

Basic Use Cases

Sign In

Use case no.	1
Applicable to	Writers, Admins and Commenters
Brief	The writer/admin/commenter signs in to the application
Pre-condition	The writer must have registered an account. Admin accounts are created when the database is created.
Trigger	The writer/admin requests to sign-in
Flow	<ol style="list-style-type: none">1. The writer/admin provides details to the app2. The webapp queries the information with the database server and reports errors if any (no user found, incorrect password etc.)3. If login details successful, the user is logged into the system.4. PHP session variables store the user's information such as real name and username5. The user is alerted that he has signed in.
Usage	<p>This facility is only accessible when no one has signed in to the webapp. When someone successfully logs in, no new users can log in.</p> <p>For new users to log in, current user has to log out.</p>

Sign Out

Use case no.	2
Applicable to	Writers, Admins and Commenters
Brief	The writer/admin/commenter signs out of the application
Pre-condition	The writer/admin must have been signed in into the application.
Trigger	The writer/admin requests to sign-out.
Flow	<ol style="list-style-type: none">1. The user sends a request to sign out.2. The entire PHP session is destroyed3. The user is alerted that he has signed out.
Usage	This facility is accessible only if someone has successfully signed in.

Create a new post

Use case no.	3
Applicable to	Writers, Admins
Brief	The writer makes a new blog post
Pre-condition	The writer/admin must have been signed in into the application.
Trigger	The writer/admin requests to write a new post
Flow	<ol style="list-style-type: none">1. The writer is shown a page to write his blog post2. The writer is required to fill in the title of the post as well as the article content (restricted to 8000 characters)3. The writer has the option to add tags to the article (separated by commas)

	<ol style="list-style-type: none"> The writer clicks 'post' to signal the completion of his post. The server checks the information and validates the request to post the article (returns error if the writer has not signed in) The server adds the data to the database The server alerts the writer that the article has been successfully posted
Usage	This facility is accessible only if the writer has successfully signed in

Alter Blog Posts

Use case no.	4
Applicable to	Writers, Admins
Brief	The writer/admin has the ability to edit or delete their previous blog posts
Pre-condition	The writer/admin must have been signed in into the application.
Trigger	The writer/admin requests to alter his blog post
Flow	<ol style="list-style-type: none"> The writer access the writer's portal The writer clicks on edit or delete next to the article he wished to edit (if he has no articles, he cannot edit/delete any) The server validates whether the writer has the privileges to change the article (server validates whether the article's writer is currently logged in or not) <ol style="list-style-type: none"> For deletion, the writer is prompted whether he is sure for editing, the writer is redirected to a page where he can view his previous blog post and make changes to it. The writer submits his changes which are updated in the database server.
Usage	This facility is accessible only if the writer has successfully signed in.

Add Comments

Use case no.	5
Applicable to	Writers, Admins
Brief	Only users with accounts (writers, admin) have the ability to comment on blog posts.
Pre-condition	The writer/admin must have been signed in into the application.
Trigger	The writer/admin writes a comment in the form below the article and clicks on 'comment'
Flow	<ol style="list-style-type: none"> The user clicks on the blog post he wants to comment on the user inputs the comment he wishes to add and clicks 'comment' The data along with the user's information is submitted to the database server. The comments are updated on the blog post.
Usage	This facility is accessible only if the writer has successfully signed in. Otherwise the writer is prompted to sign in.

Add tags

Use case no.	6
Applicable to	Writers, Admins

Brief	Only writers can add tags to their blog posts at the time of creation.
Pre-condition	The writer/admin must have been signed in into the application and is creating the blog post.
Trigger	the writer adds tags to his blog post and clicks on submit
Flow	<ol style="list-style-type: none"> 1. The writer accesses the portal to create a post. 2. The writer adds tags in the tag input box below the article body input box. (multiple tags are separated by commas) 3. The writer clicks on submit. 4. The tags are added to the database.
Usage	This facility is accessible only if the writer has successfully signed in and is creating a new article. Tags are optional.

Search articles with tags

Use case no.	7
Applicable to	Readers, Writers, Admins
Brief	All users can search blog posts with specific tags
Pre-condition	
Trigger	The user invokes a search request
Flow	<ol style="list-style-type: none"> 1. The user enters the tag keywords and initiates a search request 2. The server searches all blog posts with the given tag keyword 3. The server retrieves the titles of all the blogs 4. The server displays the relevant blog articles
Usage	This facility can be used anytime.

Read articles

Use case no.	8
Applicable to	Readers, Writers, Admins
Brief	All users can read blog posts
Pre-condition	
Trigger	The user requests a blog post
Flow	<ol style="list-style-type: none"> 1. The user either searches for a blog post (with tags) or clicks on a blog post from the home page 2. The server identifies the id of the blog article and retrieves the title, tags, body and comments of the article and displays it to the user.
Usage	This facility can be used anytime.