

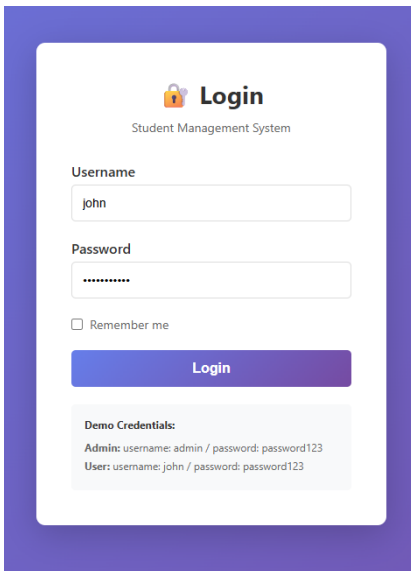
## Lab6 Homework ITCSIU22266

### Exercise 5:

```
if (isLoggedIn) {  
    // User is logged in, allow access  
    chain.doFilter(request, response);  
} else {  
    // User not logged in, redirect to login  
    String loginURL = contextPath + "/login";  
    httpResponse.sendRedirect(loginURL);  
}  
}
```

AuthFilter.java navigate user to login page if they haven't logged in yet or the session expired.

For example, user session has expired, then navigate to login page, else, gain access to the webpage



The image shows a login form titled "Login" for a "Student Management System". The form is set against a white background with a purple border. It includes a lock icon, input fields for "Username" (containing "john") and "Password" (masked with dots), a "Remember me" checkbox, and a purple "Login" button. Below the button is a "Demo Credentials" section with two entries: "Admin: username: admin / password: password123" and "User: username: john / password: password123".

**Login**  
Student Management System

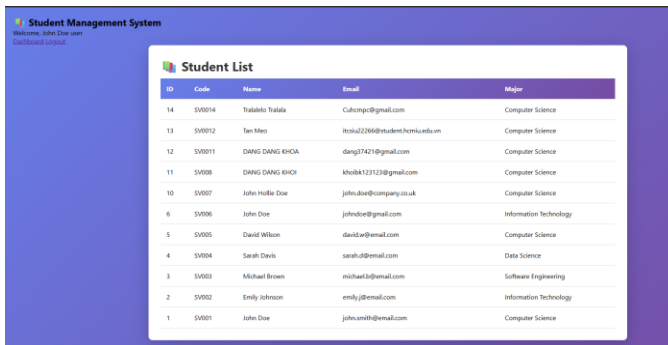
Username  
john

Password  
\*\*\*\*\*

☐ Remember me

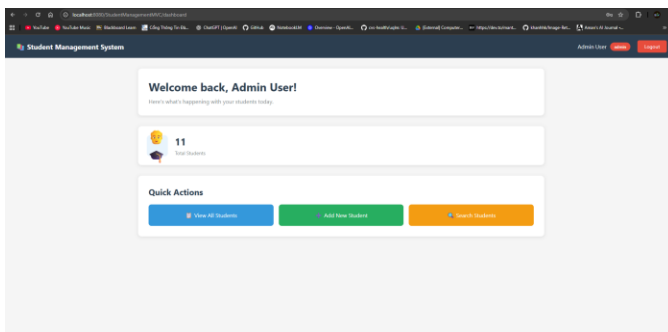
**Login**

**Demo Credentials:**  
Admin: username: admin / password: password123  
User: username: john / password: password123

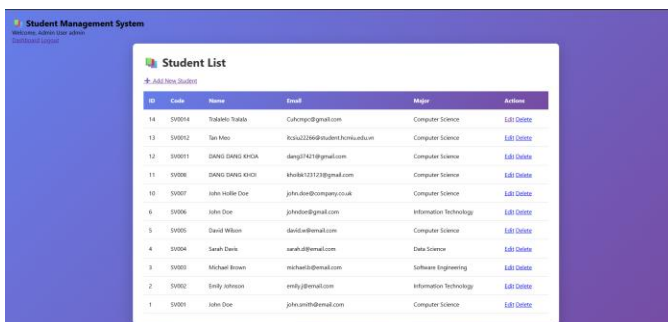



## Exercise 6:

Login as admin, will navigate to admin dashboard



Admin have permission to edit/delete student,



 **Edit Student**

Student Code \*


Format: 2 letters + 3+ digits

Full Name \*


Email \*

Major \*

Computer Science

 Update Student

✖ Cancel

 **Student List**

[+ Add New Student](#)

ID	Code	Name	Email	Major	Actions
14	SV0014	Tralalelo Tralala	cuhcmpc@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
13	SV0012	Tan Meo	itcsu22266@student.hcmiu.edu.vn	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
12	SV0011	DANG DIANG KHOA	dang37421@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
11	SV008	DANG DIANG KHOI	khoibk123123@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
10	SV007	John Hollie Doe	john.doe@company.co.uk	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
6	SV006	John Doe	johndoe@gmail.com	Information Technology	<a href="#">Edit</a> <a href="#">Delete</a>
5	SV005	David Wilson	david.w@email.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
4	SV004	Sarah Davis	sarah.d@email.com	Data Science	<a href="#">Edit</a> <a href="#">Delete</a>
3	SV003	Michael Brown	michael.b@email.com	Software Engineering	<a href="#">Edit</a> <a href="#">Delete</a>
2	SV002	Emily Johnson	emily.j@email.com	Information Technology	<a href="#">Edit</a> <a href="#">Delete</a>
1	SV001	John Doe	john.smith@email.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>

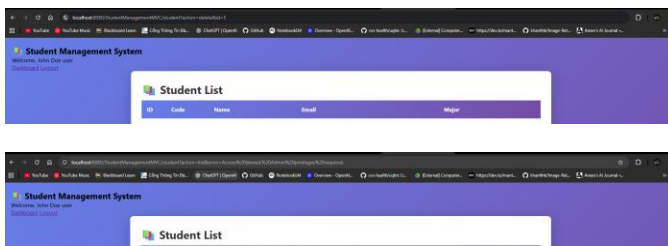
If login as user, wont have permission to edit or delete student.

**Student Management System**  
Welcome, John Doe user  
[Dashboard](#) [Logout](#)

Student List

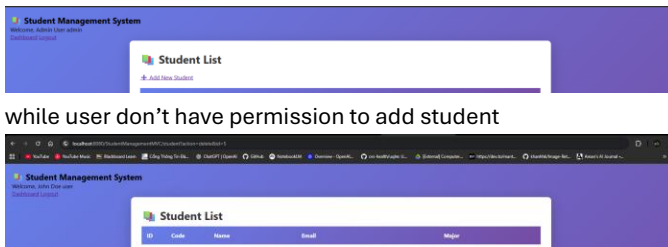
ID	Code	Name	Email	Major
14	SV0014	Trabekio Trabala	Cubemp@gmail.com	Computer Science
13	SV0012	Tan Mei	Itisid2206@student.honolulu.hawaii	Computer Science
12	SV0011	DANG DANG KHICA	dang37421@gmail.com	Computer Science
11	SV008	DANG DANG KHICH	khoh41211213@gmail.com	Computer Science
10	SV007	John Mullia Doe	john.doe@company.co.uk	Computer Science
6	SV006	John Doe	john.doe@gmail.com	Information Technology
5	SV005	David Wilson	david.w@email.com	Computer Science
4	SV004	Sarah Davis	sarah.d@email.com	Data Science
3	SV003	Michael Brown	michael.b@email.com	Software Engineering
2	SV002	Emily Johnson	emily.j@email.com	Information Technology
1	SV001	John Doe	john.smith@email.com	Computer Science

If tried to access delete action through URL, will be blocked also



## Exercise 7:

Admin has add new student function



## Exercise 8:

Add change password button to change the password of the account

The image shows two screenshots of a web application. The top screenshot is a modal titled "Change Password" with a yellow warning icon. It contains three input fields: "Current Password", "New Password", and "Confirm New Password", each with a "Show/Hide" toggle. Below the fields is a purple "Change Password" button and a link to "Back to Dashboard". The bottom screenshot is the main dashboard for an "Admin User". It features a header with "Student Management System" and navigation links for "Admin User", "Reports", "Settings", and "Logout". The dashboard content includes a welcome message, a student count of 11, and a "Quick Actions" section with buttons for "View All Students", "Add New Student", "Search Students", and "Change Password".

Full code explanation:

Access website,  
localhost:8080/StudentManagementMVC

AuthFilter.java get the cookie and session of the browser.

```

@Override
public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
    throws IOException, ServletException {

    HttpServletRequest httpRequest = (HttpServletRequest) request;
    HttpServletResponse httpResponse = (HttpServletResponse) response;

    String requestURI = httpRequest.getRequestURI();
    String contextPath = httpRequest.getContextPath();
    String path = requestURI.substring(contextPath.length());

    // Check if this is a public URL
    if (isPublicUri(path)) {
        // Allow access to public URLs
        chain.doFilter(request, response);
        return;
    }


    // Check if user is logged in
    HttpSession session = httpRequest.getSession(false);
    boolean isLoggedIn = (session != null && session.getAttribute("user") != null);

    if (isLoggedIn) {
        // User is logged in, allow access
        chain.doFilter(request, response);
    } else {
        // User not logged in, redirect to login
        String loginURL = contextPath + "/login";
        httpResponse.sendRedirect(loginURL);
    }
}

```

If user is logged in and the session still available, then grant access, else, navigate to login.

User login into the webpage with user name and password. Click remember me for faster login next time.

 **Login**

Student Management System

✖ Invalid username or password

Username

admin

Password

.....|

☒ Remember me

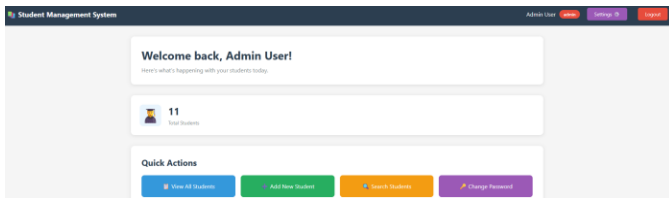
Login

Demo Credentials:

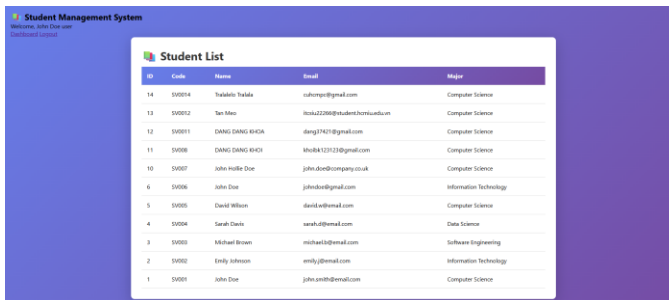
Admin: username: admin / password: password123

User: username: john / password: password123

After login, check what is the user role, if admin, show admin dashboard.



If user, show user dashboard



User don't have permission to perform actions of admin role.  
This is blocked by AdminFilter.java

```
@Override
public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)
    throws IOException, ServletException {

    HttpServletRequest httpRequest = (HttpServletRequest) request;
    HttpServletResponse httpResponse = (HttpServletResponse) response;

    String action = httpRequest.getParameter("action");

    // Check if this action requires admin role
    if (isAdminAction(action)) {
        HttpSession session = httpRequest.getSession(false);

        if (session != null) {
            User user = (User) session.getAttribute("User");

            if (user != null && user.isAdmin()) {
                // User is admin, allow access
                chain.doFilter(request, response);
            } else {
                // User is not admin, deny access
                httpResponse.sendRedirect(httpRequest.getContextPath() +
                    "/student?action=list&error=Access denied. Admin privileges required.");
            }
        } else {
            // No session, redirect to login
            httpResponse.sendRedirect(httpRequest.getContextPath() + "/login");
        }
    } else {
        // Not an admin action, allow access
        chain.doFilter(request, response);
    }
}
```



Admin can do many actions such as edit, add, delete students.

Student Management System

Admin Dashboard

Dashboard

Students

Teachers

Subjects

Classes

Reports

Settings

Logout

Student List

Add New Student

ID	Code	Name	Email	Major	Action
14	SD0014	Isabella Taylor	isabella@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
13	SD0012	Tan Mei	tanmei22264@student.honku.edu.cn	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
12	SD0011	DANG DANG KHOA	dang17471@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
11	SD008	DANG DANG KHOA	khob415121@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
10	SD007	John Hoffer Doe	john.doe@company.co.uk	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
6	SD006	John Doe	john.doe@gmail.com	Information Technology	<a href="#">Edit</a> <a href="#">Delete</a>
5	SD005	David Wilson	david.wilson@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>
4	SD004	Sarah Davis	sarah.d@gmail.com	Data Science	<a href="#">Edit</a> <a href="#">Delete</a>
3	SD003	Michael Brown	michael.b@gmail.com	Software Engineering	<a href="#">Edit</a> <a href="#">Delete</a>
2	SD002	Emily Johnson	emily.j@gmail.com	Information Technology	<a href="#">Edit</a> <a href="#">Delete</a>
1	SD001	John Doe	john.doe@gmail.com	Computer Science	<a href="#">Edit</a> <a href="#">Delete</a>

Both admin and user can change password

Change Password

Current Password

Enter current password

New Password

Enter new password

Confirm New Password

Re-enter new password

Change Password

← Back to Dashboard