Message Wall service

Juan-Luis Gorricho

Implementation

- This is a first approach implementing a message wall service.
- This approach focuses on defining interfaces and their implementations to manage the usual data of a message wall service: Message, MessageWall and UserAccess.
- This approach follows the MVC pattern, but it doesn't consider any persistence.

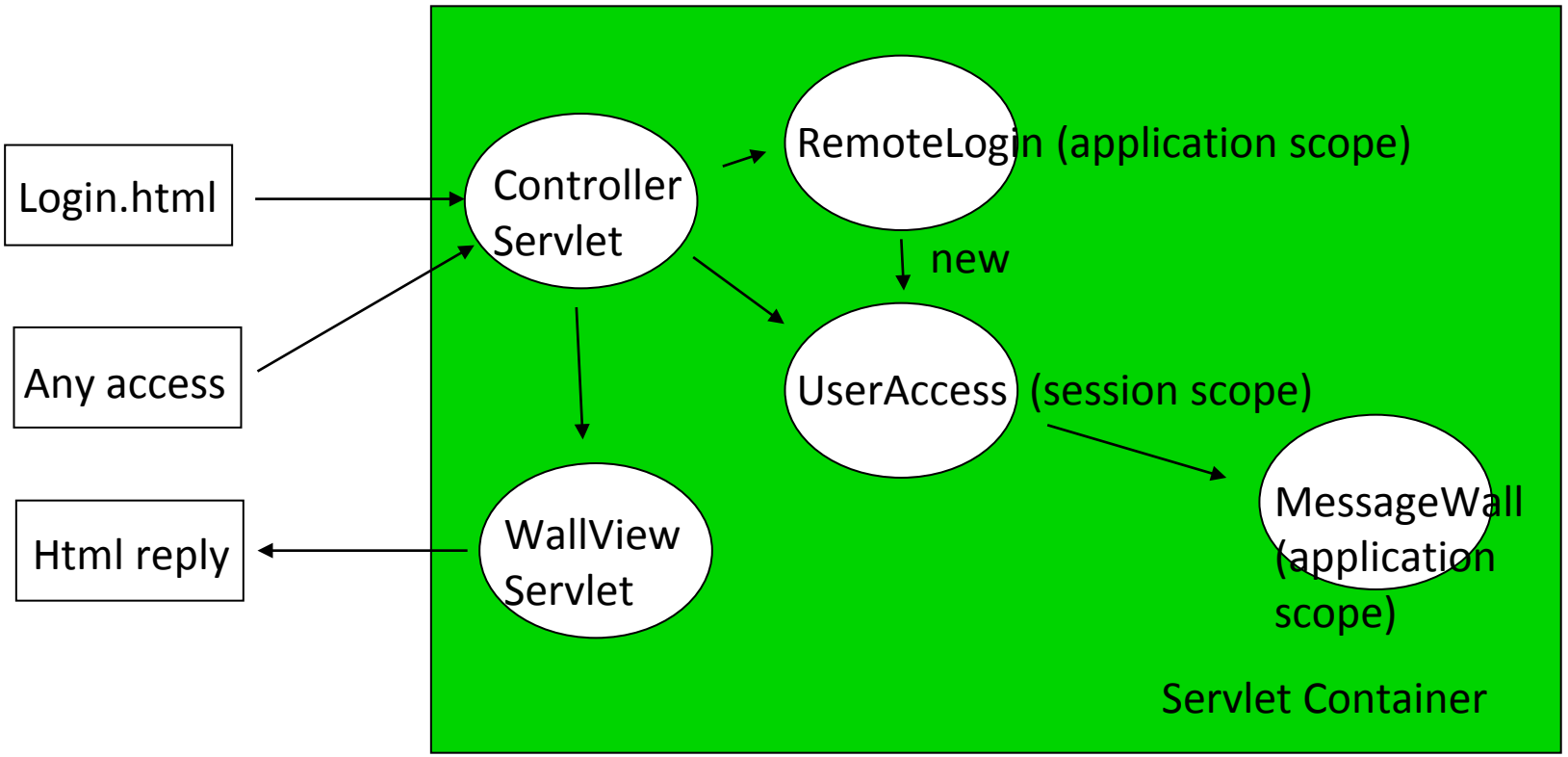
Interfaces

```
public interface MessageWall {
    void put(String user, String msg);
    boolean delete(String user, int index);
    Message getLast();
    int getNumber();
    List<Message> getAllMessages();
}

public interface Message {
    String getContent();
    String getOwner();
}
```

Interfaces

```
public interface RemoteLogin {
    UserAccess connect(String usr, String passwd);
}

public interface UserAccess {
    String getUser();
    Message getLast();
    int getNumber();
    void put(String msg);
    boolean delete(int index);
    List<Message> getAllMessages();
}
```

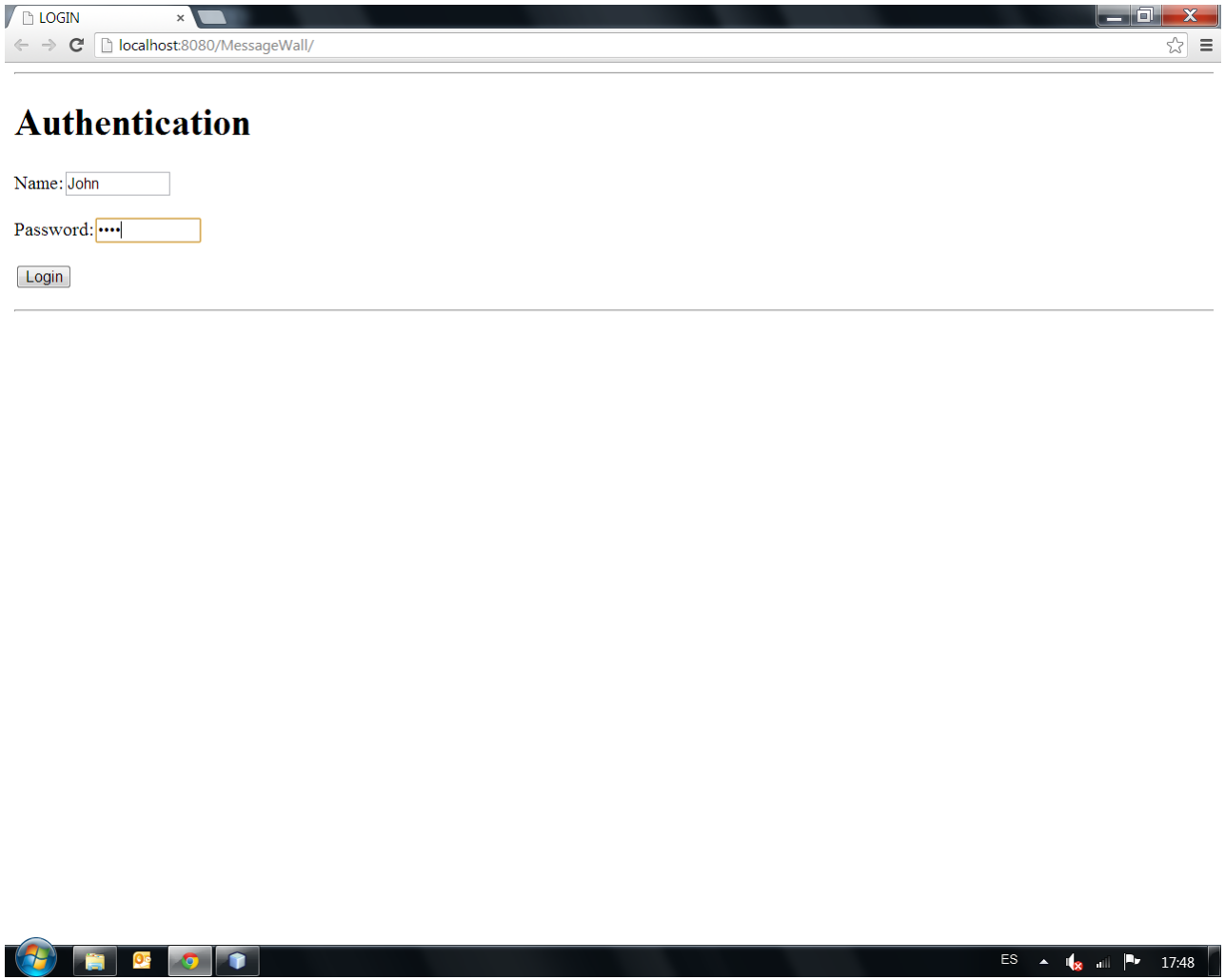
schema of classes



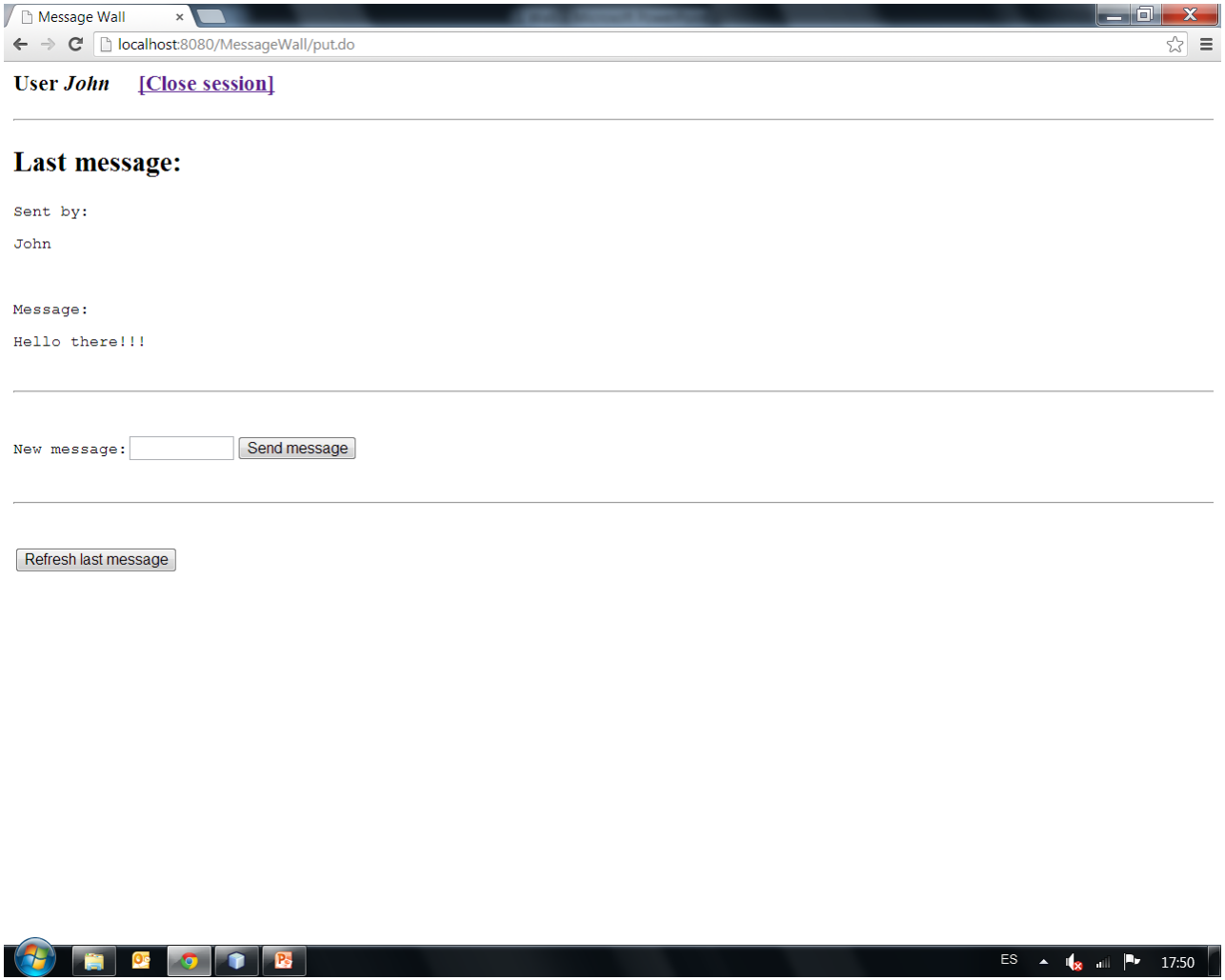
Packets

- demo.spec:
  - Message.java, MessageWall.java
  - RemoteLogin.java, UserAccess.java
- demo.impl:
  - MessageWall_and_RemoteLogin_Impl.java
  - Message_Impl.java, UserAccess_Impl.java
- demo.web:
  - ContextListener.java
  - ControllerServlet.java
  - WallViewServlet.java

Login


LOGIN
localhost:8080/MessageWall/
Authentication
Name: John
Password: ••••
Login
ES
17:48

Wall view


Message Wall
localhost:8080/MessageWall/put.do
User John [Close session]
Last message:
Sent by:
John
Message:
Hello there!!!
New message:
Send message
Refresh last message
ES
17:50

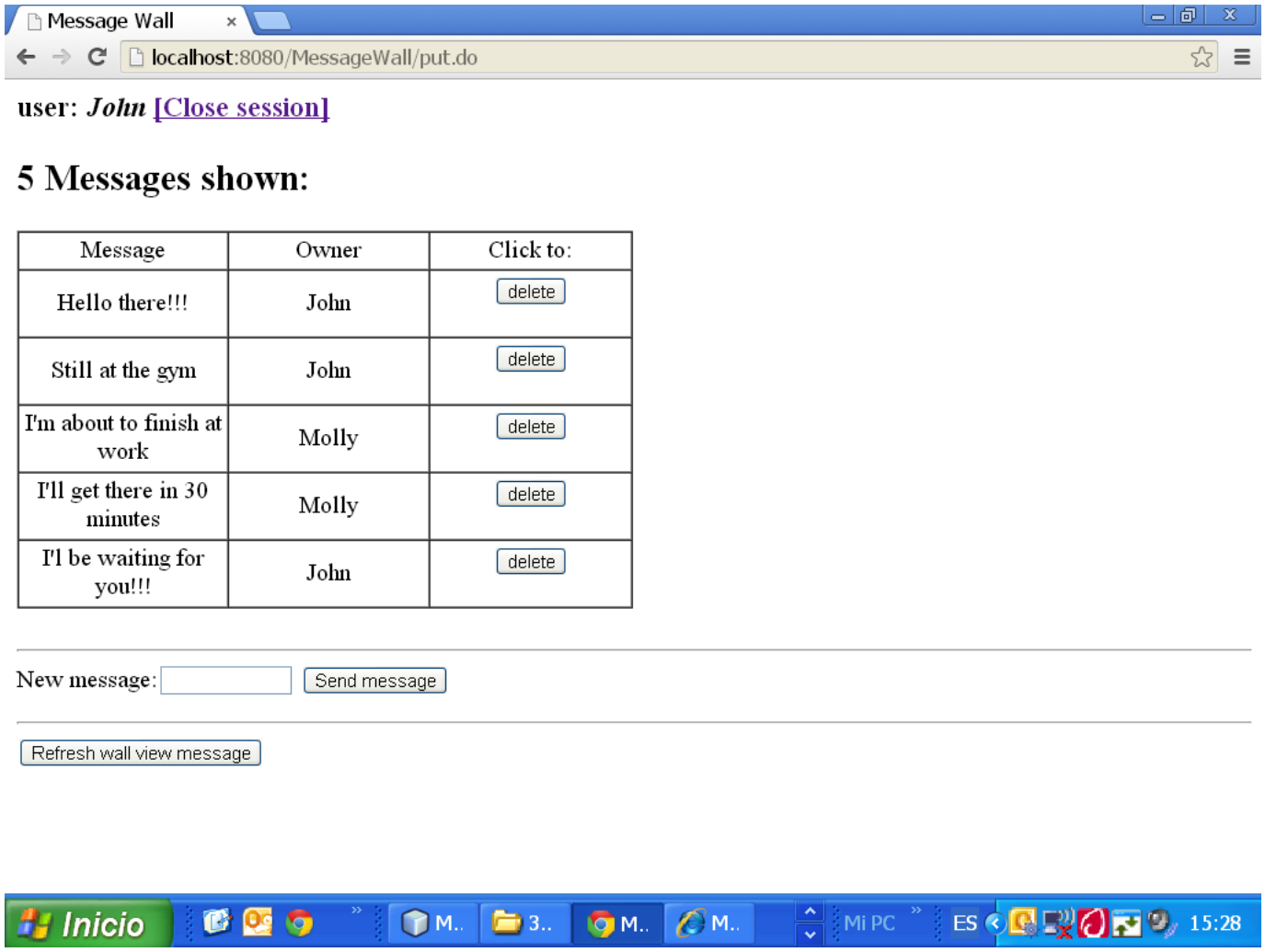
1st Exercise

- Review the initial content of the web application: MessageWall_exercise
- Complete the code for the classes in package demo.impl and the java class: ControllerServlet
- **Hint**: for the forward action made from the ControllerServlet, review the web.xml file as it specifies the servlet mapping when forwarding the request to a non-html file

2nd Exercise

- Complete the code for wallview.jsp to show all messages posted on the wall at the same time, attach each message's owner and the option to delete the message
- Modify the ControllerServlet to show the new view instead of the old one
- Add a minimal login functionality using a hash table with predefined pairs of user/password

new Wall view



Exercise

- Add the following JavaScript code for the wallview.jsp file to periodically request (polling strategy) for new messages on the wall, without having to press the refresh button (asynchronous performance):

```
<script>
	setInterval(my_function, 10000);
	function my_function() {
	    document.getElementsByName('refresh_button')[0].click();
	}
</script>
```