Project 3: CMUQFlix

CMUQ’s Movies Recommendation System
Project Objective

1. Set up a front-end website with PostgreSQL back-end

2. Allow users to login, “like” movies, and get personalized movies recommendations
Three-Tier Architecture of CMUQFlix

Front-end Tier

Web Server

Tomcat

Servlets, JSPs, Java

Middle Tier

Back-end Tier

PostgreSQL Database

Movies, Users, ..
Last Recitation

✓ **CMUQFlix**
  - Movie recommendation website
  - Adopts a three-tier architecture with front-end, middle, and back-end tiers

✓ **HTML Forms**
  - Used for getting user input

✓ **GET vs. POST Requests**
  - Used for requesting HTML pages from a web server
  - Differ in the way user data is passed to the web server

✓ **Tomcat**
  - An open-source web server
  - Serves static and dynamic HTML pages requested via GET/POST requests
  - Acts as a servlet container i.e. can invoke servlets/JSPs to generate dynamic HTML
Java Servlets

- Java classes used to generate dynamic HTML
  - They process received user input and output the appropriate HTML
- Can manage sessions and embed cookies
  - More on this shortly
- Used for complex computation and simple HTML

JavaServer Pages (JSPs)

- Static HTML pages that embed Java code
  - They process received user input using Java code and output the HTML
- Used for complex HTML and simple computation
Today

✓ Cookie management
✓ Communication between the middle & back-end tiers
What are Cookies?

• Cookies are used for:
  – Uniquely identifying users
    • E.g. a website can display the username of a returning user
  – Storing and retrieving user preferences
    • E.g. a website can remember language preference per user

• POP Quiz:
  – Aren’t IP addresses sufficient to identify users?
    • No. Clients can possess different IPs at different times and many clients may share the same IP
  – Can’t program variables store user preferences?
    • No. Variables can retain their stored values across different sessions
Cookie Management (contd.)

• Cookies from different websites are stored and managed by a user’s web browser

• Browsers typically
  – Limit a cookie size to 4K
  – Store a minimum of 20 cookies/website
  – Store a maximum of 300 cookies/user
  – Delete cookies at the end of a browsing session* (i.e. until the user closes the web browser)
  – Store the domain that embedded a cookie

*Note: To make cookies persist beyond the current browsing session, a web-server must call the `setMaxAge` method that sets the cookie’s lifetime in seconds
public class Login extends HttpServlet
{
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException
    {
        boolean newUser = true;
        Cookie unameCookie = null;

        String username = request.getParameter("username");
        String password = request.getParameter("password");

        /* Returns an array of all the cookies every sent by the
domain (eg. andrewid-db.qatar.cmu.local) to the user */
        Cookie allCookies[] = request.getCookies();

        /* Check if a username cookie has already been sent by the
server. If so, do not create and send such a cookie */
        if(allCookies != null && allCookies.length != 0)
        {
            for (Cookie c : allCookies)
            {
                if(c.getName().equals("username"))
                {
                    newUser = false;
                    unameCookie = c;
                }
            }
        }

        /* Create a new username cookie to store the username */
        if((newUser == true) && (username.length() != 0))
        {
            unameCookie = new Cookie("username", username);
            unameCookie.setMaxAge(86400); /* one day in s */
            /* addCookie must precede getWriter!!!*/
            response.addCookie(unameCookie);
        }

        /* Append the desired HTML body to the response */
        response.setContentType("text/html");
        PrintWriter outstream = response.getWriter();

        *For more information about cookies, study Java’s Cookie API
Middle to Back-End Communication

- Copy JDBC41 Postgresql Driver, Version 9.3-1101 to cmuqflix/WEB-INF/lib
- Copy DBConnection.java from Recitation 4 to cmuq/WEB-INF/src
- Modify the package name to local.cmu.qatar.db
- Compile DBConnection.java and copy DBConnection.class to cmuq/WEB-INF/classes/local/cmu/qatar/db/