

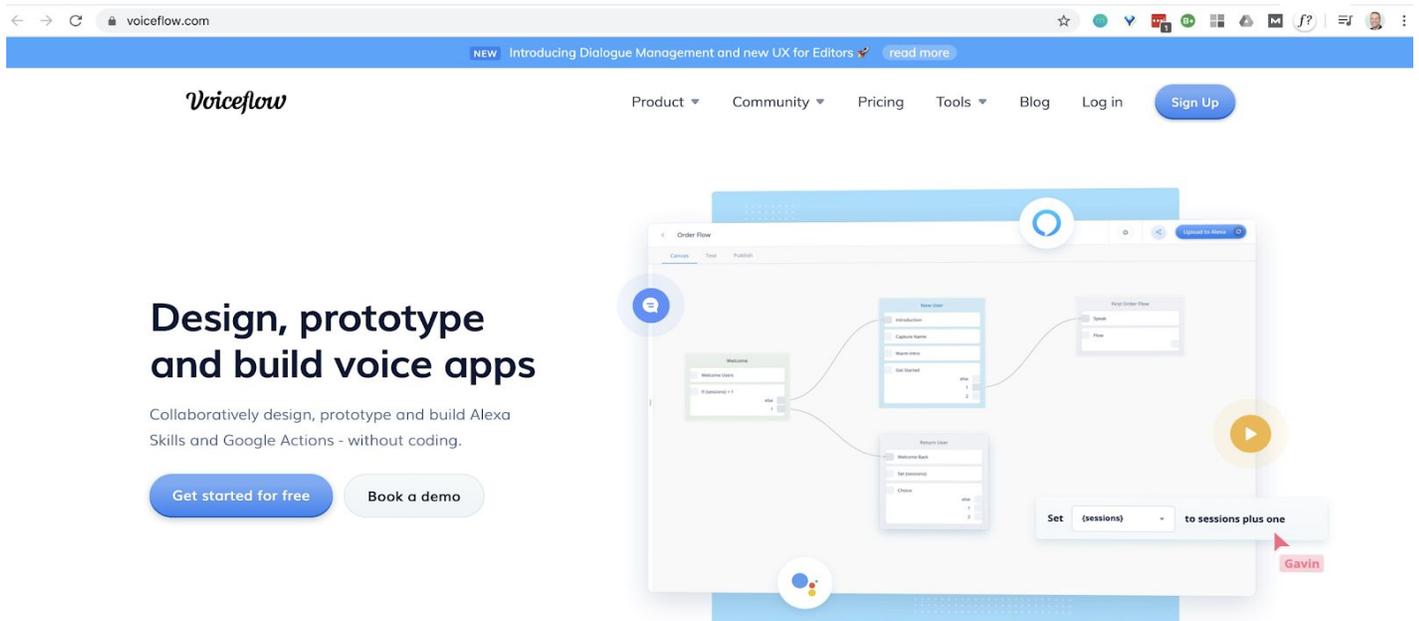
Build Your Own Voice App with Voiceflow

For Alexa

Let's build a voice app! Today we will:

- Program speak blocks
- Capture input
- Enter data (quotes)
- Configure choice
- Error handling

Signing up for Voiceflow:



The screenshot shows the Voiceflow website. The navigation bar includes links for Product, Community, Pricing, Tools, Blog, Log in, and a Sign Up button. The main content area features the heading "Design, prototype and build voice apps" and a sub-heading "Collaboratively design, prototype and build Alexa Skills and Google Actions - without coding." Below this are two buttons: "Get started for free" and "Book a demo". A large image shows a preview of the voice app interface, which includes a "Welcome User" screen, a "New User" form, and a "Return User" screen. A "Play" button is visible in the bottom right corner of the preview. A small red arrow points to a "Gavin" label in the bottom right corner of the preview.

1. Go to www.voiceflow.com
2. Start by clicking the blue sign up button on the upper right corner indicated by the arrow.
3. Proceed with the sign-up process. Enter your full name, email address, password and press “Create Account”. If you already have an account, click on “Have an account?” and sign in.
4. This is the welcome screen as a new user to Voiceflow. Press “Continue”.

Hi, Anthony

You just joined the worlds biggest community of designers and developers building for voice. We have a few questions to personalize your experience!

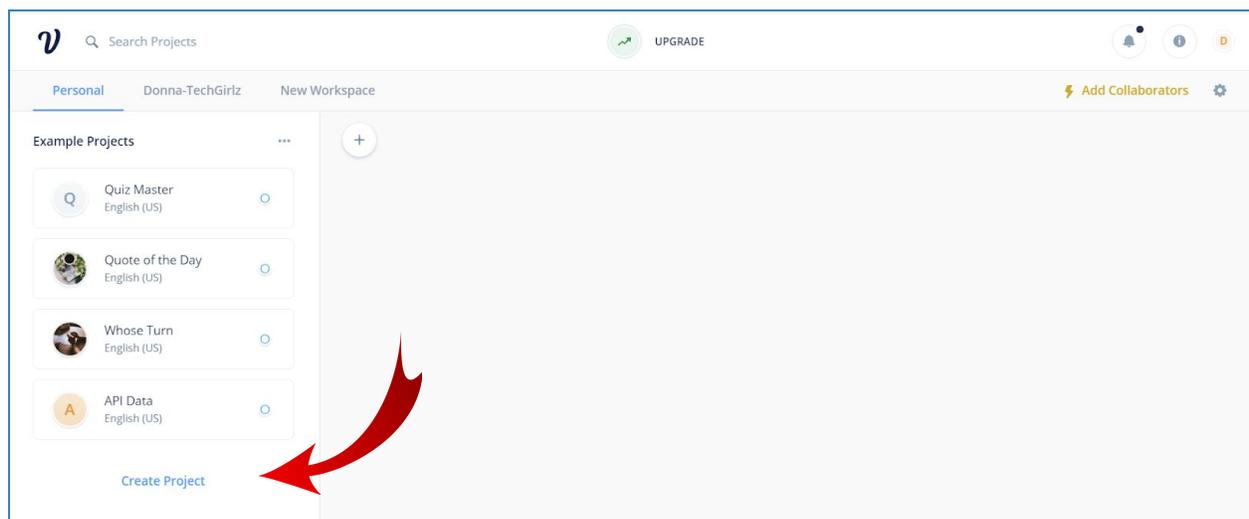
- Voiceflow team ❤️

Continue

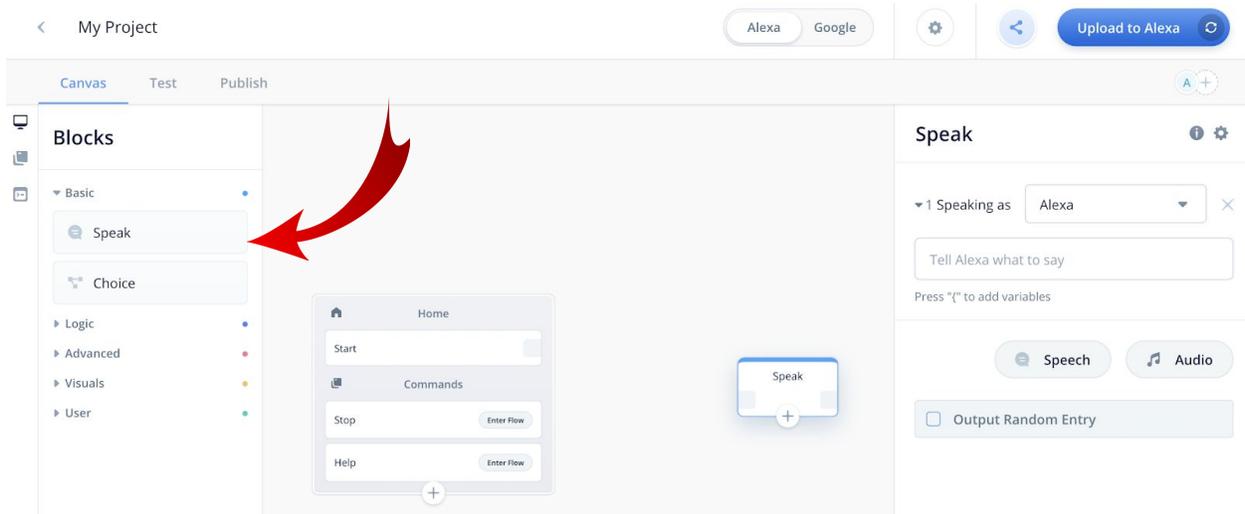
5. Choose “Just Me”, then “Continue”.
6. Click “Design & Prototype”, then “Continue”.
7. When asked about your experience, choose none, then press “Complete”
8. A welcome and an invitation to join the Voiceflow community will pop up. Close this window. There is no need to accept the invitation for the Voiceflow Community. You can join this community later if you wish.

Let's create:

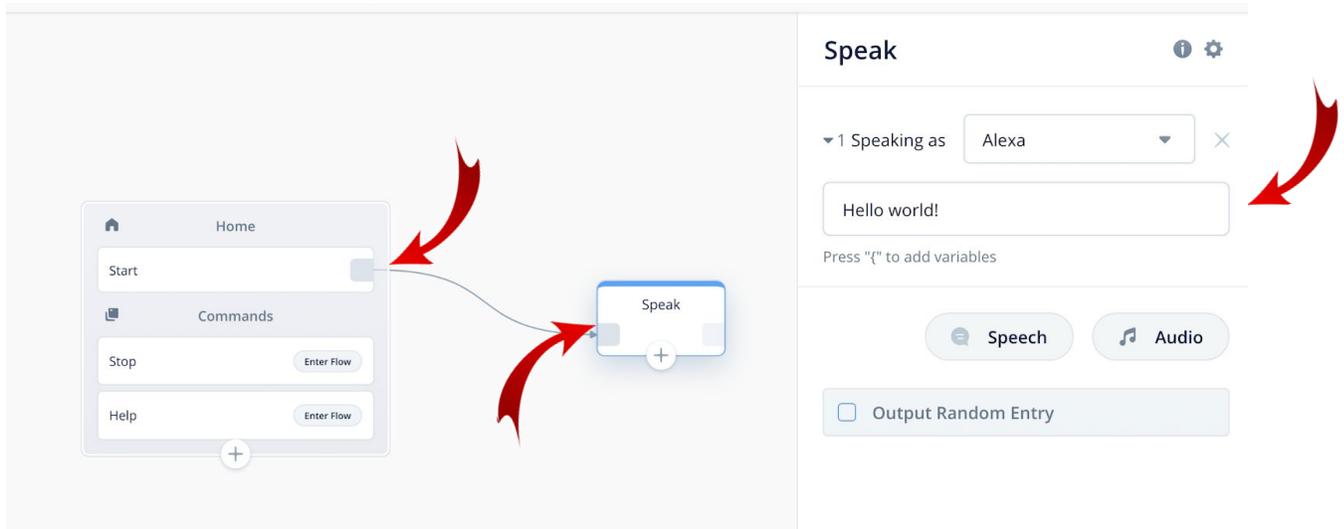
1. Click “Create Project” located at the bottom left side of the page



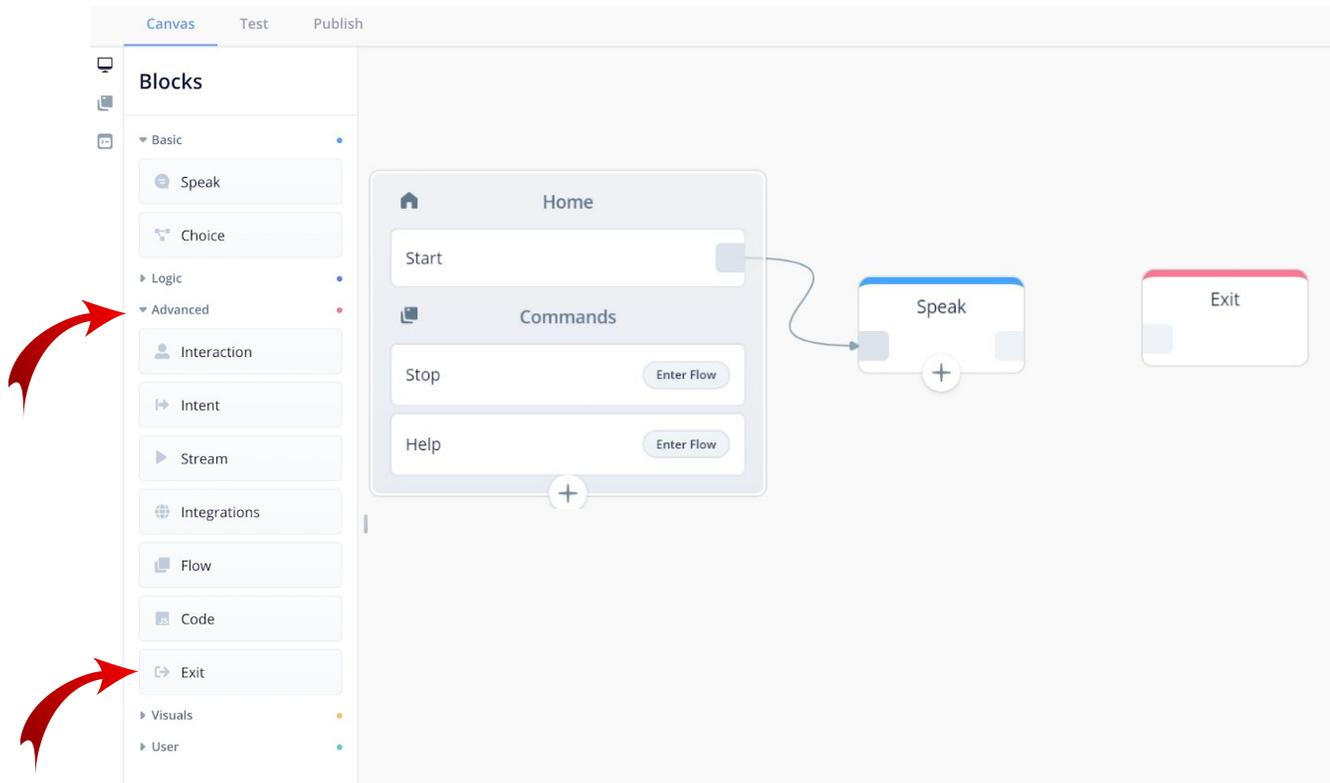
2. Click “Start a new project”
3. Note the menu of programming blocks on the left. Basic, Logic, Advanced, Visuals, User. In the Basic menu, drag a Speak block to the right of the large Home block. In this workshop we will program for Alexa:



4. Click on gray block connector on the edge of the start block, drag and connect to the speak block. If you don't see the square right away, hover your mouse on the edge of the block until you see it. Connect start to the speak block. Click on the speak block then type Hello World under System Says. Note that you can change the speaking as voice. The default “speaking as” voice is Alexa, but there are several other options.



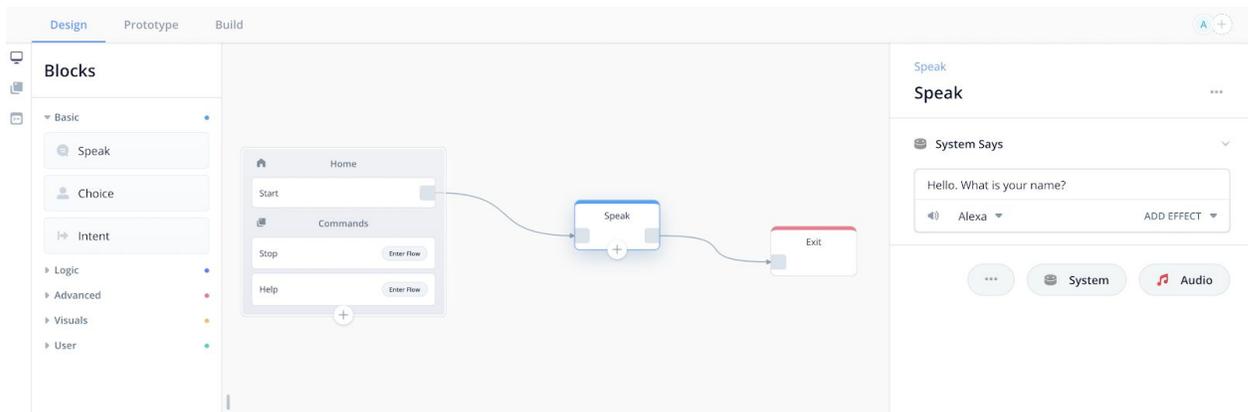
5. Find the Exit block in the Advanced block menu on the left. Drag the exit block onto your canvas to the right of the Speak block. Connect the Speak block to the exit block as indicated by the red arrows between the Speak and Exit blocks



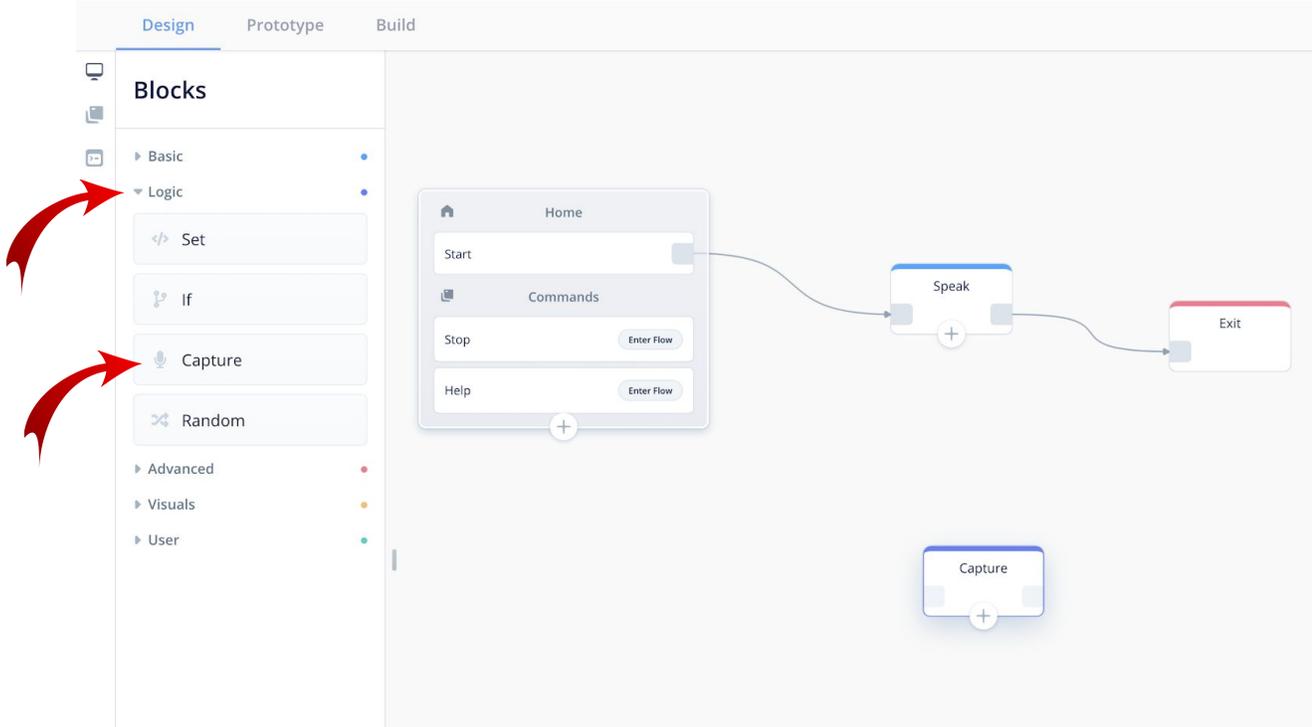
6. Click on **Prototype** at the top.

- a. On bottom right click “start test”. Do you hear something? Did you hear Hello World?
- b. Click on Design or Return to canvas to get back to the canvas. Click on the speak block and change the voice. You can then go to prototype and click “Start Test” to hear the new voice selection. You can experiment with different voices for your app.
- c. To repeat the test, click “Reset Test” You will need to return to the canvas if you would like to choose another voice.
- d. To test it , click on “Prototype” , then Start Test.

7. Go back to Design, or Return to Canvas. Click on the Speak block, change “Hello World” in System Says to “Hello. What is your name?”



8. Capture block listens to what the user says and stores the information
 - a. Now, under the Logic menu find the Capture block and drag it onto your canvas.



What about variables?

It helps to use comparisons or visuals when explaining what a variable is. Here are some ideas:

We often use variables in different ways. The name may stay the same, but the value may change throughout the program.

For example, when watching a game we talk about the “score” Score is a variable that holds a value that changes. As you make more baskets, goals, etc, the score goes up.

The score may change, the number of lives you have left in a game may change, the direction someone wants to go in a game may change, etc.

Sometimes we may want to calculate a value at the beginning and save it and use throughout the program. For example, a program may ask a user to input their name at the beginning of the program and then throughout the program use their name to address them by their name.

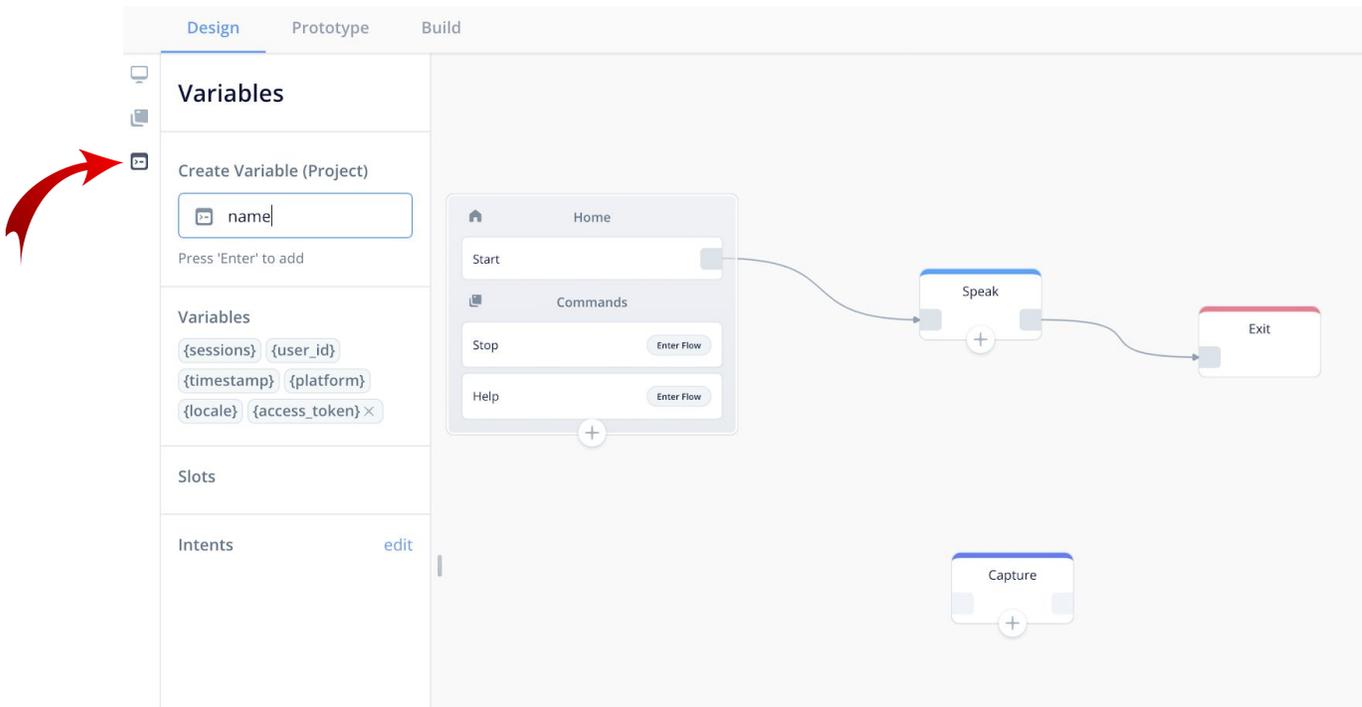
We may figure out what your monthly fee is for Netflix (according to the program you chose) and then use the amount later to report back what you owe.

Another way to explain it, is like a locker at school. It contains different books at different parts of the day. But you always know you can go to your locker to get the books.

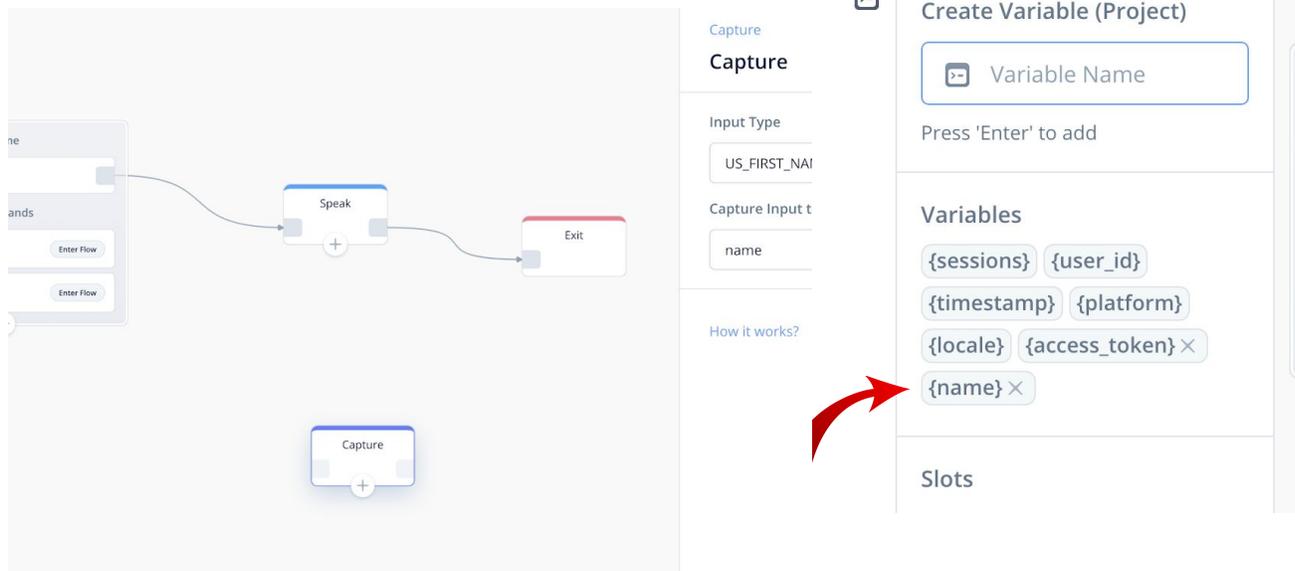
Let's set up variables:

We will be setting up a variable for name input.

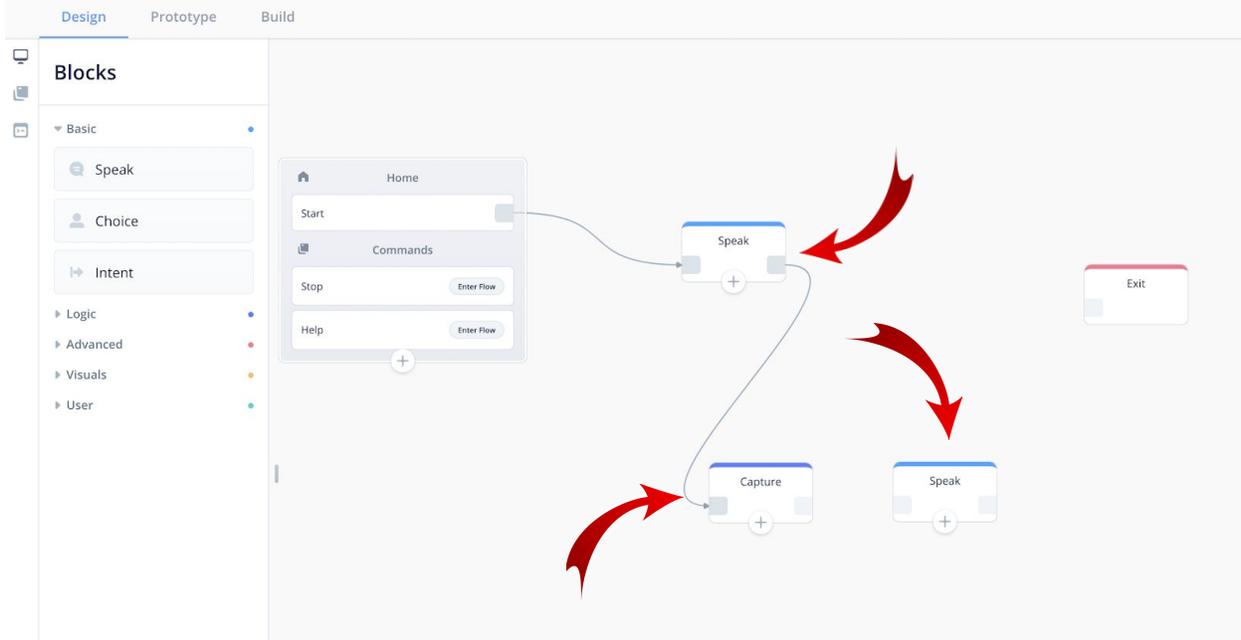
1. On left find the icon for variable as shown by the red arrow.
2. In Create Variable Project type in name. Press enter to add it to the variables list.
3. Make sure {Name} appears in the Variables list.



4. Use a variable to capture your name. **Do you see {Name} in the variables list?**
5. In the Capture block, For Input Type select: US_FIRST_NAME from the drop down menu options. Capture input to-select: select name from the drop down menu options



- You will need to connect the Capture block. However, you must first delete connection between Speak and Exit blocks.
Note that the connecting line will blink, then click on the trash can icon shown by the red arrow.
- Connect Speak block to Capture block. Then add a new Speak block to your canvas.



- Connect the output of Capture to the new Speak block.
- Rename the new Speak block to Welcome.
- Configure block as follows: System says Hello {name}. Welcome to the quote of the day.
- Then connect the Welcome block to the Exit block.

Give it a try:

Go to prototype, then “Start Test”. When you are asked for your name, press the spacebar and answer. You may also type in your name in the User says field at the bottom right.

Time for a break



What is a Database?

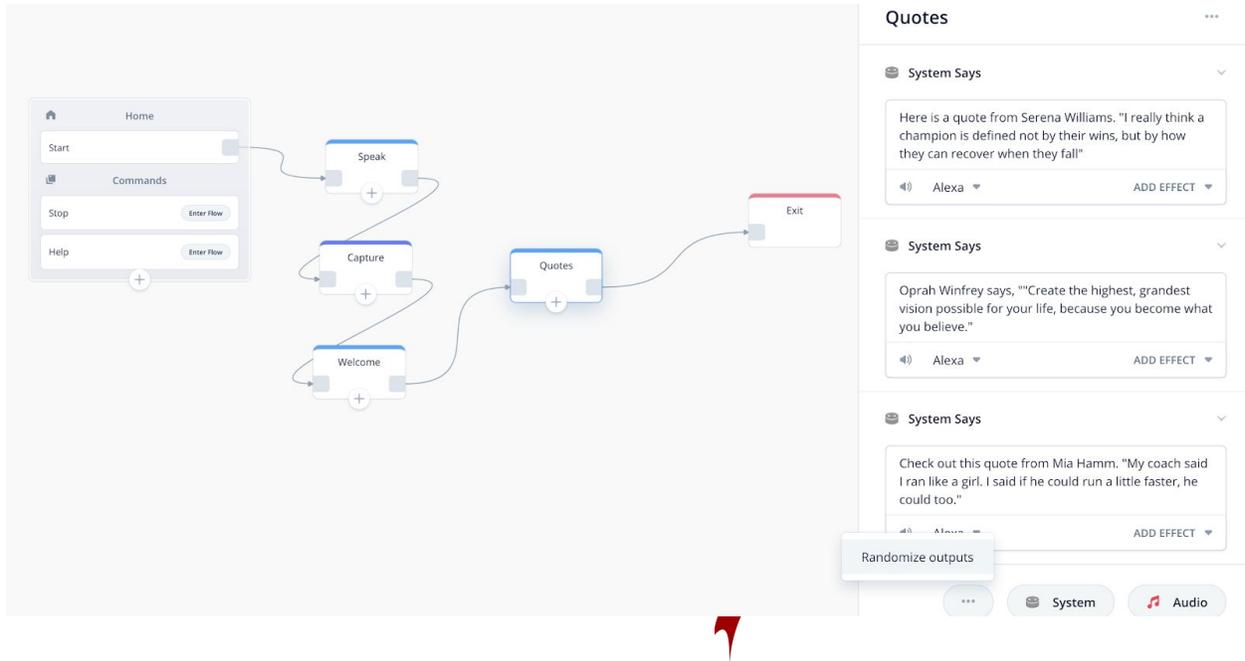
It is a place to store data (information) that computers use. Data is stored in a format that computers can readily access and retrieve the data. Databases can be small like the one we will create today or they can be very large. For example, Google maps is based on a database that contains an enormous amount of information (data).

The Quotes collection is our database. The quotes will be entered into a Speak block.

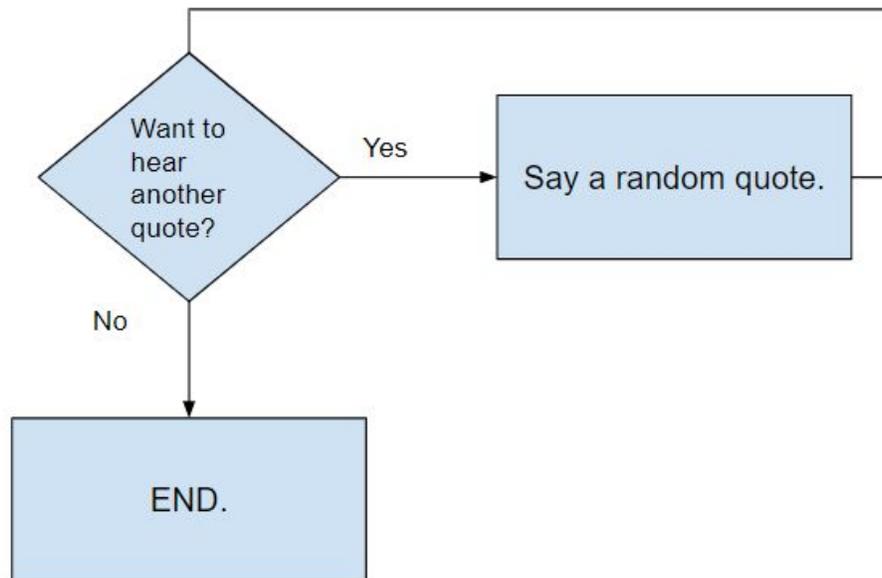
1. Add a Speak block to your canvas and rename it to Quotes.
2. Connect output of Welcome block to the input of Quotes.
3. Connect output of Quotes to Exit.

The image shows a flowchart on the left and a Speak block interface on the right. The flowchart starts with a 'Home' block containing 'Start', 'Commands', 'Stop', and 'Help' buttons. Arrows indicate a flow from 'Start' to 'Speak', 'Speak' to 'Capture', 'Capture' to 'Welcome', and 'Welcome' to 'Quotes'. 'Quotes' is connected to 'Exit'. The Speak block interface on the right shows three 'System Says' blocks, each containing a quote from a famous person: Serena Williams, Oprah Winfrey, and Mia Hamm. The bottom of the interface has a 'System' button and an 'Audio' button.

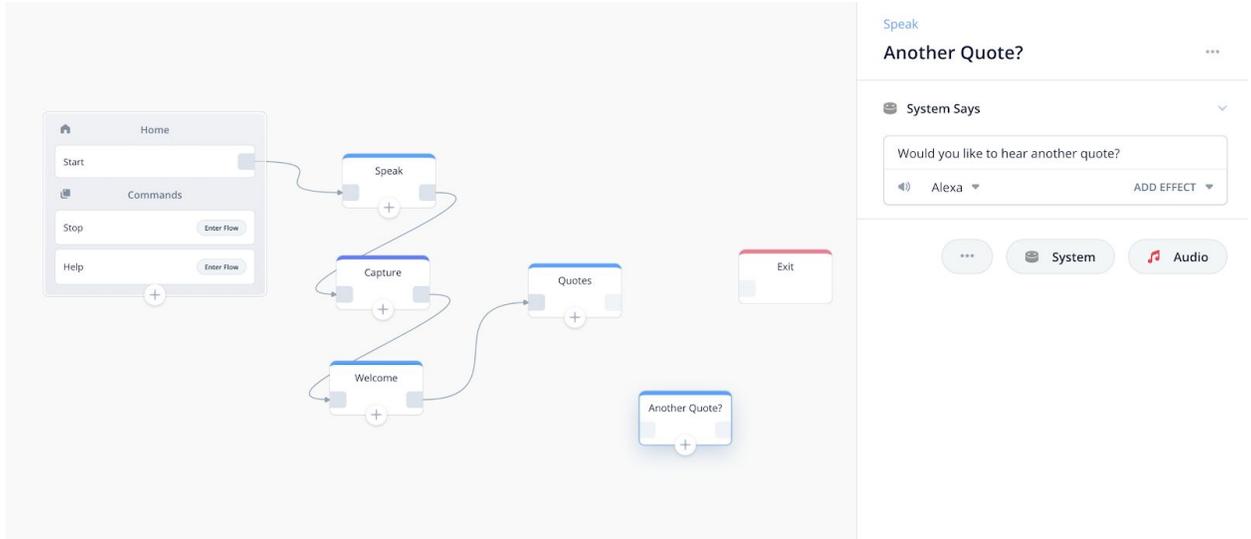
4. Choose a famous person that you admire.
5. Go to <https://www.azquotes.com/> and search your famous person's name to get a quote(s) from them.
6. Select the quote and copy it.
7. Under System Says, paste in your first quote. To add another quote, click System button at the bottom of the speak block. Paste in your next quote and repeat until you have at least 3 quotes.
8. To your quotes add "Here's a Quote from (add name of quoted person)" and "(quoted person) Said"
9. Now, randomize your quotes by clicking on the 3 dots at the bottom of the Quotes speak block next to the system button.



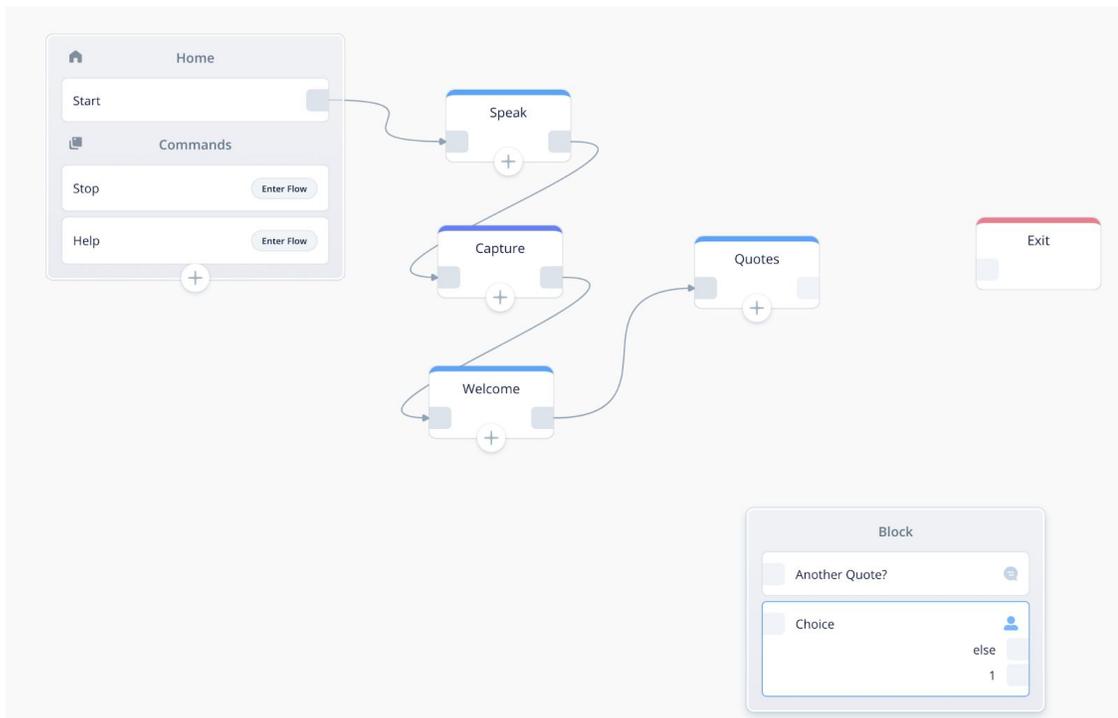
10. In the program you are building, Alexa will ask a question, If you answer “yes”, you will get one response, but if you answer “no” you will get a different response. This is best demonstrated in a flow chart.



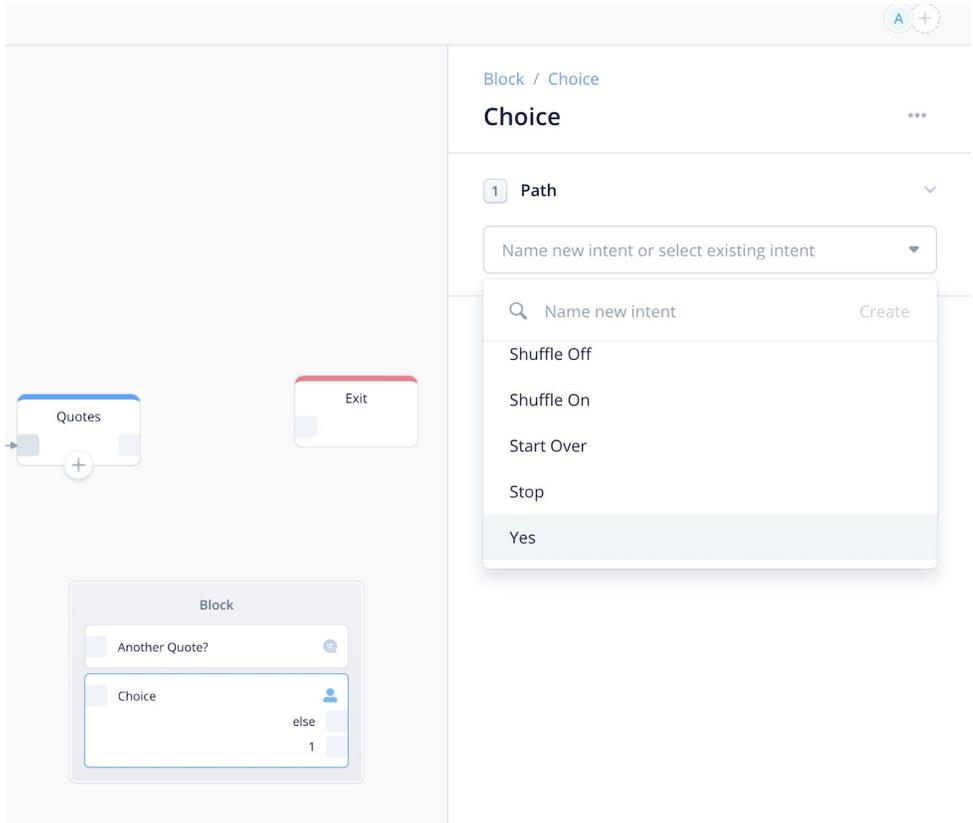
11. Add a new Speak block to your canvas and name it “Another Quote?”.



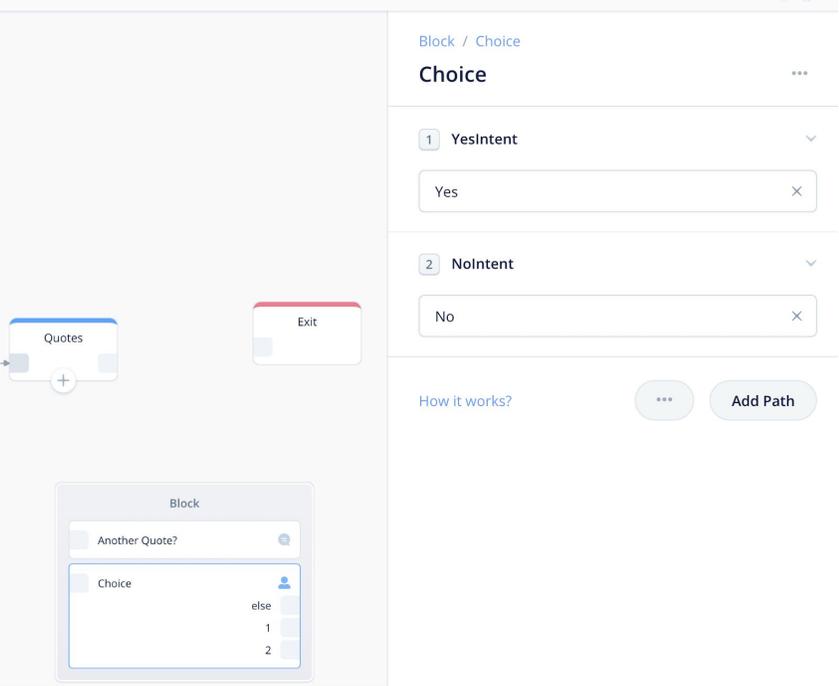
12. Add Choice block to your canvas. You can find it in the Basic block menu.
13. Click on the “Another Quote?” block and drag it into the choice block. When both blocks change color, let go. You will see that both blocks are now combined.
14. The conditional box will now look like this (below). It will now contain the question and options to configure choice. Rename this combined block Conditional Block.



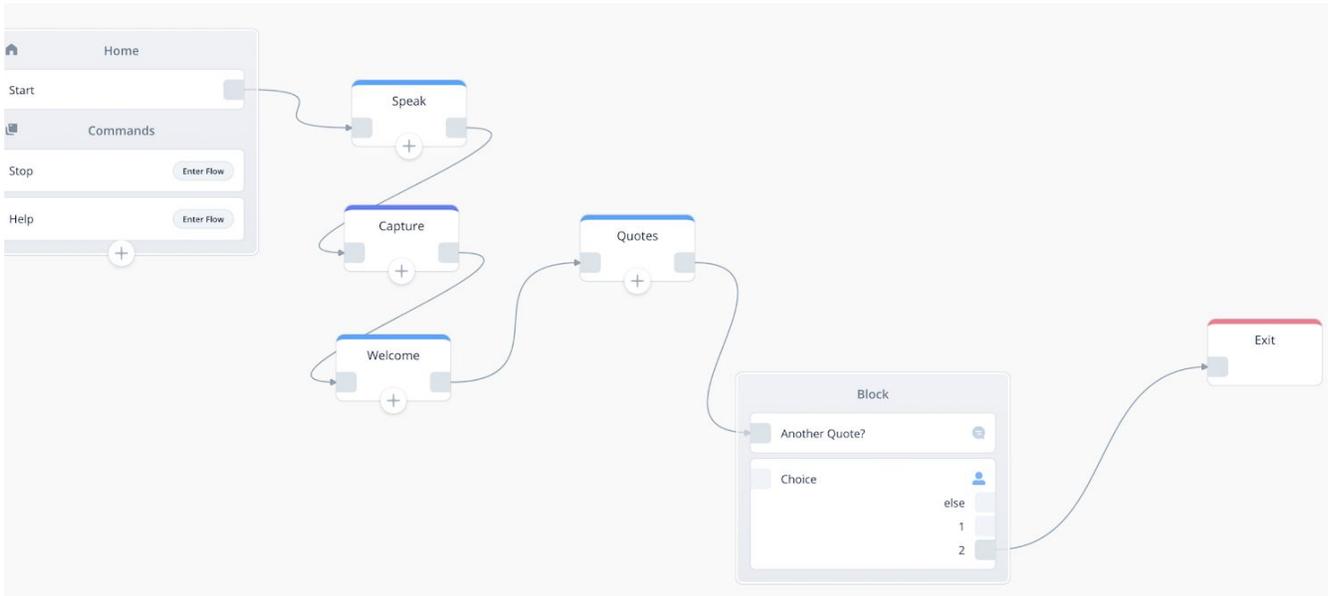
15. In the choice portion of the Conditional block, click path 1. You can name a new intent or select existing intent. Yes already exists, so choose Yes from the drop down options. Connect path 1 to Quotes (input)



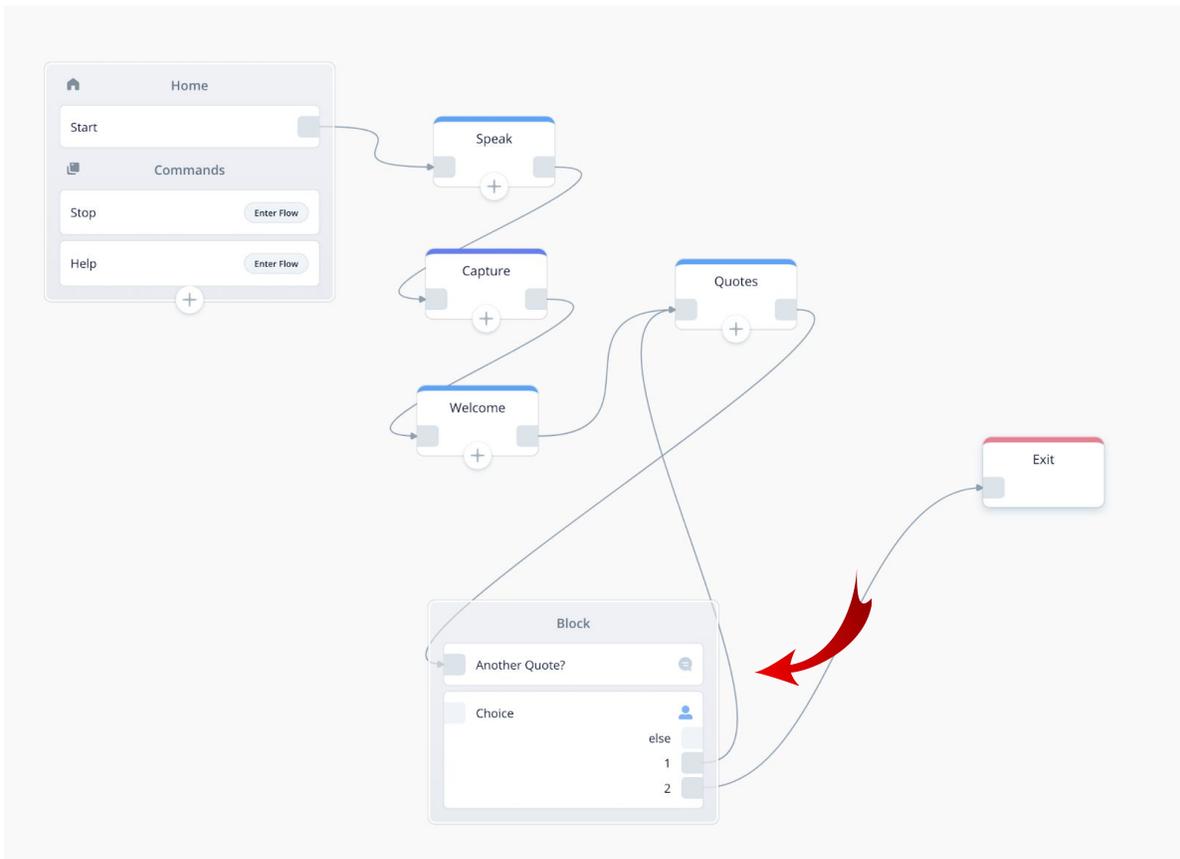
16. Add path 2 by clicking add a path. You can name new intent or or select existing intent. No already exists, so choose No from the drop down options.



17. Connect path 2 to input of exit block. If you say no, to the question Another Quote, the program will end.



18. The output of quotes connects to the input of "Another Quote?"



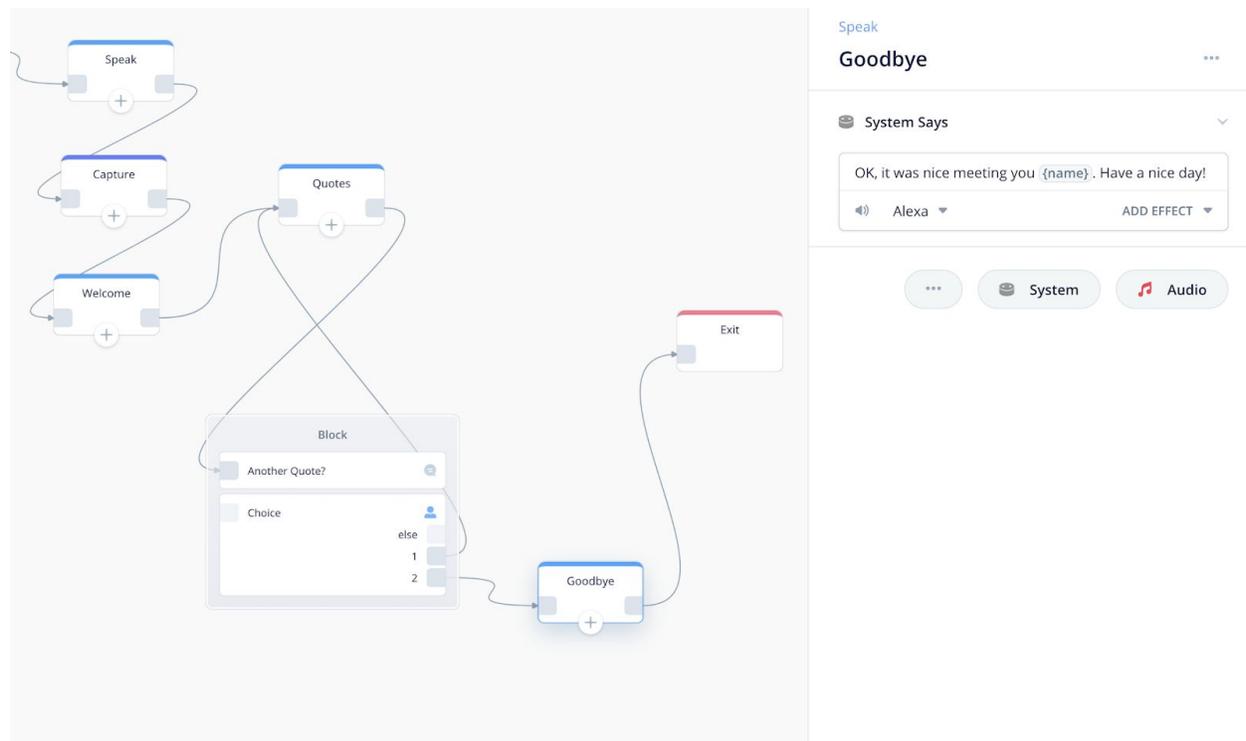
Give it a try:

Go to prototype, then “Start Test”.

What is the app doing? The system asks your name, welcomes you to the quote of the day and tells you a random quote. The system asks if you would like to hear another quote. If yes, the system says another quote. If no, the program ends.

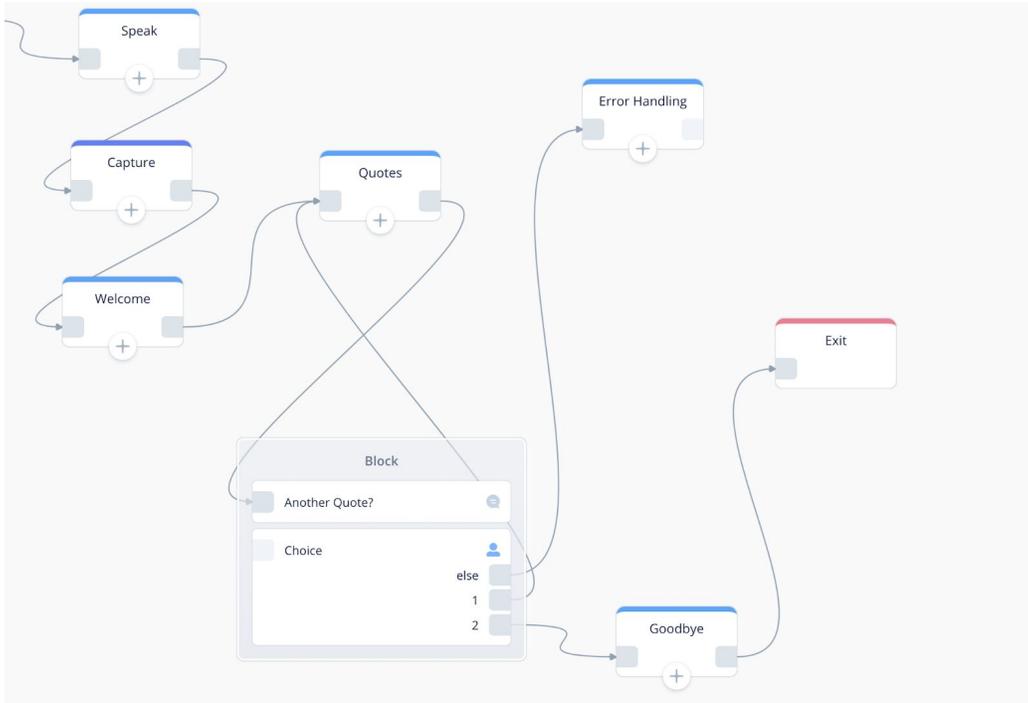
Configuring the end:

1. Add a new Speak block to the canvas. Name it Goodbye.
2. In Systems says type in “OK {name} it was nice meeting you have a good day.”
3. Delete connection between 2 and exit.
4. Connect the output of no to the input of the Goodbye block. Then connect the output of Goodbye block to the end block.

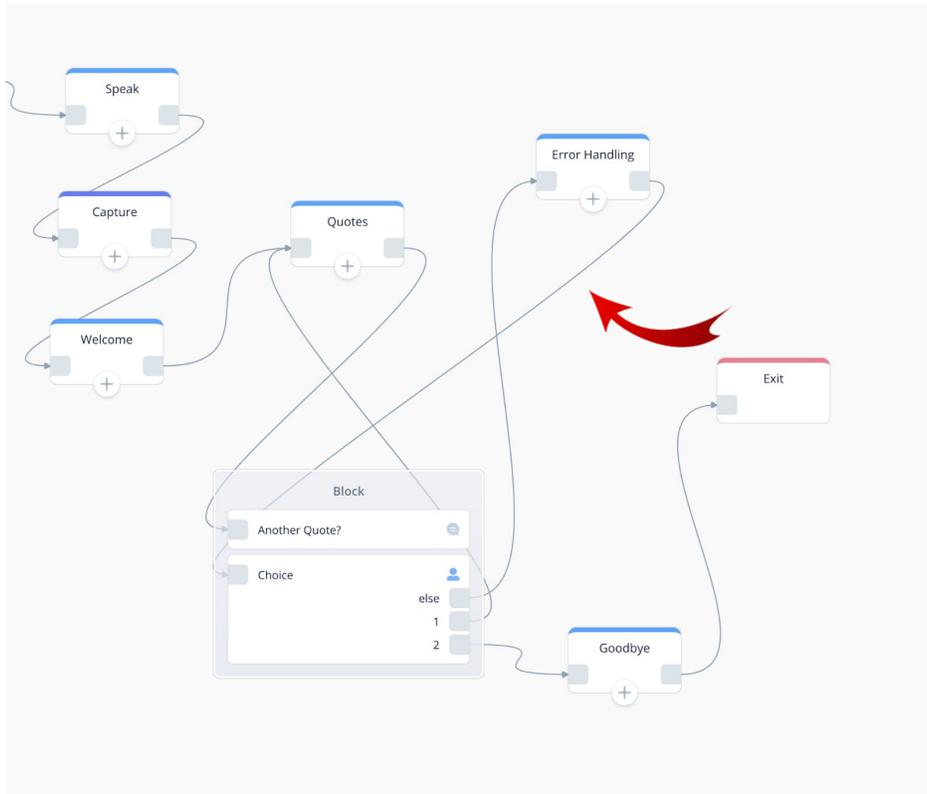


Error Handling:

1. Drag a speak onto the canvas, name it Error Handling.
2. Type this into system says “Sorry I didn't quite get that. Do you want to hear another quote? Please say yes or no.”
3. Connect the output of “else” to the input of the error handling box.



4. Connect the output of the error handling block to the input for choice in the Conditional block



Review your final results:

Go to prototype, then “Start Test”.

If a response is not yes or no, the system will say “Sorry I didn’t quite get that, please answer yes or no” The Voiceflow program can recognize some synonyms for yes or no. For example, “yeah” is not recognized as “yes” and the system will ask again, “would you like to hear another quote?”. If the response is “sure” the system recognizes that as “yes” and says another quote.

Additional resources:

- [Voiceflow Basics](#)-Voiceflow tutorial series
- [YouTube Voiceflow tutorials](#)-Voiceflow channel
- [TechGirly Create Your Own Voice App with Voiceflow Instructional Webinar](#)-describes eachstep of this workshop
- [Role Model Tip Sheet \(TechGirly\)](#) detailed questions to help you share your story and inspire girls in tech
- [Internet Safety Tips \(TechGirly\)](#) advice on how to use the Internet safely and protect yourself and your info

What do you want to build next?

