COLLEGE OF ENGINEERING

Electrical Engineering and Computer Science

ADVENTURE VAN ROOF RACK – WEB CONFIGURATOR

https://ovg-configurator-990c621e4b6fad0e7a21.o2.myshopify.dev/

David Cornett: cornettd@oregonstate.edu
Coder Myers: myerscod@oregonstate.edu
Mason Stephenson: stephmas@oregonstate.edu
Joe Thompson: thompjo5@oregonstate.edu

Shopify integrated custom web-app to streamline the roof rack design process for Orion Van Gear engineers and their customers.

BACKGROUND

This project applies computer science skills to a real-world problem. Orion Van Gear was searching for a solution to mitigate the extensive communication required to design client van roof racks with pre-existing equipment.

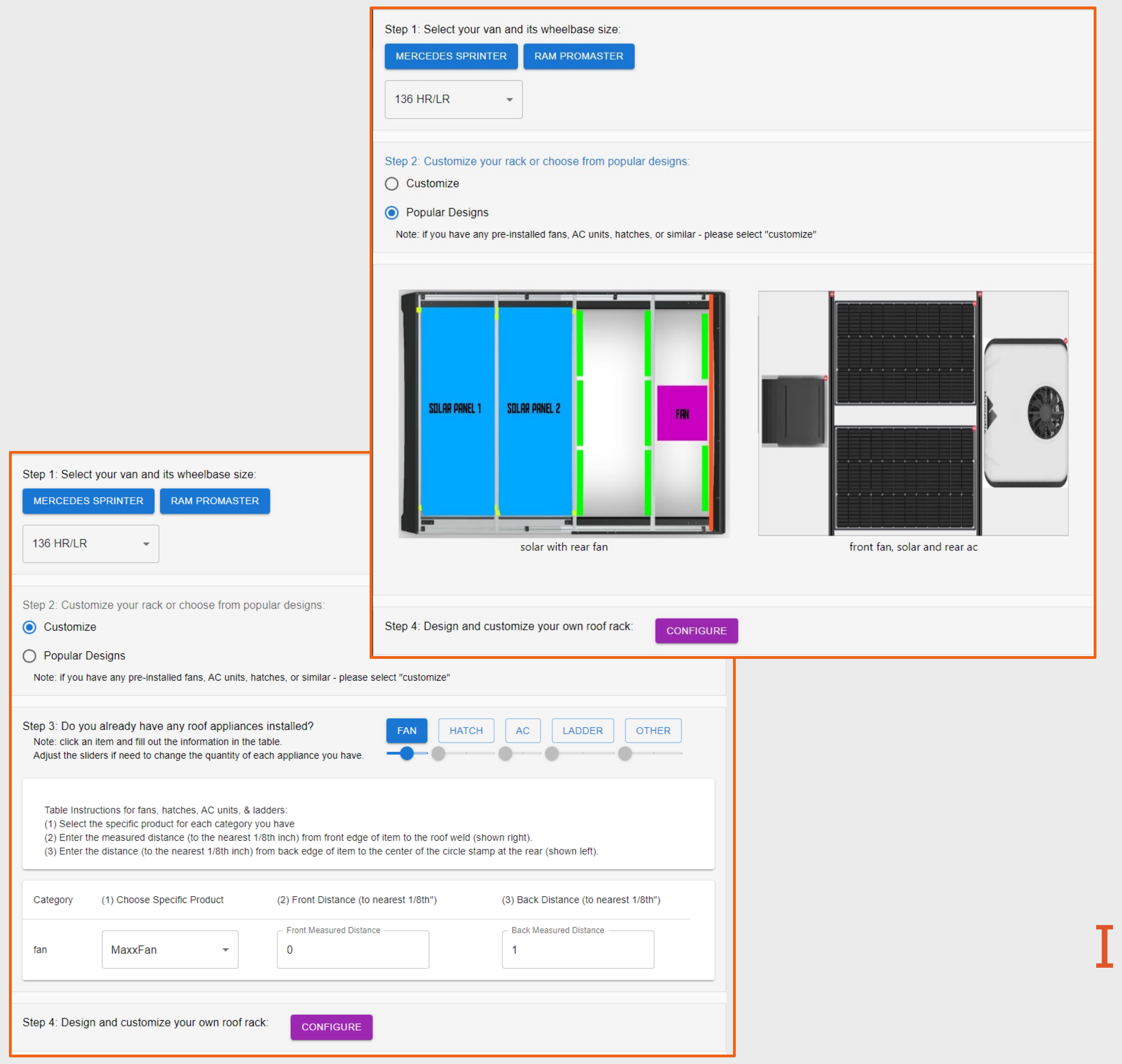
COMPLEXITY

Orion Van Gear explained the bottlenecks in the design process are checking:

- Roof rib locations relative to requested A/C Unit or Fan placements
- Fitting additional equipment such as solar or deck paneling around the roof installed equipment relative to compatible roof rack crossbar locations
- Ensuring that clients are measuring their pre-installed equipment from the same reference points

KEY TECHNOLOGIES

- React: Component-based front-end JavaScript library used to dynamically render page content.
- Remix: Full stack framework used to implement and isolate server functions from display functions.
- React-grid-layout: Grid layout library used to implement precise drag-and-drop for store products.
- Shopify GraphQL Storefront API: Provides authenticated communication with a Shopify store to fetch product details and build shopping cart.
- Shopify Oxygen: Shopify host server for custom Hydrogen-React based store front.


Step 1: Select your van and its wheelbase size:
MERCEDES SPRINTER | RAM PROMASTER
136 HR/LR

Step 2: Customize your rack or choose from popular designs:
Customize
Popular Designs
Note: if you have any pre-installed fans, AC units, hatches, or similar - please select "customize"

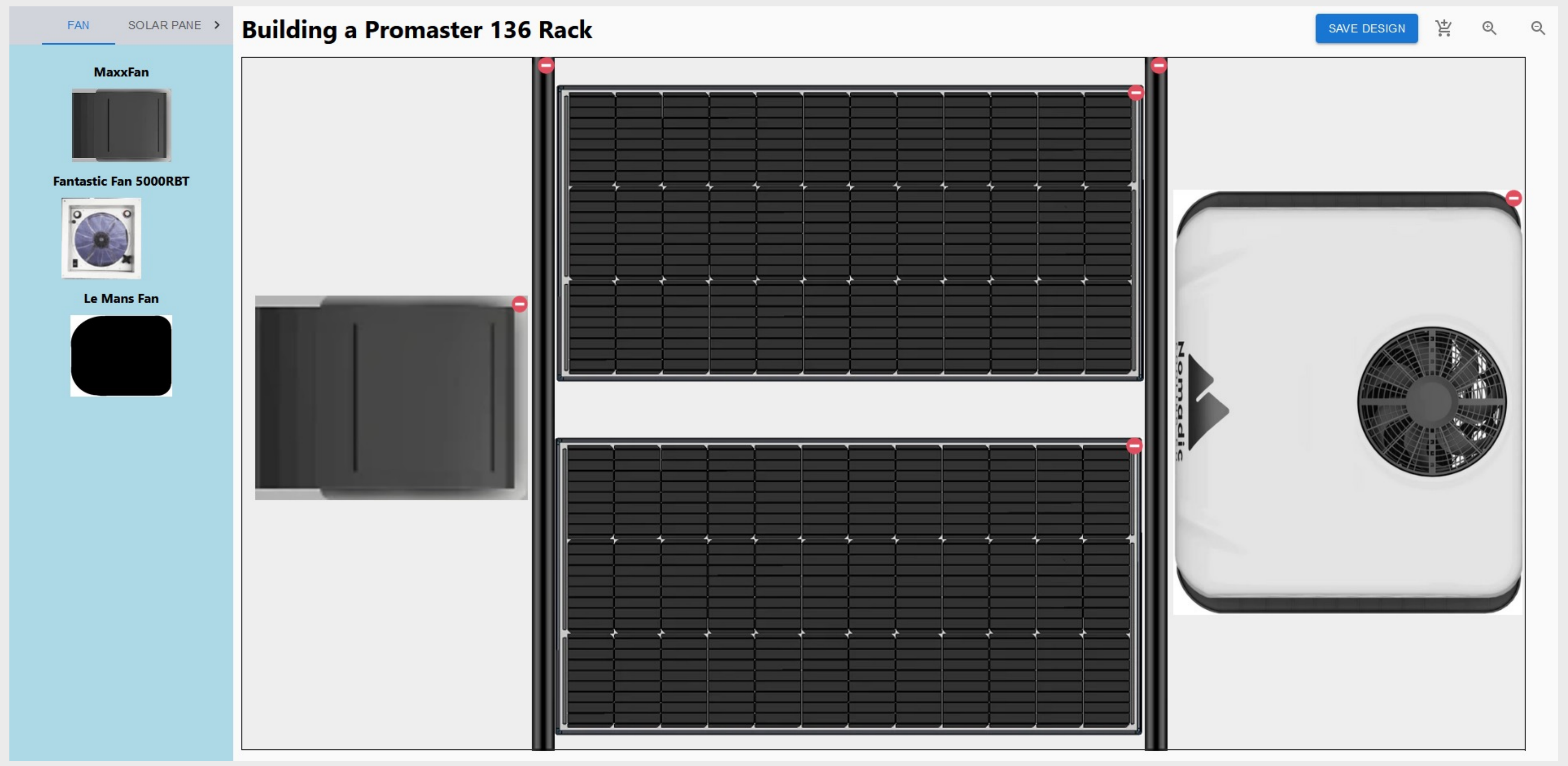
solar with rear fan
front fan, solar and rear ac

Step 3: Do you already have any roof appliances installed?
Note: click an item and fill out the information in the table.
Adjust the sliders if need to change the quantity of each appliance you have.
FAN | HATCH | AC | LADDER | OTHER

Table Instructions for fans, hatches, AC units, & ladders:
(1) Select the specific product for each category you have
(2) Enter the measured distance (to the nearest 1/8th inch) from front edge of item to the roof weld (shown right).
(3) Enter the distance (to the nearest 1/8th inch) from back edge of item to the center of the circle stamp at the rear (shown left).

Category	(1) Choose Specific Product	(2) Front Distance (to nearest 1/8th")	(3) Back Distance (to nearest 1/8th")
fan	MaxxFan	Front Measured Distance: 0	Back Measured Distance: 1

Step 4: Design and customize your own roof rack: CONFIGURE



FEATURES

The key features needed for a successful solution.

- The webapp must integrate with their existing Shopify storefront.
- The webapp must allow users to select from popular designs or create a design from scratch.
- Users must be able to specify multiple pre-existing roof appliances on their vans.
- Equipment placement must be accurate to the 0.125 inch.
- Users must be prevented from placing equipment in specific locations due to roof rack limitations, or van roof rib limitations.

After meeting this criteria new features were implemented

- The webapp must allow users to save their designs and return to them later.
- Add components used in designs to the shopping cart
- New products or configurations can be added through the Shopify Admin Portal
- Auto place crossbars and snap equipment to nearby crossbars
- Dynamic resizing to accommodate smaller screens

IMPLEMENTATION

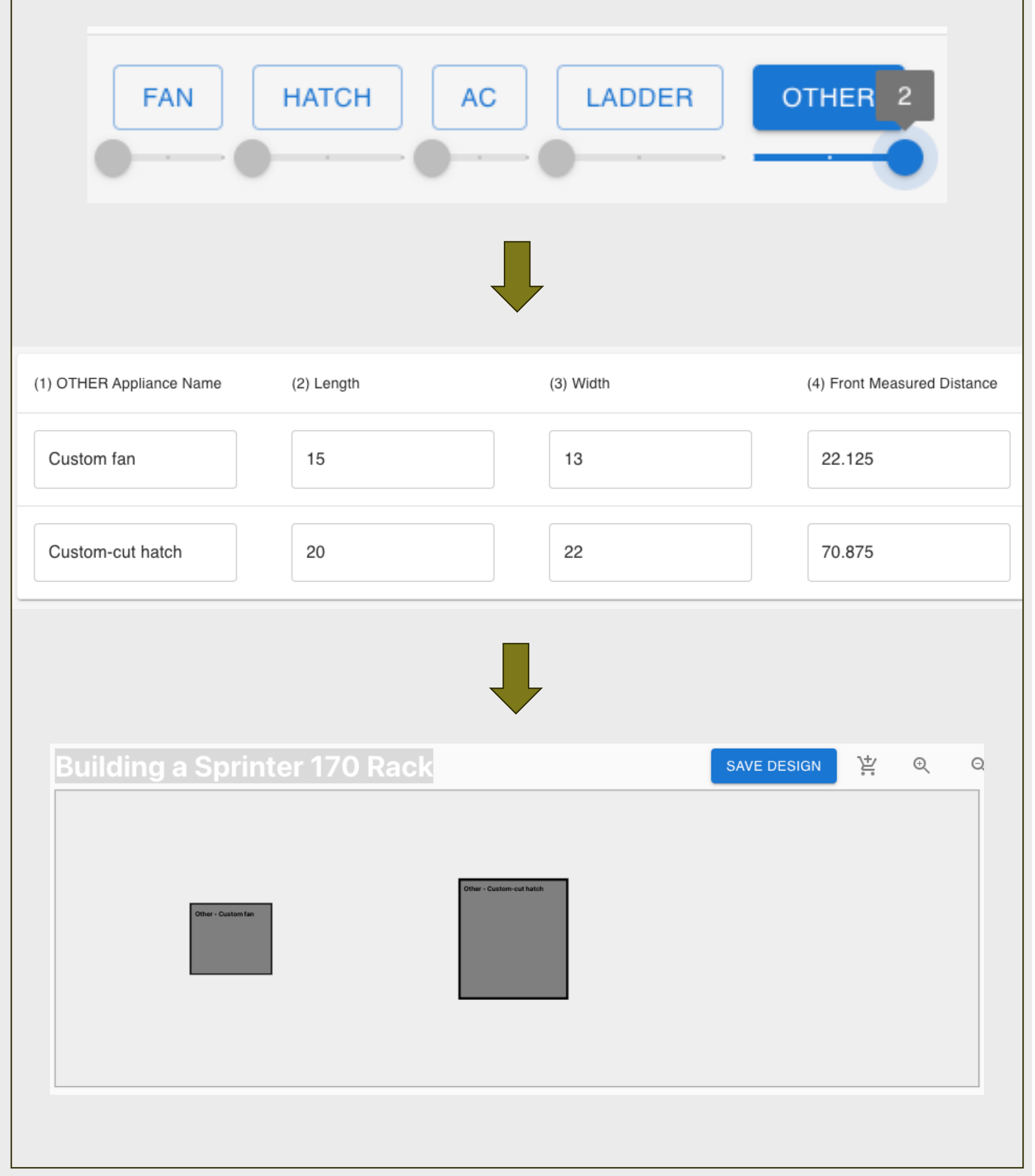
Much of the codebase contains information that is proprietary to Orion Van Gear, but here is a brief overview of our approach. Asynchronous loader functions fetch product data from the Shopify API needed for the routes on rendering. Product data is filtered based on the user's selections of van model and wheelbase. The data populates both a menu of options that the user may already have pre-installed, and a library of draggable products for the configurator.

If a user chooses to select a popular rack configuration, options are shown for their selection. If they choose to design their own, they are rooted to the design configurator, in which case URL parameters encode the user's chosen rack and any pre-installed appliances. The configurator parses roof rack size, number of crossbars, and excluded placement locations from a server-side JSON file.

Products are placed using the React-Grid-Layout library. As they are placed the equipment id, location, rotation, whether the equipment is static or not, are all built into a URL, which then gets put through the tinyurl API to shorten the extensive URL to a more user-friendly size after the user clicks to save their design. This link will allow users to return to the exact design at a later time.

When finished with their design, users can press the "add to cart" button which sends configured products to the Shopify Storefront.

User-specified Appliances



(1) OTHER Appliance Name	(2) Length	(3) Width	(4) Front Measured Distance
Custom fan	15	13	22.125
Custom-cut hatch	20	22	70.875

Oregon State University