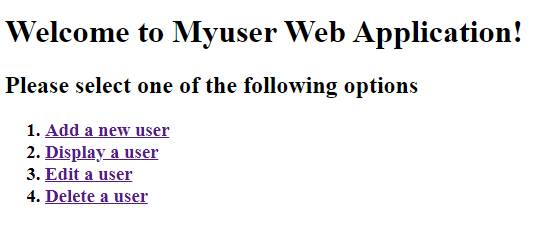
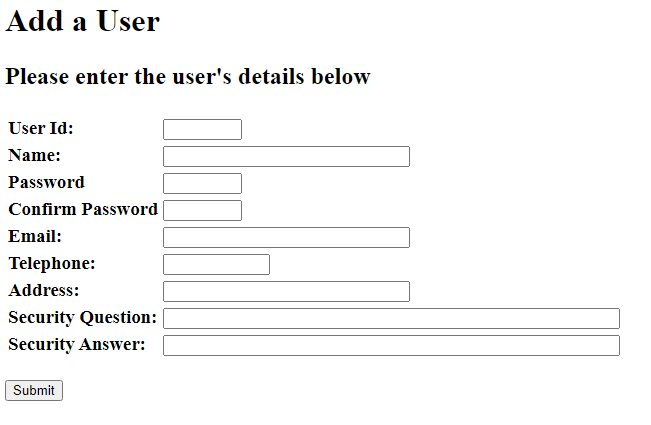
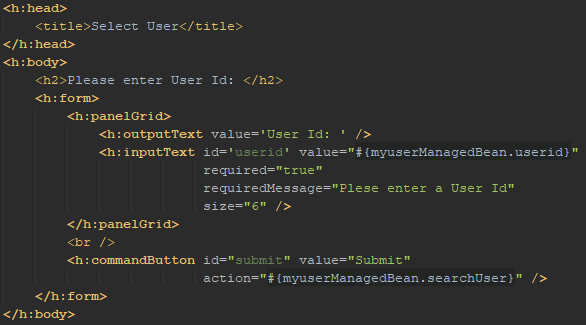
# 5.1P

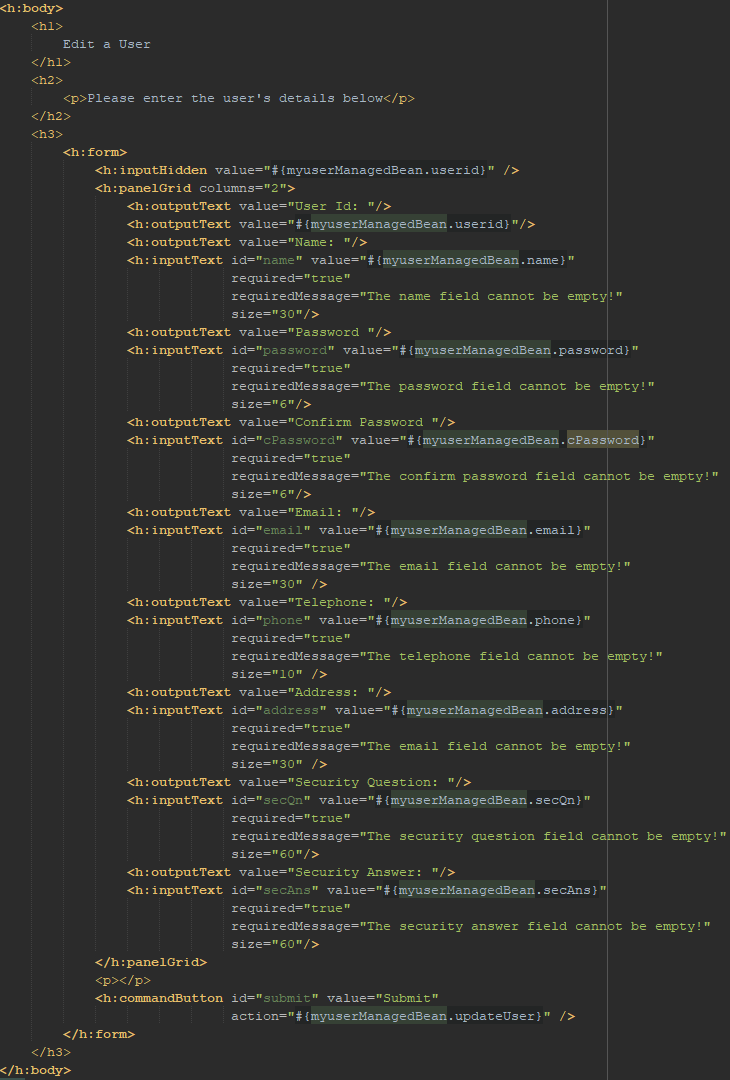
## Task 1.

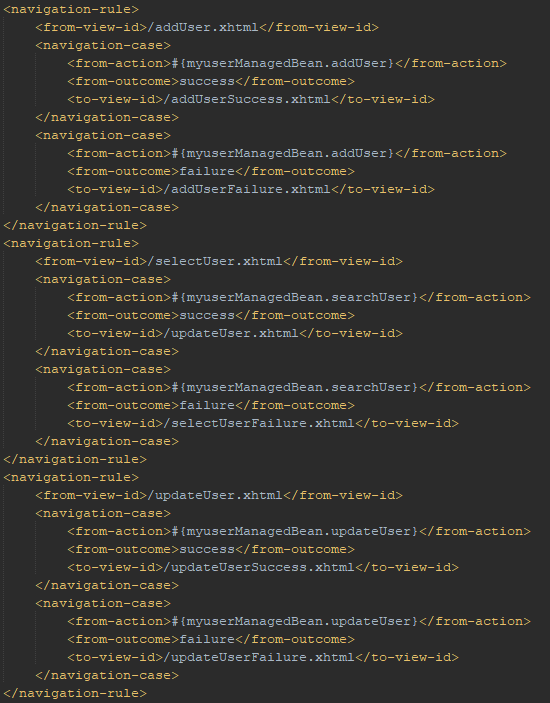
The lab was completed and the web application was created:  




## Task 2.

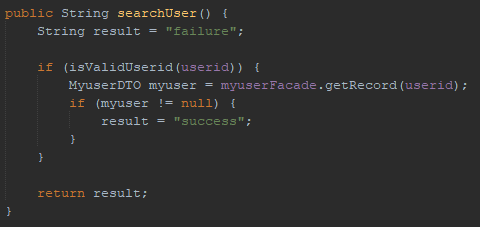
Various pages were created to facilitate the editing of a user.  
  
selectUser.xhtml:  


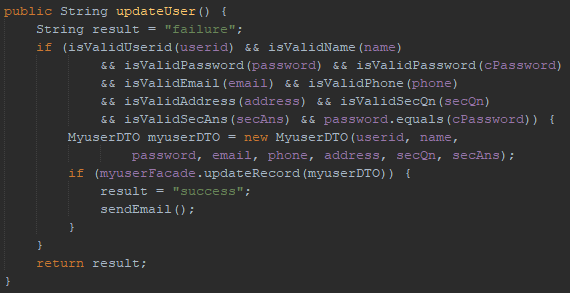
updateUser.xhtml:  


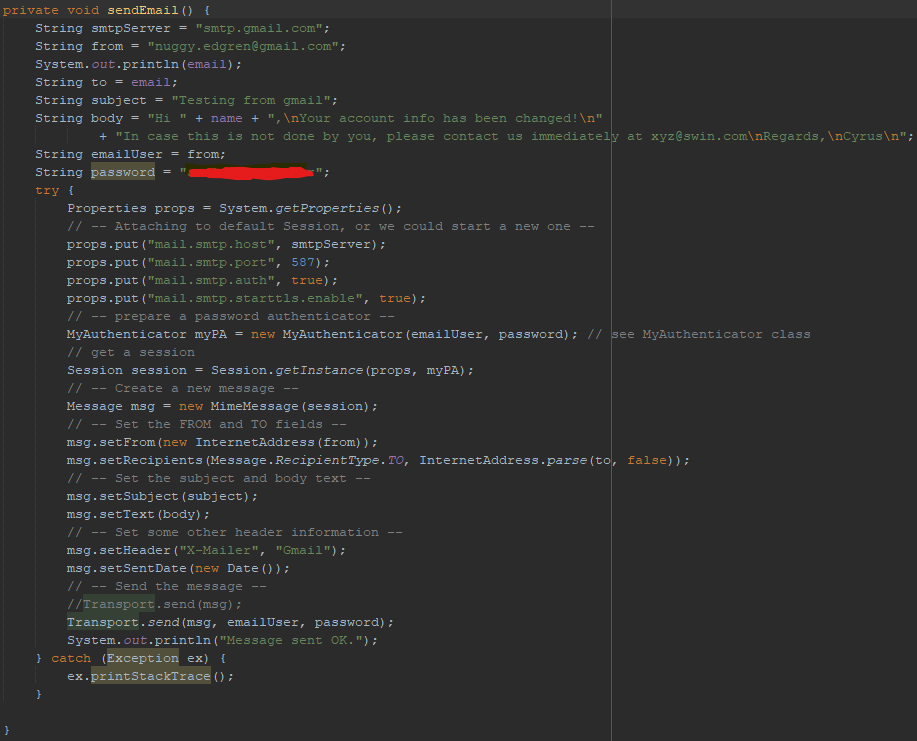
faces-config.xml:  


\*\*\* The failures and success pages were almost identical from the ones created in the lab, so for brevity they were excluded here.

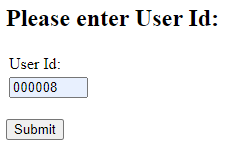
Some function needed to be added to myuserManagedBean.java also to complete this task.

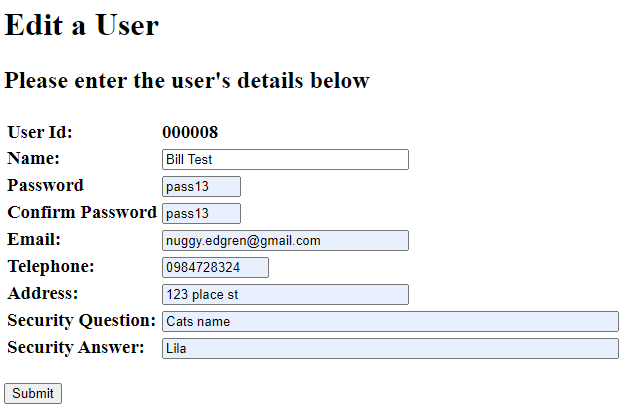
searchUser():  


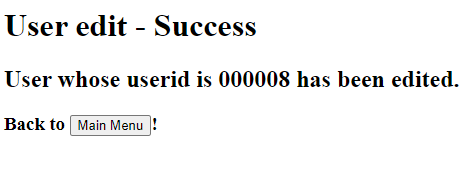
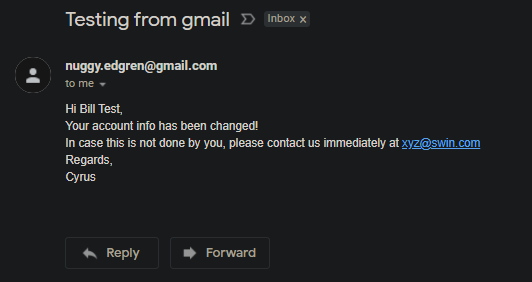
updateUser():  


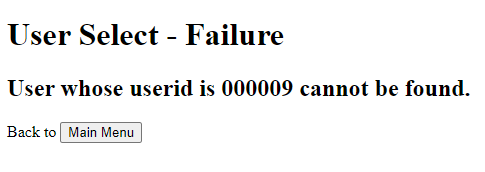
sendEmail():  


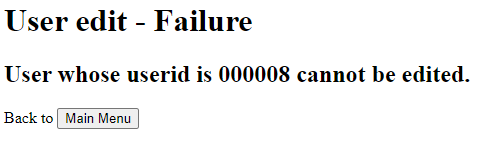
## Task 3.

When the user presses the ‘edit user’ button from the main menu they can search for a user to edit:  


If successful and the user is found the page direct to a form to update the found user:  


When this is submitted, if successful the user is notified via the web page as well as an email:  
  


If either of these steps fail, the relevant page is shown:  


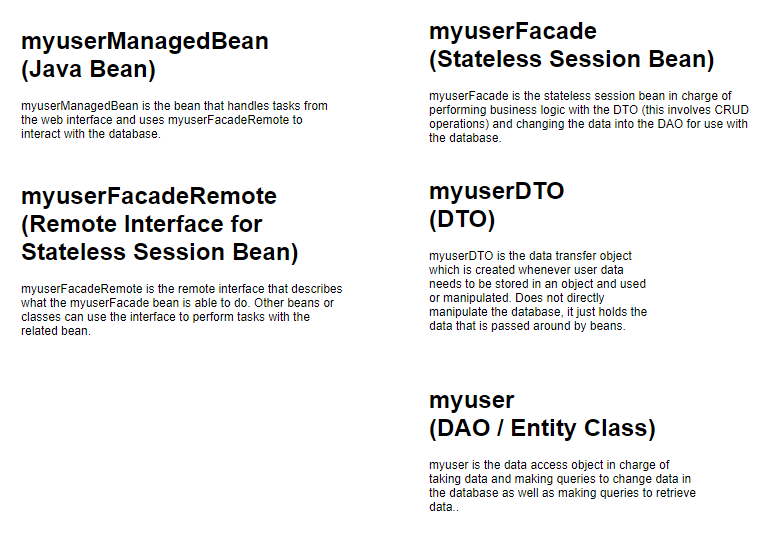


## Task 4

### 4.1

### 

### 4.2



## Task 5

### 5.1

When the user’s details have been changed I think it would be best to alert both the old and new emails of the change, this way if the change was not intended the user can take action or if it was they then know the new email was entered correctly. The largest improvement to the security of account changes would be to add two factor authentication to the login process. This could be done with a separate device or an email based solution. With protection like this the chance of an account being able to be hack and so having details changed by a third party is made much less likely. Another simpler solution would be to have a confirmation email sent to the old email address of the user and the changes would not be made unless a link was clicked or code entered from the confirmation email. While this is more secure as this would require the user’s email to also be compromised, this also introduces the problem of if the user no longer has access to the old email then the changing of account details would not be possible.

### 5.2

For the changes in the actual program, two factor authentication would probably be done more in the login/authentication part of the system which was not actually created in this task. The email confirmation could be done in the myuserManagedBean when you would send the email alerting the user to changes, instead send a confirmation email that contains a link that then triggers the rest of the database change process.