GUI Programming  
Module-Maker

short line

Julian Pahor  
10th, August, 2022

# Template

This is a template you can follow for your user documentation.

Please make sure you delete all placeholder text.

|  |
| --- |
| Note: You can add and remove headings, make sure everything in this document is relevant to your specific project and use case, |

# 

# Table of Content

[Template](#_6pub3u4d4raz)

[Table of Content](#_nrk6hwyq9w2m)

[Purpose](#_zgvrevys5t5)

[Requirements](#_qfaf04jzihtu)

[Features](#_kmov1bt5avnp)

[UI Wireframe Mockups](#_njms22vnkldo)

[Document Styling Guides](#_v62y5wtxl8z7)

[Heading](#_2svhg9kncs5e)

[Sub Heading](#_3l16654yz8l2)

[Sub Heading](#_9kv5nmrdjdib)

[Sub Sub Heading](#_dii88uhwknx0)

# 

# Purpose

Give a brief description of your project

Module-Maker is a game centered around having multiple musical boxes (modules) combining to make some form of music. You have several different types of modules that do different things, and you can interact with them to change their parameters. It’s more of an audio playground rather than a game, where you can mess around and potentially create something pleasant.

# Requirements

The program will be designed and built fully in Unity and won’t need to depend on any other dependencies or pre-installed software besides to run it. It should be low in terms of its demand for processing power.

# Features

Describe features that are a requirement for your tool. Eg: drag / drop commands, switching levels, restart levels, run through actions etc.

* Command Key to interact with modules  
  Description: Having some functionality to be able to walk up to a module and press a key in order to interact with it and bring up its own separate UI.
* Sliders  
  Description: Lots of various sliders used to change parameters within the modules, as well as for master settings for the whole game. (E.g. Master volume)
* Command Key to spawn and place modules.  
  Description: The ability to press a command key to open a window with various module choices. Being able to allow the player to pick and then place whichever module they would like to use.
* First-person character controller  
  Description: The control of the player from a first-person perspective is necessary for the program to run as well.

# UI Wireframe Mockups

Provide a sample wireframe mockup for each screen the user sees. If you have any dialog / popups, show wireframe for those dialogs. If you have tab controls, show a wireframe for each tab.  
  
For each tab, or change in view, you should show a new wireframe / mockup.

|  |
| --- |
| Note: This is an initial plan, a proposal for how your application will flow. |

Screen 1:

|  |  |
| --- | --- |
| **Image Source:** [Click Here](https://i1.wp.com/webdesignledger.com/wp-content/uploads/2012/10/wireframetools04a.jpg?w=970) | Describe the mockup, what the user sees, and purpose for the screen. Call out notable ui elements such as buttons and provide a name for later reference. |

Scene 1: Events:

* Xxxx button:  
  Describe what happens when the button is clicked?
* Close button clicked:  
  What happens? Quits straight away? Prompts to save unsaved changes?
* drag/drop:  
  How should things work when draged/dropped

# 

Screen 2:

|  |  |
| --- | --- |
| **Image Source:** [Click Here](https://i1.wp.com/webdesignledger.com/wp-content/uploads/2012/10/wireframetools04a.jpg?w=970) | Describe the mockup, what the user sees, and purpose for the screen. Call out notable ui elements such as buttons and provide a name for later reference. |

Scene 2: Events:

* Xxxx button:  
  Describe what happens when the button is clicked?
* Close button clicked:  
  What happens? Quits straight away? Prompts to save unsaved changes?
* drag/drop:  
  How should things work when draged/dropped