

Agate Mobile Game Developer Camp

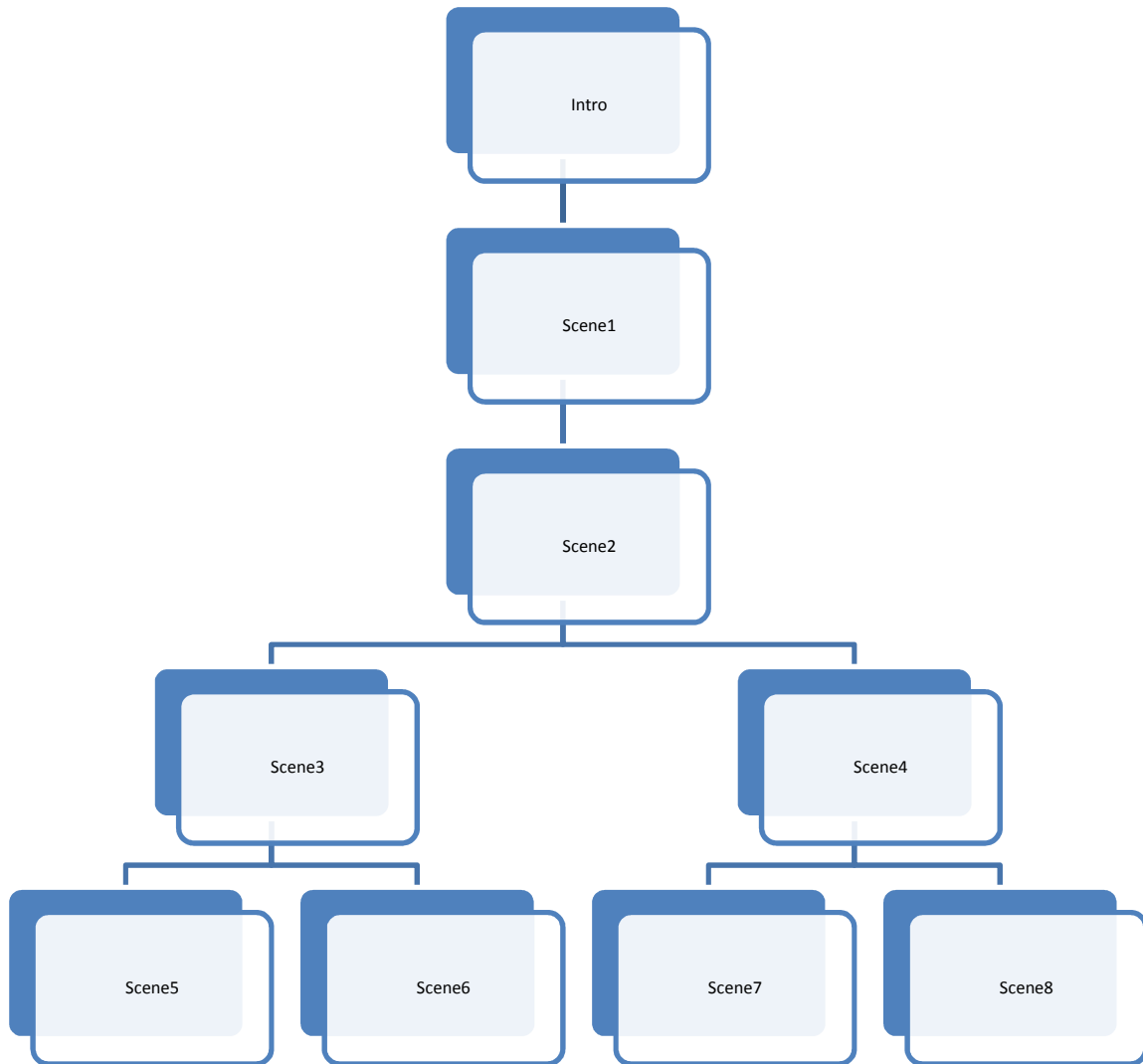
Programmer Handout – Day #4

Pada kesempatan ini, kita akan mempelajari tentang state management. State management biasanya digunakan ketika kita membuat menu awal pada sebuah game. State management menggunakan switch-case agar kita bisa memilih state. Pada dasarnya, membuat menu game membutuhkan 4 state, yaitu Start, Tutorial, About, dan Exit. Dengan adanya state tersebut, maka kita harus membuat switch-case. Contohnya seperti ini:

```
Switch (state) {
case 0: //masuk ke menu utama
    drawMenu();
    break;
case 1: //ketika di menu utama kita memilih start game, kita masuk dulu di level selection
    drawStartGame(); //ketika memilih level 1, maka set state menjadi 5
    break;
case 2: //perlihatkan tutorial yang ada
    drawTutorial();
    break;
case 3: //perlihatkan about
    drawAbout();
    break;
case 4: //exit dari permainan
    exitGame();
    break;
case 5: //draw canvas untuk lvl 1
    level1(); //ketika char kita mati, maka set state menjadi 6
    break;
case 6: //animasi ketika kita mati
    animasiDeath(); //meledak atau apapun
    showScore(); //ketika ditekan tombol, maka kembali lagi ke level selection, set state menjadi 1
    break;
```

switch-case tersebut ditaruh di gameloop, sehingga game kita akan terus mengecek state kita. Kapan state kita berubah? Jawabnya adalah ketika kita melakukan input. Switch – case di atas menunjukkan alur kasar sebuah game.

Sekarang mari kita mencoba membuat visual novel yang simple, game ini hanya memilih kanan atau kiri. Gambar berikut ini adalah urutan scene dari visual novel kita.



Asset untuk membuat visual novel di atas sudah disediakan. Pertama-tama kita buat gameMIDlet.java, kita bisa gunakan midlet kita yang sebelumnya. Untuk VnCanvas.java, mari kita buat seperti ini

```
public class VnCanvas extends GameCanvas implements Runnable{
```

Lalu deklarasikan variable global.

```
private Graphics g;
private int inputDelay = 0;

// Deklarasikan gambar
private Image gambarIntro, playHover;
private Image scene1, scene2, scene3, scene4, scene5, scene6, scene7, scene8;

// Deklarasikan scene awal
private int scene = 0;

//Deklarasikan Scene sebagai static INTEGER
private static final int Intro = 0;
private static final int Scene1 = 1;
private static final int Scene2 = 2;
private static final int Scene3 = 3;
private static final int Scene4 = 4;
private static final int Scene5 = 5;
private static final int Scene6 = 6;
private static final int Scene7 = 7;
private static final int Scene8 = 8;
```

Buat kontruktor

```
protected VnCanvas() {
    super(false);
    this.setFullScreenMode(true);
    g = getGraphics();
}
```

Jangan lupa juga membuat fungsi start(), untuk menjalankan Thread.

```
public void start() {
    Thread runner = new Thread(this);
    runner.start();
}
```

Sekarang kita buat fungsi init(), untuk menginisialisasikan gambarnya.

```
public void init() throws IOException {
    gambarIntro = Image.createImage("/Intro.jpg");
    playHover = Image.createImage("/button_play_hover.png");

    //Inisialisasi gambar
    scene1 = Image.createImage("/Scene1.jpg");
    scene2 = Image.createImage("/Scene2.jpg");
    scene3 = Image.createImage("/Scene3.jpg");
    scene4 = Image.createImage("/Scene4.jpg");
    scene5 = Image.createImage("/Scene5.jpg");
    scene6 = Image.createImage("/Scene6.jpg");
    scene7 = Image.createImage("/Scene7.jpg");
    scene8 = Image.createImage("/Scene8.jpg");
}
```

Lalu kita buat pilihMenu(), ini berfungsi untuk mengubah state, sehingga permainan ini akan terlihat alurnya. Input dari game ini hanya ada 3, fire, kiri, dan kanan. Tentu saja penggunaan input untuk beberapa scene akan sama. Maka dari itu kita mengecek scene terlebih dahulu, untuk memeriksa sedang dimana scene kita dan menentukan apa yang akan terjadi ketika kita menekan kiri jika kita berada di scene 3 (misal).

```
public void pilihMenu() {
    int keyState = getKeyStates();
    if (++inputDelay > 4) {
        if ((keyState & FIRE_PRESSED) != 0) {
            if (scene == Intro)
                scene = Scene1;
            else if (scene == Scene1)
                scene = Scene2;
            else if (scene == Scene5)
                scene = Intro;
            else if (scene == Scene6)
                scene = Intro;
            else if (scene == Scene7)
                scene = Intro;
            else if (scene == Scene8)
                scene = Intro;
        }
        if ((keyState & LEFT_PRESSED) != 0) {
            if (scene == Scene2)
                scene = Scene4;
            else if (scene == Scene3)
                scene = Scene7;
            else if (scene == Scene4)
                scene = Scene8;
        }
        if ((keyState & RIGHT_PRESSED) != 0) {
            if (scene == Scene2)
                scene = Scene3;
            else if (scene == Scene3)
                scene = Scene5;
            else if (scene == Scene4)
                scene = Scene6;
        }
        inputDelay = 0;
    }
}
```

Lalu kita buat kelas untuk menggambar masing – masing frame.

```
public void intro() {
    g.drawImage(gambarIntro, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
    g.drawImage(playHover, getWidth()/2, getHeight()-50, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene1() {
    g.drawImage(scene1, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene2() {
    g.drawImage(scene2, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene3() {
    g.drawImage(scene3, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene4() {
    g.drawImage(scene4, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene5() {
    g.drawImage(scene5, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene6() {
    g.drawImage(scene6, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene7() {
    g.drawImage(scene7, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}

public void Scene8() {
    g.drawImage(scene8, getWidth()/2, getHeight()/2, Graphics.HCENTER | Graphics.VCENTER);
}
```

Terakhir tentu saja kita harus membuat game loop. Disinilah terletak switch – case kita, switch – case kita taruh di dalam game loop, sehingga akan terus mengulang. Pertama – tama, switch akan mengecek **scene** bernilai berapa. Karena nilai awal scene kita adalah 0, maka program akan langsung menuju ke case 0. Didalam case 0, ada fungsi intro() yang menggambar menu awal dan fungsi pilihMenu() untuk mengecek input apa yang akan kita lakukan.

```
public void run() {
    try {
        //Panggil init() untuk menginisialisasi
        init();
    } catch (IOException e1) {
        // TODO Auto-generated catch block
        e1.printStackTrace();
    }
    while (true) {

        switch (scene) {
            case 0:
                intro();
                pilihMenu();
                break;

            case 1:
                Scene1();
                pilihMenu();
                break;

            case 2:
                Scene2();
                pilihMenu();
                break;

            case 3:
                Scene3();
                pilihMenu();
                break;

            case 4:
                Scene4();
                pilihMenu();
                break;

            case 5:
                Scene5();
                pilihMenu();
                break;

            case 6:
                Scene6();
                pilihMenu();
                break;

            case 7:
                Scene7();
                pilihMenu();
                break;

            case 8:
                Scene8();
                pilihMenu();
                break;
        }

        flushGraphics();
        try {
            Thread.sleep(30);
        } catch (InterruptedException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

Tantangan.

Buat menu yang terdiri dari Start Game, Tutorial, About, dan Exit Game. Implementasikan pada game terdahulu yang telah kita buat.