Implementing Page Navigation

Jesper de Jong

SOFTWARE ARCHITECT

@jesperdj www.jesperdj.com

Overview

Navigation

- Rule-based Navigation

Faces Flows

- Implementing the Checkout Process

Understanding JSF Navigation

Understanding JSF Navigation

```
<h:outputLink value="index.xhtml">
    Back to Popular Products
</h:outputLink>
```

```
<a href="index.xhtml">
    Back to Popular Products
</a>
```

Understanding JSF Navigation

```
<h:outputLink value="index.xhtml">
    Back to Popular Products
</h:outputLink>
```

```
<h:link value="Back to Popular Products"
        outcome="index"/>
```

Understanding JSF Navigation

```
<h:outputLink value="index.xhtml">
    Back to Popular Products
</h:outputLink>
```

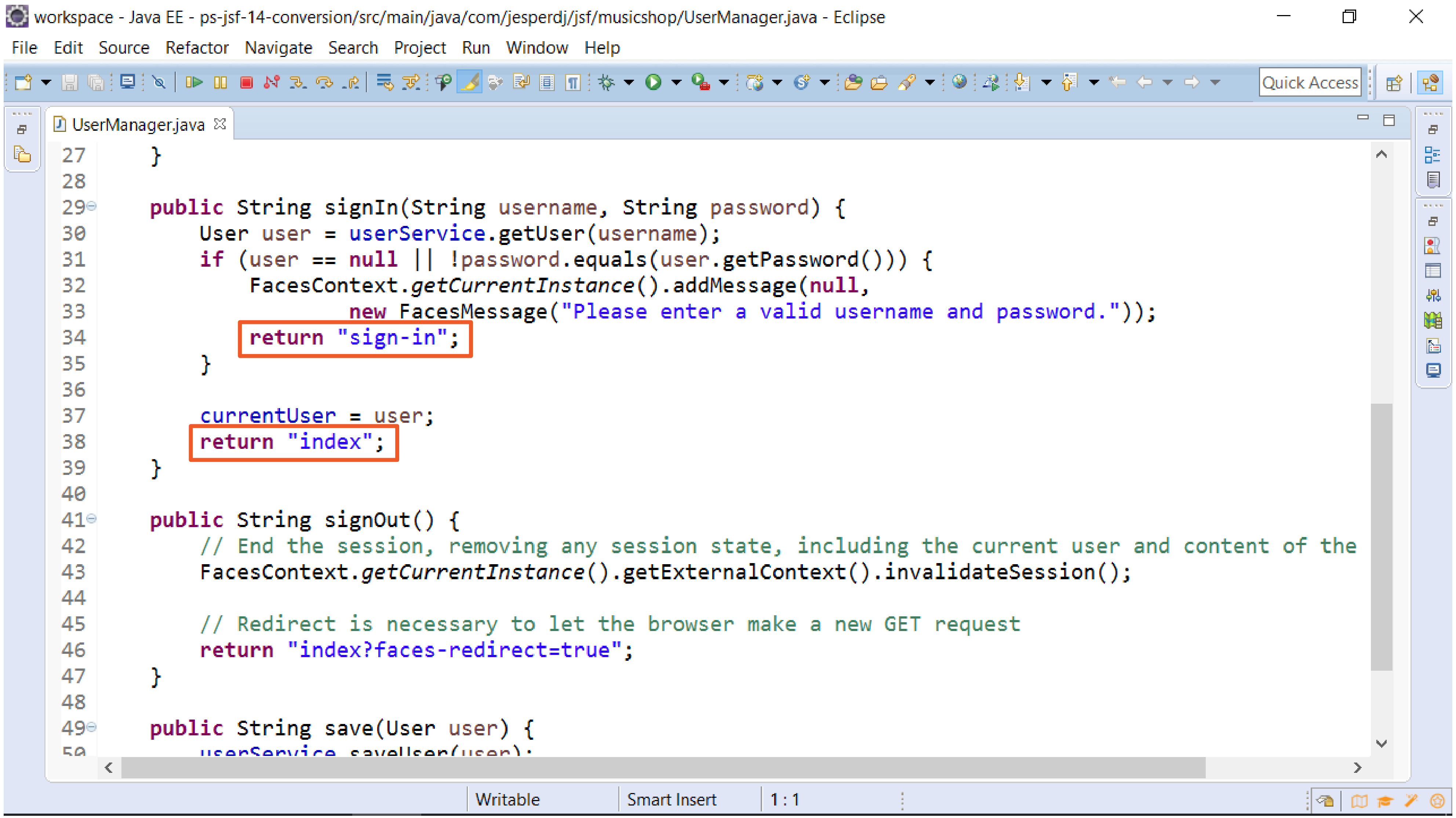
```
<h:link value="Back to Popular Products"
        outcome="index"/>
```

Understanding JSF Navigation

```
<h:outputLink value="index.xhtml">
    Back to Popular Products
</h:outputLink>
```

```
<h:link value="Back to Popular Products"
        outcome="index"/>
```

Understanding JSF Navigation


workspace - Java EE - ps-jsf-14-conversion/src/main/java/com/jesperdj/jsf/musicshop/UserManager.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Quick Access
UserManager.java
Writable
Smart Insert
1 : 1

```java
    }

    public String signIn(String username, String password) {
        User user = userService.getUser(username);
        if (user == null || !password.equals(user.getPassword())) {
            FacesContext.getCurrentInstance().addMessage(null,
                    new FacesMessage("Please enter a valid username and password."));
            return "sign-in";
        }

        currentUser = user;
        return "index";
    }

    public String signOut() {
        // End the session, removing any session state, including the current user and content of the
        FacesContext.getCurrentInstance().getExternalContext().invalidateSession();

        // Redirect is necessary to let the browser make a new GET request
        return "index?faces-redirect=true";
    }

    public String save(User user) {
        userService.saveUser(user);
```

Understanding JSF Navigation

- The navigation handler translates outcome values to actual links

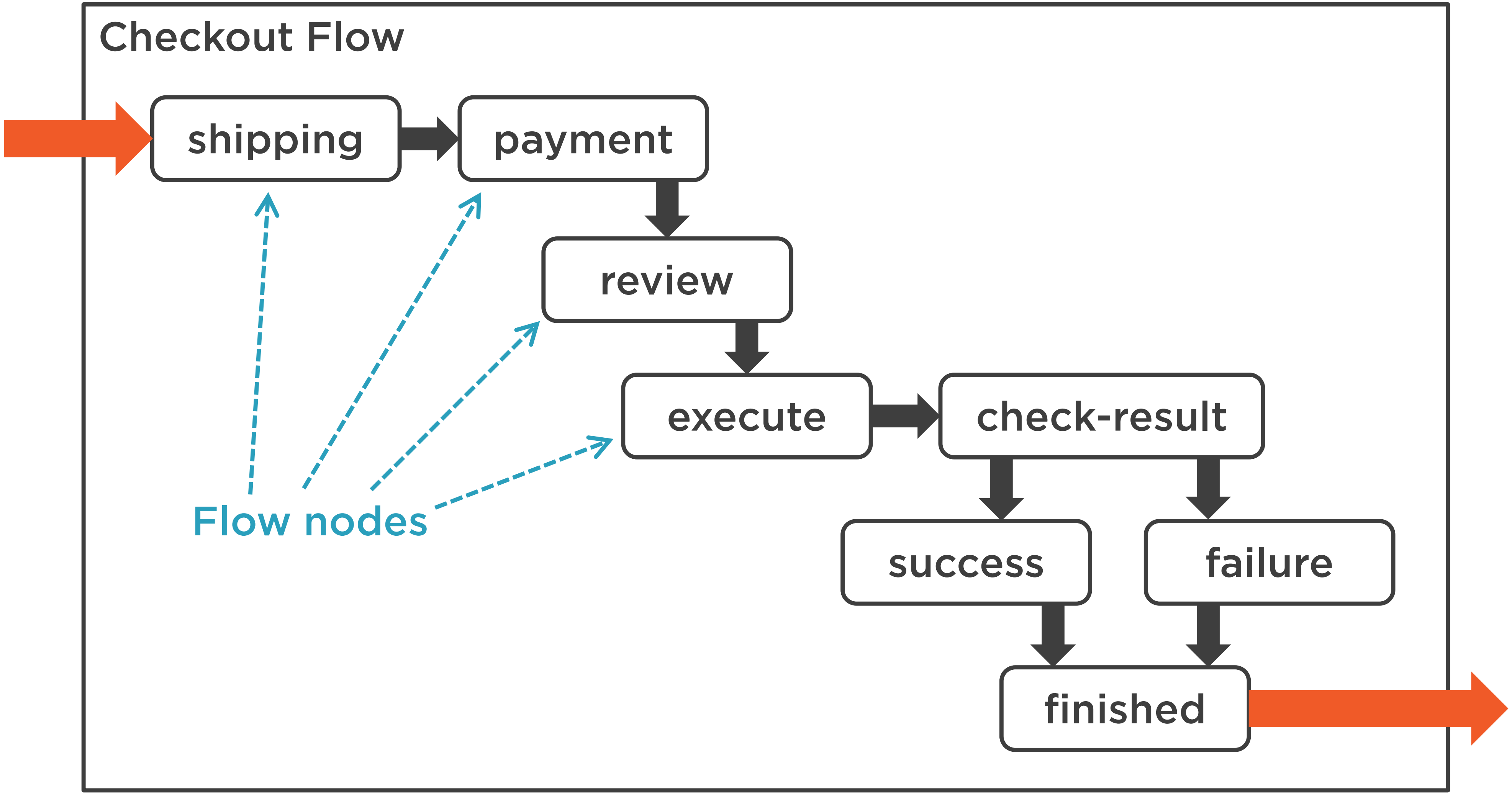
- Navigation rules can be defined in faces-config.xml

Understanding Faces Flows

The Checkout Process

- Enter shipping details
  - Name and address
- Enter payment details
  - Name, credit card type and number
- Review order
  - Confirm or cancel the order
- Payment succeeded or failed

Flow Nodes



Checkout Flow

shipping

payment

review

execute

check-result

success

failure

finished

Flow nodes

Kinds of Flow Nodes

- View node
  - Display a view (page)
- Method call node
  - Call a method
- Switch node
  - Select one of a series of cases
- Return node
  - End the flow
- Flow call node
  - Call another flow

Flows Are Similar to Methods

- **Entry point, parameters, exit points**
  - May return one or more values
- **Flow scope**
  - Flow-scoped beans are like local variables
- **Nested flow calls**
  - Call stack

Configuring Flows

- XML configuration file
  - In faces-config.xml or in a separate file
- In Java using the flow builder API
  - CDI producer method with `@FlowDefinition` annotation

Packaging Flows

- Conventional structure

```
src/main/webapp
    example
        example-flow.xml
        first-page.xhtml
        second-page.xhtml
```

- In a separate JAR file

Demo

Implement the checkout flow

Summary

Rule-based Navigation

- Navigation handler

Faces Flows

- Checkout flow

Coming Up

Building Your Own Components