

Talkbox Requirements

EECS 2311 Group 4: Jordan, Rohan and Alberto

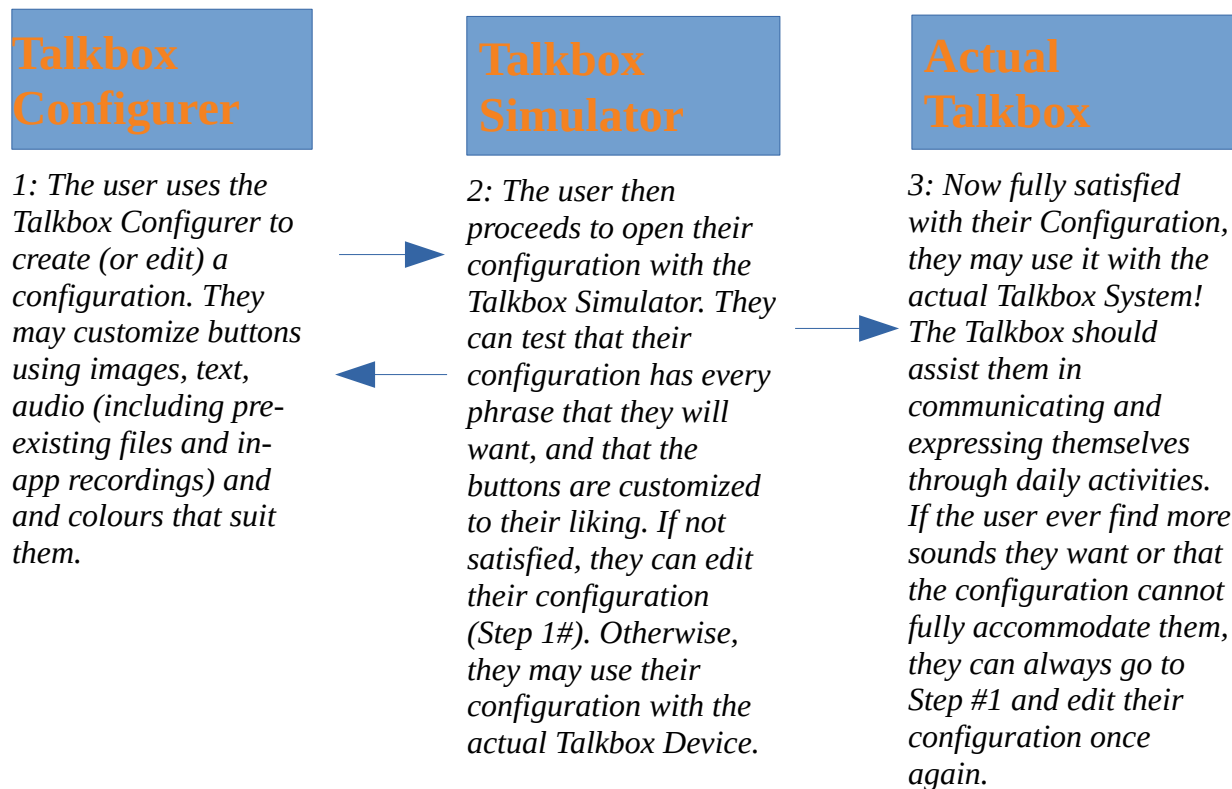
The Core Stakeholders for this project are those who need assistance speaking (and those who assist them).

To address their needs, they must be able to use our applications to setup buttons in an expressive way. They should be able to customize and personalize the buttons (by way of using images, colors, text, emojis, etc) as well as use facilities to record and select existing audio files for the buttons. The buttons should be variable in number – users could be using hundreds of unique phrases every day – so the applications must allow the users to remove, add, or edit the buttons as they see fit.

The **Talkbox Simulator** should allow a user to test whether the configuration suits them. The simulator works very similarly to the actual Talkbox device and users should time with their Configuration seeing how well they can express themselves using it – *perhaps there are phrases they have missed and still need? Does the description not suit the sound or phrase that the button is set up with? How long can the user communicate with the configuration before it is insufficient to express themselves?* The user may continuously edit their Configuration with the **Talkbox Configurer** and test it with the **Simulator** until it is fully satisfactory.

Use Cases

The typical use of the Talkbox Configurer and Simulator should be as follows:



Acceptance Tests

Beyond simply functioning, the Talkbox Simulator and Configurer need to address the needs of the Stakeholders of this project. The following Acceptance Tests should assess whether the applications do indeed assist them with daily activities. Some sample stakeholders (and their assistants/caregivers, if applicable) should run the following tests with the assistance of the User Manual.

Test 1: Using the Talkbox Configurer application to create a configuration of 6 custom buttons, each with their unique sound, and uniquely identifying label.

Particular focus should be directed towards ease-of-use. *Can sounds be recorded or selected for buttons easily? is adding or editing buttons simple?* Does the application exist the user or is it simply a pain to use?

Test One

The test should be considered a failure if:

- The user is stuck for more than 2 minutes for any one task, due to confusion
- It is too difficult to add/remove buttons, edit buttons
- App-breaking critical failures occur during the test
 - Navigating through the application or creating a setup is difficult.

The test should be considered passed if:

- The criteria for a failure is not met, and;
- There are no critical features that the stakeholders feel the application is lacking that would help address their needs.

Step 1: Open the Talkbox Configurer application.

Step 2: The user creates a new Talkbox Configuration directory.

Step 3: The user sets up a button with a sound and a name/label. The button should appear how it would when using the Talkbox Simulator and playback of the selected sound should be available.

Step 4: Repeat step 3 making 2 more buttons. The third button should have its sound recorded directly to the button setup.

Step 5: Close the Talkbox Configurer. The Talkbox Configuration directory is now set up.

Step 6: Open the Talkbox Configurer, and open an existing Talkbox Configuration directory. The user should see their previous buttons, and be able to edit them as well as add additional buttons to the configuration directory.

Step 7: Complete the test by creating 3 additional buttons.

Test 2: Using the Talkbox Simulator application to test a Talkbox Configuration, to see how well it suits the user in daily activities and to check that the sounds will be sufficient.

*The aim of the test is to show whether the Simulator can help the user determine whether a configuration will work for the user or not. Focus should be directed at the ability to **use** the configuration, not at the quality of the configuration itself (which can itself be modified by the user using the configurator.)*

Test Two

The test should be considered a failure if:

- The configuration does not appear like what the user created.
 - App-breaking critical failures occur during the test.
- The buttons do not behave consistently like they would when using the actual Talkbox Device
 - The user cannot consistently use the simulator to communicate.

The test should be considered a success if:

- None of the failure conditions are met

Step 1: Open the Talkbox Simulator application.

Step 2: Open the existing Talkbox Configuration directory from Test 1

Step 3: Ensure that the buttons and sounds are exactly as they appeared when setting up.

Step 4: Try for 3 minutes using the buttons to communicate. *Does it feel like this setup could be used longer?* If the Simulator cannot assist you in answering that question, the test has been failed. If the question can be answered effectively: whether the Talkbox Configuration is sufficient or insufficient, the Simulator has passed the test.

Talkbox Log Requirements

The Talkbox Log application differs from the other two in that it is meant for the Talkbox development team rather than the end users. The developers (that's us) need to be able to assess how their users are *actually* using the applications, rather than how they are designed. This can enable the team to:

- Research the amount of time users spend performing tasks.
- Fix bugs or design problems that are limiting the user experience
- See how updates to the software are affecting user behavior (*does the update really mean that users spend less time having to configure their buttons? Did it even really make a difference*)

Use Case: Bug Fixing

Fixing a bug can be extremely difficult when users do not share our computing environment (Operating system, hardware etc) and when we (the developers) cannot be there to witness the problems that our users encounter.

A very helpful application of the Talkbox Logger app as a part of the Talkbox platform is that it can keep track of user interactions leading up to (and including) an error in the application.

Typical Series of Events:

- 1) The user finds a bug and reports it to the Talkbox team (using the Github issues page)
- 2) The Talkbox team asks for the user's Talkbox Configuration to assist them in solving the problem, and for any screenshots that may also aid them.
- 3) The team opens the Talkbox Logger (or Configurer if the problem was with the Simulator) and views the series of events leading up to the problem, so that they **may replicate the bug** and/or figure out what is causing the problem.
- 4) With the assistance of that information, the team fixes the bug and reports the issue as fixed.

This usage case could substantially reduce the amount of time required to fix software bugs in the Talkbox project because we can see what steps led up to the error using the logger application, which makes it much faster to update the application with the problem fixed. It can also reduce the amount of time spent looking for bugs which turn out to be a mistake of the user rather than a fault of the application.

Use Case: Improving User Experience

The Logger application can also assist the developers in regularly assessing how much time their users are needing to use the applications – and what the bottlenecks preventing the smoothest experience possible are. If we find a certain task is being repeated much more than expected (for example, recording buttons or formatting images) when using the Talkbox Logger application to research user experiences, we can attempt to shorten the task or make it more effective.

Typical Series of Events:

- 1) We regularly ask users to, if they would be willing, send their Talkbox Configuration logs to us so that we can research their experiences with the app.
- 2) We research their usage of the applications and identify tasks that are being performed in ways that surprise us; *perhaps users spend much more time moving up and down sets rather than configuring buttons when using the configurator.*
- 3) We consult users on why the expected user behaviour is so different from the actual activities users perform during the usage of applications. *We don't always need to change the app just because the expected behaviour is different!*
- 4) If we have decided changes are necessary, we can perform changes to the applications so that users can spend time more effectively when using the applications.
- 5) We can then test again and see if our changes have had the effect we were expecting.

Acceptance Test: Log

A good acceptance test that can be run with the developers of the project is as follows. After the other acceptance tests have been performed by clients with the Simulator and Configurer, we take their Talkbox Configuration logs for use with this acceptance test.

- 1) Given sample Talkbox Configuration Logs, the developers will open the Talkbox Logger for viewing both the Simulator and Configurator logs.
- 2) The developers should be able to use the application to easily identify the following correctly on paper:
 - The last five actions performed by the user of the Configuration App.
 - The time at which those operations were performed.
 - At least six different kinds of actions performed.
- 3) The developers should also be able to notice what activities are engaged in most often and categorize those activities in an easy-to-view way.

The test is considered passed as long as there are no major complaints about the capabilities of the application and the second step is able to be completed accurately.

The acceptance test should determine if the Logger is capable of performing the major use cases for the developers. With the test completed, we can be assured that it will be the asset it is meant to be to the project.