

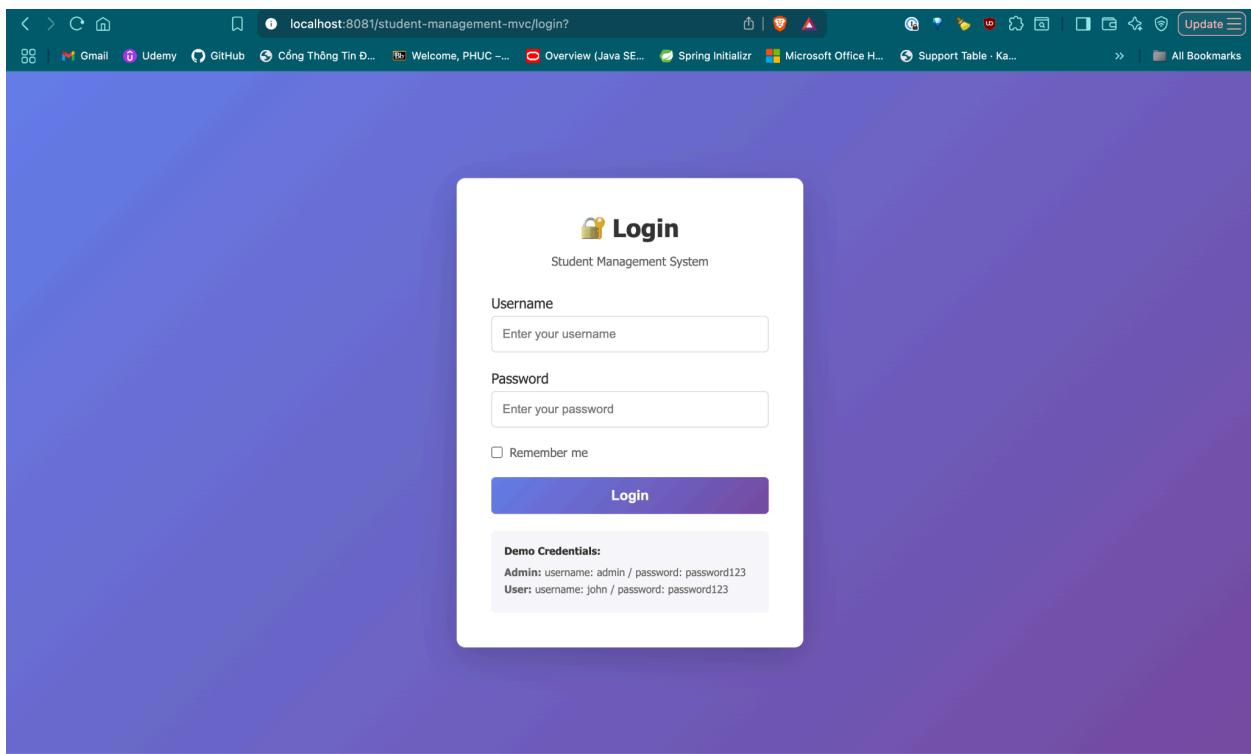
Lab 6

Github site & link

<https://github.com/alexis0704/web-dev-practice-lab/tree/main/student-management-mvc>

Testing the sample code

1. Access: `http://localhost:8080/YourApp/` → Redirected to login (AuthFilter)



When opening the root URL, the AuthFilter detects that there is no logged-in user and automatically redirects the request to the login page.

2. Login with: admin / password123 → Creates session → Redirected to dashboard

The screenshot shows a web browser window with the URL `localhost:8081/student-management-mvc/dashboard`. The title bar says "Student Management System". The top right corner shows "Admin User" and a red button with "Logout". The main content area has a header "Welcome back, Admin User!" followed by a sub-header "Here's what's happening with your students today.". Below this is a card with a student icon and the number "15" followed by "Total Students". A section titled "Quick Actions" contains three buttons: "View All Students" (blue), "+ Add New Student" (green), and "Search Students" (orange).

After entering valid admin credentials, the login servlet authenticates the user, creates a new HttpSession, stores info in it, and redirects to the dashboard page.

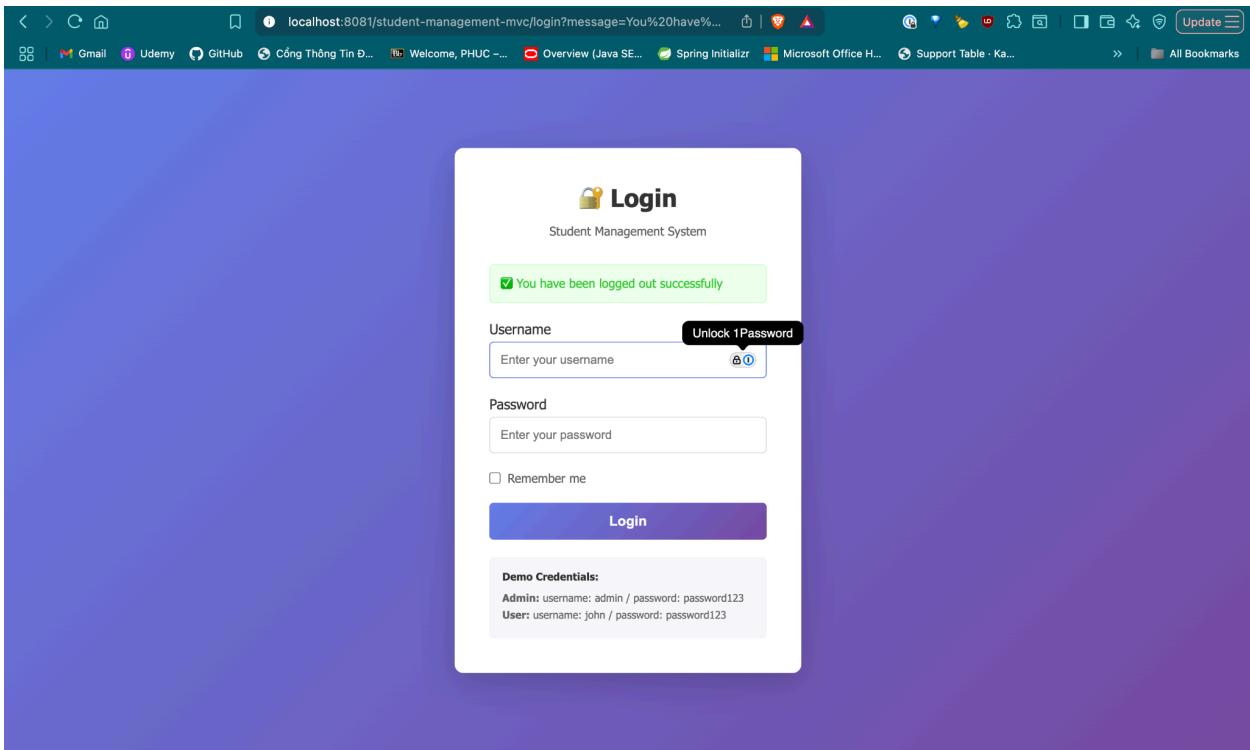
3. Click "View All Students" → Shows student list → Edit/Delete buttons visible (admin)

The screenshot shows a web browser window with the URL localhost:8081/student-management-mvc/student?action=list. The title bar says "Student Management System". The top navigation bar includes "Admin User", "ADMIN" (highlighted in red), "Dashboard", and "Logout". Below the header, a sub-header reads "MVC Pattern with Jakarta EE & JSTL". A search bar with placeholder "Search by code, name, or email" and a "Search" button is present. To its right is a dropdown menu "Filter by Major:" set to "All Majors" with an "Apply" button. Further right is a blue button "+ Add New Student". The main content area is titled "Student List" and displays a table of student records. The table has columns: ID, STUDENT CODE, FULL NAME, EMAIL, MAJOR, and ACTIONS. The data is as follows:

ID	STUDENT CODE	FULL NAME	EMAIL	MAJOR	ACTIONS
18	AB123	asakdjfhaskdf	student@example.com	Computer Science	<button>Edit</button> <button>Delete</button>
17	SV017	Le Van N	n.le@example.com	Software Engineering	<button>Edit</button> <button>Delete</button>
16	SV016	Tran Thi M	m.tran@example.com	Information Technology	<button>Edit</button> <button>Delete</button>
15	SV015	Nguyen Van L	l.nguyen@example.com	Computer Science	<button>Edit</button> <button>Delete</button>
14	SV014	Le Van K	k.le@example.com	Software Engineering	<button>Edit</button> <button>Delete</button>
13	SV013	Tran Thi I	i.tran@example.com	Information Technology	<button>Edit</button> <button>Delete</button>
12	SV012	Nguyen Van H	h.nguyen@example.com	Computer Science	<button>Edit</button> <button>Delete</button>
11	SV011	Le Van G	g.le@example.com	Software Engineering	<button>Edit</button> <button>Delete</button>

From the dashboard, the admin can open the student list. Because the user had the admin role, the page displays Edit and Delete buttons → role-based access.

4. Logout → Session invalidated → Redirected to login



Clicking Logout calls the LogoutController, this invalidates the current session and redirects back to the login page. The code clears session data and forces re-authentication.

5. Login with: john / password123 → Regular user view → No Edit/Delete buttons

The screenshot shows a web browser window with the URL `localhost:8081/student-management-mvc/student?action=list`. The title bar says "Student Management System". The top navigation bar includes "John Doe", "USER", "Dashboard", and "Logout". Below the header is a search bar with placeholder "Search by code, name, or email" and a "Search" button. To the right is a dropdown menu for "Filter by Major" set to "All Majors" with an "Apply" button. The main content area is titled "Student List" and displays a table of student records. The table has columns: ID, STUDENT CODE, FULL NAME, EMAIL, and MAJOR. The data is as follows:

ID	STUDENT CODE	FULL NAME	EMAIL	MAJOR
18	AB123	asakdjfhaskdf	student@example.com	Computer Science
17	SV017	Le Van N	n.le@example.com	Software Engineering
16	SV016	Tran Thi M	m.tran@example.com	Information Technology
15	SV015	Nguyen Van L	l.nguyen@example.com	Computer Science
14	SV014	Le Van K	k.le@example.com	Software Engineering
13	SV013	Tran Thi I	i.tran@example.com	Information Technology
12	SV012	Nguyen Van H	h.nguyen@example.com	Computer Science
11	SV011	Le Van G	g.le@example.com	Software Engineering
10	SV010	Tran Thi F	f.tran@example.com	Information Technology
9	SV009	Nguyen Van E	e.nguyen@example.com	Computer Science

At the bottom of the table are page navigation buttons: "1" (highlighted), "2", and "Next »".

Logging in with a normal user account only shows a read-only student list (no Edit/Delete buttons). Regular users have restricted permissions compared to admins.

6. Try to access: /student?action=new → AdminFilter blocks access → Error message displayed

MVC Pattern with Jakarta EE & JSTL

✖ Access denied. Admin privileges required.

ID	STUDENT CODE	FULL NAME	EMAIL	MAJOR
18	AB123	asakdjfhaskdf	student@example.com	Computer Science
17	SV017	Le Van N	n.le@example.com	Software Engineering
16	SV016	Tran Thi M	m.tran@example.com	Information Technology
15	SV015	Nguyen Van L	i.nguyen@example.com	Computer Science
14	SV014	Le Van K	k.le@example.com	Software Engineering
13	SV013	Tran Thi I	i.tran@example.com	Information Technology
12	SV012	Nguyen Van H	h.nguyen@example.com	Computer Science
11	SV011	Le Van G	g.le@example.com	Software Engineering
10	SV010	Tran Thi F	f.tran@example.com	Information Technology

When a non-admin user manually tries to open the “new student” URL, the AdminFilter checks the user’s role, detects that they are not admin, and blocks the request, redirecting back to the list page with an “Access denied. Admin privileges required.” message. This proves that admin-only actions are protected at the filter level.