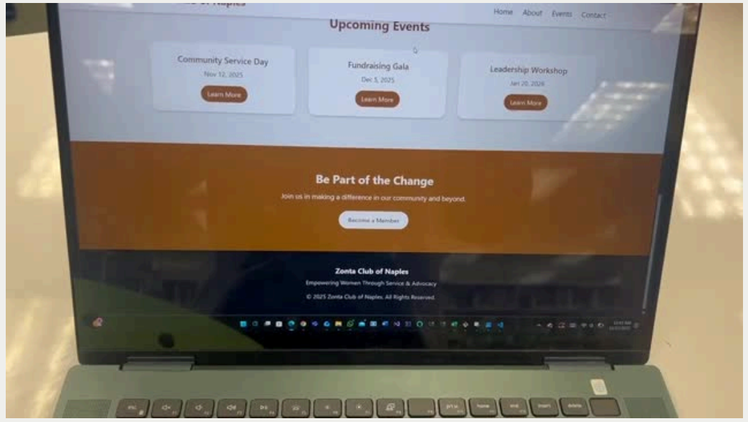
ZONTA CLUB NAPLES

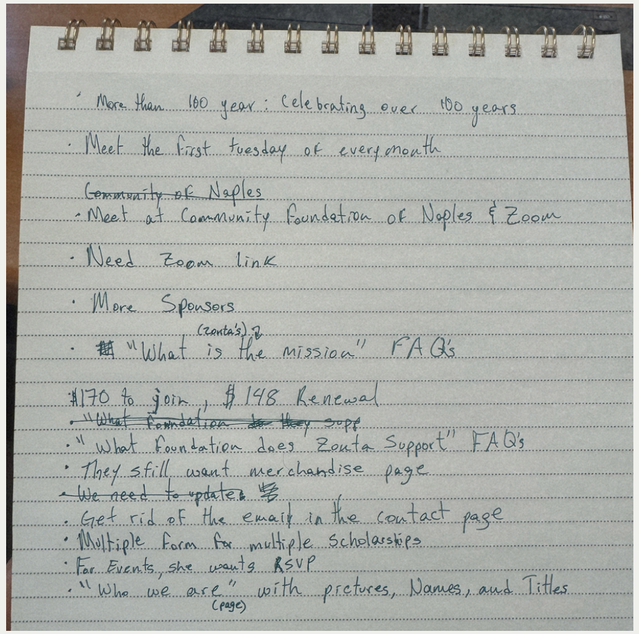
DENNIS RAY, BRIAN VELASQUEZ, ESMIR RAMIREZ, JOSHUA MARTIN, SUHAIB FARAH

OUTLINE

- The Objective
- Frontend Slide
- API Slide
- Backend Slide
- UI Tour Slide 1
- Feature Analysis
- Conclusion

THE OBJECTIVE





We began by analyzing the requirements given by the Zonta Club of Naples, and immediately began work on an SRS document. From here, we began work on the backend, allowing us to use a Rapid Prototyping Model, which we chose, due to its ability to allow for rapid feedback.

We showed some of these artifacts to the Zonta Club, in order to solicit feedback, which is shown here. This would prove to greatly help us throughout our development process.

Below were all of the necessary core artifacts that needed to be implemented:

Content Display Function- Display mission, history, leadership, and general organizational. information. (frontend)

Event Management Function- Show upcoming events, past events, and event details. (backend)

Donation Function- Allow users to make online donations through a payment provider, as well as the ability to pay dues. (backend, Stripe API)

Admin Content Management Function- Authorized users can create, edit, and delete site content. (backend)

FRONTEND TOUR

Our frontend consists of the following components:

1. A Home Page
2. A “Who We Are” Page
3. An Events Page, which connects to our backend via an API. These Events can be set by an administrator.
4. A membership application form, which sends a completed form via our API to an administrator.
5. A Scholarship Page
6. An Impact Page

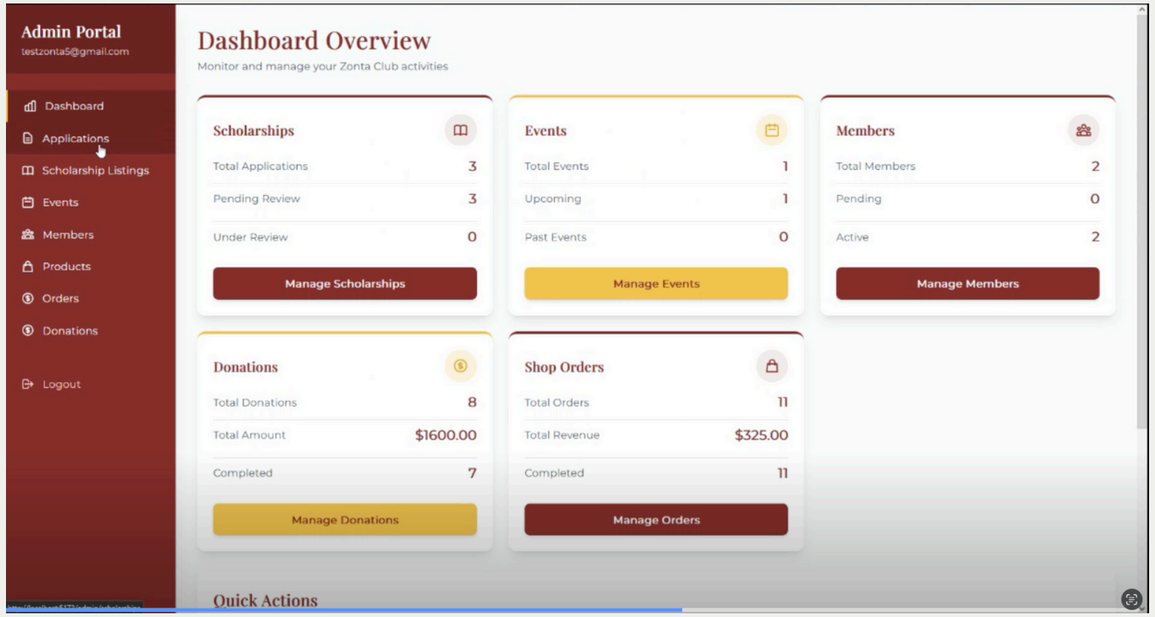
Throughout our development, we created reusable React components, so as to allow for maximally clean and modular code, so as to maximize maintenance, an important part of the software development life-cycle.

API TOUR

All elements of the website that are not static can be easily edited at the helm of an administrator, including storefront products, fulfilling orders, accepting scholarships, analyzing donations, adding members, and scheduling events.

For this, we needed a API system that would effectively allow for this to occur. Our backend, which used Express JS, offered an API, which our frontend connected to get this non-static data. Furthermore, however, our backend needed to query another API, so as to allow for the processing of payment information. It was here that we used Stripe's API as an intermediary between our backend and Stripe's payment processor.

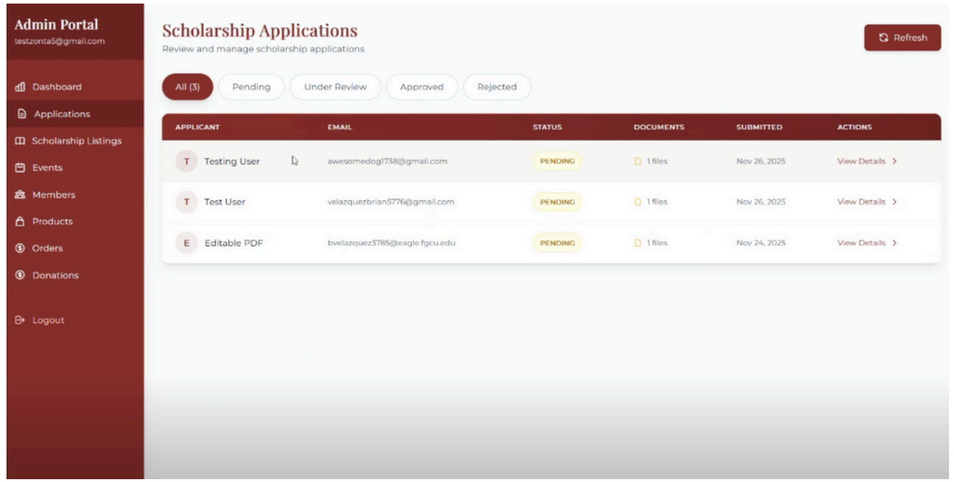
We also used Cloudinary to manage the uploading and processing of documents, like scholarship applications.

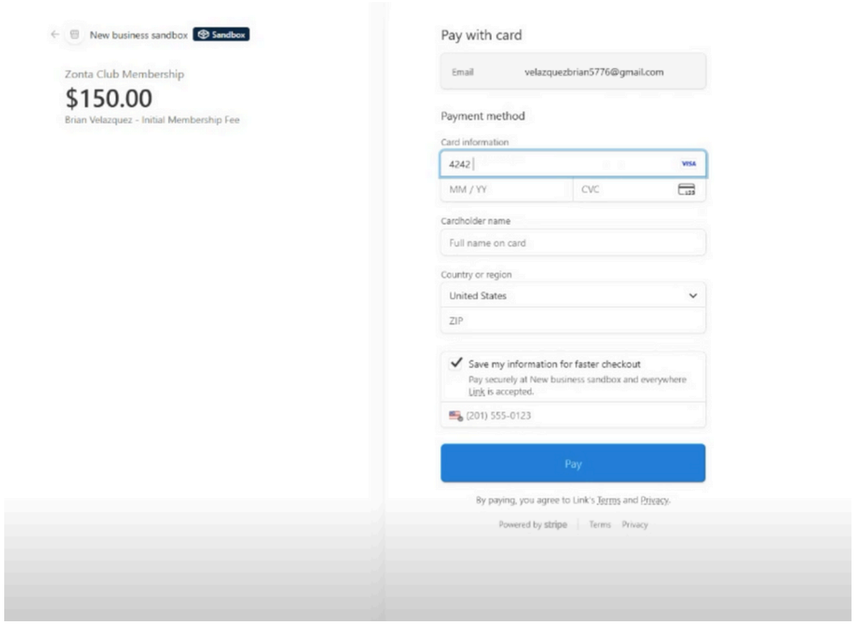


BACKEND

For the backend, we used Express JS, which allows us to work with various sources of variable data, from scholarship applications, to donations. We processed this data, such that the frontend-based admin interface is as easy to use as possible. It loads:

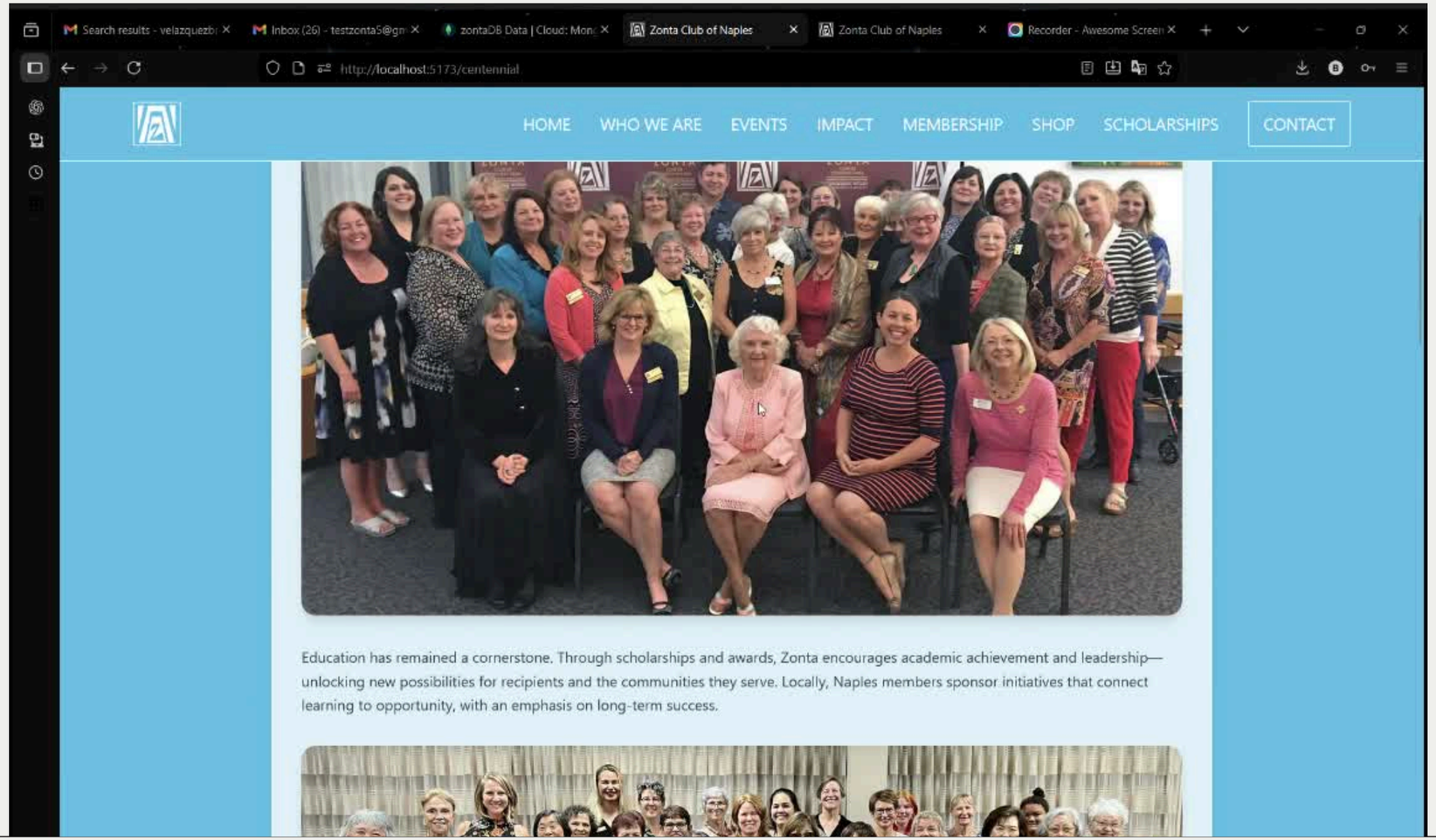
- all scholarships
- all events
- all members
- donation stats
- order stats





For our payment processor, we needed to ensure that, in addition to using APIs to contact Stripe's servers, that Stripe could also contact us. We used what is known as a webhook to facilitate this. Webhooks guarantee our system knows when payments succeed.

VIDEO DEMO


Search results - velazquezbr
Inbox (26) - testzonta5@gm
zontaDB Data | Cloud: Mon
Zonta Club of Naples
Zonta Club of Naples
Recorder - Awesome Screen
http://localhost:5173/centennial
HOME
WHO WE ARE
EVENTS
IMPACT
MEMBERSHIP
SHOP
SCHOLARSHIPS
CONTACT
Education has remained a cornerstone. Through scholarships and awards, Zonta encourages academic achievement and leadership—unlocking new possibilities for recipients and the communities they serve. Locally, Naples members sponsor initiatives that connect learning to opportunity, with an emphasis on long-term success.

WHAT WERE WE ABLE TO DELIVER?

We were able to effectively deliver a website that was within the requirements of the Zonta Club, that is, we addressed their major issues with their current website, which are as such:

- Aesthetics, necessitating the redevelopment of the Zonta's Naples website.
- Inconvenient administrative backend for content updates, blog posts, event handling, and membership information.
- Possibly integrate donation capabilities, newsletter subscription, and other dynamic content.

We effectively converted these into requirements, and turned these requirements into a detailed analysis. Furthermore, elements such as the "Who We Are" page would not have been possible, had we not adopted a rapid prototyping model, so as to effectively elicit feedback.

What would we do next time?

In future iterations, we hope to add more sources of dynamic content, including a newsletter, a search function, and the ability for members to log in and access special member-only information, such as event locations, as currently, only admins can log in.

THANK YOU!