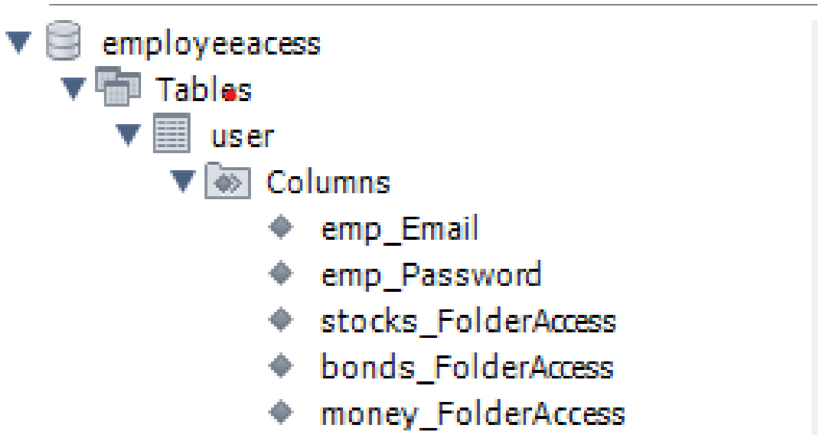
File Management System by Muhammad Taha

Project Specifications

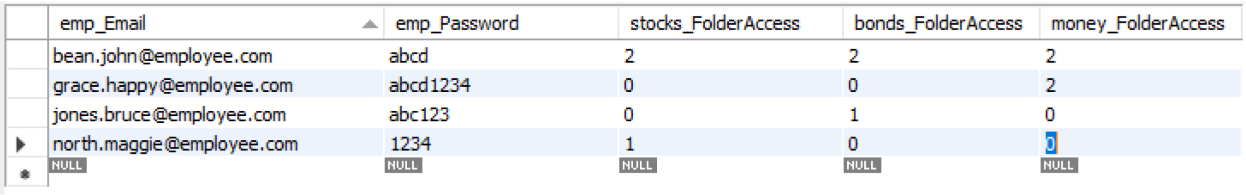
Implement a website that allows employees to access company resources via a file server. Once the employee is cleared for access to private company files, they will be able to log into the company site and download those files.

Progress Log

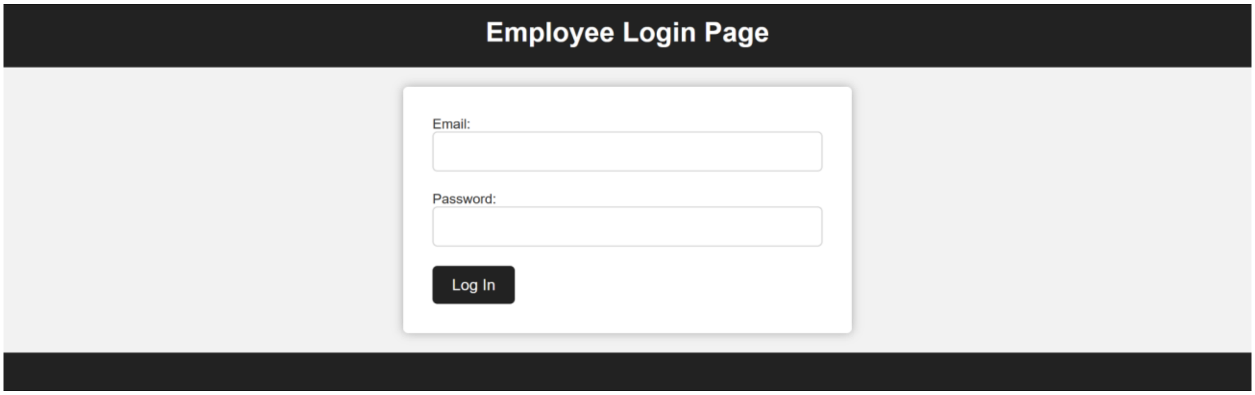
Employee database with login email addresses and passwords. And folder access columns with 0 signifying no access, 1 signifying read access, and 2 signifying read and write access.

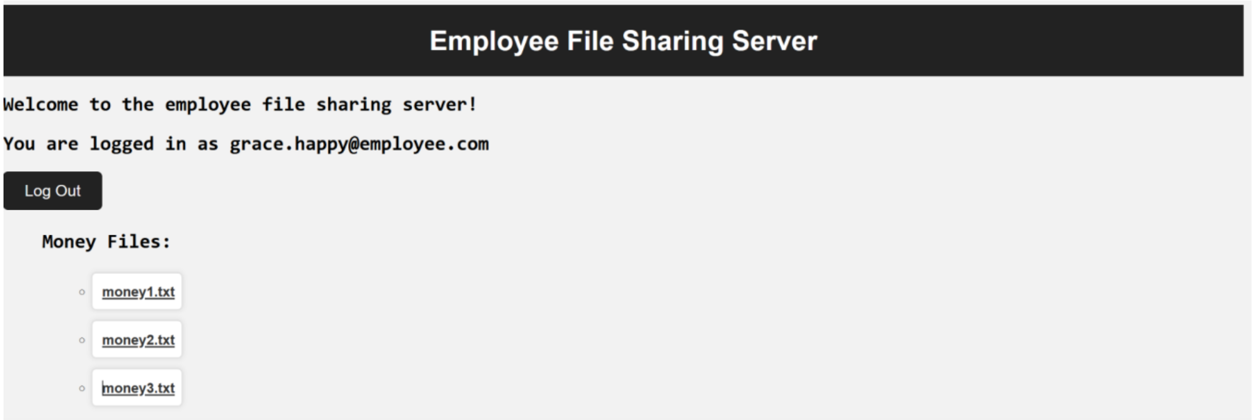


- employeeacess
  - Tables
    - user
      - Columns
        - emp_Email
        - emp_Password
        - stocks_FolderAccess
        - bonds_FolderAccess
        - money_FolderAccess



	emp_Email	emp_Password	stocks_FolderAccess	bonds_FolderAccess	money_FolderAccess
	bean.john@employee.com	abcd	2	2	2
	grace.happy@employee.com	abcd1234	0	0	2
	jones.bruce@employee.com	abc123	0	1	0
▶	north.maggie@employee.com	1234	1	0	0
*	NULL	NULL	NULL	NULL	NULL





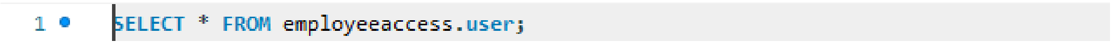
Welcome to the employee file sharing server!

You are logged in as grace.happy@employee.com

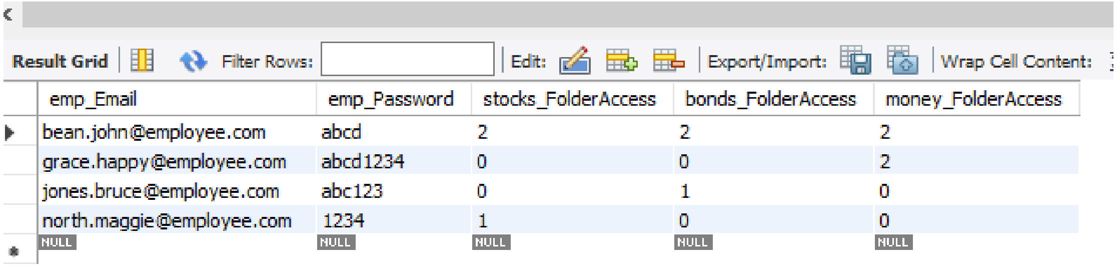
Money Files:

- money1.txt
- money2.txt
- money3.txt

Create a Table named “user” with five columns.



```
SELECT * FROM employeeaccess.user;
```

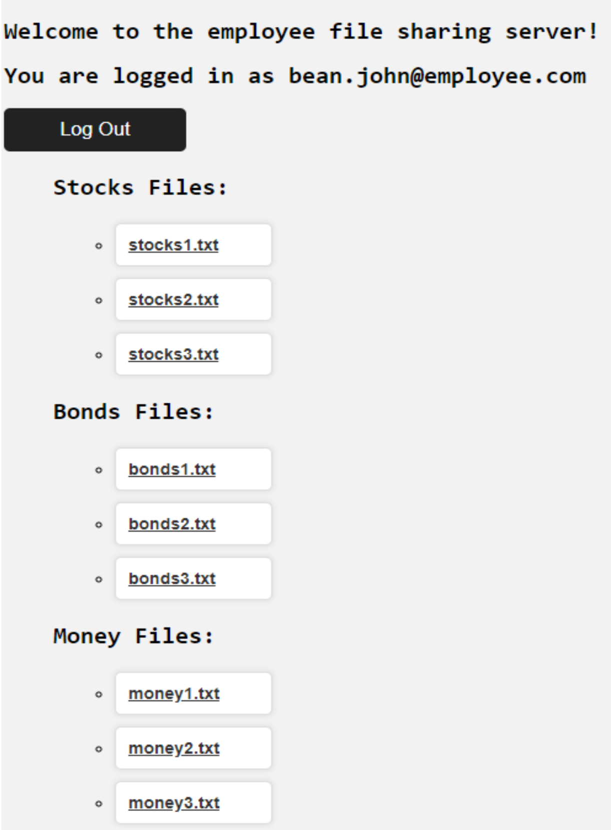


emp_Email	emp_Password	stocks_FolderAccess	bonds_FolderAccess	money_FolderAccess
bean.john@employee.com	abcd	2	2	2
grace.happy@employee.com	abcd1234	0	0	2
jones.bruce@employee.com	abc123	0	1	0
north.maggie@employee.com	1234	1	0	0
NULL	NULL	NULL	NULL	NULL

Test Log in for each user

John Bean

- User name and password works
- This user has correct read and write permissions to all files



Welcome to the employee file sharing server!

You are logged in as bean.john@employee.com

Stocks Files:

- stocks1.txt
- stocks2.txt
- stocks3.txt

Bonds Files:

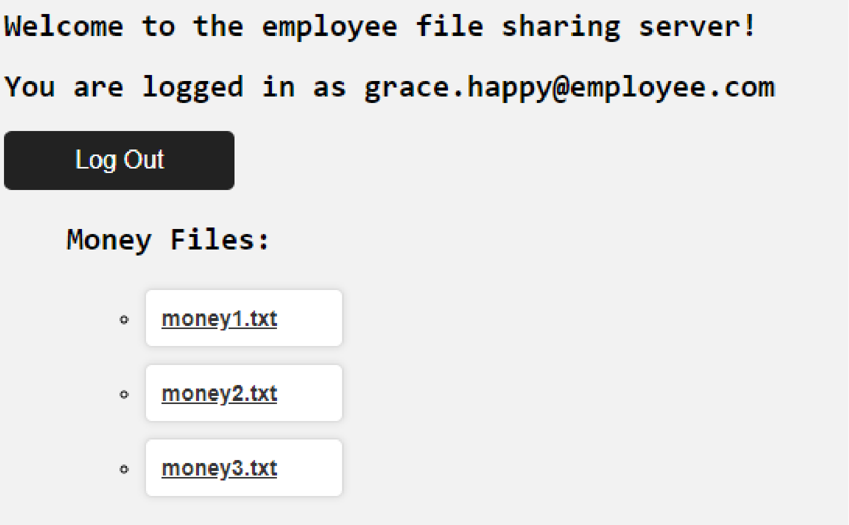
- bonds1.txt
- bonds2.txt
- bonds3.txt

Money Files:

- money1.txt
- money2.txt
- money3.txt

Grace Happy

- Username and password works
- Correct read and write permissions are applied to money file



Welcome to the employee file sharing server!

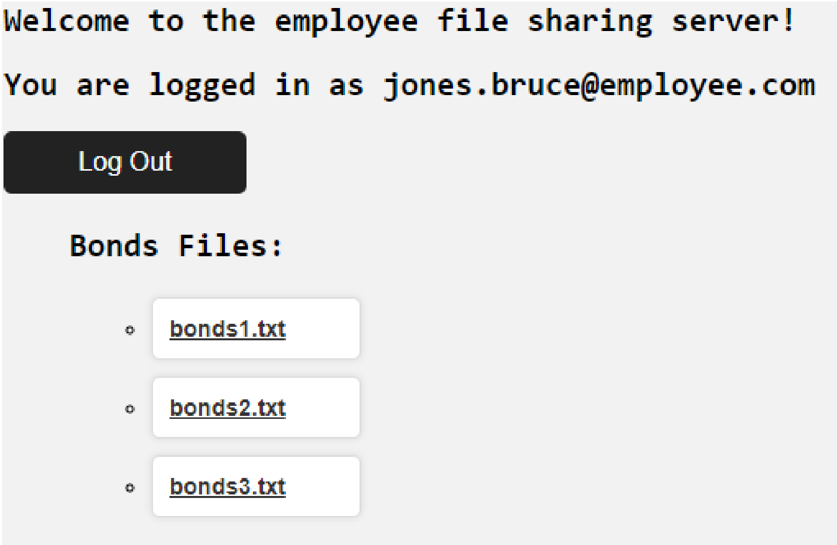
You are logged in as grace.happy@employee.com

Money Files:

- money1.txt
- money2.txt
- money3.txt

Bruce Jones

- Username and password works
- Can only read bonds. It appears that this user can write to each file even though the correct permissions are applied in MySQL



Welcome to the employee file sharing server!

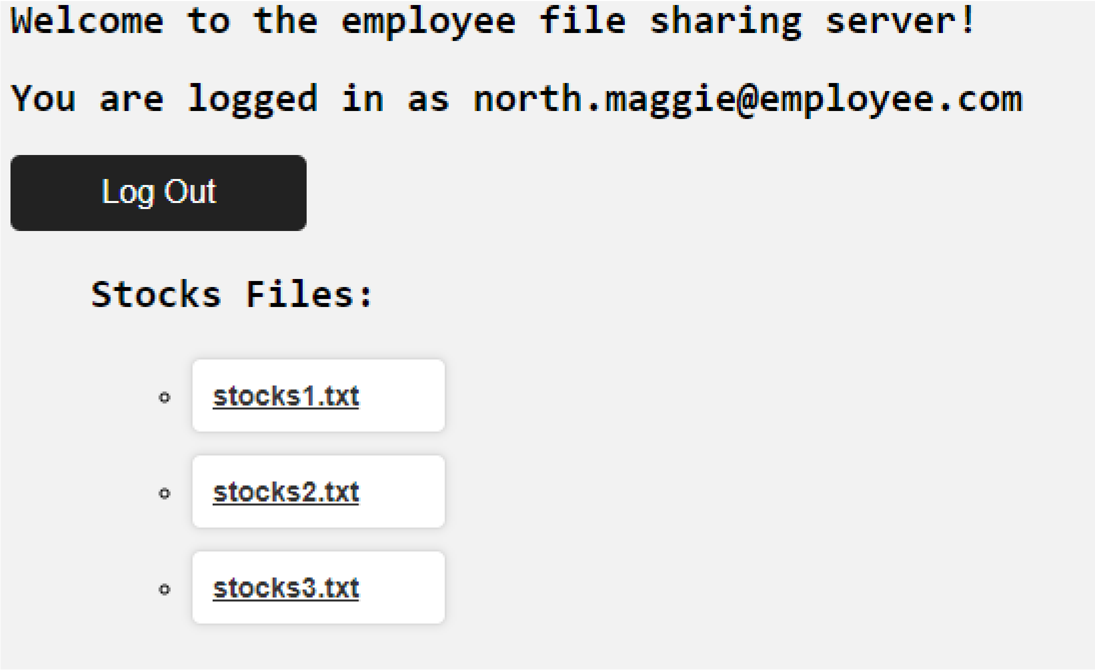
You are logged in as jones.bruce@employee.com

Bonds Files:

- bonds1.txt
- bonds2.txt
- bonds3.txt

Maggie North

- Username and password words
- Can read only stocks however it appears that this user can also write to the file as well



Welcome to the employee file sharing server!

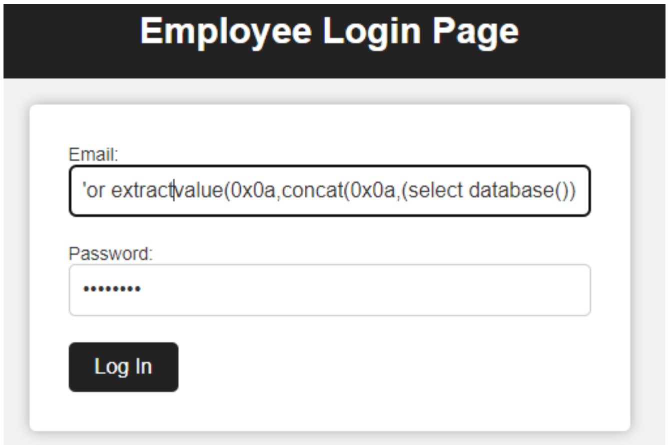
You are logged in as north.maggie@employee.com

Stocks Files:

- stocks1.txt
- stocks2.txt
- stocks3.txt

Other Testing and Security Strategy

- I attempted SQL injection to log in and I received an error message indicating to use an “@” symbol
- This is because secure coding practices were used such as input validation and verification.



Employee Login Page

Email: 'or extractvalue(0x0a,concat(0x0a,(select database())

Password: ••••••••

PROJECT SUMMARY

Project Expectations: The project’s initial plan has met the intended output. The expectation was to create an employee login page and only grant certain employees permission to read or write certain file types. With additional time, the project could be improved, for example exploring further why certain employees could “edit” a .txt file when the correct permissions were set in MySQL.

Purpose of the application: The main purpose of this web application is to provide a file management system for employees to access, download, and manage files. The application also has user authentication and authorization that allows for the secure access of user accounts with different access levels and provides a professional web interface for users to perform file operations. The project involved creating different servlets and JSP pages for different functionalities and integrating them together to create a functional web application.

Security Defense Strategies:

- Role-based access control: The application has different types of users, and each type of user has different levels of access to the application's features. This is implemented using role-based access control to ensure that users can only access the features they are authorized to use.
- Input validation: User input is validated on both the client and server-side to prevent injection attacks such as SQL injection and Cross-site scripting (XSS) attacks.

- Session management: The application uses session management to keep track of user sessions and to prevent session hijacking and other types of attacks.

Reflection: Overall, the project was a success as it met the project requirements of allowing users to access, view and download files to specific folders on the server. However, one of the main shortcomings of the project was the inability to implement more advanced features such as the upload feature due to time limitations. Despite this limitation, the project provided a good opportunity to gain experience with JSP, Servlets, and Apache Commons. It also allowed for practice in developing a web application using safe coding practices and provided a solid foundation to build upon in future projects.