MINISTRY OF EDUCATION AND SCIENTIFIC RESEARCH

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA

FACULTY OF AUTOMATION AND COMPUTER SCIENCE
COMPUTER SCIENCE DEPARTMENT

DISTRIBUTED SYSTEMS

Assignment 1

Request-Reply Communication Paradigm

A1.2: Web app using Request – Reply

Ioan Salomie
Marcel Antal
Teodor Petrican

Tudor Cioara
Claudia Daniela Pop

Ionut Anghel
Dorin Moldovan
Ciprian Stan

2017

Contents

1. Requirements 3
1.1. Functional requirements: 3
1.2. Implementation technologies: 3
1.3. Non-functional requirements: 3
2. Deliverables 3
3. Evaluation 4
3.1. Assignment Related Basic Questions: 4
3.2. Grading 4
4. Bibliography 5

1. Requirements

Design, implement and test a three-tiered distributed system to view and post flights for an airport. The system consists of the following tiers: Presentation, Business Layer and Data Access.

1.1. Functional requirements:

- Users log in. Users are redirected to the page corresponding to their role.
- Client role
  - A client can view on his/her page all the flights in a list or table.
  - A client can query for the local time of the flight arrival and departure cities computed based on cities geographical coordinates.
- Administrator role
  - The administrator can perform CRUD operations on flights (Create, Read, Update and Delete)
- Each flight consists of the following information: flight number, airplane type, departure city, departure date and hour, arrival city, arrival date and hour.
- Each city has associated its geographical coordinates: latitude and longitude.
- In order to display the local time, the geographical coordinates of the city are passed to an external web service (e.g. http://new.earthtools.org/webservices.htm) which will return the actual time value.
- The simple users will not be able to enter the administrator page (e.g. by log-in and then copy-paste the admin URL to the browser)

1.2. Implementation technologies:

- Use the following technologies: HTML, Java Servlets and Hibernate ORM.

1.3. Non-functional requirements:

- Security: use authentication in order to restrict users to access the administrator pages (cookies, session, etc.)

2. Deliverables

- A solution description document (about 4 pages, Times New Roman, 10pt, Single Spacing) containing:
  a) Conceptual architecture of the distributed system.
  b) DB design.
  c) UML Deployment diagram.
  d) Readme file containing build and execution considerations.
- Source files. The source files and the database dump will be uploaded on the personal *bitbucket* account created at the *Lab resources* laboratory work, following the steps:
  - Create a repository on *bitbucket* with the exact name: *DS2017_Group_Name_Assigment_1*
  - Push the source code and the documentation (push the code not an archive with the code or war files)
  - Share the repository with the user *utcn_dsrl*

3. Evaluation

3.1. Assignment Related Basic Questions:

During project evaluation and grading you will be asked details about the following topics:

- URI and URL
- Web Clients and Web Servers
- HTTP protocol
- GET and POST HTTP methods
- HTML web forms
- Query strings
- Hidden variables
- Cookies
- Session
- Java Servlet
- Object-Relational Mapping (ORM)

3.2. Grading

The assignment will be graded as follows:

Points	Requirements
5 p	**Minimum to pass** • HTML page for presentation, Servlets for business logic and Hibernate for data access • Database • Documentation • Correct answers to 3.1 questions
1 p	Log-in with redirect (admin/users)
1 p	Call external web service
1p	Minimum Security: The simple users will not be able to enter the administrator page
2p	Answers of Reinforcement Learning Questions of A1.1

4. Bibliography

1. http://www.coned.utcluj.ro/~salomie/DS_Lic
2. Lab Book: I. Salomie, T. Cioara, I. Anghel, T.Salomie, *Distributed Computing and Systems: A practical approach*, Albastra, Publish House, 2008, ISBN 978-973-650-234-7
3. Hibernate:
    a. http://www.tutorialspoint.com/hibernate/
    b. http://www.javatpoint.com/hibernate-tutorial
    c. http://www.javacodegeeks.com/2015/03/hibernate-tutorial.html
    d. http://www.mkyong.com/tutorials/hibernate-tutorials/
4. Maven: https://maven.apache.org/
5. Servlets:
    a. http://docs.oracle.com/javaee/6/tutorial/doc/bnafd.html
    b. http://www.tutorialspoint.com/servlets/
    c. http://www.javatpoint.com/servlet-tutorial
    d. http://www.javacodegeeks.com/2014/12/java-servlet-tutorial.html
6. HTML web forms – Servlets interaction: http://www.tutorialspoint.com/servlets/servlets-form-data.htm

Further reading:

- JSF and JSP vs Servlets
    - http://www.tutorialspoint.com/jsp/
    - http://www.tutorialspoint.com/jsf