Tyler Elton

CSC483 - Winter 2018

Dr. Allison

Group members: Jason Moehlman, Zach Nelson, Gary Landrum, Brandon Krug

Group name: Team ; DROP DATABASE;--

Use case ID: wb-04.1 - Create User

Senario

Actor: Admin of the system

Pre-conditions:

1. Web page has been activated.
2. The instructor must be logged into the system.

Description:

1. *Use case begins* when the admin accesses the users portion of the admin dashboard.
2. The admin clicks on the "Create User" button.
3. The system shall display a form for the admin to enter the user's information (name, email, etc).
4. The admin shall choose what role to assign the new user.
5. The admin shall then click on the "Submit" button after entering the information. The system will process the request to create the user.
6. *Use case ends* when the system notifies the admin if the request was successful. It will redirect the admin to the individual user's display page if successful, or the listing of all users if unsuccessful.

Alternative Courses of Action:

1. In step D.3 (step 3 of the Description section), the admin has the option to click a "Cancel" button to cancel the creation of the user.

Exceptions:

1. The website cannot connect to the database to pull all of the available users. This will not allow the creation of another user.

Related Use Cases:

1. wb-04.02 - Edit User
2. wb-04.03 - Delete User

Decision Support:

Frequency: On average, at least 5-10 users will be created per week (most will be either imported via an import script that accesses the active directory).

Criticality: High. Allows the instructor to report any problems encountered with the system.

Risk: Medium. Implementing this use case employs standard web-based technology.

Tyler Elton

CSC483 - Winter 2018

Dr. Allison

Group members: Jason Moehlman, Zach Nelson, Gary Landrum, Brandon Krug

Group name: Team ; DROP DATABASE;--

Constraints:

1. *Performance* - must keep up with many requests to create multiple users simultaneously.
2. *Reliability* - must be readily available at all times to keep up with the multiple requests.
3. *Availability* - must be available at all times and be available anywhere (on/off campus).
4. *Maintainability* - must be maintained throughout the lifetime of the system.