PROJECT SWIM AWAY!



Submitted by 6404062663134 Nuttapat Ngernshoosri

Present

Asst.Prof. Sathit Prasomphan

040613204 Object-Oriented Programming

KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH BANGKOK

Table of contact

Origin and importance	3
Benefit of Project	3
Development	5
Class Diagram	6
Classification	7
The Main Algorithm	18
Summary	18

Introduction

Origin and importance: Back then , when I was a kid. I like to play these kind of game which you need to get away from something like police or monster. So I tried to create it with java.

Project Type: Game

Benefits: It will improve your aptitude, tactic and prolem solving and having fun with it.

Project scope:

Detail

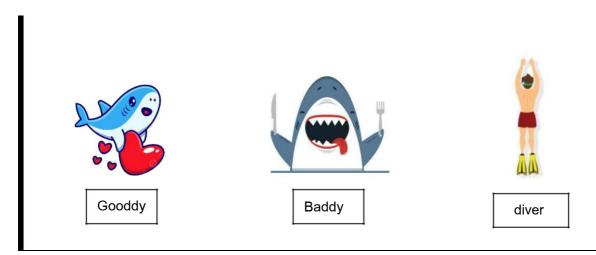
When a guy got left behind between the pacific ocean among the sharks. He has to avoid them to go back to his family. At least, he's a diver ,so he's faster than them. Let cheer him up to leave these scary guys.

How to play

press "A" to go left , press "D" to go right to get away from shark press "P" to pause press "T" to show draw time

Storyboard

-characters



-Scene

-Start





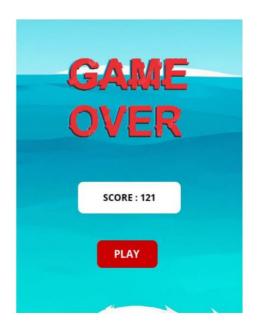
- If diver got attacked by baddy, heart will decrease and if diver got helped by goody heart will increase.





- If you pass them "YOU WIN" but if not "GAME OVER"





Benefits

- 1. For improving aptitude
- 2. For fun
- 3. To be patient

working table

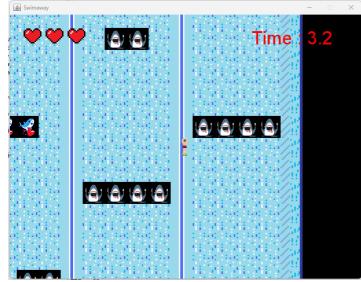
ลำดับ	รายการ	5-15 ก.ย.	16-25 ก.ย.	26-30 ก.ย.
1	หารูปตัวละครและทำกราฟฟิคต่างๆ			
2	ศึกษาเอกสารและข้อมูลที่เกี่ยวข้อง			
3	ลงมือเขียนโปรแกรม			
4	จัดทำเอกสาร			
5	ตรวสอบและแก้ไขข้อผิดพลาด			

Development:

How to play the game: First, you need to select the start button at first screen then it will appear the 'how to play stage'. Then you can choose that you want to go back or play the game. After you select play then when you entered the play scene you can move with "w,a,s,d" and you have to avoid the bad shark until you finish the score will appear on the top left.

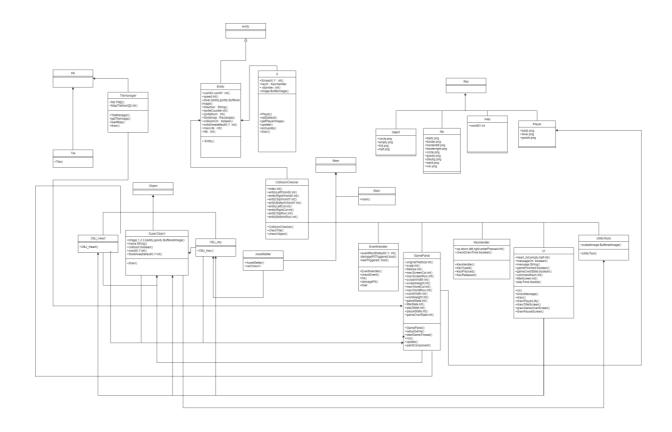








Class diagram

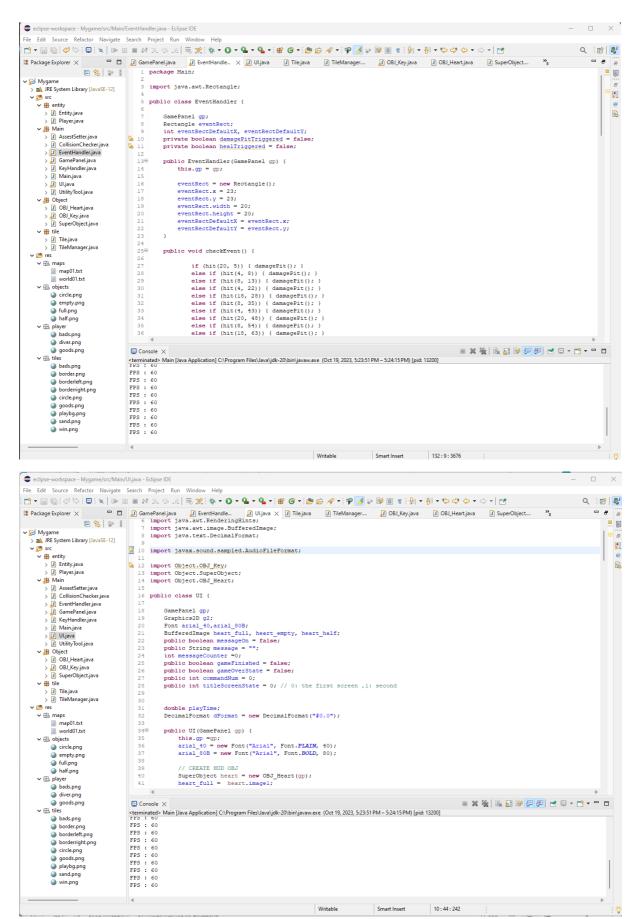


This class Diagram it represent about relation in my project int Main is Game class it will have a Main class Object class and. Then Main have 7 subclass is KeyHandler, it will check if keyboard has press, EvenHandler, it will check Events happen. GamePanel, it is the appearance of the game.UI is for what will show in window and Utilitytools is for debugging the game.Others are tiles and res. It contains the pictures that used in the game with have character enemy ,player or even tile of water, sand or border.And the last one is object that you can do something with it such as heart or collision of the border, enemy and player.

Development model: Application

Constructor:

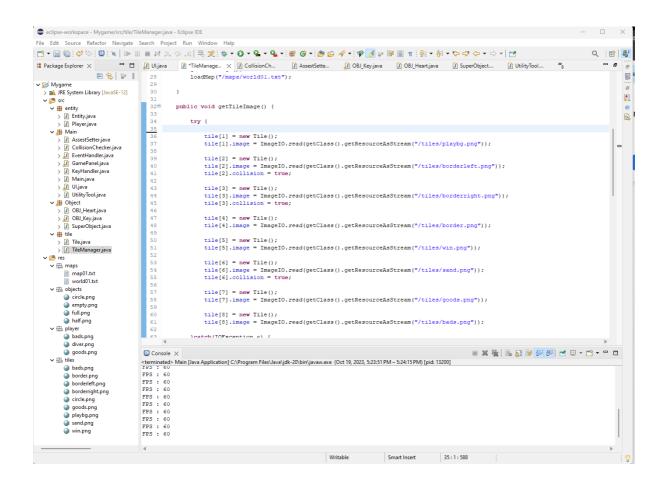
you can see in line 13 constructor to define a value in class it has GamePanel gp implement and call functions. In this case is to do some event such as get heal or hit by enemy.

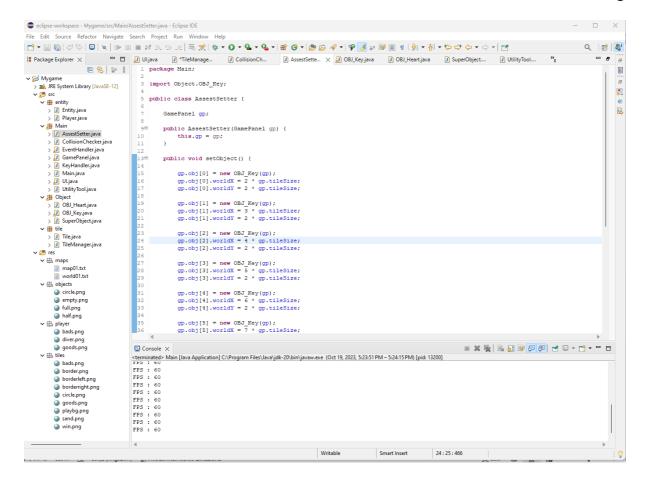


Another constructor is in 32nd line and it also called GamePanel gp which it will show something in window such as time and texts.

Encapsulation:

Gettileimage function in this picture is to get picture that's allocate in /tiles/ which using array to contain it and has to define number to collect and store it.





setObject in this picture will set the X and Y in the map.

Composition:

```
eclipse-workspace - Mygame/src/Main/CollisionChecker.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

\[ \begin{align*}
    \begin{align*}
   \

    ✓ Mygame
    → MRE System Library [JavaSE-12]
    ✓ Src
    ✓ entity
                                                                                     3 import entity.Entity;
                                                                                                                                                                                                                                                                                                                                                                                                                             .
                                                                                    5 public class CollisionChecker {
                                                                                                                                                                                                                                                                                                                                                                                                                             @
           >  Entity.java
>  Player.java
>  Main
                                                                                              public CollisionChecker(GamePanel gp) {
                Main

1 AssestSetter.java

2 CollisionChecker.java

2 EventHandler.java

3 KeyHandler.java

4 KeyHandler.java

5 Main.java
                                                                                                this.gp = gp;
                                                                                                public void checkTile(Entity entity) {
                                                                                                       int entityLeftWorldX = entity.worldX + entity.solidArea.x;
int entityRightWorldX = entity.worldX + entity.solidArea.x + entity.solidArea.width;
int entityTopWorldY = entity.worldY + entity.solidArea.y;
int entityBottomWorldY = entity.worldY + entity.solidArea.y + entity.solidArea.width;
                                                                              16 17 18 19 20 21 22 23 24 25 26 27 28 29 31 32 33 34 35 36
                    Ul.java

☑ UtilityTool.java

             ✓ ♣ Object

> ② OBJ_Heart.java

> ② OBJ_Key.java

> ③ SuperObject.java

★ tile

> ② Tile.java

> ① TileManager.java
                                                                                                      int entityLeftCol = entityLeftWorldX/gp.tileSize;
int entityRightCol = entityRightWorldX/gp.tileSize;
int entityTopRow = entityTopWorldY/gp.tileSize;
int entitySottomRow = entitySottomWorldY/gp.tileSize;
                                                                                                     int tileNuml, tileNum2;
                                                                                                           case "up":
    entityTopRow = (entityTopWorldY - entity.speed)/gp.tileSize;
    tileNuml = gp.tileN.mapTileNum[entityLeftCol][entityTopRow];
    tileNum2 = gp.tileN.mapTileNum[entityReftCol][entityTopRow];
    if(gp.tileN.tile[tileNum1].collision == true || gp.tileN.tile[tileNum2].collision == true) {
        entity.collisionOn = true;
    }
}
           ∨ ⊕ maps
                                                                                                     switch(entity.direction) {
case "up":
           maps
map01.txt
world01.txt
world01.txt
clip objects
circle.png
mpty.png
full.png
half.png
          ∨ ⊕ player
                                                                                                                      break;
                     a bads.png
          bads.png
diver.png
goods.png
tiles
bads.png
border.png
borderleft.png
                                                                                                                                                                                                                                                                                                                  <terminated> Main [Java Application] C:\Program Files\Java\jdk-20\bin\javaw.exe (Oct 19, 2023, 5:23:51 PM – 5:24:15 PM) [pid: 13200]
                     borderright.png
                     a circle.pna
                    goods.png
playbg.png
sand.png
win.png
  | Import java.awt.image.BufferedImage;
                                                                                                                                                                                                                                                                                                                                                                                                                 - -
                                                                                                                                                                                                                                                                                                                                                                                                                                 5 6 import Main.GamePanel;
                     0
                                                                                  8 public class Entity {
                    Min

) AssestSetter,java

) CollisionChecker,java

) Eventtlandler,java

) GamePanel,java

) KeyHandler,java

) Main,java
                                                                         GamePanel gp;
public int worldX, worldY;
public int speed;
                                                                                                 public BufferedImage diver,baddy,goody;
public String direction;
                    > III Ul.java
                        UtilityTool.java
              Object

OBJ_Heart.java

OBJ_Key.java

SuperObject.java
                                                                                                public Rectangle solidArea;
public boolean collisionOn = false;
               public int solidAreaDefaultX, solidAreaDefaultY;
                                                                                                   //CHARACTER STATUS
                                                                                                  public int maxLife;
public int life;

→ ⊕ maps

               map01.txt
map01.txt
world01.txt
objects
circle.png
                                                                                                public Entity(GamePanel gp) {
    this.gp = gp;
                        empty.pngfull.pnghalf.png
              ∨ ⊕ player
                        a bads.png
               bads.png
diver.png
goods.png
titles
bads.png
border.png
borderleft.png
circle.png
                                                                                                                                                                                                                                                                                                                     Chancle X

(terminated Main [ava Application] C\Program Files\Java\jdk-20\bin\javaw.exe (Oct 19, 2023, 5:23:51 PM - 5:24:15 PM) [pid: 13200]

FFS : 60

FPS : 60
                        circle.png
```

In this you can see the relationship between these two classes Which sharing the same WorldX and Y.

Inheritance

```
> Mygame
> m MF System Library [JavaSE-12]
> m MF Eystem Library [JavaSE-12]
> m src

# import java.in.genenel:
6 import Main, GamePanel:
6 import Main, GamePanel:
                                                                                                               import javax.imageio.ImageIO;
import Main.GamePanel;
public class OBJ_Heart extends SuperObject {

| Main
| AssestSetter.java
| CollisionChecker.java
| EventHandler.java
| GamePanel.java
| KeyHandler.java
| Main.java
| Main.java
                                                                                                                                    GamePanel gp;
                                                                                                      Ul.java
                            > Dobject
> Dobj
                                                                                                                                      image1 = ImageIO.read(getClass().getResourceAsStream("/objects/full.png"));
image2 = ImageIO.read(getClass().getResourceAsStream("/objects/empty.png"));
image3 = ImageIO.read(getClass().getResourceAsStream("/objects/half.png"));
image1 = uTool.scaleImage(image1.gp.tileSize,gp.tileSize);
image2 = uTool.scaleImage(image2,gp.tileSize,gp.tileSize);
image3 = uTool.scaleImage(image3,gp.tileSize,gp.tileSize);
                   tile
Tile.java
Tile.java
                                                                                                                                                 pcatch(IOException e) {
    e.printStackTrace();
               maps
map01.txt
world01.txt
world01.txt
circle.png
mpty.png
full.png
half.png

ping
player
bads.png
diver.png
goods.png
titles
bads.png
border.png
borderfett.png
borderfett.png
borderiderd
player
player
player
player
player
player
player
player

    playbg.png
    sand.png
    win.png

                                                                                                                                                                                                                                                                                                                                                               Smart Insert 17:14:259
```

In this picture you will see class OBJ_Heart extends SuperObject. Which contains image and variables and methods And it can use everything in SuperObject by extends it. And in this case it extends SuperObject which contains images that will being used in this OBJ_Heart class.

```
eclipse-workspace - Mygame/src/Object/OBJ_Heart.java - Eclipse IDE
lorer X 🖳 🗖 🖟 Main,java 🖟 GamePanel,java 🖟 EventHandle... 🖟 Tile,java () *TileManage... / CollisionCh... / KeyHandler,java () OBJ_Heart,java X 🛂 🚼 Sackage Object;
 □ Package Explorer ×
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   ✓ 

Mygame

Mgame

Mgam
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   >  Entity.java
>  Player.java

>  Main
                                                                                                   AssestSetter.java
                        >  CollisionChecker.iava
                            EventHandler.java
GamePanel.java
KeyHandler.java
                        > 🄃 Ul.java
> 🗓 UtilityTool.java
                    Object

DBJ_Heart.java

DBJ_Key.java

DSJ_Key.java

DSJ_Key.java

UBJ_Key.java
                       > 1 Tile.java
> 1 TileManager.java
                 ✓ ∰ maps
                              map01.txt
              world01.txt

objects
circle.png
empty.png
full.png
half.png

→ 

⊕ player

⊚ bads.png

                              diver.pn
               <terminated> Main [Java Application] C:\Program Files\Java\jdk-20\bin\javaw.exe (Oct 19, 2023, 5:23:51 PM – 5:24:15 PM) [pid: 13200] FES: 60
                              circle.png
                              a goods.png
                              playbg.png
sand.png
win.png
```

Polymorphism:

```
orer X
✓ ∰ Mygame

> ➡ NRE System Library [JavaSE-12]

✓ ∰ src

✓ ∰ entity
                                                                 }catch(Exception e) {
                                                                                                                                                                                                                                                              <u>...</u>
                                                          public void draw(Graphics2D g2 ) {
         > 🗓 Entity.java
> 🗓 Player.java
                                                                                                                                                                                                                                                             (a)
         while(worldCol < gp.maxWorldCol && worldRow < gp.maxWorldRow) {</pre>
                                                                     int tileNum = mapTileNum[worldCol][worldRow];
                                                                  int worldX = worldCol * gp.tileSize;
int worldY = worldRow * gp.tileSize;
int screenX = worldX - gp.player.worldX + gp.player.screenX;
int screenY = worldY - gp.player.worldY + gp.player.screenY;
           UtilityTool.iava
      Object

OBJ_Heart.java

OBJ_Key.java

SuperObject.java
                                                                   if(worldX + gp.tileSize > gp.player.worldX - gp.player.screenX &&
worldX - gp.tileSize < gp.player.worldX + gp.player.screenX &&
worldY + gp.tileSize > gp.player.worldY - gp.player.screenX &&
worldY - gp.tileSize < gp.player.worldY + gp.player.screenY (gp.player.screenY &&</pre>
         > 1 Tile.java
> 1 TileManager.java
                                                                             g2.drawImage(tile[tileNum].image, screenX, screenY, gp.tileSize,gp.tileSize,null);
      maps
map01.txt
map01.txt
morld01.txt
                                                                      worldCol++;
                                                                   if(worldCol == gp.maxWorldCol) {
   worldCol = 0;
                                               125
126
127
128
                                                                         worldRow++;
            half.png
      129
                                           Console X

<terminated> Main [Java Application] C\Program Files\Java\jdk-20\bin\javaw.exe (Oct 19, 2023, 5:23:51 PM – 5:24:15 PM) [pid: 13200]

FPS : 60

FPS : 60
      a borderright.png

    borderright,
    circle.png
    goods.png
    playbg.png
    sand.png
    win.png

                                                                                                                                                                              35:1:588
```

```
> MM MRE System Library [lavaSE-12]
> MM RE System Library [lavaSE-12]

> MB rec | 4 import java.awt.Rectangle;

4 import java.awt.mage.BufferedImage;

5 mport Main.GamePanel;

8 import Main.UtilityTool;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     .
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   0
                       Main

AssestSetter,java

CollisionChecker,java

EventHandler,java

EventHandler,java

Main,java

Miain,java

Ultijava

Ultijava

Ultijava

Object

Object

Object

SoperObject,java

SoperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java

"BosperObject,java
                                                                                                                                  public BufferedImage image, image1, image2, image3, baddy ,goody;
                                                                                                                                                                         public burreturnaye maye, mayer, mayer, mayer, mayer, public bring name;
public boolean collision = false;
public int worldX;
public Rectangle solidArea = new Rectangle(0,0,48,48);
public int solidAreaDefaultX = 0;
public int solidAreaDefaultY = 0;
UtilityTool uTool = new UtilityTool();
                                                                                                                                                                        public void draw(Graphics2D g2, GamePanel gp) {

→ ∰ tile
→ ☑ Tile.java
→ ☑ TileManager.java
                                                                                                                                                                              if(worldX + gp.tileSize > gp.player.worldX - gp.player.screenX &&
worldX - gp.tileSize < gp.player.worldX + gp.player.screenX &&
worldY + gp.tileSize > gp.player.worldY - gp.player.screenY &&
worldY - gp.tileSize < gp.player.worldY + gp.player.screenY) {</pre>
                  # res

| maps
| map01.txt
| world01.txt

| world01.txt

| in world
                                                                                                                                                                                                          g2.drawImage(image, screenX, screenY, gp.tileSize,gp.tileSize,null);
                   → player

→ bads.png

→ diver.png

→ goods.png
                                                                                                                                  Console ×

**Leminated- Main [Java Application] C\Program Files\Dava\jdk-20\bin\javaw.exe (Oct 19, 2023, 5:23:51 PM - 5:24:15 PM) [pid: 13200]

FPS : 60

FPS : 60
                                                                                                                                                       4
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   bads.png
border.png
borderleft.png
borderlight.png
circle.png
goods.png
playbg.png
sand.png
win.png
                                                                                                                                                                                                                                                                                                                                                                            Writable Smart Insert 12:55:250
```

For this Draw in polymorphism, you can see that draw is being used in the different classes and different parameters for different result.



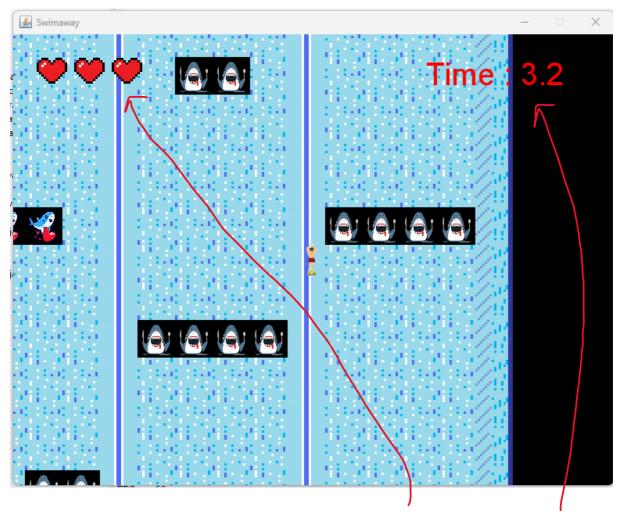
In my game's menu, it has 3 components are 2 buttons and 1 String

Event Handling: using "w" and "s" to move the cursor up and down.

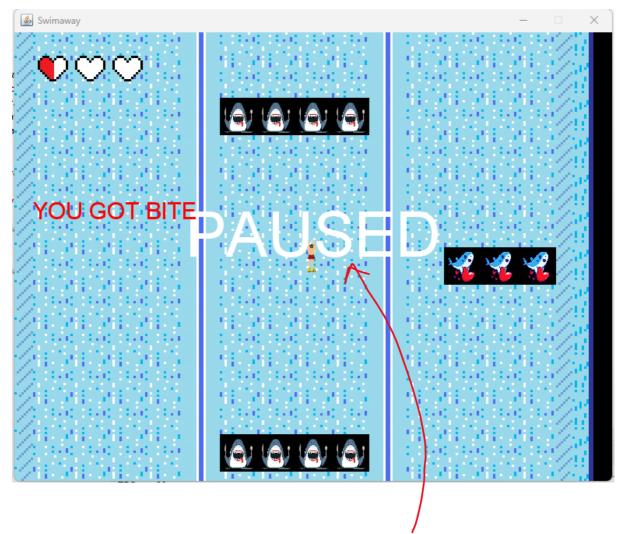


In my game's How to play, it has 3 components are 2 buttons and 1 String

Event Handling: using "w" and "s" to move the cursor up and down.

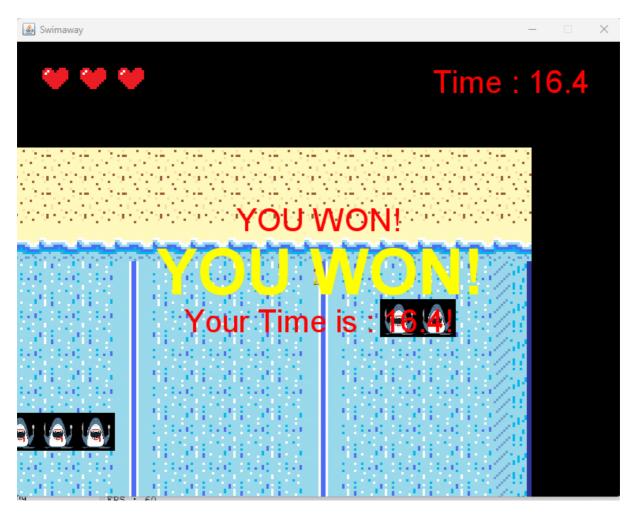


In game-play, it has 2 components are 3 images and 1 String Event Handling: using "w", "a", "d" to move the character forward, right and left.



In pause screen, it has 1 component is 1 String Event Handling: using "p" to resume the pause and play.

And In won,gameover screen, it has 1 component is 1 String Event Handling: no event handling.





The main Algorithm: in this game it will check the heart of player. If it's 0 that's mean game over but if no, it continues until player went into goal. But if you collect the heart after hitting shark, it will increase health of the player to continues also.

Summary

Problems: At first, I want to create the game that can only move 3 columns but for the game, it couldn't do that because of the tiles and map which I want to create the bigger map so I desired to use freely move instead. So player can move everywhere in the map except moving backward. And in the game it's still has some bug when player moves sometime the hit detection won't work and I tried to fix it but It seems nothing worked.

Program highlight: the character player or diver and move around the map freely which gives player freedom to move and have many playstyle to use.

Advice to Prof.: need more time to learn about the game coding and web coding but for now I did it not that good so I want to practice more.