

Project 2 Data Validation - Written

Kevin Schaich (netid: krs252)

Design Rationale:

Written is a project I've been wanting to do for a while. Most of the design is modeled after Medium (medium.com) because I love their site and design, and it's something I've been wanting to experiment with for quite some time now. All the articles, content and images are property of their respective owners, and I have cited the source for each of the articles I used at the end of the article.

Data Validation:

- No duplicate article titles
 - See isDupe in main.php. Strip input title of any non-word characters and compare this pattern to the array of current articles.
 - Prevents errors in view.php. If duplicates were allowed multiple articles would be displayed here.
- Title, author, and category all must be less than 80 characters
 - See isValid in main.php. Check if the input string is less than 80 characters.
 - Keeps design of page clean and responsive
- Title, author, and category contain only alphanumeric characters, hyphens, spaces, and underscores
 - See isValid in main.php. Accept only "word" and "whitespace" characters as well as hyphens.
 - Keeps _GET searches tidy in URL bar and doesn't cause errors because of special characters
- No fields are empty in write.php. (except image, which is not required.)
 - See isValid in main.php and logic structure in write.php. Check if input is empty(), i.e. if it is set and is not empty.
- Check if image URL is valid upon article submission.
 - See isPic in main.php. Check if the URL is valid first, and also check if it ends in

one of (.jpg | .jpeg | .gif | .png)

- Don't allow malicious code/JavaScript injections
 - Use `htmlspecialchars()` anywhere `_POST` or `_GET` data is retrieved from the user or used/printed anywhere on the site

Example Queries (assuming data.txt has not been modified since submission):

Example 1

- General search: "two"
 - Returns all articles containing the written word "two" in any of the title, author, category, content, or date fields:
 - Greece - The Deal and the Risks
 - Writing Code for Humans
 - And Then There Were Two
 - Jony Ive Is a Redwood
 - Winter Storm Survival Tips
- Advanced search: input "two" into the title search field
 - Returns only articles containing "two" in their title field:
 - And Then There Were Two

Example 2

- Advanced search: input "design" into the content search field
 - Returns articles containing "design" anywhere in the content:
 - The Green Button Energy Data Movement Grows Up
 - The False Market Fit
 - Jony Ive Is a Redwood
 - Tell a four word story
- Advanced search: input "design" into the content search field and "business" into the category field
 - Returns articles containing "design" anywhere in the content and "business" as their category:
 - The False Market Fit
 - Tell a four word story
- Advanced search: input "design" into the content search field and "business" into the

category field and “trevor” into the author search field

- Returns articles containing “design” anywhere in the content and “business” as their category and “trevor” in their author field:
 - The False Market Fit

WOW factors:

- Clean, minimalist design with custom fonts
- Custom favicon
- Custom JavaScript/jQuery for hidden search/advanced search menu (see main.js)
 - When general/advanced search is revealed, first field gets focus for user input
- Responsive design with media queries for mobile devices
- Extensive use of PHP including dynamically loaded page content using require()
- Filter query links along top of page linking to advanced searches for respective categories
 - Don't display when the header width exceeds the page width
- Author and category for each article are links to an advanced search to that respective author or category
- Sticky forms in write.php so content is not lost if user needs to correct input
 - In-line validation, forms and help notices are all on the same page with no redirection
- Ability to actually “view” articles in the catalog rather than just “filter” through them
 - Accomplished using dynamic PHP content injection