# Web Application Functionality

The main purpose of this project is to develop a simple web application with some secure techniques. The web application is developed with HTML/CSS Bootstrap, JavaScript/JQuery for the front side and PHP/MySQLI for the back office. Our web application is called “User-Blog”, which basically allows users to register, update personal profile and leave comments to each other.

## Use Cases

|  |  |
| --- | --- |
| **Use case ID** | **Use case name** |
| 1 | Register |
| 2 | Log in |
| 3 | Update avatar |
| 4 | Comment |
| 5 | Change password |
| 6 | Administrator bans user |
| 7 | Administrator deletes comment |

### Register

**Goal:** User register a new account.

**Actor:** User

**Precondition:** User is currently in the homepage.

**Main Success Scenario:**

1. User input username.
2. User input password.
3. User input password again.
4. User input email address.
5. User clicks the register button.
6. Application shows that the user has successfully registered.

**Extensions:**

2.1 The password does not satisfy the following criteria: “Password must be at least 8 characters long and must contain at least 1 upper case, 1 lower case and 1 digit”

3.1 The repeat password is not the same as the previous password inputted in the second step.

4.1 Email address is not with valid format.

5.1 Username already exist in the database.

**Post condition:**

The user successfully registered a new account.

### Log in

**Goal:** User logs in with his/her account.

**Actor:** User

**Precondition:** User is currently in the homepage and he/she has a valid account.

**Main Success Scenario:**

1. User input username.
2. User input password.
3. User clicks the login button.
4. Application shows that the user has successfully logged in.

**Extensions:**

3.1 Wrong username and password combination.

**Post condition:**

The user successfully logged in.

### Update avatar

**Goal:** User changes his/her avatar.

**Actor:** User

**Precondition:** User has already logged in.

**Main Success Scenario:**

1. User click “browse” button.
2. User choose a file.
3. User clicks the Submit button.
4. Application shows that the user has successfully updated the avatar.

**Extensions:**

3.1 User choose a file whose format is NOT “jpg”, “jpeg”, “gif” or “png”.

**Post condition:**

The user successfully updated the avatar.

### Comment

**Goal:** User sends some comments.

**Actor:** User

**Precondition:** User has already logged in.

**Main Success Scenario:**

1. User types some comments.
2. User clicks the “submit” button.
3. The comment is successfully sent.

**Extensions:**

None.

**Post condition:**

The comment is successfully sent.

### Change password

**Goal:** User changes the account password.

**Actor:** User

**Precondition:** User has already logged in.

**Main Success Scenario:**

1. User input old password.
2. User input the new password.
3. User input the new password again.
4. User clicks “change password” button.
5. Application shows that the user has successfully updated the password.

**Extensions:**

2.1 The password does not satisfy the following criteria: “Password must be at least 8 characters long and must contain at least 1 upper case, 1 lower case and 1 digit”

* 1. Repeat password does not match the new password inputted in the second step.

4.1 Old password is not correct.

**Post condition:**

The user successfully changed the account password.

### Administrator bans user

**Goal:** Administer deactivate a normal user’s account.

**Actor:** User

**Precondition:** User is logged in as administer.

**Main Success Scenario:**

1. User goes to the “admin” page.
2. User clicks “ban user” button.
3. Application shows that selected user’s account is deactivated.

**Extensions:**

None.

**Post condition:**

The selected account is successfully banned.

### Administrator deletes comment.

**Goal:** Administer deletes some comments.

**Actor:** User

**Precondition:** User is logged in as administer.

**Main Success Scenario:**

1. User goes to the “admin” page.
2. User clicks “delete comment” button.
3. Application shows that selected comment is deleted.

**Extensions:**

None.

**Post condition:**

The user successfully deleted that selected comment.

# Web application Misuse Cases

**Register/Log in/change password:**



For the register/log in/change password function, we mainly focusing on preventing “SQL injection”.

**Update avatar:**



For the update avatar function, we mainly focusing on preventing “path traversal”.

**Comment:**



For the comment function, we mainly focusing on preventing “cross site scripting”.

Additionally, indexing is prevented for all pages and uploaded files. In other words, users will get “403 access denied” error if he/she tries to access some page or uploaded file by URL. In this case, hackers will not be able to access any file by “path traversal”.

## Risk of attack

The frequency of an attack in a user blog is low since this web application does not involve any finical activity.

## Consequences of attack

In case of an attack, there could be several consequences:

* Users could lose their comments.
* Users could lose their account.
* Database chaos.

**Personal evaluation\_Jiaqi Ni:**

This project is the first time for me designing a secured web application. During the development of this project, it was actually quite easy to design and build the basic functionalities, although there were still some minor exception, which were all solved without taking too much time. However, securing the program is new for us and some unexpected exceptions did show up and take us quite a lot of time. Panties of research was conducted to find out the possible solutions, and fortunately, every insecure points revealed were successfully fixed. Meanwhile, we also learned a lot of new stuff during the process of solving these exceptions, which could definitely be highly helpful for further developer life.

In summary, this project is quite successful. Through this project, I have a better overview about how to design, build and secure a program, a better knowledge about how to work in as team, etc. Generally speaking, I gained a lot of valuable experience for secure programming, which could be pretty helpful for my further career.

**Peer accessment\_Jiaqi Ni:**

Name: Jiaqi Ni

Group: User blog

Fill in the form for every team member, including yourself

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Group** | **Products: total of column is 100%** | | | **Process: Mark (1- 10)** | | | | |
| **Student name** | **Research** | **Development of web application / hacking attacks** | **Report writing** | **Pro-Active  attitude** | **Amount of work done** | **Quality of work done** | **Communication within group** | **Total appreciation as team member** |
| Jiaqi Ni | 16% | 20% | 18% | 8 | 8 | 8 | 7.5 | 8 |
| Kalina Petrova | 20% | 25% | 24% | 8 | 9 | 9 | 8 | 9 |
| Mengchuan Liu | 16% | 20% | 10% | 8 | 8 | 8 | 8 | 8 |
| Jiefan Lin | 16% | 15% | 10% | 8 | 8 | 7 | 7 | 7.5 |
| Jianfei Feng | 16% | 10% | 16% | 7 | 7 | 7 | 7 | 7.5 |
| Ishant Upadhyay | 16% | 10% | 22% | 7 | 8 | 8 | 7.5 | 7.5 |