

COSC 360 Milestone Report — *BlogSphere*

Team Members: Siya Gupta, Cooper Ross

Project Summary:

BlogSphere is a dynamic and interactive blogging platform designed for user-generated content, social networking, and personal profile management. Users can register with their name, email, and profile image, then proceed to create blog posts with optional images, comment on others' posts, add friends, and customize their profile. The frontend is developed using HTML, CSS, and JavaScript, while the backend is powered by PHP and MySQL. BlogSphere offers a seamless experience with real-time updates, asynchronous interactions, and robust client- and server-side security.

Milestone Feature Overview:

To ensure client-side security, pages redirect unauthenticated users based on session checks using localStorage. Logging out clears session data, and input fields include basic validation. On the server side, all database interactions are protected with prepared SQL statements to prevent injection, and image uploads are renamed and stored securely to mitigate file-based attacks. The project is designed to be hosted on the cosc360.ok.ubc.ca server and supports GitHub integration, Apache compatibility, and proper file permission management using chmod.

Discussion threads (comments) are fully integrated with the database, with each comment stored asynchronously and displayed using live fetch() calls. The platform implements full functionality through PHP endpoints for users, posts, comments, and friendships. Posts can be searched using a keyword filter available directly on the home page, even without logging in.

Registration allows users to input their name, email, password, and profile image. Authenticated users can create new posts, comment on others' posts, and view/edit their own profiles. The profile page displays user data, recent posts, and friend count, and includes a built-in "Edit Profile" section.

Code Features & Enhancements:

To meet Milestone 2 requirements, several key features were added to profile.html and backend files. The profile page now includes a collapsible form for editing the username, bio, and profile image. This form is toggled with an "Edit Profile" button and submitted using JavaScript via an asynchronous fetch() call with a FormData payload.

The backend logic for updating user information is handled by update_profile.php, which accepts multipart form data, stores images securely in the /uploads directory, and updates the database using parameterized SQL queries. After a successful update, the profile reloads dynamically.

Form fields are auto-populated with current values when the profile loads, and the friend count is retrieved from `friend_count.php` in real-time. Access to the profile page is restricted to logged-in users only. Supporting backend files such as `get_user.php`, `update_profile.php`, `friend_count.php`, and `get_posts.php` handle data retrieval and updates securely. Additionally, a `theme.js` script maintains consistent UI theming across pages.