

Workshop: Cubicle - Part 4

"Cubicle" is a place, where you can browse some of the most popular Rubik's cubes in the world and add some new cubes that you have discovered.

This application was created as a workshop for the [JS Back-End course](#) at the [Software University](#).

This is the final part of the workshop, so if you missed some part, make sure you complete it before you continue because all parts of this workshop are related to each other.

Main task

As a final step you should implement some validation, notification to improve the user experience and protect the data that is stored into the database.

Installing dependencies

You can use [express-validator](#) to do some **validation** and **sanitization**

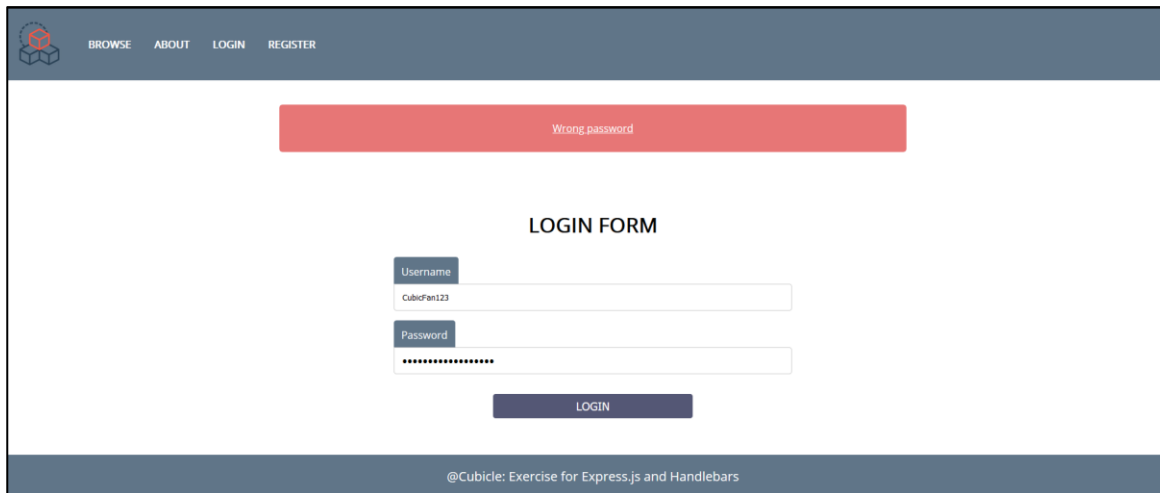
Validations

User

Before some user register or login, implement the following validations:

- **Username**
 - Should be **unique** (add "**unique: true**" property to each **User Model** username)
 - Should consist only with **English letters** and **digits**
 - Should be at least **5 characters** long
- **Password**
 - Should consist only with **English letters** and **digits**
 - Should be at least **8 characters** long
- **Re-Password** - (only in Register Page)
 - Should be the **same** as the given password

The screenshot shows a web application interface for a registration form. At the top, there is a navigation bar with links: BROWSE, ABOUT, LOGIN, and REGISTER. Below the navigation bar is a red notification bar with the text "({ Your notification message })". The main content area is titled "REGISTER FORM". It contains three input fields: "Username" with a "New" label, "Password" with a "Password" label, and "Re-Password" with a "Re-Password" label. Each field has a corresponding label above it. Below the input fields is a "REGISTER" button. The footer of the page says "@Cubicle: Exercise for Express.js and Handlebars".

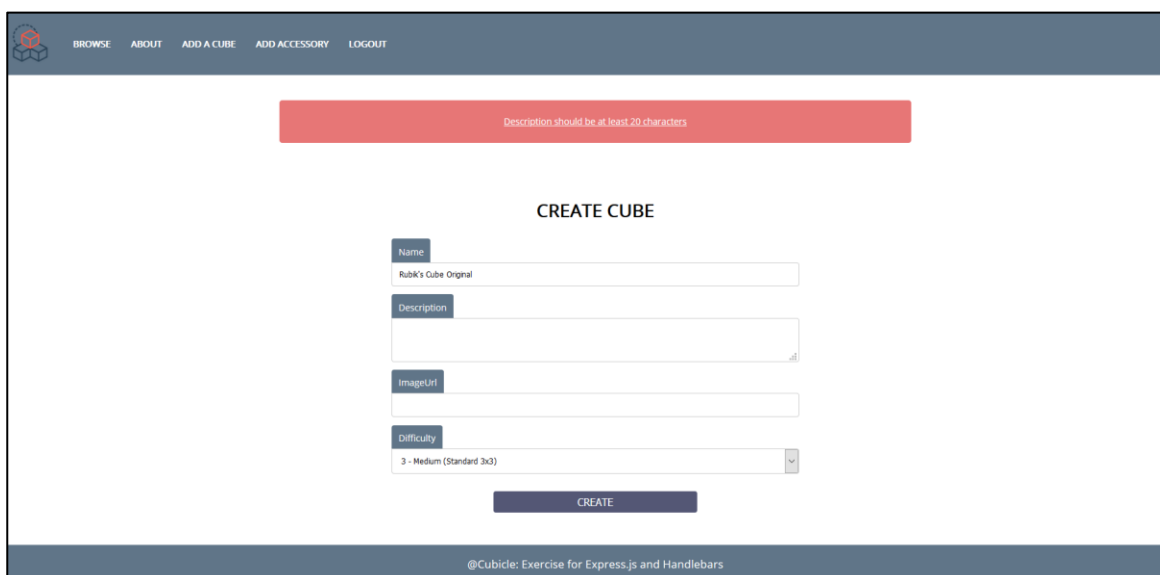


The screenshot shows a web application interface with a dark blue header containing a logo and navigation links: BROWSE, ABOUT, LOGIN, and REGISTER. Below the header, a red error message box displays "Wrong password". The main content area is titled "LOGIN FORM" and contains two input fields: "Username" with the value "CubicleFan123" and "Password" with masked characters. A "LOGIN" button is positioned below the password field. At the bottom of the page, a footer indicates "@Cubicle: Exercise for Express.js and Handlebars".

Cube and Accessory

The **cube** and **accessory** forms also should have some validation:

- **Name** - At least **5 characters long**, who could be **English letters, digits** and **whitespaces**
- **Description** - At least 20 characters, who could be **English letters, digits** and **whitespaces**
- **ImageUrl** - Referring to actual picture (starts with **http://...** or **https://...**)



The screenshot displays a web application interface with a dark blue header containing a logo and navigation links: BROWSE, ABOUT, ADD A CUBE, ADD ACCESSORY, and LOGOUT. A red error message box at the top states "Description should be at least 20 characters". The main content area is titled "CREATE CUBE" and features four input fields: "Name" (containing "Rubik's Cube Original"), "Description" (empty), "ImageUrl" (empty), and "Difficulty" (a dropdown menu set to "3 - Medium (Standard 3x3)"). A "CREATE" button is located at the bottom of the form. The footer at the bottom of the page reads "@Cubicle: Exercise for Express.js and Handlebars".

Error handling

In case of errors, for instance If the given username is **already taken, does not exists** or the given **password** (while login) is **incorrect**, handle the error properly and show an appropriate notification.

Wrong password

LOGIN FORM

Username
CubicFan123

Password
.....

LOGIN

@Cubicle: Exercise for Express.js and Handlebars

Template update

If you want to use the notifications which are shown in the pictures, you should modify a little your **main layout hbs** file. Create a **div** with **id="notifications"**, between the **header** and the **main** element, which holds a single **paragraph** with **id="notification-message"**. Check the picture below:

```
views > main-layout.hbs > html
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>{{ pageTitle }}</title>
7   <link href="/css/site.css" type="text/css" rel="stylesheet">
8   <link rel='shortcut icon' type='image/x-icon' href='/images/favicon.png' />
9 </head>
10
11 <body>
12   <div class="container">
13     <header>...
27   </header>
28   <div id="notifications">
29     <p id="notification-message"></p>
30   </div>
31   <main>
32     {{{ body }}}
33   </main>
34 </div>
35 <footer>@Cubicle: Exercise for Express.js and Handlebars</footer>
36
37 </body>
38
39 </html>
```

Also, you should append the following styles into the **site.css** file

```
#notifications {  
  margin: 0 auto;  
  margin-top: 0px;  
  margin-bottom: 0px;  
  text-align: center;  
  width: 50%;  
  padding: 0.5%;  
  margin-top: 2%;  
  margin-bottom: 2%;  
  background: rgb(231, 118, 118);  
  color: white;  
  border-radius: 5px;  
  text-decoration: underline;  
}
```

Good Luck! 😊