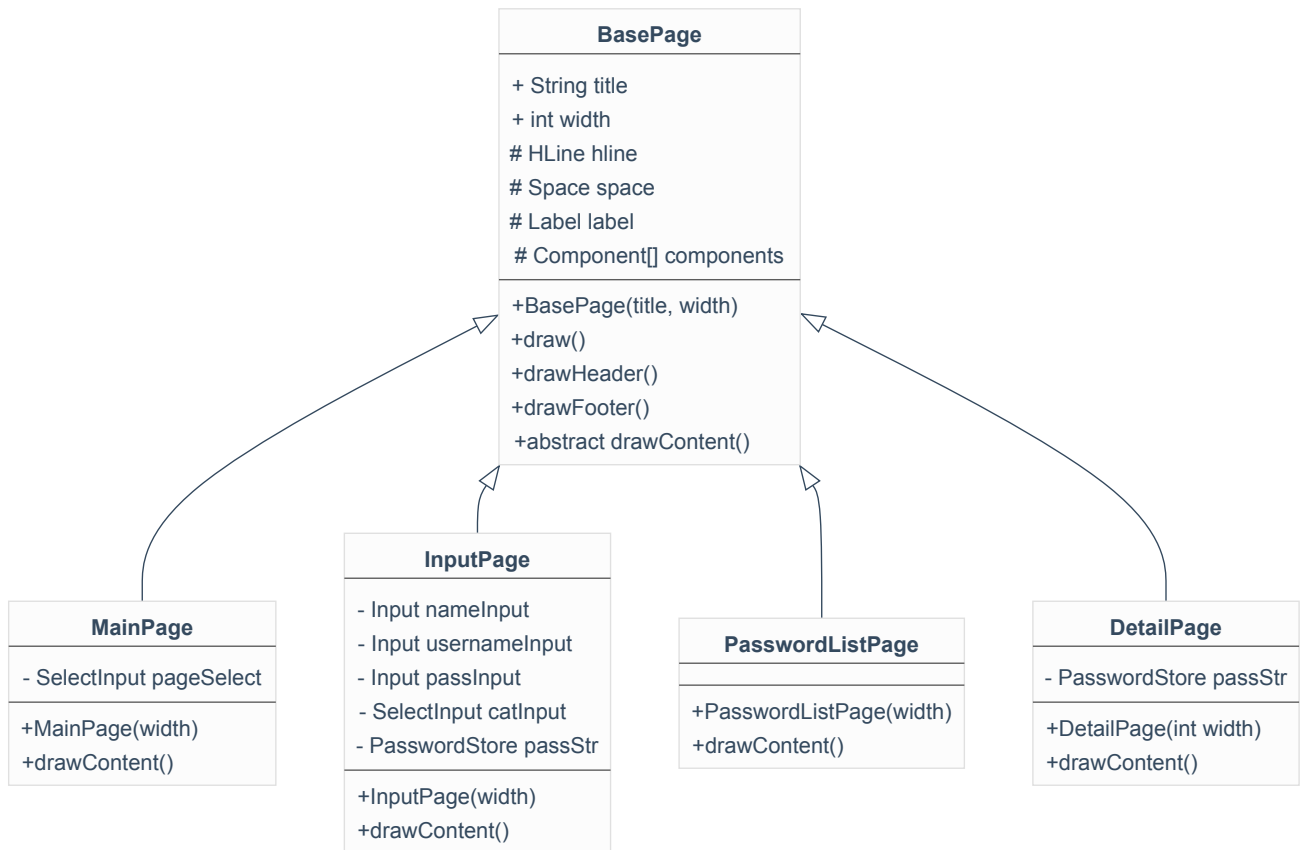


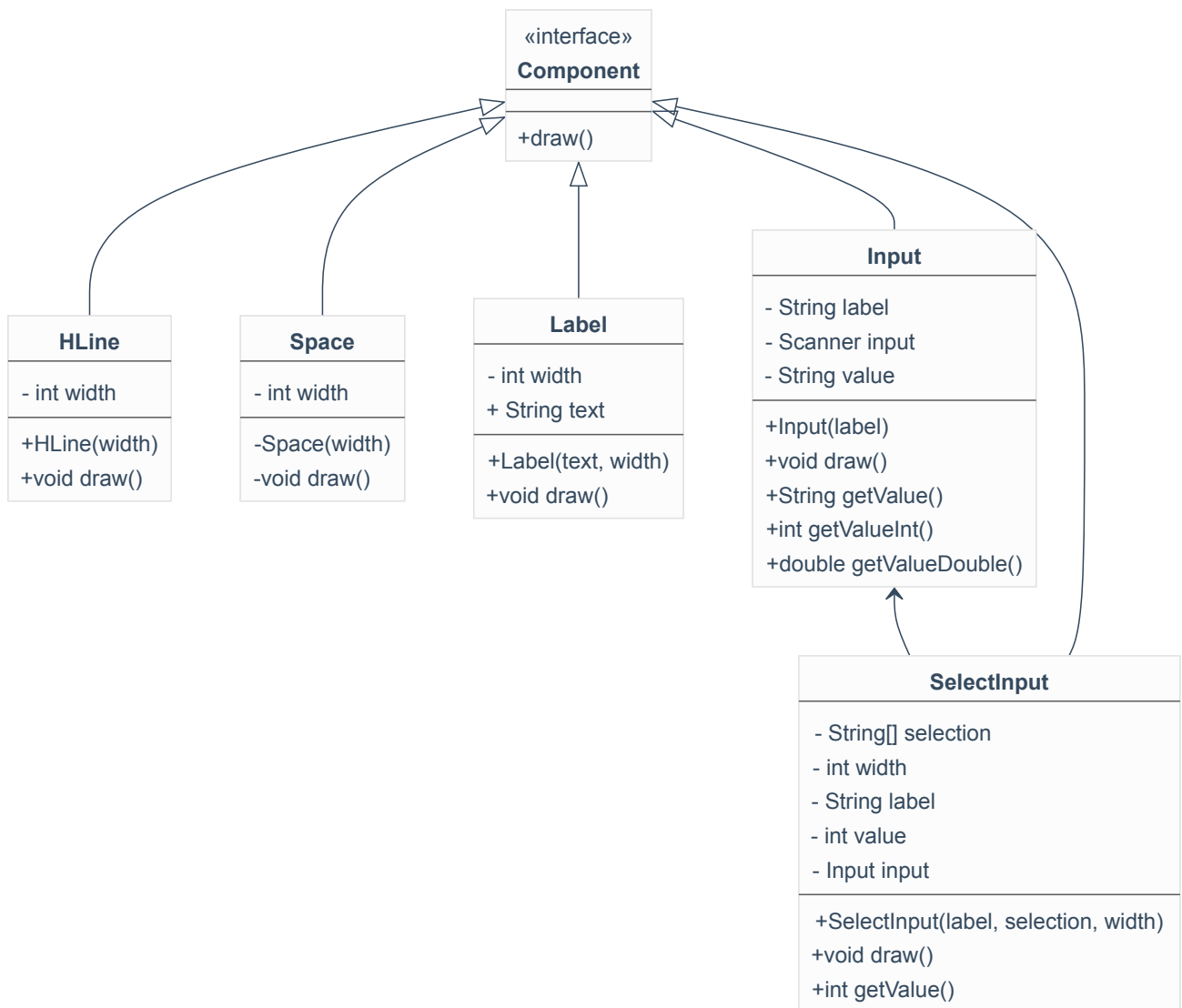
PBO - Modul Praktik 6

Pada modul ini kita akan mengimplementasikan konsep pewarisan dan polimorfisme pada studi kasus Password Store dengan memodifikasi terutama pada class UI.

Struktur class yang akan kita modifikasi adalah sebagaimana ditampilkan pada class diagram berikut:



Adapun struktur untuk component UI kita bisa menggunakan **interface** sebagai berikut:



Abstract Class BasePage

- Deklarasi atribut dan method sesuai ditunjukkan pada class diagram di atas.
- Pada atribut `components` sebaiknya menggunakan `ArrayList` untuk memudahkan manipulasi penambahan komponen UI.
- Constructor digunakan untuk inisialisasi semua atributnya menggunakan value dan instance object yang relevan.
- Method `draw()`, `drawHeader()`, dan `drawFooter()` berisi koding yang kurang lebih sama dengan koding pada page sebelumnya.
- Method `drawContent()` memiliki modifier `abstract` sehingga tidak ada implementasi tubuh fungsinya.

Interface Component

- Interface component hanya berisi satu method saja yaitu `draw()` yang nanti akan diimplementasikan pada class implement-nya.
- Pada class komponen UI lain seperti `HLine`, `Input`, `Label`, `SelectInput`, dan `Space` hanya perlu menambahkan implementasi pada bagian deklarasinya. Selebihnya tidak perlu ada perubahan.

```

public class HLine implements Component {
    // ...
}
  
```

```

public class Input implements Component {
    // ...
}

public class Label implements Component {
    // ...
}

public class SelectInput implements Component {
    // ...
}

public class Space implements Component {
    // ...
}

```

Class MainPage

Merupakan class kongkrit yang melakukan pewarisan dari class `BasePage`. Karena sudah melakukan pewarisan class tersebut, maka class ini hanya perlu menambahkan komponen yang akan digunakan pada constructor, melakukan perulangan dan menambahkan logika program pada method `drawContent()`. Berikut adalah koding lengkapnya.

```

public class MainPage extends BasePage{

    SelectInput pageSelect;

    public MainPage(int width){
        super("Aplikasi Penyimpanan Password", width);
        this.components.add(new Label("Selamat datang di aplikasi Password Vault",
this.width));
        this.components.add(new Label("Simpan password anda dengan aman di sini",
this.width));
        this.components.add(new Space(this.width));

        String [] pages = {"Input Password", "Tampil Password", "Keluar Aplikasi"};
        this.pageSelect = new SelectInput("Pilih halaman berikut:",
                                pages, this.width);
        this.components.add(pageSelect);
    }

    @Override
    protected void drawContent(){

        int select;

        for(Drawable widget: this.components){
            widget.draw();
        }

        select = this.pageSelect.getValue() - 1;
        switch (select) {

```

```

        case 0 -> {
            drawFooter();
            new InputPage(this.width).draw();
        }
        case 1 -> {
            drawFooter();
            new PasswordListPage(this.width).draw();
        }
        case 2 -> {
            new Label("Menyimpan data ... ..", this.width).draw();
            DataPassword.saveCSVData();
            new Label("Terima kasih telah menggunakan aplikasi", this.width).draw();
            drawFooter();
            System.exit(0);
        }
        default -> {
            new MainPage(this.width).draw();
        }
    }
}
}
}

```

Implementasi Class Page Lainnya

Implementasikan konsep pewarisan pada class page lainnya dengan mengikuti langkah berikut:

- Extends class `BasePage`
- Hapus semua method yang sudah tidak diperlukan lagi.
- Tambahkan semua komponen UI yang diperlukan pada page tersebut.
- Implementasikan method `drawContent` dengan melakukan loop pada semua komponen dan menambahkan logic pada koding yang diperlukan.
- Lakukan hal tersebut pada `InputPage`, `PasswordListPage`, dan `DetailPage`
- Pastikan program masih bisa berjalan seperti biasa.