

PV230 - cvičení 6 - postup

PV230 - Cvičení 6

jsp/basket/view_maximized.jsp

- do portletu košíku z minulého cvičenia pridáme do detailného (maximalizovaného) zobrazenia tlačidlo 'zpět', ktoré nás vráti na celú stránku.

```
<portlet:renderURL var="backURL" windowState="normal">
</portlet:renderURL>
<input type="button" onclick="location.href = '${backURL}'" value="<spring:message
code='label-back' />" />
```

- z názvu produktu urobíme link, ktorý bude smerovať naspäť na celú stránku (windowState="normal") a zobrazí v portlete detailu produktu vybraný produkt
- samostatne vytvoríme v zozname objednávaných produktov nový stĺpec, ktorý bude obsahovať link () na odstránenie danej položky (tj. <portlet:actionURL ... >) z košíku. Použijeme metódu `BasketService.removeProduct(ProductDTO product, String userId)`.
- vytvoríme odkaz na pokračovanie v objednávaní produktov

```
<c:if test="${!empty basket.items}">
<portlet:renderURL var="continueURL">
<portlet:param name="<%=PARAM_CONTROLLER%>" value="<%=CONTROLLER_ORDER%>" />
</portlet:renderURL>
<input type="button" onclick="location.href = '${continueURL}'"
value="<spring:message code='label-continue' />" />
</c:if>
```

OrderViewController.java

- do `OrderViewController` pridáme render metódu na vykreslenie nasledujúcej stránky

```
@Autowired
private OrderingService orderingService;

@RequestMapping
public String renderDefault(RenderRequest request, Model model) {
    if (!model.containsAttribute(ATTRIBUTE_CUSTOMER)) {
        String userId = request.getRemoteUser();
        CustomerDTO customer = orderingService.getCustomer(userId);
        model.addAttribute(ATTRIBUTE_CUSTOMER, customer);
    }
    return JSP_VIEW_ORDER;
}
```

- vytvoríme `jsp/basket/view_order.jsp`
- pridáme metódu, ktorá spracuje akciu objednania
- po objednaní presmerujeme užívateľa na stránku s potvrdením o tom, že objednávka prebehla úspešne