

Custom Component Geliştirme

CustomComponent ile Bileşen Geliştirme

- Vaadin, mevcut UI bileşenlerini kullanarak yeni bileşenler oluşturmaya imkan tanır
- Bunun için yapılması gereken;
 - **CustomComponent'i extend** eden bir sınıf oluşturmak
 - Bu sınıfın constructor'ı içerisinde view'ı oluşturup, **setCompositionRoot()** metoduyla bu view'ı kök bileşen olarak tanımlamak
- Genellikle kök bileşen bir **ComponentContainer** implementasyonudur

CustomComponent

```
public class AddressComponent extends CustomComponent {
```

```
    private TextArea address;  
    private ComboBox city;  
    ...
```

```
    public AddressComponent() {
```

```
        GridLayout addressLayout = new GridLayout(2, 3);  
        addressLayout.setSpacing(true);
```

```
        address = new TextArea("Address");  
        addressLayout.addComponent(address, 0, 0, 0, 2);
```

```
        ...
```

```
        setCompositionRoot(addressLayout);
```

```
    }
```

```
    //getters & setters
```

```
    ...
```

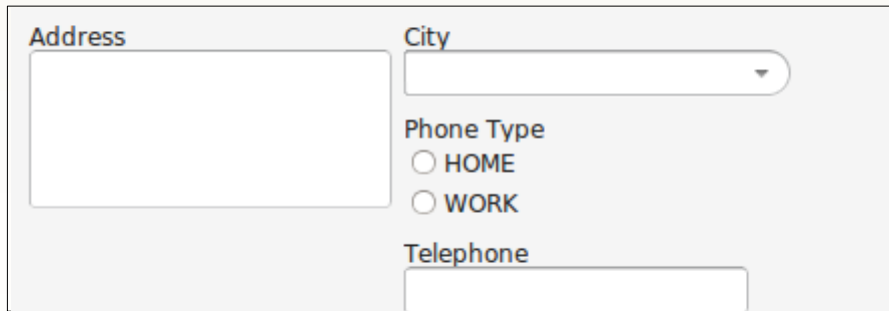
```
}
```

CustomComponent
extend edilir

Custom bileşenin kök bileşeni
mutlaka belirtilmelidir

CustomComponent Kullanımı

```
public class PetClinicUI extends UI {  
  
    protected void init(VaadinRequest request) {  
  
        VerticalLayout layout = new VerticalLayout();  
        setContent(layout);  
  
        ...  
  
        final AddressComponent addressComponent = new AddressComponent();  
        layout.addComponent(addressComponent);  
        ...  
    }  
}
```



Address

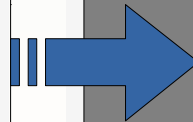
City

Phone Type

☐ HOME

☐ WORK

Telephone



First Name

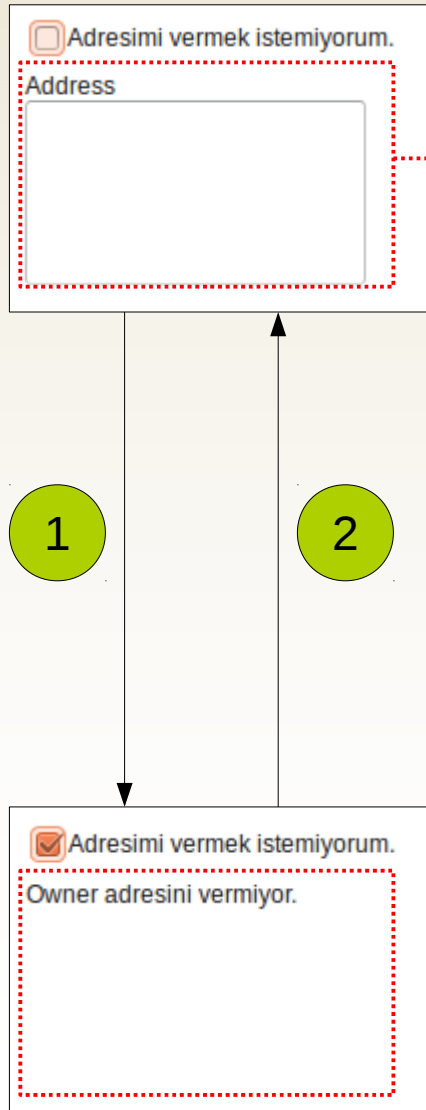
Last Name

Add

Component Replacement

- RIA tabanlı “single-page” web uygulamalarında sayfalar arası geçiş yerine sayfanın içerisindeki **belirli bölümlerin değişmesi** söz konusudur
- Bu işlem Vaadin'de **ComponentContainer** üzerinden gerçekleştirilir
- **replaceComponent(oldComp,newComp)** metodu component tree'de eski bileşen yerine yeni bileşeni yerleştirir

Component Replacement



```
final TextArea address = new TextArea("Address");

final Label label = new Label("Owner adresini vermiyor.");

final Layout addressLayout = new VerticalLayout();
addressLayout.addComponent(addressComponent);

final CheckBox check= new CheckBox("Adresimi vermek istemiyorum.");
check.addValueChangeListener(new ValueChangeListener() {

    @Override
    public void valueChange(ValueChangeEvent event) {

        if(check.getValue()) {
            addressLayout
            1      .replaceComponent(address, label);
        } else {
            addressLayout
            2      .replaceComponent(label, address);
        }
    }
});
```

- Harezmi Bilişim Çözümleri
- Kurumsal Java Eğitimleri
- <http://www.java-egitimleri.com>
- info@java-egitimleri.com

