Bird Dropper

The All-In-One Web Application for any Level of Birder

Luke Maschoff, Joshua Trujillo, Aaron Beninghaus, Ben Blair, and Jonathan Ro

Github Accounts:

- Luke Maschoff: luma1620 & LMascher (most of the code I wrote was through LMascher, but for some reason doesn't show up on the contributors for some reason, but shows in the commit history).
- Joshua Trujillo: jotrujil03
- Aaron Beninghaus: linexz
- Ben Blair: ben-blair
- Jonathan Ro: JRhooooo

Bird Dropper is an all in one solution for any avid bird watcher to track the birds they have spotted, search for information regarding birds they are curious about, and even find out information about a bird just from uploading an image if they don't know what the bird is called. With Bird Dropper's social feature, users are also able to follow their friends and see what birds they have spotted as well, and are able to like and comment on users posts. Users are able to create collections of birds, so if they are tracking a certain category, or if they are on a trip to a certain region, they are able to have a collection of these specific birds. On top of our unique search features, social media features, and collection features, Bird Dropper also has a random bird search feature if they are in a learning mood, and also has a Fact of the Day, that provides users with a new fun fact every single day. Bird Dropper is an all in one web application for any bird enthusiast, regardless of their experience.

Our Repo: https://github.com/jotrujil03/bird-dropper/tree/main

Our GitHub Project Board: https://github.com/users/jotrujil03/projects/4/views/1

Video Demonstration:
https://drive.google.com/file/d/18OGEzfnKZ6qwYLt7P-ew83-FFoCLEvVz/view?usp=sharing

Our website is hosted via Render, and you can look at it here via this link:
https://bird-dropper-web.onrender.com/

Contributions:

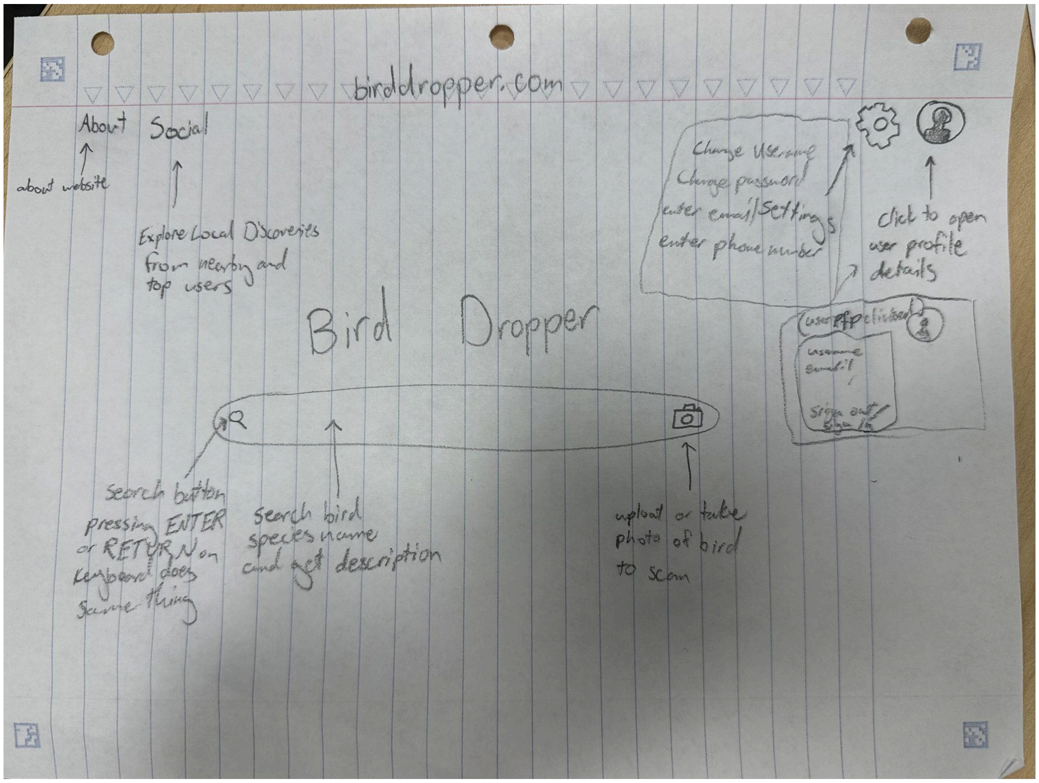
- Luke Maschoff:
  - Profile Page
  - Search Bar Functionality
  - Collection Page Functionality
  - Random Bird Search Button
  - Fun Fact of the Day feature
  - API Integration for searching by image
  - Render fixing
  - Front end of home page, and about page
  - Comments on code
  - Condensed index.js
- Joshua Trujillo:
  - Navigation Bar Frontend & Backend
  - Login/Register Backend fixes
  - Search Bar backend functionality for typed queries and image upload
    - Wikipedia Crawler & Google-Vision implementation
  - Notification System Frontend & Backend
  - Render/Database setup/fixes
  - Cookies for user sessions
  - Repo Initialization
  - README setup.
- Aaron Beninghaus:
  - Collections Page Frontend & Backend
    - Upload/Delete Photos
    - Add Description
    - Ability To like Collection Post
    - Sort collection by most recent or by likes
  - Social Page Following Functionality
    - View Following Post’s only option
    - Search Users and Follow them
  - Following Page Frontend & Backend
    - Allow you to view the collection of the people you follow
    - See who follows you
  - Settings Page Backend
    - Change username and password
  - Profile Page Bio Backend

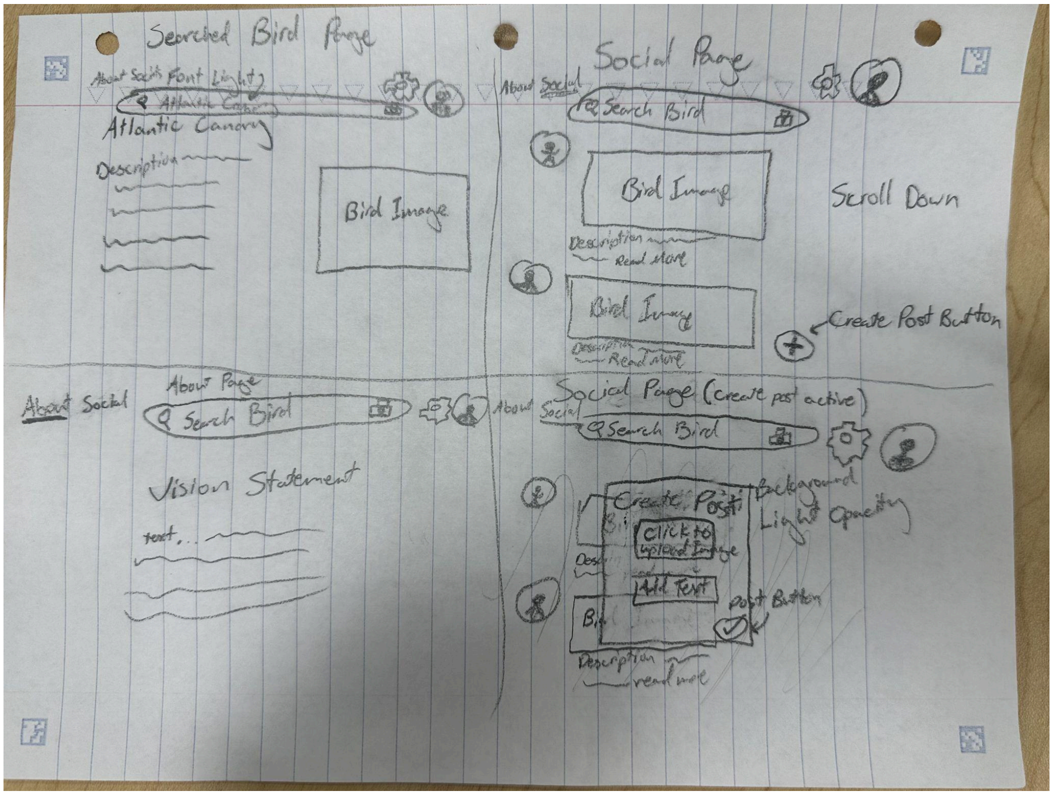
Ben Blair:

- Front End

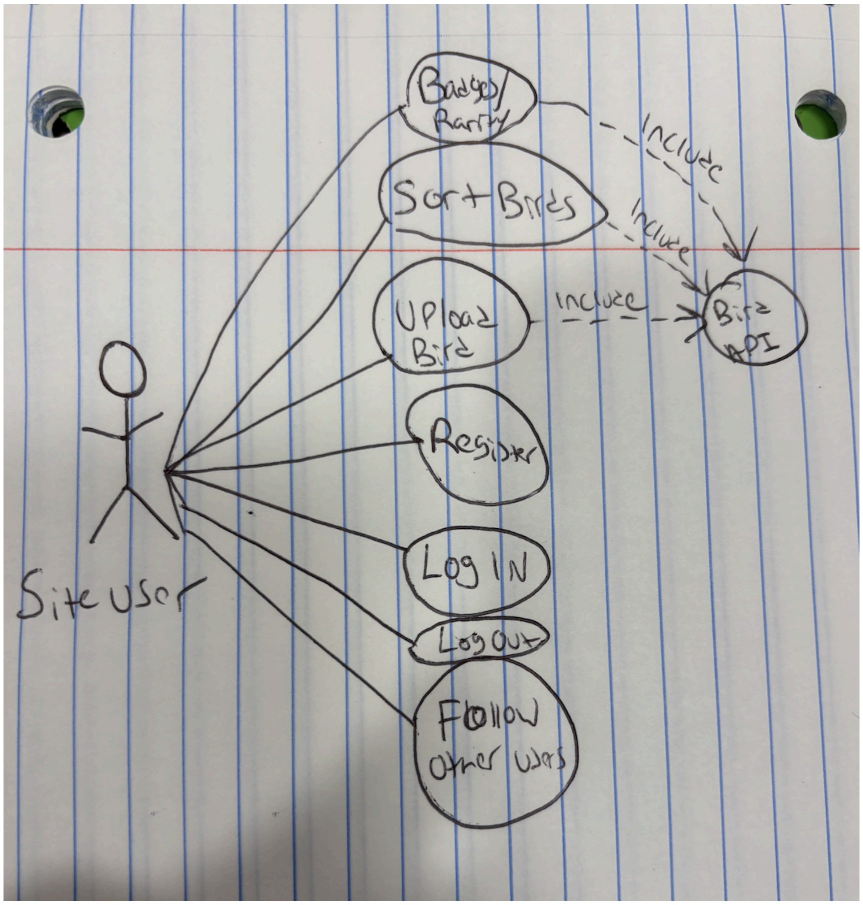
    - Home Page
    - About Page
    - Social Page
    - Favicon
    - Browse Popular Birds Page
    -
  - Back End
    - Social Page
      - Post
        - Photos go to database (supabase)
      - Like/Comment buttons
      - Delete Post/Comment
    - Store posts/profile pictures to database (Supabase)
- Jonathan Ro:
  - Profile Page
    - Front and Back End
  - Uploaded Photos Show On User Profile
    - Front and Back End
  - Login Page
    - Back End
  - Register Page
    - Back End
  - User Database
    - Back End
  - View Other Profiles
    - Front and Back End

Wireframes:


birddropper.com
About
Social
about website
Explore Local Discoveries from nearby and top users
Change Username
Change password
enter email
enter phone number
Settings
click to open user profile details
user pfp clicked
username
email
Sign out/ Sign In
Bird Dropper
search button pressing ENTER or RETURN on keyboard does same thing
search bird species name and get description
upload or take photo of bird to scan


Searched Bird Page
About Social Font Light
Atlantic Canary
Atlantic Canary
Description
Bird Image
Social Page
About Social
Search Bird
Bird Image
Description
Read More
Bird Image
Description
Read More
Scroll Down
Create Post Button
About Page
About Social
Search Bird
Vision Statement
text...
Social Page (create post active)
About Social
Search Bird
Create Post
Click to upload Image
Add Text
Background Light Opacity
Post Button
Description
read more

Use Case Diagram:



Test Plan Observations:

1. Render Home Page

- Our user visited the homepage of our website, by clicking on our link provided
- The user expected to land on the home page as the first thing when they clicked the link
- The user was correct in their assumption, and landed on our home page right away
- They were able to access all the features that they assumed they were able to before they saw our landing page, including the social feature, sign up/login buttons, and our search features
- We expected this result, and everything went smoothly, resulting in no changes needed for rendering the home page

2. Registration

- Our user was able to register, using the sign up button
- The user successfully was able to easily navigate to the profile icon in the top right, and when they clicked profile, they easily found the register button. No confusion on this step
- At first, the user tried to get through registration quickly, and for their email just put a bunch of characters. Obviously this isn't good so we made a change here, and made sure to validate for legitimate emails
- We made it so that if a user doesn't give a valid email address, we tell them that they have to try again with a valid email address
- We understood that some users may just want to sign up for one time, so we somewhat expected this

3. Failed Login

- When our user was instructed to input a wrong login, they did so and were able to receive a message clearly stating that they inputted an invalid username/password
- This is exactly what we wanted to happen, and our users behavior was consistent with what we expected
- We considered stating if it was specifically the email, or password that was invalid, but after doing research and seeing how big, popular websites do this, they don't explicitly say, and we assumed that this is industry standard for security reasons

4. Successful Login

- After we instructed our user to fail a login, we then instructed them to login successfully
- The user successfully navigated to the login button, and logged in with their correct credentials
- This behavior was exactly what we expected, and the user was redirected to the home page, which is exactly what we wanted
- The user reported that they knew they were successfully logged in because they could now see their profile picture
- We did not have to make any changes, as this is exactly what we wanted to happen.