Reiman Garden Butterfly Application

Design Document

Team #5
Client: Nathan Brockman
Advisor: Ashraf Gaffar
Members:
Bailey Wanders, Johann Guepjop, Caleb Donavon, Sam Sells
Email: sddec22-05@iastate.edu
Website: sddec22-05.sd.ece.iastate.edu

Fall 2022, Version 2

Development Standards & Practices Used

Standards:

- **ISO/IEC/IEEE 23026-2015:** Engineering and Management of websites for system, software, and service information (Systems and Software Engineering)
- **ISO/IEC 23026-2006:** Recommended Practices for the Internet, Web site Engineering, Management, and Life Cycle (Software Engineering)
- **IEEE 2301-2020:** Guide for Cloud Portability and Interoperability Profiles (CPIP)

Practices:

- Agile Development method
- S.O.L.I.D. Design principles

Summary of Requirements

- Create user hierarchy (visitor, registerd user, superuser)
- Improve and modernize design of site
- Improve user input forms
- Create website template for other organizations to use
- Build scalable database
- Reduce number of security vulnerabilities

Applicable Courses from Iowa State University Curriculum

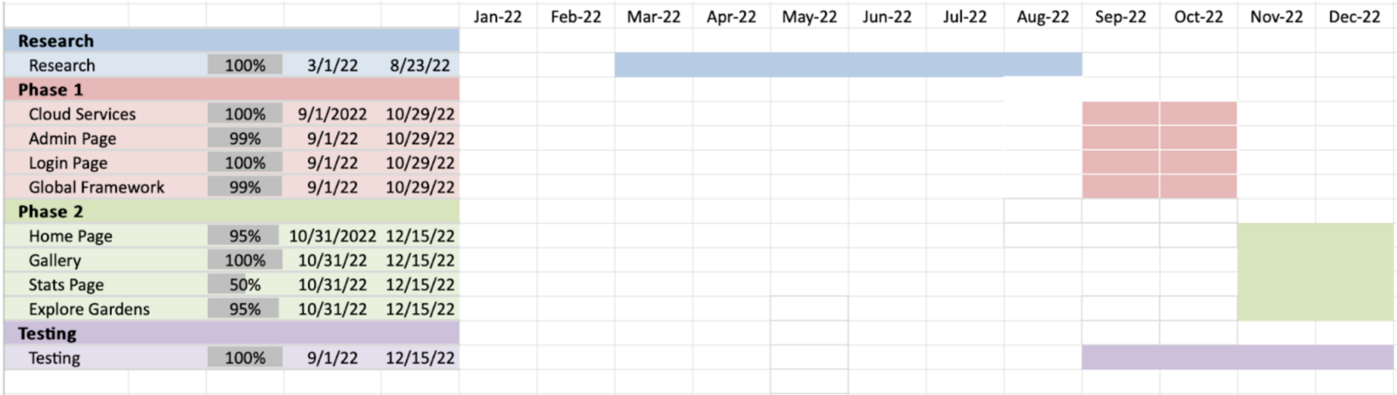
- COMS 227 : Object-oriented Programming
- COMS 228 : Introduction to Data Structures
- COMS 309: Software Development Practices
- CPRE 281: Digital Logic
- CPRE 381: Computer Organization and Assembly-level Programming
- CPRE 308: Operating System: Principles and Practice
- ENGL 314: Technical Communication

New Skills/Knowledge acquired that was not taught in courses

- Cloud-based development
- Reducing and preventing security vulnerabilities
- Specific front-end development tools used in project
- Building high-fidelity prototypes

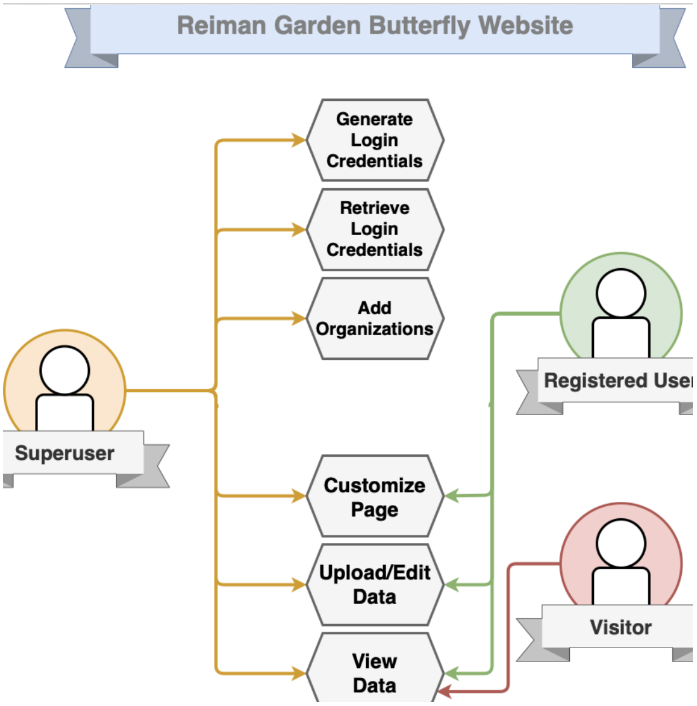
Table of Contents

List of figures/tables/symbols/definitions



				Jan-22	Feb-22	Mar-22	Apr-22	May-22	Jun-22	Jul-22	Aug-22	Sep-22	Oct-22	Nov-22	Dec-22
Research															
Research	100%	3/1/22	8/23/22			🟦	🟦	🟦	🟦	🟦	🟦				
Phase 1															
Cloud Services	100%	9/1/2022	10/29/22									🟥	🟥		
Admin Page	99%	9/1/22	10/29/22									🟥	🟥		
Login Page	100%	9/1/22	10/29/22									🟥	🟥		
Global Framework	99%	9/1/22	10/29/22									🟥	🟥		
Phase 2															
Home Page	95%	10/31/2022	12/15/22											🟩	🟩
Gallery	100%	10/31/22	12/15/22											🟩	🟩
Stats Page	50%	10/31/22	12/15/22											🟩	🟩
Explore Gardens	95%	10/31/22	12/15/22											🟩	🟩
Testing															
Testing	100%	9/1/22	12/15/22									🟪	🟪	🟪	🟪

Gantt Chart of Project Tasks



Use case Diagram of Project

1 Team

1.1 Team Members

Bailey Wanders

Samuel Sells

Johann Guepjop Megaptche

Caleb Donavon

1.2 Required Skill Sets for Your Project

Front-End Development - JavaScript, CSS, etc.

Cloud Engineering - research and development

Back-End- PHP scripting

1.3 Skill Sets covered by the Team

Front-End Development - Sam, Bailey, Caleb, Johann

Cloud Engineering - Sam, Bailey, Caleb, Johann

Back-End- PHP scripting - Caleb, Johann

1.4 Project Management Style Adopted by the team

Agile: 2-week sprints

Product Owner: Nathan Brockman

Scrum Master: Sam

Dev Team: Bailey, Caleb, Johann

1.5 Initial Project Management Roles

Product Owner: Nathan Brockman

Scrum Master: Sam

Dev Team: Bailey, Caleb, Johann

2 Introduction

2.1 Problem Statement

Our project is based upon improving/building a site for Reiman gardens that meets the requirements of the 3 user types. There are three main improvements we are working on. Improvement of the data entry side for admins. This includes the abilities to add shipments, enter butterfly data, and the ability to edit the entries. Next modernize the site's features to be up to date and appeal visually to the users. This aspect is more focused on the needs of a visitor of the butterfly garden. This includes a updated statistics and gallery page that is user friendly. The last pertains to outside butterfly gardens by scaling the project to allow them to create their own site. They will be able to customize their own site and enter data for their garden. All of these sites will have a home landing page controlled by Reiman Gardens .

2.2 Requirements & Constraints

Functional requirements

- Superusers can create pages, accounts, or credentials for other organizations or individuals
- Superusers can retrieve login credentials for individuals
- Registered users, which includes superusers, can create accounts for other registered users
- Registered users can upload and edit data on their organization's site
- Registered users can customize their organization's site
- All users, which includes visitors, can view data for all organizations

Resource requirements (labor, IDE/software/etc, time, money

- Labor: At least 4 experienced developers are required to complete this project
- Materials: Team members will need
  - a version control tool (e.g. GitLab)
  - a database to be shared between all organizations
  - A service to host each organization's site
- Time: The project should be functional and meets the functional requirements stated above in 1
  - year
  - **(constraint)**
- Budget/Cost: The project will be funded by Reiman Gardens with a budget determined by the
  - client **(constraint)**

Qualitative aesthetics requirements

- All elements of each site should be consistent (buttons, fonts, headers, etc.)
- Each organization should be able to customize the color scheme of their particular site
- All pages of the site should reduce the amount of whitespace
- Each site should maintain style on mobile and desktop platforms

Economic/Market requirements

- Each site should be easy enough to navigate by users of all technical ability levels in order to reach the maximum number of site visits
- The project should streamline several processes in order to be appealing to as many organizations

as possible

Environmental requirements

- Acknowledge product lifecycle to minimize potential work needed in the future
  - Cut down on future resources needed
- Our project should not be needlessly complex or have unnecessary pieces
  - Operational Efficiency
- During extended use of our web app caution and operational guidance should be given on aspects like
  - Temperature, Air Intake, and humidity
  - How intensive is the software running?
- Cut down on carbon emissions and increase energy efficiency by using established datacenters like the cloud

UI requirements

- Able to access all parts of the website through a mobile device with ease
- Users can't navigate away from the website on the kiosk copy of the web app
- Page should be operable without any data to report
- Should be able to maintain several dozen facilities on the backend
  - Should be able to handle up to 100 active users
- Our webpage should follow the three click rule
  - Able to get anywhere on the site within three clicks
- Web app should have distinct differences between user groups.
- UI should be able to send and receive data from the cloud
- Information, data, photographs, and anything else should be displayed in a visually pleasing way

2.3 Engineering Standards

- **ISO/IEC/IEEE - International Standard - Systems and Software Engineering - Engineering and Management of websites for system, software, and service information:**

- **IEEE/ISO/IEC 23026-2015:**
  - This standard is likely to be relevant to our project since it provides a set of guidelines on how to design efficient web pages for users to have easy access to the information displayed. This will help us improve the current user interface and set up for the butterfly gallery.

- **ISO/IEC 23026:2006, Software Engineering - Recommended Practices for the Internet - Web site Engineering, Management, and Life Cycle: IEEE/ISO/IEC 23026-2006:**
  - This standard could also be helpful be helpful in the completion of our project as it relates to the development and management of web site. Differently from the previous standard, it does not provide information on UI and design. However, by using both standards, we can design a more effective and long lasting product for our clients.

- **IEEE Guide for Cloud Portability and Interoperability Profiles (CPIP): IEEE 2301-2020:**
  - This standard relates to areas such as application interfaces, portability interfaces, management interfaces, interoperability interfaces, file formats, and operation conventions. Considering how much information and data needs to be stored and managed, the group has been leaning towards using some form of cloud computing (AWS for now). This set of standards can help us in our usage of cloud services to ensure that necessary information is properly managed.

2.4 Intended Users and Uses

Use Cases

- Client at butterfly park wanting to learn more about butterflies
- Butterfly park looking to organize their data
- Entomologist wanting to change standards for butterfly keeping
    - Look through data and statistics
- Common client of the park, not knowledgeable with technology, but using the kiosk in the park
- Butterfly park admin wanting a tool to adhere to USDA standards and fill out a yearly report
- Mobile user wanting to see how many butterflies are in the park and what kinds by using a mobile
- device

3 Project Plan

3.1 Project Management/Tracking Procedures

The team has chosen to adopt an agile project management style. We chose this method because this is the style that our client is most familiar with. This method will also allow us to break the project down into smaller, two-week sections. This will also provide more consistent deliverables for our client to view, giving them a more concrete view of our progress.

The team will be using Git and GitLab to track progress throughout the project. This is where team members will add files, resolve issues, and organize goals.

3.2 Task Decomposition

- We need to examine the existing database to determine if it meets our needs
- The long-term goal is to have a unique frontend for each organization, but all use the same database
- Site-Wide
    - Each page and its elements should be customizable by each organization
    - Remove access to associated social media pages
        - NO EXTERNAL CLICKS
    - Site needs to be scalable based on platform (mobile vs desktop)
    - Minimize excess whitespace
- Main/Home page
    - This page needs to keep most of the same elements that it already has but the style of the entire site can be changed
    - Necessary element is “Butterfly of the Day”
    - Need to add the total number of butterflies in flight and number of species in flight as a homepage element
    - Element that connects users to other organizations’ version of site
- Stats page
    - Generation of all stats needs to be automated
    - Necessary stats
        - Number of butterflies currently in flight

        - Number of species currently in flight
        - Total number of butterflies per year
        - Unique butterfly species
        - Chart that shows origin of all species
    - The “Butterflies in Flight” section does not need such a long timeline
    - Remove “Nathan’s Notes” from stat page
- Gallery
    - Needs to default to species currently in flight
    - Clean up whitespace
    - Keep pictures as large as possible
    - Be able to search for species within gallery
    - Clicking image OR text will bring user to species information page
- Cloud Services
    - Establish a cloud-based server to run the app through
    - Ensure handling of upwards of 100 concurrent users
    - Establish User hierarchy
- Admin Page
    - Create admin page for new organizations
    - Auto-generate username and password that can later be changed
    - View Shipments
        - Group shipments by date or supplier into single table entry
        - Minimize table until needed
        - Releases/Emergences
    - Add Shipment
        - Confirm Submission pop-up
        - Option to add/remove species entry as needed
    - Import/Export Shipments
    - Edit/Add Note

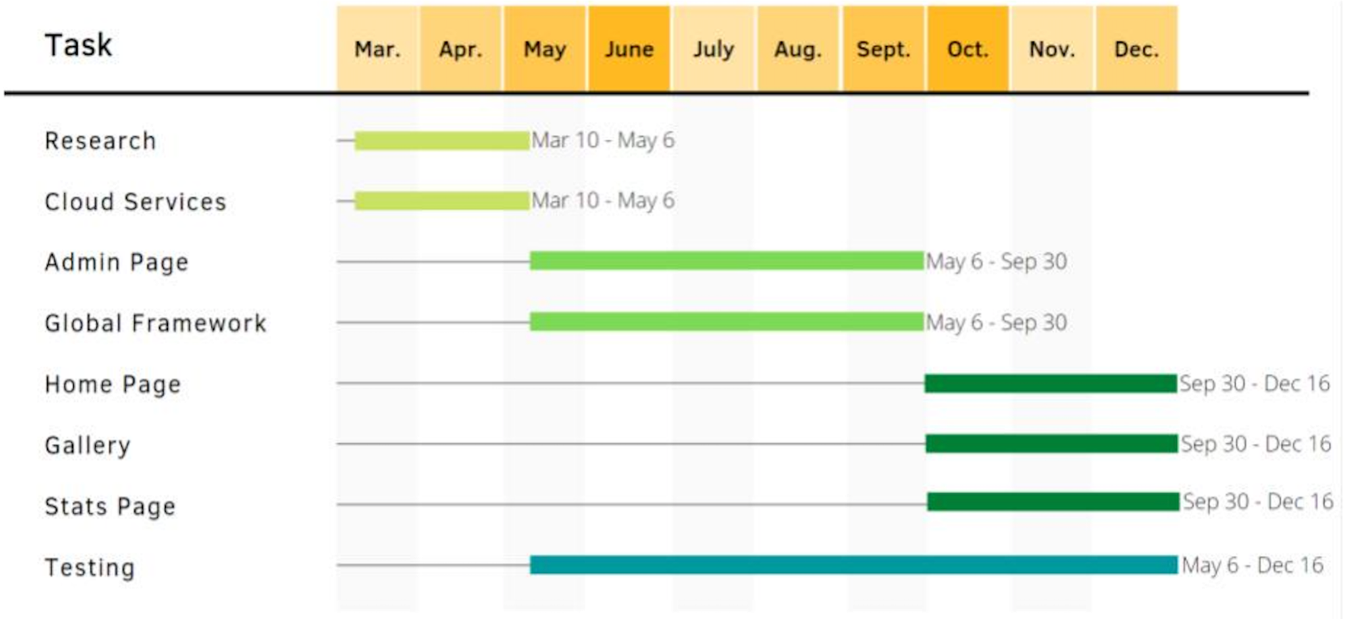
3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

Our proposed progress metrics will involve working through broad tasks that will slowly get more completed as we complete smaller tasks that fall under each of them. These Milestones are as follows:

1. Cloud Services
2. Home Page
3. Admin Page
4. Gallery
5. Stats Page
6. Global Park Framework
7. Testing
8. Research

Each of the above–mentioned tasks will be distributed into each of these 6 main tasks. Once all tasks and requirements have been distributed we will assign each of them a percentage as to their relevance to the overall completion of the broad task. A task will be considered completed when it is ready to be deployed on the main web app for public use. This evaluation will be relative to each task, but can also hinge on our client if needed. Basic functionality will be assessed through tests that will also be written by the team.

3.4 Project Timeline/Schedule



Sprint Milestones (more specifics can be found under the Task Decomposition section)

1. Research Cloud services and decide on which one we are going to use for our project
2. Setup the cloud service with basic functionality
3. Set up a server to hold necessary data
4. Test the cloud service to see if features needs are met
5. Establish User Hierarchy and Login
6. Create the View/ Add/Import/Export Shipments
7. Test Admin features to ensure implementation
8. Create a customizable homepage for each user
9. Create gallery
10. Create the Stats Page
11. Testing and closing for any unfinished problems
12. Testing and closing for any unfinished problems
13. Finalization

3.5 Risks And Risk Management/Mitigation

Risks	*Category*	*Probability*	*Impact*	*How to Avoid*
The cloud service does not fit our project's needs	Technical, Predictable	70%	2	Test multiple cloud services and research the best option.
User's unhappy with customizability	Customer Characteristic, Technical	*30%*	3	Testing, and surveying our client to get his opinion (user-driven design)
Our testing doesn't encompass all edge cases	Technical	30%	2	Independently develop tests to that in parallel we cover more ground (parallelism)
Design incompatible with our client	Business	20%	1	Constant communication with our client to make sure we're on the same page

Visually Unappealing Gallery	Customer Characteristic	15%	3	(User-driven design)
Unable to use Flutter	Technical	35%	4	Analyze the program before our implementation and ensure compatibility
Cannot generate a new site for each user	Technical	30%	1	Testing cloud services and talk to faculty advisor

Impact values: 1-catastrophic, 2-critical, 3-marginal, 4-negligible

3.6 Personnel Effort Requirements

Milestone	*Person Hours*	*Explanation*
Cloud Services	48	(4 sprints * 6hrs/week*4people)/2 tasks in milestone
Research	48	(4 sprints * 6hrs/week*4people)/2 tasks in milestone
Admin Page	64	(4 sprints * 6hrs/week*4people)/3 tasks in milestone
Global Framework	64	(4 sprints * 6hrs/week*4people)/3 tasks in milestone
Testing	124	(10 Sprints *6hrs/week* 4 people /)over 4 focus areas + (4 sprints * 6hrs/week*4people)/3 tasks in milestone
Home Page	60	(10 Sprints *6hrs/week* 4 people /)over 4 focus areas
Stats Page	60	(10 Sprints *6hrs/week* 4 people /)over 4 focus areas
Gallery	60	(10 Sprints *6hrs/week* 4 people /)over 4 focus areas

3.7 Other Resource Requirements

Requirements

- Cloud Service
- Git Repository
- IDE
- Code libraries

4 Design

4.1 Design Context

4.1.1 Broader Context

Area	**Description**	**Examples**
Public health, safety, and welfare	The main community we are targeting is the general public. Our secondary target would be other butterfly gardens hoping to share butterfly data and information. Our project is not of high risk of harming people if implemented incorrectly. The worst effects of our app being implemented ineffectively are poor customer interactions. This could potentially hurt business.	Bringing awareness to potentially endangered species, creating website accessible to all
Global, cultural, and social	Our project will aim to bring awareness to butterflies. Through our app we can reach a large audience of people and teach them more about butterflies. Our app also aims to help butterfly parks by giving them a platform to share their information with each other. If we don't succeed, butterfly practices could be hindered by this lack of ability to communicate.	It is also the goal of our client to provide a source of information about butterflies. It is important to both the team and the client to provide accurate information

Environmental	Our project will be used by many users around the world that are maintaining their own site for their butterfly garden. It is important that this stays suited for both staff and guests for years to come. We also aim to reduce our carbon footprint as much as possible.	Reducing the carbon footprint can improve the environment in which the butterflies live.
Economic	The goal of our project is to bring awareness to the general public of not only butterflies, but butterfly gardens as well. Creating a space where general users can receive more information has the potential to bring business to these butterfly gardens.	Product needs to remain affordable for organizations involved, product creates opportunities for economic advancement, brings business to organizations

4.1.2 User Needs

Visitor

- Visitors need a way to learn about butterflies quickly and concisely because they are visiting the park or are curious about butterflies around the world.

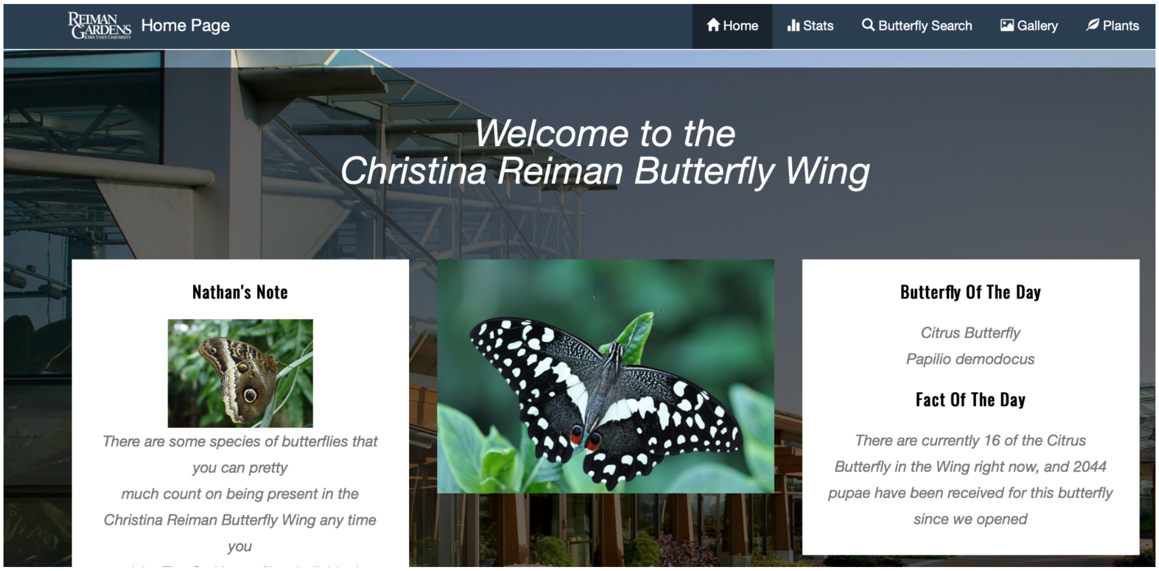
Superuser

- Superusers need a way to share their information with the public and store it in an efficient manner because there is no preexisting platform to do this and they could benefit from the organization that our website would provide.

Admin

- Admin users need a way to share information about their organization in order to bring business to their facility.
- Admin users need a way to collect data in a single location because analyzing data collected over a long period of time can help them make smart business decisions.

4.1.3 Prior Work/Solutions



Welcome to the
Christina Reiman Butterfly Wing

Nathan's Note

There are some species of butterflies that you can pretty much count on being present in the Christina Reiman Butterfly Wing any time you

Butterfly Of The Day

Citrus Butterfly
Papilio demodocus

Fact Of The Day

There are currently 16 of the Citrus Butterfly in the Wing right now, and 2044 pupae have been received for this butterfly since we opened

Pictured above is the current version of our project.

Advantages and disadvantages of current version of project

Advantages	Disadvantages
● Implemented ticker system for documenting emergences ● Functioning gallery ● Searches for both the scientific name and the common name ● Functioning homepage ● Functioning admin page ● See full shipping information broken down by which types of butterflies are ordered and by what supplier	● Not all pages are fully-functional ○ ex) Butterfly search page ● Is un-expanded and only addresses the current facility ● Inefficient and outdated design ● Unable to edit entries of shipments ● On some forms, input can not be deleted if added mistakenly ● Insecure username and password ● Unsearchable shipping information

4.1.4 Technical Complexity

1. The design consists of multiple components/subsystems that each utilize distinct scientific, mathematical, or engineering principles –AND–
2. The problem scope contains multiple challenging requirements that match or exceed current solutions or industry standards.

- The design includes a cloud-based database, specifically Amazon Web Services which has set a precedence for all cloud-based services in security and efficiency.
- The design includes a complex user hierarchy with three different levels, each with unique functionalities and requirements.
- The design of the project needs to allow different pages within the site to act independently of each other. Features that are shared between pages need to be within their own class. This is in accordance with the Single Responsibility Principle, which is well known within the developer community.

4.2 Design Exploration

4.2.1 Design Decisions

1. The team decided to start a new design from scratch, rather than using the preexisting version.
2. The team decided to use Amazon Web Services for the cloud-based database.
3. The team decided on a plan of action for working concurrently with other teams working on this project.
4. The team needs to decide what technologies to use in specific feature implementation.

4.2.2 Ideation

- Team decided to use Amazon Web Services for the cloud-based database
- Options considered:
    - AWS
    - Google Cloud
    - IBM Cloud
    - Azure
    - Digital Ocean

4.2.3 Decision-Making and Trade-Off

<< Best option Worst option>>

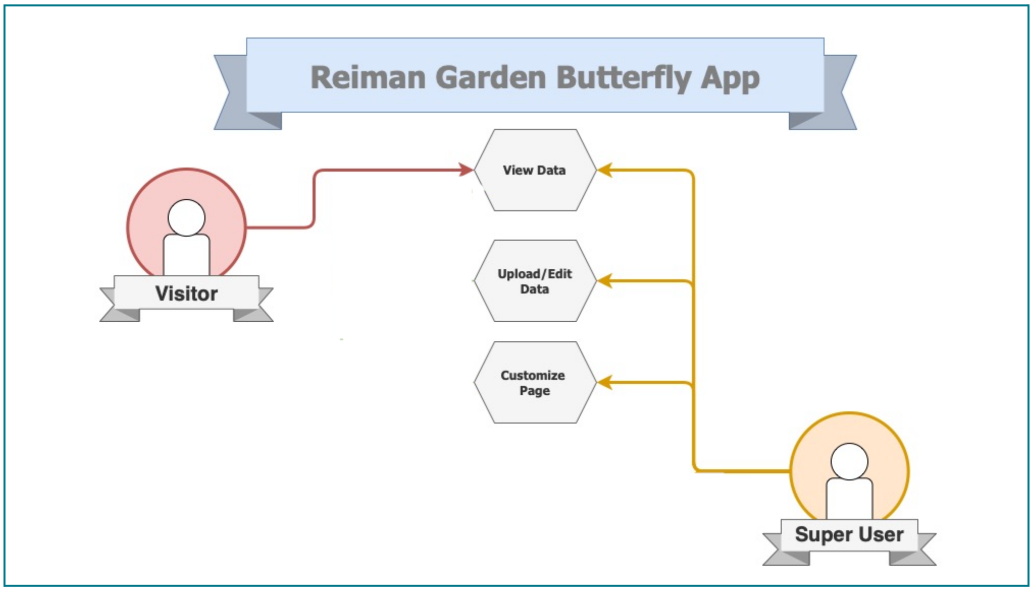
AWS	Google Cloud	Azure	Digital Ocean	IBM Cloud
Secure, flexible, efficient, easy to implement, well-documented	Flexible, scalable, easy to implement. The storage options are not ideal. The options provided by the most economic	Not as well-documented, easy to implement, not most efficient. Many hidden fees when doing data transfers.	Not as common, used in current version of project.	Not recommended by faculty advisor. Very “black-box” you send them information and you get something

	choice for our project could be insufficient for our project. Has a large breadth of services. Strong user community and precedence.			returned without really being able to see what they are doing to your information. This makes it hard to debug.

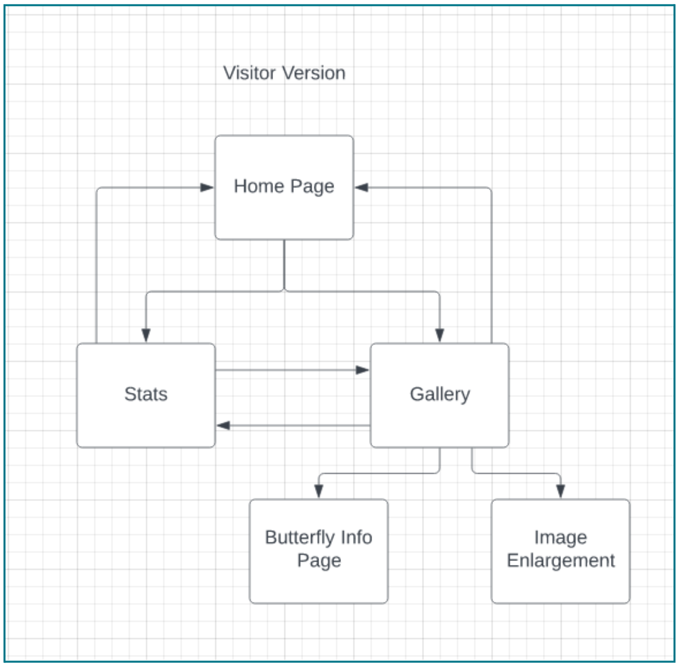
As stated above, the team chose to use Amazon Web Services as it is considered to be the most efficient and secure web service. Through testing and research AWS seemed to fit best with this project. It seemed the easiest to implement and the most flexible.

4.3 Proposed Design

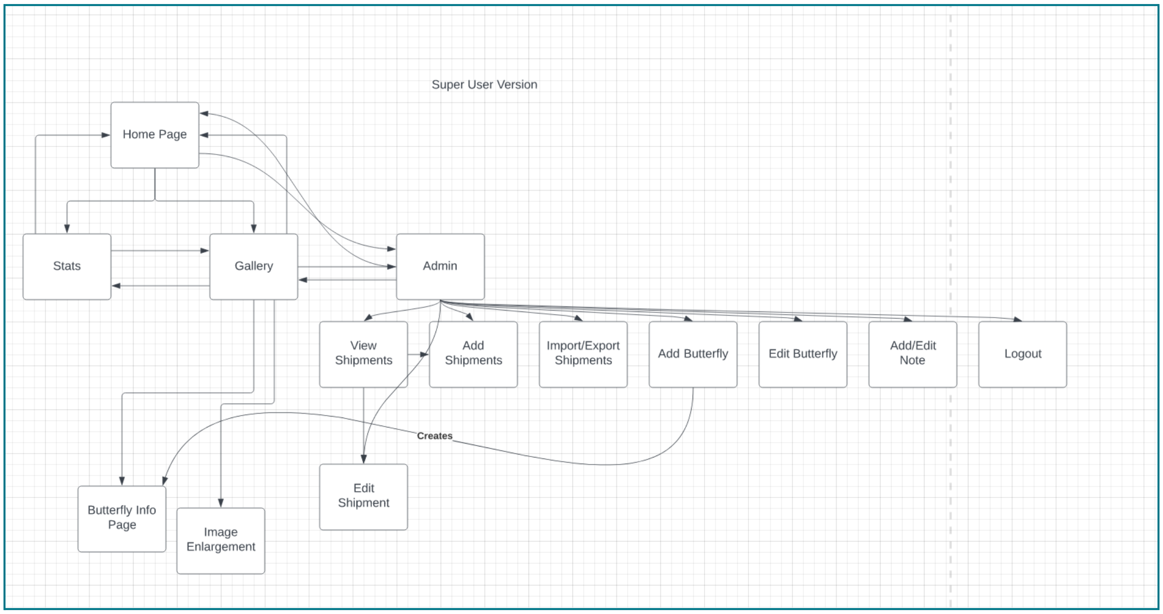
4.3.1 Design Visual and Description



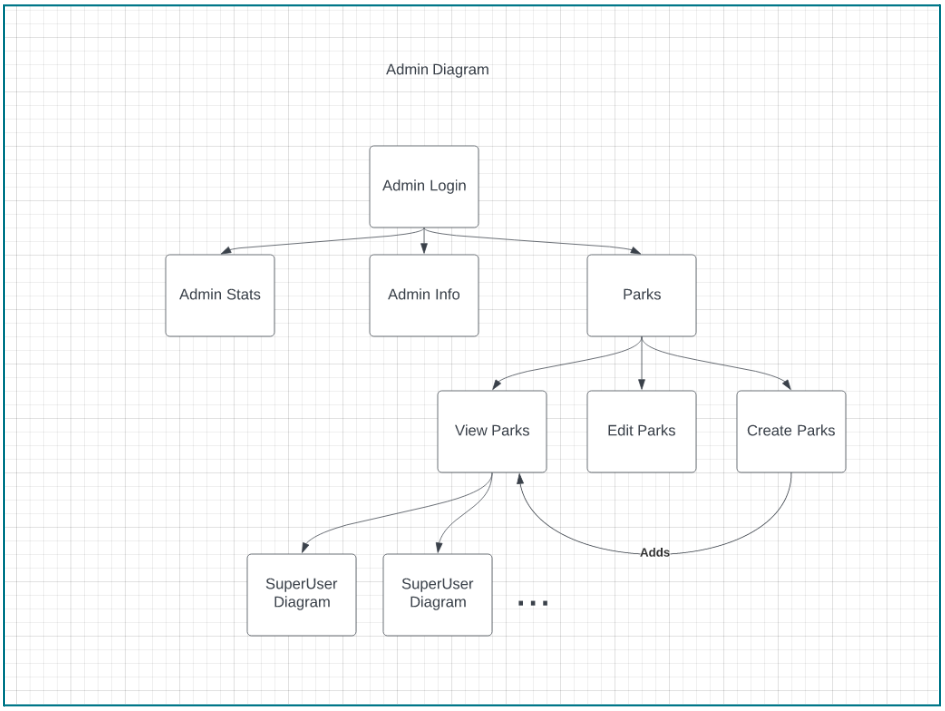
Use case diagram of current design



Screen Diagram for the Visitor Version of the Reimann Web-application



Screen Diagram for the Superuser Version of the Reiman Web application



Screen Diagram for the registered user version of the Reiman Web application

In the user hierarchy of the current design, there are only two levels. The super user can view data, upload and edit data, and customize elements on a page. Visitors can only view data. The current design does not include the "admin" or "registered user" as in the proposed design.

4.3.2 Functionality

The current site is intended for Reiman Garden visitors to be able to view information about butterfly species that are currently in flight. In this regard, the current design satisfies functional requirements. The "butterfly search" is a non-functional requirement that is not met with the current design.

The current site is also intended to be a space where employees can collect information regarding previous and upcoming shipments of butterflies, as well as in-flight species. The current design meets these requirements but does not exceed them. Employees have found several work-arounds in order to use the site as they intended.

4.3.3 Areas of Concern and Development

- Having a fully functioning framework that allows other butterfly parks to create a site that is similar to Reiman Gardens.
- Dealing with issues on AWS that causes loss in time working on other parts of the project.
- In our schedule we are going to start working on the framework before starting on other pages. This allows time to get the framework functioning.
- The research we have done has shown that AWS is the best cloud service for our project.

4.4 Technology Considerations

One example of the technological consideration that we faced is our final design of the search feature to be implemented in a couple places throughout the website. This will be used for our clients to search through butterflies, shipping information, and other collected data. We were at one point considering using a machine learning algorithm to learn the behaviors of the currently logged in account. While this would perhaps provide slightly better ease of use for our product, we decided that our search terms are not diverse enough to make such a time investment. So we decided on simple alphanumeric search.

Another design decision we discussed was how to minimize the size of our database. One of the things our client stressed to us was that they wanted to store a large amount of butterfly related images on the database. Images can be very cumbersome to store so we came up with a solution to cut down on needlessly stored images. Instead of each of the butterfly parks storing their own images of the butterflies we will only allow the main Reiman Gardens account to store those pictures. If another institution wants to use their picture they can send it to Reiman for approval. This will cut down on needlessly saving the same image or similar images over and over and reduce the cost of the database.

4.5 Design Analysis

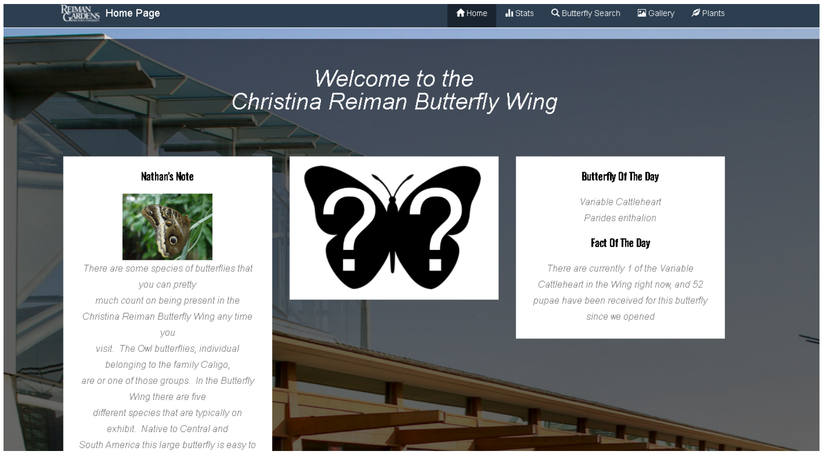
Our proposed design does work because it satisfies the requirements that our client has asked for. The design was made by going through the requirements and goals for this project then split them into individual tasks. Each task can be separated out on the schedule based on the timeline each task should be completed in order to finish on time.

There are some milestones in the design that should be started or completed before others begin. For instance, the admin and global framework are important milestones to have implemented before the home, stat, and gallery pages. While for those last three milestones mentioned can be worked on and completed around the same time.

4.6 Design Plan

Proposed Design Tasks by Module

1. *Cloud Services*
   a. To improve efficiency and reliability, we will be transferring the currently established backend of the web app to the cloud specifically, Amazon Web Services. The current iteration uses a backend called Digital Ocean which functions well but is slightly outdated. Switching to AWS will allow us to have an easy-to-use API to make server calls more fluid and responsive. This service has been determined to be the best by our team members and is the current industry standard when it comes to web services.
2. *Home Page*
   a. This page will be kept mostly the same functionality-wise. The current iteration of the web app as pictured below has a butterfly of the day which is randomly selected from the available butterflies in the park. This functionality will be kept as well as the editable notes section that can display relevant information that a park wants its guests to know. The visuals of this page will be changed significantly and will be more fluid and less blocky. The final design of this page is pending. As this website will be opened up globally this website will also display an interactive globe showing the locations of butterfly parks from around the world that have joined. These will display the organization's website if clicked and will provide the user with a sense of connectedness. Below is an image of the current webpage.



Welcome to the
Christina Reiman Butterfly Wing

Nathan's Note

There are some species of butterflies that you can pretty much count on being present in the Christina Reiman Butterfly Wing any time you visit. The Owl butterflies, individual belonging to the family Caligo, are or one of those groups. In the Butterfly Wing there are five different species that are typically on exhibit. Native to Central and South America this large butterfly is easy to

Butterfly Of The Day

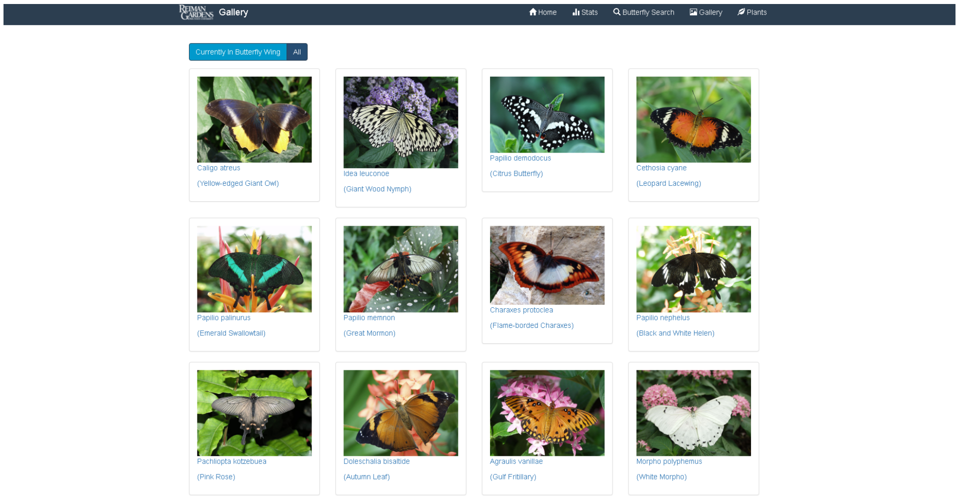
Variable Cattleheart
Parides erithalion

Fact Of The Day

There are currently 1 of the Variable Cattleheart in the Wing right now, and 52 pupae have been received for this butterfly since we opened

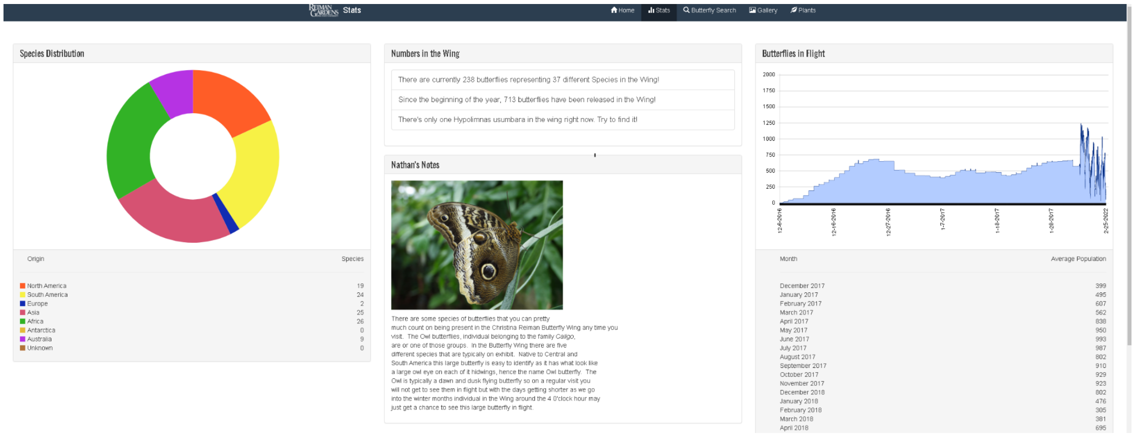
3. *Gallery*
   a. This page's functionality will be kept mostly the same. Currently, a user of the site can view a list of all the butterflies in the organization's database. These pictures are displayed in a grid pattern. If clicked, a page with the butterfly's information will be displayed. This information is given upon the creation of the butterfly in the admin section of the web app. The changes that will be made to this page include the minimization of white space. There is a lot of dead area in the gallery currently. The pictures are blocky and their presentation is uninspired. We propose a full-screen display of rotating butterflies. This will keep the variety of the different butterflies but also make use of all that white space. This will of course have to be mobile friendly and significant effort will be made by the team to make sure that it is appealing on both the desktop and the mobile version of this application. We also propose adding a search function that will appear in a bar at

the top of the picture. This will allow you to search for butterflies that you want to see pictures of. While typing the carousel image display will pause and return to this grid showing you butterflies whose name matches the search result. This will update dynamically as you type. Below is an image of the current webpage.



4. *Stats Page*

a. This page's functionality will be largely changed. Some of the stats that you see represented on the page will remain, but the format of the statistics will be changed to a more visually appealing layout. One thing to note is the constricted nature of the graph on the left on butterflies in flight, this is because of the constrictive nature of the page formatting and will be changed in our project. Another thing that will change about this site is the integration with the statistical tool Flutter. This is a statistical analysis tool that was developed by a senior design team in 2017 and is focused on a broader analysis of the data presented by Reiman Gardens. Statistics are an important part of any advertising campaign. According to the American Statistical Association, statistics are very important to creating rapport with clientele. It provides both legitimacy to the business and portrays professionalism (Roth, 1916). Currently, the statistics section of the website is adequate, but through visual updates and broader more interesting statistics



Origin	Species
🟧 North America	19
🟨 South America	24
🟦 Europe	2
🟥 Asia	25
🟩 Africa	26
🟨 Antarctica	0
🟪 Australia	9
🟫 Unknown	0

There are currently 238 butterflies representing 37 different Species in the Wing!

Since the beginning of the year, 713 butterflies have been released in the Wing!

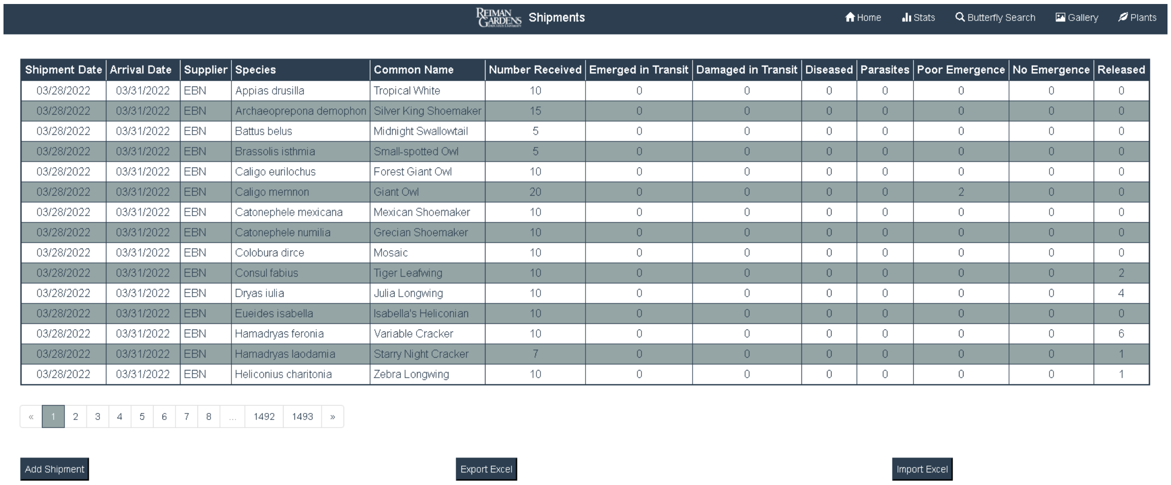
There's only one Hypolimnas usumbara in the wing right now. Try to find it!

There are some species of butterflies that you can pretty much count on being present in the Christina Reiman Butterfly Wing any time you visit. The Owl butterflies, individual belonging to the family *Caligo*, are or one of those groups. In the Butterfly Wing there are five different species that are typically on exhibit. Native to Central and South America this large butterfly is easy to identify as it has what look like a large owl eye on each of it hidwings, hence the name Owl butterfly. The Owl is typically a dawn and dusk flying butterfly so on a regular visit you will not get to see them in flight but with the days getting shorter as we go into the winter months individual in the Wing around the 4 0'clock hour may just get a chance to see this large butterfly in flight.

Month	Average Population
December 2017	399
January 2017	495
February 2017	607
March 2017	562
April 2017	838
May 2017	950
June 2017	993
July 2017	987
August 2017	802
September 2017	910
October 2017	929
November 2017	923
December 2018	802
January 2018	476
February 2018	305
March 2018	381
April 2018	695

provided by Flutter, it will become an even bigger draw to new clients. Below is an image of the current webpage.

5. *Admin Page*
   a. The admin page is where a majority of the mechanical changes will take place. When we expand this web app to allow for organizations around the world to use it, the admin page will have to be the most adaptable to these new organizations. Organizations will use their data which may differ greatly from the data valued by Reiman Gardens. This will be accounted for by a system of defaults. Each new organization will have the same initial admin page tracking the same information. If this is inadequate for their needs the user will be able to enter new information to be tracked and the database will accommodate these needs. As for changes to the current admin page they are as follows. The view shipments page will be updated to be able to display based on shipments instead of individual butterflies. This will reduce the amount of time it will take to scroll through the web app to find the shipment you're looking for. We will also be adding a total releases and emergences section to the shipping data. Mr. Brockman has told us that he commonly miscounts the number of butterflies that have emerged from their cocoon while releasing them and would like to be able to see the total number of that type of butterflies that could emerge visible in the admin tab during data input. We will provide exactly this. We will also be adding a way to edit currently imputed butterfly data from shipments. Currently, there is no way to edit this data once you have submitted it. This has led behind-the-scenes workers at Reiman to have to add in negative shipments to correct data. We will avoid this in our design by adding an edit shipment functionality. This section will also be where a majority of our settings will be to control the specifics of the rest of the website for the admin user. This will allow them to do things such as switch out logos. Change what information is displayed and reorganize their version of the website. Each website will have default settings that you can return to. No changes from one organization to another will affect any other organization's changes. Below is an image of the current webpage.



Shipment Date	Arrival Date	Supplier	Species	Common Name	Number Received	Emerged in Transit	Damaged in Transit	Diseased	Parasites	Poor Emergence	No Emergence	Released
03/28/2022	03/31/2022	EBN	Appias drusilla	Tropical White	10	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Archaeoprepona demophon	Silver King Shoemaker	15	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Battus belus	Midnight Swallowtail	5	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Brassolis isthmia	Small-spotted Owl	5	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Caligo eurilochus	Forest Giant Owl	10	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Caligo memnon	Giant Owl	20	0	0	0	0	2	0	0
03/28/2022	03/31/2022	EBN	Catonephele mexicana	Mexican Shoemaker	10	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Catonephele numilia	Grecian Shoemaker	10	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Colobura dirce	Mosaic	10	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Consul fabius	Tiger Leafwing	10	0	0	0	0	0	0	2
03/28/2022	03/31/2022	EBN	Dryas iulia	Julia Longwing	10	0	0	0	0	0	0	4
03/28/2022	03/31/2022	EBN	Eueides isabella	Isabella's Heliconian	10	0	0	0	0	0	0	0
03/28/2022	03/31/2022	EBN	Hamadryas feronia	Variable Cracker	10	0	0	0	0	0	0	6
03/28/2022	03/31/2022	EBN	Hamadryas laodamia	Starry Night Cracker	7	0	0	0	0	0	0	1
03/28/2022	03/31/2022	EBN	Heliconius charitonia	Zebra Longwing	10	0	0	0	0	0	0	1

6. *Global Park Framework*
    a. The final area of change to Reiman Gardens' web app will be the expansion of its backend to accommodate opening it up to global use. This will involve creating a user hierarchy where there will be three different kinds of users. The first is a visitor to the site. This type of user will be unable to see the admin tab and will only be able to view the website. They can view websites of any organization and as per the Home Page section will be encouraged to do so on the globe feature of the page. The second type of user will be the administrator. This type of user will be the leader of an organization. They will have the power to edit their website and retrieve their information. To access this ability they will have to log in using a username and password that they set. For an organization to gain access to this site they will need to contact the final user, the super-user. This is the level of access granted by Reiman Gardens. They will be able to create new blank template organization webpages to be able to hand out to other organizations and will be able to view and retrieve the data that they post on the site. They will also be able to hide or remove users if their accounts go dormant. They will also have a username and password to log in to and will have access to the special features described above under their version of the admin tab.

4.7 Design Evolution

In this section, we will discuss how the functional and non-functional design of the project has changed over the course of the implementation and testing phases.

1. *Functional design*
    1. *Organization accounts*
        1. In our initial design of the database, each organization would have had several possible accounts linked to it. We initially planned to have organization objects being separate from individual accounts, such that a single organization could be linked to multiple accounts. Due to the difficulty faced by working aws cognito user pools, the team had to take a different approach. In out final design, each organization will only be linked to single user. The organization is automatically created when a user first signs in after creating an account.
    2. *Stats Page*
        1. Our initial design for the web application included a stats page displaying several stats from a specific organization. Unfortunately, due to time constraints this work could not be completed in time. Our design has had to change to completely remove this page from the web application.
    3. *List Attributes*
        1. When designing our initial schema (graphql database structure) we had intended for some objects to have list attributes in a one-many or many-many fashion; for example, organizations can have many orders, an order can have many order items, etc. When we started working with graphql, we quickly realized that adding list attributes to objects through graphql mutations was not an easy task. Keeping our time constraint in mind, and after reaching out to needed experts for help, we opted to

change our approach due to a lack of a positive response. We moved to a new design with each object having a "marker" to link the object to the needed object. For instance, orders belonging to a specific organization have an "orgID" field with the id of that organization. Similar fields were added across the schema.

2. *Non-functional Design*
    1. *Customization*
        1. In our initial design, we planned a wide array of customization options for each organization to truly make their page their own: adding font customization, background color, and more. Due to time constraints, and after discussing with our client, we opted to considerable cut down on our customization plans. We modified our design such that customization options will be more limited, with more aspects of the app being hard coded across all organization pages.

4.8 Our Design in Comparison

In this section we will talk about how our implementation addresses the problems our client faces and how well it does so in comparison to other similar applications.

Our application is essentially an inventory management system combined with a frontend website for butterfly parks. There are similar products that do one of those things, but they have never been combined together for the express purpose of managing a butterfly park. Because of this unison of products we are able to efficiently manage inventory and then display relevant information to potential customers. This creates a one stop shop for butterfly parks and allows us to mold the application to fit the specific needs of a butterfly park. Through the design process we have tried to tailor our website to the specific use cases of a butterfly park manager as outlined by our client. This again gives us the edge over our competitors, because of our local proximity we have been able to cut out unnecessary fluff that can often come with an over the counter application. This allows us to cut costs and allow our clients to only pay for what they need. During the design process we were also able to reference an old version of the application and see what areas worked and what areas didn't. This also gives our product an edge over the more general competitors as we were able to see first hand what areas need the most attention and again, mold the application around these problem areas to better fit our clients needs. These factors together combine to make our application stand out in a competitive market.

5 Testing

5.1 Unit Testing

1. Forms
    a. Mockito, Selenium

    b. Using Mockito (or similar) to automate a browser for user-input tests.
2. Database requests
    a. Postman
    b. Checking post/get requests to and from the server for accuracy and speed.
3. Links
    a. QUnit/JUnit
    b. Write QUnit/JUnit tests to ensure that the websites Context (which page the website is on) is correctly changed in all instances that we expect
4. CSS standards compliance
    a. Uploading existing CSS files to a third party system to check compatibility with browsers
5. Exporting shipment data
    a. Ensure that the data being exported is compliant with Excel and contains all the necessary data for a USDA report. This can be done through tests to ensure that the data object has no null fields.

5.2 Interface Testing

1. APIs
2. AWS (Amazon Web Services)
3. Receiving data from the server

For API and database testing, we will use tools such as REST-Assured, Postman, or JQuery to test the sharing of data between different areas of the project and between the frontend and backend.

For AWS testing, we will use Amazon Inspector for network accessibility and the security state of our application.

5.3 Integration Testing

1. Creating a new user
    a. Creating a new user should append their information to the database and also create a new webpage with their bare essentials for them to edit. This is another functional requirement for our project.
2. Creating a new organization
    a. Similar to the above (Issue 1)
3. Organization-based queries
    a. Our webpage will need organization data to load specific aspects of the webpage per organization. To do this we will query the database reactively. To ensure the functionality of this feature we need to make sure that organization data is passable between different application contexts.
4. Populating butterfly species data
    a. This is the main data point for the butterfly section of the database and the website. The website will reactively load butterfly information upon request. It is critical to ensure that this data loads properly.
5. When an organization logs in, the correct page is loaded
    a. One of the requirements for our app is to expand it for use with multiple organizations and not just Reiman Gardens. It follows that testing this feature would be critical.

For each of these scenarios, we will be using a Mocking API such as Mockito to create faux objects to insert into the receiving interface. Using this we can then test that the webpage loads correctly without using real data.

5.4 System Testing

1. Usability
    a. Test different aspects of the project with real users
2. Cross-platform compatibility
    a. Compatibility-testing tools such as BrowserStack or XBOSoft
3. Unit tests of input forms and database requests
    a. Specified above
4. Interface tests of receiving data from the server
    a. Specified above
5. Integration tests of populating butterfly data and correct organization login
    a. Specified above

To ensure that our system functions correctly we have identified three main areas to assess our choices of tests. Those are performance, load, and stress testing. These will ensure that the website runs quickly and efficiently (such as loading the gallery in a timely manner), can handle extreme loads (such as overly complex, or large queries), and ensure it can handle the expected number of users as described in the design document. This can be done with tools such as JMeter.

5.5 Regression Testing

One tool we will use for regression testing would be Git, which would allow our team to test the integration of new features with previous features in a local environment before adding them to the shared/production repository.

We will also use other tools for frontend regression testing such as Selenium or IBM Rational Functional tester for a software-based approach to regression testing. For the backend, we will use Avo Assure, Telerik Test Studio, or other similar tools.

5.6 Acceptance Testing

We will create several prototypes for our different users and allow them to give feedback on the main features they will use in our application. This will allow us to make changes based on real market feedback. These features will be selected by interviewing customers. We will also be in contact with Nathan and show him all prototypes to make sure that all of his requirements are met.

5.7 Security Testing (if applicable)

It is important that we keep the information and data of the butterfly organizations secure. To do this, we can use tools such as Security AppScan or SQLMap to test for vulnerabilities in our application.

We can also use “fuzzing” tools such as OWASP WSFuzzer to test several different inputs to find easily hackable software bugs within the application.

5.8 Results

We do not have any results at the time of this assignment. This is because we are attempting to remain flexible in our design ideology and therefore will go with a "test-as-we-go" approach. Below are some of the tests conducted during the implementation of the project

1. User cookies
    1. Following our current design, visitor-side pages (homepage, gallery, parks page) have a special url containing the url of a specific organization. This organization url is used to load organization specific information for that page. On the administrative side however, the user has to signin to access administrative functionalities. Upon signing in, the organization ID is kept as a cookie in localstorage for easy retrieval at any point in the app as this information is paramount for most functionalities. Visitor side pages, however, use a dummy account to enable access to aws storage for retrieval of images. We rigorously tested the app to confirm that all visitor side pages were blocked off administrative pages by navigating the app in several ways.
        1. We found a bug whereby if an administrator moved to a visitor side page, upon returning to the administrative side, the web application would mistakenly consider the dummy account to be signed in a display dummy information. A page reload would be needed to re-actualize the page and for the app to use the proper organization account.
2. Map API
    1. Our map API uses location information (address, city, state, etc) to obtain longitudes and latitudes and display the organization on the map in the parks page. We tested the efficiency of this api by using several different addresses from different places in the US and overseas.
        1. We discovered that the api was not very effective at mapping addresses outside of the us. Several addresses in France, for instance, could not be mapped (could not get longitudes and latitudes). More so, so addresses in the us could not be mapped.
        2. The best fix for this would be to use a more efficient api. However, most of the good apis found by the team are not free to use. Due to time constraints, we decided to remain with the current api as we did not have the time to get the funds through at client.

6 Implementation

Our plan for implementation next semester is to follow our Gantt chart from section 3.4. The first aspect we are focusing on is working on setting up the cloud service AWS. The focus is to have this completed between the end of this semester and start of the Fall semester. After that the two components to implement at the start of next semester is the admin's page and global framework. Next, we will build the home, stats, and gallery page. Throughout this whole process we will be creating and running tests to ensure all pieces work individually and as a whole.

6.1 Implementation Details

Below is a list of all the major functionalities of our web-app as well as some words on how they are implemented.

- Global Framework
    - The website has visitors, organizations/users, super users (Reiman Garden). The functionalities/page are separated into visitor side pages (pages that can be accessed by visitors) and administrative pages (pages that can only be accessed by users). Some specific functionalities are reserved to Reiman Garden, such as suspending and/or deleting existing organizations.
- Sign In
    - The sign in functionality is done using the AWS Cognito service. We essentially encapsulate the admin panel within a withAuthentication module meaning that you can't view the admin panel without being an authenticated user. The signin state is also kept track of using a useState react hook. This way we can keep track of when someone is signed in and change the page accordingly. This is used to hide the admin panel functionality for instance or to change the sign in button to a sign out button. Whenever we need to know the currently authenticated user we are able to query the Cognito service to retrieve the username. We use this to assist us with some user centric database queries.
- Admin Functionality
    - Add/Edit Butterfly
        - Whenever the Edit Butterfly page is loaded, the database is queried for all existing butterflies in an asynchronous useEffect. The butterflies obtained can be selected in a dropdown option bar. Once a butterfly is selected, the butterfly properties are filed in the appropriate fields. These fields can then be edited as needed. The edited fields can be saved in order to edit the butterfly information. The changes can also be canceled using the cancel button. Lastly, the butterfly can be deleted using the delete button. Once any of these options are selected, a pop up confirmation appears prompting the user to confirm the required action or to discontinue and negate the action selected. When adding butterflies however, the fields are left empty, to be filled by the user. Once again, the changes can be saved or discarded as needed. On the Add butterfly page however, the database is not queried for currently existing butterflies.
    - Display/Edit Shipment
        - Whenever this page is loaded an asynchronous function is run using a useEffect hook. In this function the database is queried for all Shipment objects that contain the Organization ID of the currently signed in user. This ID is found by fencing the actively signed on user and querying the database for a matching username. Then all matching shipments are loaded into a table along with a button to view more which contains the ID of the shipment in the corresponding row. When this button is clicked the page navigates to the packingList page where the shipment ID is passed through the url as a search parameter. Once here a similar process occurs for loading the order items. An edit button is loaded here instead of a view more button. This button is tied to a conditionally rendered form where the user can fill out updates to the order item and upon a press of the submit button the changes will be sent to the database by way of a GraphQL mutation.
    - Add Shipment

    - Every time this page is loaded, a useEffect runs that retrieves all order objects from the currently signed in organization using the organization ID. These orders are displayed in a tabulated manner similar to the display shipment page. For every order, an add order item button is available to add order items to the corresponding order. More so, an add order button is placed at the top of the page enabling orders to be appended to the current organization. Upon every order addition, the page is reloaded to show the updated order list.
  - Edit Organization
    - When this page is loaded, a useEffect is used to get the organization attributes which are loaded into the appropriate fields. The fields can then be modified as needed to change organization attributes: logo, covermedia, section header color, header color, organization name, menu color, location, url, etc. The changes can be saved by clicking the save changes button at the bottom of the page. The locations information entered, is used to compute the organization latitude and longitude. These parameters will be added to the organization object and used to display the organization on the parks page. The page can be reloaded manually to see made changes.
  - Customization Page
    - This page contains a form where users can update their customization options for an organization. These form fields are filled by again finding the currently active user that is saved inside of local storage. Then we filter a listOrganizations query to find the correct user and fill the form out according to the data associated with them. These variables are each kept inside of a useState hook that are updated dynamically whenever new data is entered. A react-color-palette api is used to create a color selection tool for all color inputs. All image inputs have their own handler functions that associate the current data with the url for a picture then store the picture on a separate database to save space. When they are loaded the same process happens but in reverse. Location can also be set on this page which automatically calls another API to geocode the String location into a geoJSON object form which we can read the latitude and longitude. These fields are then separately stored in the return useState data. When the submit button is clicked a graphql mutation is called and given the collective return useState data, this updates the data on the backend and reloads the page so the user can see the updates live.
  - Customize Modules
    - Whenever this page is loaded and asynchronous useEffect hook is used to obtain all organization modules using the organization ID of the currently signed in organization. The desired module can be selected from the dropdown menu at the top of the page. The title, content, and image of the module can then be modified as needed. The changes made can be saved, or canceled, leading to a page reload. The module can also be deleted by using the delete button available on the page. On the second half of the page, an add module section is available to add additional sections. A module title, content, and image can be added to the new module, which

        - will update the internal new module object. These changes can be saved or discarded by clicking the save or cancel buttons respectively. Newly created modules will have the organization ID added to them in order to keep track of the organization owning the module.
    - Import/Export Shipments
        - Whenever this page is loaded, a useEffect hook is used to get all organization shipments using the ID of the current organization. These shipments are displayed in a tabulated manner with each row/shipment having an export shipment option. Once this button is pressed, the a pop up is displayed: a csv file can be exported by clicking the export csv link, or the export action can be voided by clicking the cancel button. The page also has import shipment button at the bottom of the page. This button can be used to import shipments. Once clicked a csv file can be selected from the current device. The selected csv will be converted into order, and underlying order items, all of which will have the organization ID added to them.
    - Manage Organization
        - This page is only available to the super user, Reiman Garden. A list of all current organizations is displayed in a tabulated manner with the current of the organization shown: active, suspended, deleted. A button is available for each of the 3 possible actions; the "activate" button can be used to convert a suspended organization back into an active organization. The "suspend" button can be used to suspend an active organization: suspended organizations can not be accessed by their owners, and the url cannot be seen/accessed by visitors. The "delete" button can be used to delete organizations from the system. More precisely, these organizations will be marked as deleted and kept in the system as such. Deleted organizations cannot be reactivated, unlike suspended organizations. Similarly to suspended organizations, however, deleted organizations are unattainable by both administrators and visitors.
- Parks Around the World
    - This page makes use of an API called react leaflet. This is an API that renders a map based on the larger OpenStreetMap API. This page also queries a list of all organizations using GraphQL and then filters out all of the entries with a latitude and longitude. From here it makes new Marker components that are dynamically loaded onto the map using the coordinates found from the query. These markers are also outfitted with an onclick function that navigates to the specified organizations url. This information is passed inside of the Marker component to the onclick function.
- Gallery
    - This page will retrieve all butterflies currently available in the database. The images associate with each butterfly is displayed with the scientific or common name below, according to the user's choice using the buttons at the bottom of the page. Each butterfly image can be clicked to load the butterfly specific information/details.
- Homepage

    - When the homepage is loaded, the active modules currently associated to the organization are loaded in a useEffect hook using the organization url from the current full url. The modules are displayed on the page from the top down, one after another.
    - Butterfly of the day
        - This is a functionality randomly selects a butterfly from that organization's list of active butterflies and makes it the butterfly of the day. This is a module that loads on the organization's home page. To implement this, the team queried the database for all order items that currently have butterflies in flight that are a part of the current organization's orders. From here, a butterfly is randomly selected and the relevant information is pulled from the database to create a component to be displayed on the home page.

- Url Management
    - The url for all visitor side pages (homepage, gallery, parks page, etc) contain an organization specific url. This url is used to identify the organization to be used to display information on each respective page. These urls can be added in the customization page for each organization. Routes are created for each organization url obtained from the full list of organizations currently in the database. For administrative pages (edit/add butterflies, edit/add shipments, import/export shipments, customizae organization/module, etc) there is no organization url contained in the url.

6.2 Known Bugs

As we finish development of this project we think that it is imperative to inform our potential users about some possible bugs and known errors they could encounter and what to do if they occur.

- Shipments or other objects not loading correctly
    - This can be caused by an overload to our server. To remedy this you may have to delete some old data to be able to fully see the new information entered.
- Wrong shipments displayed or unfamiliar admin panel
    - This is caused by a mix up with the active user in the admin panel. What you are looking at is the admin panel of our dummy account. The remedy to this situation would be to sign out and then sign back in to reset your user identity.
- Pin not showing up on map page
    - This is caused by an error with the API that geocodes your address information into readable latitude and longitude coordinates. This often occurs when the customize page is double loaded, either by another account or by double clicking the submit button. To remedy this you may have to re-enter your information on the customize page. Then your pin should show up on the map. Another possible cause could be a browser incompatibility, you could also try using a different browser.
- General loading error
    - Sometimes the page will display unexpected results, a good first step if you see these is to log out of your account and then sign back in and renavigate to the page

that you were trying to view. A lot of the time this will solve the issues you are facing.

6.1 Areas of Potential Improvement

As we underwent development of this project we were unable to fully realize the design for our web application. During the time crunch we focussed instead on broad functionality and a few of the finer details went to the wayside. Below is a list of improvements that we believe will benefit the application that we were not able to achieve. We have also listed some potential ideas for developing these ideas.

- Search
  - Currently there is no functionality for a search function for both the shipments page and the gallery page. This means that it can be quite difficult to find a specific object given a large enough table size. We recommend adding a search bar that can dynamically change the loaded table by filtering results by the string the user types. To make it easier with the shipments you could create a calendar user input that will reload the page with only the shipments from the imputed date or range of data. With the gallery it will probably be best to only sort by matching butterfly names, although it will be important to also search by the organization specific common name.
- Design
  - During our development process, functionality came before design so some of the pages do not look aesthetically pleasing. This can be improved by CSS pages. Each of the components that we loaded should have a unique ID that you can use CSS to format. Specifically the admin pages need the most work in this regard.
- Database Concerns
  - We ran into a lot of problems with the DynamoDB database. One of those was our inability to figure out how to create an object with a list of other objects inside and push that to the table. Ideally this relational design would be the most efficient design for our use case, but we were unable to get it to work. Changing the database to this design might also fix some of the later issues if successful.
  - In order to fix the above problems we needed to instead associate specific IDs with each object to keep track of which item to associate them with. So if an organization has multiple orders which could have multiple order items. Then we would need an organization ID and an order ID respectively to accurately keep track of data. This implementation was successful, but does query the database a lot more than needed. There is also a limit to how many objects can be returned with a query meaning that if there were a lot of entries not all of them would be able to be returned with the queries as is. We were able to remedy this with the order items specifically by implementing pagination, but this would also have to be done with all other query-able objects should the table size get too large.
- Additional Functionality
  - There are a couple functional components that we were unable to finish that we think would greatly improve the quality of the project those are as follows
    - Stats Page

- This page was present on the previous team's project. It essentially compiles a lot of the information found in the shipment data into readable graphs, charts, etc. We were able to implement an API to create these graphs, but were not able to do so with live data. To create this page you will have to query the database in a similar manner to other pages and parse the relevant data out. Then you can use the present APIs to load the charts onto the page.

7 Professionalism

7.1 Areas of Responsibility

Area of Responsibility	**IEEE Code of Ethics numbers**	**Difference from NSPE**
Work Competence	Point 6 of the IEEE code of ethics follows a similar approach with respect to work competence. Both codes of ethics state that work needs to be completed with the highest technical skills possible and only within one's speciality.	Little to no difference between the two standards. Both state that technical work should only be provided in areas where one has experience or qualification.
Financial Responsibility	Point 3 of the IEEE code of ethics mentions that conflicts of interest need to be properly avoided and reported whenever they occur.	NSPE states that engineer's should make statements in an objective manner. IEEE makes no such specification
Communication Honesty	Points 2 And 5 of the IEEE code of ethics make clear that it is paramount to improve understanding between individuals and to seek and report truthfully the capabilities of emerging technologies.	Little to no difference between the two standards. Both state that conflicts of interested should be fully disclosed and avoided

Health, Safety Well-Being	Points 6 and 7 ask us to be constantly vigilant in our pursuit of new technologies and to seek out criticism on our current work, and also to not undertake jobs outside of our competence levels. When we take a job we are unsure of how to finish it is easy to make mistakes. Some of those mistakes could lead to failure of our product. In extreme cases, like on an airplane, this can lead to safety concerns.	IEEE states that engineers should strive to improve technical skills by utilizing others, professional development, etc.. NSPE does not discuss how engineers need to improve their skills.
Property Ownership	Point 2) Although the team does not foresee any conflicts of interest, they need to be aware of it and be willing to share that with the stakeholders/clients	Little to no difference between the two standards. Both state that conflicts of interested should be fully disclosed and avoided
Sustainability	Point 5 of the IEEE code of ethics states to understand the application of new technologies and potential consequences that may occur.	IEEE states that engineers should strive to improve technical skills by utilizing others, professional development, etc.. NSPE does not discuss how engineers need to improve their skills.
Social Responsibility	1.) The IEE first code of ethics to be aware of how our decisions as engineers affect the health and wellness of the general public, and to act immediately if we sense a breach of this code. This demonstrates care for the environment in which we are creating our design projects and demonstrates great Social Responsibility.	Little to no difference between the two standards. Both state that the safety, health, and welfare of the public is of the utmost importance.

7.2 Project Specific Professional Responsibility Areas

Area of Responsibility	**Competence Level and Relevance**
Work Competence	Relevance: High Since our project consists of improving an existing web application created by previous senior design teams, there is a need to deliver a more robust and technologically up-to-date product. Team performance: Medium The team has not had much experience with such projects, but our basis in web development appears to be a stable enough foundation for growth.
Financial Responsibility	Relevance - Low We will not be handling any resources that require money during our project and our client has not given us a budget. Team Performance- Low
Communication Honesty	Relevance: High The end product of this project is to be used by our clients and several other institutions. As such, there is a need to honestly and truthfully report our accomplishments in order to ease their usage of the product Team Performance - High The team has already shown great signs of fluid communication with our client and intends to uphold these standards throughout the entire project
Health , Safety, Well Being	Relevance: Low Our project is not of high risk of harming people if implemented incorrectly. The worst effects of our app implemented poorly are poor customer interactions. This could potentially hurt business. Team Performance - High
Property Ownership	Relevance: Medium We do have access to data that Reiman owns in the form of their butterfly emergence data. We have to respect our bounds of this information and only transfer it to and from locations in the web app and not modify it or offload it elsewhere. Team Competence: High
Sustainability	Relevance: High Our project will be used by many users around the world that are maintaining their own site for their butterfly garden. It is important that this stays suited for both staff and guests for years to come.

	Team Performance -Medium
Social Responsibility	Relevance - High This project will directly benefit employees and visitors to Reiman Gardens, as well as other facilities with butterfly gardens Team Performance - Medium The team has worked directly with client to determine what features will be the most beneficial and effective

7.3 Most Applicable Professional Responsibility Area

We think that Work Competence is incredibly important to our project. Our client has stressed that they want to be able to use this website as a landing page for people around the world who are interested in butterflies, and to expand the knowledge of butterfly pavilions. This requires us to be on the top of our game when it comes to handling these important requests to a satisfactory level to our client. The client has also stressed the importance of a couple of necessary features that we must have. The biggest hurdle to these features is our own competence level in their implementation.

8 Closing Material

8.1 Discussion

As we move into the next phase of our schedule we will move more heavily into implementation. So far we have been mostly designing the prototype and finalizing design decisions. We have carefully made these decisions in accordance with our clients suggested requirements. We have also planned extensively for risk mitigation to make the transition into implementation as smooth as possible. We have discussed our proposed plan with our client and he has approved it for implementation.

8.2 Conclusion

This concludes the Design Document for the revamping of Reiman Gardens' web app. The goal of this project is to update the website visually, improve the data input experience, and expand the web app to reach a wider audience. To accomplish this goal, each of the website's main pages will be systematically updated and improved, and the current backend will be uprooted and moved to the cloud to account for the new user hierarchy changes. This project will be done over the course of a year by volunteers from the senior design team number five. So far the team has been able to make use of the risk matrix to avoid any catastrophic hurdles. In the future I think that this project could have benefitted from an earlier development of the prototype. This would allow us a clearer picture of what we want the project to look like early on in the design process. This also would have eased a lot of the decisions we made while our design was still foggy. If you have any questions or comments about the project as laid out in this document please contact the members of the team at sdecc22-05@iastate.edu. We thank you for your time and consideration.

8.3 References

CPWG/2301_WG - Cloud Profiles WG (CPWG) Working Group. (2006, June). IEEE SA - IEEE Guide for Cloud Portability and Interoperability Profiles (CPIP). SA Main Site. Retrieved April 24, 2022,

from https://standards.ieee.org/ieee/2301/5077/

Roth, H. L. (1916, December). The application of statistics to advertising ... - jstor.org. Publications of the American Statistical Association Vol. 15, No. 116 . Retrieved April 11, 2022, from https://www.jstor.org/stable/pdfplus/2965767

Technical Committee : ISO/IEC JTC 1/SC 7. (2015, May 21). Software Engineering — Recommended Practice for the Internet — Web Site Engineering, Web Site Management, and Web Site Life Cycle. ISO. Retrieved April 24, 2022, from https://www.iso.org/standard/41294.html

Technical Committee : ISO/IEC JTC 1/SC 7. (2020, October 5). Systems and software engineering — Engineering and management of websites for systems, software, and services information. ISO. Retrieved April 24, 2022, from https://www.iso.org/standard/62440.html

8.4 Appendices

8.4.1 Team Contract

Team Name Team #5

Team Members:

1) Caleb Donavon___________________ 2) Samuel Sells_____________________

3) Johann Guepjop Megaptche________ 4) Bailey Wanders___________________

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
   a. Meet with team weekly Sunday at 4 pm
   b. Client meetings: Bi-weekly, subject to change. Have constant email communication with the client, if additional information is needed
   c. Advisor meetings: Bi-weekly, normally Thursday afternoon. Can be scheduled on demand if needed
2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):

    a. Discord (between team members)
    b. Email (with advisor and client)
3. Decision-making policy (e.g., consensus, majority vote):
    a. Consensus
    b. In the cases of split decisions, seek advice from faculty advisor or client, then decide again
4. Procedures for record-keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
    a. Bailey will take minutes each meeting and share them with the team Google Drive in the appropriate document

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    a. Notify if you are unable to attend a meeting at least 24 hours in advance
    b. If you miss a meeting, it is your responsibility to find out what you missed.
    c. Punctuality is expected unless previous notice has been given.
    d. You're expected to participate in all discussions in the meeting and put forth your best effort.
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    a. Adhere to team set time management goals
    b. Divide work evenly among team members
    c. Participation is mandatory in team assignments
    d. If unable to fulfill responsibilities, let other team members know
3. Expected level of communication with other team members:
    a. Attend weekly team meetings for progress updates
    b. Reach out to other team members with questions or concerns immediately

4. Expected level of commitment to team decisions and tasks:
    a. Put in your best effort
    b. Aim for five hours of work per week
        i. This number is subject to change throughout the semester

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
    a. Team organization: Bailey
    b. Client interaction: Bailey
    c. Individual Component Design: Every member is expected to participate in component creation and design

    d. Testing: Every member is expected to participate in testing
2. Strategies for supporting and guiding the work of all team members:
    a. Adhere to scrum meeting guidelines where we share our experiences over the past week and reach out for guidance if we need to
    b. Practice open communication in the Discord server if anyone needs help with their tasks
3. Strategies for recognizing the contributions of all team members:
    a. Accurately discuss and reflect on the past week's work in the reports.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
    a. Sam: I've worked on several long-term projects as a Front-End developer where I was responsible for creating web and android applications. I am most familiar with Java and Javascript, although I have used other languages frequently. I enjoy CSS and HTML design.
    b. Caleb: Worked on web development for both front-end and backend aspects for admissions. Main languages are javascript, html, CSS, and php. Have experience with C and Java from courses.
    c. Bailey: Significant experience in front-end and UI development. Most familiar with Java, React, C++, C, and HTML.
    d. Johann: Worked on a long term backend project in network security using the C/C++. Got certification for knowledge of HTML5, CSS, and JavaScript but don't have much experience using these. Little experience with React and php. Significant course work and personal experience with Java and python.
2. Strategies for encouraging and support contributions and ideas from all team members:
    a. All team members will be willing to help each other when one member requests it
    b. Acknowledging good ideas proposed by a team member
3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)
    a. Discuss it at the weekly team meeting
    b. Send out a message in the Discord server
    c. Reach out to faculty advisor/professor for assistance in improving the environment

Goal-Setting, Planning, and Execution

1. Team goals for this semester:
    a. Pick a design to move forward with for the butterfly app

    b. Focus on small deliverables to perfect the existing project before moving on to mass deployment for global clients.
    c. Have clear and detailed "schematics" for each of the components we are going to work on and communicate these with our advisor.
2. Strategies for planning and assigning individual and team work:
    a. At the weekly team meeting, a list of tasks will be assigned to each team member and designated in that week's report
    b. Team members can ask other members for more information on what they can do to help further the project
3. Strategies for keeping on task:
    a. Constant communication with team members
    b. Address difficulties early and often
    c. Being prepared for team meetings i.e., group deadlines.
    d. Eliminate distractions when they appear

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?
    a. Depending on the severity, first we will discuss with the offender to see if we can work it out internally
2. What will your team do if the infractions continue?
    a. Reach out to faculty advisor or professor for assistance

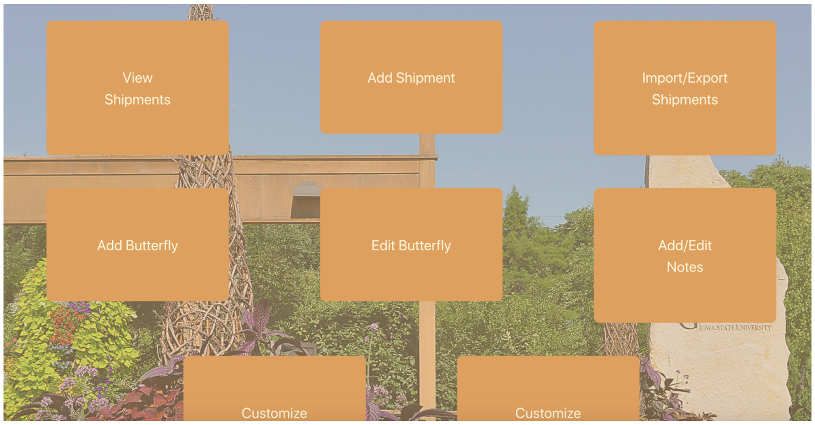
****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*
b) *I understand that I am obligated to abide by these terms and conditions.*
c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*
1) _Sam Sells____________________________________ DATE ___2/13/2022______
2) Bailey Wanders_________________________________ DATE _2/13/22__________
3) _Caleb Donavon________________________________ DATE 2/13/2022_________
4) _Johann Guepjop Megaptche____________________ DATE 2/13/2022_________

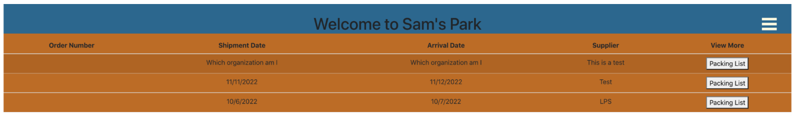
8.4.2 Operation Manual

- What is our project?
    - Our project is a web based application that is designed to act as both a storage service for butterfly parks around the world to manage their inventory as well as a website for them to display information about their park. This application will take two forms, one that a guest user will be able to interact with and another that administrators (park owners) will be able to use.
- Using our project
    - Admin

- Accessing
    - In order to access the admin side of our application you will need to navigate to the url www.flutr.org/signin. Here you will be able to sign in to your account or sign up for an account if this is the first time you will be accessing the site. The sign up process will require two-factor authentication. Once you've signed in you will be brought to the admin panel screen. Here you will see a list of buttons that will allow you to access the admin functionality.



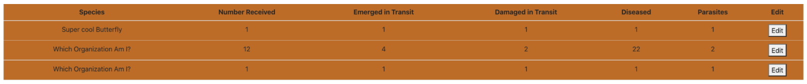
- View Shipments
    - The first functionality of our webpage is the view shipments page. This can be accessed by clicking the button of the same name and will bring you to a page where a list of shipments will be displayed. These shipments can have their butterfly content accessed by clicking on the packing list button.



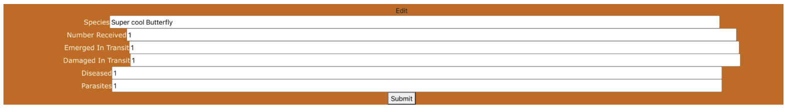
Welcome to Sam's Park

Order Number	Shipment Date	Arrival Date	Supplier	View More
	Which organization am I	Which organization am I	This is a test	Packing List
	11/11/2022	11/12/2022	Test	Packing List
	10/6/2022	10/7/2022	LPS	Packing List

From here you will be able to see the butterfly information from each order as well as an edit button that you can click to change any information about a butterfly.



Species	Number Received	Emerged in Transit	Damaged in Transit	Diseased	Parasites	Edit
Super cool Butterfly	1	1	1	1	1	Edit
Which Organization Am I?	12	4	2	22	2	Edit
Which Organization Am I?	1	1	1	1	1	Edit

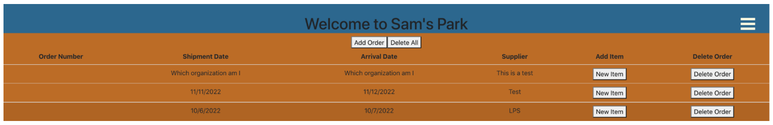


Edit

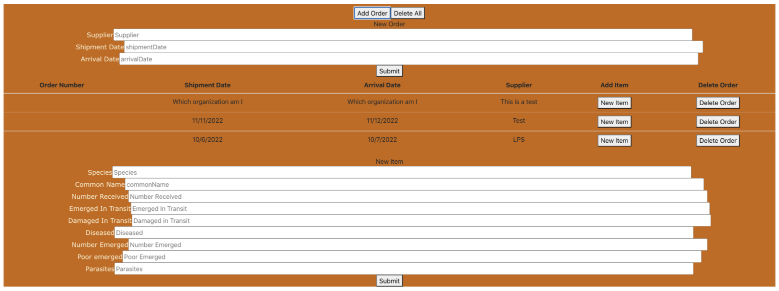
Species Super cool Butterfly
Number Received 1
Emerged In Transit 1
Damaged In Transit 1
Diseased 1
Parasites 1

Submit

- Add Shipments
  - The second functionality of the admin page is the add shipments button. This button will take you to the add shipments page where you will again see a list of your shipments. If you want to add a new shipment you can click the add order button at the top of the screen. If you want to add specific butterfly information to an order (like how many of a certain species arrived) you can click the new item button on the order which you want to edit. And finally if you want to delete an order the delete order button is also available to you.



Order Number	Shipment Date	Arrival Date	Supplier	Add Item	Delete Order
	Which organization am I	Which organization am I	This is a test	New Item	Delete Order
	11/11/2022	11/12/2022	Test	New Item	Delete Order
	10/6/2022	10/7/2022	LPS	New Item	Delete Order

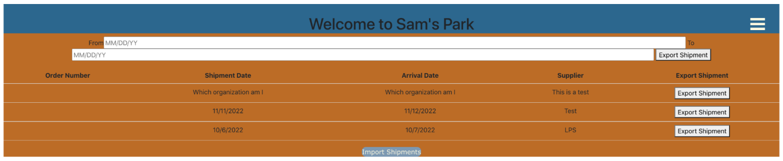


Supplier Supplier
Shipment Date shipmentDate
Arrival Date arrivalDate

Order Number	Shipment Date	Arrival Date	Supplier	Add Item	Delete Order
	Which organization am I	Which organization am I	This is a test	New Item	Delete Order
	11/11/2022	11/12/2022	Test	New Item	Delete Order
	10/6/2022	10/7/2022	LPS	New Item	Delete Order

Species Species
Common Name commonName
Number Received Number Received
Emerged In Transit Emerged In Transit
Damaged In Transit Damaged in Transit
Diseased Diseased
Number Emerged Number Emerged
Poor emerged Poor Emerged
Parasites Parasites

- Import/ Export Shipments
  - This is the 3rd functionality of our admin panel and here you will be able to import or export shipments from your records. The shipments will be exported to a csv file. You can export any specific order or a date range from the text boxes above. The import shipments button at the bottom of the page will allow you to import orders from a csv file, but only 25 at a time due to a limitation of our AWS partners.

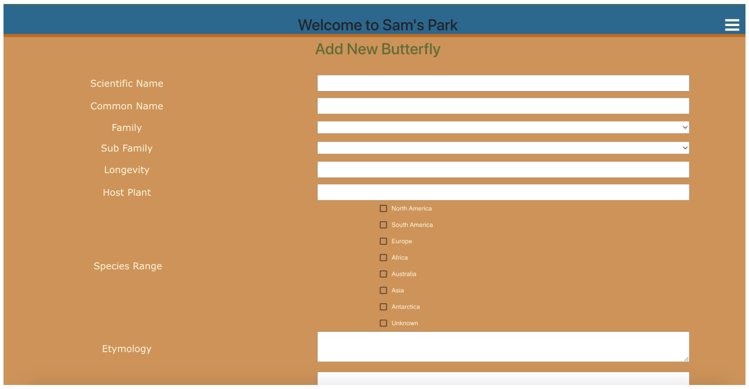


From MM/DD/YY To MM/DD/YY

Order Number	Shipment Date	Arrival Date	Supplier	Export Shipment
	Which organization am I	Which organization am I	This is a test	Export Shipment
	11/11/2022	11/12/2022	Test	Export Shipment
	10/6/2022	10/7/2022	LPS	Export Shipment

- Add Butterfly
  - This is the 4th functionality of our admin panel and allows you to add a butterfly to your park. This butterfly is a way for you to keep track of information about butterflies in your park. You can add pictures and information that will show up in the gallery for visitors to

view.



Welcome to Sam's Park

Add New Butterfly

Scientific Name

Common Name

Family

Sub Family

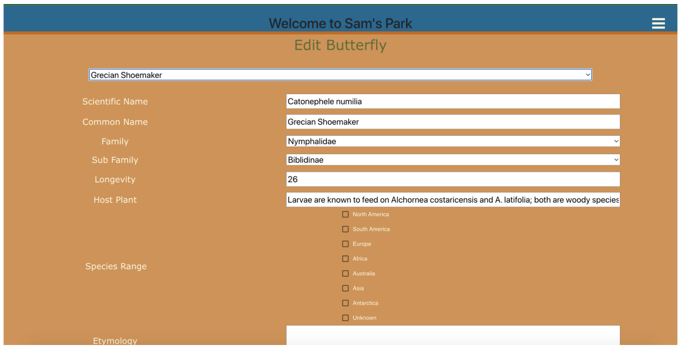
Longevity

Host Plant

Species Range
□ North America
□ South America
□ Europe
□ Africa
□ Australia
□ Asia
□ Antarctica
□ Unknown

Etymology

- Edit Butterfly
  - Here you can edit any of the butterflies that you've added from the add butterfly page. You can select a butterfly from the drop down menu at the top of the page. After which the fields will be updated with their current information.



Welcome to Sam's Park

Edit Butterfly

Grecian Shoemaker

Scientific Name: Catonephele numilia

Common Name: Grecian Shoemaker

Family: Nymphalidae

Sub Family: Biblidinae

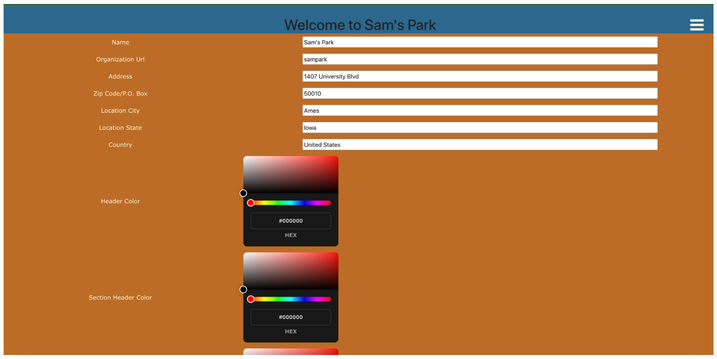
Longevity: 26

Host Plant: Larvae are known to feed on Alchornea costaricensis and A. latifolia; both are woody species

Species Range
□ North America
□ South America
□ Europe
□ Africa
□ Australia
□ Asia
□ Antarctica
□ Unknown

Etymology

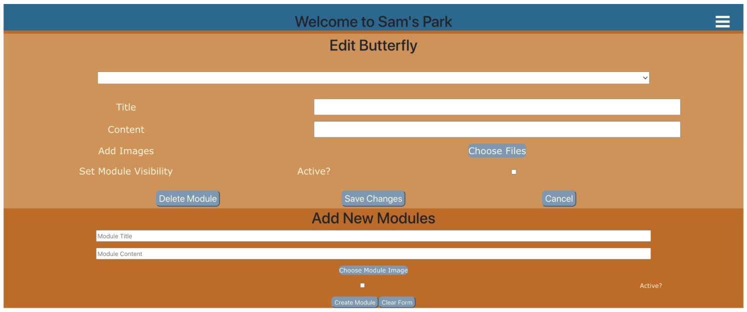
- Customize Page
  - Here you'll be able to change the look of your organization to make it your own! You can change things like the color palettes as well font, where you're located and more! Each Butterfly Park will also have its own unique url to navigate to. You can change this here. If you want to send out a link that just includes the home page to your park it will be as follows: www.flutr.org/[your park url]. Make sure that any logos or cover media are correctly sized.



Welcome to Sam's Park

Name	Sam's Park
Organization Url	sampark
Address	1407 University Blvd
Zip Code/P.O. Box	50010
Location City	Ames
Location State	Iowa
Country	United States
Header Color	#000000 HEX
Section Header Color	#000000 HEX

- Customize Modules
    - Here you can add modules that will display on your homepage. These are essentially little text blurbs that you can use to inform your visitors about upcoming events and more.



Welcome to Sam's Park

Edit Butterfly

Title

Content

Add Images — Choose Files

Set Module Visibility — Active? □

Delete Module — Save Changes — Cancel

Add New Modules

Module Title

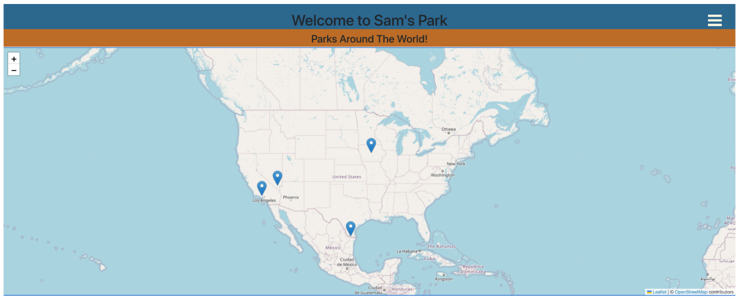
Module Content

Choose Module Image

□ Active?

Create Module — Clear Form

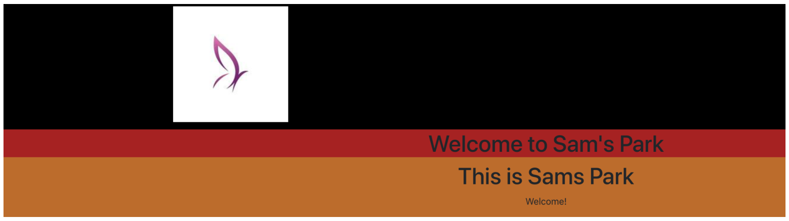
- User
    - Parks Around the world
        - This page displays a map that will show all the parks who have entered a location around the world. The map can be zoomed in and out using the buttons on the top left and you can pan around the map by holding left click. If you want to view a parks page that is displayed on the map you can click on the markers and that will navigate you to the other parks home page.



Welcome to Sam's Park

Parks Around The World!

- Homepage
  - The homepage for each of the butterfly parks will be unique! From here you can see information about the park as well as use the burger menu in the top right to navigate to other pages

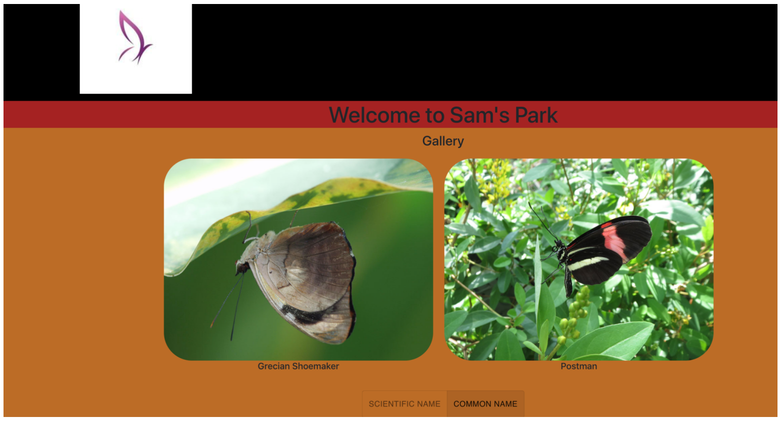


Welcome to Sam's Park

This is Sams Park

Welcome!

- Gallery
  - The gallery page is where you can view all the butterflies that are present in all of the gardens. You can switch between scientific name and local name which might switch depending on which parks page you are visiting! You can also click on the butterfly to view a page with facts about them.



Welcome to Sam's Park

Gallery