

Bo Fu

Pablo Rivas

Project 2-Write Up

3 May 2017

Final Write Up: Puzzle Game

Abstract:

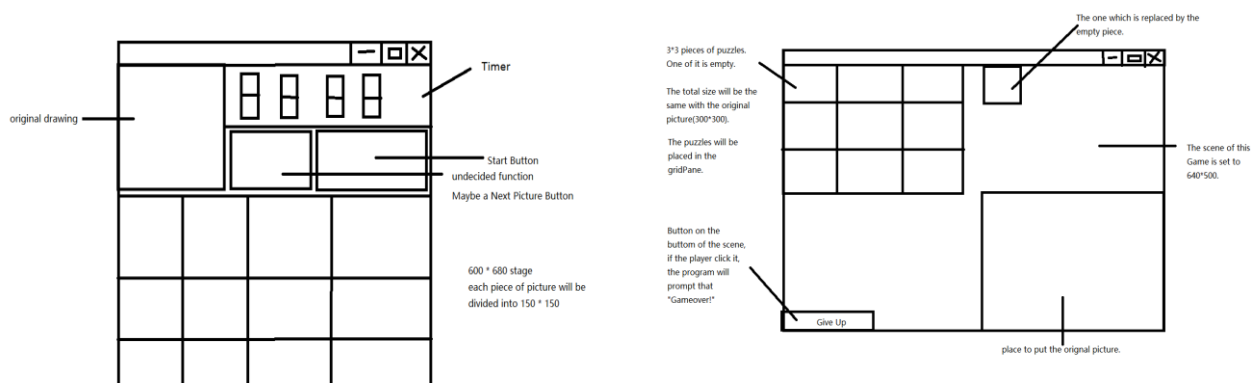
For the Project-2, I am going to design an application program of puzzle game.

Within this paper, I am going to explain the process of each step from I have come up with this idea, to finally all my idea in mind has come into real as a program.

Introduction:

The subject for this Project is a puzzles game. The game will be generating random ordered pieces of picture, and player will try to repair the picture into the original picture.

After few tries of design, I have come up a draft layout. The one on the left is actually the



one that I first want to try. However, since I know only few of the knowledge about JavaFX

so after many times failure, and learned from the others I have finally come up with the

design which is shown on the right. The motivation for me to design a program of Puzzle game is because I would like to learn the game design concentration and become a game designer, the process of design a game through the self-learning process will help me get deeper understanding of Java.

Detailed System Description:

The puzzle game is a game played on a 3*3 square, the pictures are divided into 9 pieces, and one of it will be empty. The puzzles will be random ordered; player are allowed to switch the pictures which is surround the empty one. If the player has win this game, the program will prompt to them that a window showing “Success!”. Here is the UML diagram of this program.

Puzzles Game
Order: int
PicMatrix: int
Perfection: boolean
Layout()
button()
Timer()
randomOrder(Order)
Win(Order, PicMatrix)

Before

PuzzleGame	
m	int
n	int[]
random()	int[]
random_num()	int[]
iso(int[] num)	boolean
swapping(int, int)	
issucc()	boolean
findmun(int[] n)	int
myevent	
handle(Mouse Event)	

After

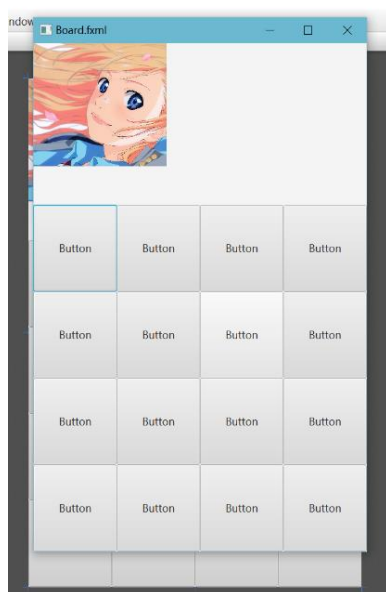
On the left, is the UML diagram which I wants to use at first, but as I have mentioned before, I find it is not the proper one to achieve this program, so I finally decide to change it entirely into the right one showing above. The PuzzleGame class contains the main

function program, the myevent class is only a class deal with the Mouse Event when clicked.

Within PuzzleGame, m is an integer that represent for the number of empty puzzle, n is an array which has the picture place into it. Random() is to generate Inversion sequence of number, random_num() is to generate 8 numbers without repeat, iso() is to make sure that the Inversion pair of the sequence of number will be an even number (but not zero). The swapping() is use to swap the puzzles which is been clicked and also place around the empty one. The issucc() is always check for the puzzles if they are in the right place after each click.

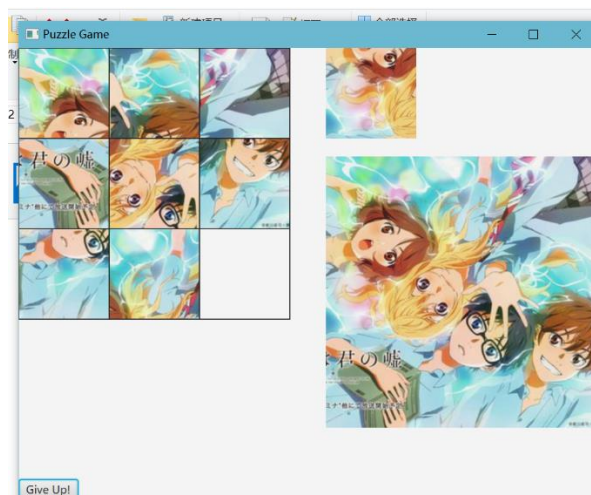
Requirement:

This game will be designed by using the knowledge of JavaFX, array, Java.Math.random, Inversion. Since the JavaFX is a knowledge that we will not learned in class, I find many troubles while I learned it. At first, after I learned Button and GridPane, I decided to write this program by using Button.setGraphic() to set all the picture into the button, and all the button will be fit in the 4*4 GridPane. If people click the button, the button



can change the location. After fell to do it by coding, I try to use the JavaFX scene builder and eclipse, because that will be easier to build the scene. However, the problems next I find is that I do not know how to change the picture while click the button. After watch some tutor video in YouTube, and search on the Java program online community, I find a tutor write the process of coding a puzzle game when he is a beginner to

JavaFX, his creative idea helps me a lot when I read his note write online. He has discuss about how to create a method can random place the number and has a sequence of number has even number of Inverse pair, which means that the puzzle



will always possible to repair into the original picture, do not have the possibility to have last two pieces cannot repair. Another idea which I think is so creative is that instead of putting all the picture into buttons, just add a simple Mouse Click Event will help to swap the picture easier than code for each button. With all his help, the program can be finally created which is shown above.

Literature Survey:

With the brief understanding of the game, I find that I still need to work on the JavaFX parts to support all of my idea come into real. This puzzle game can be set different picture by change the name of the picture in the folder, and it does not need to cut each picture through the Photoshop, because it can cut all of the picture by itself, using the rectangle in JavaFX. This is the biggest different than the other puzzle game, because user can download every image into the folder and play it without too much setting. The picture is placed in even number of Inversion pair, so it can be repair every time player play it.

User Manual:

First while player open the game, there is already a random placed picture in the game, the user can start to play it. The rule is really simple, player can move the puzzle near the empty one, so that the can repair the puzzle into the original picture. There is an alarm set in the game when user win it will automatically prompt out and show congratulation to the user. After few tries, the player might want to play with other picture, the way to change the picture is not so hard. First of all, get a picture which should be the size of 300*300. If the picture is larger than that size, the program will still work, but puzzles might not get full square of pictures. 300*300 is a perfect size for this game because I find if the picture is too large, even with full screen running this game, it might not show everything. After put the picture into the folder, rename the picture's name on this folder or change the name in the code (I have common the place to set the picture's name in the code). There is a "give up" button, it only work to tell you that the game is over, it would not close the game, but since the window itself have a close button, so it is not needed to add this function into the give up button.

Conclusion:

I have to admit that since I took a long time confusing about the JavaFX while I am coding, I end up it in rush. All the design in this program focus more on the functional parts, I pay less attention about the user experience while they use it. I have to say that I set a really high goal to achieve, and the way to achieve it is much harder than I expect. The "give up"

button should be the place to change the picture, so that users will not need to set the picture by their self, and also a notice for the player how much step they use to repair the picture. However, since the deadline is coming, I have to end it up after many failed tries.

However, the process of self-learning JavaFX is really beneficial to me, because I get to learned a lot knowledge that is useful, and I can hardly wait to come back to China and work along with my friends there to develop a better game by using the JavaFX (They have tried to program the Plant VS. Zombie last semester together). This project really helps me to learn that how to self-study the programing which is always focus on the smallest and the easiest part at first. After overcome all those difficulty, I believe I can get one step closer to the real software developer.

References:

Introduction to Java Programming 10th edition

Acknowledgement:

Last but not least, I want to express thank to all the people who have helps me along with the process of come up with the program.

Thank you Lsxxk, even we do not even meet before, but your note helps me a lot as a beginner of the JavaFX. Your experiences and ways of thinking change me a lot as a programmer, you teach me more than how to write program, but also how to come up the blueprint at the beginning.

Thanks for the YouTube uploader who has upload a video of using eclipse for JavaFX, even I fail to achieve it by using it, but still thank your help, you have introduced me a new way to create scene in JavaFX.

Also, thanks to professor Pablo Rivas, I still remember at the begin of this semester I have come up with the topic of this project with you. Even though JavaFX is not the area for you, but you still teach me how to use the limited resources to self-learn it. Give a man a fish; you have fed him for today. Teach a man to fish; and you have fed him for a lifetime. The things more important than the knowledge is the way how to learn the knowledge.

Thank you for everybody helps me in this project again, without your support, this project cannot even finish by myself, all of you have played an important role through this semester.