1: Scenario overview

3.5.1 Scenario overview

Now that you have all the knowledge you need and your Eden AI account, let's have a closer look at the scenario you will build.

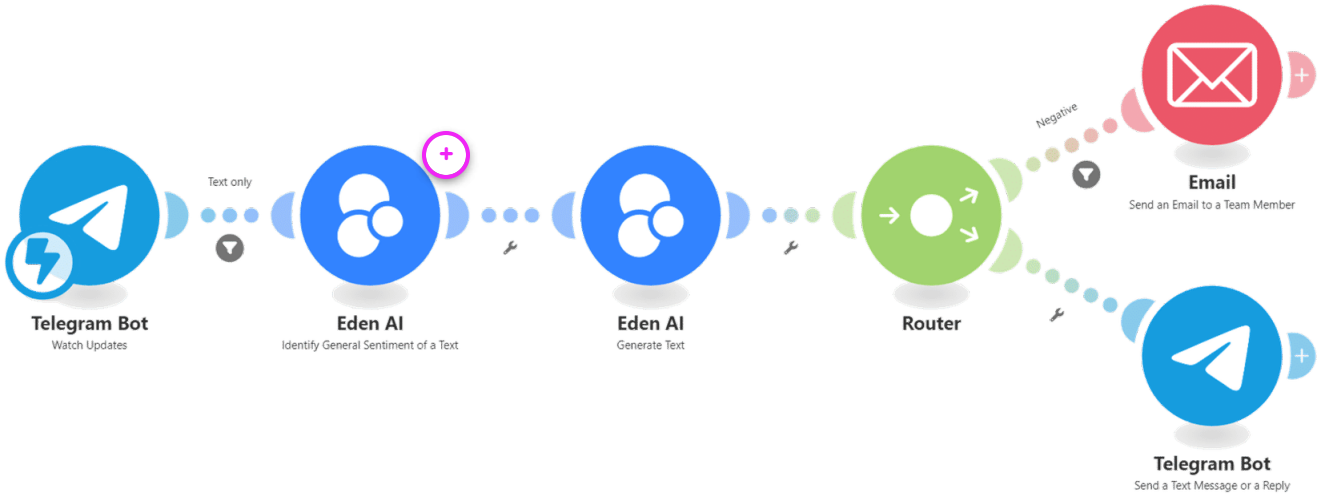
Here's an overview of the scenario.

Click each + to see what each module does.



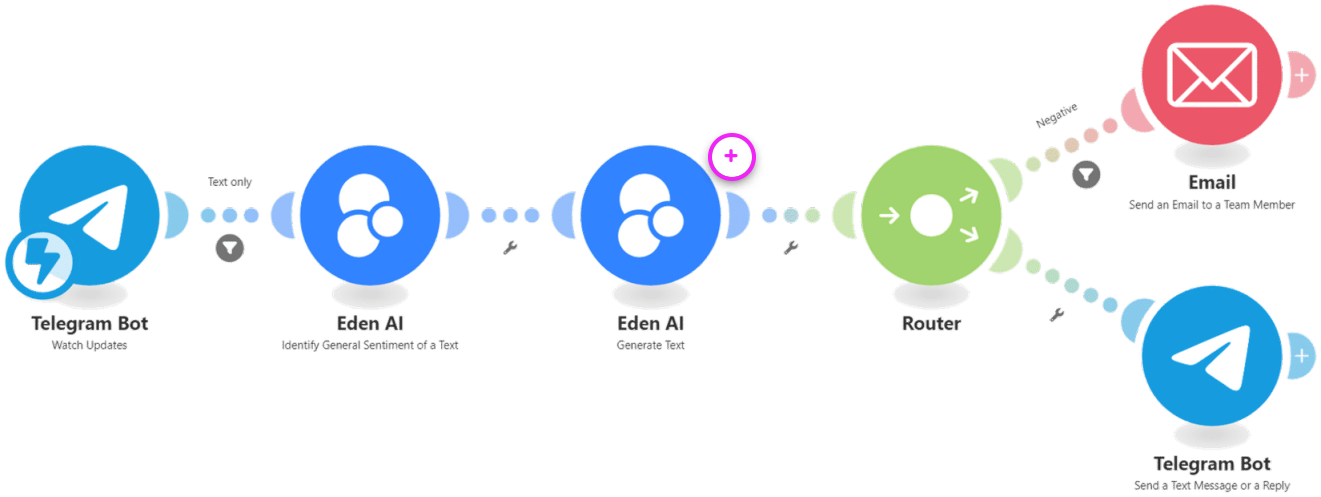
Telegram Bot>Watch Updates

Watch for new messages from customers in the Telegram chat with a customer service chatbot that you create.
When your Telegram bot receives a message, it triggers the scenario.



Eden AI> Identify General Sentiment of a Text

Use AI to perform sentiment analysis on the text message received.
A filter is placed before it to make sure only text messages, and not images or voice notes, are passed to this module.



Eden AI> Generate Text

Use AI to generate text starting from a prompt. In this case the prompt contains the text message received. The prompt asks to generate a reply for the customer.

Flow control> Router

Generate two routes to handle negative messages.

- **1st | Negative**: messages with negative sentiment will pass through this route, to make sure you are notified about them.
- **2nd**: all the messages go through this route to send a response to the user. Note that it is not a fallback route and there is no filter, this means **also** negative messages will be processed by it.

Email> Send an Email to a Team Member

Send yourself an email to get notified about negative comments.
This module doesn't require any connection set up. You can easily select the email of the people in your Make Team, yourself in this case.

Telegram Bot > Send a Text Message or a Reply

Send a Telegram message. In this case, the message generated by AI in the Eden AI module is sent back to the user in the chat they initiated with the customer service chatbot.

If you don't feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

Continue to 3.5.2: Use case preparation

3.5.2 Use case preparation

Before building the use case, you need to create the Telegram bot. Let's create the bot.

Work through each stage before you continue.

Step 1

Get Telegram

To create a Telegram bot, the first thing you need is, surprise surprise, Telegram!

If you haven't already, download and **install Telegram** on your mobile device. Once installed, you can either use it on your phone or access the **web version**.

Note that to be able to access the web version you need to have installed the Telegram app on your phone first.

Step 2

Create the bot

In Telegram, **search for BotFather** to start creating your bot. Select the one with the **blue verification badge** to make sure it is the official one.

In the BotFather chat, if it's the first time you've used BotFather, click **Start**.

Then type **/newbot** and press **Enter** to start creating your bot. The BotFather replies with detailed instructions on how to proceed.

Next, you need to name your bot, like **Make_AI** and press **Enter**.

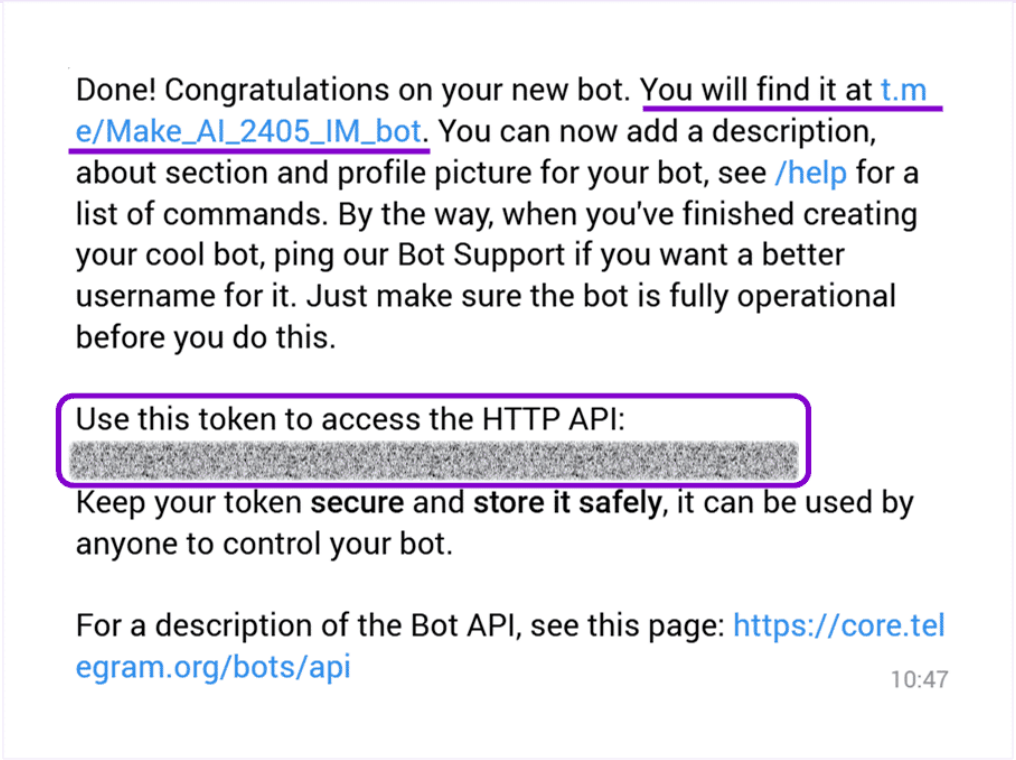
Then, you have to give your bot a unique username that ends with **bot**. For example, you can add to your bot's name today's date and your initials to create something like: **Make_AI_2401_IM_bot**.

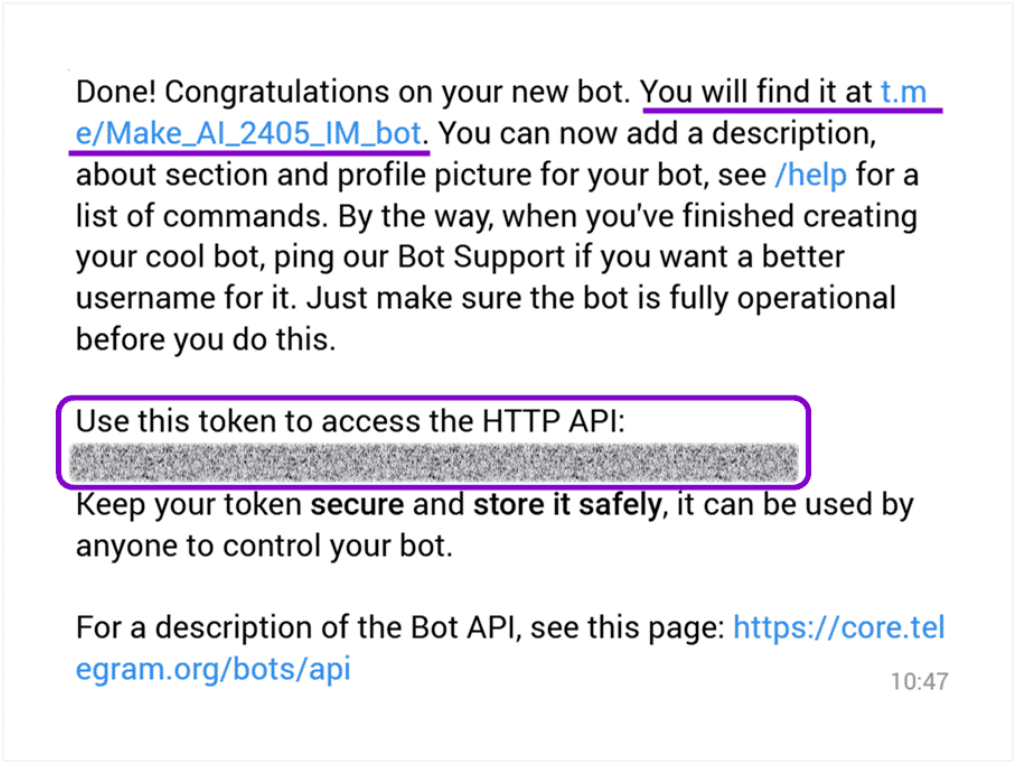
Note: If the BotFather tells you that the user name has already been taken, you can try to add a different number or variation until you find an available

one.

Once you have successfully created your bot, the BotFather will display some useful information about it.

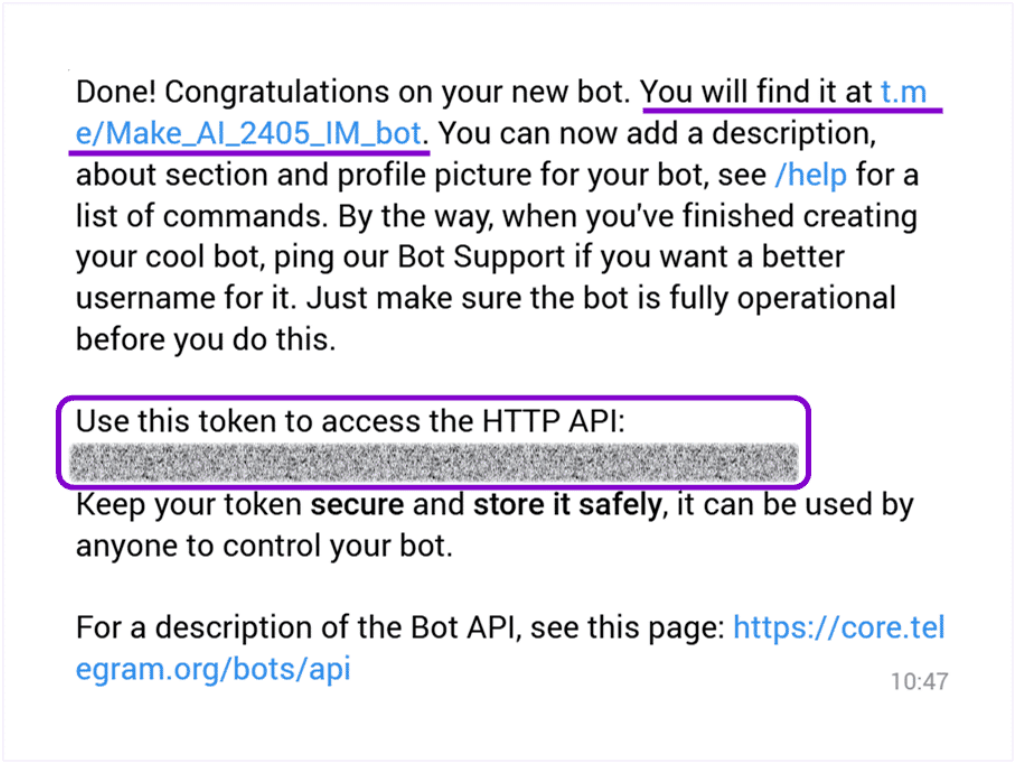
Click each + to learn more.





Link

Link where you can access the chat with your bot. If you click on it, you will be redirected to the chat with the bot that you can use to interact with it.



API token

The **API token**. It is the API key, the secret password that will allow you to interact with your bot. Use your mouse to highlight it and copy it. You will need it to set up the connection to Telegram within Make.

You will need this information when you will build your scenario. Let's do it!

Continue to 3.5.3: Build it

3.5.3 Build it

When building this scenario, you will set up a webhook in the Telegram bot module so it can detect new messages.

Webhooks

Webhooks are a technology that allows applications to communicate with each other and perform actions when something happens. In this case, the webhook will receive a a notification from Telegram when someone sends a message to your chatbot. When a message arrives, it starts the scenario.

That's everything you need to know for this use case.

Webhooks are very powerful, but can be a bit complex, so if you want to learn more, check out the course What are webhooks?

Let's build the scenario.

Step 1

Telegram Bot> Watch Updates (1/7)

1 Create the scenario

In Make, create a new scenario and call it **AI_C03_U03_UC1_MakeAI**.

2 Add the module

Add a **Telegram Bot> Watch Updates** module.

3 Create the webhook

The first step is to create a webhook. Click **Create a webhook**. Name your webhook, for example using the same name of you bot (here **Make_AI_2406_IM_bot**).

4 Create the connection to Telegram

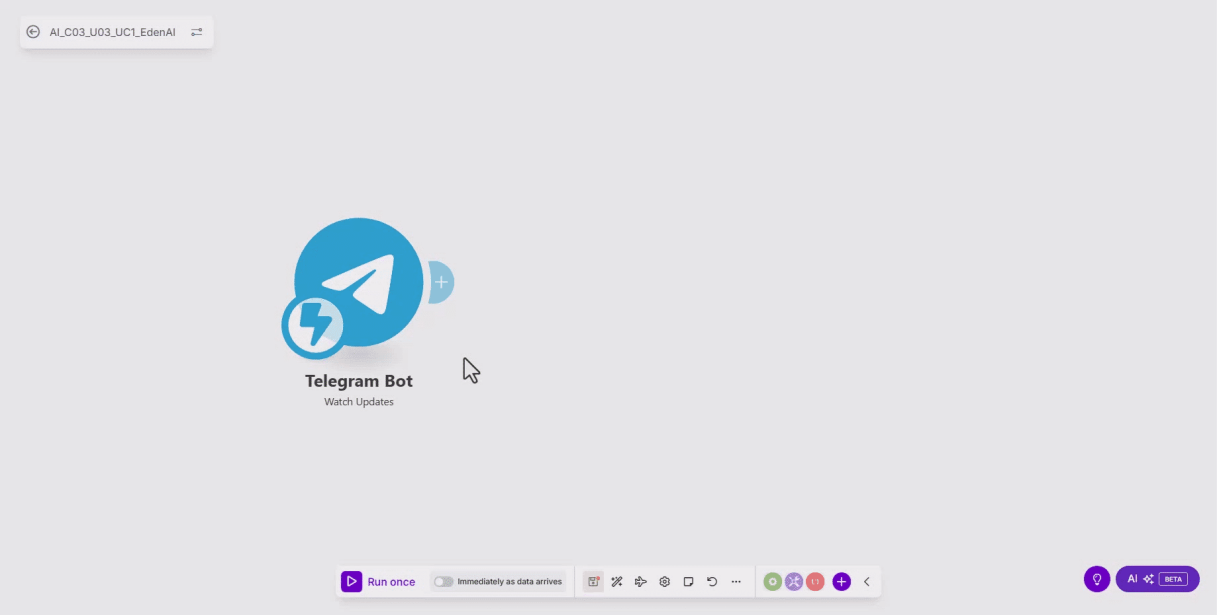
Click **Create a connection**. Paste the **API token** that you have obtained from the **BotFather** to connect to your bot. Click **Save** to save the connection.

5 Save

Click **Save** again to save your webhook and **Save** to save your module.

Step 2

Eden AI > Identify General Sentiment of a Text (2/7)



1 Add the module

To perform sentiment analysis on the message received add an **Eden AI > Identify General Sentiment of a Text** module.

2 Create the connection

Click **Create a connection** to link it to the Eden AI account you created earlier. Name the connection **Make_AI_EdenAI**.

Next, go to your Eden AI account and copy the **API Key**. Paste it into the API Key field and click **Save**.

3 Set up the module

In the **Text** field map the Message: Text from the **Telegram** module.

Next, you have to choose the AI provider that will perform the sentiment analysis. **OpenAI** provides the best results. Go ahead and select it.

For **Language**, select **Auto detection** from the dropdown menu, to let Eden AI automatically detect the language of the text message.

4 Save

Click **Save** to save the module.

Step 3

Filter (3/7)

To make sure that only text messages are passed through, **add a filter** between these first two modules.

1 Add the filter

Click on the link between them and select **Set up a filter** to open the filter setup. Label it **Text only**.

2 First condition

In the Condition, map the Message: Text item from the **Telegram** module and select **Basic operators: Exists**.

This ensures that only text messages are passed. For example, if a photo is sent, the Message: Text item won't be present, and the message won't

be passed to the **Eden AI** module.

3 Second condition

Click **Add AND rule** to add another condition. In the top field, map the Message: Text item again. Then select **Text operators: Not equal to** and type **/start** in the lower field. Press **Escape** to close the suggestion window.

The **/start** message is sent to the chat when you first interact with your bot, and you don't want to process it. This condition ensures that the **/start** message is filtered out.

4 Save

Click **Save** to save the filter.

Step 4

Eden AI > Generate Text (4/7)

1 Add the module

Add an **Eden AI> Generate Text module**. This will generate the text that you will use to reply to Telegram messages. No need to set up the connection again, the module will select the one you created earlier.

2 Set up

In the **Text** field paste the following prompt that contains the instructions to generate a reply to the customer message:

> You received the following message from a customer: [MAP MESSAGE]. Reply to the message: - Provide a simple response of around 20 words. - Keep a professional tone. - If the message is negative, offer a solution. - Answer in the same language of the customer.

Remove the **[MAP MESSAGE]** instruction (including the parentheses) and map the **Message: Text** from the **Telegram** module.

Select **OpenAI** as the **Provider** as it's the one with the best performance (sentiment detected with higher accuracy).

3 Save

Click **Save** to save the module.

Step 5

Router and Filter (5/7)

You want to get notified about negative messages, so let's create a route for this.

1 Add the module

Add a **Flow Control> Router** module.

2 Add the filter

Then click the **Filter** of the 1st route. Label it **Negative** and, in the **Condition** field, map the Result: General sentiment item from the Identify **General Sentiment of a Text module**.

Select **Text operators: Equal to (case insensitive)** and type **negative** in the lower field. Choose case insensitive to be on the safe side, in case AI returns

the sentiment with different capitalization.

3 Save

Click **Save** to save the filter.

This ensures that only messages with negative sentiment will pass through this route.

Step 6

Email> Send an Email to a Team Member (6/7)

1 Add the module

In the 1st route, add an **Email> Send an Email to a Team Member** module to receive notifications about the negative messages.

2 Set up

From the dropdown menu, select your email address and enter **Negative message from a customer** in the **Subject** field.

Then, highlight, copy, and paste the following message into the **Content** field.

```
There was a negative comment from <b> [MAP FIRST and LAST NAME]</b><br><br><b>Message:</b> <i>[MAP MESSAGE]</i><br><br> <b>Response sent:</b></br> <i> [MAP REPLY]</i>
```

Time to do some mapping.

Remove the **[MAP FIRST and LAST NAME]** instruction (including the parentheses) and map the `Message: From: First Name` and `Message: From: Last Name` items from the **Telegram** module.

Repeat the process for **[MAP MESSAGE]**, mapping the `Message: Text` item from the **Telegram** module.

Do the same for **[MAP REPLY]**, mapping the `Result: Generated text` item from the **Generate text** module.

3 Save

Click **Save** to save the module.

Note that the email message contains HTML tags for text formatting:

- **
** adds a line break in the text
- **** makes the enclosed text bold
- **** makes the enclosed text italic

Step 7

Telegram Bot> Send a Text Message or a Reply (7/7)

The final step is to send a reply to the user in the bot chat. All the messages, including the negative ones, will pass through the 2nd route, so add the module to send a reply in this route.

1 Add the module

Add a **Telegram Bot> Send a Text Message or a Reply** module. There's no need to set up the connection again, as it's reused from the last time you configured it.

2 Set up

In the **Chat ID** field, map the Message: Chat: ID item form the first **Telegram** module, which contains the chat ID with your bot. This ensures that the replies is sent to the correct chat.

Be careful not to select **Edited Channel Post: Message ID**. If you do, your use case won't work.

In the **Text** field map the **Result: Generated text** item from the **Generate text** module to use the reply generated by AI.

3 Save

Click **Save** to save the module and save your scenario as well.

Continue to 3.5.4: Run it

3.5.4 Run it

Time to test the use case.

Turn **ON** your scenario. In the **Schedule settings** pop up, check that **Immediately** is selected and click **Save**.

Then save your scenario as well.

Go to **Telegram** and open the chat with the bot that you created earlier. Click **/start** to begin chatting.

Let's test three different conditions.

Click each one to learn more.

Positive message 🙂

In the chat with the bot, type a positive message like:

This telescope is amazing! I was even able to see the moon with it!!

Press **Enter**.

Wait for the scenario to run and see the bot's reply to your message.
Note that it might take some time for the scenario to run.
This is the text generated by your scenario!
Cool right?

Negative message 😕

Now type a negative message like:

> **This telescope is awful! I wasn't able to observe dark matter!**

Press **Enter** and wait to see the reply from the bot.

Note that it might take some time for the scenario to run.
Check your inbox and you should have received the email that notifies you about this negative message.

Message in another language 🇮🇹

You can now try to type your message in another language, your mother tongue, any other language you speak, or use your friend Google Translate.

Since one of the creators of this course is Italian, you can type:

> **Che figo questo telescopio!**

Press **Enter**. Note that it might take some time for the scenario to run.

See the scenario replying to you in Italian, making you dream of pizza, spaghetti, parmesan, gelato, caffe, and all the amazing things they have over there.

Well done! You've tested your scenario. Now you know how to use AI tools in your scenario to handle customer messages.

Your job is done here. Go back to the **Use case menu** and pick another use case.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

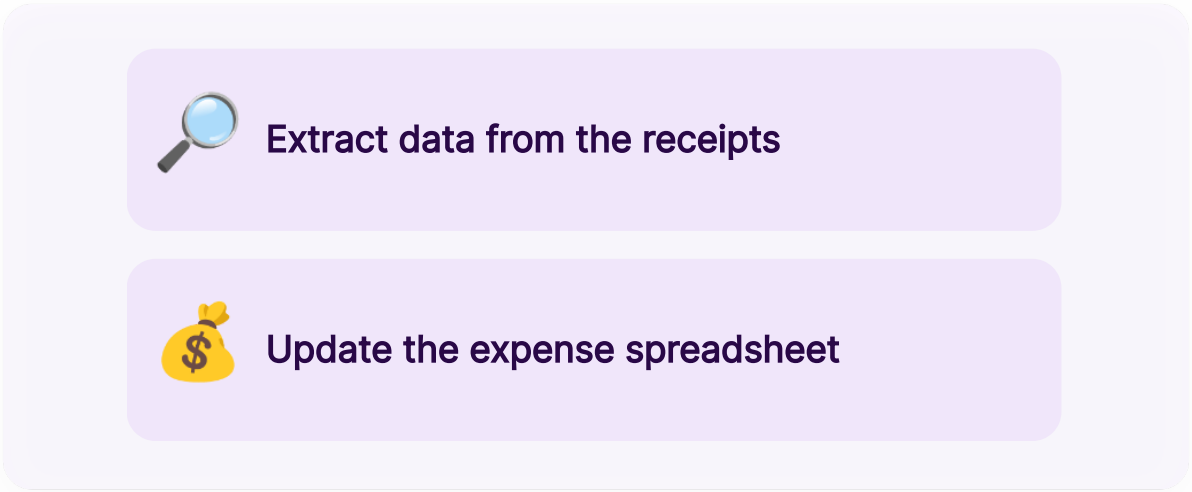
3.6 Use case 2: Extract data from pictures

Alicia travels frequently for work, and all her expenses are covered by the *Spider from Mars agency*.

To get reimbursed, she needs to fill out a spreadsheet with all the details. This process takes a lot of time, and she often misses something, having to pay for those *Life on Mars Bars* herself.

So, she has decided to take pictures of her receipts and save them as PDFs, and wants to create a scenario that helps her handle them.

The scenario will analyze the PDFs of the receipts and:



This is what you will build. If you'd like to try a different one, you can return to the Use case menu by clicking the button below, otherwise let's go build it!

Go to the Use case menu

TAKE ME BACK!

Before you start, let's have a look at the theory topic you will need to know for this use case:

1 Structured and unstructured data

1: Structured and unstructured data

1

Structured and unstructured data

AI can help analyze unstructured data.

Let's start by exploring structured and unstructured data. ***Click each one to learn more.***

Structured data

Structured data is information that is **organized in a clear way, for example in tables with rows and columns**. This makes it easy to apply mathematical operations, search and analyze.

Unstructured data

Unstructured data is information not organized in a set way. It can include things like emails, photos, and videos. It’s harder to deal with because it doesn’t have a clear structure, so finding useful information takes more work.

With structured data, **software can automatically access, process, and analyze information** without human help. It's clear what the data represents and where to find it. For example, in a table about sales, the software can take the sales numbers and calculate totals or averages.

Unstructured data, on the other hand, usually **needs manual review, cleaning, or organization** before it can be analyzed or used in automation. For instance, a person might need to extract sales numbers from a picture or a PDF and put them into a table so the software can process them.

AI can help analyze unstructured data by doing the work for you and organizing unstructured data into a structured format. It can identify key elements within the data, such as dates, names, or locations, and organize them into an easy-to-manage structured format.

Continue to 3.6.1: Choose how to build it

3.6.1 Choose how to build it

Now you know everything you need to build this use case. All that's left is to choose which apps you want to use. Pick your favorite, you can always come back later and try the other option too.

Click the button to see the building instructions.

Make AI apps

BUILD IT!

Eden AI

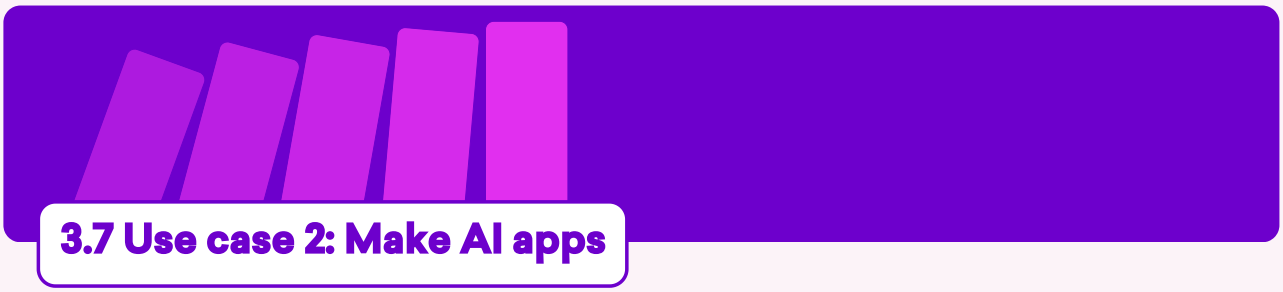
BUILD IT!

If you don’t feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!



3.7 Use case 2: Make AI apps

You have decided to build use case 2 with Make AI apps. Amazing choice! For this use case you will use the Make AI Content Extractor.

For this app you don't have to set up any AI provider. Make AI Content Extractor uses the **Make's AI provider** by default. Make has also configured the LLM module, the prompt, and settings for you. All you need to do is provide the data, and the module will process it.

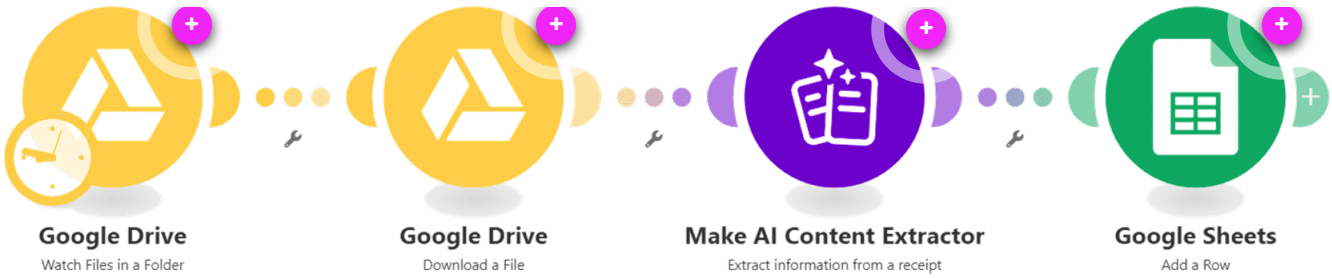
Continue to 3.7.1: Scenario overview

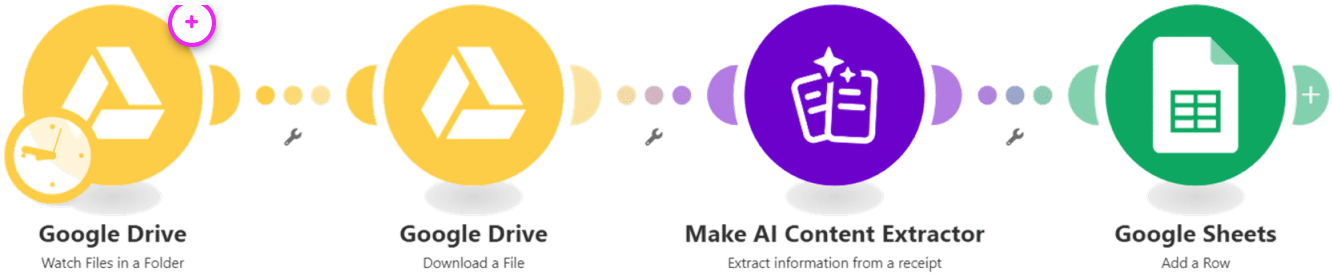
3.7.1 Scenario overview

Let's have a closer look at the scenario you will build.

Here's an overview of the scenario.

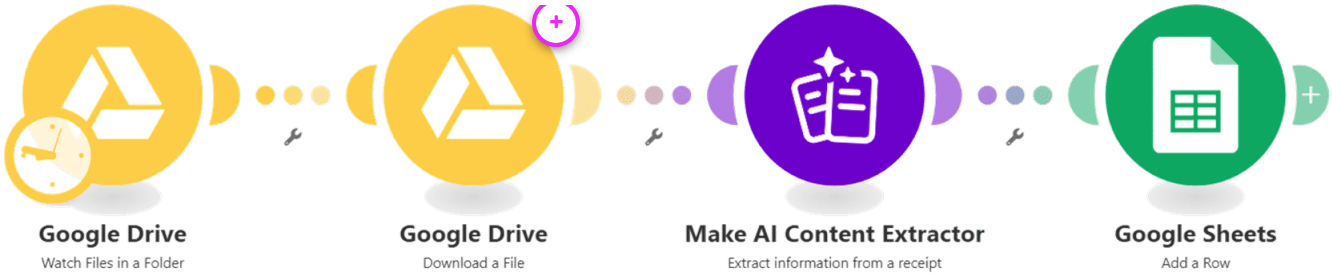
Click each + to learn more.


Google Drive
Watch Files in a Folder
Google Drive
Download a File
Make AI Content Extractor
Extract information from a receipt
Google Sheets
Add a Row



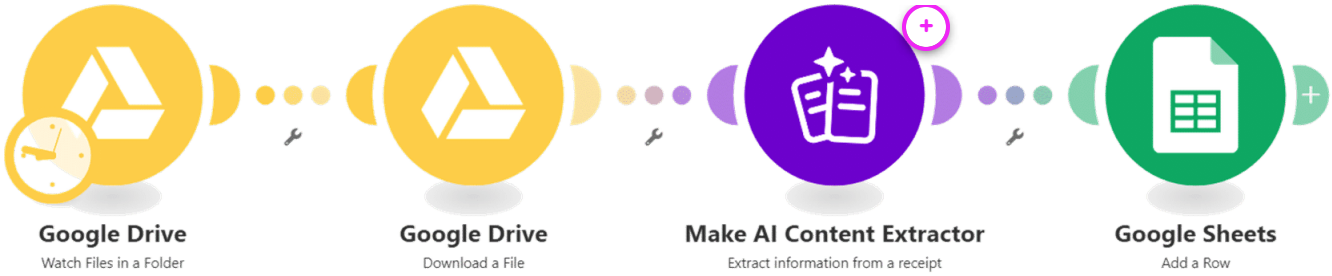
Google Drive> Watch Files in a Folder

Watch for new PDFs of a receipt in a Google Drive and return the file ID that can be used for downloading.



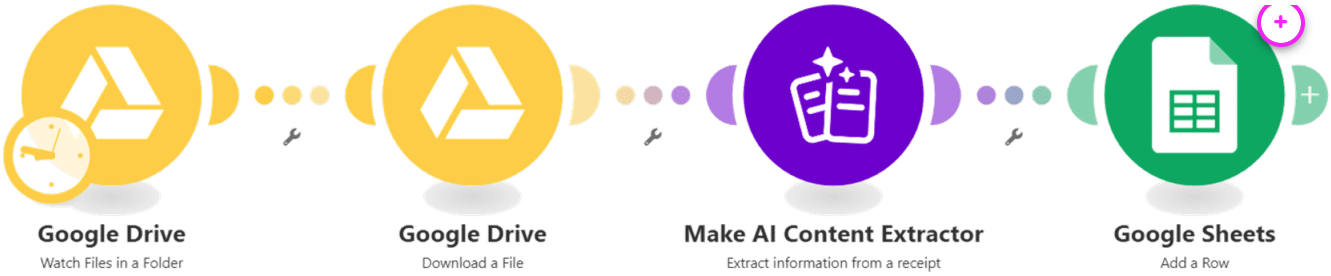
Google Drive> Download a File

Download the PDF of the receipt.



Make AI Content Extractor> Extract information from a receipt

Use AI to extract information from the image of a receipt, organizing unstructured data.



Google Sheets> Add a Row

Save the receipt data extracted by AI to a Google Sheet.

If you don’t feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

Continue to 3.7.2: Use case preparation

3.7.2 Use case preparation

Let's make sure everything is ready before building the use case.

First, download the three images of the receipts.

Click each download button below to get them.

FILE

receipt1.png

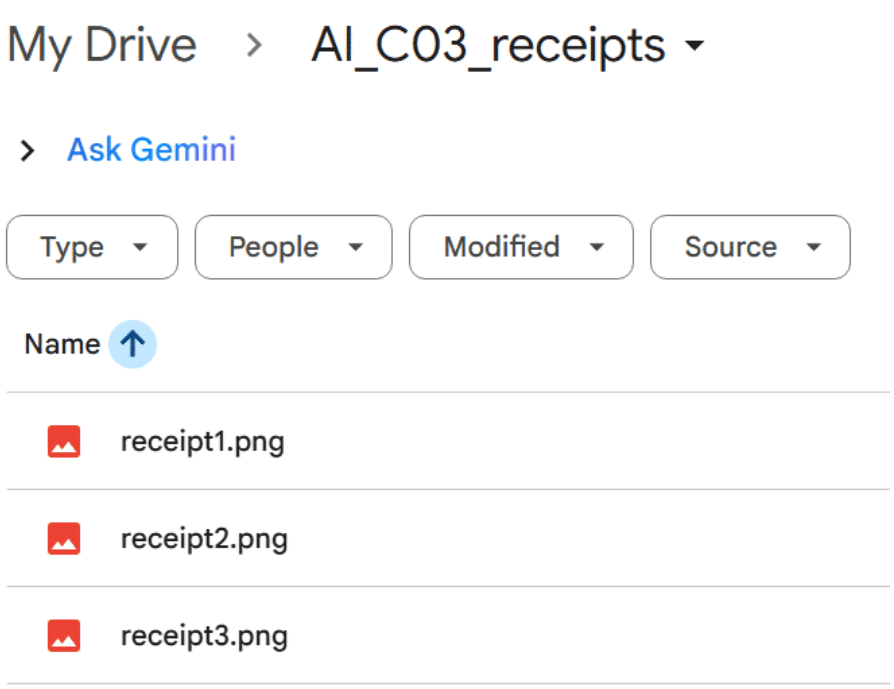
57.1 KB

Let's prepare everything you need for the use case.

Work through each stage before you continue.

Step 1

Prepare the Google Drive with the receipts



To get started, create a **Google Drive folder** to store the PDFs of your receipts.

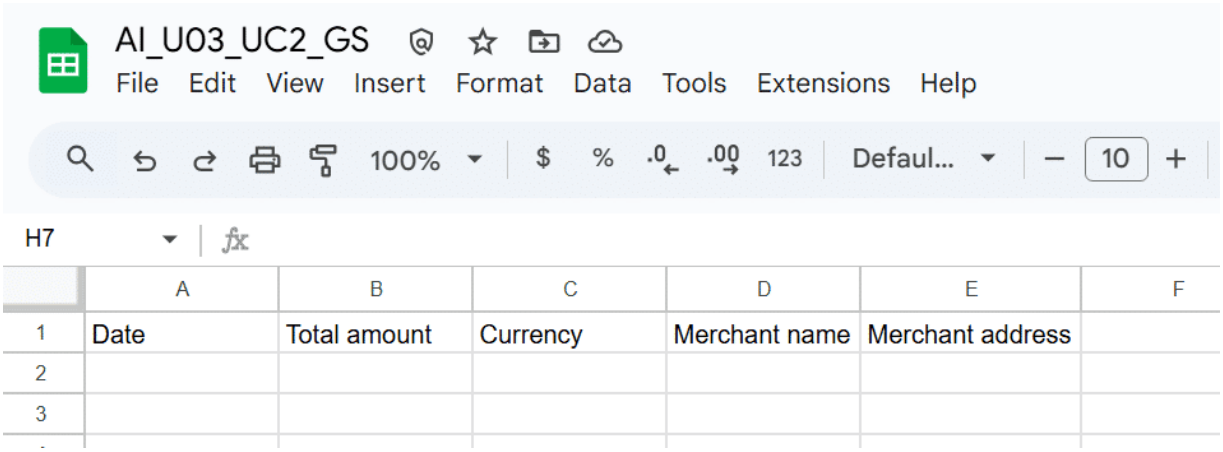
Go to Google Drive and create a new folder. Name it **AI_C03_receipts**.

Upload the receipts that you have downloaded earlier to this folder.

Note that the folder only needs to contain the receipts and no other files, since the scenario will detect all the items that are present.

Step 2

Create the spreadsheet



Next, you need to create the **Google Sheet** to store the receipt information extracted by AI.

Go to your **Google Drive** and, in a different folder to the one you created before, create a new **Google Sheet**. Name it **AI_U03_UC2_GS**.

Name the columns **Date**, **Total amount**, **Currency**, **Merchant name**, and **Merchant address**.

You're all set! Time to build the use case.

Continue to 3.7.3: Build it

3.7.3 Build it

This use case uses Google Drive. To use it within Make, you need to create a connection. If you're on a free Google account, follow these instructions to authenticate.

Let's build the scenario.

Work through each stage before you continue.

Step 1

Google Drive> Watch Files in a Folder

1 Create the scenario

In Make, create a new scenario and call it **AI_C03_U03_UC2_MakeAI**.

2 Add the module

To detect the files in your folder, add a **Google Drive> Watch Files in a Folder** module.

3 Create the connection

Click **Create a connection** to connect it to your **Google** account. Name the connection **Make_AI**, then click **Sign in with Google** and follow the steps.

4 Set up

Under **Select the Folder to be Watched**, choose the folder you created earlier (**AI_C03_receipts**) and change the **Limit** from 2 to 5, to increase the maximum

number of results returned and pick up all the 3 receipts.

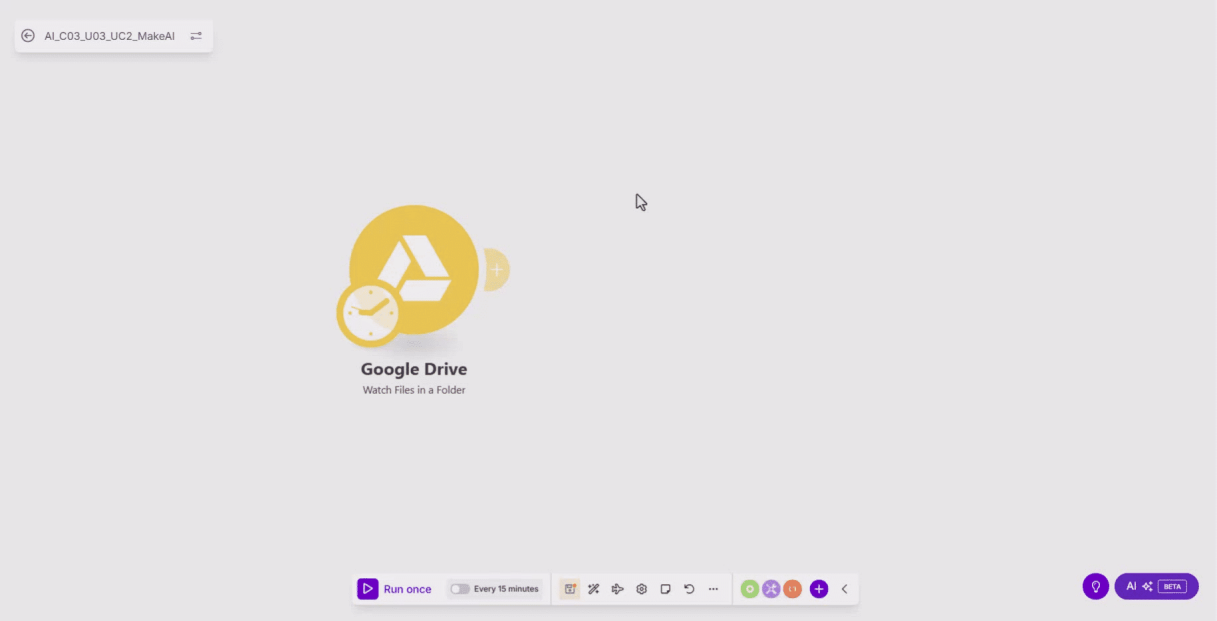
5 Save

Click **Save** to save the module.

When prompted to **Choose where to start**, select **All** and click **Save**.

Step 2

Google Drive> Download a File



Next, you want to download the files from your **Google Drive** folder.

1 Add the module

To do this, add a **Google Drive> Download a File** module. Notice that the connection with your **Google** account is already established, so you don't have to do anything for this.

2 Set up

Under **File ID**, map the File ID item from the first module, it is the ID of the receipts in the folder.

3 Save

Click **Save** to save the module.

Step 3

Make AI Content Extractor> Extract information from a receipt

1 Add the module

To extract the information from the PDFs of your receipts, add a **Make AI Content Extractor> Extract information from a receipt** module.

2 Set up

For **Receipt source**, select **File** from the dropdown menu to specify how you will provide the receipt to the module. In this case, you will use a file instead of a URL.

Under **File**, choose **Google Drive - Download a File** to specify that the receipts downloaded in the previous module will be the input file for this module.

3 Save

Click **Save** to save the module.

Note that the module shows how many credits it will use for each bundle.

Step 4

Google Sheets> Add a row

The final step is to add a **Google Sheet** module to save the information extracted from your receipts using AI into the spreadsheet.

1 Add the module

Add a **Google Sheets> Add a row** module, and connect to your **Google account** if it isn't picked up already.

2 Set up

Under **Spreadsheet ID**, select the **Google Sheet** you have created earlier (**AI_U03_UC2_GS**), and under **Sheet Name**, choose **Sheet 1**.

Now, map the extracted elements to the corresponding columns.

- In the **Date (A)** field, map `Transaction time: Content`.

- For **Total amount (B)**, map `Total: Content`.
- In the **Currency (C)** field, map `Subtotal: Value currency: Currency symbol`.
- For **Merchant name (D)**, map `Merchant name: Content`.
- For **Merchant address (E)**, map `Merchant address: Content`.

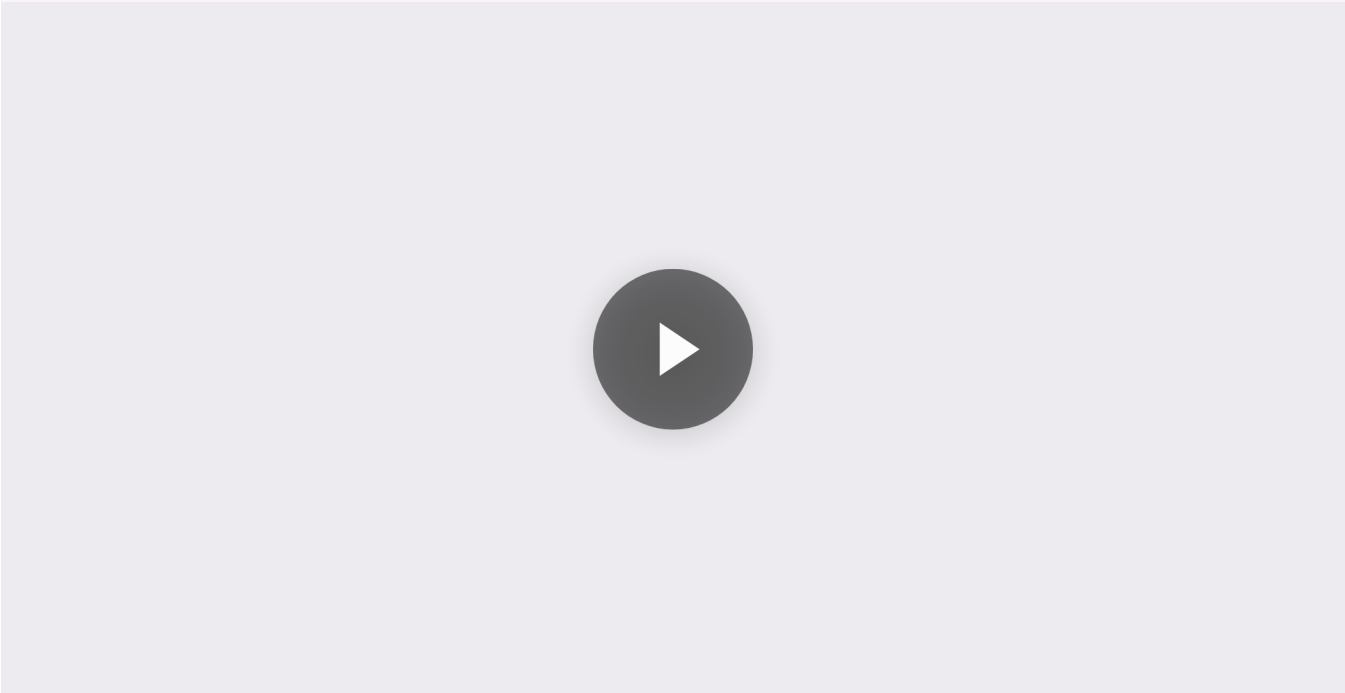
3 Save

Click **Save** to save the module and save the scenario as well.

Continue to 3.7.4: Run it

Time to test the use case.

In the scenario, click **Run once**. When your scenario finishes processing all three receipts, go and open your **Google Sheet**: the scenario has filled in the extracted information.



Note that sometimes not all the items are detected, which may result in empty columns. You may need to manually check the results to ensure all data is present. As you've learned in the

previous courses, while AI is very powerful, it's not perfect, and human intervention can improve its accuracy.

Well done! You know how to use AI to extract information from unstructured data.

Your job is done here. Go back to the **Use case menu** and pick another use case. ***Click the button below.***

Go to the Use case menu

TAKE ME BACK!

3.8 Use case 2: Eden AI

You have decided to build use case 1 with Eden AI. Great choice!

If you haven't seen it yet, it’s now time to learn what Eden AI is, how it works, and how to set up your account so you can start using it in your scenarios.

To do this, head over to the dedicated Eden AI section. You can access it from any of the three Eden AI use cases, and at the end you’ll find a button to return here and start building.

Eden AI information and set up

LEARN MORE

If you have already set up your Eden AI account, skip this step and have a look at the scenario you will build.

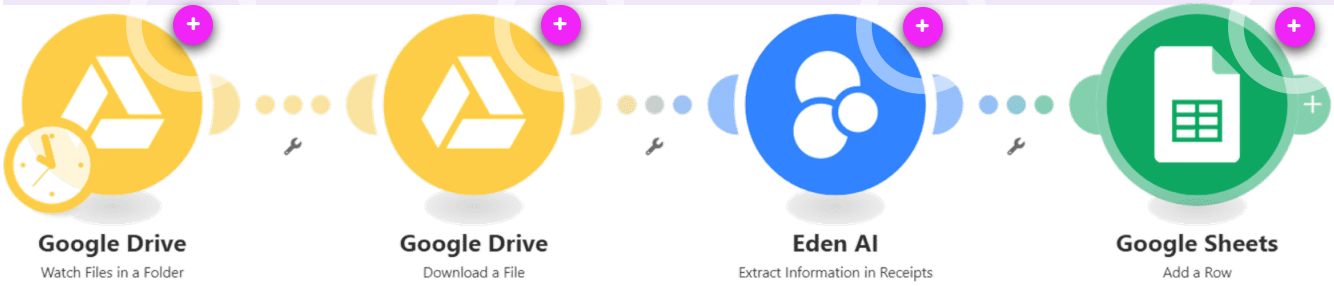
Continue to 3.8.1: Scenario overview

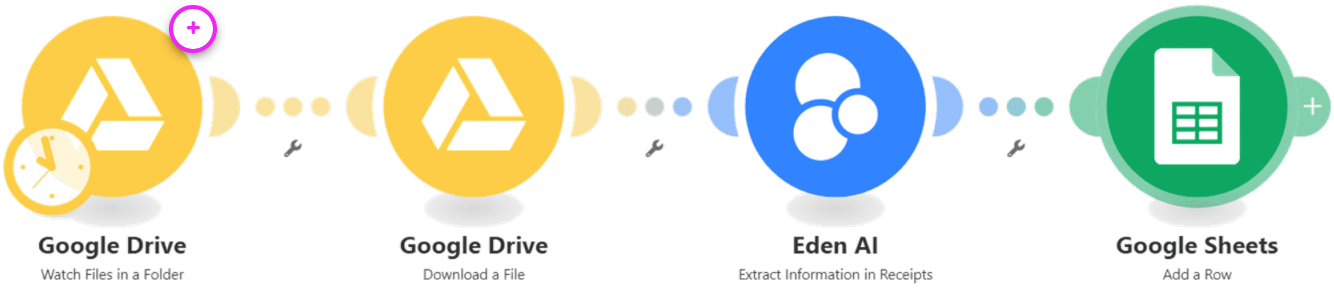
3.8.1 Scenario overview

Now that you have all the knowledge you need and your Eden AI account, let's have a closer look at the scenario you will build.

Here's an overview of the scenario.

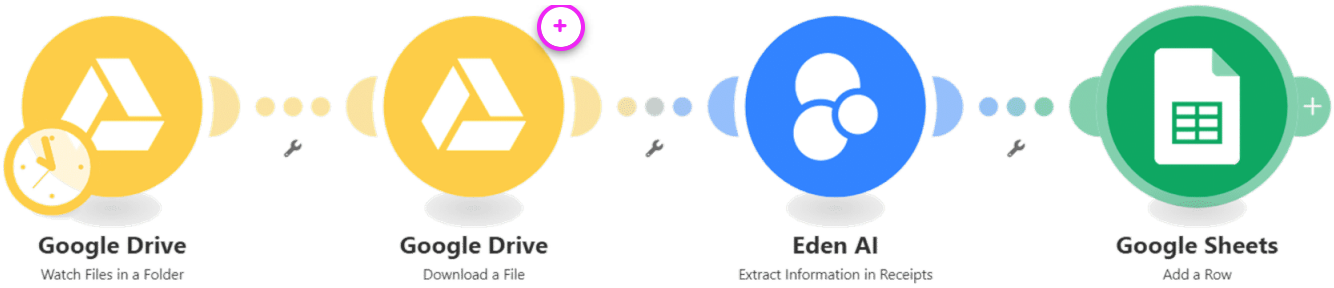
Click each + to see what each module does.





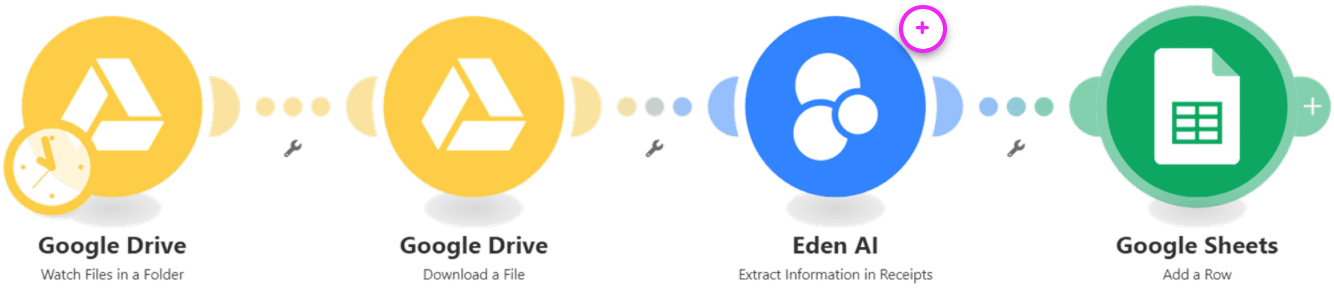
Google Drive> Watch Files in a Folder

Watch for new PDFs of a receipt in a Google Drive and return the file ID that can be used for downloading.



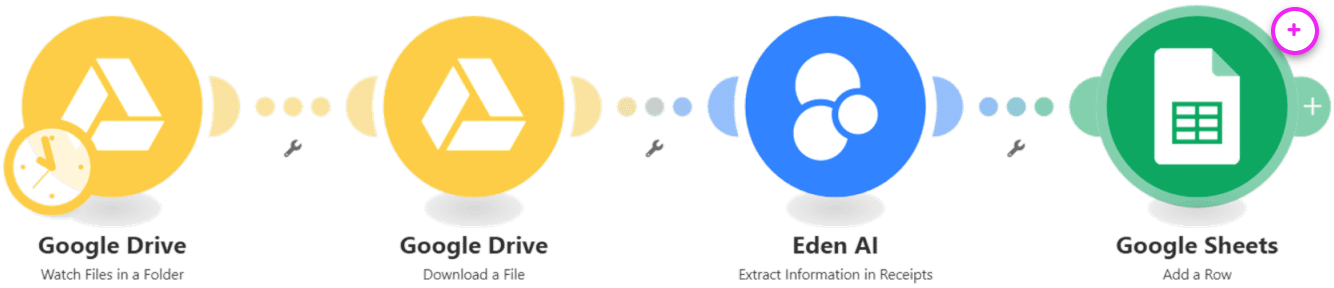
Google Drive> Download a File

Download the PDF of the receipt.



Eden AI> Extract Information in Receipts

Use AI to extract information from the image of a receipt, organizing unstructured data.



Google Sheets> Add a Row

Save the receipt data extracted by AI to a Google Sheet.

If you don't feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

Continue to 3.8.2: Use case preparation

3.8.2 Use case preparation

Let's make sure everything is ready before building the use case.

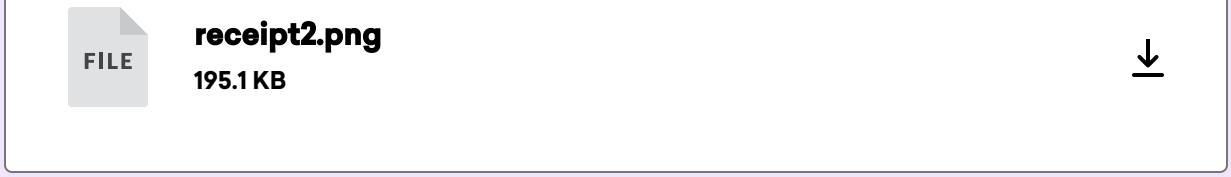
First, download the three images of the receipts.

Click each download button below to get them.

FILE

receipt1.png

57.1 KB

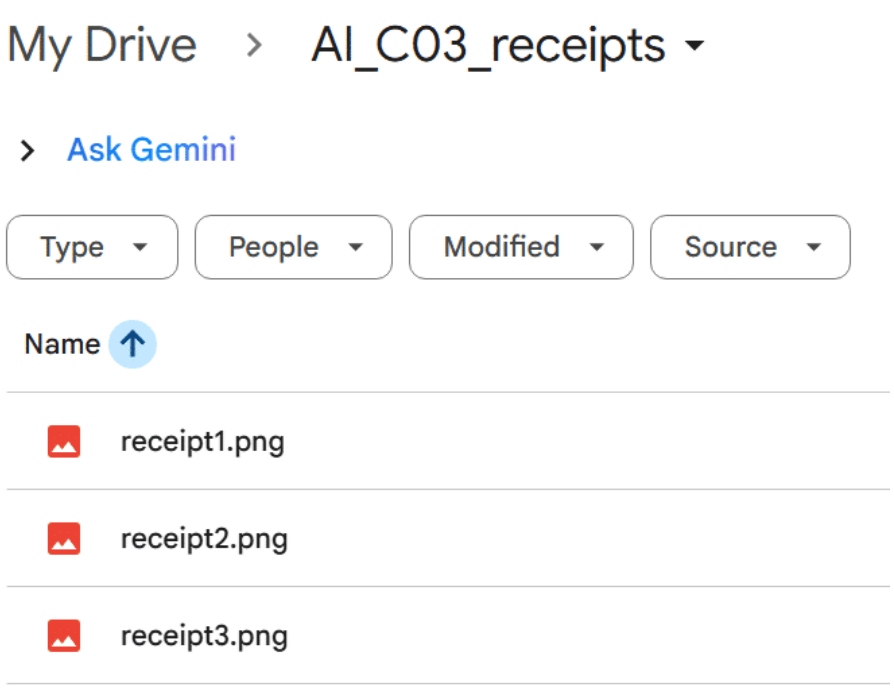


Let's prepare everything you need for the use case.

Work through each stage before you continue.

Step 1

Prepare the Google Drive with the receipts



To get started, create a **Google Drive folder** to store the PDFs of your receipts.

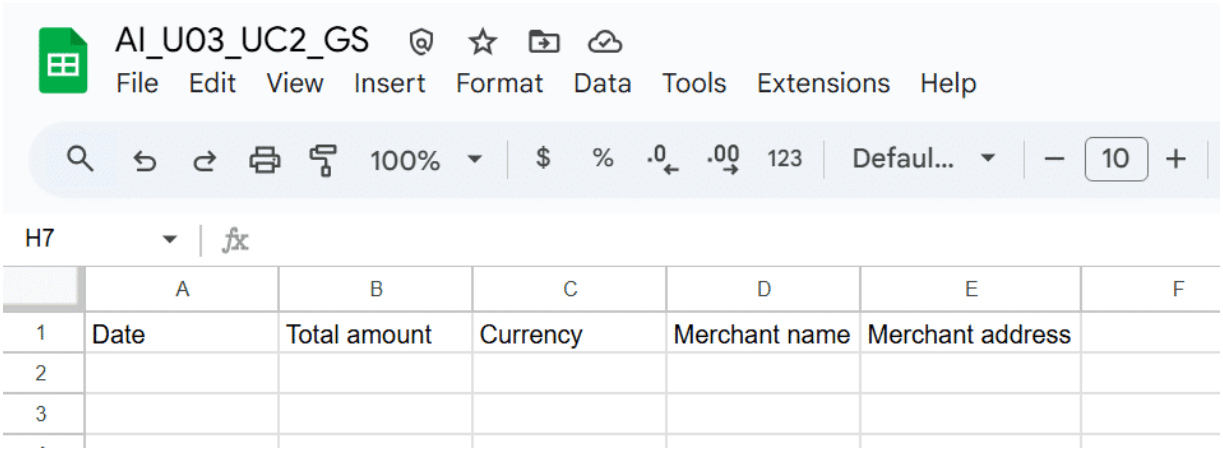
Go to Google Drive and create a new folder. Name it **AI_C03_receipts**.

Upload the receipts that you have downloaded earlier to this folder.

Note that the folder only needs to contain the receipts and no other files, since the scenario will detect all the items that are present.

Step 2

Create the spreadsheet



Next, you need to create the **Google Sheet** to store the receipt information extracted by AI.

Go to your **Google Drive** and, in a different folder to the one you created before, create a new **Google Sheet**. Name it **AI_U03_UC2_GS**.

Name the columns **Date**, **Total amount**, **Currency**, **Merchant name**, and **Merchant address**.

You're all set! Time to build the use case.

Continue to 3.8.3: Build it

3.8.3 Build it

This use case uses Google Drive. To use it within Make, you need to create a connection. If you're on a free Google account, follow these instructions to authenticate.

Let's build the scenario.

Work through each stage before you continue.

Step 1

Google Drive> Watch Files in a Folder

1 Create the scenario

In Make, create a new scenario and call it **AI_C03_U03_UC2_EdenAI**.

2 Add the module

To detect the files in your folder, add a **Google Drive> Watch Files in a Folder** module.

3 Create the connection

Click **Create a connection** to connect it to your **Google** account. Name the connection **Make_AI**, then click **Sign it with Google** and follow the steps.

4 Set up

Under **Select the Folder to be Watched**, choose the folder you created earlier (**AI_C03_receipts**) and change the **Limit** from 2 to 5, to increase the maximum

number of results returned and pick up all the 3 receipts.

5 Save

Click **Save** to save the module.

When prompted to **Choose where to start**, select **All** and click **Save**.

Step 2

Google Drive> Download a File

Next, you want to download the files from your **Google Drive** folder.

1 Add the module

To do this, add a **Google Drive> Download a File** module. Notice that the connection with your **Google** account is already established, so you don't have to do anything for this.

2 Set up

Under **File ID**, map the File ID item from the first module, it is the ID of the receipts in the folder.

3 Save

Click **Save** to save the module.

Step 3

Eden AI> Extract information from a receipt

1 Add the module

To extract the information from the PDFs of your receipts, add an **Eden AI> Extract Information in Receipts** module.

If you have already set up the connection with EdenAI, Make will select it automatically. You can skip the connection part below and continue with the module set up.

2 Create a connection

If you haven't already set up a connection with Eden AI, click **Create a connection** to link it to the Eden AI account you created earlier. Name the connection, **Make_AI_EdenAI**.

Next, go to your Eden AI account and copy the API Key. Paste it into the **API Key** field and click **Save**.

3 Set up

Under **File**, choose **Google Drive - Download a File** to specify that the receipts downloaded in the previous module will be the input file for this module.

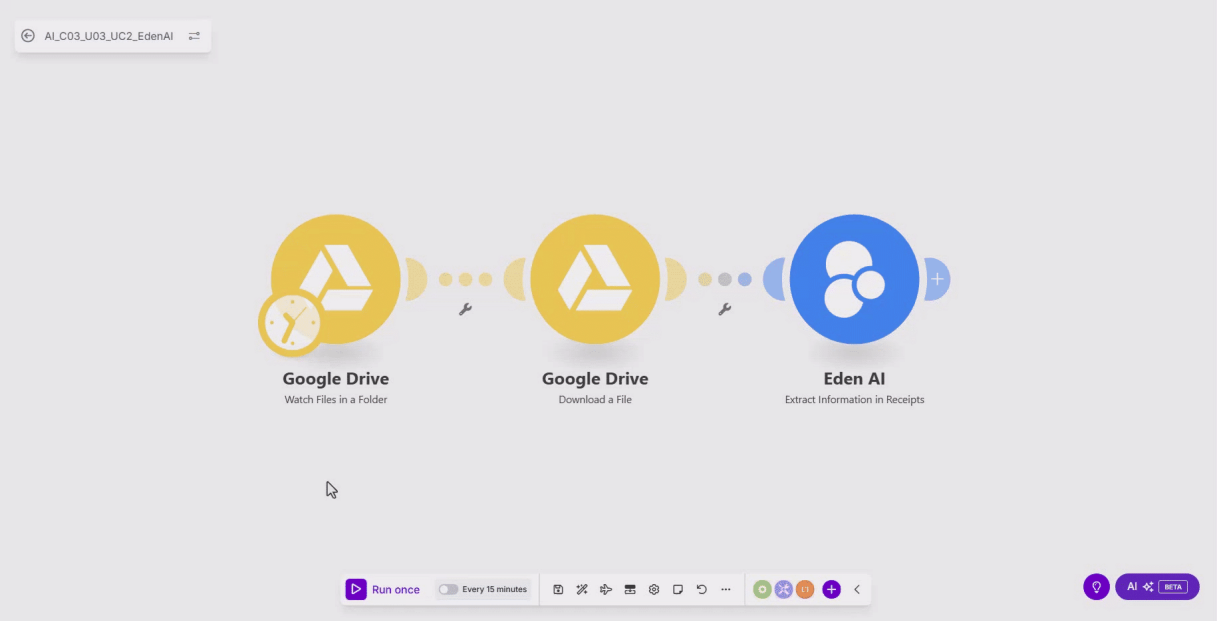
Select **AWS** as the provider, as it performed the best during testing performed by the Make Academy team. Choose **English** as the **Language**.

3 Save

Click **Save** to save the module.

Step 4

Run the scenario



Now, you need to run your scenario so that different items from the **Eden AI> Extract Information in Receipts** module are available for mapping.

After saving your scenario, click **Run once** and see that the scenario runs and process the data.

The scenario might take a while to run while Eden AI is processing all the data. Nothing for you to do here, just wait patiently.

You can check the output of the **Eden AI** module to see the cost of extracting the information from the receipts.

Step 5

Google Sheets> Add a row

The final step is to add a **Google Sheet** module to save the information extracted from your receipts using AI into the spreadsheet.

1 Add the module

Add a **Google Sheets> Add a row** module, and connect to your **Google account** if it isn't picked up already.

2 Set up

Under **Spreadsheet ID**, select the **Google Sheet** you have created earlier (**AI_U03_UC2_GS**), and under **Sheet Name**, choose **Sheet 1**.

Now, map the extracted elements to the corresponding columns. You can find all the items under `Results: Extracted data` in the **Eden AI** module.

- In the **Date (A)** field, map `Results: Extracted data: Date`.
- For **Total amount (B)**, map `Results: Extracted data: Invoice total`.
- In the **Currency (C)** field, map `Results: Extracted data: Locale.Currency`.
- For **Merchant name (D)**, map `Results: Extracted data: Merchant information.Merchant name`.
- For **Merchant address (E)**, map `Results: Extracted data: Merchant information.merchant_address`.

3 Save

Click **Save** to save the module and save the scenario as well.

Continue to 3.8.4: Run it

3.8.4 Run it

Time to test the use case.

Right-click on the **Google Drive> Watch Files in a Folder** module and select **Choose where to start**. In the pop up window, select **All** to ensure all receipts are detected and click **Save**.

Click **Run once**. When your scenario finishes processing all three receipts, go and open your **Google Sheet**: the scenario has filled in the extracted information.

Note that sometimes not all the items are detected, which may result in empty columns. You may need to manually check the results to ensure all data is present. As you've learned in the previous courses, while AI is very powerful, it's not perfect, and human intervention can improve its accuracy.

Well done! You know how to use AI to extract information from unstructured data.

Your job is done here. Go back to the **Use case menu** and pick another use case.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

3.9 Use case 3: Social media post creation

Alicia runs a blog where she shares space-related news and information. She regularly posts articles on popular topics, but she is very busy and would like a scenario to assist her.

She saves the titles and descriptions of the topics she wants to discuss in her blog in a spreadsheet. The scenario will **use the information in the spreadsheet** and:

✍️ **Create a post for a specific topic**

Create a post for a specific topic

Save the post in a Google document, using a Google Docs template

You can use AI tools to generate images for your blog posts. Although this use case doesn't include it to avoid consuming too many credits, you can explore this option if you're interested.

This is what you will build. If you'd like to try a different one, you can return to the Use case menu by clicking the button below, otherwise let's go build it!

Go to the Use case menu

TAKE ME BACK!

Before you start, let's have a look at the theory topic you will need to know for this use case:

1 Google Docs templates

1: Google Docs templates

1

Google Docs templates

A Google Docs template is a pre-made file you can use as a starting point to create a new document. Make created this approach!

In Make you can use the **Google Doc > Create a Document from a Template** module to create a document from a template and fill it with the data from other modules.

The template is a **Google Docs** document that has placeholders for the information you want to add.

Placeholders are pieces of text inside double curly brackets: **{{text}}**.

They represent **dynamic content**, which gets replaced with actual data when you process the document. For example, **{{name}}** in a template becomes a specific person's name.

It refers to information that changes or is customized based on data, context, or user interaction.

With Make, you can set up templates with placeholders like **{{text}}**, and the module will automatically replace them with relevant data. This allows you to create dynamic documents without manually entering information each time.

Additionally, Make will **create a new document** to ensure the original remains intact for future use.

This is what you will be using when building this scenario.

Continue to 3.9.1: Choose how to build it

3.9.1 Choose how to build it

Now you know everything you need to build this use case. All that's left is to choose which apps you want to use. Pick your favorite, you can always come back later and try the other option too.

Click the button to see the building instructions.

Make AI apps

BUILD IT!

Eden AI

BUILD IT!

If you don’t feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

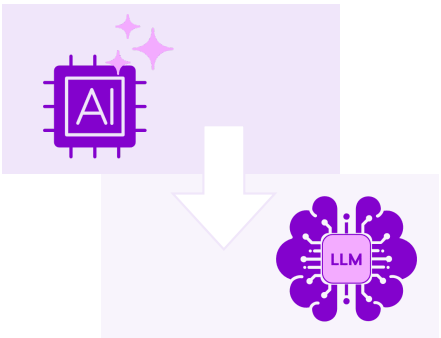
Go to the Use case menu

TAKE ME BACK!

3.10 Use case 3: Make AI apps

You have decided to build use case 3 with Make AI apps. Amazing choice!

For this use case, you will use the Make AI Toolkit.



AI provider

As you've seen in the previous section, to use the modules of this app you need to connect it to an AI provider. In the use case you will use the **Make's AI provider**.

If you're on Pro plan and above, you can also choose other AI providers if you have a connection available and want to test it. Just note that the results might be different from the ones shown here.

Model

For the LLM model, follow the best practice from the previous units: start with the smaller model and switch to a larger one if the results aren't good enough. For this use case, the Make team tested the Small model, and it works well.

Continue to 3.10.1: Scenario overview

3.10.1 Scenario overview

Let's have a closer look at the scenario you will build.

Here's an overview of the scenario.

Click each + to learn more.


Google Sheets
Watch New Rows
Make AI Toolkit
Simple Text Prompt
Google Docs
Create a Document from a Template

Google Sheets> Watch New Rows

Check the Google Sheet for new rows containing the information to generate the blog post.

Make AI Toolkit> Simple Text Prompt

Using a prompt and the information stored in the Google Sheet, generate the blog post.

Google Docs> Create a Document from a Template

Create a Google Doc containing the text of the blog post starting from a template. The document will contain the title of the blog and the blog post. The file name would be the same as the blog title.

If you don't feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

Continue to 3.10.2: Use case preparation

3.10.2 Use case preparation

For this use case, you need two things: the Google Sheet with the blog post topics and the Google Docs template that the scenario will use to create the documents with the AI-generated blog posts.

Start by ***copying the text below***, it is the information that you will paste in the spreadsheet.

Title	Summary

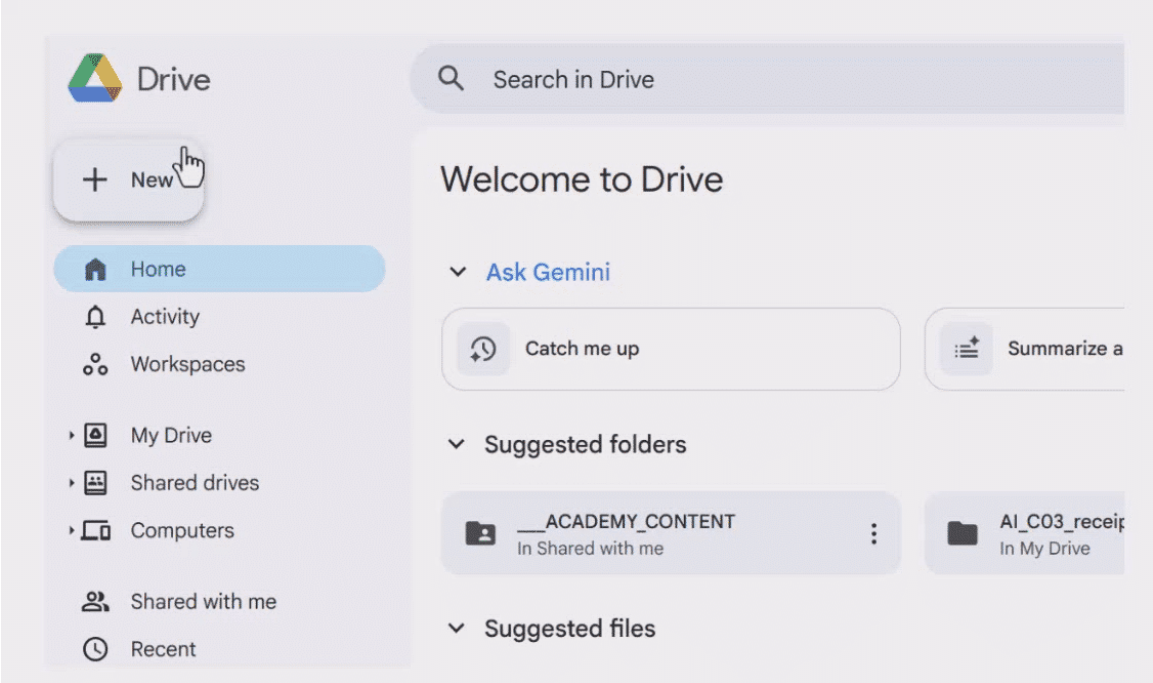
Poor Pluto	Pluto is no longer a planet
Put a ring on it	Saturn has rings made of ice and rock

Let's keep preparing everything you need for the use case.

Work through each stage before you continue.

Step 1

Google Sheet



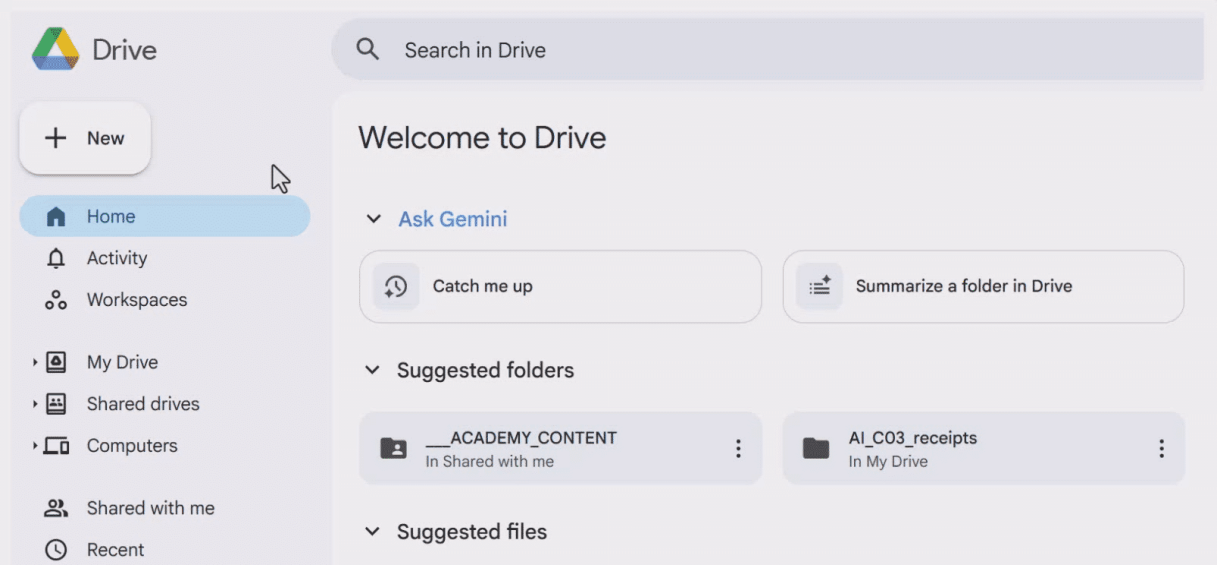
The first thing you need to do is to create a **Google Sheet** that stores the topics of your next blog posts.

Go to **Google Drive**, create a new **Google Sheet** and name it **AI_U03_UC3_blog**.

Paste the content you just copied it into the sheet.

Step 2

Google Docs template



Next, you need to create the template that your scenario will use to create a document for saving the blog content it generates.

In **Google Drive** create a new **Google Doc** and call it **AI_U03_UC3_template**.

Select and copy this text placeholder:

{{title}}

{{content}}

Paste it in the document and you're ready to start building the scenario.

Continue to 3.10.3: Build it

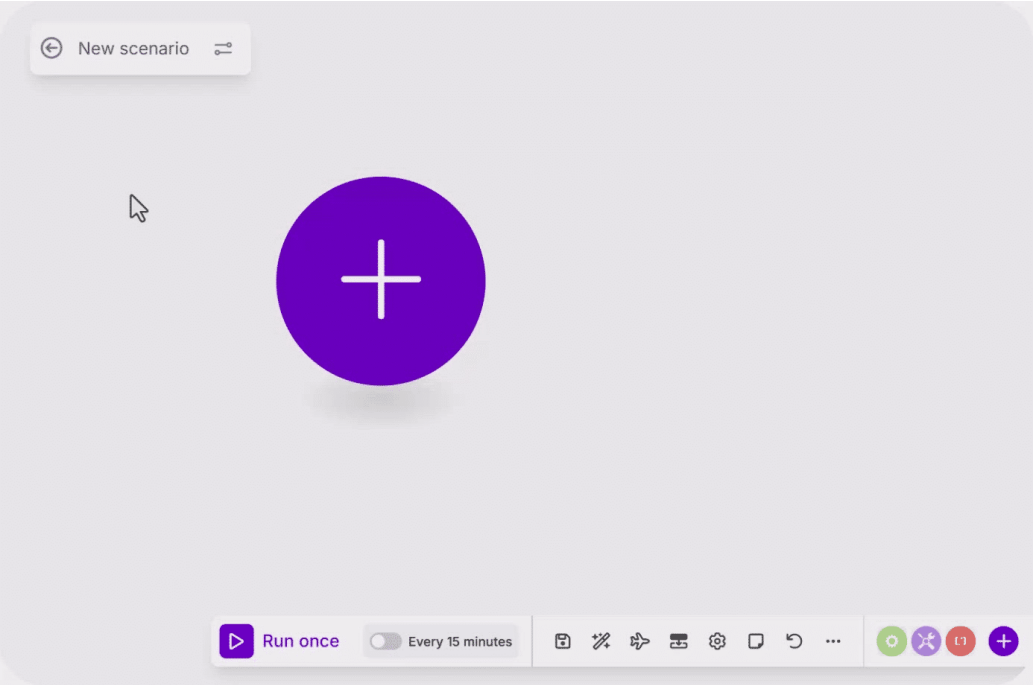
3.10.3 Build it

Let's build the scenario.

Work through each stage before you continue.

Step 1

Google Sheets> Watch New Rows



1 Create the scenario

In Make, create a new scenario and call it **AI_C03_U03_UC3_MakeAI**.

2 Add the module

To detect any new row in your spreadsheet, add a **Google Sheets> Watch New Rows** module.

3 Create the connection

Click **Create a connection** to connect it to your **Google** account if it is not automatically set up. Name the connection **Make_AI**, then click **Sign in with Google** and follow the steps.

4 Set up

Under **Spreadsheet ID**, select the **Google Sheet** you have created earlier (**AI_U03_UC3_blog**), and under **Sheet Name**, choose **Sheet 1**.

5 Save

Click **Save** to save the module. When prompted to **Choose where to start**, select **All** and click **Save**.

Step 2

Make AI Toolkit> Ask anything

Next, you want to generate the text according to the content in the Google Sheet.

1 Add the module

Add the **Make AI Toolkit> Simple Text Prompt** module.

2 Create the connection

If you're on a Pro plan or above, click **Create a connection** to connect it to an AI provider. Choose **Make's AI provider**.

If you're on Free or Core plan, **Make's AI provider** is already selected.

Name the connection **Make_AI_Toolkit**, and click **Save**.

3 Set up the module

For the **Model**, select **Small** from the dropdown menu.

In the Text field copy and paste this prompt to generate the blog post:

> From this title: "[MAP TITLE]" and this summary: "[MAP SUMMARY]", generate a short text of maximum 200 words that covers the subject. The tone has to be neutral, easy to read for readers that don't know the subject.

Remove the **[MAP TITLE]** instruction (including the parentheses **[]**, but not the quotations **""**) and map the **Title** item from the **Google Sheets** module.

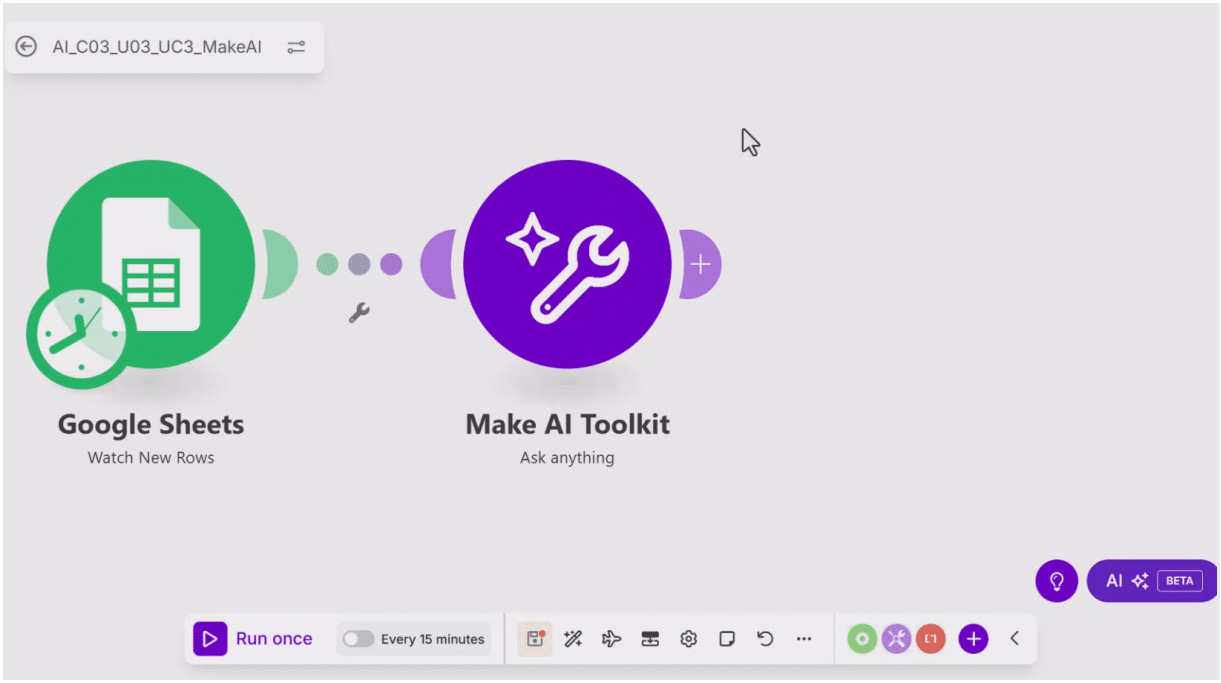
Repeat the process for **[MAP SUMMARY]**, mapping the **Summary** item from the **Google Sheets** module.

4 Save

Click **Save** to save the module.

Step 3

Google Docs> Create a Document from a Template



1 Add the module

To save the generated text into the template, add a **Google Docs> Create a Document from a Template** module.

2 Set up the connection

For the **Connection**, select the **Make_AI** connection you created earlier from the dropdown menu. Click **Continue** to update the connection permission and follow the instructions.

3 Set up

Under **Document ID**, select the Google Doc template you created earlier (**AI_U03_UC3_template**).

In the **Value** section you can map the information in the placeholders.

Under **Title** map the Title item from the **Google Sheets** module, while for the **Content**, map the Answer from the **Make AI Toolkit** module.

For the **Title** of the document that represents the file name, map again the Title item from the **Google Sheet** module.

Under **New Document's Location**, choose where in **Google Drive** you want to save the documents.

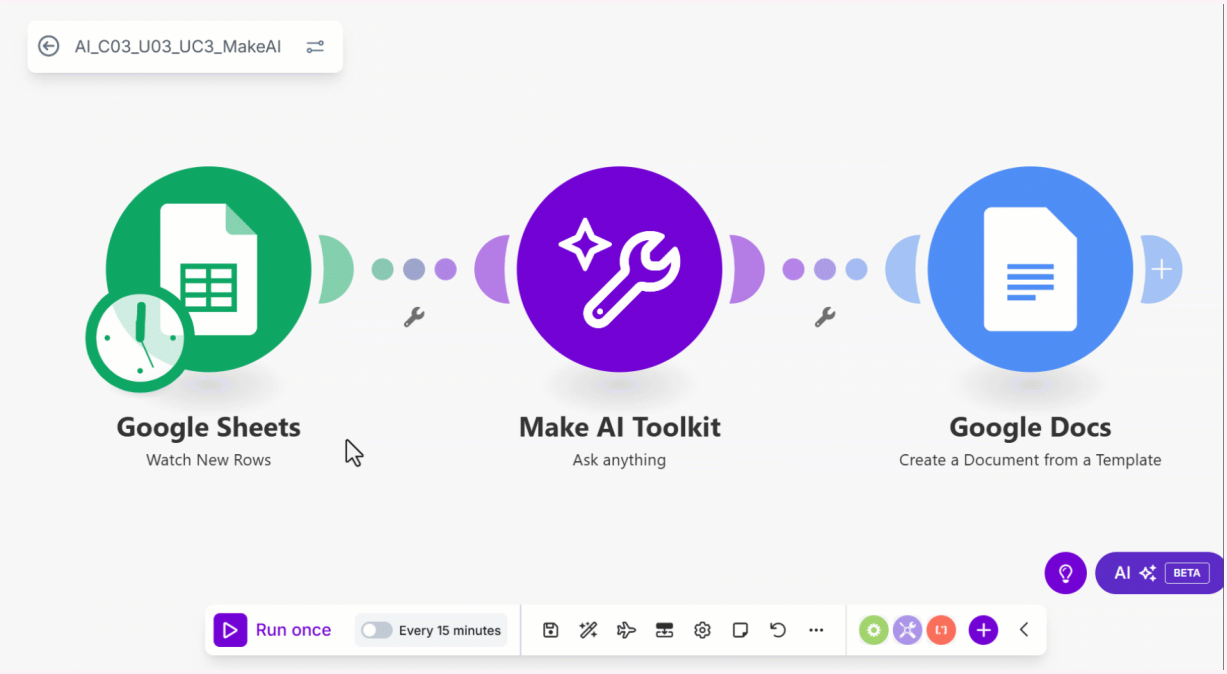
4 Save

Click **Save** to save the module and save your scenario as well.

Continue to 3.10.4: Run it

3.10.4 Run it

Time to test the use case.



After saving your scenario, click **Run once**. When your scenario finishes processing, go and have a look at the folder where you saved your documents.

Open the documents and read the blog post that AI generated for you!

Well done! You now know how to use AI to create content and save it in documents using a template.

Your job is done here. Go back to the **Use case menu** and pick another use case. ***Click the button below.***

Go to the Use case menu

TAKE ME BACK!

3.11 Use case 3: Eden AI

You have decided to build use case 3 with Eden AI. Great choice!

Now it’s time to learn what Eden AI is, how it works, and how to set up your account so you can start using it in your scenarios.

To do this, head over to the dedicated Eden AI section. You can access it from any of the three Eden AI use cases, and at the end you’ll find a button to return here and start building.

Eden AI information and set up

LEARN MORE

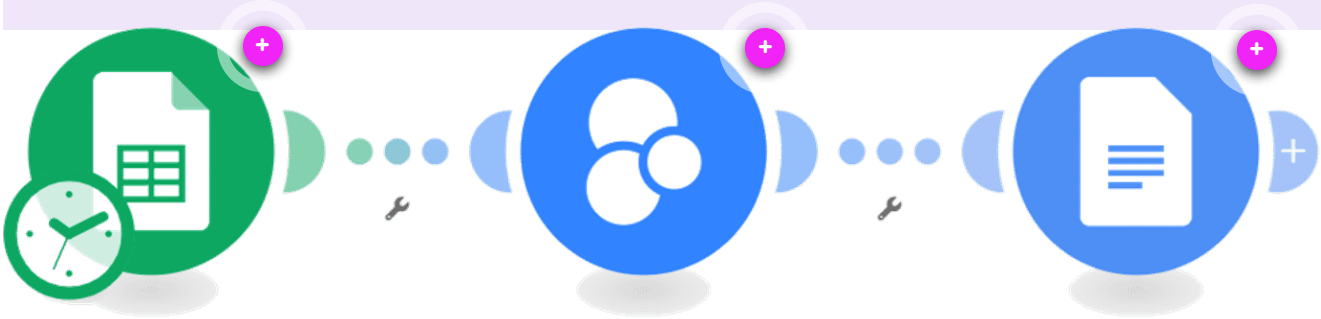
Continue to 3.11.1: Scenario overview

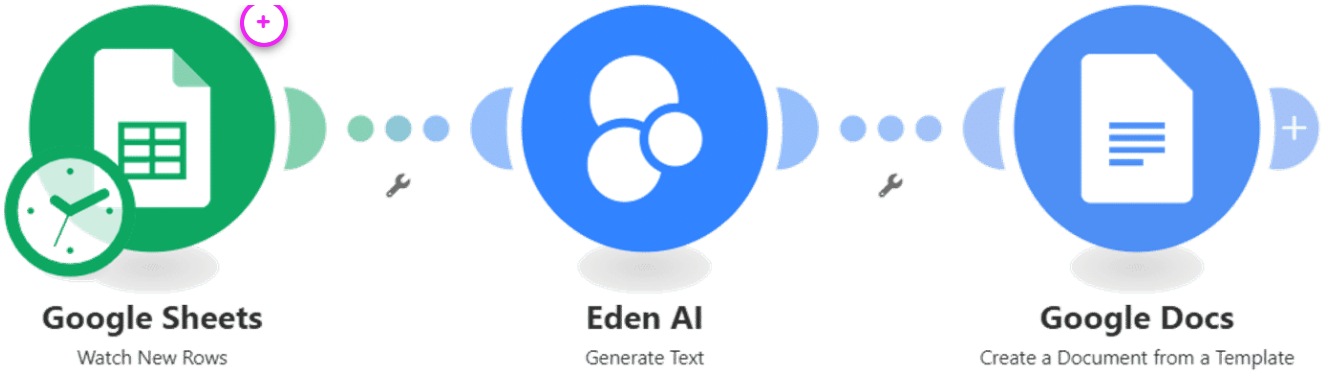
3.11.1 Scenario overview

Now that you have all the knowledge you need and your Eden AI account, let's have a closer look at the scenario you will build.

Here's an overview of the scenario.

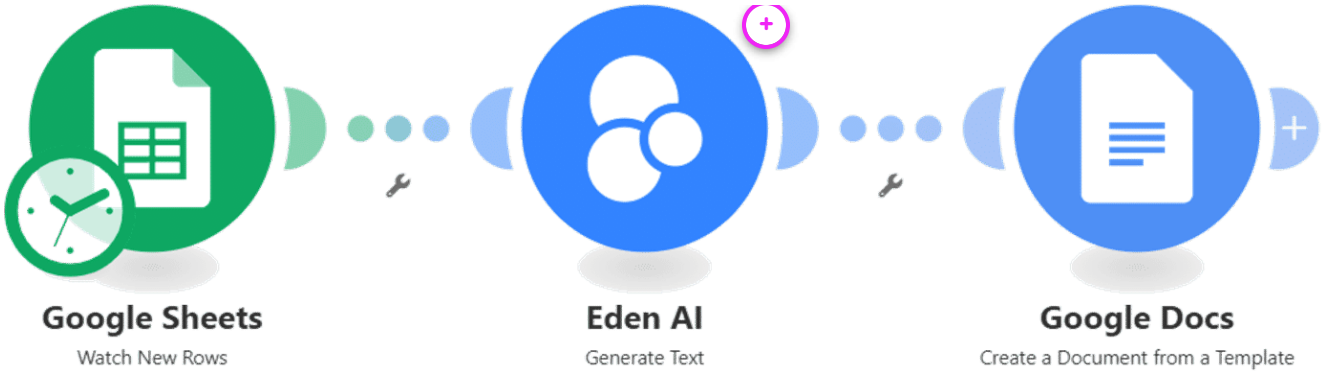
Click each + to learn more.





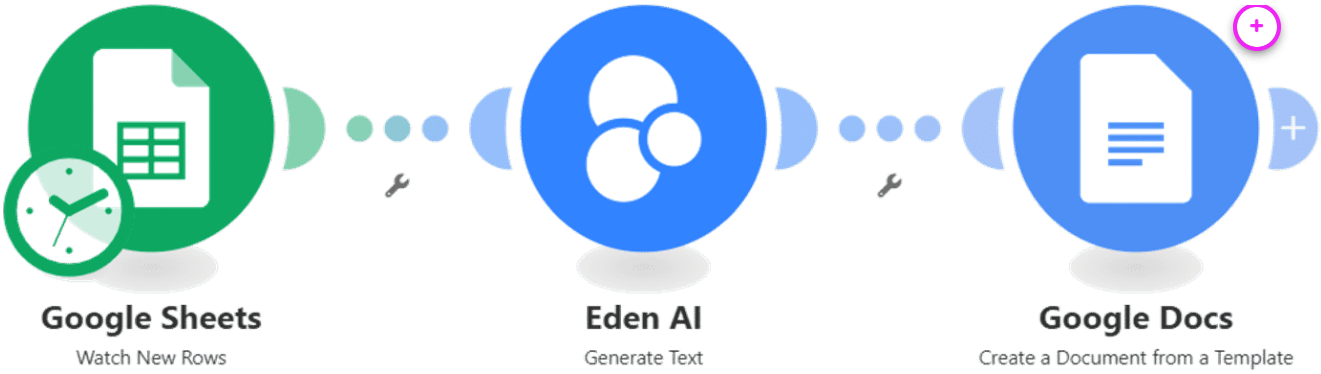
Google Sheets> Watch New Rows

Check the Google Sheet for new rows containing the information to generate the blog post.



Eden AI> Generate Text

Using a prompt and the information stored in the Google Sheet, generate the blog post.



Google Docs> Create a Document from a Template

Create a Google Doc containing the text of the blog post starting from a template. The document will contain the title of the blog and the blog post. The file name would be the same as the blog title.

If you don't feel like building this use case, no problem! You can go back to the Use case menu and pick another one instead.

Click the button below.

Go to the Use case menu

TAKE ME BACK!

Continue to 3.11.2: Use case preparation

3.11.2 Use case preparation

For this use case, you need two things: the Google Sheet with the blog post topics and the Google Docs template that the scenario will use to create the documents with the AI-generated blog posts.

Start by ***copying the text below***, it is the information that you will paste in the spreadsheet.

Title	Summary

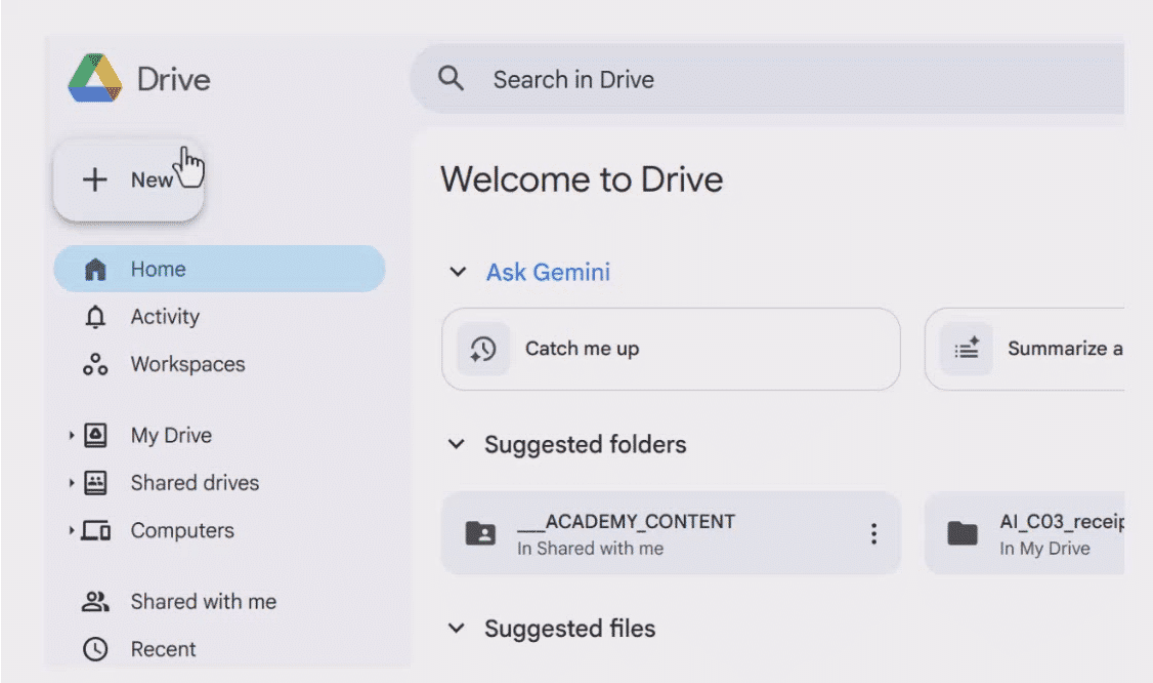
Poor Pluto	Pluto is no longer a planet
Put a ring on it	Saturn has rings made of ice and rock

Let's keep preparing everything you need for the use case.

Work through each stage before you continue.

Step 1

Google Sheet



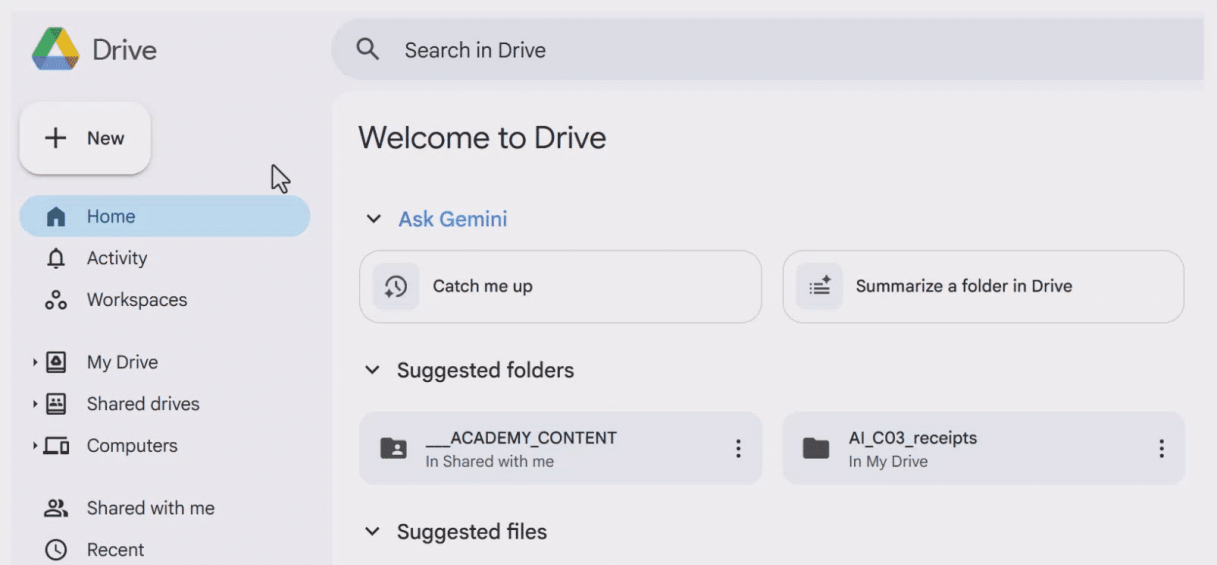
The first thing you need to do is to create a **Google Sheet** that stores the topics of your next blog posts.

Go to **Google Drive**, create a new **Google Sheet** and name it **AI_U03_UC3_blog**.

Paste the content you just copied it into the sheet.

Step 2

Google Docs template



Next, you need to create the template that your scenario will use to create a document for saving the blog content it generates.

In **Google Drive** create a new **Google Doc** and call it **AI_U03_UC3_template**.

Select and copy this text placeholder:

{{title}}

{{content}}

Paste it in the document and you're ready to start building the scenario.

Continue to 3.11.3: Build it

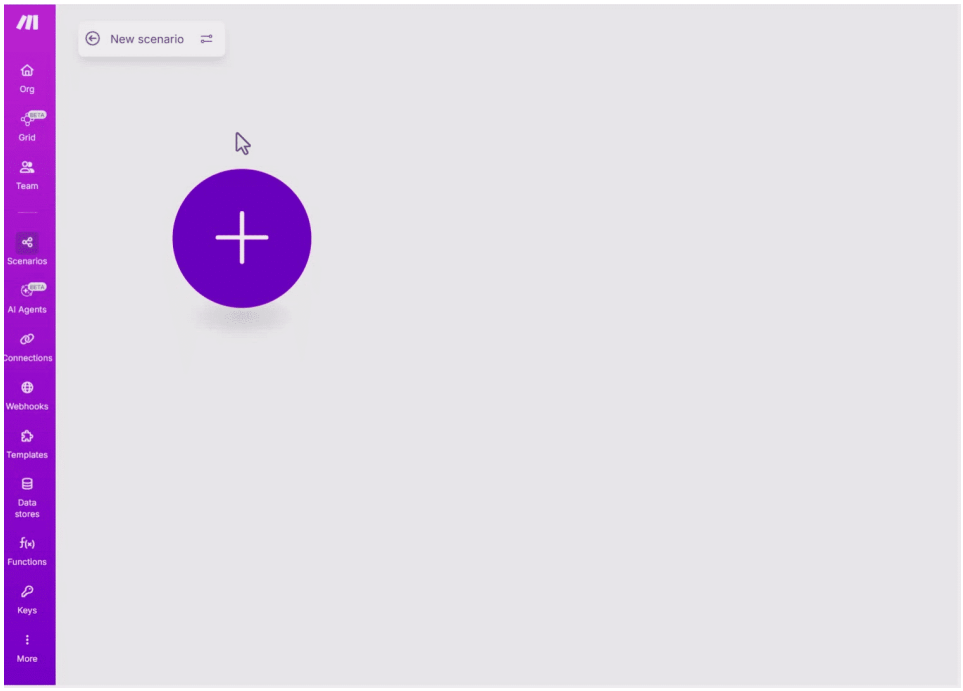
3.11.3 Build it

Let's build the scenario.

Work through each stage before you continue.

Step 1

Google Sheets> Watch New Rows



1 Create the scenario

In Make, create a new scenario and call it **AI_C03_U03_UC3_EdenAI**.

2 Add the module

To detect any new row in your spreadsheet, add a **Google Sheets> Watch New Rows** module.

3 Create the connection

Click **Create a connection** to connect it to your **Google** account if it is not automatically set up. Name the connection **Make_AI**, then click **Sign in with Google** and follow the steps.

4 Set up

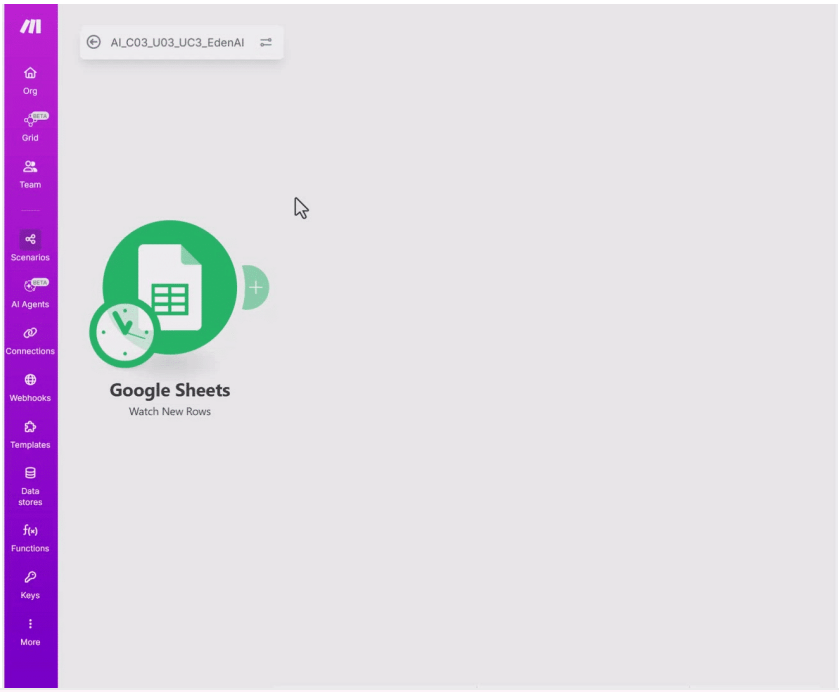
Under **Spreadsheet ID**, select the **Google Sheet** you have created earlier (**AI_U03_UC3_blog**), and under **Sheet Name**, choose **Sheet 1**.

5 Save

Click **Save** to save the module. When prompted to **Choose where to start**, select **All** and click **Save**.

Step 2

Eden AI> Generate Text



Next, you want to generate the text according to the content in the Google Sheet.

1 Add the module

Add the **Eden AI> Generate Text** module.

2 Create the connection

If you haven't already set up a connection with Eden AI, click **Create a connection** to link it to the Eden AI account you created earlier. Name the connection, **Make_AI_EdenAI**.

Next, go to your Eden AI account and copy the API Key. Paste it into the **API Key** field and click **Save**.

3 Set up the module

In the **Text** field copy and paste this prompt to generate the blog post:

> From this title: "[MAP TITLE]" and this summary: "[MAP SUMMARY]", generate a short text of maximum 200 words that covers the subject. The tone has to be neutral, easy to read for readers that don't know the subject.

Remove the **[MAP TITLE]** instruction (including the parentheses **[]**, but not the quotations **""**) and map the Title item from the **Google Sheets** module.

Repeat the process for **[MAP SUMMARY]**, mapping the Summary item from the **Google Sheets** module.

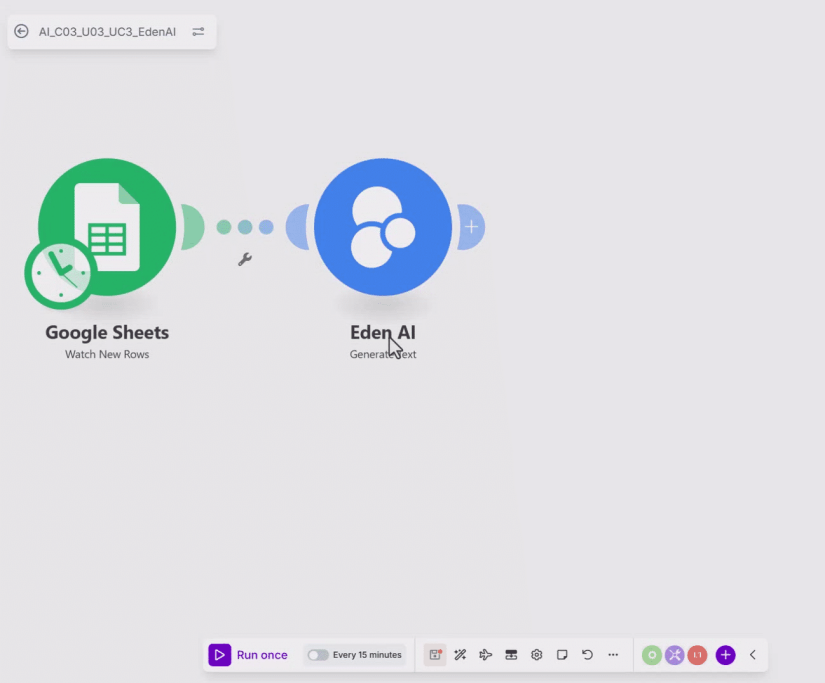
Select **OpenAI** as the provider, because the Make Academy team found it generates the most accurate text during testing.

4 Save

Click **Save** to save the module.

Step 3

Google Docs> Create a Document from a Template



1 Add the module

To save the generated text into the template, add a **Google Docs> Create a Document from a Template** module.

2 Set up the connection

For the **Connection**, select the **Make_AI** connection you created earlier from the dropdown menu. Click **Continue** to update the connection permission and follow the instructions.

3 Set up

Under **Document ID**, select the Google Doc template you created earlier (**AI_U03_UC3_template**).

In the **Value** section you can map the information in the placeholders.

Under **Title** map the Title item from the **Google Sheets** module, while for the **Content**, map the Result: Generated text item from the **Eden AI** module.

For the **Title** of the document that represents the file name, map again the Title item from the **Google Sheet** module.

Under **New Document's Location**, choose where in **Google Drive** you want to save the documents.

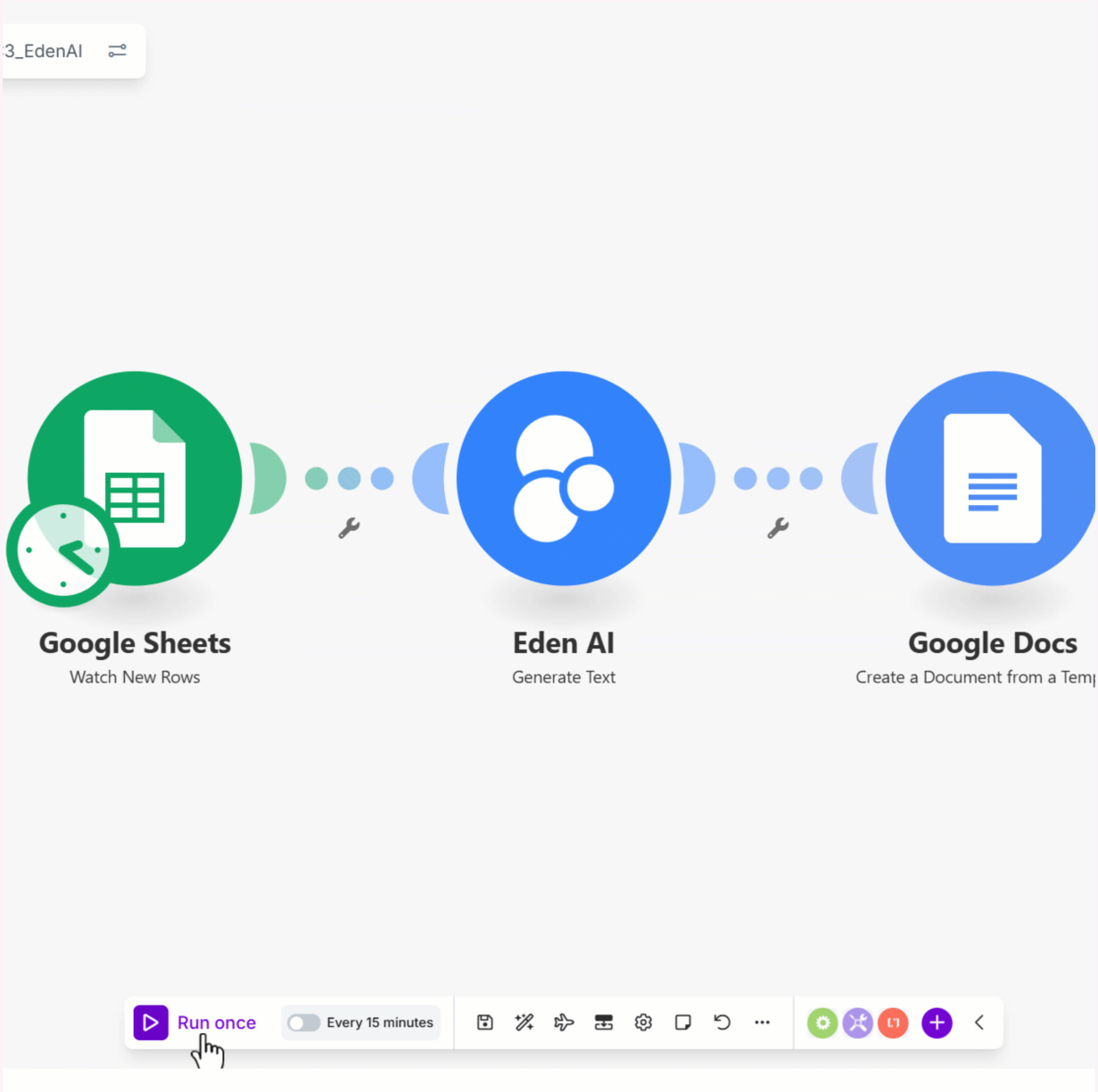
4 Save

Click **Save** to save the module and save your scenario as well.

Continue to 3.11.4: Run it

3.11.4 Run it

Time to test the use case.



After saving your scenario, click **Run once**. When your scenario finishes processing, go and have a look at the folder where you saved your documents.

Open the documents and read the blog post that Eden AI generated for you!

Well done! You now know how to use AI to create content and save it in documents using a template.

Your job is done here. Go back to the **Use case menu** and pick another use case. ***Click the button below.***

Go to the Use case menu

TAKE ME BACK!

Eden AI

Eden AI is a platform that groups services from different AI providers and has ready to use modules for specific tasks.

An AI provider is a company or service that offers AI tools or technology for specific tasks, like for example:

- **OpenAI** - Offers tools like ChatGPT and GPT models for natural language processing, text generation, and summarization.
- **Amazon Web Services (AWS)** - Provides AI services for image and video analysis, text-to-speech and text analysis.

Within Eden AI you can **choose one or more AI providers for your tasks**.

To find the best provider, you can select all of them, run the scenario, and pick the one with the best results, the most accurate response in this case.

This will take time and use your Eden AI credits. But don't worry, the people who created this course have already done this for you, and in the instructions of every use case, you will find the best AI provider for the task in terms of quality vs price. You can use that one or try others if you like.

Note that the different providers may have different behaviors for a specific task, depending on their model settings and data they use. You may get different results with varying accuracy when using them.

Normally, to perform specific tasks using AI, you have to figure out **which prompts to write and how to set the right settings for each task**. This can take time and practice.

Eden AI makes it easier by providing ready-to-use modules for specific tasks. These modules handle the prompts and settings for you, you just give them your data, and they process it. For example, **Eden AI** offers a **Sentiment Analysis** module that handles everything for you.

You can get an **Eden AI account with free Eden AI credits via Make**. By signing up with an affiliated link, you'll receive enough Eden AI credits to complete all the use cases in this unit. Let's set it up!

Get an Eden AI account

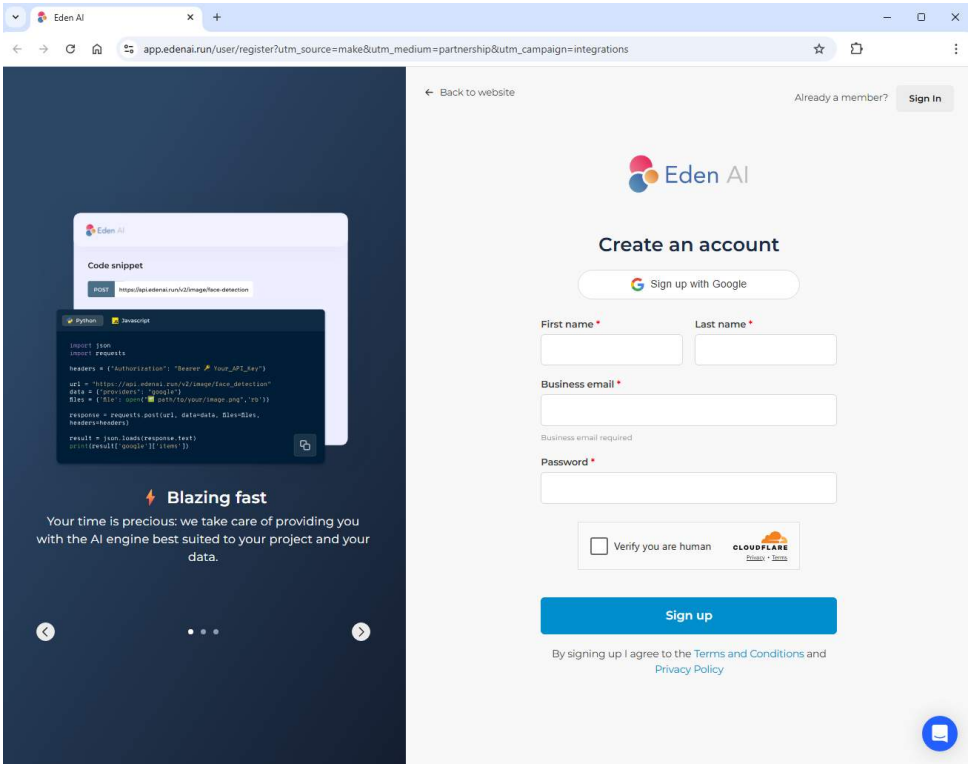
Get an Eden AI account

To be able to use Eden AI you need to sign up and create and account.

Let's create the account.

Step 1

Referral link



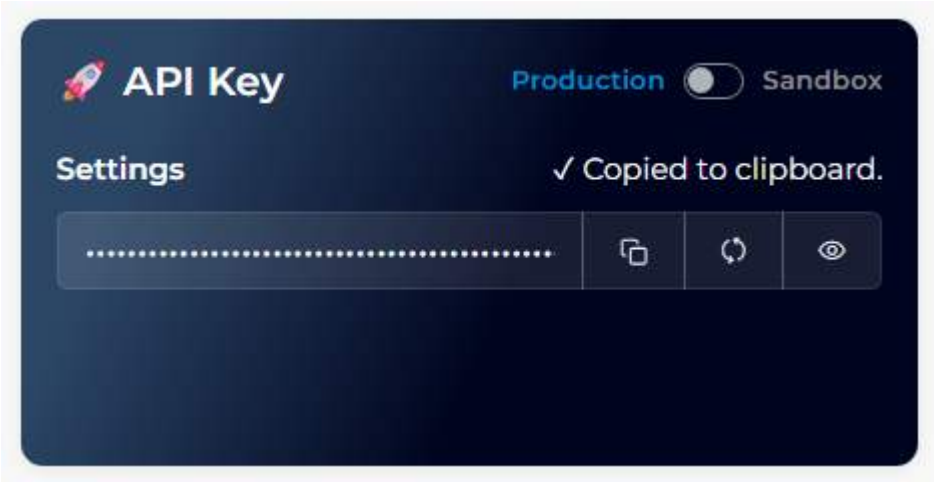
To create an Eden AI account and receive free credits, [use this referral link](#).

> **Note:** the free credits are only available to new users, existing accounts aren't eligible.

Choose **any of the options you prefer to sign up**. Fill in all the required information and create your account.

Step 2

Get the API Key



To be able to use the Eden AI services in Make, **you need an API key**. An API key is like a password that will allow you to connect to the Eden AI API and use its services.

Let's see where to find it.

In the homepage, notice the **API key** block with the masked API key and a **copy** button.

This is the information you will need when building your scenario. You can copy it now or come back to this page later.

Awesome! Your Eden AI account is ready, and you can now start building your scenarios. Click the

button below to go back to your use case.

Use case 1

Customer service scenario

3.5 USE CASE 1

Use case 2

Extract data from pictures

3.8 USE CASE 2

Use case 3

Social media post creation

3.11 USE CASE 3

Unit complete!

Have you built all the use cases you wanted? You still have time to go back and create a quick scenario to see the power of AI combined with Make.

If you are satisfied with what you've done, congrats!

You have completed the course.

From here, you can keep exploring AI in Make by learning about AI agents, or jump straight into adding AI to your own scenarios.

make | academy

Mark this task complete to continue to the next unit.