Dhrumil Patel
April 21, 2016

BookStore Project

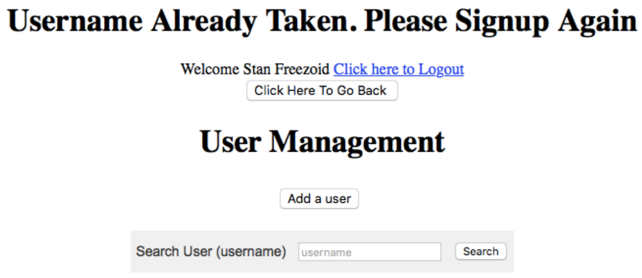
Part 3

This part of the project is more focused on backend perspective. The easiest part of the project was to convert user input queries into prepared statements. I basically had to follow the syntax. The hardest part was to implement the hash conversion and workflow. Preventing sql injection was interesting. Quotes in text user field works now. For converting and storing password in hash format, the md5Val function is created in Users class. The md5Val function takes a string and returns the corresponding md5 hash bigInteger. User can now view Book List without Login in. The program outputs an appropriate message when a username is already taken or a password is not entered according to requirements. All these functionalities has been added to admin input text fields too. No Special Configurations are required.

Sample Accounts:

UserName	Password	Account Type
StanZ	adminAcc2	admin
dpate85	wqEra123	regular
admin	test1234	admin
HarveyB	8752vCxai	regular

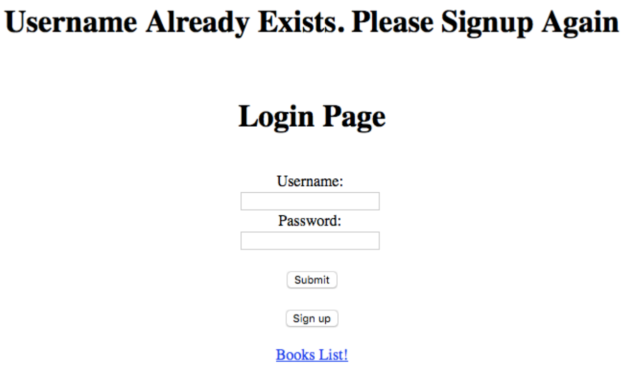
1) ***New Username:*** The first required function was to output a message when a username is already taken. This is added in admin signup page too. (Pic - Admin : Left , Regular : Right)



Username Already Taken. Please Signup Again

Welcome Stan Freezoid Click here to Logout

User Management



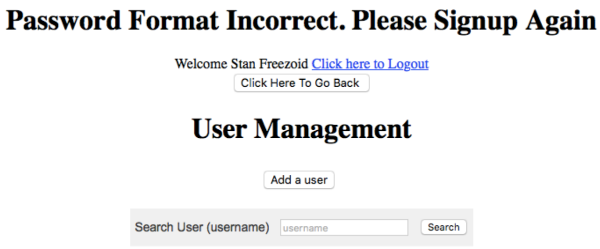
Username Already Exists. Please Signup Again

Login Page

Username:

Password:

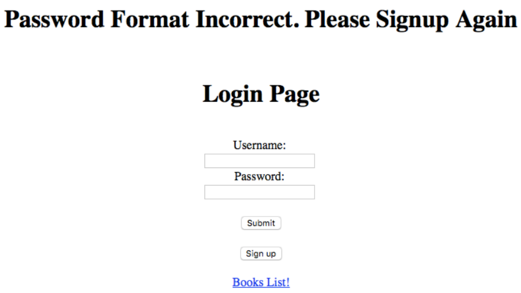
2) ***'Hard' Password :*** The user is required to enter a password of at least 8 alphanumeric characters. It must contain Letters and Digits. If any of those criteria is not met, a message appears in login page alerting the user. (Pic - Admin - Left, Regular Right)



Password Format Incorrect. Please Signup Again

Welcome Stan Freezoid Click here to Logout

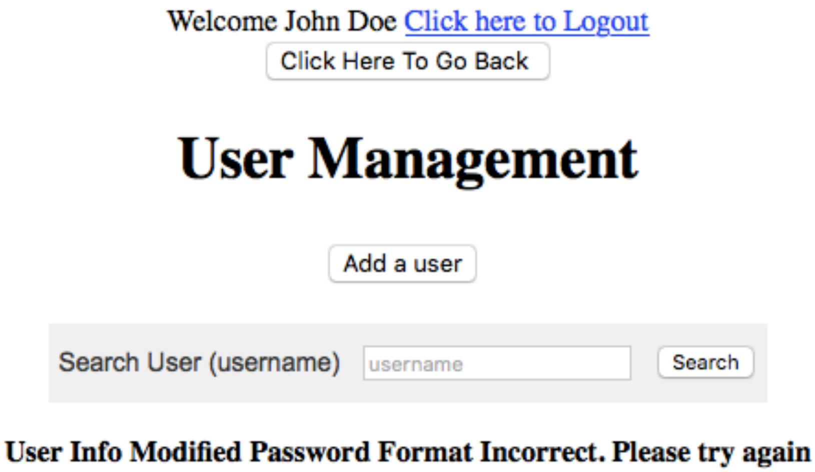
User Management



Password Format Incorrect. Please Signup Again

Login Page

- This message is also displayed when admin changes a password via user modification.



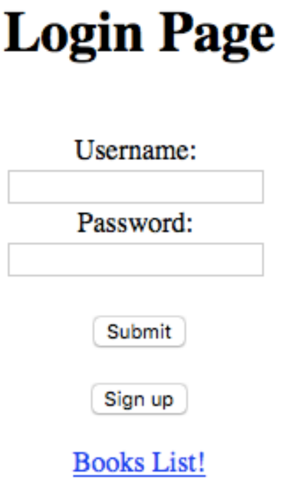
Welcome John Doe Click here to Logout

User Management

User Info Modified Password Format Incorrect. Please try again

3) Protection Against SQL injection.: First of all, I sanitized the string and then used that string in prepared statement query. But that didn't work, So i used just converted the queries into prepared statement. It seemed to escape the special characters pretty well, which resulted in prevention of sql injection. We just needed to convert the query into prepared statement if a user input was involved. So, all the queries which required/used text field were converted to prepared statement. Other queries which worked internally were not converted as there was no chance for the user to alter them via Sql Injection. The pages which changed were UserManagement.jsp, Login.jsp, Users.java, OrderStatistics.jsp, OrderStatistics2.jsp, AddUser.jsp, ShoppingCart.jsp, ShowBooks.jsp. All these pages required some kind of input from user.

4) Workflow : The project can now view Books List without logging in. But they do need to log in when order is placed. When they click order on ShowBooks.jsp page without logging in, user gets redirected to Login Page. I have added a link in Login Page to view the book list. It can also be viewed with the URL directly. A banner is showed in the top if the user is not logged in.



Clicking The "Books List!" takes you to ShowBooks.jsp page.

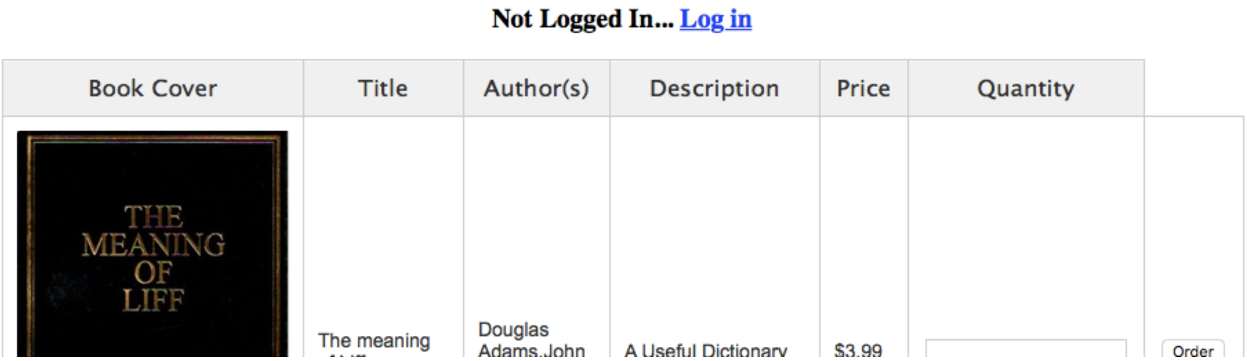
It can also be accessed with this URL

(i.e, without clicking the "Books List!") :

http://localhost:8080/bookstore/ShowBooks.jsp

The header will display "Not Logged In…"

Clicking the "Log In" will redirect to Login.jsp

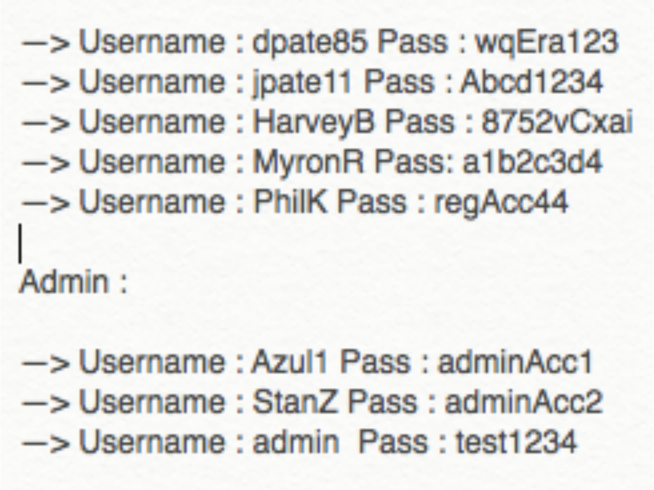


Not Logged In... Log in

Book Cover	Title	Author(s)	Description	Price	Quantity	
THE MEANING OF LIFF	The meaning	Douglas Adams,John	A Useful Dictionary	$3.99		Order

5) Hashed Password : As described above, the md5Val function converts every string to Hashed password and that gets stored to database. The password gets converted into md5 format whenever a password in entered in any page (signup and user management). All the account passwords in database currently had been converted to md5 format. Username and Pass of all accounts is in the screenshot below.

All accounts :



```
—> Username : dpate85 Pass : wqEra123
—> Username : jpate11 Pass : Abcd1234
—> Username : HarveyB Pass : 8752vCxai
—> Username : MyronR Pass: a1b2c3d4
—> Username : PhilK Pass : regAcc44

Admin :

—> Username : Azul1 Pass : adminAcc1
—> Username : StanZ Pass : adminAcc2
—> Username : admin  Pass : test1234
```