Raquel Jones Danzel Capers

CSCI 380

12/11/15

Final Project

Fractaural

**User Goal**: “I want to make art along with my music.”

**Hierarchical Task Analysis:**

0.Turn music into art (Plan 0: Do 1,2,3 in order and 2 can only be done after 1 is completed and 3 can only be done after 1 or 2 is completed)

1. Make Music ( Plan 1: according to requirements)

**1.1 Enter music (Plan 1.1: 1.1.1-1.1.5)**

**1.1.1 Click menu button**

**1.1.2 Select instrument**

**1.1.3.1 Drums**

**1.1.3.2 Strings**

**1.1.3.3 Keyboard**

**1.1.3.4 Other**

**1.1.3 Select version of instrument**

**1.1.4 Select pitch**

**1.1.4.1 Move slider**

**1.1.5 Press desired location on music sheet**

1.2 Load existing music (Plan 1.2: 1.2.1-1.2.4)

1.2.1 Click menu button

1.2.2 Click load

1.2.3 Select file

1.2.4 Press file

1.3 Edit music

1.3.1 Modify music

1.3.1.1 Delete section of music

1.3.1.1.1 Select music block

1.3.1.1.2 Click edit button

1.3.1.1.3 Press delete button

1.4 Save music

1.4.1 Click menu button

1.4.2 Click save

1.4.3 Enter file name

1.4.4 Press file

1.5 Play music

1.5.1 Click menu button

1.5.2 Click play

**2. Convert Music to Art**

**2.1 Sync with mobile device**

**2.1.1 Click sync button**

**2.2 Convert music**

**2.2.1 Click convert button**

**2.3 Save art**

**2.3.1 Click menu button**

**2.3.2 Click save**

**2.3.3 Enter file name**

**2.3.4 Press file**

3. Share Creation

3.1 Send to user

3.1.1 Click share button

3.1.1 Click email

3.1.3 Enter user email

3.1.4 Press Send

3.2 Post online

3.2.1 Click share button

3.2.2 Select media

3.2.2.1 Facebook

3.2.2.2 Twitter

3.2.2.3 Other

3.2.3 Press send

Figure 1 below shows the home page that users will start off with.

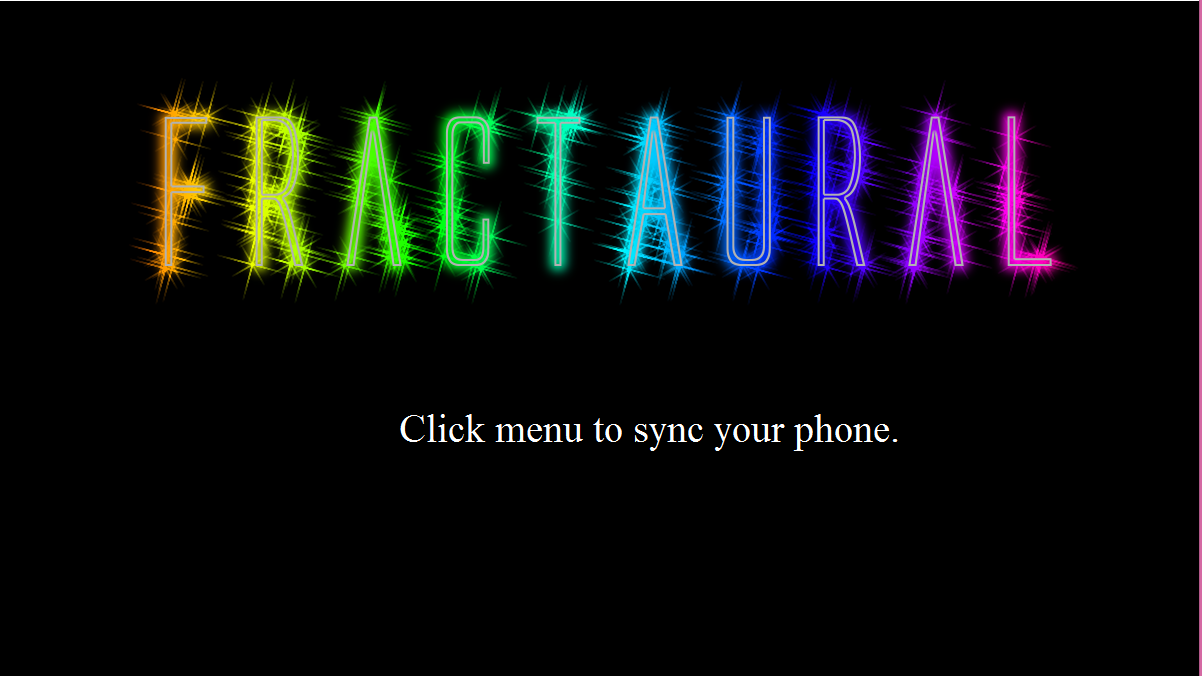


Figure 2 below show the drop down list of Menu named menu. Allows user to connect to mobile device and clear artwork to start over. All other functions are inactive for this stage in development.

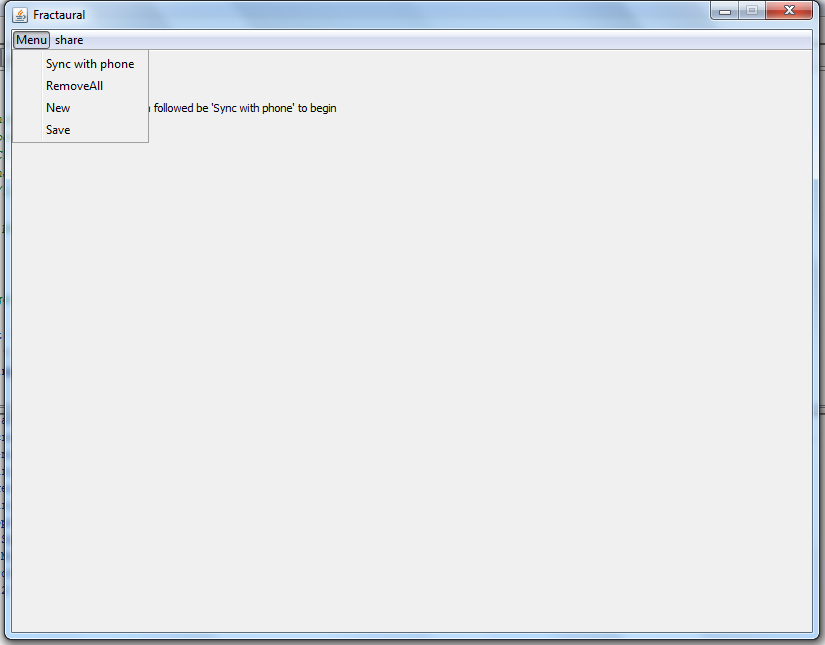


Figure 3 below show the drop down list of Menu named share. All other functions are inactive for this stage in development.

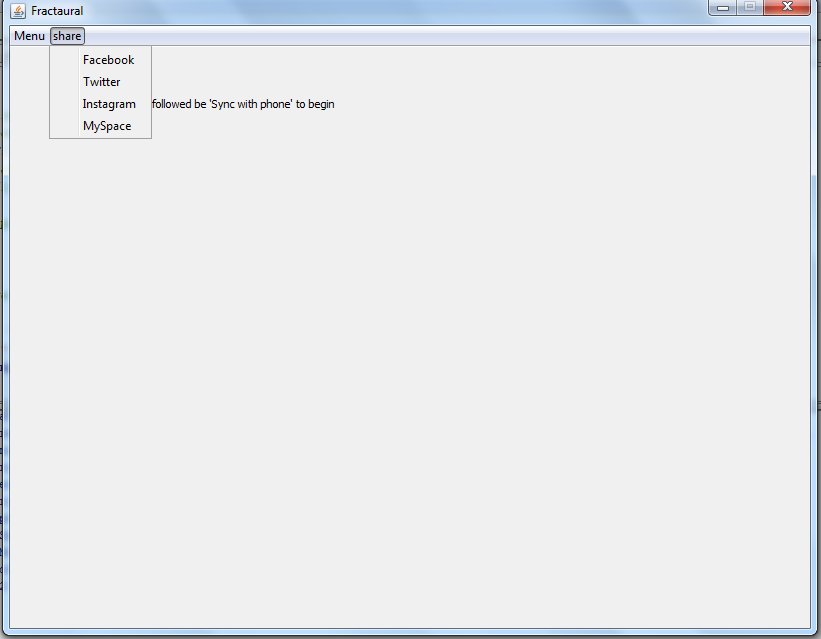


Figure 4 below shows the display after a few notes have been played.

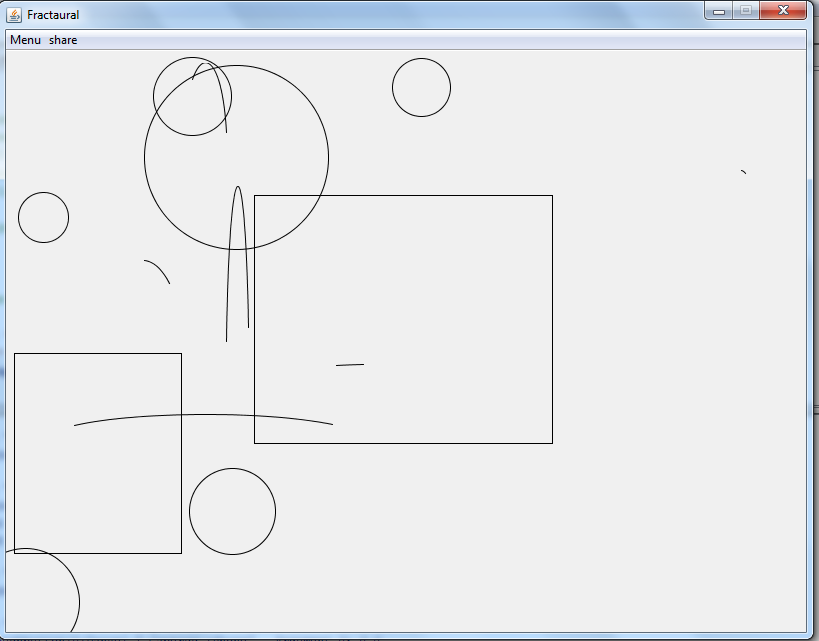


Figure 5 below shows the results of selecting an inactive function.

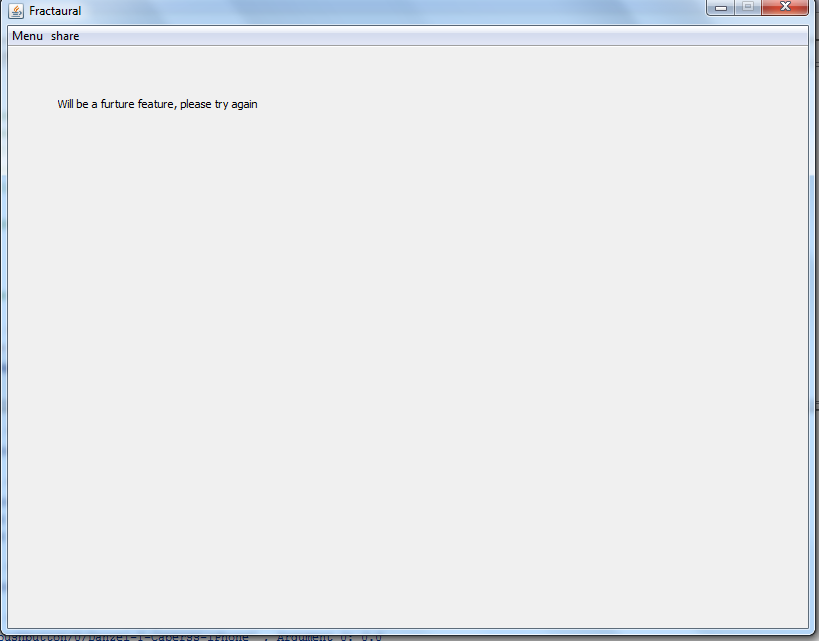
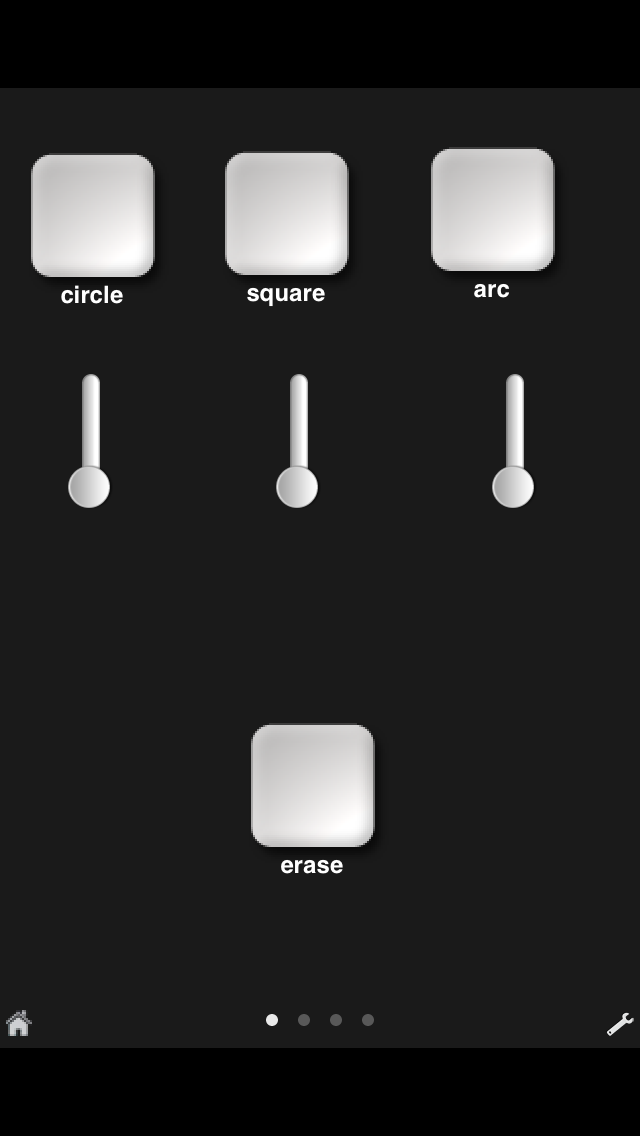


Figure 6 gives a layout of the mobile device being used for the project. Three buttons up top represent notes that can be played, the sliders are inactive but will allow users to change the pitch of a note, will be applied in a future update. The last button at the bottom is a reset button that clears the display window.



**README instructions:**

After downloading the following files: “main.py”, “view.py”, “controller.py” and “model.py”. You will need to download the following app for ios or android: “Mrmr” you will need to set up your IP address by going to Prefs icon to change your server ip address and port to match your computer in file controller.py line 71. Change OSC mobile display as shown in figure 6 and assign each button addresss to the appropiate “oscIn.onInput” assignment as seen in controller.py lines 73-76. After running the program main.py you need to go to the menu bar and select “Sync with phone” to start the process of linking the phone to the server. Once the computer has setup a server you connect to it by selecting Add a server on the main page of the Mrmr app and manually adding the server IP address.

**appendix:**

# main.py

#

# Author: Danzel Capers, Raquel Jones

# Email: capersdt@g.cofc.edu, jonesra1@g.cofc.edu

# Class: CSCI 380

# Assignment: Final Project

# Due Date: 12/11/2015

#

# Purpose: Setup project by running view.py, model.py and controller.py

#

# Input: None

#

# Output: Message

#Imports modules

from model import \*

from view import \*

from controller import \*

# do whatever else needs to be done (if anything)

print

print "Click Menu then select Pair with Phone to conect with moblie device..."

print

# view.py

#

# Author: Danzel Capers, Raquel Jones

# Email: capersdt@g.cofc.edu, jonesra1@g.cofc.edu

# Class: CSCI 380

# Assignment: Final Project

# Due Date: 12/11/2015

#

# Purpose: Displays windows for user to interact with

#

# Input: None

#

# Output: Displays

from gui import \*

from osc import \*

import model, controller # import other application modules

frontPage = Display("Fractaural", 1200, 800)

logo = Icon("logo.png")

frontPage.setColor(Color.BLACK)

#syncButton = Button("Sync with phone",callOSC)

#frontPage.add(syncButton, 450,500)

frontPage.add(logo, 50, 0)

model.textlabel.setFont( Font("Serif", Font.PLAIN, 40) )

frontPage.add(model.textlabel, 400, 400)

#Creates Menu menu

menu = Menu("Menu")

menu.addItem("Sync with phone", controller.callOSC)

menu.addItem("RemoveAll", controller.removeALL)

menu.addItem("New", controller.callNothing) #Future feature

menu.addItem("Save", controller.callNothing) #Future feature

frontPage.addMenu(menu)

#Creates Share menu

share = Menu("share")

share.addItem("Facebook", controller.callNothing) #Future feature

share.addItem("Twitter", controller.callNothing) #Future feature

share.addItem("Instagram", controller.callNothing) #Future feature

share.addItem("MySpace", controller.callNothing) #Future feature

frontPage.addMenu(share)

# controller.py

#

# Author: Danzel Capers, Raquel Jones

# Email: capersdt@g.cofc.edu, jonesra1@g.cofc.edu

# Class: CSCI 380

# Assignment: Final Project

# Due Date: 12/11/2015

#

# Purpose: Setup functions

#

# Input: Variables from model.py and gui images from view.py

#

# Output: Displays from view.py

import view, model

from gui import \*

from random import \*

from osc import \*

#Draws a Circle with random specs

def drawCircle(message):

model.Play.midi(model.note1)

model.x = randint(0, view.frontPage.getWidth())

model.y = randint(0, view.frontPage.getHeight())

model.z = randint(0, 100)

circle1 = Circle(model.x, model.y, model.z)

circle1.setColor(Color.RED)

view.frontPage.add(circle1)

#Draws a Rectangle with random specs

def drawRectangle(message):

model.Play.midi(model.note2)

model.x1 = randint(0, view.frontPage.getWidth())

model.y1 = randint(0, view.frontPage.getHeight())

model.x2 = randint(0, view.frontPage.getWidth())

model.y2 = randint(0, view.frontPage.getHeight())

rectangle1 = Rectangle(model.x1, model.y1, model.x2, model.y2)

rectangle1.setColor(Color.GREEN)

view.frontPage.add(rectangle1)

#Draws a Arc with random specs and plays a single note

def drawArc(message):

model.Play.midi(model.note3)

model.x1 = randint(0, view.frontPage.getWidth())

model.y1 = randint(0, view.frontPage.getHeight())

model.x2 = randint(0, view.frontPage.getWidth())

model.y2 = randint(0, view.frontPage.getHeight())

model.angle1 = randint(0, 180)

model.angle2 = randint(0, 180)

arc1 = Arc(model.x1, model.y1, model.x2, model.y2,model.angle1,model.angle2)

arc1.setColor(Color.WHITE)

view.frontPage.add(arc1)

def removeAll(message):

view.frontPage.removeAll()

def removeALL():

view.frontPage.removeAll()

def callNothing():

view.label1 = view.Label("Will be a furture feature, please try again")

view.frontPage.add(view.label1, 50, 50)

def callOSC():

view.frontPage.remove(model.textlabel)

oscIn = OscIn(5380)

#oscIn.onInput("/.\*",drawCircle)

oscIn.onInput( "/mrmr/pushbutton/4/Danzel-T-Caperss-iPhone", drawCircle )

oscIn.onInput( "/mrmr/pushbutton/5/Danzel-T-Caperss-iPhone", drawRectangle )

oscIn.onInput( "/mrmr/pushbutton/6/Danzel-T-Caperss-iPhone", drawArc )

oscIn.onInput( "/mrmr/pushbutton/3/Danzel-T-Caperss-iPhone", removeAll )

# model.py

#

# Author: Danzel Capers, Raquel Jones

# Email: capersdt@g.cofc.edu, jonesra1@g.cofc.edu

# Class: CSCI 380

# Assignment: Final Project

# Due Date: 12/11/2015

#

# Purpose: Setup variables

#

# Input: None

#

# Output: None

from music import \*

from gui import \*

# initialize Variables

text = "Click menu to sync your phone."

textlabel = Label(text, CENTER, Color.WHITE, Color.BLACK)

x1 = 0

y1 = 0

x2 = 0

y2 = 0

z = 0

angle1 = 0

angle2 = 0

note1 = Note(C4,HN)

note2 = Note(F4,HN)

note3 = Note(B4,HN)