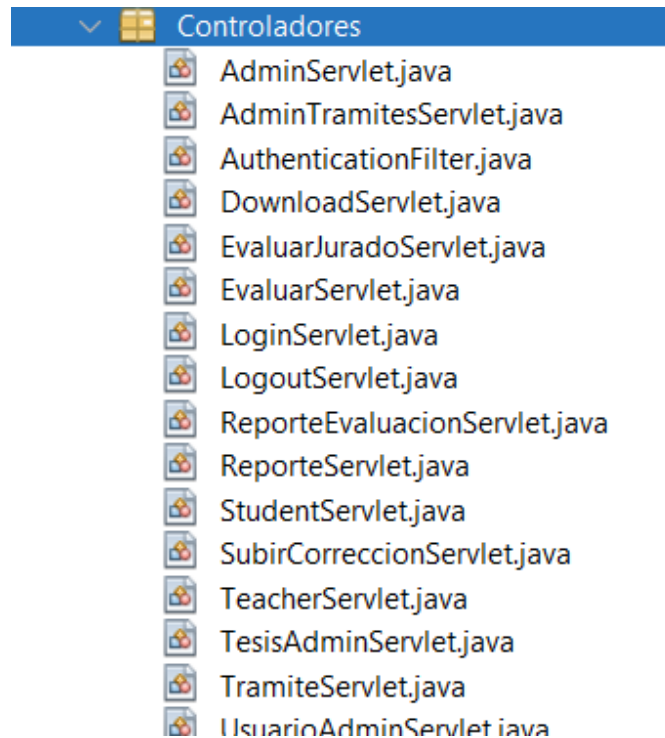


CONTROLADOR



```
package Controladores;

import Datos.TesisDAO;
import Datos.TramitesDAO;
import Datos.UsuarioDAO;
import Modelos.DocumentoRequisito;
import Modelos.Notificacion;
import Modelos.Sustentacion;
import Modelos.Tesis;
import Modelos.Tramite;
import Modelos.Usuario;
import java.io.IOException;
import java.sql.SQLException;
import java.util.List;
import jakarta.servlet.RequestDispatcher;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import jakarta.servlet.http.HttpSession;

@WebServlet(name = "AdminServlet", urlPatterns = {"/AdminServlet"})
public class AdminServlet extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

        HttpSession session = request.getSession(false);
        if (session == null || session.getAttribute("usuarioLogueado") == null) {
            response.sendRedirect("login.jsp");
            return;
        }
        Usuario adminUser = (Usuario) session.getAttribute("usuarioLogueado");

        // --- LOGICA DE FLASH SCOPE (URL LIMPIA) ---
    }
}
```

```

package Controladores;

import Datos.TramitesDAO;
import Modelos.DocumentoRequisito;
import Modelos.Sustentacion;
import Modelos.Usuario;
import Servicios.NotificacionService;
import java.io.IOException;
import java.sql.Timestamp;
import java.util.List;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import jakarta.servlet.http.HttpSession;

@WebServlet(name = "AdminTramitesServlet", urlPatterns = {"/AdminTramitesServlet"})
public class AdminTramitesServlet extends HttpServlet {

    private NotificacionService notificacionService;

    @Override
    public void init() throws ServletException {
        notificacionService = new NotificacionService();
    }

    // Método auxiliar para redirección limpia y seteo de tab activo
    private void redirigirAdmin(HttpServletRequest response, HttpSession session, String tab) throws IOException {
        session.setAttribute("flash_activeTab", tab);
        response.sendRedirect("AdminServlet");
    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    }
}

```

```

package Controladores;

import java.io.IOException;
import jakarta.servlet.Filter;
import jakarta.servlet.FilterChain;
import jakarta.servlet.FilterConfig;
import jakarta.servlet.ServletException;
import jakarta.servlet.ServletRequest;
import jakarta.servlet.ServletResponse;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import jakarta.servlet.http.HttpSession;

/**
 * Filtro de autenticación y autorización para la aplicación.
 */
public class AuthenticationFilter implements Filter {

    @Override
    public void init(FilterConfig filterConfig) throws ServletException {
        // Inicialización (si es necesaria)
    }

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

        HttpServletRequest httpRequest = (HttpServletRequest) request;
        HttpServletResponse httpResponse = (HttpServletResponse) response;
        HttpSession session = httpRequest.getSession(false); // No crear una nueva sesión

        String path = httpRequest.getServletPath();

        // --- 1. Rutas Públicas (Excluidas del filtro) ---
        // Permitir acceso al LoginServlet, LogoutServlet y a la página de login
        if (path.equals("/login.jsp") || path.equals("/LoginServlet") || path.equals("/LogoutServlet")) {
            chain.doFilter(request, response);
        }
    }
}

```