

Bab 6

Layer Manager

6.1 Tujuan

Setelah mempelajari bab ini, Pelajar diharapkan mampu untuk :

- Memahami dan menggunakan class LayerManager
- Menggambar beberapa sprite dan di satukan pada LayerManager
- Menentukan posisi titik awal LayerManager pada screen mobile
- Mengatur besar screen yang di tampilkan pada mobile
- Menggambar animasi pada LayerManager

6.2 Pengenalan

Pada bab sebelumnya, kita telah membahas class sprite, tentu saja kita dengan mudah meletakkan lebih dari sebuah sprite pada layar tetapi bagaimana dengan pemandangan. Pada hakekatnya semua game memiliki background yang dianimasikan atau tetap, untuk memberikan efek visual. Disinilah awal mulanya LayerManager, sesuai namanya, yaitu mengelola layer grafis.

6.3 Penggunaan LayerManager

LayerManager adalah subClass dari GameCanvas

Untuk menambahkan sebuah layer, method `append(Layer layer)` telah tersedia

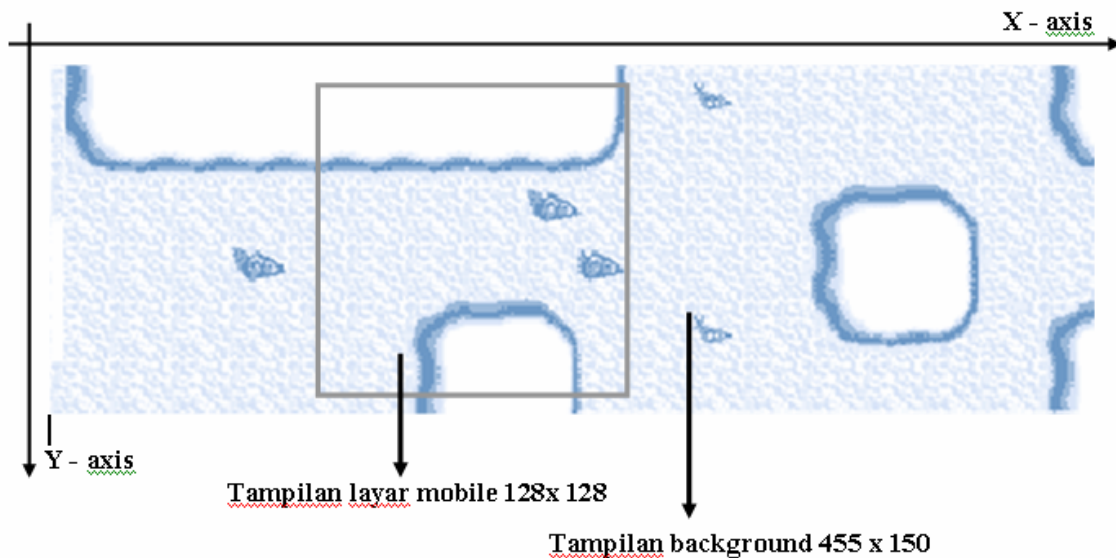
```
LayerManager layer = new LayerManager();
```

```
layerManager.append(layer);
```

Untuk mendapatkan sebuah layer dengan indeks yang spesifik, menggunakan method `getLayerAt(int index)`. Untuk memperoleh jumlah total layer pada `LayerManager` gunakan method `getSize()`. Untuk menghapus sebuah layer gunakan method `remove(Layer layer)`. Untuk menambahkan sebuah layer pada indeks yang khusus gunakan method `insert(Layer layer, int index)` yang akan memungkinkan Anda untuk melakukan hal tersebut. Untuk menampilkan layer panggil method `paint(Graphics g, int x, int y)`.

6.3.1 Menggambar di `LayerManger`

Pada sesi sebelumnya method yang disebutkan cukup jelas dan Peserta benar-benar dapat mengimplementasikannya tanpa menggunakan `LayerManager`. Walaupun demikian, ada satu method yang menyediakan fitur yang tepat, yaitu method `setViewWindow(int x, int y, int width, int height)`. Fitur ini memungkinkan Anda untuk meng-set ukuran jendela yang dapat dilihat oleh user dan bagian dari layer yang ditampilkan. untuk lebih mudah dimengerti dengan melihat gambar berikut ini.



Gambar 1: Sistem `LayerManager`

Untuk mengeset tampilan menggunakan method `setViewWindow`. Method yang diimplementasikan pada diagram diatas adalah `setViewWindow(90,10,128, 128)`. Dua pertama menunjukkan letak gambar dari

sudut kiri atas dengan koordinat pada (90,10). Dua angka kedua 128 x 128 menunjukkan ukuran lebar dan tinggi yang sekarang digunakan.

GameCanvas menggunakan LayerManager:

```
.....
public class ExampleGameCanvas extends GameCanvas implements Runnable {
    .....
    // Penggunaan Sprites
    private Sprite playerSprite;
    private Sprite backgroundSprite;

    // Layer Manager
    private LayerManager layerManager;

    // Konstruktor dan inisialisasi
    public ExampleGameCanvas() throws Exception {
        .....
        // Memuat gambar untuk Sprite
        Image playerImage = Image.createImage("/pesawat.png");
        playerSprite = new Sprite (playerImage,40,40);

        Image backgroundImage = Image.createImage("/background.png");
        backgroundSprite = new Sprite(backgroundImage);

        layerManager = new LayerManager();
        layerManager.append(playerSprite);
        layerManager.append(backgroundSprite);
    }

    // Method untuk menampilkan grafis
    private void drawScreen(Graphics g) {
        .....
        // menampilkan semua layer
        layerManager.paint(g,0,0);
        flushGraphics();
    }
}
```

6.3.2 "LayerManager sederhana"



Gambar 2: Sprite di LayerManager

LayerManager menggunakan method setViewWindow

Pada demonstrasi penggunaan setViewWindow sebelumnya, berikut ini adalah potongan kodenya (timpalah method drawScreen).

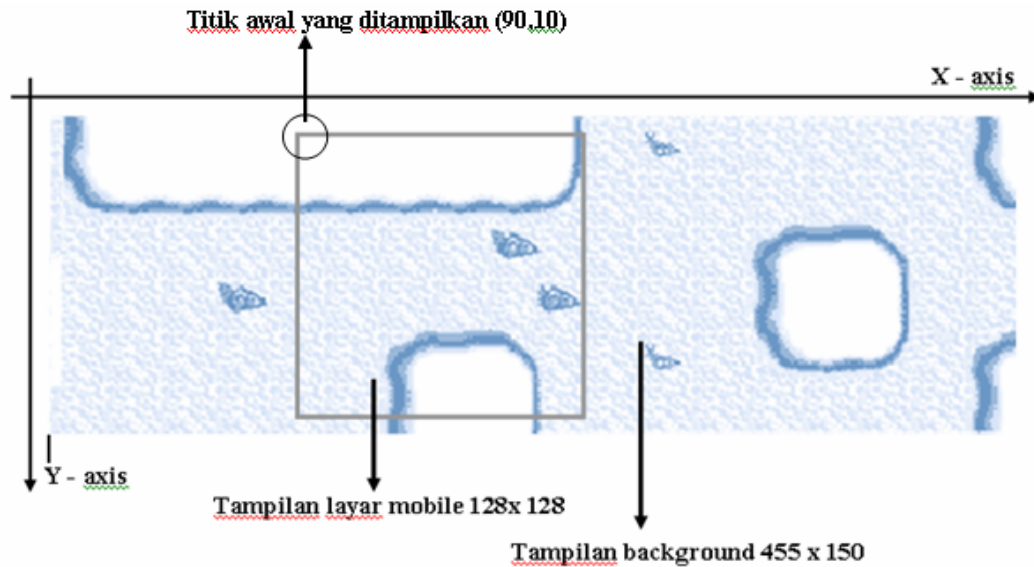
```
// Method untuk menampilkan grafis
private void drawScreen(Graphics g) {
    g.setColor(0xffffffff);
    g.fillRect(0, 0, getWidth(), getHeight());
    g.setColor(0x0000ff);

    // update posisi sprite
    playerSprite.setPosition(currentX,currentY);

    // menampilkan semua layer
    layerManager.setViewWindow(55,20,140,140);
    layerManager.paint(g,20,20);
    flushGraphics();
}
```

Yang perlu diperhatikan bahwa penggunaan metod setViewWindow pada tampilan area mobile akan pada posisi ditengah layar. Jika Anda ingin mengatur posisi area ini, Anda perlu meletakkan nilai tepat ke dalam metod paint yang pada yaitu untuk meletakkan di tengah layar nilai yang digunakan adalah 20 pixel pada sumbu x dan 20 pixel pada sumbu y. Nilai ini akan mengimbangi layar itu sendiri. Contohnya pada gambar 3.

6.3.3 Animasi pada LayerManager



Gambar 3: letak titik awal di LayerManager

Anda mungkin telah mengetahui bahwa gambar background yang telah kita gunakan lebih besar dari tampilan layar itu sendiri.

Pada contoh ini, dimulai pada titik (90,10). Semua yang telah Anda lakukan adalah membuat nilai 90 menjadi sebuah nilai dinamis sehingga dapat diturunkan jika Anda ingin menggeser latar belakang ke kiri atau menaikkan nilainya jika Anda ingin menggeser latar belakang ke kanan.

Kode berikut ini mengilustrasikan bagaimana hal ini dapat dilakukan, untuk lebih mudahnya sprite smiley hijau telah dihilangkan dan sekarang jika Anda menekan tombol kanan game akan menggeser ke kanan dan jika Anda menekan tombol kiri game maka gambar akan bergeser ke kiri.

Sebagai catatan, hanya nilai horizontal yang dibuat dinamis, tidak akan terjadi apa-apa jika Anda menekan tombol ke atas dan ke bawah.

Animasi pada LayerManager

```
.....
public class ExampleGameCanvas extends GameCanvas implements Runnable {
    .....
    Image backgroundImage;
    private Sprite backgroundSprite;

    private LayerManager layerManager;

    // Metode untuk menangani masukan user
    private void input() {
        int keyStates = getKeyStates();
        if ((keyStates & RIGHT_PRESSED) != 0) {
            scnX = Math.min(getWidth(), scnX + 1);
        }
        if ((keyStates & LEFT_PRESSED) != 0) {
            scnX = Math.max(0, scnX - 1);
        }
    }

    // Metode untuk menampilkan grafis
    private void drawScreen(Graphics g) {
        .....
        layerManager.setViewWindow(scnX,scnY,140,140);
        layerManager.paint(g,20,20);
        flushGraphics();
    }
}
```



Gambar 6.4

Animasi background sederhana menggunakan LayerManager

Tekanlah tombol kiri atau kanan pada game untuk melihat efek animasi pergerakan background.