

MUSIC APP DOCUMENTATION

1-PURPOSE OF THE DOCUMENT(page2)

2-APP WORKFLOW(page3)

3-FOLDERS STRUCTURE(page4)

4-INTERNAL ARCHITECTURE(page6)

5-ANATOMY OF EACH SCENE(page8)

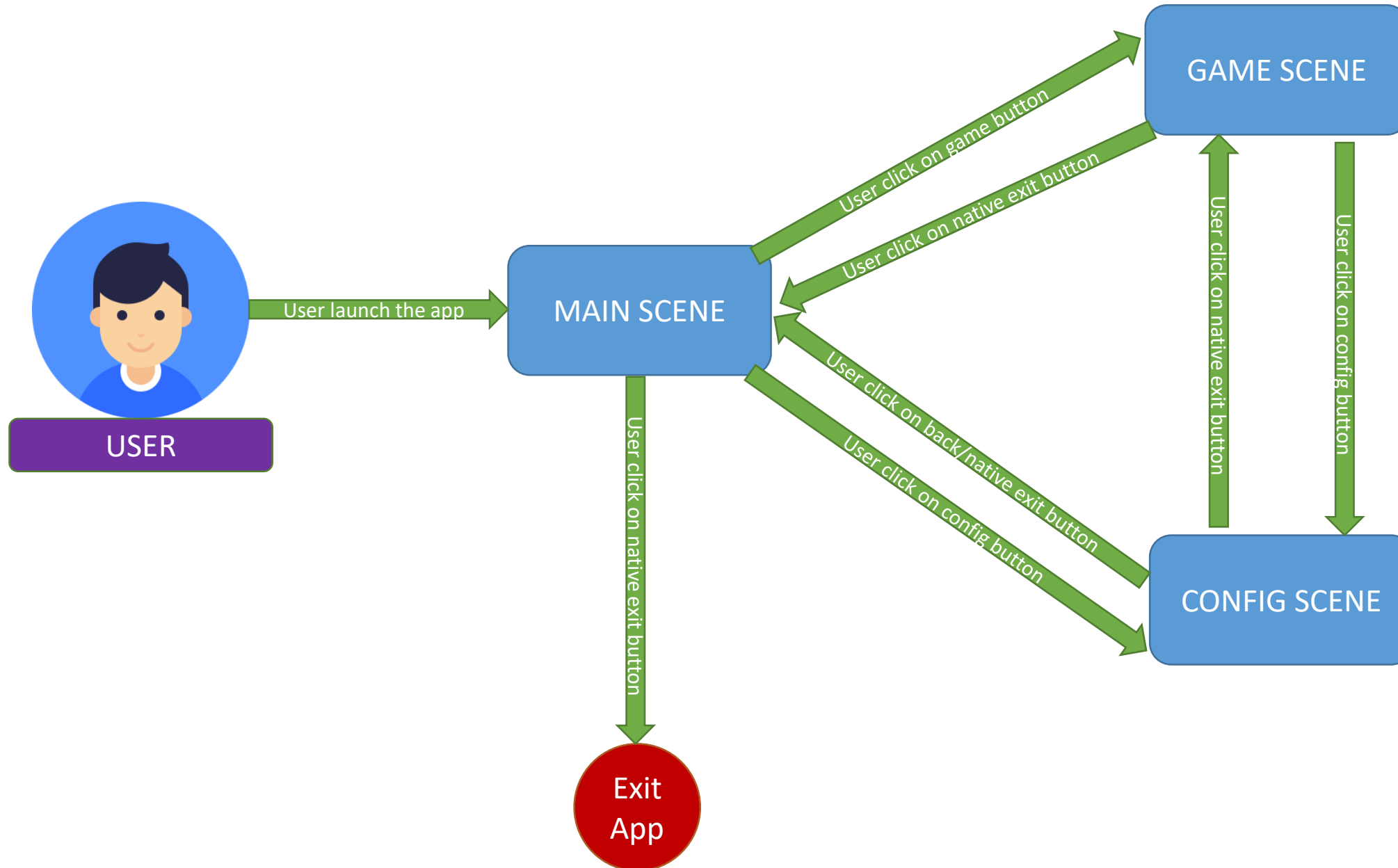
6-CLASS REFERENCE(page13)



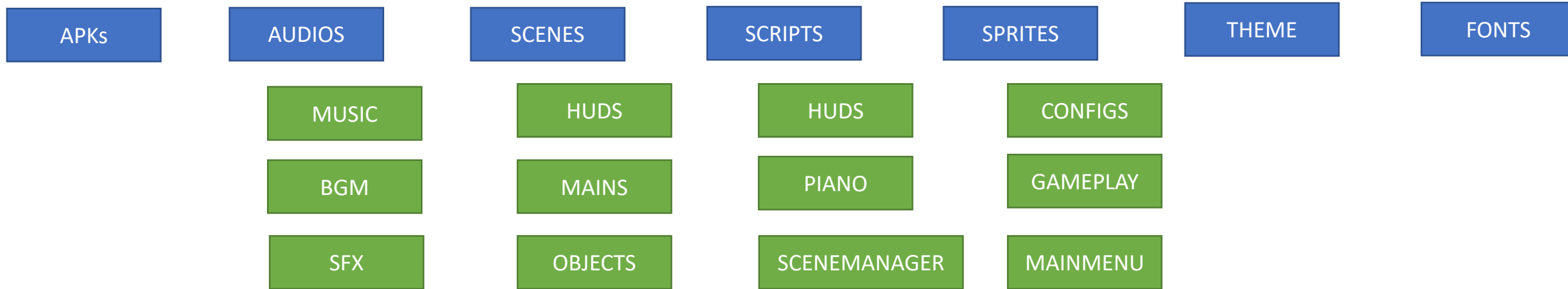
PURPOSE OF THE DOCUMENT

The purpose of this document is to present the internal architecture of the application so that it can be understood and modified by another developer without too much difficulty.

APP WORKFLOW



FOLDERS STRUCTURE(1)

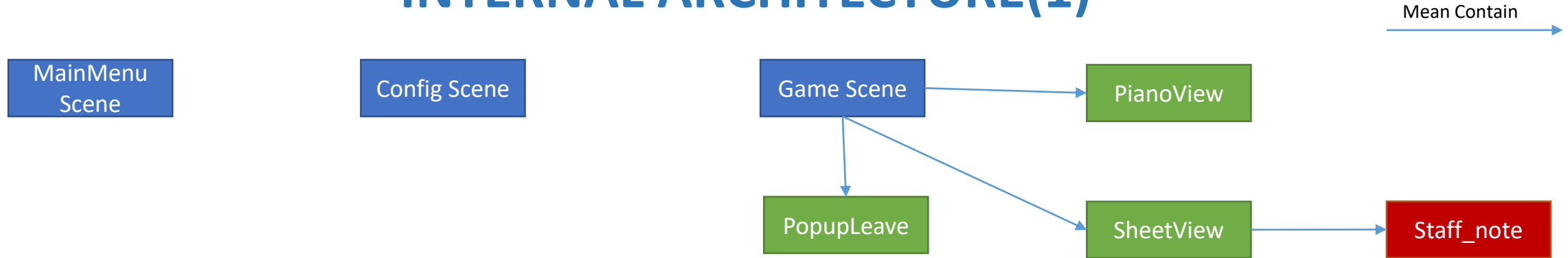


- **APK:** this folder will contain the different apk/aab builds of the app.
- **AUDIOS:** this folder contains the different sounds of the application (music, bgm, sfx)
 - **MUSIC:** music are in this folder.
 - **BGM:** BGM are in this folder
 - **SFX:** sfx like piano key sounds are in this folder.
- **SCENES:** this folder contains all scenes of the app (scenes of screens, scenes as prefab).
 - **HUDS:** all hud prefabs are in this folder.
 - **MAINS:** all the main scenes of the app are here(mainmenu scene, config scene, game scene, loading scene ...).
 - **OBJECTS:** all prefab used in other scene are in this folder

FOLDERS STRUCTURE(2)

- **SCRIPTS:** this folder contains all scripts of the application.
 - **HUDS:** all huds/ui scripts are in this folder.
 - **PIANO:** all the scripts concerning the piano's behaviour are in this folder.
 - **SCENEMANAGER:** all scripts for handle each main scene of the app is in this folder.
- **SPRITES:** this folder contains all images of the application.
 - **CONFIG:** config scene images are in this folder.
 - **GAMEPLAY:** gameplay(where are played) scene images are in this folder.
 - **MAINMENU:** mainmenu scene images are in this folder.
- **THEMES:** this folder contains all ui theme of the app.
- **FONTS:** this folder contains all text fonts of the app.

INTERNAL ARCHITECTURE(1)



- **MainMenu Scene:** this scene is the front door of the application. it contains buttons that lead to other scenes (config, game). it is located in the main scenes folder.
- **Config Scene:** this scene contains all the configuration of the application (tempo, background, ...). it is accessible from the mainmenu scene and from the game scene, it is located in the main scenes folder.
- **Game Scene:** this scene is the most important one in the app. this is where the gameplay is played. it contains the piano view and the sheet view.

INTERNAL ARCHITECTURE(2)

- **PopupLeave:** it is a scene(but used as prefab) that is called in the game scene to ask the user to confirm if he wants to leave the game scene or not.
- **PianoView:** it is a scene(but used as prefab) that contains the view of the piano with all keys. it is called in the game scene.
- **SheetView:** this is the scene(but used as prefab) that contains the sheet and is called in the game scene.
- **Staff_note:** it is the scene(but used as prefab) that contains the musical note that appears in the music sheet.

ANATOMY OF EACH SCENE(1)

MainMenu scene

this scene is handled by the script [MainMenu.gd](#)

When button down, call `_on_btnGame_button_down()` function and this function play scale animation

When button released, call `_on_btnGame_pressed()` function to go to game scene



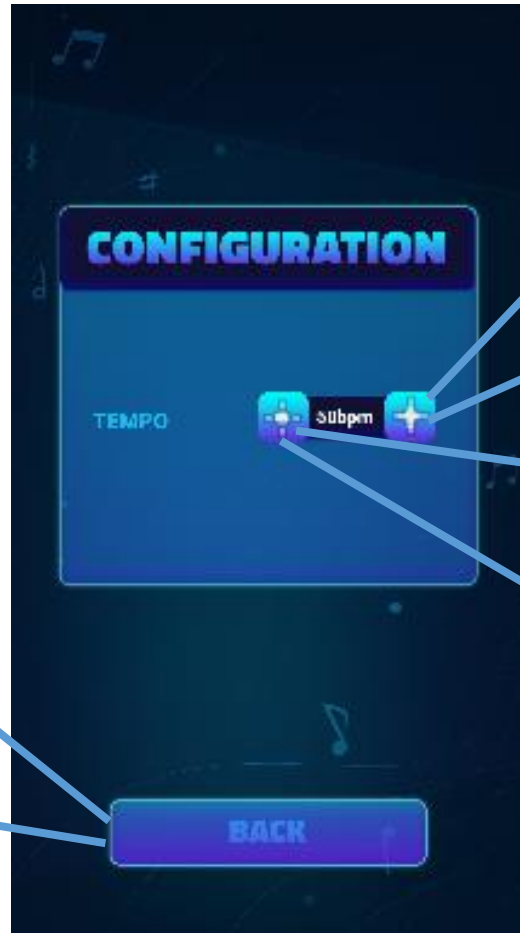
When button down, call `_on_btnConfig_button_down()` function and this function play scale animation

When button config released, call `_on_btnConfig_pressed()` function to go to config scene

ANATOMY OF EACH SCENE(2)

Config scene

this scene is handled by the script `ConfigMenu.gd`



When button down, call `_on_btnBack_button_down()` function and this function play scale animation

When button released, call `_on_btn_pressed()` function to go to game scene or main menu scene depending on the previous scene.

When button down, call `_on_btnAdd_button_down()` function and this function play scale animation

When button config released, call `_on_btnConfig_pressed()` function and this function:

- Scale the button back to normal
- - increase the value of tempo

When button down, call `_on_btnRemove_button_down()` function and this function play scale animation

When button config released, call `_on_btnRemove_pressed()` function and this function:

- Scale the button back to normal
- Decrease the value of tempo

ANATOMY OF EACH SCENE(3)

GameScene scene

this scene is handled by the script `GameScene.gd`

When button down, call `_on_btnRemove_button_down()` function and this function play scale animation

When button released, call `_on_btnConfig_pressed()` function to go to config scene

the piano view. this view is an independent scene with its own behaviors

view of the Sheet. it's an independent scene with its own behavior

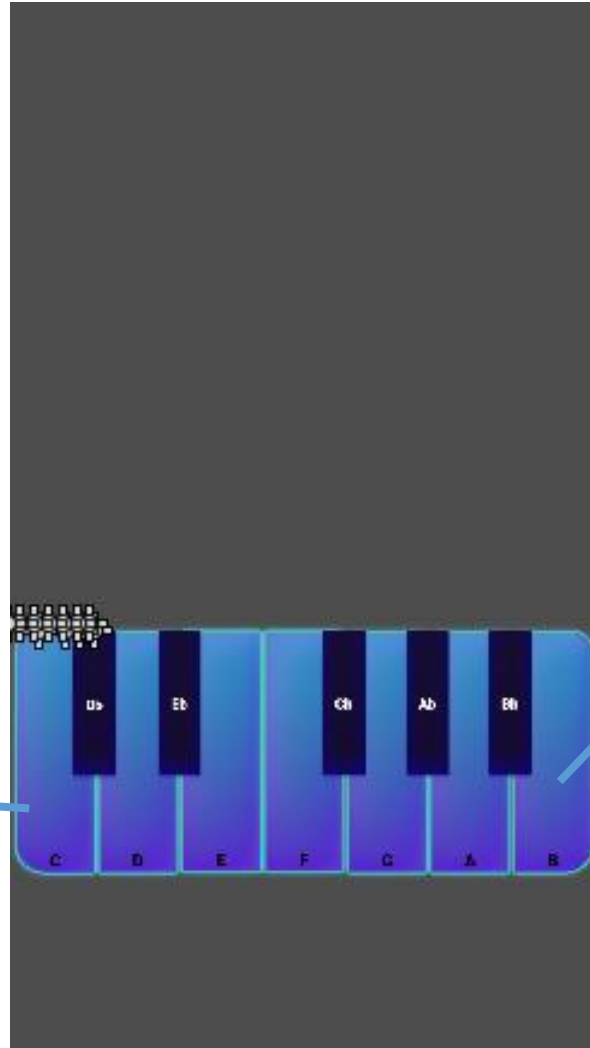
ANATOMY OF EACH SCENE(4)

PianoView scene

this scene is handled by the script `PianoPlay.gd` and it is used in the game scene

Each piano key is handle by the script
`Pianokey.gd`

When Key down, call
`_button_down()` function and
this function :
-play scale animation
➤ Play the sound according
to the key pressed
➤ Emit an
`on_pianokey_pressed`
event(this event will be
use to evaluate the result
after)



When button released, call
`_pressed()` function to get
the scale back to the
normal,

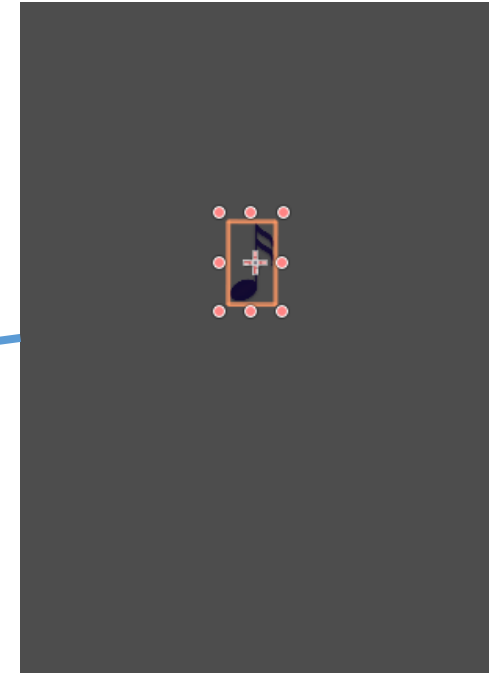
ANATOMY OF EACH SCENE(5)

SheetView scene

this scene is handled by the script [SheetView.gd](#) and it is used in the game scene



Instantiate here



- Each line represents the position of a note in the staff and is handled by the script `StaffNoteLine.gd`.
- Note are instantiated on the right and move to the left on the corresponding line.

The staff note scene(used as prefab) to instantiate in the sheetview. After instantiation, the staff note move from right to left,

CLASS REFERENCE

Class references are in a Zip file on github. [ClassReference.Zip](#)