**Software Design**

* **Sign up**

**User can create an account**

* User enters an email
* User enters a password
* User re-enters a password
* If email is a valid GMU address
  + Account is created in the Authentication Database
  + Account is created in the Realtime Database
* If email, password, or password re-entry are invalid
  + Displays the appropriate error message
* **Login**

**User can access their account**

* User enters an email address
* User enters a password
* Email and password are sent to the Authentication Database for login
* If login is successful
  + Redirects user to profile page
* If login is **not** successful
  + - Displays invalid credentials error message
* **Profile Page**

**User can edit classes they can provide help in**

* User enters a CS course number
* User presses button
* Hashmap is retrieved from database
* If the key (course #) exists
  + Get value (array of tutors)
  + Append user’s email to the array
* If the key (course #) does **not** exist
  + Create an array containing user’s email
  + Add a new key (course #) value (tutor array) pair to the hashmap
* Set the new value in firebase

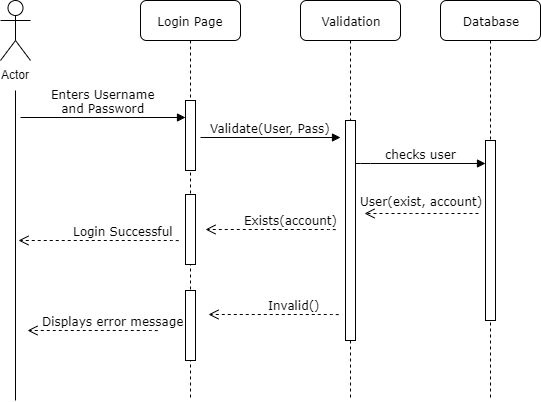
**Logout button**

Links back to sign in page

* **Tutor Page:**

**User can enter a CS course number that they want to find a tutor for**

* User enters a CS course number
* User presses button
* Hashmap is retrieved from database
* If the key (course #) exists
  + Get value (array of tutors)
  + Uses a random number generator to select a tutor
  + Retrieves the tutors account from database
  + Retrieves their tokens
  + Increments their tokens
  + Sets the value in database
  + Retrieves current user’s account from database
  + Retrieves user’s tokens
  + Decrements tokens
  + Sets the value in database
  + Outputs the tutors email to screen
  + If the key (course #) does not exist
* Output error message to screen (“No tutors currently available”)

****

