Lightning Talk 6

FLUTR - TEAM 11

PROBLEM OVERVIEW

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.
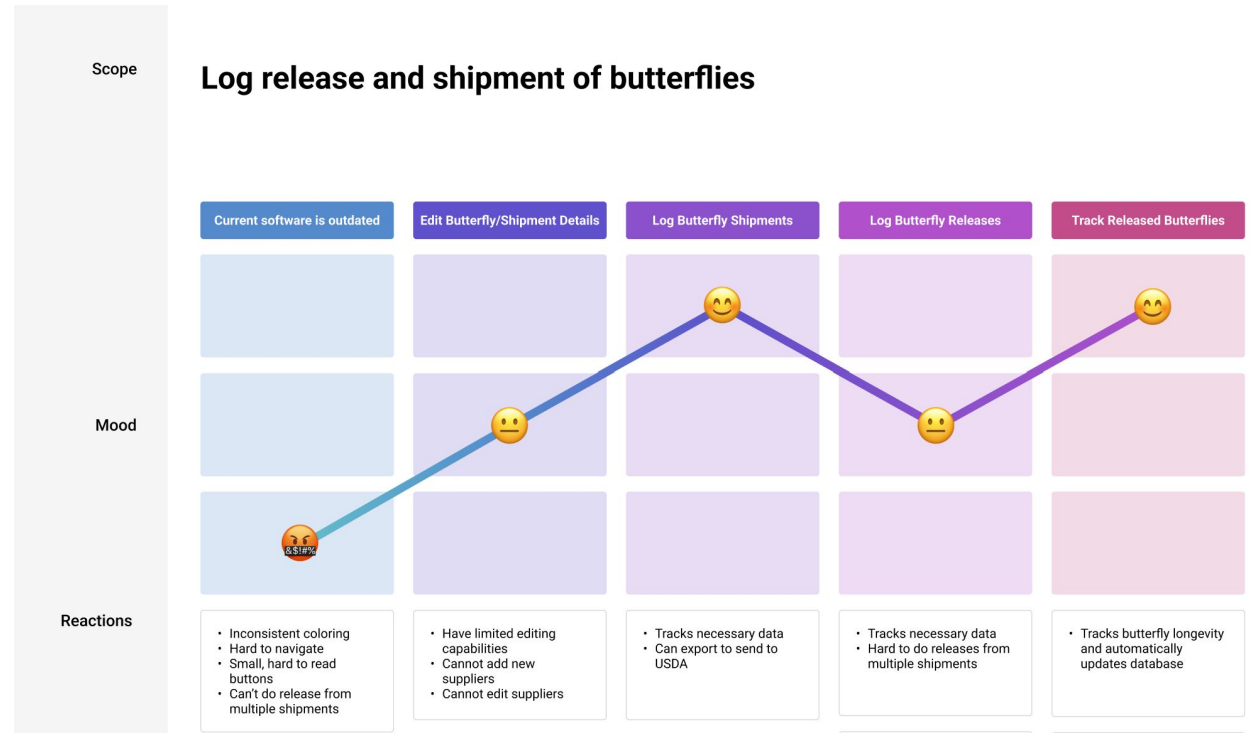
For employees, Flutr will allow for easy tracking of both butterfly shipments as well as releases. As employees receive shipments and release butterflies into the pavilion, they will be able to mark off which butterfly was released, in order to keep consistent tracking of the butterflies they release. They will also be able to track which butterflies had a failed cocooning. Employees will have the ability to edit past data, and export tracked data to spreadsheets to report to the FDA.

For visitors, the tracked data will be taken and formatted into a user-friendly interface that allows for easy learning 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.

Journey Map

Journey Map for client

- Begins with issues from current software
- Transitions to our solution

Scope	Log release and shipment of butterflies				
	Current software is outdated	Edit Butterfly/Shipment Details	Log Butterfly Shipments	Log Butterfly Releases	Track Released Butterflies
Mood	😡 &$!#%	😐	😊	😐	😊
Reactions	• Inconsistent coloring • Hard to navigate • Small, hard to read buttons • Can't do release from multiple shipments	• Have limited editing capabilities • Cannot add new suppliers • Cannot edit suppliers	• Tracks necessary data • Can export to send to USDA	• Tracks necessary data • Hard to do releases from multiple shipments	• Tracks butterfly longevity and automatically updates database

Pros & Cons

- Based on market research

	Flutr	Reiman Gardens	In-house Data Tracking
Pros	Software can be shared to other pavilions	Good information for guests, good butterfly tracking	Complete customizability for pavilions
Cons	Cannot add new suppliers	Outdated, clunky	Technically difficult for other pavilions to produce

Technical Complexity

Technologies Involved

- React, DigitalOcean, Springboot
- Need to ensure each of these can communicate properly

Familiarity with Technologies

- Good amount of experience with Springboot from 309
- Team leader did a cloud computing internship
- All relatively new to React

Human

- Humans will be interfacing with our product every day.
- General Users will see the customer facing side of things
  - They want to be able to learn about the location they are at.
  - With our application, they will be able to easily navigate to their location's site and see the information served to them by our automatically generated content or the content the admins want shown to the general users.
- Admins will be using the application every day of work to record their activities related to butterflies.
  - They need an easy and quick way to enter this information.
  - They need to be able to edit information previously entered.
  - Our application will easily help them complete all of this.

Economic

- Flutr improves upon existing solutions like the previous iterations of this project in the following ways:
  - Standardizes butterfly tracking and information sharing globally across partner flight houses.
  - Customization features allow both Reiman Gardens and other flight houses to tailor the system to their needs, manage employees, and control access levels, enhancing user experience and efficiency.
  - Allows for editing entered data in databases, in case of errors or later updates to shipments.
- A potential drawback of our solution could be:
  - The switch to our version of Flutr could involve a change in how Reiman Gardens and other flight houses currently track their butterflies and shipments. Developing detailed and user-friendly documentation could help address this challenge, and require less on-site training and maintenance.

Technical

Our iteration of Flutr incorporates modern technologies like React.js, SpringBoot, and MongoDB, which demonstrates our ability to use current and well-known technologies for robust and long-lasting solutions. Designing our project using these frameworks also allows us to showcase the expertise we've developed through our coursework and experiences so far.

Considering the environmental factors in designing the kiosk hardware reflects a high level of internal complexity, considering the innovation we have to apply to address such unique circumstances.

Externally, we also needed to also take into account the user-centricity of our system and develop an engaging and efficient user interface, making it accessible to users of varying technical backgrounds. The system's architecture, considering both hardware and software needs, reflects our understanding and expertise in developing solutions that accommodate a broad range of user groups.