# Funktionslista för login-system

## Class.user.php

### checkUserRegisterInput($username, $email, $password, $passwordConfirm){

* Körs när användaren skickar in register-formuläret
* Kolla om användarens inmatade username eller email finns i databasen
* Kolla att lösenorden stämmer överens
* Kolla att lösenordet är minst 8 tecken
* Kolla att e-posten är giltig
* Returnera true för att köra registreringsfunktion
* Om inte lyckas: Felmeddelande på register.php

### }

### register($username, $email, $password){

* Kryptera det inmatade lösenordet password\_hash(lösenord, krypteringsmetod)
* Lägg in den inmatade infon i table-user
* Om det lyckats: Redirect till login.php
* Om inte: Felmeddelande på register.php

### }

### Login($username\_email, $password){

* Körs när användaren skickar in login-formuläret
* Hämta användarens info från databasen där användarnam eller email stämmer överens med inmatningen
* Om inget resultat: return false
* Jämför inmatat lösenord med sparat lösenord men password\_verify(inskrivet lösenord, hämtad hash ur databasen)
* Om lösenordet är korrekt ->
  + Skapa sessionsvariabler med användarinfo (user\_role, user\_id, user\_name)
  + return true + redirect till home.php
* Om inte -> return false + felmeddelande visas för användaren på login.php

### }

### checkLoginStatus(){

* Kollar om sessionsvariabeln ex. $\_SESSION[’user\_id’] har ett värde.
* Om finns -> return true
* Om inte -> redirect till login.php

}

### checkUserRole(användarens\_rollvärde, jämförelsetal){

* Kolla om användarens rollvärde är större än eller lika stort som jämförelsetalet
* Om lika stort eller större: return true
* Om inte: return false

}

### redirect($url){

* Flyttar användaren till en ny url.
* Header(location: $url);

### }

### logout(){

* Unset sessionens arrayvärden
* Förstör sessionen
* Flytta användaren till login-sidan

### }

### editUserInfo($userID, $username, $email, $password\_old, $password\_new, $password\_confirm, $firstname, $lastname(){

* Körs när användaren skickar in edituser-formuläret
* Kolla om lösenordet har uppdaterats
  + Kolla att lösenorden stämmer överens
  + Kolla att lösenordet är minst 8 tecken
  + Kryptera det inmatade lösenordet password\_hash(lösenord, krypteringsmetod)
* Kolla att e-posten är giltig
* Uppdatera den inmatade infon i table-user

}

### changeUserStatus($userid, $statusvalue){

* Uppdatera användarens status till $statusvalue

### }

### deleteUser($userid){

* Radera användaren från databasen, cascade länkade värden

### }

### searchUsers($input){

* Matcha $input med LIKE %$input% från user-tabellen
* Returnera resultaten
* Loopa lista på adminpage

### }

getUserInfo($userId) {

* SELECT användardata från databasen där ID överstämmer med $userId
* Fetch()
* Return datan till variabeln

}