Virginia Tech

Multimedia, Hypertext, and Information Access Capstone
CS4624

Blacksburg, VA 24061

Instructor: Dr. Edward Fox

Client: Dr. Francesco Ferretti

May 11, 2023

SharkPulse

Authors
Khanh Pham
Catalina Lemus
Mohammed Al Ansari
# Table of Contents

List of Figures .................................................. 3

Executive Summary ............................................. 4

1 Introduction .................................................. 5
   Team Member Roles ........................................... 6
   1.1 Requirements and Objectives .............................. 7
   1.2 Client ...................................................... 7
   1.3 Challenges ................................................ 7

2 User Manual ................................................ 6
   2.1 User Login .................................................. 6
   2.2 Validation Monitor ........................................ 7
      2.2.1 Instagram Validation Monitor ......................... 8
      2.2.1 Flickr Validation Monitor ............................ 10

3 Developer’s Manual ..................................... 14
   3.1 Navigate Monitor for Instagram or Flickr ............. 14
   3.2 Interact with R-shiny Google Maps. ..................... 15
   3.3 Instagram and Flickr ..................................... 16

4 Future Work and Projections .......................... 23
   4.1 Plan for this semester ................................. 23

Acknowledgments ........................................... 23

References .................................................. 24
# List of Figures

- Figure 1: User Login Interface ........................................... 7
- Figure 2: SharkPulse Tab Bar, Monitor Hovering. .................. 9
- Figure 2.1: Media Source Toggle and Map. .......................... 9
- Figure 3.1: Instagram Validation Monitor Map. .................... 10
- Figure 3.2: Instagram Image Validation Form. ....................... 11
- Figure 4.1: Flickr Validation Monitor Map. .......................... 12
- Figure 4.2: Flickr Validation Form ..................................... 13
- Figure 4.3: Flickr Validation Thank You. ............................. 13
- Figure 5.0: The validation monitor Nav Bar .......................... 15
- Figure 5.1: get_name() function ....................................... 17
- Figure 5.2: taxonomy relational scheme .............................. 17
- Figure 5.3: get_insta_data() function ............................... 18
- Figure 5.4: Instagram relation scheme ................................ 19
- Figure 5.5: get_flicker_data() function .............................. 20
- Figure 5.6: data mining relational scheme ........................... 21
Executive Summary

The team was required to redesign, update, and implement changes to an existing website called SharkPulse for a project focused on monitoring global wild shark populations. Currently, the project is led by Dr. Francesco Feretti, an Assistant Professor for the Department of Fish and Wildlife Conservation. The website was built using WordPress with frontend CSS and HTML, and the backend support was provided by PHP, Javascript, and R Shiny to implement the Validation Monitor Page. Our team’s primary objectives were to convert the static framework of the website to a dynamic, responsive one. Additionally, we aimed to convert the Instagram monitor from a PHP script to a R Shiny app and merge the Instagram and Flickr monitors into a single page, which would allow toggling between the two pipelines.

The Validation Monitor is a user interface that validates records collected from Flickr and Instagram (from the data_mining and Instagram tables). Since shark photos are primarily collected from social media platforms like Instagram and Flickr, the accuracy of the shark information may not be completely reliable. To address this issue, a form is provided with each picture of the shark for users to fill in, in order to validate the originality, type, pieces, and location of the shark. At the start of the semester, the Monitor's user interface was not responsive, dynamic, or easily navigable by users. Now at the end of the semester, we have successfully combined two versatile pipelines on one page and redesigned both the Validation Forms for the Instagram and Flickr Pipeline. Users can now toggle between social media sources for images to validate and fill out that specific source’s form. Additionally, we added a feature for common name and scientific species autocompletion, supporting users in filling out the Validation form more easily. Furthermore, the map in the Instagram validation form now functions properly, showing the current location of the shark marker (which is now a draggable marker) and allowing users to search for another location by name with Google Maps autocomplete, helping validate the shark's accurate location.

1 Introduction

Although sharks play a vital role in our ocean’s ecosystem they are one of the planet’s most data-deficient animals. Sharks face many threats, such as those from shark finning, that have led
to a rapid decline in their populations. These are vulnerable animals facing threats of overfishing and loss of habitat with estimates of up to 100 million killed each year. Sharks are the most threatened group of marine animals and for many species there is not enough data to confirm their conservation status. SharkPulse is a web and mobile application that allows users to help gather data on the world’s shark populations. This application encourages people to become involved in the conservation of sharks by allowing users to report their shark sightings and validate given shark image data. SharkPulse raises awareness of the current status of sharks through data visualization of its shark data records.

Our team is focused on the shark Validation Monitor feature of this website. SharkPulse gathers shark images through two social media platforms, Instagram and Flickr. For each source there is a validation monitor allowing users to select an image to validate and then provide key data validating the possible shark sighting. This system provides incentives for users by providing them with a point reward system for identifying and validating shark images. Our team’s goal is to improve the instagram validation monitor, so that users are able to provide location information easily. We also hope to generally improve the user experience through the addition of more autofill features and making aesthetic changes. With an improved user experience more citizens will be encouraged to use the application. Through this application any citizen willing to provide their knowledge and time can contribute to the conservation of sharks by providing shark sightings or validating images.

**Team Member Roles**

Khanh Pham: Back-end Lead/ Note Taker Lead
This team member was tasked with redefining the current social media source pipelines so that there would instead be one pipeline which can be toggled between.

Catalina Lemus: Front-End Lead/ Presentation Lead
This team member was tasked with refining the frontend of the Validation Monitor forms with the goal of also making it more simple to the user.

Mohammed Al Ansari: Team Lead/ Meeting Coordinator
This team member was tasked with solving the challenge of finding a library in order to autofill maps location names in an r-shiny application

1.1 Requirements and Objectives
After discussing with our client, we have established specific, measurable, and achievable goals. These objectives are aimed at enhancing the design of the current website, produce a generalized validationMonitor app able to load different social network data streams and adapt the validation form to the specific nature of the social network data and the information that needs to be validated. The generalized validationMonitor will be done by merging two existing validation apps, one for Flickr and another one for Instagram. The Flickr app is already Rshiny. The Instagram app has been written in PHP. The main requirements for this project are:

1. The ability to navigate the validation monitor pages viewing either Instagram or Flickr photos.
2. Improve website’s User Interface, make validation form more user friendly and responsive.
3. Integrate Instagram Validation form with Google Maps allowing users to search for location and move a marker to an accurate position.

1.2 Client
The individual that we are serving as our client is Dr. Francesco Ferretti, who currently holds the position of an Assistant Professor in the Department of Fish and Wildlife Conservation. Dr.
Ferretti is dedicated to studying how human activities have impacted marine ecosystems and finding ways to promote sustainable usage of marine resources. To accomplish this, he concentrates his research on applying data science techniques and analyzing big data to address ecological challenges and come up with innovative solutions for the ocean. He also has passion for sharks which inspired him to build the SharkPulse project.

### 1.3 Challenges

While working on SharkPulse, our team encountered various obstacles during the process. One of the main challenges was that our team members lacked familiarity with the technology stack currently in use. We had no prior experience working with R Shiny and WordPress, and as a result, we were unable to immediately contribute to the website. We had to invest time and effort into learning these platforms, which delayed our progress in making an immediate impact on the website. We also experienced a hard time when trying to set up and run it locally since we could not find the way to automatically install packages in Rstudio. Therefore, we had to install each package manually which was time consuming. Additionally, we experienced a hard time using Google Map Places libraries to implement places autocompletion for the search bar.

### 2 User Manual

#### 2.1 User Login

The users of SharkPulse can start out by first logging in. This is in order for users to have an account connected to any of their actions on the Validation Monitor. When a user logs into SharkPulse the shark images they validate are tracked and used for earning points which places users on a leaderboard. Users are incentivized to login to their account in order to earn points. Users are able to login with an existing account, register to create a new account, or use their Google account to login, see Figure 1.
2.2 Validation Monitor

The validation monitor is a gamified system used to validate shark images provided from Instagram or Flickr. SharkPulse is looking for more shark images to be validated by humans in order to gather more data on sharks. You can navigate to the validation monitor by selecting either the “Instagram -Validation Monitor” option or the “Flickr -Validation Monitor” option. There are two separate maps for each social media source, Instagram and Flickr. You can initially pick between the monitor when hovering over the “Monitoring” tab on the SharkPulse website, as seen in Figure 2. Validation_Rec.mov is a quick demo video to walk through use of the validation monitor, changes and new features completed by the team.

![User Login Interface](image)
Figure 2: SharkPulse Tab Bar, Monitor Hovering

Once you are on the monitor page of either social media source you are capable of toggling between sources all within the same webpage. As seen in Figure 2.1 a user can choose to switch between sources through a toggle option at the top of that source’s map.

Figure 2.1: Media Source Toggle and Map

2.2.1 Instagram Monitor

After selecting the “Instagram - Validation Monitor” option or selecting “Instagram” on the map’s media toggle a user is directed to a validation monitor on the SharkPulse web page shown in Figure 3.1. This monitor web page allows users to view the available images to validate. Each black shark fin on the map represents an image taken from Instagram in need of
human verification. This map can also be zoomed in and out of using the + and - buttons on the upper left corner of the map. After clicking on a shark fin a user is directed to a form to use for image validation as seen in Figure 3.2.

![Validation Monitor Map](image)

**Figure 3.1:** Instagram Validation Monitor Map

This form is filled out by the user based on the Instagram image provided in the upper left portion of the form. The user must fill out the shark’s common name, species name, shark location, and answer three yes or no questions before they submit the form. This form is adapted to an Instagram sourced image specifically with a need for human verification of an image location and having an extra question about whether the image is a repost. The question about whether the image is of a shark, the common name, and species name are required to fill out the form. The possible options for shark common name and species name are autofill options which a user can scroll through. Information on the suggested species of the photo and the identification guide to help a user identify shark information are provided on the form right above the questions. Once a user has finished filling in the form they can select “submit” at the bottom right of the form. If a user wants to exit the form without submitting any information, they can select the X in the top right corner of the form.
Figure 3.2: Instagram Image Validation Form

2.2.2 Flickr Validation Monitor

After selecting the “Flickr - Validation Monitor” option or “Flickr” on the map’s media toggle a user is directed to a validation monitor on the SharkPulse web page shown in Figure 4.1. The user can then select a shark image to validate by clicking on one of the black shark fins visible on the map just like in the Instagram Validation Monitor. After selecting a shark fin the user will be directed to a new form containing a shark image in the upper left, location data in
the lower left and questions to answer on the right hand side as shown in **Figure 4.2**.

![Map Image](image)

**Figure 4.1**: Flickr Validation Monitor Map

The user is already provided with the location data in this case due to the geotag provided by Flickr. Here a user does not need to validate the location data of the given shark image. The shark common name and species name are also autofill questions here so that a user can scroll through options for and type to find. Since this media source is Flickr, the form questions are changed so that only two questions are asked, since the previous repost question is not relevant. A user is able to exit this form anytime by selecting the X at the top right corner of the form. Then after a user submits their form using the submit button on the bottom right corner of the form they are directed to a thank you page as shown in **Figure 4.3**. This confirms that the user has successfully submitted a new shark image validation including the data the user provided. At the bottom of the page the user can also then select the return to map button in order to return to the original Flicker Validation map they were previously viewing.
Please validate the following shark image.

Figure 4.2: Flickr Validation Form

Thank You

Here is the information you have submitted:

1. Shark: yes
2. Common Name:
3. Species Name:

4. image ID: 4312
5. email:

Figure 4.3: Flickr Validation Thank You
3 Developer’s Manual

3.1 Navigate Monitor for Instagram or Flickr

1. The code shown in Figure 5.0 creates a Shiny user interface for a "ValidationMonitor" application and stored ui.R files. The ui.R code uses several packages which need to be installed and loaded before the application can run. The required packages are:
   ● shiny: The basic package for building web applications using R.
   ● DBI: Provides a generic interface to communicate with different types of databases.
   ● RPostgreSQL: An implementation of the DBI interface specifically for PostgreSQL databases.
   ● ggmap: An extension of ggplot2 for visualizing spatial data and maps.
   ● leaflet: A package to create interactive web maps.
   ● mapview: Provides functions to visualize spatial data and models on top of static maps from various online sources.

2. To install these packages, use the following R command:
   ```
   install.packages(c("shiny", "DBI", "RPostgreSQL", "ggmap", "leaflet", "mapview"))
   ```

3. The Validation Monitor UI has a nav bar contains two tabs:
   ● Flickr tab: includes a leafletOutput named "sharkmap", which will display an interactive map created with the leaflet package. The map is intended to take up the full width and height of its container.
   ● Instagram: similar to the Flickr tab, but leafletOutput named "shark" instead of “sharkmap”. The structure and functionality are identical to the Flickr tab, with a full-width and height map and a transparent control panel.
Flickr and vice versa, follow these steps:

```r
shinyUI(navbarPage("ValidationMonitor",
  id = "nav",
  collapsible = TRUE,
  tabPanel(
    tags$span("Flickr", icon("flickr")),
    # Custom CSS to hide the default logout panel
    tags$head(
      tags$link(rel = "stylesheet", type = "text/css", href = "style.css")
    )
  ),
  fixedPage(tabPanel(
    "Map",
    div(class = "outer",
      # Include our custom CSS
      # mapping
      leafletOutput("sharkmap", width = "100\%", height = "100\%"),
      # aggiungi pannello trasparente
      absolutePanel(
        id = "controls",
        class = "panel panel-default", fixed = TRUE,
        top = 1, left = "auto", right = 25, bottom = "auto",
        width = "auto", height = "auto"
      )
    )
  ))
),
  tabPanel(
    tags$span("Instagram", icon("instagram")),
    # Custom CSS to hide the default logout panel
    tags$head(
      tags$link(rel = "stylesheet", type = "text/css", href = "style.css")
    ),
    # Insert Instagram content here
    fixedPage(tabPanel(
      "Map",
      div(class = "outer",
        # Include our custom CSS
        # mapping
        leafletOutput("shark", width = "100\%", height = "100\%"),
        # aggiungi pannello trasparente
        absolutePanel(
          id = "controls",
          class = "panel panel-default", fixed = TRUE,
          top = 1, left = "auto", right = 25, bottom = "auto",
          width = "auto", height = "auto"
        )
      )
    ))
  )
))
```

**Figure 5.0**: The Validation Monitor Nav Bar

### 3.2 Interact with R-shiny Google Maps

To interact with R-shiny Google Maps, follow these steps:
1. Install the 'shiny' package for building the R-shiny app for integrating Google Maps in R-shiny. This will provide the necessary tools and libraries to create the interactive map.

2. Add a map to the UI using the 'googleway' package, allowing the user to interact with it and view the locations they search for. This map can be customized to suit the needs of the application, such as setting the initial zoom level and map type.

3. Create a server function using the 'googleway' package to retrieve the location data that the user has entered. This function can be used to fetch the location data entered by the user and process it for display on the map.

4. Use R-shiny reactive expressions to filter specific places based on user input and display them on the interactive map. This allows users to filter the locations they want to view on the map based on specific criteria, such as distance or type of location.

5. In addition to displaying locations on the map, add a sidebar with additional information about the locations, such as photos, reviews, or ratings.

3.3 Autocomplete options of shark names.

1. The get_name() function in Figure 5.1 retrieves both scientific and common names from the taxonomy_2 table in Figure 5.2. This table, taxonomy_2, contains updated data on elasmobranch species as listed in the IUCN Red List database. It includes various columns such as taxon ID, kingdom, phylum, class, order, family, genus names, scientific name, taxonomic authority, infra-rank or infra-name information, species population, conservation category, and the main common name of the species.

2. The get_name() function is called within the shinyServer, which houses the server-side logic of the Shiny application. The output returned by this function is then passed to the common name and species name input fields in each pipeline. This allows for autocompletion of shark names as users type.
Figure 5.1: get_name() function

```r
get_name <- function() {
  con <- connectPelagic()
  fetch(
    dbSendQuery(con,
    statement =
      paste(  
        "select distinct scientific_name, main_common_name from taxonomy_2;",  
        sep = ""
      ),
      n = -1
    )
  )
}
```

Figure 5.2: taxonomy relational scheme

3.4 Instagram and Flickr Tables

1. The get_insta_data() shown in Figure 5.3 function retrieves data from Instagram table related to pelagic marine life. The Instagram table as shown in Figure 5.4 includes various columns such as the date of the post, the text of the post, the common and species name of the marine life featured in the post, the latitude and longitude of the location where the post was made, the post URL, the name of the shark featured in the post, any notes related to the post, the probability of the species being correctly identified, the
location where the post was made, whether the post has been reposted, whether the post has been validated, the validator's name, the confidence level of the validation, any hashtags associated with the post, the user's email address, the post ID, the aquarium where the post was made, the time the validation was made, and any comments related to the post. This data can be used for analysis or display on a website or application related to pelagic marine life.

```r
get_insta_data <- function() {
  con <- connectPelagic()
  fetch(dbSendQuery(con,
    statement =
    paste("select * from instagram where ((sd_species='Galeocerdo cuvier')
    or (sd_species='Rhincodon typus')) and (repost='no')
    and (longitude is not null) and (validated='f')
    order by random() limit 10;", sep = "")
    ), n = -1)
}
```

**Figure 5.3**: get_insta_data() function
2. The `get_Flickr_data()` in **Figure 5.5** function retrieves data related to pelagic marine life from the `data_mining` table. The table as shown in **Figure 5.6** includes various columns such as the ID of the post, the date and time the post was made, the search query that returned the post, the latitude and longitude of the location where the post was made, the name of the image file, any notes related to the post, the user’s email address, whether the post has been validated, information about any sharks featured in the post (including common name and species name), information about any additional users who have contributed to the validation process, information about the top three identified species and their probabilities, the URL of the post, the owner of the post, any comments related to the post, the aquarium where the post was made, the time of the validation, whether the
validation was performed by an expert, the common name of the species as identified by an expert, and the species name as identified by an expert. This data can be used for analysis or display on a website or application related to pelagic marine life.

```r
get_flickr_data <- function() {
  con <- connectPsql()
  fetch(dbSendQuery(con,
    statement =
      paste("(select img_name, id, date, time, latitude, longitude,
        validated, url, top1_species, top2_species,
        top3_species, top1_probability,
        top2_probability, top3_probability
        from data_mining
        WHERE (validated='f') AND (control='f') AND (species_name_1 is not null)
        AND (top1_probability is not null) AND (url is not null)
        order by random() limit 5) union
        (select img_name, id, date, time, latitude, longitude,
        validated, url, top1_species,
        top2_species, top3_species, top1_probability,
        top2_probability, top3_probability
        from data_mining
        WHERE ((is_shark_1='True') OR (is_shark_1='TRUE'))
        AND (url is not null) order by random() limit 10)
      union
      (select img_name, id, date, time, latitude, longitude,
      validated, url,
      top1_species, top2_species, top3_species, top1_probability,
      top2_probability, top3_probability
      from data_mining where (top1_probability is null) AND (url is not null)
      order by random() limit 1);", sep = "")
  ), n = -1)
}
```

**Figure 5.5:** get_flickr_data() function
<table>
<thead>
<tr>
<th>Column</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
</tr>
<tr>
<td>date</td>
<td>date</td>
</tr>
<tr>
<td>time</td>
<td>time without time zone</td>
</tr>
<tr>
<td>query</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>latitude</td>
<td>numeric</td>
</tr>
<tr>
<td>longitude</td>
<td>numeric</td>
</tr>
<tr>
<td>img_name</td>
<td>text</td>
</tr>
<tr>
<td>notes</td>
<td>text</td>
</tr>
<tr>
<td>users_email</td>
<td>text</td>
</tr>
<tr>
<td>validated</td>
<td>boolean</td>
</tr>
<tr>
<td>is_shark_1</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>is_shark_2</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>users_email_1</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>users_email_2</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>common_name_1</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>common_name_2</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>species_name_1</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>species_name_2</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>control</td>
<td>boolean</td>
</tr>
<tr>
<td>yes_e10</td>
<td>integer</td>
</tr>
<tr>
<td>no_e10</td>
<td>integer</td>
</tr>
<tr>
<td>comment</td>
<td>character varying(500)</td>
</tr>
<tr>
<td>url</td>
<td>character varying(350)</td>
</tr>
<tr>
<td>owner</td>
<td>character varying(250)</td>
</tr>
<tr>
<td>top1_species</td>
<td>text</td>
</tr>
<tr>
<td>top2_species</td>
<td>text</td>
</tr>
<tr>
<td>top3_species</td>
<td>text</td>
</tr>
<tr>
<td>top1_probability</td>
<td>text</td>
</tr>
<tr>
<td>top2_probability</td>
<td>text</td>
</tr>
<tr>
<td>top3_probability</td>
<td>text</td>
</tr>
<tr>
<td>aquarium</td>
<td>character varying(50)</td>
</tr>
<tr>
<td>valtime</td>
<td>timestamp with time zone</td>
</tr>
<tr>
<td>expertval</td>
<td>boolean</td>
</tr>
<tr>
<td>expertcommon</td>
<td>character(100)</td>
</tr>
<tr>
<td>expertspecies</td>
<td>character(100)</td>
</tr>
</tbody>
</table>

**Figure 5.6:** data mining relational scheme
4 Evaluation

4.1 User Testing

Changes were made to the design of both the Flickr and Instagram forms within the Validation Monitor. These design changes were made with the purpose of having a more easily navigable experience for the user with basic actions such as submitting the form and exiting the form. Switching between Instagram and Flickr Monitors was also changed with this same purpose. To test these changes and receive feedback I asked users to test the Validation Monitor before the changes were made and then to later test the Validation Monitor after these changes were made. Before the changes were made based on the user feedback we concluded that they were experiencing trouble finding where to submit each form, at first skipping over the button entirely; had confusion over the maps on each form (specifically the Instagram form); felt overwhelmed by the structure of the form; and felt like the style of the form was not cohesive with the rest of the website. We made changes to try to directly address these concerns.

Testing after the changes made to the monitor included performing the same tasks as before the changes and seeing if users felt this was a more easily navigable experience. From their feedback, the general insight we gained was that being able to toggle between the Instagram and Flickr maps was an improvement that they were able to quickly use to navigate without confusion. When asking them to perform actions on the forms themselves they found that the submit button was now clear and where they would expect it; they also found the simple structure of the form a little less overwhelming when looking for information. They really enjoyed the new map on the Instagram form, finding it easy to navigate and clear in what it was asking for.
5 Future Work & Projections

5.1 Future work for subsequent semester

While working on the SharkPulse App this semester, our team figured out several places that should be improved by people in subsequent semesters. First thing, there are three buttons: Pulse monitor, Flickr Monitor, and Instagram monitor on the monitoring page (which does nothing). We need to change the functionality of these buttons given the changes we made where a user can toggle between media sources on their given maps within the same webpage. There should only be one button for monitor validating since toggling is done on one page. Currently, SharkPulse only gets photos from 2 main sources, Flickr and Instagram; we need to ensure that they fetch photos from other sources, in addition to Instagram and Flickr. One goal is to source images from Twitter as well, in the future. Additionally, we should add more OAuth options to the login page, instead of just allowing login with Google. This will provide users with more options which third parties can use in order to log in and access the website. Finally, we need to improve the Identification page by replacing the 2D picture of shark parts with a 3D photo, which will allow users to zoom in and out to more easily identify shark species. By implementing these changes, we can enhance the functionality and user experience of SharkPulse, and make it a more comprehensive and easily navigable platform for identifying and tracking shark populations.

Acknowledgments

Client: Dr. Francesco Ferretti, Assistant Professor, Department of Fish and Wildlife Conservation. SharkPulse creator, ferretti@vt.edu
Mentor: Dr. Edward Fox, Professor, Department of Computer Science, fox@vt.edu
Graduate Student Expert: Jeremy Jenrette, CS4624 Spring 2021, jjeremy1@vt.edu
References