**-----Class LoginView-----**

/\*\*\* Submit login informations\* @param username a string representation of the username that users enter  
\* @param password a string representation of the password that users enter  
\* @throws IllegalArgumentException if the parameter is of wrong type  
\*/  
public void submitLoginInfo(String username, String password)  
throws IllegalArgumentException {

send username and password to controller via POST request

}

/\*\*  
\* Notifies success/fail to user  
\* @param message a string representation of the message to be notified to the user.  
\*/  
public void notifyGuest(String message)  
throws IOException{

notify message to user;

}

**-----Class LoginController-----**

/\*\*  
\* Receive username and password from the view, check if these inputs are valid or not  
\* @param username a string representation of the username that users enter

\* @param password a string representation of the password that users enter  
\* @return return true if user input is valid, otherwise return false.  
\*/  
public Boolean validateUserInput(String username, String password){

if username is empty

return false

else if password.length < 5

return false

else

return true

}

/\*\*  
\* Validate user with username and password provided  
\* @param username a string representation of the username that users enter

\* @param password a string representation of the password that users enter  
\* @return return true if user is valid, otherwise return false.  
\*/  
public Boolean validateUser(String username, String password){

if validateUserInput(username, password) != true

return false

User user = find User with username

if user is null

return false

else

if !user.validateUser(password)

return false

else

return true

}

/\*\*  
\* Notifies success/fail to user  
\* @param message a string representation of the message to be notified to the user.  
\*/  
public void notifyGuest(){

send this.notifyMessage to LoginView class;

}

**-----Class User-----**

/\*\*  
\* Validate user with password provided

\* @param password a string representation of the password that users enter  
\* @return return true if the password of the user matches the password provide, otherwise return false.  
\*/  
public Boolean validateUser(String password){

return this.password == password

}