Contents

Introduction

This report outlines the requirements and functionality of a fully completed log-in system. The system as requested, consists of a main login page, an account registration page, a reset password facility, a post-login welcome page and finally a logout facility. However, in order to provide a more successful and efficient system, a password change page has been added to the facilities available to the user when logged in. First of all, the login page offers the ability to the user to log in using their pre-existing account credentials or otherwise create a new account. Should the user wish to create a new account, a page is provided in which one is required to input necessary information for an account registration. A successful account registration requires the creation of a password by the user, which adheres to a specific format for security purposes. It must contain at least eight characters, of which at least one must be numerical and one alphabetical, in order to be accepted by the system. In addition to this, an email address will only be accepted if it complies to a valid format and has not been previously registered within the database. A pre-login reset password facility is also implemented to account for the possibility, that a user has forgotten their password. As a result, an email is sent to the user that provides them with access, via the form of a link, to retrieve a new randomly generated password. The welcome page and the option to change the current password are only accessible when the user is logged in.

Contained within the report, is an extensive step-by-step analysis of two major use cases in order to showcase some of the aforementioned features of the end-system. A detailed flow diagram has been included to depict the structural design and all possible states of the system a user may experience. Furthermore, a combination of different technologies was used in both client and server side which are analysed and can be seen in the technical architecture section provided. Lastly some of the design choices are further explained and analysed at end of the report.

Website URL: http://lamp0.cs.stir.ac.uk/~dfa/CSCU9W61

Use Cases

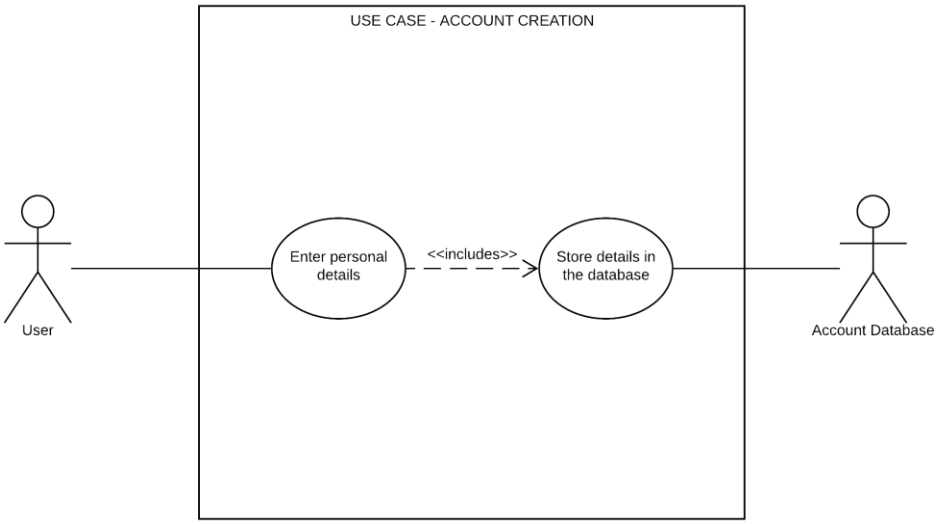
The following use cases were specifically chosen as they represent two major functions of the system. Both are accompanied by a use case diagram for visualization purposes.

Account Creation

The use case involves the procedure of creating a new user account. Every step included for successfully creating a new account is stated and any situational errors that may arise have been accounted for.

Primary actors: User **Receiving actors:** Account Database
Requirements: User must have an email address
Pre-conditions: User is currently on the login page
List of scenarios and events:
Normal scenario:
1. User clicks on the ‘Sign up’ link at the bottom of the log-in page 2. User enters the following details on the form: a. First name b. Last name c. Valid email address (‘Email available’ status is displayed to the user) d. Password e. Confirmation of password 3. User clicks the ‘Submit’ button 4. The Account Database is updated with the new user details 5. An account for the user is created 6. ‘Account created’ status is displayed to the user 7. User is redirected to the log in page after three seconds
Abnormal scenario:
In step 2 of the normal flow, the user enters their details. The system will not allow the user to create an account if: a. Any of the fields are left empty b. The first name and/or the last name provided, contain numerical or special characters. c. The email address format provided is invalid d. The email address provided clashes with an existing account e. The password format provided is invalid (Password must be at least 8 characters long and contain at least one number and one letter) f. The password and confirm password fields do not match An appropriate error message is displayed to the user if any of the above occurs when incorrect values are entered into the fields, as well as if the user attempts to submit the form
Assumptions:
User intends to create a new account
Post–conditions: Account for a new user is created

Account Creation Use Case Diagram



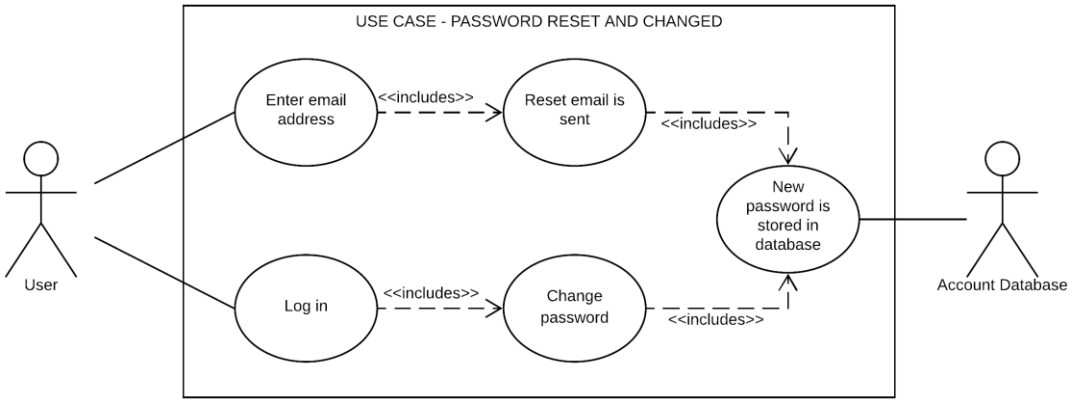
Password Reset and Changed

The following use case assumes that the user has forgotten their current password and involves the process of requesting a new password, as well as changing it to a new personal one. Every step included for successfully resetting and changing the password is stated and any situational errors that should arise have been accounted for.

Primary actors: User **Receiving actors:** Account Database
Requirements: User must already have a pre-existing account
Pre-conditions: User has forgotten their password and wishes to reset it
List of scenarios and events:
Normal scenario:
1. User clicks on the ‘Forgot your password?’ link of the log-in page 2. User enters their email address that corresponds to an existing account 3. User clicks the ‘Reset Password’ button 4. The Account Database is then updated with a new token for resetting the password 5. An email is sent to the user, containing a link to retrieve their new password 6. User is redirected to a confirmation page, which prompts the user to check their email 7. User follows the link emailed to them. 8. The Account Database updates the password of the specific user. 9. User accesses the page which contains their new password. 10. User clicks the ‘Home Page’ button to return to the log in page. 11. User logs in to their account using their new credentials. 12. User clicks the ‘Change password’ button. 13. User provides a valid new password. 14. User clicks the ‘Change Password’ button. 15. The Account Database sets the password of the user to their desired one. 16. ‘Password changed successfully’ status is displayed to the user 17. User is redirected to the welcome page after three seconds

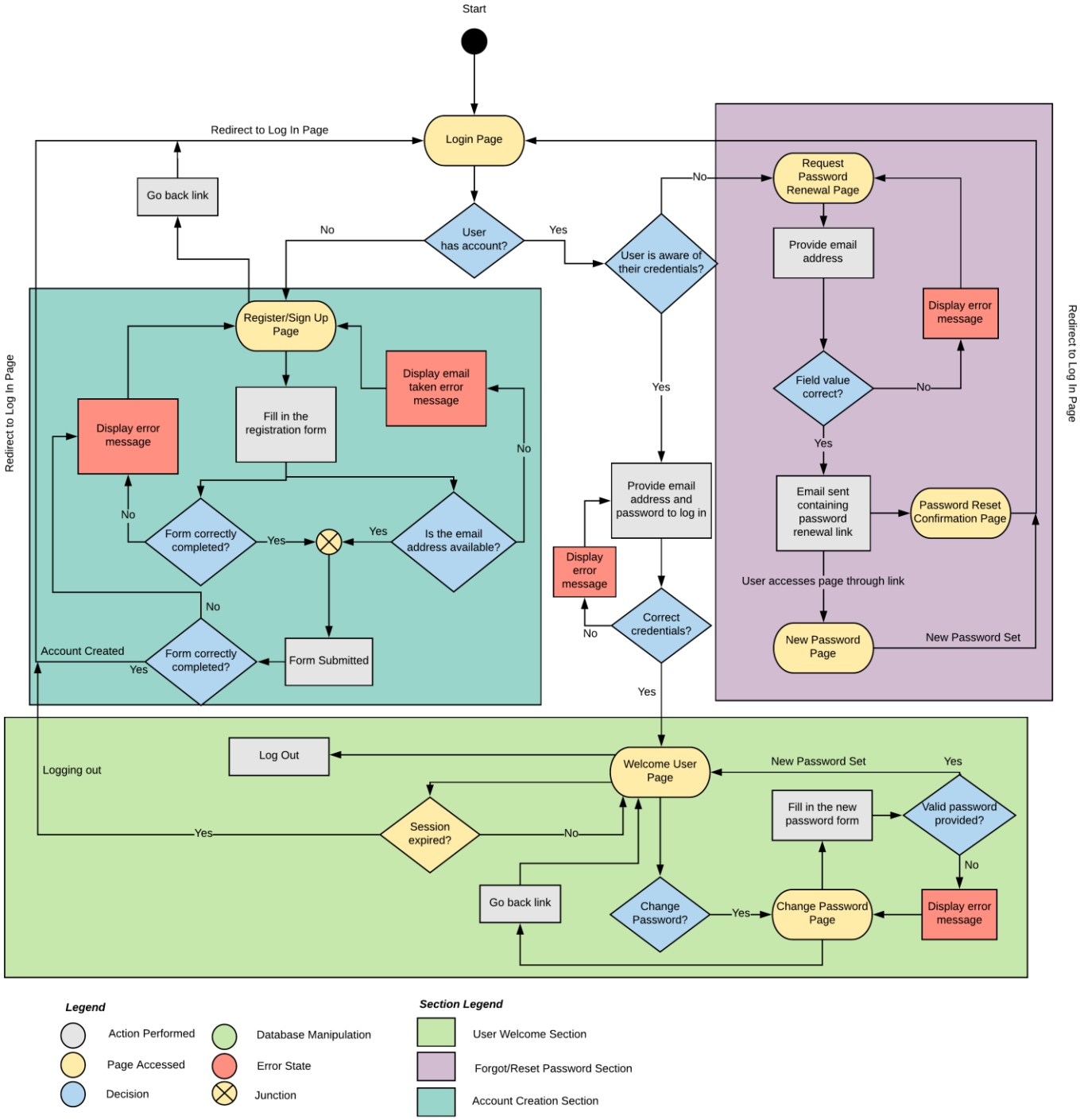
Abnormal scenario:
In step 3 of the normal flow, the user enters their email address. The system will not allow the user to reset their password if: a. The field is empty b. The email address format is invalid. c. The email address provided does not exist in the database. In step 11 of the normal flow the user attempts to log in. The system will not allow the user to log in if: a. The fields are empty b. The email/password combination is invalid c. The email or password formats are incorrect In step 13 of the normal flow the user enters their new desired password. The system will not allow the user to change their password if: a. The fields are empty. b. The password and confirm password values do not match c. The password entered is the same as the one already in use. d. The password and confirm password formats are invalid. e. The old password field value does not match the current password stored in the database. An appropriate error message is displayed to the user if any of the above occurs when incorrect values are entered into the fields, as well as if the user attempts to proceed with submitting the form
Assumptions:
User is currently not logged in
Post–conditions: Password is reset, and user successfully changes their password

Password Reset and Changed Use Case Diagram



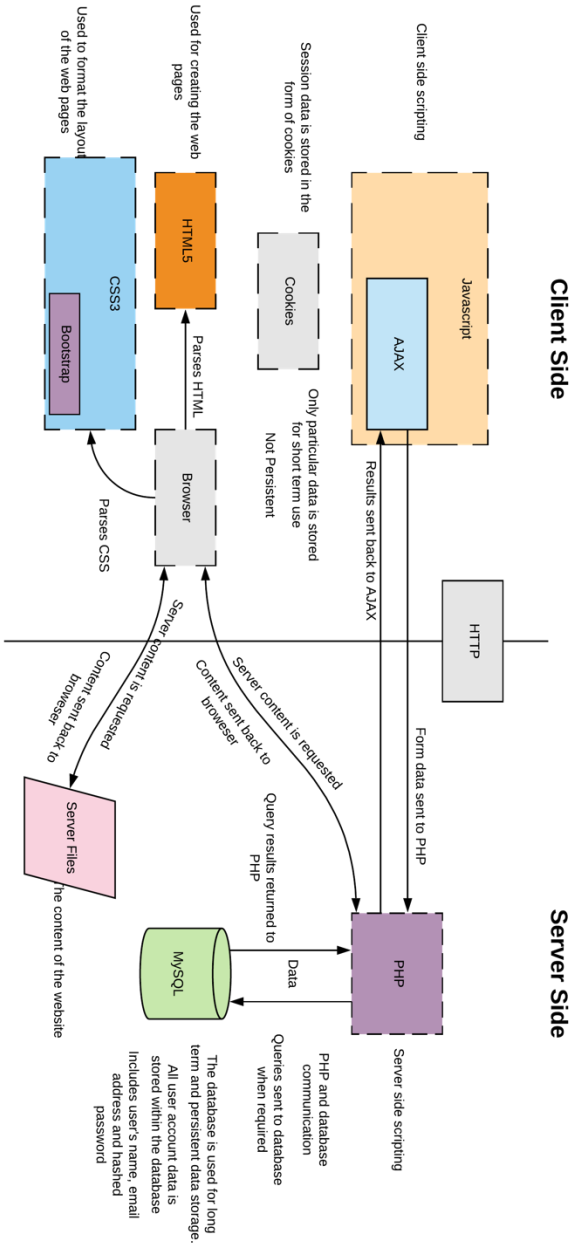
Design

The flow diagram below is a visual representation of the login system's internal and external design. The starting point of the diagram is the login page and it is assumed, the initial state of the user is not logged in. All possible states of the system are depicted; however, the majority of error messages have been abstracted in order to create a more concise and readable diagram.



Architecture

The diagram below depicts the technical architecture of the overall system. It displays the complete communication of data sent between the client side and the server side, and highlights the technologies run on either side. Moreover, the diagram visually demonstrates the purpose and role of each technology employed, emphasizing where data is stored for both long term persistence and short-term session control. In more detail, HTML5, CSS3 (including the Bootstrap Framework) and JavaScript are run on the client. On the other hand, the server side consists of PHP scripts as well as a MySQL database for long term and persistent storage of data. While the PHP scripts are run on the server, their output is sent to the client. In addition, AJAX provides the client-side access to the server scripts and therefore enables a more dynamic communication between client and server. Lastly, it is important to specify that session data is stored in the form of cookies, thus maintaining particular states of the system (e.g. keep a logged in state active). The default 24-minute session timeout is in operation.



Further Decision Choices

This section provides further analysis of the design choices made in order to ensure a complete end-system. First of all, all input forms are validated in both client and server side. Appropriate messages are displayed when an error occurs and equally when a field has been filled successfully. With regards to the registration page, when the user enters a value into one of the input fields, its outline becomes green or red, based on the validity of the value. Furthermore, a 'confirm password' field is appropriate to make sure the password is set properly.

As far as the log-in page is concerned, the user is prompted to enter their email address and password to access their account. To minimize input errors, a function to view the typed password is provided in the password field of the login page. Once the user is logged in, they are provided with a welcome page and a change password facility. In addition, while logged in, the user is unable to access the account creation and forgot password pages. Moreover, with regards to security, the PHP scripts have been modified to ensure protection against SQL injections, that may compromise the system's database.

Another major facility of the login system, is the forgot/reset password section. When the user enters their valid email, a link is emailed to the them which contains a token for their new password. Accessing this link provides the user access to their new password. Upon successful password reset, a confirmation message is displayed to the user.

Lastly, to provide a complete product experience, a change password section is available to the user when they are logged in to their account. The old/current password is required, as well as a new password that adheres to the format specifications. A further requirement to the password is also being added: The new password cannot be the same as the current password.