VIETNAM NATIONAL UNIVERSITY OF HO CHI MINH CITY

THE INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Web Application Development - IT093IU

Homework Lab 6: AUTHENTICATION & SESSION MANAGEMENT

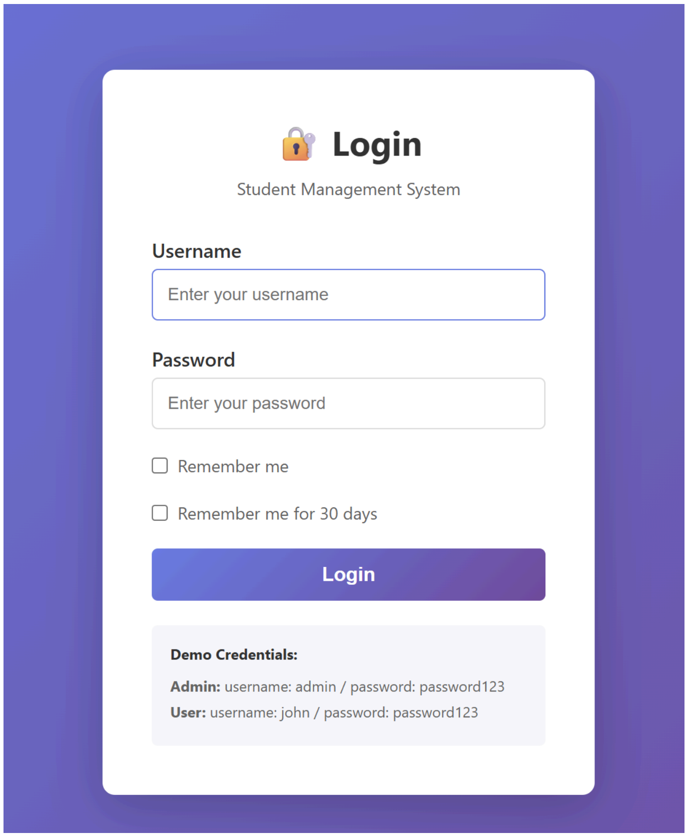
Nguyễn Vũ Thành Tính - ITCSIU23039

Lab Instructor: N.T.Nghia

Github Links:

https://github.com/ngvuthtinh/Web_Lab/tree/main/Lab_6

1. Login page



🔐 Login

Student Management System

Username

Enter your username

Password

Enter your password

□ Remember me

□ Remember me for 30 days

Login

Demo Credentials:

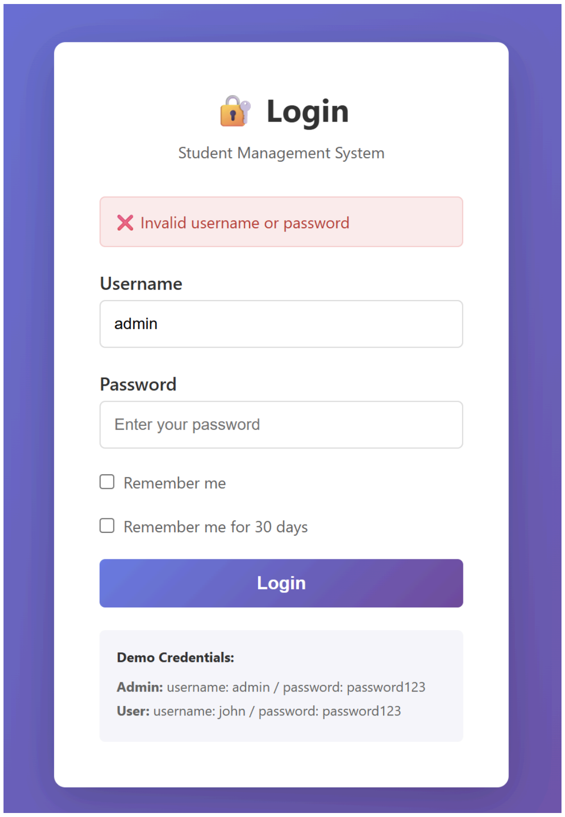
Admin: username: admin / password: password123

User: username: john / password: password123

Explanation

- User accesses: http://localhost:8080/StudentManagementMVC/.
- The request passes through AuthFilter (file: filter/AuthFilter.java).
- AuthFilter checks whether a session exists: If session is null, user is not logged in.
- AuthFilter redirects the user to "login".
- The redirected request is handled by LoginController.doGet() (file: controller/LoginController.java).
- LoginController forwards the request to: views/login.jsp

2. Login with wrong credentials (error message)



🔒 Login

Student Management System

✕ Invalid username or password

Username

admin

Password

Enter your password

□ Remember me

□ Remember me for 30 days

Login

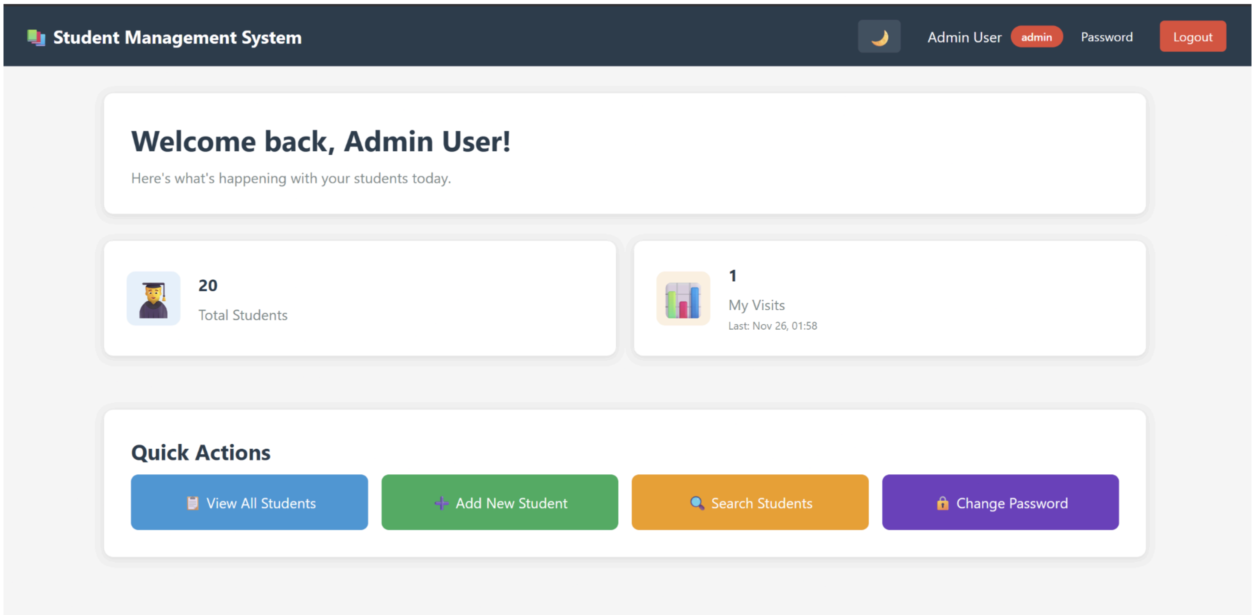
Demo Credentials:

Admin: username: admin / password: password123
User: username: john / password: password123

Explanation:

- User submits login form in login.jsp with action="login" and method="post".
- Request is processed by LoginController.doPost().
- Controller calls UserDAO.authenticate(username, password).
- UserDAO validates the hashed password using BCrypt.checkpw().
- Validation fails (returns null), so LoginController sets an error message attribute.
- Controller forwards back to views/login.jsp to display the error.

3. Dashboard after login



Welcome back, Admin User!

Here's what's happening with your students today.

20
Total Students

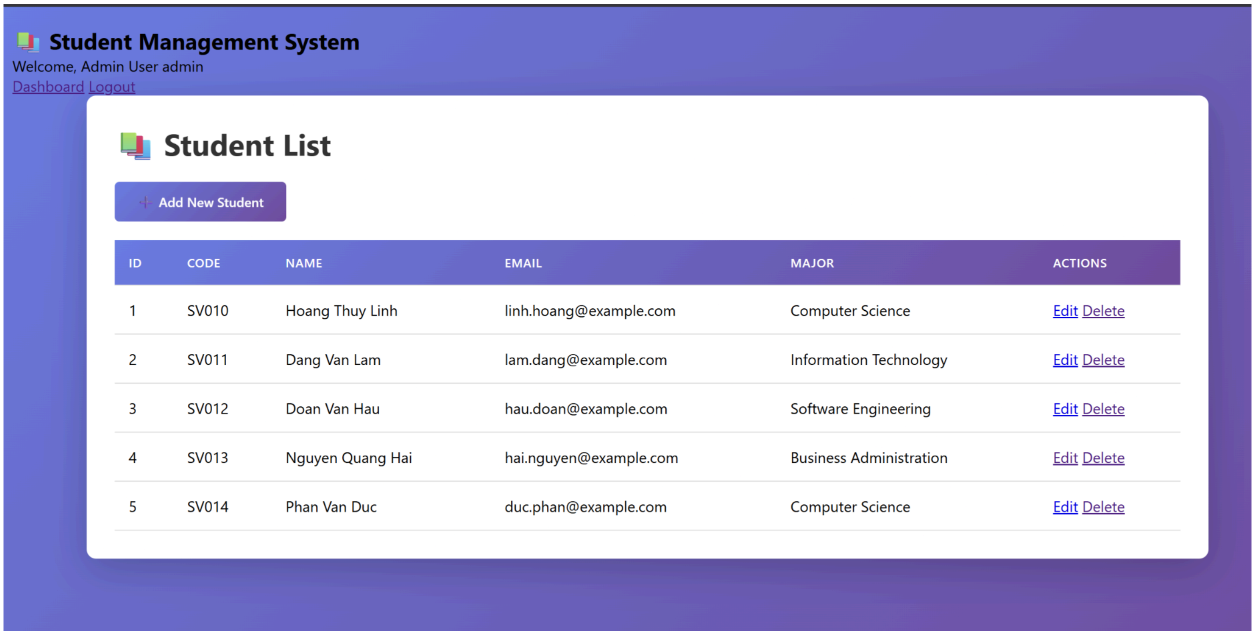
1
My Visits
Last: Nov 26, 01:58

Quick Actions

Explanation:

- UserDAO returns a valid User object upon successful authentication.
- LoginController creates a new session: session.setAttribute("user", user).
- Controller redirects to /dashboard.
- Request reaches DashboardController.doGet() (file: controller/DashboardController.java).
- DashboardController forwards to views/dashboard.jsp to render statistics and user info.

4. Student list as admin (with Edit/Delete buttons)



📚 **Student Management System**

Welcome, Admin User admin

Dashboard Logout

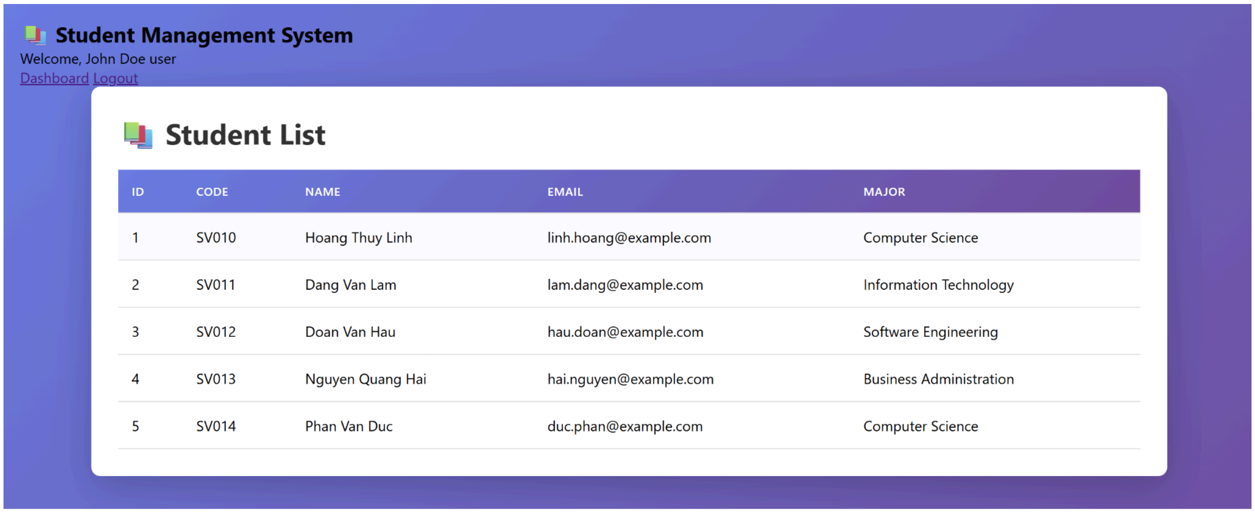
📚 Student List

ID	CODE	NAME	EMAIL	MAJOR	ACTIONS
1	SV010	Hoang Thuy Linh	linh.hoang@example.com	Computer Science	Edit Delete
2	SV011	Dang Van Lam	lam.dang@example.com	Information Technology	Edit Delete
3	SV012	Doan Van Hau	hau.doan@example.com	Software Engineering	Edit Delete
4	SV013	Nguyen Quang Hai	hai.nguyen@example.com	Business Administration	Edit Delete
5	SV014	Phan Van Duc	duc.phan@example.com	Computer Science	Edit Delete

Explanation:

- Admin clicks "View All Students", sending a request to: /student?action=list.
- Request passes through AuthFilter (allowed due to active session).
- Request is handled by StudentController.doGet() (file: controller/StudentController.java).
- Controller calls studentDAO.getAllStudents() and attaches the list to the request.
- Controller forwards to views/student-list.jsp.
- JSP checks sessionScope.role: Since role is "admin", Edit/Delete buttons are displayed.

5. Student list as regular user (without Edit/Delete)



Student Management System

Welcome, John Doe user

Dashboard Logout

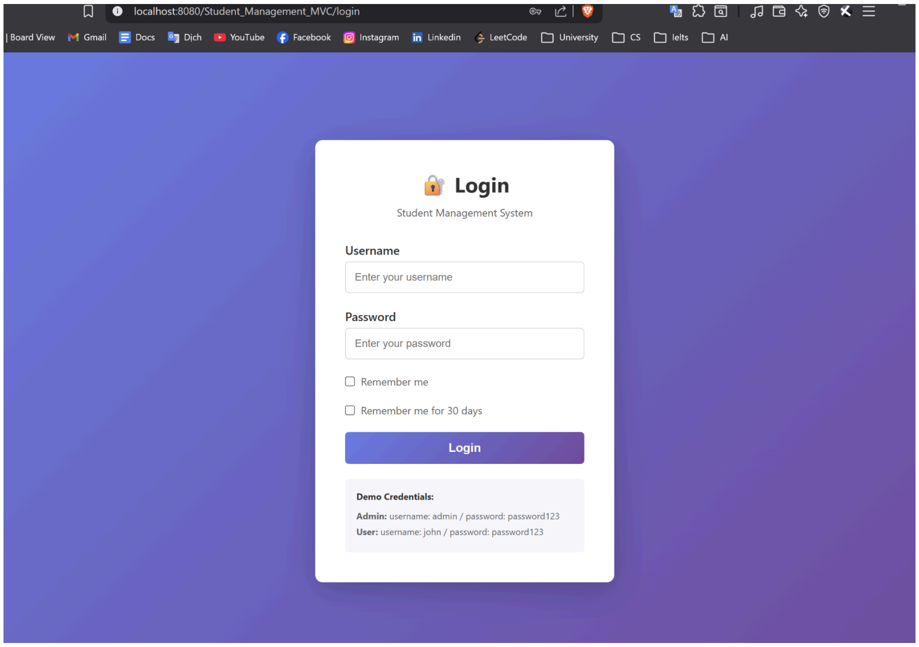
Student List

ID	CODE	NAME	EMAIL	MAJOR
1	SV010	Hoang Thuy Linh	linh.hoang@example.com	Computer Science
2	SV011	Dang Van Lam	lam.dang@example.com	Information Technology
3	SV012	Doan Van Hau	hau.doan@example.com	Software Engineering
4	SV013	Nguyen Quang Hai	hai.nguyen@example.com	Business Administration
5	SV014	Phan Van Duc	duc.phan@example.com	Computer Science

Explanation:

- Regular user (role "user") accesses /student?action=list.
- StudentController retrieves student data and forwards to views/student-list.jsp.
- JSP checks sessionScope.role: Since role is "user", the check fails.
- The "Add New Student" button and "Actions" column (Edit/Delete) are hidden from the view.

6. Accessing protected page without login (redirect)



🔒 **Login**

Student Management System

Username

Enter your username

Password

Enter your password

□ Remember me

□ Remember me for 30 days

Login

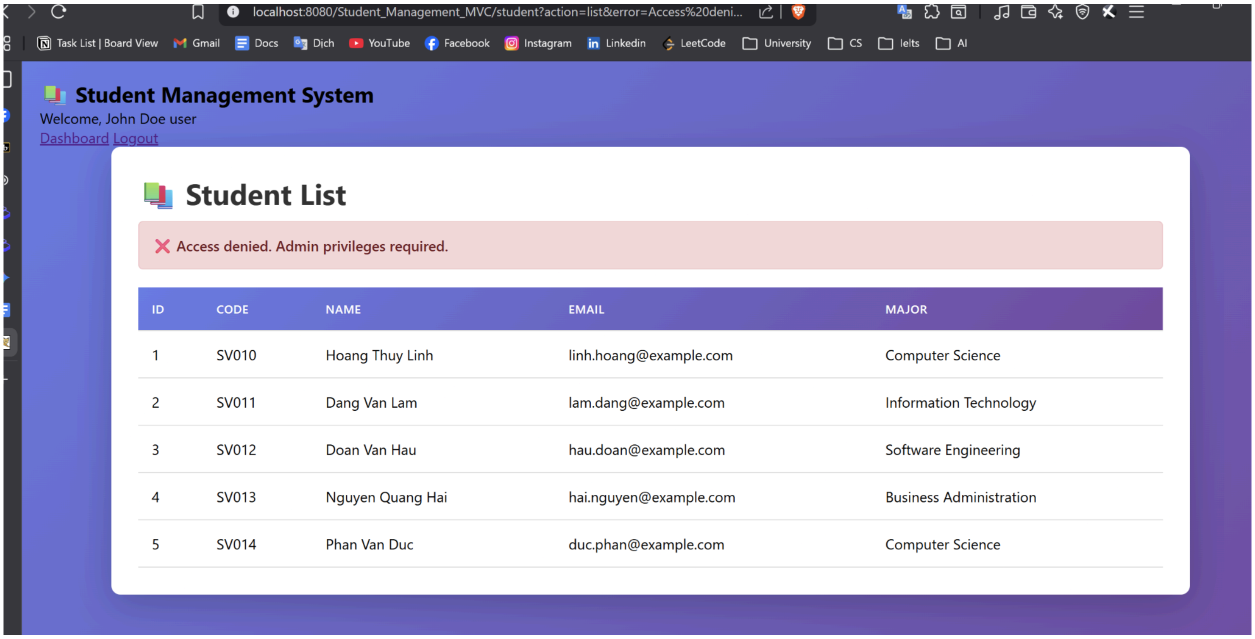
Demo Credentials:

Admin: username: admin / password: password123
User: username: john / password: password123

Explanation:

- User attempts to access protected URL: /dashboard.
- Request passes through AuthFilter.
- AuthFilter checks for session.getAttribute("user").
- Since no user is in session, AuthFilter blocks the request.
- AuthFilter redirects the user to /login.

7. Non-admin trying admin action (blocked)



Student Management System

Welcome, John Doe user

Dashboard Logout

Student List

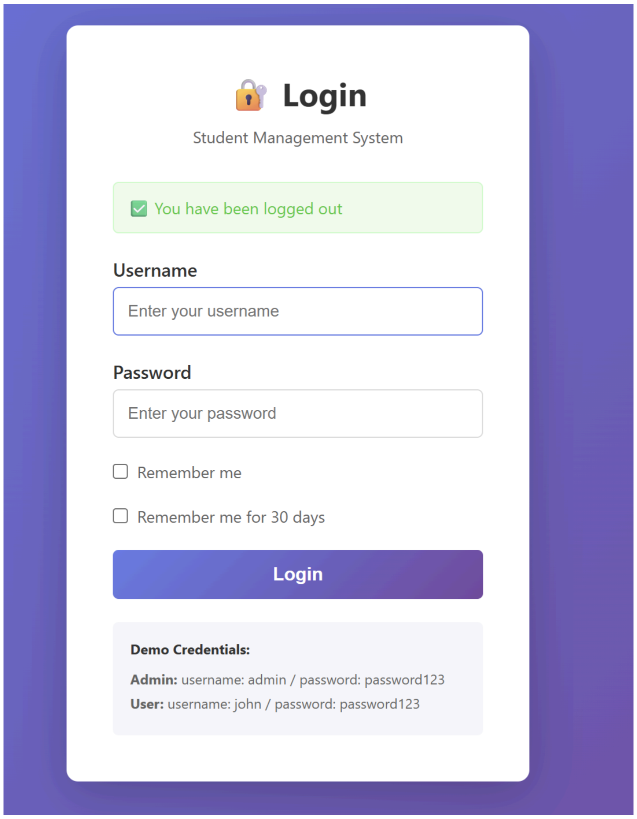
✕ Access denied. Admin privileges required.

ID	CODE	NAME	EMAIL	MAJOR
1	SV010	Hoang Thuy Linh	linh.hoang@example.com	Computer Science
2	SV011	Dang Van Lam	lam.dang@example.com	Information Technology
3	SV012	Doan Van Hau	hau.doan@example.com	Software Engineering
4	SV013	Nguyen Quang Hai	hai.nguyen@example.com	Business Administration
5	SV014	Phan Van Duc	duc.phan@example.com	Computer Science

Explanation:

- Regular user manually enters URL: /student?action=new.
- Request passes AuthFilter (session exists) but reaches AdminFilter (file: filter/AdminFilter.java).
- AdminFilter checks the user role: if (!user.isAdmin()).
- Since the role is "user", AdminFilter blocks the request.
- The filter redirects the user to the list page with an error parameter: error=Access denied

8. Logout success message



🔐 Login

Student Management System

✅ You have been logged out

Username

Enter your username

Password

Enter your password

□ Remember me

□ Remember me for 30 days

Login

Demo Credentials:

Admin: username: admin / password: password123

User: username: john / password: password123

Explanation:

- User clicks logout, sending request to: /logout.
- Request hits LogoutController.doGet() (file: controller/LogoutController.java).
- Controller invalidates the session using session.invalidate().
- User is redirected back to /login with a success message.