



Fall Lightning Talk

FLUTR - TEAM 11



INTRODUCTION

Flutr is an all-in-one web app for flight houses that aims to aid both flight house employees and visitors, each with their own goals.

For employees:

- Easy tracking of shipments and releases
- Allows employees to keep track of butterflies in flight in pavilion
- Allows employees to keep track of poor or ill butterflies
- Employees will have the ability to edit past data
- Can export data to spreadsheets to report to the USDA

For visitors:

- Data formatted into a user-friendly interface that helps them learn about the butterflies in the pavilion
- The landing page will include statistics like butterfly locations, the rarest butterfly in the pavilion, a “butterfly of the day”, and more
- Visitors will be able to learn about each butterfly they see, with the hopes of creating a more enjoyable user experience

REQUIREMENTS

Butterfly Tracking:

- Track/add/edit shipment providers and individual shipments in a database.
- Update predictive lifespan data and delete end-of-life butterflies automatically.
- **Intuitive and quick logging interface for release** and documentation for training and reference.

General Public Experience:

- Engaging platform for accessing butterfly facts and statistics.
- **Kiosk with improved cooling and sealing to operate in high humidity and temperature conditions.**
- Kiosk with security features and QR codes to prevent unauthorized access to admin functions and enable on-site learning.

Customization and Maintenance::

- Separate management and user access levels to allow houses to independently manage their own systems and staff.
- **Website customization options for each flight house** to tailor their tracking systems.

DESIGN OVERVIEW

In order to keep costs down and uptime at a max, we will be hosting the entire system on DigitalOcean.

We decided to split the team into frontend and backend halves, with each side focusing on either the GUI and frontend API calls, or the backend database management and data serving.

By focusing first on the laboratory side of the application and then moving to the visitor-facing side, we can populate the database with solid data before adding special features for the visitors to interact with.

FRONTEND OVERVIEW

- Flutr's Frontend is built with React.js
- The React application is hosted using NGINX.
- React was chosen because it is the new standard for web development.
 - Component based, fast working framework.
- Currently Comprised of 24 Unique pages
 - These pages are reconfigured and altered in theme based on each location's specifications
- Fetch library for HTTP Requests.
- Only allows users to visit pages they are authorized to

BACKEND OVERVIEW

- APIs developed with Spring Boot, database architecture in MongoDB
 - APIs are stateless and ensure efficient and seamless communication with the frontend
- Multi-tenant database design allows for efficient and dynamic data handling
- Comprehensive error handling mechanisms
 - Clear logging for maintenance
- User security system using JWT authentication for role-based and localized data access
 - CORS policies applied to secure request/response flow with the client
- Integration with DigitalOcean Droplets and Spaces
 - Scalable hosting and efficient CDN for file uploads and distribution

CURRENT PROGRESS/OUTCOMES

- At this point we have a minimum viable product
- We are working on fixes and quality of life improvements
- More features on the way
- Results of Nathan's Demo at Conference:
 - Every Butterfly Garden in attendance was made aware
 - Some on board to help test the application

PENDING ISSUES/CONCERNS

- Still lots more bugs to fix
 - CORS issues
- Complete public facing features
 - Statistics and Butterfly of the Day
- Add better user feedback
 - Error messages, more stylized response messages.
- Spreadsheet Import/Export
- Cleaner looking forms. Visual Tweaks.