CS814 Course Project Report

Submitted by

ARIJIT KARALI (202IS005)

ASHISH HARIYALE (202IS006)

Department of Computer Science and Engineering

National Institute of Technology Karnataka

P.O. Srinivasnagar, Surathkal

Mangalore-575025 Karnataka, India

January 2021

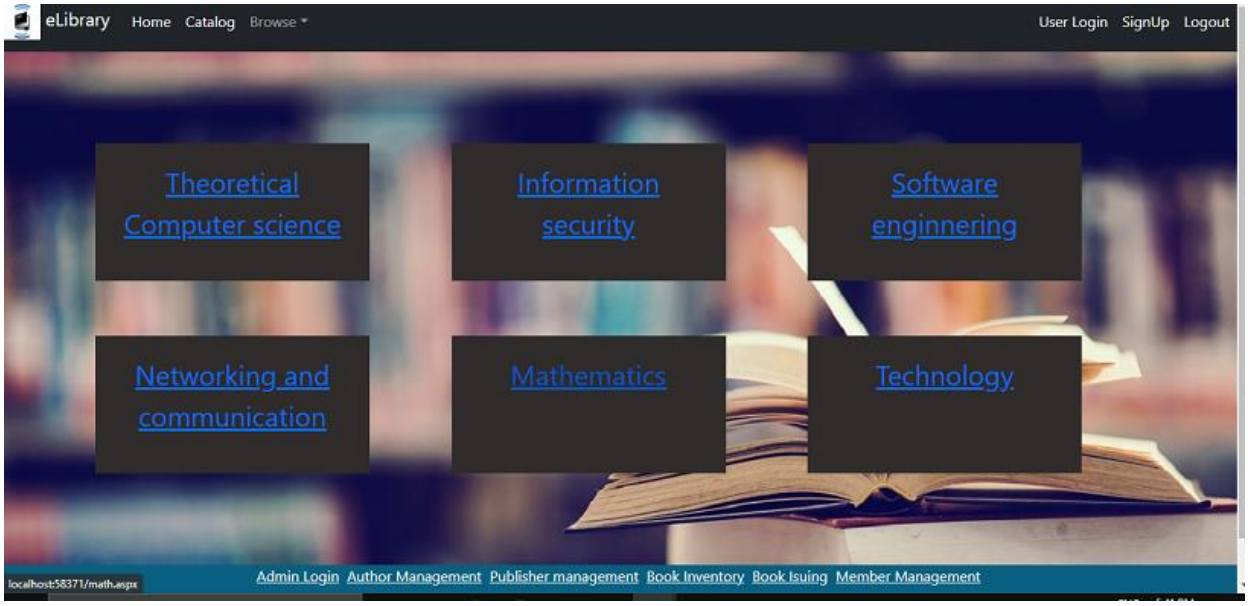
CONTENTS:

INTRODUCTION

Application is an e journal management system.

It contains 6 sections namely

1. Theoretical Computer science
2. Information security
3. Software engineering
4. Mathematics
5. Networking and communication
6. Technology



On this 6 broad fields the website is divided and the articles are distributed into. Each section is independent of other section. Each section is administered by their own editors.

The website can be used by readers to read articles or as an author to submit new papers.

USE OF APPLICATION, DIFFERENT USERS AND FUNCTIONALITIES

Use:

- Readers to read articles.
- Authors to upload new papers.
- Editors/Reviewers to manage/administer journal.

Different users:

Member type

1. Reader
2. Author

Admin type:

1. Reviewer
2. Editor
3. Editor-in-chief/manager.

Functionalities:

Readers can view different types of articles in 6 different sections.

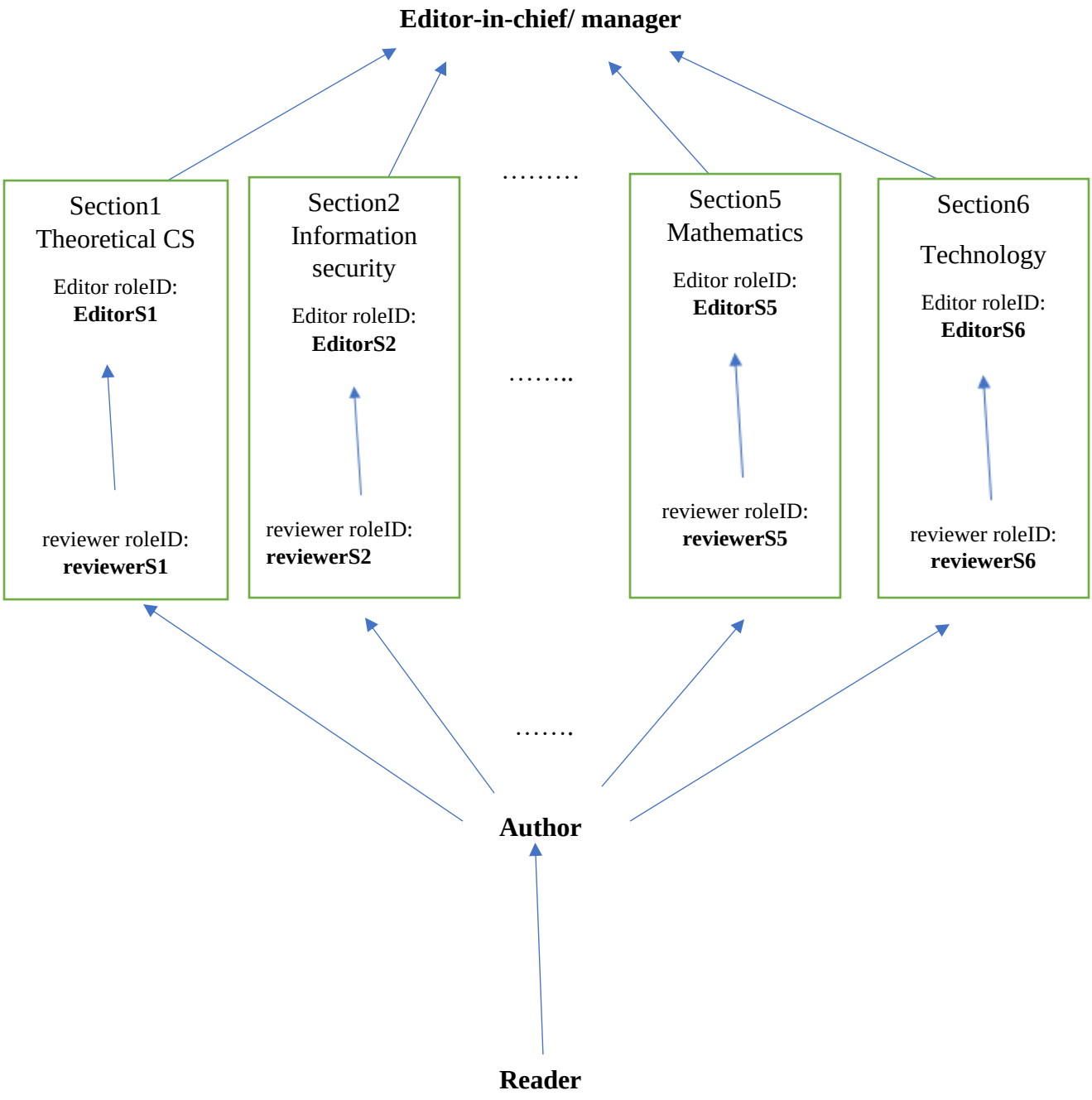
Authors can make new submissions and a get a review by the editors and reviewers under a particular section.

Editors are responsible for administering 1 section. Under his administration, he can accept new submissions and forward the submission request to peer reviewers by assigning two reviewers.

Also the **reviewer**s are responsible for giving reviews on every article they are being passed on by the editor in his section.

Editor-in-chief or manager are capable of adding deleting updating section in the website. Also they have authority to administer section editors, reviewers, authors.

ROLE HIERARCHY IN EJOURNAL MANAGEMENT



ACCESS POLICY OF DIFFERENT ROLES:

Policy1: Readers can only view different section but cannot download/upload manuscript.

Policy 2: userprofile pages and other pages must only be accessable after login.

To authenticate the user.

Policy 3: important data folders which hold informations related to articles submitted but yet not published are kept hidden/unaccessable for unauthorized users and also for non admin roles like readers and authors. For example, authors/readers will not be allow to read what are the articles that are submitted but currently under review.

Policy 4: Editors and reviewers under a section are isolated from another section and their editors and reviewers. Every section is administered by a predefined sectioneditor (role:editorS1 for section1).

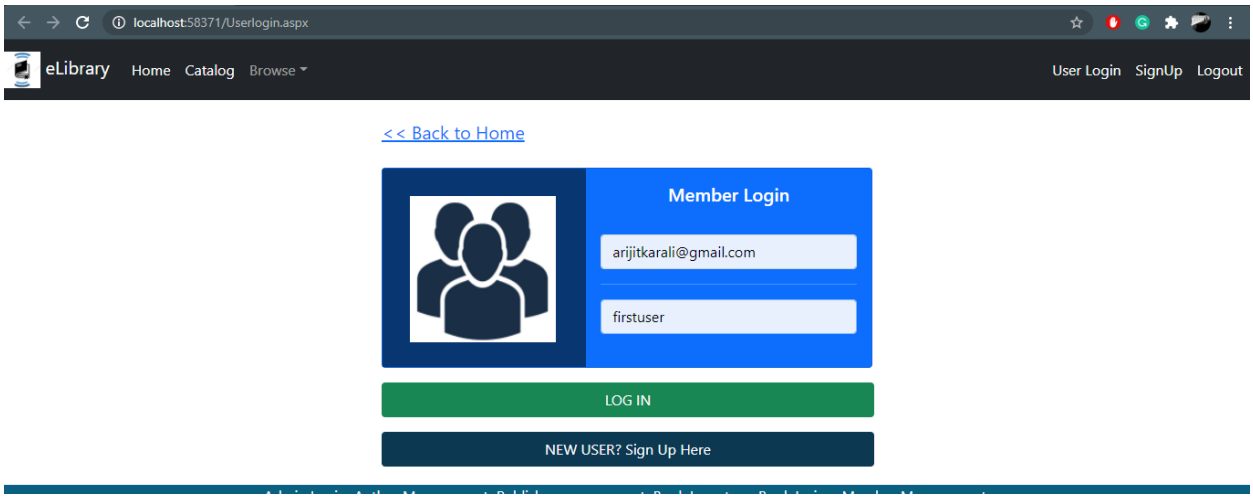
Policy 5: Editor-in-chief/Journal manager can inherit all roles from his descendants.

Example of access control demo

POLICY2 demo

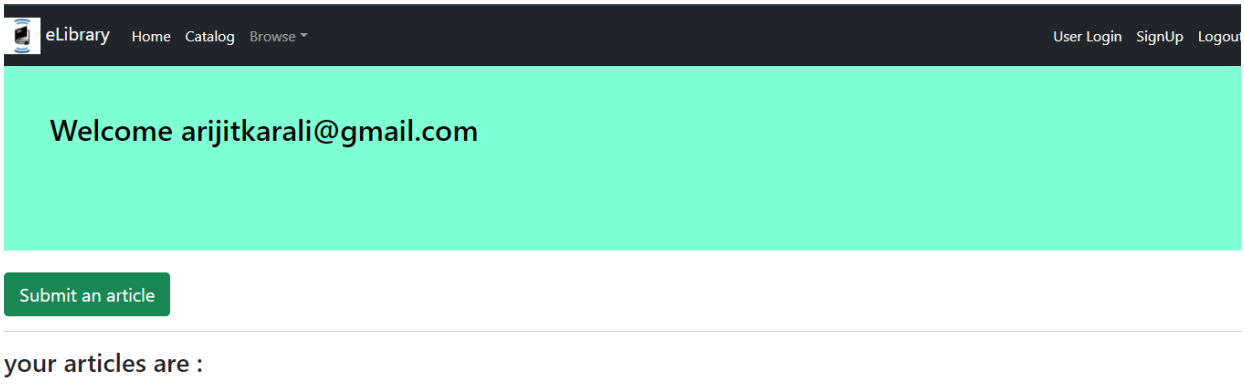
Reader login

Authentication is done using id,password and creating session

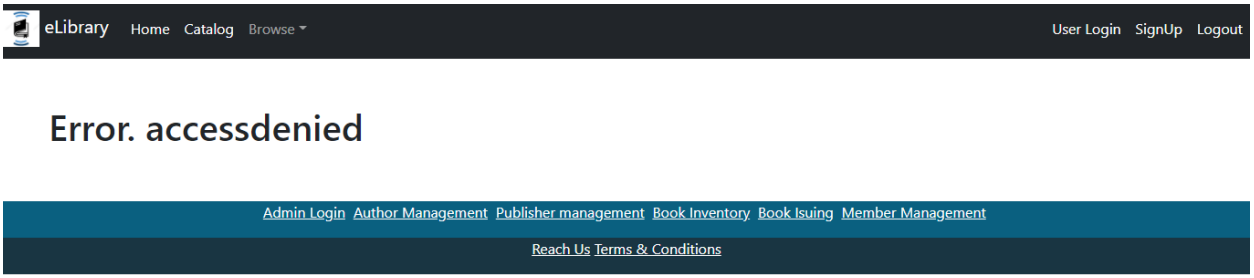



```
            Session["userid"] = TextBox1.Text;
            Response.Redirect("userprofile.aspx");
        }
```

2. access to profilepage is granted



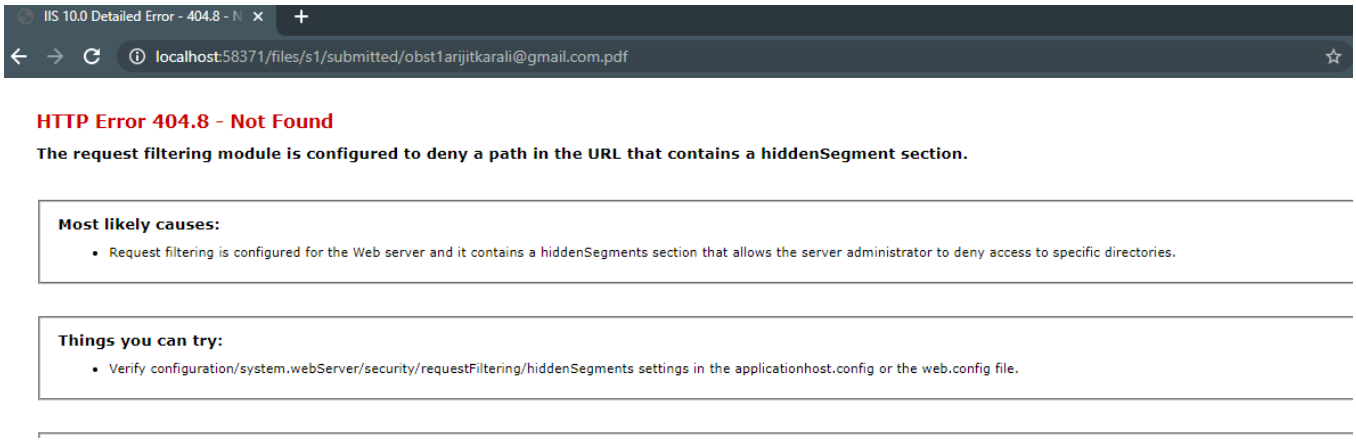
Access is denied for unauthorized access (without login) using sessions



POLICY3 DEMO

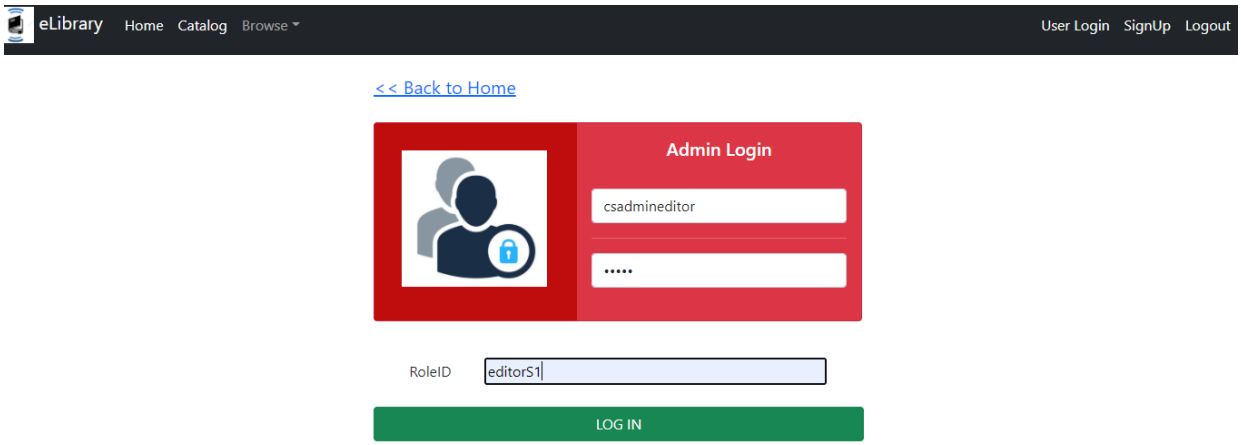
Keeping the directory “files” classified using web.config

```
</system.web>
<system.webServer>
  <security>
    <requestFiltering>
      <hiddenSegments>
        <add segment="files"/>
      </hiddenSegments>
    </requestFiltering>
  </security>
</system.webServer>
```



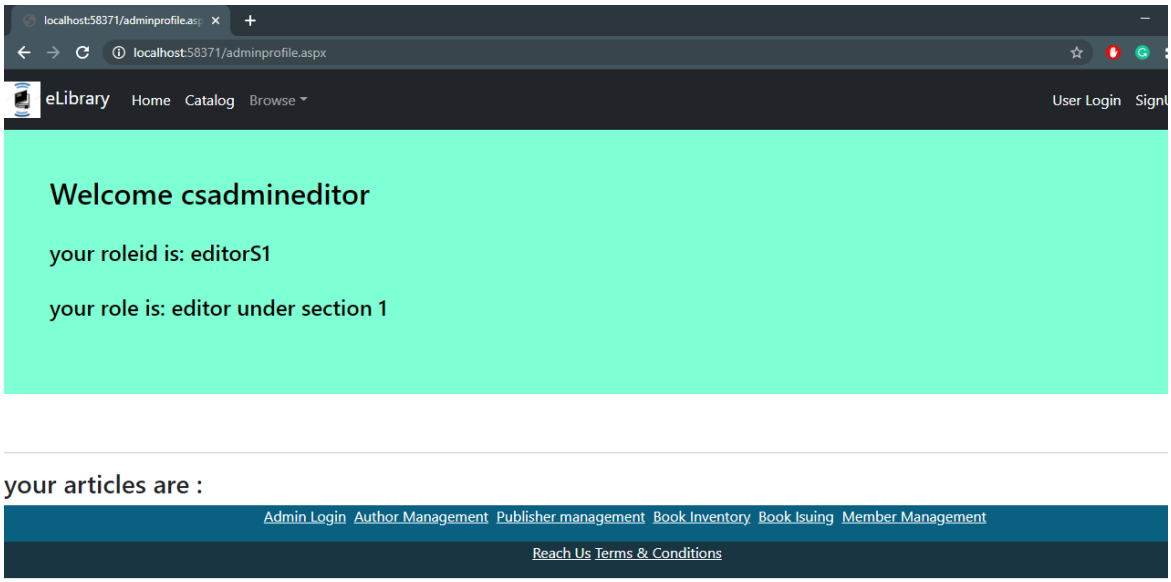
POLICY4 DEMO

Editors and reviewer access controls are done using adminid, password and roleid and session and storing <adminid,password,roleID> into database.




```
        }
        Session["adminid"] = TextBox1.Text;
        Session["roleid"] = TextBox3.Text;
        Response.Redirect("adminprofile.aspx");
    }
```

```
16      {
17          if (Session["adminid"] != null)
18          {
19              adminid = Session["adminid"].ToString();
20              Label1.Text = "Welcome " + adminid;
21              Label1.Visible = true;
22          }
23          else
24          {
25              Response.Redirect("errorpage.aspx?error=accessdenied");
26          }
27          if(Session["roleid"].ToString()!=null)
28          {
29              roleid = Session["roleid"].ToString();
30              Label2.Text = "your roleid is: "+roleid;
31              Label2.Visible = true;
32          }
33
34          if (roleid == "editorS1")
35          {
36
37              Label3.Text = "your role is: editor under section 1";
38              Label3.Visible = true;
39          }
```



your articles are :

Conclusion

The proposed project on e-journal system is a good example of how Role based access control is important in real world applications. This system contains a set of roles and forms a role hierarchy. The policies are minutely being examined and appropriate access controls are added. This sets a good implementation of RBAC ideas on multiple set of users.

Project rbac is implemented using c# session, .net web configuration features, roleid, id password with database and front end on asp.net and bakend on c#.