

Java Technologies - Lab 4

[valid 2022-2023]

JavaServer Faces, Facelets. JNDI and Resources.

Continue the application created in [the previous lab](#).

Compulsory (1p)

Implement an *efficient* way for obtaining connections to the database.

- Configure a **connection pool** and a **JDBC resource** using an administrative tool (such as GlassFish/Payara Console or asadmin).
 - Create **DatSource** objects using either JNDI directly or *resource injection*.
-

Homework (2p)

Create the pages using **templates**:

- *page.xhtml*: describing the general aspect of the application pages: header, content, footer. The header should display the title and might include a menu bar.
The footer will display a copyright notice and the current version of the application. The *header*, *footer* and *the menu bar* should all be in separate .xhtml files.
 - *dataView.xhtml*: a generic page for displaying data as a *list*, *dataTable*, etc.
 - *dataEdit.xhtml*: a generic page for editing data. This could be a *dialog* containing a generic *form*.
-

Bonus (2p)

- Create a **composite** component that describes a datatable with support for CRUD operations. Rewrite the pages for managing the teams and the cities in order to use the composite component.
- Use the components **ajax** and **poll** in order to continuously display how many active sessions are in progress and information about the execution of your algorithm (in Choco solver this can be implemented using a search monitor).

Bibliography

- [Slides](#)
- [The Web Tier](#)
- [JavaServer Faces Technology](#)
- [Introduction to Facelets](#)
- [JavaServer Faces 2.2 Facelets Tag Library Documentation](#)
- [Java SE Tutorial: Java Naming and Directory Interface](#)
- [Java EE Tutorial: Resources](#)

Additional Readings

- [Extending PrimeFaces components](#)