|  |  |
| --- | --- |
| Use case | Browsing the posts in the blog |
| Actor | Anyone |
| Trigger | Clicking on the Posts section |
| Precondition | There are posts in the database |
| Basic Flow | • Connect to the database.  • Retrieve posts from the database.  • Display titles of posts on the webpage.  • If the number of titles exceed threshold (like 20), display them on the next page (have a bar indicating number of pages at the bottom of webpage).  • Clicking on the title of a post will display the contents of the post. |

|  |  |
| --- | --- |
| Use case | Log in |
| Actor | Admin |
| Trigger | Entering the correct password |
| Precondition | Not logged in yet |
| Basic Flow | • Obtain password entered from user.  • Check if password contains special characters to prevent injection attack.  • Compare the password against the correct password. If password is correct, user is logged in. Create a cookie in the user’s computer that expires when browser is closed indicating the logged in status. |

|  |  |
| --- | --- |
| Use case | Post blog posts |
| Actor | Admin |
| Trigger | Clicking on Post when posts are written |
| Precondition | Logged in |
| Basic Flow | • Check the post for injection attack.  • Connect to the database.  • Write the post into the database |

|  |  |
| --- | --- |
| Use case | Delete blog posts |
| Actor | Admin |
| Trigger | Clicking on Delete when a post is open |
| Precondition | Logged in |
| Basic Flow | • Connect to the database  • Delete the post from the database |

Basic UI design is complete besides the posts section.

Still learning to implement the database feature.