Project 2

Enterprise Application Integration (MEB)
Department of Informatics Engineering
Delivery date: 29 November 2023

1 2 9 0
FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE D
COIMBRA

Objectives

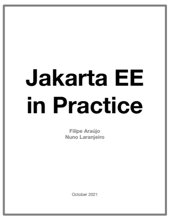
Gain familiarity with the development of tiered enterprise applications using the **Jakarta Enterprise Edition (Jakarta EE)** model. This includes the development of applications based on **Enterprise JavaBeans (EJB)**, the development of **SOAP** and **REST Web Services,** the use of a persistence engine compliant with the **Java Persistence API (JPA),** and the creation of a **web front-end**. Overall, you should build at least one Enterprise Archive, which can be deployed in an Application Server.

Final Delivery

- You must submit your project in a zip file using Inforestudante. Do not forget to associate your work colleague during the submission process.
- The submission contents are:
    - All source code of the requested applications ready to compile and execute.
- Make sure you can download and quickly execute the submitted assignment (for demonstration purposes).
- After submitting, you are required to register the (extra-class) effort spent solving the assignment. This step is mandatory. Please fill the effort form at:
  https://docs.google.com/spreadsheet/viewform?formkey=dHFha3NuWE1pYWJDbTJMTWcxTWRMS1E6MA

References

Jakarta EE in Practice by Filipe Araújo and Nuno Laranjeiro



Jakarta EE
in Practice

Filipe Araújo
Nuno Laranjeiro

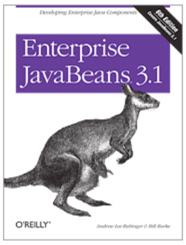
October 2021

This book is provided along with the course materials and provides a practical view of several important Jakarta EE technologies. In *Chapter 1* you can find information about the required software (in Section 1.1) and conceptual information about Maven, the tool we will be using to compile and build the projects (in ***Section 1.3***).

After reading the first chapter, you can find the information about the technologies associated with this project in ***Chapter 2*** (Java Persistence API), ***Chapter 3*** (Enterprise Java Beans), ***Chapter 4*** (SOAP web services), ***Chapter 5*** (REST web services), *Chapter 6* (Web applications, ***Section 6.1***). In ***Chapter 8*** you can find information associated with a full setup, allowing you to make use of all previous technologies integrated into a single application, which is packaged as an Enterprise Archive in an Application Server.

Other resources

There is extensive bibliography online about Jakarta EE technologies. A good reference is:



Enterprise JavaBeans 3.1 (6th Edition)
by Andrew Lee Rubinger and Bill Burke

O'Reilly Media
ISBN 0596158025
September 24, 2010

Other useful references include:

- The **Java EE Tutorial**, available at https://eclipse-ee4j.github.io/jakartaee-tutorial/
- **WildFly** documentation available at https://docs.wildfly.org/

Project Description

In this project, we will build a tiered application, using Jakarta EE technologies. The goal is to simulate a scenario where we need to query data and integrate services. We will use a **data persistence tool** (JPA/Hibernate) to manage data about **researchers and publications**. Next, we will build **EJBs** to provide access to the JPA managed information and **web services** (provided by the application server) to provide operations to a **web service client** application. Finally, we will create a **web front-end** to provide information to users using browsers. Overall, we are going to build the following set of applications:

a) An application that loads data about **researchers and their publication titles** into a database.
b) An application that loads data about **publications** into a database (the same one as before, to simplify the setup).
c) A SOAP web service that uses an EJB that can access the **researchers'** database and provides detailed information about **researchers**.
d) A REST web service that uses an EJB that can access the **publications** database and provides advanced information about **publications**.
e) A client application that can query both web services.
f) A web front-end to display basic information regarding **researchers/publications**.

The following paragraphs describe the minimum requirements, but you are free to create other operations as needed, as long as they do not simplify the expected implementation (ask your Professor in case of doubt). Note that some parts of the exercise are slightly different than what they would be in a real scenario, the goal is to have more problem diversity and gain broader knowledge on enterprise application development. You may assume that researcher names and publication titles are unique.

a) Loader – Researchers

This application will load researchers' information from an XML file into the database, using JPA. This database will hold researcher information and publication titles. Thus, no further information about publications exists in this database and we will also assume that the publication titles are unique.

a) Loader – Publications

This application will load publications information into a database, using JPA. The publications information should be in an XML file and must not include any specific relation to researchers, i.e., it's just a list of publications each publication holding a few simple attributes, such as title, citations, etc.

b) SOAP Web Service – Researchers

This service has access to the database (via EJB) and provides the following functionality:

- List all researchers in the database.
- List N researchers names that have the highest citation count (N is an argument of this operation).
- List researchers with the option of sorting by one of their simple attributes (the client may specify which one). This operation should also allow specifying the sort order.

c) REST Web Service – Publications

This service has access to the database (via EJB) and provides the following functionality:

- List detailed information about all publications.
- List detailed information about a given set of publications (titles are passed as an argument to the operation).

d) Client application

A command-line client application capable of using all functionality provided by the web services. It should also provide the following additional operations:

- Show the complete information of a researcher, which includes the detailed information about its publications.
- Show the complete information about one publication, including detailed information regarding its authors.

e) Web front-end

Finally, we will create a web front-end so that we can use the system in a browser. The functionality to be supported by the web front-end is the following:

- Show the list of researcher names present in the database.
- Your own custom functionality. Use your time wisely!

Good Work!