Optimizing Data Analysis: Creating an Accessible Web Interface for SAMOS Data Blacklist Management

Andrea Miranda, Amanda Lovett, Shawn R. Smith

Center for Ocean-Atmospheric Prediction Studies, Marine Data Center, Florida State University, Tallahassee FL 32306

Abstract

In 2005, the Center for Ocean-Atmospheric Prediction Studies (COAPS) launched the Shipboard Automated Meteorological and Oceanographic System (SAMOS) initiative. The SAMOS project aims to provide high-quality marine meteorological and near-surface oceanographic observations through data provided from thousands of research, vessels and voluntary observation ships across the world.

However, when receiving such a large quantity of metadata, the SAMOS database is bound to intake instances of data deemed unwanted (for a variety of reasons). This type of "blacklisted" data would then have to be manually stored in the internal database through a MySQL interface, a task that could potentially fall into the hands of a user that is not familiar with this software; herein lies the challenge.

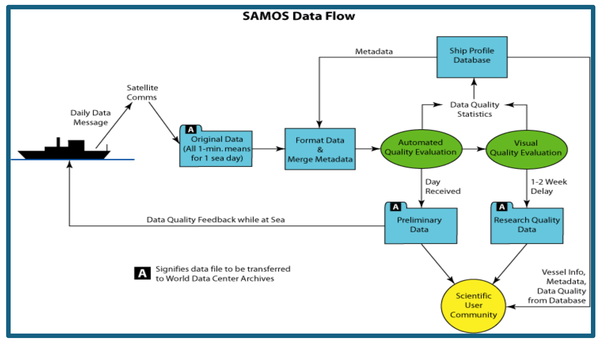
The goal of this project is to develop a web user interface in which internal non-technical users can add, remove, or modify unique variable identifiers stored within a blacklist via a more user-friendly web user interface. Through an understanding of HTML, PHP development, MySQL, and relational databases, we have created PHP functions to generate a blacklist data management interface that does not require proficiencies in other software.

By implementing this webpage, we aim to create a more accessible means of updating the unwanted data blacklist, optimizing the task of data analysis, and bridging the gap between programmers and data analysts.

Methodology

Preliminary Research

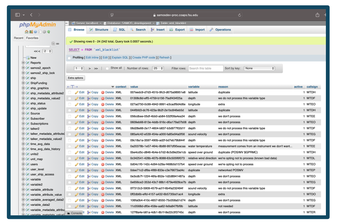
To effectively analyze and modify the existing codebase, I first built a strong foundation in HTML, PHP, MySQL, and relational databases. I reviewed scholarly articles on PHP's role in web applications managing large datasets, providing a theoretical framework for understanding the SAMOS developmental website's database connection. With this foundation, I collaborated with my research mentor, web analysts, and developers to analyze the code linking the database to the website. I also examined the pre-existing email whitelist code as a reference for implementing similar functions.



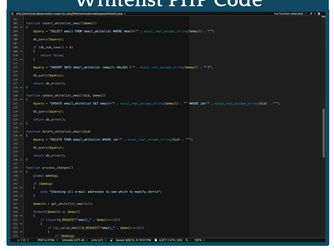
Implementation and Modifications

1. HTML Enhancements: Added columns to display relevant data on the webpage.
2. Print Form Function: Enabled display of current data values from the XML blacklist database.
3. Update Function: Allows users to add new data values, automatically updating the database via a submit box and button.
4. Delete Function: Implemented a checkbox feature for users to remove specific entries while maintaining database consistency.

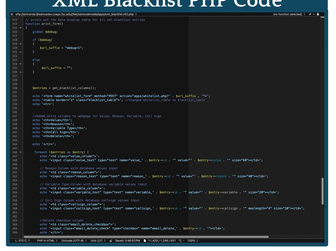
Code Progression and SAMOS blacklist DB



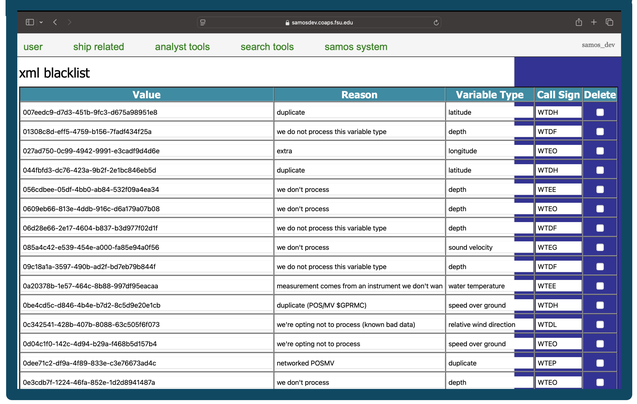
Whitelist PHP Code



XML Blacklist PHP Code



Results



Value	Reason	Variable Type	Call Sign	Delete
007eedc9-d7d3-451b-9fc3-d675a98951e8	duplicate	latitude	WTDH	
01308c8d-eff5-4759-b156-7fadf434f25a	we do not process this variable type	depth	WTDF	
027ad750-0c99-4942-9991-e3cadf9d4d6e	extra	longitude	WTEO	
044fbfd3-dc76-423a-9b2f-2e1bc846eb5d	duplicate	latitude	WTDH	
056cdbee-05df-4bb0-ab84-532f09a4ea34	we don't process	depth	WTEE	
0609eb66-813e-4ddb-916c-d6a179a07b08	we don't process	depth	WTEO	
06d28e66-2e17-4604-b837-b3d977f02d1f	we do not process this variable type	depth	WTDF	
085a4c42-e539-454e-a000-fa85e94a0f56	we don't process	sound velocity	WTEG	
09c18a1a-3597-490b-ad2f-bd7eb79b844f	we do not process this variable type	depth	WTDF	
0a20378b-1e57-464c-8b88-997df95eacaa	measurement comes from an instrument we don't wan	water temperature	WTEE	
0be4cd5c-d846-4b4e-b7d2-8c5d9e20e1cb	duplicate (POS/MV $GPRMC)	speed over ground	WTDH	
0c342541-428b-407b-8088-63c505f6f073	we're opting not to process (known bad data)	relative wind direction	WTDL	
0d04c1f0-142c-4d94-b29a-f4688b5d157b4	we're opting not to process	speed over ground	WTEO	
0dee71c2-df9a-4f89-833e-c3e76673ad4c	networked POSMV	duplicate	WTEP	
0e3cdb7f-1224-46fa-852e-1d2d8941487a	we don't process	depth	WTEO	

Picture of the finalized web page with data values that mirror the database and can be added, updated, and deleted externally

Acknowledgements

I would like to thank and acknowledge the entire team at the Marine Data Center at COAPS:

Conclusion

We aimed to develop a fully functional blacklist that mirrors the SAMOS SQL database on the website, integrated via backend PHP functions. This structured approach will enhance data accessibility, streamline database interactions, and support effective data management.

While no definitive findings are expected yet, this project lays the groundwork for future research and analysis within the SAMOS initiative. The improved dataset will benefit data analysts, environmental engineers, and other users while potentially inspiring further projects using the modified architecture.

References

Donahoo, M. J., & Speegle, G. D. (Gregory D.) (2005). SQL: Practical Guide For Developers (1st Ed.). Elsevier.

Göritz, A. S., & Birnbaum, M. H. (2005). Generic HTML Form Processor: A Versatile PHP Script To Save Web-Collected Data Into A MySQL Database. Behavior Research Methods, 37(4), 703–710. Https://Doi.org/10.3758/BF03192743

Kim, H., So, B. H., Han, W. S., & Lee, H. (2020). Natural Language To SQL: Where Are We Today? Proceedings Of The VLDB Endowment, 13(10), 1737–1750. Https://Doi.org/10.14778/3401960.340197

Röhm, U., Brent, L., Dawborn, T., & Jeffries, B. (2020). SQL For Data Scientists: Designing SQL Tutorials For Scalable Online Teaching. Proceedings Of The VLDB Endowment, 13(12), 2989–2992.