Yeemacs Documentation

Yeemacs is a custom made text editor meant for usage alongside the Yeet language. It has been purposefully limited in its functionality in order to provide a smoother experience to the inexperienced. For this reason, we have come to refer to it as an LDE: a light development environment.

In order to better fill the role of the visual interface to the interactive teaching tool that we created, Yeemacs is meant to start by default in fullscreen mode with no way to minimise. We decided on this feature as the Raspberry Pi setup it is run on is meant to only be used for the purposes of learning programming through Yeet. However, for easier assessment and marking, as well as testing, it currently starts in windowed mode and also features a quit button. The startup mode is defined as a macro in yeemacs.c which can easily be edited.

Its current features include line numbering, a console for output, undo and redo functionality through both buttons and keyboard shortcuts, a scrollable window, syntax highlighting and auto indent.
It also boasts a couple of other buttons on the left-hand side that are used to manipulate files.

The 'Open' button prompts the user to select a file on disk to be loaded.
The 'Save' button provides 'Save As' functionality, always asking the user to either select a destination file or type a new file name.

The 'New' button will require the user to save the current file, if they had not previously done so, or saves the file in its current location otherwise. It then clears the editor.
The 'Run' button will do much the same as the 'New' button, but instead of clearing, it will run the current file using the Yeet interpreter, printing both standard output and standard error to console. It also clears the previous output before starting.
Furthermore, it is worth mentioning that theming the editor is easy due to our choice of libraries, as it utilises the system-wide GTK theme.

To compile and run the editor and the language, gtk-3.0 and gtksourceview-3.0 are required as libraries. The drivers utilise the WiringPi library, but they are still under development and are not actively incorporated in the functionality of the language, beyond syntactic representation.
In order to properly use Yeemacs, the following steps must be followed:

- cd to the extension directory
- make
- run ./yeemacs while in the extension directory
- start coding
- RUN your code. This will create a save prompt if the current file has not been saved.

It is important to note that the arm11_01/extension/yeet_lang/demo/ directory contains functioning programmes which showcase some of the language's most interesting features. They should run once opened and provide a good example of the functionality that is allowed by the language in its current state.
Finally, we have decided to leave CMakeLists files throughout the project in order to enable an alternative mode of compiling and running, however, we recommend using Makefiles.