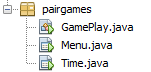
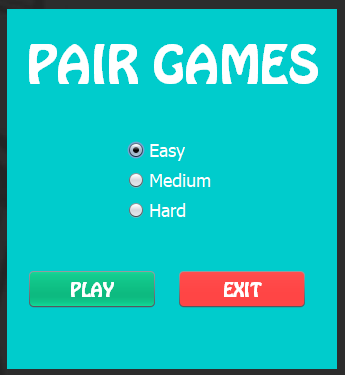
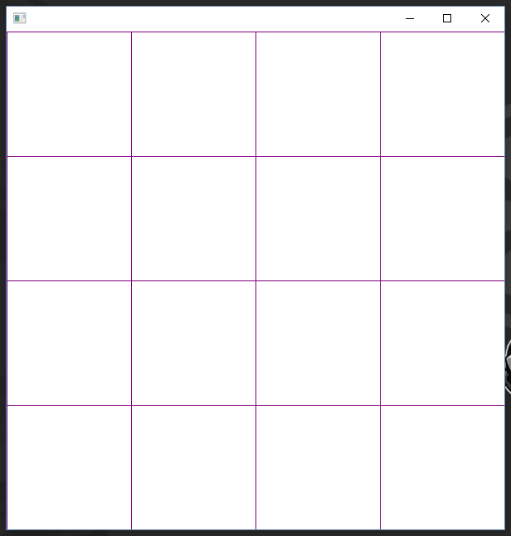
1. **MODUL 1 CLASS DAN OBJECT, SERTA PERANCANGAN BERBASIS OBJEK**

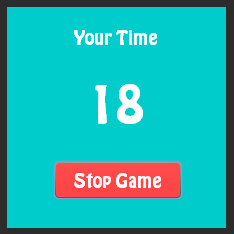
GAMBAR : CLASS DAN OBJECT



GAMBAR : PERANCANGAN BERBASIS OBJEK







1. **MODUL2 CONSTRUCTOR DAN DESTRUCTOR**

SOURCE CODE :

public Time(int timePlay) {

initComponents();

mTextViewWaktu.setText(String.valueOf(timePlay));

timer = new Timer ();

timer.schedule (new CountDownTime(),0,1000);

}

public Time() {

}

1. **MODUL 3 ARRAY OBJECT**

SOURCE CODE :

List<Tile> tiles = new ArrayList<Tile>();

for (int i = 0; i < NUM\_OF\_PAIRS; i++) {

tiles.add(new Tile(String.valueOf(c)));

tiles.add(new Tile(String.valueOf(c)));

c++;

}

Collections.shuffle(tiles);

1. **MODUL 4 PEWARISAN (INHERITANCE)**

SOURCE CODE :

public boolean cancel() {

closeTime();

return super.cancel();

}

1. **MODUL 5 OVERLOADING FUNGSI**

SOURCE CODE :

@Override

public void run() {

if(detik > 0) {

mTextViewWaktu.setText(String.valueOf(detik));

detik --;

} else if (detik == 0) {

cancel();

}

}

1. **MODUL 6 VIRTUAL FUNCTION**

SOURCE CODE :

public void getLevel(){

if (mRadioEasy.isSelected()){

timePlay = 60;

NUM\_OF\_PAIRS = 8;

NUM\_PER\_ROW = 4;

} else if (mRadioMedium.isSelected()){

timePlay = 30;

NUM\_OF\_PAIRS = 10;

NUM\_PER\_ROW = 5;

} else if (mRadioHard.isSelected()){

timePlay = 10;

NUM\_OF\_PAIRS = 10;

NUM\_PER\_ROW = 5;

}

}

1. **MODUL 7 ABSTRACT CLASS DAN INTERFACE JAVA**

SOURCE CODE (ABSTRACT CLASS) :

public class Menu extends javax.swing.JFrame {