



Unified Caterpillar Tracker

Team:

Michael Gradle
Jonah Besselievre
Rose Druce-Hoffman

Ricky Smith
Gabriel Owen
Kristen Hawken

Client:

Nathan Brockman (Reiman Gardens)

Advisors:

Dr. Shana Moothedath



Impact

- Easier tracking + data input = more time spent analyzing caterpillar growth
- Jenny Shrum and members of the On Sacred Ground Organization
- Could be used by other groups
- Small step towards better caterpillar/butterfly conservation



Functional Requirements



Data Collection and Data Sharing

- Easy to find and use toggles, for opting in/out of shared data or seeing global platform collections.
- Letting the user decide what they want on their screen.
- Various data exports

Ease of Data Collection

- Gamified program that leaves users feeling rewarded for contributing data.
- Gather data automatically whenever possible (APIs)
- Purchase and integration of different sensors for tracking environment

Monetary Requirements

- Entirety of service must be relatively inexpensive
- < \$30 monthly
- Migration should be inexpensive and viable

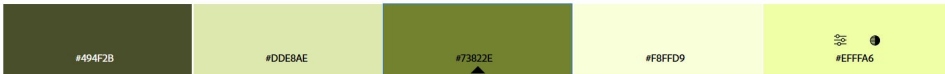


Early Concept

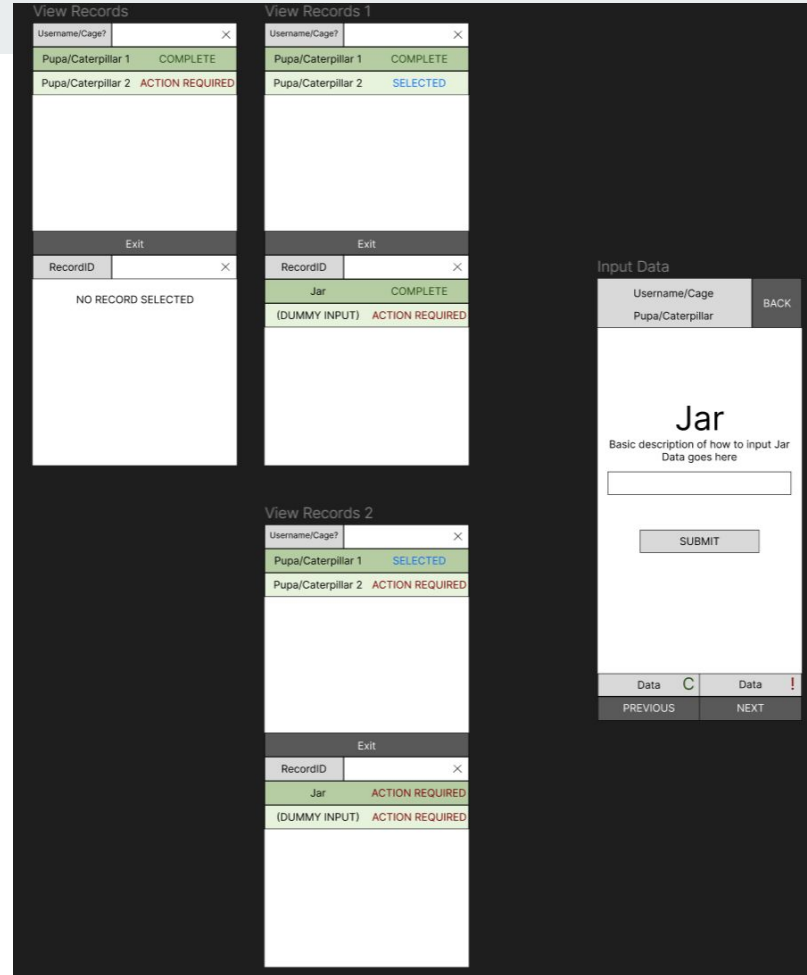
- Screen is split to show individual records of caterpillar
- Simplistic description
- No import/export implementation yet

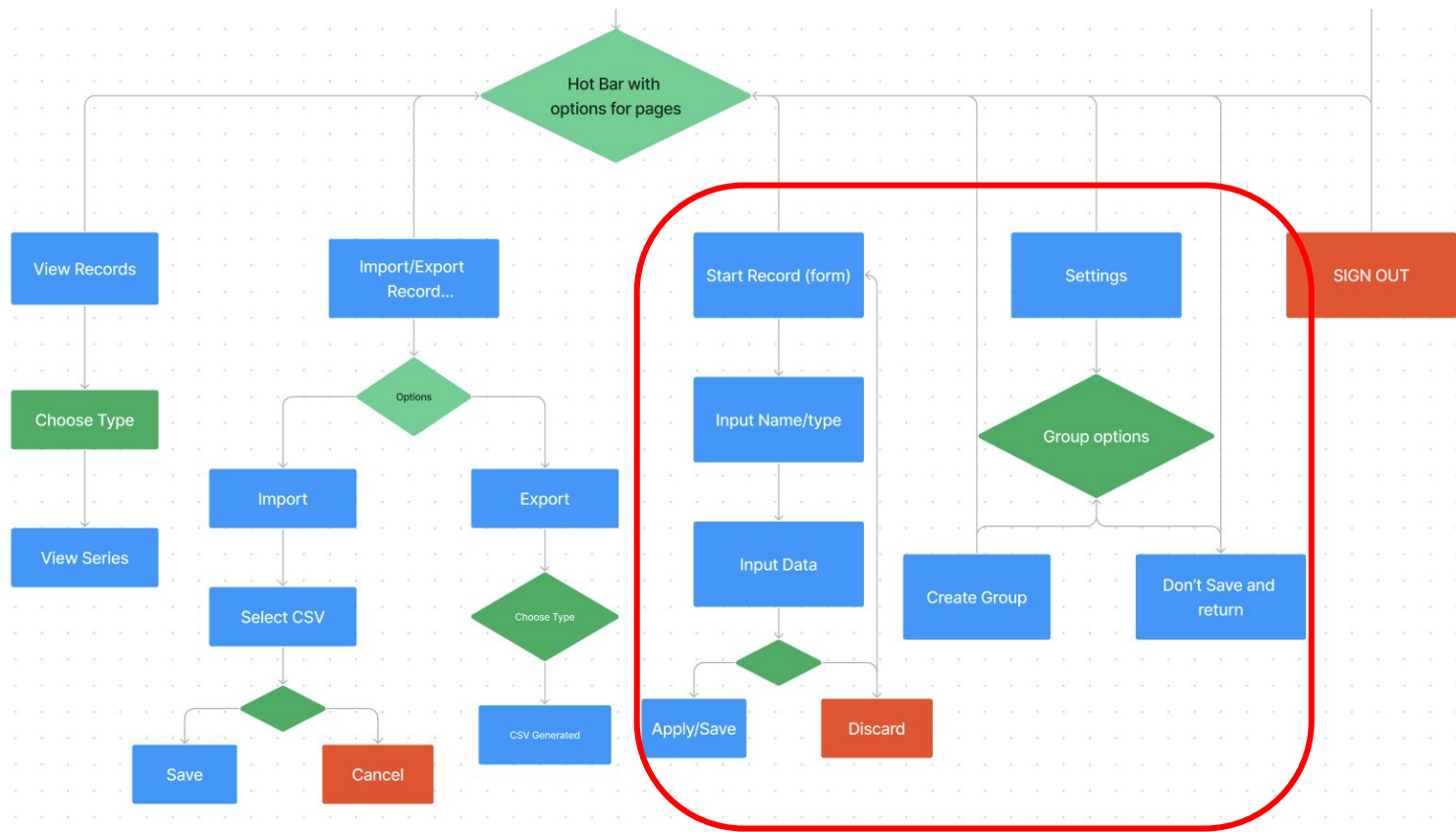


Color Harmony Color Mode
Monochromatic ▾ RGB ▾



#494F2B #DDEBAE #73822E #F8FFD9 #EFFFAG





Tools



React / JavaScript

- React-table
- Moment
- Papaparse
- Many, many more

AWS

- Amplify
- Cognito
- DynamoDB


3rd Party APIs

- Automation purposes
- Google Maps (GPS)
- Metomatic (weather)

Start Screen

User Creation / Login

- Simple, easy, few inputs
- For the scale of our app, we don't need too much user data
- Forgot Password option if a user does not remember their login credentials




Login

Email

Password

Forgot Password? [Click Here](#)

Create Account Login



Create Account

Name

Email

Password

Re-enter Password

Create Account Login

Site-wide Authentication

```
// This is a custom hook that checks if the user is authenticated. If the user is not authenticated, it will redirect the user to the login page.
function PrivateRoute({ children }) {
  const navigate = useNavigate()

  // Check if the user is authenticated when the component mounts
  useEffect(() => {
    checkAuthState()
  }, [])


  // Check if there is a current user, if not, redirect to the login page
  async function checkAuthState() {
    try {
      await getCurrentUser()
    } catch {
      navigate("/")
    }
  }

  // Render the child components if the user is authenticated
  return <{children}>
}

const routes = [
  { path: "/", element: <LoginSignup />, isProtected: false },
  { path: "/dash", element: <Dashboard />, isProtected: true },
  { path: "/setup", element: <CreateAccount />, isProtected: true },
  { path: "/settings", element: <Settings />, isProtected: true },
  { path: "/view", element: <ViewRecords />, isProtected: true },
  { path: "/import", element: <ImportRecords />, isProtected: true },
  { path: "/create", element: <CreateRecords />, isProtected: true },
  { path: "/export", element: <ExportRecords />, isProtected: true },
]

// This is the main App component that renders the routes. It also renders the PrivateRoute component for protected routes.
function App() {
  return (
    <div className="App">
      <Routes>
        {routes.map((route) =>
          route.isProtected ? (
            // If the route is protected, render the PrivateRoute component
            <Route
              key={route.path}
              path={route.path}
              element={<PrivateRoute>{route.element}</PrivateRoute>}
            />
          ) : (
            // If the route is not protected, render the route element
            <Route key={route.path} path={route.path} element={route.element} />
          )
        )}
      </Routes>
    </div>
  )
}
```

Dashboard



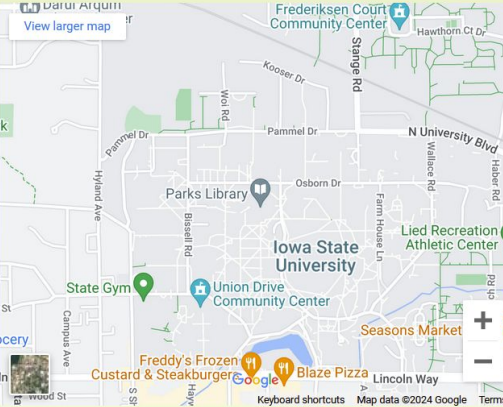
- Import Records
- View Records
- Settings
- Create Records
- Logout

Welcome!

Releases today: 0

Releases this season: 0

Total collections this year: 0

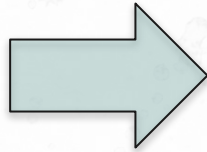


Current Weather

Create Record

New Record
Select Type:

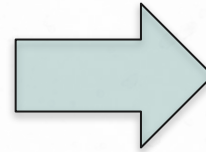
Edit Record
Select Type:



New Caterpillar
Select Field:
Date:

Previous **Next**

Review
Cancel



New Caterpillar

Form Data:

IndividualID: 24SI004
Date: 2024-04-30
Site: Third Lagoon
Location: N 42.XXXXX W -93.XXXXX
Time: 20:39
Plant: Sisymbrium
Stage: 1st Instar
Collected By: Shrum

Back **Submit**
Cancel

View Records

Import Records View Records Settings Create Records Logout

View Caterpillar Records

ID	Cage	Stage	
<input type="checkbox"/>	24BR001	01	Butterfly
<input type="checkbox"/>	24SI002	Beta02	2nd Instar
<input type="checkbox"/>	24LE001	Alpha01	1st Instar
<input checked="" type="checkbox"/>	24SI001	01	Pupa

Date	Status	Stage	Location	Comments
04-27	OK	1st Instar	B5	N/A

Delete Caterpillar?

New Check

Caterpillars **Chrysalides**

Import Records View Records Settings Create Records Logout

Perform Check

Caterpillar Check

Set Caterpillar:

Date	Status	Stage	Location	Comments
04-27	OK	2nd Instar	B3	N/A

Select Field:

Date:

Previous **Next**

Review

Delete Chrysalis?

New Check

Caterpillars **Chrysalides**

Import Records View Records Settings Create Records Logout

View Chrysalis Records

ID	Position	Orientation	Deformation	
<input checked="" type="checkbox"/>	24SI001	Middle Third	Head Up	None
<input type="checkbox"/>	24BR001	Upper Third	Head Up	None

Date	Cage	Slot	Status	Comments
04-27	01	0A	Good	N/A

Select Field:

Date:

Previous **Next**

Review

Delete Chrysalis?

New Check

Caterpillars **Chrysalides**

Import Records View Records Settings Create Records Logout

Perform Check

Chrysalis Check

Set Chrysalis:

Date	Cage	Slot	Status	Comments
04-27	01	0A	Good	N/A

Select Field:

Date:

Previous **Next**

Review

Delete Chrysalis?

New Check

Caterpillars **Chrysalides**

Edit Record

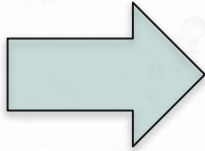
Import Records View Records Settings Create Records Logout

New Record

Select Type:

Edit Record

Select Type:



Import Records View Records Settings Create Records Logout

Edit Caterpillar

Set Caterpillar:

Edit Checks?

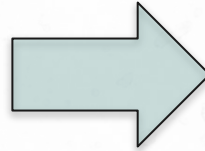
Select Field:

Date:

Previous **Next**

Review

Cancel



Import Records View Records Settings Create Records Logout

Edit Caterpillar

Set Caterpillar:

Back to caterpillar?

Date	Status	Stage	Location	Comments
<input type="checkbox"/> 2024-04-27	Molting	Pupa	MS Stem	

Cancel

Groups

Settings **Group**

You're currently not in a group!

This screen will show you all of your current invitations, giving an option to Accept/Deny them. Alternatively, you can create a new group.

Create Group

Invitations

Name	Owner
<input checked="" type="checkbox"/> TestGroup	testuser3@gmail.com

Accept Invitation?

Settings **Group**

TestGroup

Owner: testuser3@gmail.com

Permission Level: Owner

User	Permissions
<input type="checkbox"/> testuser2@gmail.com	Admin
<input type="checkbox"/> testuser3@gmail.com	Owner

Invite User

Leave Group



Import/Exporting Data



Importing

- Accepts .CSV files
- Using the Papaparse library for data parsing
- Generates a table view so the user can see what data will be imported
- Not fully working, we are having difficulties mapping data to a database object

Exporting

- Downloads as a .CSV
- Using the Papaparse library
- Creates the headers that are in the database
- Working, but we want to add more filtering features

Exporting Screenshots



Export Records

[Click here to import a Record](#)

- Export Caterpillar Data
- Export Caterpillar Check Data
- Export Chrysalis Data
- Export Chrysalis Check Data
- Export Release Data



Individual ID	Created By	Created On	Updated On	Date Found	Location Found	Time Found	Host Plant
24BR003	mgradle@iastate.edu	2024-04-23T21:52:14.055Z	2024-04-23T21:52:14.055Z	4/23/2024		16:51	Brassica

CSV data from exporting Caterpillar Data

Difficulties

- New tools and languages to learn, slowed development time
 - React, Amplify, JavaScript, HTML/CSS
 - AWS services and integration were very new to us
- Some errors were hard to track down and fix
 - Error handling in JS is not simple
 - Settings in different AWS services impacted hosting and collaboration
- Change in requirements
 - Reworking the data models
 - Changing and removing functionality.
- COST!!





Future Additions/Changes



Greater Customizability

- Accept various forms with a wider variety of fields
- Better filtering options on exporting data
- Allow better organization/group interactions
- User is able to create a form with custom fields

User Interface

- Improve the UI to look a bit more modern
- Make the website scale better from mobile → desktop

More Rigorous Testing

- Create a full test suite to test functions of the app
- Chromedriver (Selenium) to create automated tests
- Find all potential sources of errors in the application

Conclusion

- We learned many valuable skills from this project
 - Client interaction
 - Web development
 - Cloud services
- Overall, we are really proud of the project
 - However, always room for improvement
- Hopefully, the app meets the standards of the client and will be put to good use



*Thank You for Listening
Questions?*