Table of Contents

Preamble

This software is free (as in “beer”) and provided as is. If you are ever charged for this demand your money back.

You may contact the author with suggestions and bug reports (or if you want the source code and laugh at it) at:

speech2keys@gmail.com

Introduction

Speech2Keys is a tool that lets you speak words and interpret them as keyboard events. This can be used to send a key (or a sequence of keys) to the active window (e.g. a game). You can also define responses that will be played over your speakers/headphones when this happens.

One set of words (“key phrases”) and their associated pressed key(s) is called a “command”. A set of commands is combined into a profile. You can have as many profiles as you like (e.g. one for each game), but only one profile can be active at a time.

Using headphones is highly recommended to prevent responses or in-game sounds from also triggering commands.

Prerequisites

You have to have .NET 4 (or higher) installed which can be gotten from Microsoft.

http://www.microsoft.com/en-US/download/details.aspx?id=17718

A microphone is required. Headphones are strongly suggested.

This software has been tested on Windows 7 (32 and 64 bit machines).

How to Create a Profile

Make sure your headphones are connected and your microphone is turned on and working.

- Launch the program by double clicking on the Speech2Keys.exe file.
- You will be presented with a quick launch screen with three options. Choose “Create a New Profile”.
- On the next screen you will be prompted to enter the profile name. Do so and click “Finish”.

You will be shown the main screen with some standard commands already created in the central list box (more on these later).

Hint. At this point you can already try out the voice recognition. Hit the “Start Voice Recognition” button and say the word “Alexa” into your microphone. Speech2Keys should respond with an audible “Listening” and the message “COMMAND RECOGNIZED: AIName” in the right text box.

Click once the command “AIName” in the central list box. In the text box on the right you will be shown information on the command. Note that it has no key assigned to it. This is intentional (more on these special commands later).

To end voice recognition hit the “Stop Voice Recognition” Button. You should always end voice recognition while editing commands.

Any changes you make while voice recognition is running will not take effect until you have stopped and restarted the voice recognition via the buttons!

Creating Your First Command (quick version)

- Click the huge “Add New Command” button (or just hit enter).
- You will be prompted to give your command a name (E.g. type in “Cannon”). This is *only* a name and *not* what you will say to trigger the command.
- On the next screen press the key you use in-game to fire the cannon (e.g. press “c”. You will see “c down” and “c up” appear in the list box). Hit the “Next” button.
- On the next screen enter what you want to say (e.g. “fire cannon” or “shoot”). If you want to, just hit enter and keep adding key phrases. The command will be triggered on any one of them. *You must define at least one key phrase.* Hit the “Next” button when you're done.
- Skip the responses screen for now by hitting the “Finish” button. Responses are optional.

Voilà. That's it. The “Cannon” command should now have appeared in the central list box. Clicking it once will display info on it.

Hit “Start Voice Recognition” and try it out by saying the key phrases into your microphone. If the command is recognized it will be shown in the text box on the right.

If a key phrase is not recognized by the software check your spelling first!

To quickly check that the key is actually pressed start a text editor and bring it to the foreground (click on it). Then say your key phrase and the letter “c” should appear.

*Stop the voice recognition, go to the File menu and save your profile as a *.s2k file. You can open and read these in a text editor if you like.*

Caveats

If you name a command the same as an already existing one you will get a warning that this command will be overwritten. You can hit the “Cancel” button at any point before hitting the final “Finish” button to prevent overwriting a command.

Key phrases must be unique. If you define a key phrase that has already been defined in another command you will get an error. You can only add the command after resolving this conflict by going back and deleting/changing the duplicate key phrase.

Editing and Deleting

In all lists you can (multi)select and hit the 'del' key on your keyboard to delete selected items (commands, key phrases, responses, keys pressed, etc.)

To edit an item double click it. This works for commands, key phrases and responses.

Creating Commands with Key Sequences and Pauses

As an example I will use a fairly elaborate “Request Docking” command I have created for the game “Elite: Dangerous”.

- Click “Add New Command” button
- Give it a name (e.g. “Dock me”)
- type in a key or key sequence. In this case the sequence is: t, F8, and then on the number pad the keys 3,3,5,2,5 and finally the home key. This will do the following with my in-game

key bindings (which are heavily altered from the defaults):

- t will target the object in my sights (which will be the station I want to dock at)
- F8 will bring up the contacts menu
- Number pad 3 is bound to the “go to next tab” function, so we do that twice which will display all current contacts. The current target (the station) will be highlighted.
- Number pad 5 is bound to “select”, This will select the station and bring up the available options for it.
- The “request docking” option is one down. I bound number pad 2 to “move selection down” in the game.
- Number pad 5 will select this option.
- Finally the home key will return me to cockpit view.

To get this all to work I have to add a couple of pauses in between, because the bringing up of menus is not instantaneous in the game. Adding pauses can be done with the four pause buttons. Pauses will be added after a selected item (if none is selected the pause will be added at the end. You can either add pauses during or after typing the sequence). You can also select multiple items and pressing one of the pause buttons will add an appropriate pause after each selected item. So the final command looks like this:

t down

t up

Pause: 0.1 seconds

F8 down

F8 up

Pause: 1 second

Num3 down

Num3 up

Pause: 0.1 seconds

Pause: 0.1 seconds

Num3 down

Num3 up

Pause: 0.5 seconds

Num5 down

Num5 up

Pause: 0.1 seconds

Num2 down

Num2 up

Pause: 0.1 seconds

Num5 down

Num5 up

Pause: 0.1 seconds

home down

home up

- in the key phrases screen I add the phrases “Request docking”, “Get docking permission”, “Dock with station” and “Dock to station”. (I like to give multiple phrases. The first few that come to mind should cover it so I don't have to remember one exact phrase)
- On the responses screen I type “docking in progress” and “sending docking request”, and for good measure I check the “use standard responses” check box in the upper right corner which will automatically add the responses “Roger”, “Affirmative”, “Will do” and “You got it” . One response is chosen at random each time the command is issued.

Seeing the game click through all this (what would be a tedious/repetitive task otherwise) is neat.

Hint: If the game has a toggle button (e.g. “g” for turning ship lights on or off) – but you want to have the appropriate commands and responses then I would create two commands: One with key phrases and responses like “lights on” and another with “lights off“-related key phrases/responses – and have both commands bound to the “g” key.

Hint: To quickly test all your commands hit the “Start Voice Recognition“ button. Then click once on each command to see what key phrases are associated with it and try them all out. The recognition engine uses the .NET speech library and has problems with some words. In that case remove the key phrase or replace it with something that is better recognized.

Hint: Avoid too short key phrases (like “go” and “to”). Medium sized words or small phrases work best, like”engines”, “fire laser”, “target next enemy”, “open hailing frequencies”, etc.

Play around. You'll quickly get the hang of what works and what doesn't.

Quick launch Options

When you loaded up the program there were two other buttons besides “Create a New Profile”on the quick launch screen:

“Load and Launch Profile”

and

“Edit an Existing Profile”.

Both do almost the same thing. They prompt you to load a profile. The first option will just automatically press the “Start Voice Recognition” button so you can get on with loading/playing your game.

Menu Items

At the top there is a Menu Strip with “File”, “Commands” and “Configuration” menus.

In the file menu you have the option to load, save or create a new profile.

Warning: Creating a new profile or loading a profile will cause all unsaved changes to be lost!

The “Commands” menu gives you access to the special commands (but you can also double click on them from the list for the same result) and the “Add New Command” option which does the same as the “Add New Command” button.

“Configuration” lets you edit the standard responses.

Standard Responses

Standard responses are there so you don't have to add the same responses for every command. These are generic things like “Roger” or “Affirmative”. You can add your own (or delete ones you don't like). Standard responses are only used in commands which have the check box in their responses screen checked.

Special Commands

The commands listed below are created per default when creating a new profile and cannot be deleted.

(If you really want to disable them you could set their key phrases to something weird - like “ajkhqehfowehfoewhiofwhe”)

Special Command: AIName

Per default the key phrase for this command is “Alexa”. You can change this to anything you like. It has no keys associated with it and will always respond with “listening”. This is a quick check to see whether the program is still running and will also reset the current recognition loop.

Additionally, if the “Pause Speech Recognition” command (see below) has been issued, speaking the “AIName” key phrase will reactivate voice recognition.

Special Command: Jokes

Here you can add key phrases after which the program will recite a joke (or anything else) from a given number of files.

With the “Add Joke File” button you can add as many files as you like. Joke files must be plain text files and there must be at least one blank line between each joke (and the jokes must, obviously, not contain any blank lines themselves). All jokes will be stored in the profile, so the profile may become rather large if you have many megabytes of joke files handy. As with all other responses the joke to recite will be selected at random. If you select no file then the response will be “I have nothing funny to say”.

Defaults for triggering the “Jokes” command are (feel free to add to/delete from these):

- Tell me a joke
- Give me a joke
- Tell me something funny

Special Commands: Pause Speech Recognition and Reactivate Speech Recognition

After issuing the “Pause Speech Recognition” command the program will keep running, but will not react to anything *but* the “Reactivate Speech Recognition” command *or* the “AIName” command. Either of these will wake up the program to full functionality.

Defaults for the “Pause Speech Recognition” command are (feel free to add to/delete from these):

- Go offline
- Stop listening
- Stop speech recognition

Defaults for the “Reactivate Speech Recognition” command are (feel free to add to/delete from these):

- Resume speech recognition
- Start listening
- Come online
- Reactivate speech recognition

Special Command: Stop Speech Output

This command will abort speech output of the current response (e.g. if you have long responses or the program is in the middle of telling a very long joke and something other is happening which requires your attention).

Default key phrases for the “Stop Speech Output” command are (feel free to add to/delete from these):

- Shut up
- Stop speech output
- Be silent

Special Commands: Teamspeak On and Teamspeak Off

Experimental. Intended to also have the program go into standby mode (just like “Pause Speech Recognition” and “Reactivate Speech Recognition”), but will also press the key configured for Teamspeak. Probably does not work yet as I haven't figured out how to send a key to a non-active window. For now it is advisable to use the “Pause Speech Recognition” command, then press your Teamspeak button. After you're done with Teamspeak say the key phrase for “Reactivate Speech Recognition” or the “AIName” command.

Defaults for “Teamspeak on” are (feel free to add to/delete from these):

- Teamspeak on
- Teamspeak

Defaults for “Teamspeak off” are (feel free to add to/delete from these):

- Over and out
- Teamspeak off

Thanks and legal stuff

A shout-out to Shay for the InputManager.dll that is used in this software for sending key events! His excellent article on this can be found here:

http://www.codeproject.com/Articles/117657/InputManager-library-Track-user-input-and-simulate