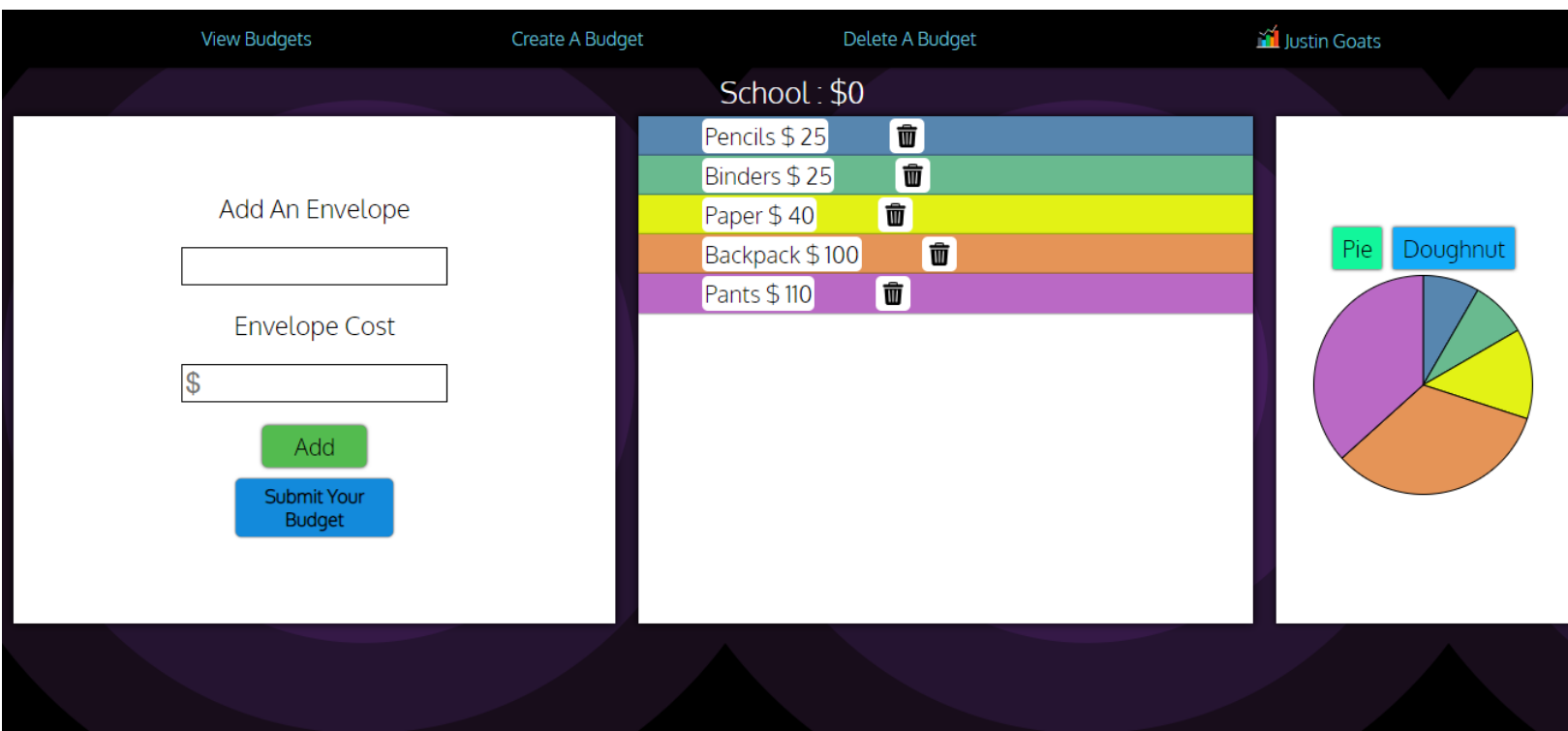


An Overview Of "Budget App"



Description

A simple custom budget app for creating personal budgets quickly and easily. Signed in users can create an account and then proceed to create a custom budget for their needs.

Purpose

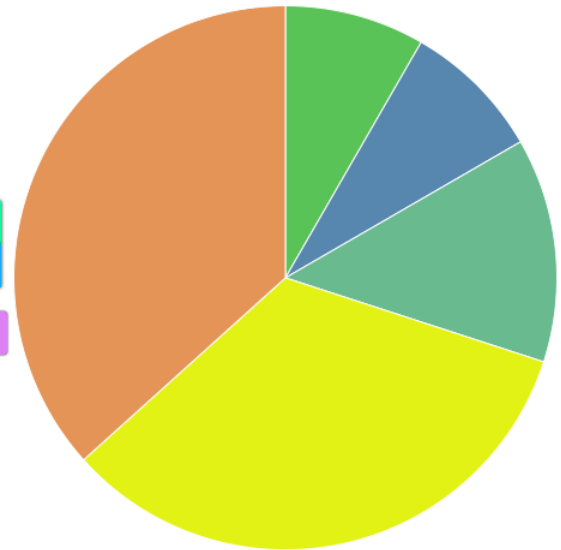
I have a hard time saving money due to my love of eating out and until I created a budget using this app and seeing how much I really spend on food every month compared to all my other expenses, I was shocked.

Objective

My main objective for this application was the ability to create a visual budget using graphs to make it much easier to view data. I also wanted to make it as easy as possible to create a budget. Other budget applications I have seen make it way more complicated than it really needs to be.

Pencils \$ 25
Binders \$ 25
Paper \$ 40
Backpack \$ 100
Pants \$ 110

Pie
Doughnut
Back To Budgets



Duration

The application ended up taking me a little over a month to complete.

Half of that time was figuring out a secure way to administer authentication using JWT's. I did a lot of research and concluded that storing the JWT in session storage and local storage was not an option because it's highly susceptible to XSS attacks

I ended up storing the JWT in memory inside my React application's state. This approach was more secure but it would also mean that if the user refreshed the page, they would be logged out. I have since found a better way to handle authentication and sessions using HTTP cookies alongside JWT's which is still secure but also keeps a user signed in on a page refresh.

Challenges

One of the hardest challenges for me was creating protected client side routing. I was used to server side protected routes by just using some custom authentication middleware with Express. However I had no idea how to accomplish this on the client side with React. After trying out several methods with little success it eventually dawned on me that I can create a React component that checks for a JWT and if one exists, I get redirected using "this.props.history.push" which is available from "react-router-dom". This ended up being a nice solution and makes it impossible to get to the dashboard page without a valid JWT.

Technologies Used

HTML CSS JavaScript React.js

Webpack Express.js MongoDB

Mongoose SASS JSON Web Tokens