UNIVERSITY OF CALIFORNIA,
SANTA BARBARA

Improving Monitoring and Evaluation of Enforcement in Coastal Marine Protected Areas and Fisheries

A Capstone Project submitted in partial satisfaction of the requirements for the degree of Master of Environmental Data Science
for the
Bren School of Environmental Science & Management

by

KIRAN FAVRE
ELISE GONZALES
JARED PETRY
ADELAIDE ROBINSON

Committee in charge:
GAVIN McDONALD
CHRISTINA (NAOMI) TAGUE
RUTH OLIVER

JUNE 2023

Improving Monitoring and Evaluation of Enforcement in Coastal Marine Protected Areas and Fisheries

As developers of this Capstone Project documentation, we archive this documentation on the Bren School's website such that the results of our research are available for all to read. Our signatures on the document signify our joint responsibility to fulfill the archiving standards set by the Bren School of Environmental Science & Management.

_____
Kiran Favre

_____
Elise Gonzales

_____
Jared Petry

_____
Adelaide Robinson

The Bren School of Environmental Science & Management produces professionals with unrivaled training in environmental science and management who will devote their unique skills to the diagnosis, assessment, mitigation, prevention, and remedy of the environmental problems of today and the future. A guiding principle of the School is that the analysis of environmental problems requires quantitative training in more than one discipline and an awareness of the physical, biological, social, political, and economic consequences that arise from scientific or technological decisions.

The Capstone Project is required of all students in the Master of Environmental Data Science (MEDS) Program. The project is a six-month-long activity in which small groups of students contribute to data science practices, products, or analyses that address a challenge or need related to a specific environmental issue. This MEDS Capstone Project Technical Documentation is authored by MEDS students and has been reviewed and approved by:

_____
Gavin McDonald

_____
Ruth Oliver

_____
Date

Acknowledgments

We would like to express our gratitude to those who have supported and helped guide our team throughout this project.

Faculty Advisors:	Gavin McDonald, Environmental Markets Lab Christina (Naomi) Tague, Bren School Environmental Science & Management Ruth Oliver, Bren School of Environmental Science & Management
Clients:	Silvia Bor, WildAid Marine Meaghan Brosnan, WildAid Marine
Special Thanks:	Jamie Montgomery, Bren School of Environmental Science & Management Samantha Csik, National Center for Ecological Analysis & Synthesis Kat Le, National Center for Ecological Analysis & Synthesis

Table of Contents

Abstract

Marine protected areas (MPAs) are protected ocean areas that restrict certain human activities and typically include fishing limitations. MPAs can improve the health of marine species and habitats and are a critical tool in ocean conservation efforts. However, their effectiveness is often limited by a lack of enforcement, which allows for extensive illegal fishing despite their protected status. WildAid Marine partners with MPAs globally to improve law enforcement measures. They designed the Marine Protection System (MPS) Tracker to quantify and monitor progress in MPA enforcement. Through this project, the capstone team created a reproducible workflow for the data collected using the MPS Tracker. This included reformatting existing data into a consistent tabular data set and streamlining data entry by creating a data entry application for the MPS Tracker data. The team also created a shareable data visualization tool to summarize and highlight trends. These deliverables allow WildAid Marine to increase their efficiency in monitoring and improving MPA enforcement, and successfully share progress with donors and stakeholders.

1. Executive Summary

Marine protected areas (MPAs) are known to offer extensive benefits to both ecosystems and communities (FAO 2020). However, a significant issue is that over half of all MPAs lack adequate enforcement (Edgar et al. 2014). This enforcement gap results in communities suffering losses of billions of dollars each year due to illegal fishing (Agnew et al. 2009). To address this challenge, WildAid Marine is dedicated to equipping MPAs and fishing communities with the necessary tools to enforce their fishing regulations. This approach aims to not only safeguard the communities' livelihoods but also protect the health and sustainability of the ecosystems that support them.

With the goal of quantifying, monitoring, and ultimately improving the enforcement of MPAs and fisheries, the client, WildAid Marine, created the Marine Protection System (MPS) Tracker. This tracking system is a unique way of measuring various metrics of enforcement performance of their partner MPAs and fisheries. Site managers assess their local sites by scoring subcategories within five essential categories for an effective and well-enforced MPA. This data allows them to monitor progress and better understand where they may be lacking, ultimately assisting them in their work and improving the conservation benefits of the MPAs being monitored.

This capstone project created products that WildAid Marine can use to:

- input data from global site managers
- automatically format and securely store data
- visualize the data to create a story for partners and donors, demonstrating how WildAid Marine has influenced communities' ability to monitor and enforce MPAs

In order to address the needs of the client and produce all deliverables, the team started with reformatting the existing MPS Tracker data. This included four years of category scores and related comments from 24 sites stored in Excel files. This step was completed programmatically within R and resulted in all existing data being saved in a single Google Sheet. Once the data were reformatted, the team created a data entry application using R that interfaced with the final Google Sheet. This application allows site managers to enter data and have it append the newly entered data to the Google Sheet automatically. Finally, the team created a second application that pulls the MPS Tracker data from the Google Sheet to show interactive visualizations that the client will share with partners and donors.

These two applications are hosted online and integrated together into one pipeline. Site managers are able to enter their data into the data entry application. This data is automatically appended to the Google Sheet, which then updates the data visualization application. WildAid Marine will continue to use both applications as they expand their work.

2. Problem Statement

MPAs restrict human activity in sensitive regions. This ensures the protection of marine habitats and associated wildlife while contributing to the food and financial security of the surrounding communities (FAO 2020). Currently, over 60% of MPAs are reported to have ineffective enforcement (Edgar et al. 2014), limiting the potential of their conservation capabilities. WildAid Marine's BLUEPRINT for Marine Protection™ was developed with the goal of improving conservation through data-driven enforcement solutions for MPAs with global partnerships. Wildaid Marine's MPS Tracker is the first of its kind in standardized MPA enforcement tracking. This tool aids the organization in quantifying and monitoring progress in MPA enforcement. WildAid Marine is expanding the scope of their MPS Tracker from 24 to 250 sites and needs the infrastructure necessary to serve more MPAs and countries.

Prior to this project, the data collected by site managers from the MPS Tracker was generated and compiled manually by the Marine Program Manager. This led to errors and inconsistencies, lost files, and excess time spent on data processing. There was also no efficient method to enter and summarize data or share visualizations of the MPS Tracker data with donors. WildAid Marine is a nonprofit that needs funding from donors and partners in order to progress and make changes in MPA enforcement. With the previous data workflow, their work risked being overlooked or underestimated by essential partners. They also struggled to track the impact of their work without an organized data storage system.

3. Specific Objectives

The main objective of this project was to create an integrated workflow for the MPS tracker, allowing both streamlined data entry and data visualization.
This can be further broken down into three objectives achieved by the capstone team.

1. Reformat the existing MPS Tracker data from 24 sites, which were stored in Excel files, into a tabular format that is consistent across years.
2. Develop a data entry interface to enter future years of MPS Tracker data that is connected to the finalized data set.
3. Create a method for WildAid Marine to showcase their data to donors and partners through a data visualization tool pulling from the finalized data set.

4. Summary of Solution Design

4.1 Approaches & Methods

The first step in the project approach was to reformat existing MPS Tracker data provided by WildAid Marine into a machine-readable format. All coding was completed in RStudio, a free, open-source integrated development environment. The team used R to systematically restructure the provided Excel spreadsheets and combine them into one Google Sheet. The team utilized the *tidyxl* and *unpivotr* packages to restructure the non-tabular Excel data into a tabular data set. They then created a lookup table with all possible previous names for a given subcategory, which was used to convert each subcategory to the final, consistent name. Further details on the previous data and output format are described in detail in Section 7.3. Completing this step programmatically ensured that no data entry errors were incorporated into the existing data, allowed the process to be reproducible, and enabled the team to adjust the data structure as the final format evolved throughout the course of the project. From this point on, the Google Sheet the team created in this process will be referred to as the MPS Tracker data set. The team also created and stored metadata for the MPS Tracker data set in the shared Google Drive.

After restructuring the existing data, the team created one application for data visualization and another application for data entry using the *Shiny* package in R. They built two separate applications in order to specialize them for their intended use. The team designed the data entry interface to automatically update the MPS Tracker data set based on user inputs in the application. They designed the data visualization to display existing and future data. The data entry tool will be used by the site managers at WildAid Marine and managed by the Marine Program Manager. The data visualization tool will be used and managed by the Marine Program Manager to show potential donors or partners the impact of WildAid Marine's work in MPA monitoring and evaluation.

4.2 Data Entry Application

To align with the client's priorities, the data entry application is linked to the MPS Tracker data set stored on Google Drive. This data workflow ensures the longevity of WildAid Marine's MPS Tracker data, saves the Marine Program Manager time previously spent on manual data compilation, and offers ease of use for employees of the client organization as they grow. Using this application allows site managers to enter data remotely from project sites globally. The application appends new, inputted data to the MPS Tracker data set. Key components of the data entry interface include built-in data validation, the ability for site managers to revisit and edit previously entered data, and authentication to prevent unauthorized users from entering or viewing data. The application references a list of sites that are actively monitored by the MPS

Tracker. This site list is stored in another Google Sheet, which allows the Marine Program Manager to change or add sites to the list as the scope of the tracker increases without needing to change the code. *Googlesheets4* provides the primary functionality that allows the user to enter and append data to the MPS Tracker data set, as well as repopulate the data entry interface with draft versions of the data for a specific site and year. The application is secured with user authentication using the *shinyauthr* package in R. As the data entry interface was developed, the team facilitated testing and incorporated feedback from the capstone client, the site managers, and faculty advisors.

4.3 Data Visualization Application

A separate R *Shiny* application contains the data visualizations. The application leverages the *googlesheets4* package to read in data from the MPS Tracker data set. This component offers a range of interactive and informative visualizations that allow users to explore WildAid Marine's data. The main user of this application will be the Marine Program Manager, who will leverage this application to show the progress made using their MPS Tracker to donors. Key aspects of the data visualization include an interactive map that displays information and motivation for conservation across all of WildAid Marine's sites, a filterable and searchable chart of the entire data set for interactive data exploration, a summary table of the data and user-input defined plots that are used for comparing sites across years or category scores across sites. The integration of Google Sheets also enables the Marine Program Manager to remove sites from the site list, as well as rows from the data set from visualizations, as outlined in Sections 7.4.3 and 7.1.1. This feature allows the Marine Program Manager to hide previous sites that are no longer active or categories that are no longer monitored, ensuring the visualizations remain up-to-date and relevant. As the team developed designs, they facilitated product testing with both the client and peers. The feedback they received was incorporated into the final product.

5. Products and Deliverables

This project entailed developing products to meet the client's needs and completing academic deliverables to satisfy the Master of Environmental Data Science academic guidelines.

5.1 Client Products

The WildAid Marine team has sole access to the products created by this project. This includes the MPS tracker data set, the data entry and visualization applications, and all associated files. The reformatted MPS tracker data set and data entry application allows the client to efficiently input future years of data. The data visualization application allows for presenting this data to donors and partners. The private Google Drive contains the MPS Tracker data set, a text file with

metadata, a site list with coordinates for all active and previous sites, and a spreadsheet containing categories and subcategories that can be used to modify the subcategories being tracked. A user manual written by the capstone team provides detailed instructions on how to operate and update the Google Sheet and applications, access the data, and troubleshoot any potential issues. The detailed file structure and full list of files are outlined in Appendix A.

6. Summary of Testing

6.1 Reformatted Data Testing

The reformatted data underwent testing using the *testthat* package to ensure that the formatting was consistent and accurate. Tests included checking for duplicate subcategories for a single site and year, ensuring that subcategories, categories, and indicator types are only those listed, and confirming that the score was an integer value between 1 and 5. The testing also ensured that there were no NAs in the year or site columns, the correct number of reformatted rows were present for each year, and that all initial Excel files were included in the MPS Tracker Data Set.

6.2 Data Entry Application Testing

To ensure that the data entry system responded and functioned correctly, the *shinytest* package was used to record and test actions within the application. A prototype of the data entry application was also tested with both the client and site managers to confirm functionality and ensure it met the client's needs. The capstone group tested the application by entering test data using the application and verified that the data was automatically appended to the MPS Tracker Data Set. Data validation was built into both the data entry system to ensure that no incorrect data could be entered in the future. For example, before the user can input information about category scores, they must select a country, site, and evaluator, or the application will not allow data entry. The scores inputted are also selected from a drop-down menu, ensuring that only valid values are input.

6.3 Data Visualization Application Testing

The prototype of the data visualization application was tested with the client both for functionality and to ensure that it met the needs of the organization. Integration testing was performed by manually entering test data into the data entry application to ensure that data was correctly input into the MPS Tracker Data Set and then appeared in the data visualization.

7. User Documentation

This section provides instructions for updating sites, navigating the applications, troubleshooting common issues, updating the applications, and ensuring consistency in data quality for future data changes. It also covers topics such as backup and restoration procedures and archive access. A copy of this documentation is also contained in a User Manual stored on the client's Google Drive for ease of access. The intended audience for this manual is the client (the Marine Program Manager), site managers entering data, and those attempting to troubleshoot these applications. Throughout the remainder of this document, the primary application users for the data entry and data visualization applications are referred to as the user(s). A user conducting administrative tasks to update the applications is referred to as the application manager, these tasks will likely be completed by Marine Program Manager, or someone hired by WildAid Marine Program to work with the applications.

Refer to Appendix A for the directory structure and a complete list of files handed over to the client following the completion of the project.

7.1 General Application Instructions

7.1.1 Updating Sites

There is a Google Sheet in Google Drive titled "site_list". Updating the sites in this file will update both the map in the data visualization and the list of sites that can be selected in the data entry application. For each site that WildAid Marine partners with, there is a row in this table with relevant information. This application manager can update the site list by adding a new row with site information. The application manager should complete all fields when adding a new site and the site should be marked as "active" in the active_site column. If the application manager does not have coordinate or area information available when filling it out, they can be left blank if needed. However any sites without coordinates will not show up in the visualization application map. If a site is no longer partnered with WildAid Marine, the active_site column should be marked as "previous". Marking a site as "previous" will remove the site as an option for the data entry application, as well as remove it from the site map in the visualization application. Updating a site will not remove the site from the data or from the other visualizations within the application besides the map. To remove a site from all visualizations besides the map, see Section 7.4.3. If a site name on the site list is written incorrectly, it can be updated on the site list which will update the map and data entry system. However, it is *critical* that every row in the compiled data for that site also be updated with the same corrected site name. The user can update every instance of a site name using the Google Sheets find and replace functionality in the MPS Tracker Data Set. The application manager should ensure that "match case" and "match entire cell contents" are checked when replacing.

7.1.2 Language Control

If the application is opened with Google Chrome, the user will have the capability to translate to their preferred language. To do so, the user should set their preferences in Google Chrome to turn on translation for the preferred language. Begin by locating the 'Customize and Control Google Chrome' dropdown menu indicated by three dots on the right of the taskbar. Navigate to Settings > Languages > Google Translate and select 'Use Google Translate' and select the language preference. Then, when on the data entry application, the user should right-click and select 'Translate' from the drop-down menu. More information on translating using Google Chrome can be found on Google's help page.

7.1.3 Shiny Account Requirements

The cost of hosting an application on the Shinyapps.io server fluctuates based on the needs of the program and the number of hours the applications are used. The currently selected plan is the Starter Plan, which provides the client with 100 active hours per year for users to access the application and costs $145 annually. If the client organization identifies the need to modify the number of hours to host the applications, they can find more information on the Shiny website. The capstone group used their budget to pay for one full year of hosting the applications on the Shinyapps.io server. The client will be expected to cover this cost once the capstone funding is depleted if they decide to proceed with the paid plan.

7.1.4 Password Protection

Usernames and passwords to log in are owned and distributed by the application manager. The passwords are stored in a Google Sheet named 'app_passwords' that the application manager can update as needed. This file has two tabs and each one is for a different application. The applications are accessible to authorized users only. To update the password for an application the application manager should navigate to the app_passwords Google Sheet. There are two tabs on this sheet, one for the visualization application and one for the data entry application. To change the username or password for a specific user, the application manager should update the password or username column with the desired value.

7.1.5 Troubleshooting Guidelines

If the application does not load the login page when launched, the user should ensure that there is a stable internet connection. If the application is still not loading with a stable connection, the application manager should try the following recommendations.

In the event that either of the applications crash or the code breaks, WildAid Marine will need someone with knowledge of R to fix and redeploy the applications. To begin working on the application, the applications manager should fork the Github repositories within the Github

organization and clone them locally. The applications are set up using the *renv* package to keep track of the versions of packages used for the creation of the app. This should allow future users to utilize the same version of the packages used in the initial creation of the app, and prevent dependency issues. To work with the renv files see the package documentation. The contents of the user manual, as well as documentation within the code, provide information on how to navigate potential errors, however, unanticipated errors may occur. When applicable, the team has provided the file name where the application code is stored and the relevant code chunks to examine and update. The easiest method to locate a specific code chunk within a file is using the keyboard search function to search for the code or associated comments.

The application is too slow with multiple users
The application manager should refer to the section on scalability in this guide to Shiny, and consider following the guidelines provided if applicable. Updating the code to utilize asynchronous shiny is another option that could potentially improve the speed of the applications with multiple users.

Issues with MPS Tracker data set formatting
If the application loads but an error occurs when the user enters data, it is possible that the formatting of the MPS Tracker data set was updated and is creating errors. The application manager can restore the file from a backup to ensure the issue is not related to the formatting of the file. For restoring from backups, see Section 7.3.3.

Google Sheets 4 and *Google Drive Authentication*
If one of the applications fails to start, it is possible that the application is no longer authorized to access necessary files. Before modifying the code, the application manager should confirm that the files accessed by the tracker have not been moved, and file names have not been changed. Moving files could cause the application to lose access to the required data, and moving them back to the original directory could restore access. If this does not resolve the issue, the code for Google Authentication is stored at the top of the global.R file of both applications. At the time of project completion, the authentication was set up with the personal emails of capstone team members. If the application has become de-authenticated, the application manager should update the email in both of the authorization chunks of code for the application in question. The email used must be one with permission to access all of the necessary files in the shared Google Drive. The application should be run locally and authenticated with an updated email before redeploying.

```
# auto-authenticate google sheets ... this will have you interactively
authenticate using browser
 options(gargle_oauth_cache = ".secrets/")
## auto authenticate without browser
```

```
  gs4_auth(
  cache = ".secrets",
  email = "email@gmail.com" # email for authentication
  )

drive_auth(cache = ".secrets",
           email = "email@gmail.com") #email for authentication
```

Disconnection from the MPS Tracker Data Set

If the MPS Tracker data set becomes disconnected from either of the applications, the application manager should check that the folder URL in the code is the correct folder where the data is being stored. The user can find this URL in this code near the top of the global.R file within the applications' repositories. They should confirm the data set is contained within the folder linked by this URL and has the correct file name: mps_tracker_data. They should also confirm that there are no duplicates of this file in the folder. If the URL needs to be updated, it should be changed to the URL for the folder which contains the data set. The application manager should ensure that the new folder is accessible to the account authenticated within the application.

```
#read in all the data
#url to the folder that contains the data set Google sheet
folder_url <-
"https://drive.google.com/drive/u/1/folders/1AvavGBfoZx_ThcXVn5gL_buQkip76ZtQ
"

files <- drive_ls(folder_url) |>
  filter(name == "mps_tracker_data") #name of data set file
#save the file id
main_sheet_id <- as_id(files)
#read in the data set
main_sheet <- read_sheet(main_sheet_id) |>
  mutate(year = as.numeric(year))
```

If following the above steps does not resolve the issue, the application manager can run the application locally and troubleshoot based on the errors received.

7.2 Data Entry Application Instructions

7.2.1 Accessing the Application

The data entry application can be accessed through the link provided to the client. The user will be taken to a page to enter their username and password. Once the 'login' button is selected, the user is taken to a page to enter scores for the first category of the MPS Tracker.

7.2.2 Entering Data

After the user is logged in, they are taken to a page to input information for four fields:

- Evaluator Name: The site manager or managers who are scoring the subcategories.
- Year: This refers to the year the site is assessed for (not the year the data is entered). The default for this field is set to the previous year, as site evaluations are typically completed in January of the following year.
- Country: The country in which the site being evaluated is located, selected from a dropdown list.
- Site: The site being evaluated is selected from a dropdown list, options for the site will automatically update based on the country selected. If the site for which the user needs to enter data is not available, they can contact the application manager to update the site list, see Section 7.1.1.

After entering their name, site, year, and country, the user clicks the "Save and Continue" button to proceed to the next tab. All fields on this initial page are required and error messages will occur if they are not completed. This tab also has drop-down menus with detailed instructions for entering data and navigating the application.

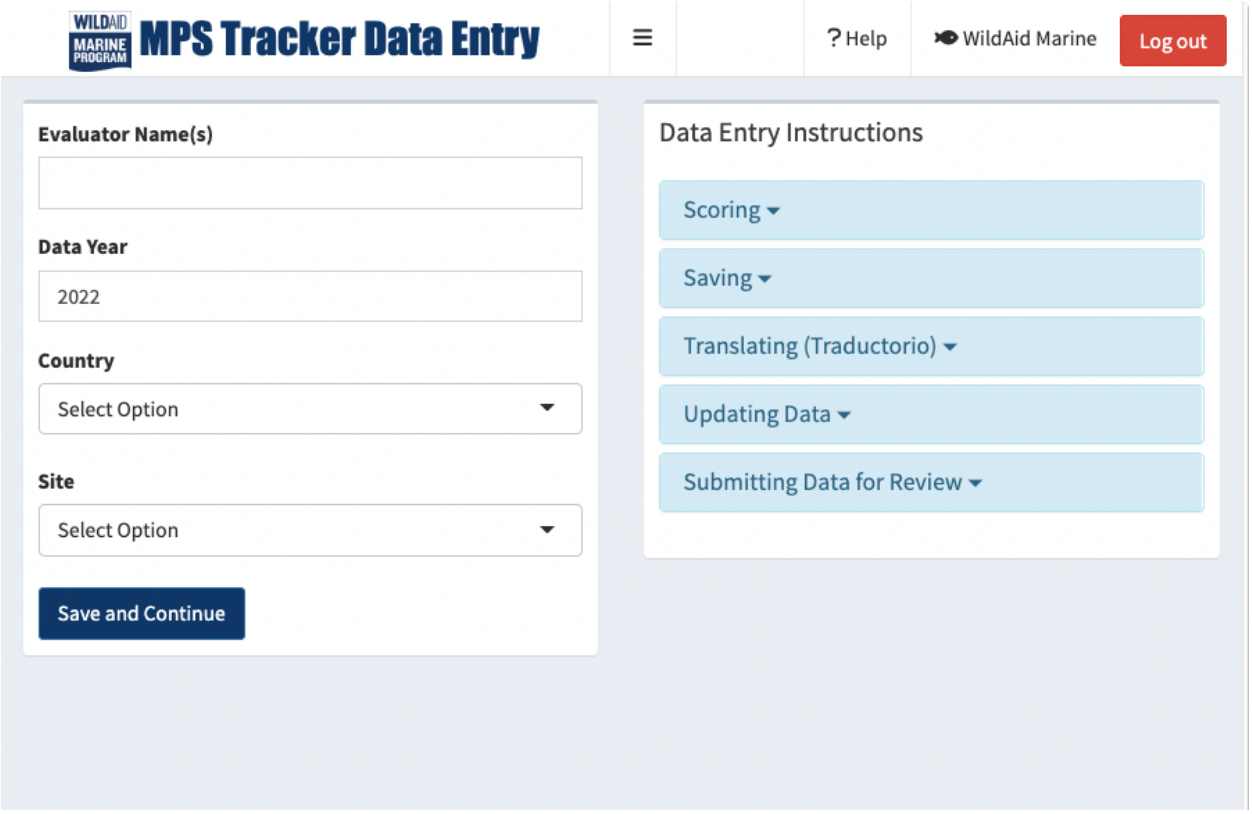

Figure 1. Home page for the Data Entry Application. The user will enter mandatory fields, and read instructions on this page.

Once the user has entered the required information on the home page, they are taken to the first category tab. Each tab represents a category which currently includes: “Surveillance and Enforcement”, “Policies and Consequences”, “Training and Mentorship”, “Community Engagement”, and “Consistent Funding”. Each box on the page consists of a single subcategory, containing a drop-down menu for scores and a text box to enter comments for that category. Within each subcategory box, there are guidelines for scoring each subcategory to provide context and ensure consistency. There is also a dropdown to see the scores recorded at that site for that subcategory in previous years. It is not a requirement to fill out all subcategories at once; this can be done over multiple sessions within the data entry application. No warning is given on each individual category page if all subcategories are not completed; however, the final summary tab will notify the user if they have not completed all of the subcategories. When the user is done entering data on a given category tab, they click the “Save and Continue” button to proceed to the next tab. At this point, each subcategory for which they have entered data will be appended to the MPS Tracker data set as a single row. If no comments are entered for a subcategory, an NA is entered for that column. If a user would like to enter or edit data for a specific category only, they can navigate to that tab using the sidebar menu. However, the “Save and Continue” button must be used to save data that has been entered in a given category tab, and the “Save and Continue” button on the home page must be used to begin data entry. If the user exits or logs out of the application without clicking “Save and Continue”, all data entered since the last save will be lost.

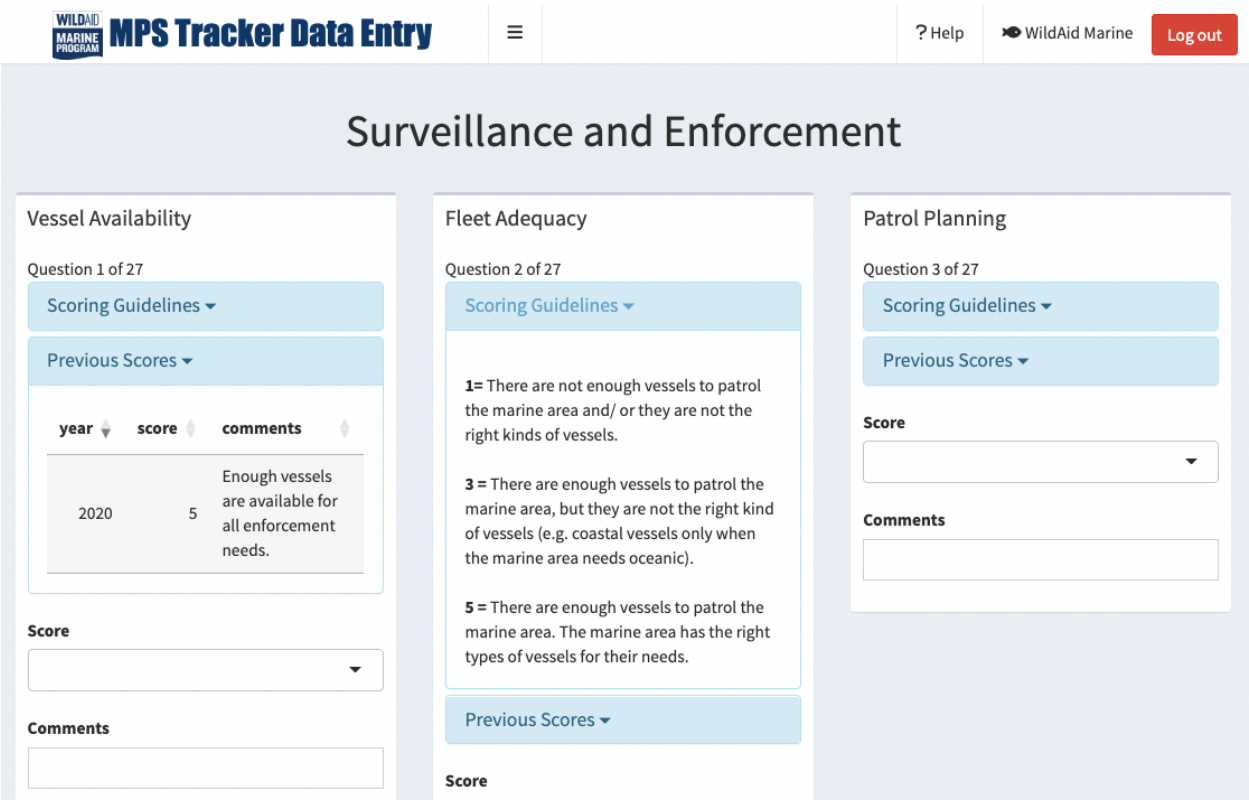

Figure 2. Surveillance and Enforcement category tab, showing results of clicking on the "Scoring Guidelines" and "Previous Scores" drop-downs on subcategory boxes. The user will enter scores on this page.

The final tab in the tracker is the summary tab. This tab allows the user to review all data that has been entered for the selected year and site. The icon at the top will notify the user of how many subcategories they have completed, and will show a red exclamation mark if subcategories are missing, or a green check box if all subcategories have been completed (Figure 3). When the user clicks save and submit, the system will confirm that all subcategories have been entered. If the data is complete, the application will display a message (Figure 4), append a row to the completed_data_log file, and the application will reload. Appending data to the completed_data_log file will allow the application manager to check to see what data has been completed. See Section 7.3.4 for more details on monitoring the completed_data_log file.

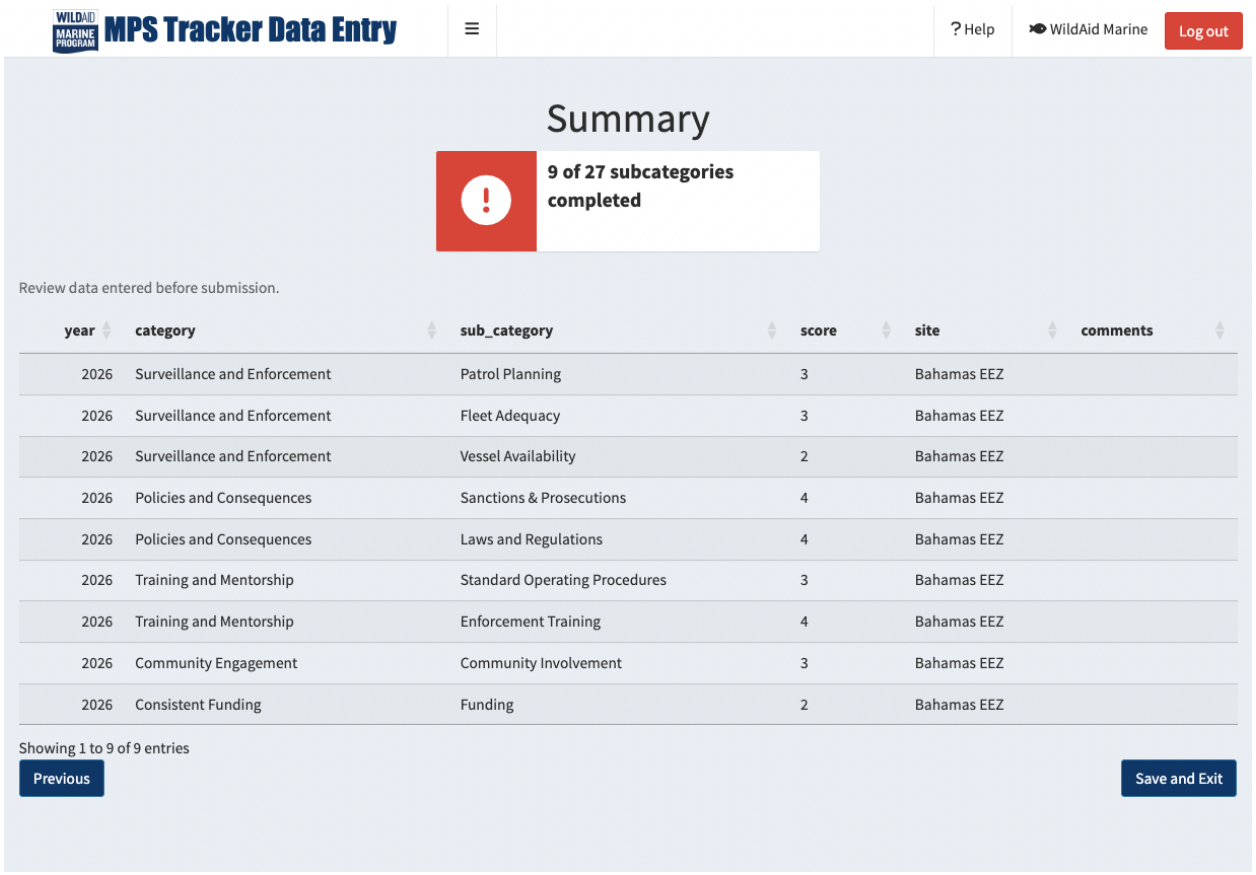

Figure 3. Summary tab, the table shows all subcategories that have been entered for the selected country and site. The popup at the top of the page shows how many subcategories the user has filled out and saved.

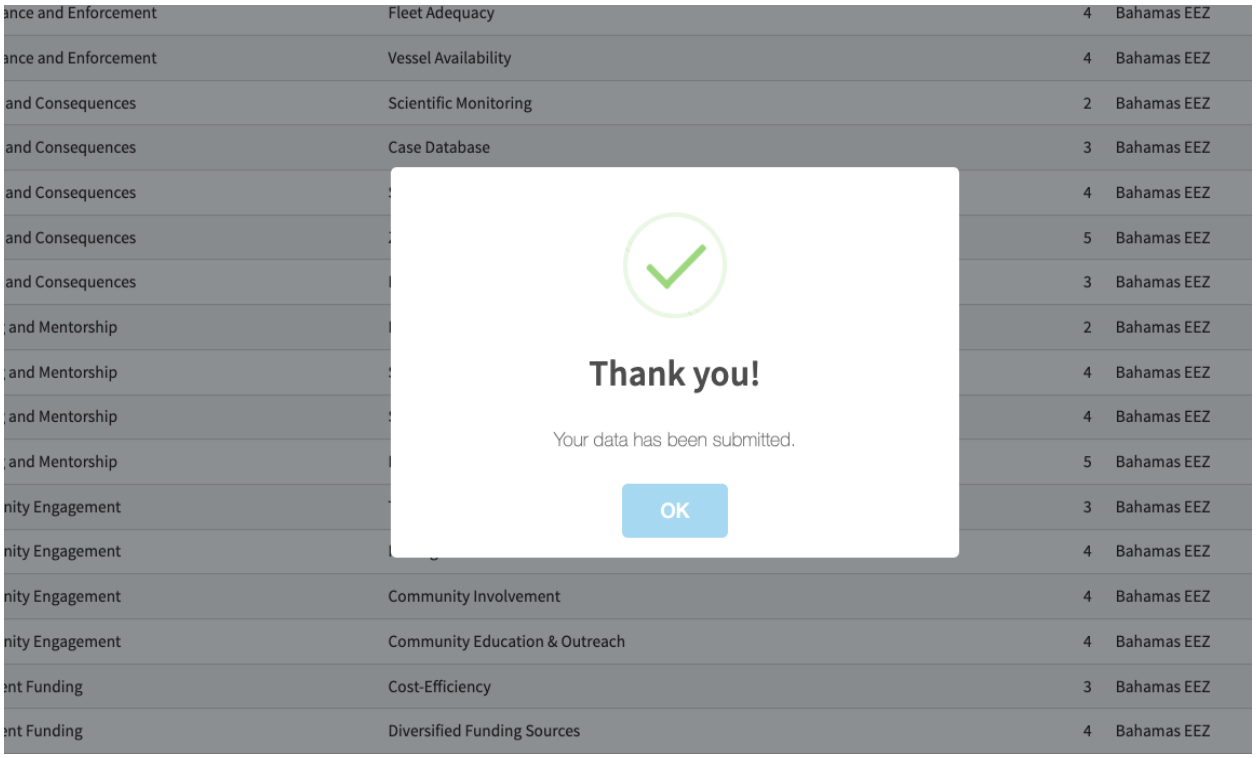

Figure 4. Data submission message. The message above is displayed when data has been completed and submitted on the final summary tab.

7.2.3 Updating and Viewing Existing Data

For site managers, if the user selects a year and site combination for which any data has already been entered, the application will automatically populate the scores and comments for each subcategory with existing data from the MPS Tracker data set. If these values are updated within the application, the application is designed to overwrite the previous values for that particular subcategory. Before overwriting occurs, a warning will be displayed on the next tab, asking the user to confirm the overwriting of existing data. If the user selects okay, the values will be updated, if they select cancel or exit out of the application, only new data will be added to the data set, and any updated values on the previous page will not be changed.

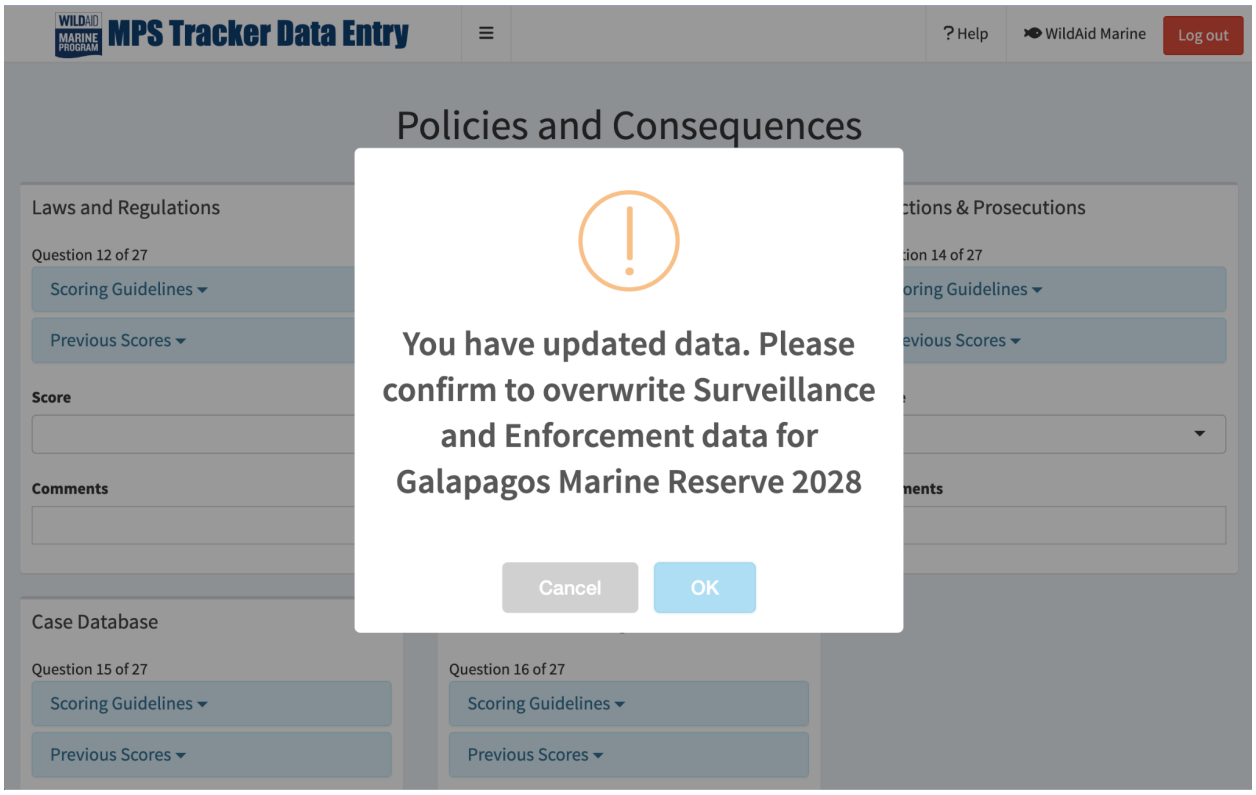

Figure 5. Warning message displayed when the user updates existing data values. The warning message will include the category, site, and year for which values are being updated.

The application manager can access data that is entered using the application in multiple ways. They can update the data through the application, or they can directly access the "mps_tracker_data" file containing the MPS Tracker Data Set within the shared Google Drive. If edits need to be made, the application manager can navigate to the cell of interest on this Google Sheet file and change the value. When updating data directly within the data set, the application manager should be careful to not overwrite any historical data unintentionally.

7.2.4 Updating Subcategories

The application was designed so that subcategories can be updated without any coding. Subcategory boxes are rendered based on the content of the app_subcategory table stored in the lookup_tables folder on the client's Google Drive. The application manager should make a backup copy of this file before making any edits, in case any unexpected errors occur from updating. To add a new subcategory, the application manager should add the name of the subcategory to the subcategory column. They should fill out the tab column with the tab name that corresponds to the category to which the subcategory belongs (see the sheet for reference). The application manager should add the updated scoring criteria for a score of 1, 3, and 5 to the score_1, score_3, and score_5 columns respectively. Since all other columns are calculated using

a formula, the application manager should pull down the cell above each of the other columns and ensure the values are calculated correctly. If the formula does not appear to be updating the cell correctly, they can either edit the formula or update the values manually. They should ensure that there are no blank values in the new row and that there are no spaces in the tab or any of the id columns.

subcategory	tab	score_1	score_3	score_5	score_id	id	comment_id	previous_table	collapse_id
Vessel Availabil	enforcement	Vessels are in p	Vessels are in g	Vessels are in good condition. Vessel availability	Ves_Ava_score	Ves_Ava	Ves_Ava_commer	table_Ves_Ava	Ves_Ava_collapse
Fleet Adequacy	enforcement	There are not en	There are enoug	There are enough vessels to patrol the marine a	Fle_Ade_score	Fle_Ade	Fle_Ade_commen	table_Fle_Ade	Fle_Ade_collapse
Patrol Planning	enforcement	The enforcemen	The enforcemen	The enforcement agency follows a strategic, dat	Pat_Pla_score	Pat_Pla	Pat_Pla_comment	table_Pat_Pla	Pat_Pla_collapse
Patrol Executio	enforcement	The enforcemen	The enforcemen	The enforcement agency conducts frequent, cor	Pat_Exe_score	Pat_Exe	Pat_Exe_commen	table_Pat_Exe	Pat_Exe_collapse
Patrol Equipme	enforcement	Patrol vessels la	Patrol vessels ar	Patrol vessels are equipped with both essential	Pat_Equ_score	Pat_Equ	Pat_Equ_commer	table_Pat_Equ	Pat_Equ_collapse
Intelligence So	enforcement	Intelligence infor	The enforcemen	The enforcement agency uses multiple channel	Int_Sou_score	Int_Sou	Int_Sou_comment	table_Int_Sou	Int_Sou_collapse
Investigation Pi	enforcement	There are no boi	There are some	There are boarding and chain of custody proced	Inv_Pro_score	Inv_Pro	Inv_Pro_comment	table_Inv_Pro	Inv_Pro_collapse
National Institu	enforcement	Overlapping juri	Overlapping juris	Institutions have clearly defined responsibilities	Nat_Ins_score	Nat_Ins	Nat_Ins_comment	table_Nat_Ins	Nat_Ins_collapse
International Ir	enforcement	Overlapping juris	Overlapping juris	Individual countries have clearly defined respon	Int_Ins_score	Int_Ins	Int_Ins_comments	table_Int_Ins	Int_Ins_collapse

Figure 6. Lookup Table for all subcategories, used by the data entry application for generating UI and entering data. Columns in which data should be updated are highlighted in yellow.

The application manager can rename a subcategory on the main_lookup table and it will change future data entry for that subcategory. However, it is important that they also change the name of the subcategory for every row in the existing data, see Section 7.2.6 or the data will not be connected across years. The team recommends avoiding renaming subcategories unless absolutely necessary.

If the application manager no longer wants future data to be collected for a given subcategory, they should delete the row for that subcategory from the main_lookup table. Removing a subcategory here will only remove it from being scored for future years, and will not remove it from the visualization. In order to remove the subcategory from the visualization all rows scoring that subcategory must be set to no for the visualization_include column see Section 7.4.3

7.2.5 Updating Categories

If the application manager needs to add or remove a category, they will need to use R and redeploy the application. This is a multi-step process that will require coding knowledge and ideally experience with R Shiny applications. The guidelines below are designed to walk the application manager through the process of updating a category, however, they may need to troubleshoot and should perform testing after updating to ensure the functionality of the data entry system remains intact.

The first step will be updating the lookup table, the application manager should add a row for each subcategory following the steps in Section 7.2.4. The application manager should choose a value for the tab name that is one word related to the name of the category and fill out all rows with that tab name.

Once the lookup table is updated, the application manager will need to update the code accordingly. They should first access the ui.R file of the data entry repository, where they will need to add the new category in two places.

Within the ui.R file the application manager should search for the following comment: “a new tab item can be added here". They can add code to create the new tab item below this comment. The following code can be used to create the necessary UI functionality, the application manager should replace all fields highlighted in yellow with the necessary fields based on the accompanying comments.

```
    # Consistent Funding tabItem----
    tabItem(
      #update to the name of the tab, must match tab in lookup table
      tabName = "funding",
      fluidRow(
        #update this with the actual name of the category
        h1("Consistent Funding", align = "center"),
        br()
      ),

      #update this to say ui_tabname
      uiOutput("ui_funding"),

      # start third fluid row

      #update the previous and next button number to one higher than the
following numbers if this is the first time a new category is being added.
Otherwise, the number must be one higher than the highest “previous” and
“next” buttons in the application.
      fluidRow(
        column(10, actionButton("prev_4", "Previous", class = "btn-primary",
style="color: #FFFFFF; background-color: #094074")),
        column(2, align = "right",

actionButton("next_6", "Save and Continue", class = "btn-primary",
style="color: #FFFFFF; background-color: #094074")),
      ) # end fourth fluid row
    ), # end consistent funding tab item
```

After updating the ui.R file, the application manager can update the server.R file.

First, they should search for the following comment in the server file “add a new category menu item here”. The application manager should add code below this comment to have the new

category appear in the sidebar menu. Add the following code to this section, replacing the values highlighted in yellow based on the provided comments.

```
#update tab name to the name of the tab, must match tab in lookup table

#update text to the name of the category

menuItem(
        text = "Consistent Funding",
        tabName = "funding",
        icon = icon("table")
      ),
```

They should search for the section titled “add buttons and data entry functionality for a new category here”. Below the section title is where the application manager can add the following code, updating all items highlighted in yellow based on the accompanying comments. You will also need to update the next button and data entry for the previous tab to ensure the buttons function correctly and enter data on the correct tab.

```
#start consistent funding tab actions ----
  #consistent funding tab data entry

  #start consistent funding data entry

  #replace with the name of the following tab, this will likely remain as
summary if you are placing the new tab last
  observe(
    if (entry_con() && input$tabs == "summary") {

      # read in the google sheet
      # need to do this each time we write in case multiple people are on the
app
      # identify the url
      # also read in for checking for existing data
      main_sheet <- read_sheet(main_sheet_id) |>
        mutate(year = as.numeric(year))

      #replace all instances of “con” with the three letter code for the
subcategory
      con_lookuptable <- main_lookuptable |>
        filter(tab == "funding") #change to the name of the tab, must match
ui and lookup table
```

```
      #initialize blank data frame
      append_data <- tibble()

      for (i in seq_along(con_lookuptable$subcategory)) {
        # name of the subcategory
        con_sub_category_name <- con_lookuptable$subcategory[i]

        # get the name of the score id
        con_score_input <- con_lookuptable$score_id[i]

        # get the value of the score
        con_score_value <- input[[con_score_input]]

        # name of comment id
        con_comment_input <- con_lookuptable$comment_id[i]

        # get the value of the comment
        con_comment_value <- input[[con_comment_input]]


        #replace the category value with the full name of the category (what
you want to appear in the data)
        con_row <- data_entry_function(google_instance = main_sheet_id,
google_data = main_sheet, year_entered = input$year_input, category =
"Consistent Funding", sub_category_entered = con_sub_category_name,
indicator_type = "Process Indicator", score = con_score_value, country =
input$country_input, site_entered = input$site_input, comments_entered =
con_comment_value, evaluator = input$name_input)
        if (!is.null(nrow(con_row))){
          append_data <- bind_rows(con_row, append_data)

        }

      }
      if (nrow(append_data) >0){
        sheet_append(main_sheet_id, data = append_data) }
      entry_con(FALSE)
    }
  )

  entry_con <- reactiveVal(FALSE)
  # consistent funding tab previous button
  #change to one higher than the highest number previous button and next
buttons, needs to match ui previous and next button
  observeEvent(input$prev_4, {
    newtab <- switch(input$tabs,
                     "community" = "funding",
                     "funding" = "community")
```

```
    updateTabItems(session, "tabs", newtab)
  }) # end consistent funding tab previous button

   # consistent funding next button
  observeEvent(input$next_6, {
    newtab <- switch(input$tabs,
                     "funding" = "summary",
                     "data" = "summary")
    # change funding to the name of the tab, only update summary if it is not
the next tab
    updateTabItems(session, "tabs", newtab)
    entry_con(TRUE)

      warning_ready(TRUE)


  })


# replace all instances of funding with the name of the tab for the new
category


   #generate funding ui
  funding_row <- main_lookuptable |> filter(tab == "funding")

  #replace with a unique variable name

   f <- list()
  current_row <- fluidRow()
  box_counter <- 0

  for (i in 1:nrow(funding_row)) {
    current_place <- funding_row[i,]
    box <- sub_category_box(current_place,sub_category_number)
    column <- column(width = 4, box)
    current_row <- tagAppendChild(current_row, column)
    box_counter <- box_counter + 1
    sub_category_number <- sub_category_number +1

    if (box_counter == 3 || i == nrow(funding_row)) {
      # Add the current row to the list and reset the counter and row
      f[[length(f) + 1]] <- current_row
      current_row <- fluidRow()
      box_counter <- 0
    }
  }

  output$ui_funding <- renderUI(f)
```

```
  outputOptions(output, "ui_funding", suspendWhenHidden = FALSE)

  #end consistent funding ui generation

# end consistent funding tab actions
```

7.2.6 Updating Subcategories and Categories in the Data Set

If a category or subcategory is renamed, the applications manager will need to update the name of the subcategory in the data for every row containing that subcategory. Because this has the potential to create inconsistencies in the data set, this should only be done if absolutely necessary. If needed, categories and subcategory names can carefully be updated using find and replace (under the edit tab in Google Sheets). This should be completed in addition to the renaming of the subcategory within the application described in Section 7.2.4. If categories are split or combined in the future, the team recommends updating previous data programmatically to ensure no errors are created during the process and data remains consistent.

7.3 Data Set and Data Management

7.3.1 Decisions Made in the Reformatting Process

Final names for subcategories are based on the names used in the 2022 version of the MPS Tracker. For example, the 2020 subcategory, "MPA has access to various sources of intelligence information for illegal activities," was renamed "Intelligence Sources" across all years.
When WildAid Marine had removed a subcategory entirely in the 2022 version of the tracker, the score was maintained, and the team renamed the subcategory to a shortened version. For example, the subcategory, "A decrease in the number of indirectly caught (e.g. by long lines, gill nets, etc.) protected species," was removed from the tracker after 2020. The score for this category was maintained for the years in which it was collected, and the subcategory was renamed "Decreased Bycatch."

When two subcategories were combined in the 2022 version of the tracker, the team averaged the score and combined it into the final category. For example, the 2021 version of the tracker has subcategories titled "Management Laws" and "Fishing Regulations". WildAid Marine combined both into one subcategory "Laws and Regulations" in 2022. In the reformatted data, the two scores for these subcategories in 2021 were averaged and combined into the "Laws and Regulations" subcategory. The comments from both subcategories were combined.

When one subcategory was split into two categories in 2022, the team duplicated the score from earlier years of data to provide a value for both final subcategories. For example, WildAid

Marine split the subcategory "Public institutions - Coast Guard/ Navy, Park Wardens, MPA Managers, Central Govt Reps (defense, environment, tourism, fishing/aquaculture), local government reps collaborate on enforcement and have the resources/will for enforcement." into "National Institutional Collaboration" and "International Institutional Collaboration" in the 2022 version of the tracker. The team also duplicated scores from earlier years so that both final categories have a score across all years where data was collected.

7.3.2 Metadata

A completed metadata file is stored in Google Drive, containing information on data sets created during this project as well as data collection methodology. UCSB Environmental Markets Lab's (emLab) metadata standards were used as a template when creating the metadata (Environmental Markets Lab 2022). Refer to Appendix B for complete metadata.

7.3.3 Restoring data from backup files

The team designed the data entry system to create and store a backup file roughly once every 14 days when there is activity on the application. Backup files are stored in the "backup" folder within the "data" folder on the drive in the format "mps_tracker_data_YYYYMMDD". To replace the current data set with a backup, the user should delete the "mps_tracker_data" file and replace it with the backup Google Sheet. The backup file must be renamed "mps_tracker_data" and be in the same folder as the original data set for the applications to pull from the correct Google Sheet. There also cannot be any files with the same name in this folder. Restoring from a backup should only be necessary in emergency situations, such as if the file containing the MPS Tracker data set has been deleted, or previous years of data have accidentally been overwritten. The application manager should also be aware that if any of the categories, or subcategory names have been updated or changed since the last backup, these will need to manually be updated for the entire backup data set as described in Section 7.2.6.

7.3.4 Reviewing and Finalizing Data

The team recommends that the application manager continue to review the data and check for any errors after it has been submitted by on-site managers. Reviewing can be done directly in the MPS tracker data set on Google Sheets, or through the explore data tab in the visualization application, which will allow for easy ordering of data, and querying for particular year and site. The completed_data_ log Google Sheet was designed to help the application manager keep track of which sites have data entered. This file is updated automatically by the data entry system when data entry is submitted for review, see Section 7.2.2. The finalized column is set to "no" by default and can be set to yes after data has been reviewed and finalized by the application manager. At the request of the client, data will immediately appear in the visualization

application once it is entered, even if the data has not been finalized.

year	country	site	date	evaluator	finalized
2021	Bahamas	Bahamas EEZ	2023-06-08	adelaide	no
2034	Bahamas	Bahamas EEZ	2023-06-08	Elise	no

Figure 5: completed_data_log. This Google Sheet is updated when a user completes all subcategories for a site and is designed to help keep track of which sites have completed data each year. The finalized column can be manually set to yes to indicate the data has been reviewed and is considered final.

7.3.5 Recommendations for ensuring quality of the data going forward

Although changing the tracker may be necessary in future years, it is best to minimize changes to the data collected when possible. Consistently monitoring subcategories will allow trends to be observed in those categories over time. When changes are made to any files or aspect of the data, ensure these changes are documented in the metadata and made consistently across the following files:

- MPS Tracker data set: mps_tracker_data
- Metadata: metadata.txt
- Subcategory lookup table: app_subcategory_lookup
- Site list: site_list

Additional recommendations include:

1. Maintain a machine-readable and tidy format for the data.
2. Follow consistent naming conventions across all files.
3. Maintain a consistent file structure.
4. Use machine-readable file and column names (use "_" instead of spaces, with consistent capitalization)

7.4 Data Visualization Application Instructions

7.4.1 Accessing the Application

The data visualization application can be accessed with the link provided to the client with an internet connection. The application is not intended to be publicly accessible. Once the username and password are entered and the "login" button is selected, the user will be taken to the homepage for the data visualization application.

7.4.2 Operating the Application

Details for how to operate the application are included within the application itself. The loading times should be relatively short, simply refresh the page if loading is taking too long. The visualizations are set up to default to certain user inputs that will show enough data for the plot to appear relevant. This is so that if the application is shared with partners or donors, the graphs will already be visually appealing. The application is intended to provide a platform for interpreting data in the future when the organization has acquired more data. At the time of project finalization, there were certain inputs that the user could select which would result in a single data point or a null value. In the future, when WildAid Marine has collected more data, this will be less frequent. If the client would like sites with minimal data, or previous sites, removed from the application, instructions for doing so are included in Section 7.4.3. The application also includes a link to the Technical Documentation so these instructions may be accessed from inside the application.

7.4.3 Removing Data From Visualizations

The application manager has the capability to remove raw data from the visualizations if need be. To do this, the program manager will change the Google Sheet titled “mps_tracker_data” which is stored in the Google Drive. There is a column within this Google Sheet titled “visualization_include.” There are two options for this column: “yes” or “no”. By default, when the data is input into this Google Sheet, the “visualization_include” variable will be “yes.” If there is data that should not be included, the application manager has the capability to manually change the “visualization_include” variable to “no”. When the data is read into the web application, the code is built to remove any observations within the data where this variable is marked “no.” At the time of project completion all rows of previous data within the “Community and Conservation benefits Category” were set to “no”. This was done at the request of the client, as the MPS Tracker no longer monitors this category.

Similarly, as the active site list evolves over time, the application manager will need to adjust the site list in the site_list that feeds into the map in the visualization application. The last column (farthest right) is labeled “active_site”. The entries that are set to “current” will be fed into the map within the visualization application. If the application manager wishes to exclude outdated sites from this map, all they need to do is change the entry from “current” to “previous”. Note that changing the status to “previous" will also remove the site as an option for data entry.

8. Archive Access

The raw data used in this project is the property of WildAid Marine and will not be archived publicly. The data, metadata, and accompanying documentation will be stored on the client's private Google Drive. Code for this project is stored publicly in repositories in the group's Github organization, and needs to be accessible to the client in the long term in case modifications may be needed.

9. References

1. Agnew, David J., John Pearce, Ganapathiraju Pramod, Tom Peatman, Reg Watson, John R. Beddington, and Tony J. Pitcher. 2009. “Estimating the Worldwide Extent of Illegal Fishing.” *PLOS ONE* 4 (2): e4570. https://doi.org/10.1371/journal.pone.0004570.
2. “Change Chrome Languages & Translate Webpages - Computer - Google Chrome Help.” n.d. Accessed June 9, 2023. https://support.google.com/chrome/answer/173424?hl=en&co=GENIE.Platform%3DDesktop#zippy=%2Cturn-translation-on-or-off.
3. Edgar, Graham J., Rick D. Stuart-Smith, Trevor J. Willis, Stuart Kininmonth, Susan C. Baker, Stuart Banks, Neville S. Barrett, et al. 2014. “Global Conservation Outcomes Depend on Marine Protected Areas with Five Key Features.” *Nature* 506 (7487): 216–20. https://doi.org/10.1038/nature13022.
4. Environmental Markets Lab. 2022. “EmLab Standard Operating Procedures.” 2022. https://emlab-ucsb.github.io/SOP/.
5. FAO. 2020. “The State of World Fisheries and Aquaculture 2020.” https://doi.org/10.4060/ca9229en.
6. “Introduction to Renv.” n.d. Accessed June 9, 2023. https://rstudio.github.io/renv/articles/renv.html.
7. “Shinyapps.Io.” n.d. Accessed June 9, 2023. https://www.shinyapps.io/.
8. “Shiny - Articles.” n.d. Accessed June 9, 2023. https://shiny.posit.co/r/articles/.
9. Wickham, Hadley. 2014. “Tidy Data.” *Journal of Statistical Software* 59 (September): 1–23. https://doi.org/10.18637/jss.v059.i10.

10. Appendices

Appendix A. File Structure

Files included in Google Drive, which were handed over to the client at the end of the Capstone Project:

```
UCSB_MPS_tracker
├── data
│   ├── backup
│   │   └── x backup_files (Google Sheets)
│   ├── metadata
│   │   ├── mps_tracker_metadata.txt
│   │   └── 4 template.xlsx files
│   ├── mps_tracker_data
│   │   └── mps_tracker_data (Google Sheet)
│   ├── raw
│   │   └── 34 .xlsx files
│   └── site_list
│       └── site_list (Google Sheet)
├── lookup_tables
│   ├── app_subcategory
│   ├── app_passwords
│   ├── reformat_subcategory
│   ├── reformat_site
│   └── reformat_subcategory_tidy
└── user_manual
    └── client_user_manual
```

Data:

- **mps_tracker_data:** complete MPS Tracker data set
- **site_list:** list of sites actively and previously monitored by WildAid Marine
- **backup:** folder containing backup files of MPS Tracker data set
- **raw:** original Excel files used to create the compiled MPS tracker data set
- **Metadata:** contains metadata for mps_tracker_data and site_list Google Sheets.
    - **mps_tracker_metadata.txt:** full metadata
    - **Template files:** contain previous versions of the tracker used to generate existing data prior to 2023 for reference.

lookup_tables:

- **app_subcategory:** used by the data entry application to determine which subcategories data entry should be available for
- **app_passwords:** used by the data entry and visualization applications to determine usernames and passwords

User manuals: A copy of the “User Documentation” section from the Technical Documentation.

Appendix B. MPS Tracker Metadata

This appendix provides more information about the MPS Tracker data set, site list, and associated data files provided to the client. It is intended to serve as a standalone document separate from the technical documentation and is stored as a .txt file on the client's Google Drive. The team created metadata to provide further details about the specific columns contained in the data files created in the capstone process, which WildAid Marine will continue to update. This mps_tracker_metadata file was generated on April 13, 2023, by the capstone team.

GENERAL INFORMATION

Title of Data set: Marine Protection System (MPS) Tracker Data Set

Organization: WildAid Marine

Primary Contact: Silvia Bor Bor@wildaid.org

Date of data collection: 2019-Ongoing

Geographic location of data collection: Global

PROJECT INFORMATION

This data set was created as part of a Master of Environmental Data Science (MEDS) Capstone project at UC Santa Barbara within the Bren School of Environmental Science and Management.

Github: https://github.com/iMPAct-capstone

SHARING/ACCESS INFORMATION

This data set is the property of WildAid Marine and is not intended for public reuse. Data sharing will occur only with the permission of this organization.

DATA DESCRIPTION & FILE OVERVIEW

Description: The data set contains ongoing WildAid Marine Protection System Tracker data. This tracking system was designed to measure various metrics of enforcement performance of MPAs or fisheries with whom WildAid Marine partners.

File Description: The MPS Tracker data and associated files are stored in the UCSB_MPS_Tracker/data directory. Files in the data directory include mps_tracker_data, site_list, tracker templates for previous years of data, original Excel files, and backup files. Backup files are saved periodically and stored in the backup folder. These are saved in the format mps_tracker_data_YYYYMMDD.

mps_tracker_data contains the latest version of the MPS Tracker and is updated regularly. The site_list file contains a list of both active and previous sites. Active sites are included in both the visualization application map and as options for data entry.

Supplementary template files store additional information on scoring metrics and indicators, see methodological Information.

METHODOLOGICAL INFORMATION

MPS Tracker data is collected on an annual basis at the site level. A site can include an entire EEZ, a single marine protected area or fishery, or a network of smaller protected marine areas. In 2023, the system through which MPS Tracker data was collected and synthesized was redone as part of the MEDS capstone project.

Data collected for 2022 and prior: Excel files containing the latest version of the MPS Tracker were sent to on-site program managers annually. Once they received the tracker, on-site program managers scored each subcategory based on provided metrics using their knowledge of the site and through discussions with on-site partner organization staff. Regional managers and additional staff worked cooperatively to fill out the tracker as needed. When complete, trackers were emailed back to the Marine Program Manager for review. Adjustments to scores were made through discussions with Marine Program Managers to ensure the scores were accurate and followed the provided metrics. The content monitored in the tracker was updated on an annual basis. Because of these updates, subcategories monitored, scoring criteria, the names of subcategories, and the category label of each subcategory varied from year to year. There were

also variations in the version of the tracker filled out across sites within the same year. Metrics (how to score, scoring criteria), categories, and subcategories used for each year of data collection are available in the template versions of the tracker in the metadata folder. An additional tab containing supplemental information for each subcategory is available for 2021 and 2022. This folder includes one template for the tracker for a year; however, there were minor differences between versions of the tracker used at sites for a given year.

Data collected for 2023 onward: On-site program managers are notified annually that they should begin filling out the MPS Tracker for their sites. Managers log on to the data-entry R shiny application using a username and password. On-site project managers score each of the categories based on the provided metrics using their knowledge of the site and through discussions with on-site partner organization staff. The metrics provided and the subcategories monitored are consistent with those in the 2022 version of the tracker. Multiple project managers can work in conjunction to fill out the tracker, and drafts can be saved over time. When the entire tracker is entered, the Marine Program Manager will review and work with the on-site managers to ensure data accuracy. The data does not need to be verified by the Marine Program Manager before it is used in the visualization application.

Scoring: All subcategories are scored with an integer between 1 and 5. Scores are based on provided metrics.

Methods for processing the data: MPS Tracker data collected for 2022 and prior was reformatted and incorporated into the current data set. Subcategories were named using the final naming convention in the 2022 version of the MPS Tracker. When a subcategory was removed entirely in the 2022 version of the tracker, the score was maintained and the sub_category was renamed to a shortened version. For example, the subcategory "A decrease in the number of indirectly caught (e.g. by long lines, gill nets, etc.) protected species" was removed from the tracker after 2020. The score for this category is maintained for years since it was collected and the subcategory is renamed "Decreased Bycatch"

When two subcategories were combined in the 2022 version of the tracker, the score was averaged and combined into the final category. For example, the 2021 version of the tracker has subcategories titled "Management Laws" and "Fishing Regulations". Both are in one subcategory, "Laws and Regulations", in the 2022 version of the tracker. In the reformatted data, the two scores for these subcategories in 2021 are averaged and combined into the "Laws and Regulations" subcategory. The comments from both subcategories are combined.

When one subcategory is split into two categories in the 2022 version of the tracker, the score from earlier years of data is duplicated to provide a value for both final subcategories. For example the subcategory "Public institutions - Coast Guard/ Navy, Park Wardens, MPA

Managers, Central Govt Reps (defense, environment, tourism, fishing/ aquaculture), local government reps collaborate on enforcement and have the resources/ will for enforcement." is split into "National Institutional Collaboration" and "International Institutional Collaboration" in the 2022 version of the tracker. The scores from the earlier version are duplicated so that both final categories have a score across all years where data was collected.

Quality-assurance procedures performed on the current data: Data validation is built into the data entry system. Country and site are selected from a pre-populated list, and sites are updated based on the country selected. Scores are selected from a drop-down list. Duplicate entries for a single site and year are not able to be entered into the system.

Quality Control performed on the previous data set: Quality checks were implemented using the *testthat* package in R. This included checking for duplicate subcategories for a single site and year; ensuring subcategories, categories, and indicator types are only those listed; ensuring score is an integer value between 1 and 5; and confirming there are no NAs in year or site.

People involved with sample collection, processing, analysis, and/or submission: Data is collected by on-site project managers, and processing is completed by Marine Program Manager. Data reformatting was completed by the MEDS capstone team: Kiran Favre, Elise Gonzales, Jared Petry, and Adelaide Robinson

---
DATA-SPECIFIC INFORMATION

<u>mps_tracker data:</u>

Format: Google Sheet
Number of variables: 10
Number of rows: 945 on 4/10/23

Variable list, defining any abbreviations, units of measure, codes or symbols used:
year: data year
category: overall category evaluated
sub_category: specific category evaluated
indicator_type: type of indicator monitored by subcategory
score
country
site
comments

entered_by: site evaluator(s)
visualization_include: yes or no variable to identify rows of data that should not be included in the visualization
Missing data codes: blank cells (standard NA storage for Google Sheet)
Specialized formats or other abbreviations used: None

site_list:
Format: Google Sheet
Number of variables: 10
Number of rows: 41 on 4/10/23

Variable list, defining any abbreviations, units of measure, codes or symbols used:
Country:
Site:
Latitude: WGS 84
Longitude: WGS 84
Partners:
Size:
PMS: on-site project managers
Status: Refers to WildAid Marine's blueprint for marine protection plan.
https://marine.wildaid.org/about-us/our-model/

1 - Discovery: Identify prospective project sites and in-country partner organizations, based on our criteria for success.
2 - Partnership: Establish a long-term partnership agreement and begin joint fundraising for the project.
3 - Enforcement Design: Conduct a full assessment of the current enforcement system and develop a complete, multi-year Marine Protection System (MPS) Plan.
4 - Implementation:
5 - Mentorship:
6 - Regional Leadership:

Active_site:

Active: currently a wildAid Marine site
Previous: former WildAid marine site

Year Started Working

Missing data codes: Blank
Specialized formats or other abbreviations used: None

Annual Templates: Internal WildAid Spreadsheets user in previous years for data collection. Included for historical context only.

2019_template

Category
Score 1-5
How to Score
Notes

2020_template

Category
Score 1-5
Scoring Criteria
Scoring Notes

2021_template

Tab 1: Data Collection

Category
Status 1-5
Status Scoring Criteria
Scoring Notes

Tab 2:

INDICATOR:
specific, observable & measurable characteristics, actions, or conditions that show whether or not the desired change has happened"

ACTIVITIES: actions (to be conducted by WA Marine or partners) associated with delivering project goals"

OUTPUTS:
direct, immediate & measurable short-term results associated with the project" "OUTCOMES
medium-term consequences that relate directly to project goals
(i.e. anticipated changes in knowledge, skills, attitudes, behavior, condition, or status)"

IMPACT: long-term & indirect consequences influenced by the project"

Definitions

Tab 3: Goals and Indicators

ESSENTIAL?
CATEGORY
YEAR INDICATOR
EXAMPLE GOALS
Definitions

2022_template

Tab 1: Data Collection

Category

Status 1-5 (or n/a)
Please Score each Site or Agency Using the Following Criteria
Scoring Notes

Tab 2: Additional Information

CATEGORY
STATUS
INDICATOR: specific, observable & measurable characteristics, actions, or conditions that show whether or not the desired change has happened" "ACTIVITIES
actions (to be conducted by WA Marine or partners) associated with delivering project goals"
OUTPUTS: direct, immediate & measurable short-term results associated with the project"
OUTCOMES: medium-term consequences that relate directly to project goals
(i.e. anticipated changes in knowledge, skills, attitudes, behavior, condition, or status)"
IMPACT: long-term & indirect consequences influenced by the project"
Definitions

Tab 3: Goals and Indicators

ESSENTIAL?
CATEGORY
YEAR
INDICATOR
EXAMPLE GOALS
Definitions

main_lookup:

Lookup table for all subcategories and scoring criteria currently used in the MPS Tracker Data Entry process.