

# ECOWEAR

*Rent And Resell Designer Fashion*

## **PREPARED BY:**

Sehar Ali

[seharfatimaali6@gmail.com](mailto:seharfatimaali6@gmail.com)

+923017755111

## **SUPERVISOR:**

PROFESSOR PRIYANKA SAMANTA

[SAMANTA@BROOKLYN.CUNY.EDU](mailto:SAMANTA@BROOKLYN.CUNY.EDU)

# CONTENTS

- Problem & idea
- Design and architecture
- Frontend implementation and UX
- Firebase, Stripe & hosting work
- Timeline & time log highlights



## INTRODUCTION

EcoWear is a simple, secure marketplace to rent or resell designer fashion, from bridal lehengas and groom sherwanis to luxury bags and heels, so you can enjoy high-end style without the one-day price tag. Built for affordability and sustainability, it's easy to list verified pieces, browse curated categories, compare rent vs. buy, and check real-time availability. We support a global audience with a country/currency selector, clear sizing, and safe checkout. Accessibility is built in, screen-reader labels, keyboard-friendly navigation, high-contrast and reduced-motion options, so everyone can participate. By extending the life of special-occasion outfits, we reduce waste and make designer style more attainable, one event at a time.

- ★ WEB DEVELOPMENT  
code and layout scripting

# PROBLEM: ONE-DAY OUTFITS, LONG-TERM WASTE

- DESIGNER BRIDAL OUTFITS, GROOM SHERWANIS, AND LUXURY ACCESSORIES ARE VERY EXPENSIVE BUT WORN ONCE.
- FAMILIES EITHER STORE THEM IN CLOSETS OR SELL THEM FOR A FRACTION OF THE PRICE.
- THERE'S NO SIMPLE, TRUSTED MARKETPLACE FOCUSED ON SOUTH ASIAN BRIDAL / FORMAL WEAR WITH RENT + RESALE IN ONE PLACE.
- I WANTED A PRODUCT THAT IS AFFORDABLE, SUSTAINABLE, AND INCLUSIVE (ACCESSIBILITY + GLOBAL USERS).

# EcoWear in One Line

- A web marketplace to rent or resell designer fashion (bridal lehengas, groom sherwanis, bags, heels).
- Clear Rent vs Buy pricing, curated categories, reviews, and wishlist.
- Built as a single-page web app with modern UI and a path to a Firebase + Stripe backend.

# My Role (Solo Project)

- Sole owner of UX, frontend code, and architecture decisions.
- Did all HTML/CSS/JavaScript, state management, and localStorage data model.
- Researched and experimented with Firebase Auth & Hosting and Stripe payments.
- Managed time log, documentation, and final demo + slide deck myself.

# SYSTEM ARCHITECTURE DIAGRAM

- Frontend: HTML, CSS, PHP, JavaScript and TypeScript
- Backend: Firebase(Authentication, Firestore, Hosting), Firebase Storage(images), Cloud Functions(webhooks).
- Payments: Stripe (test mode) for card-like flows
- Hosting/CI: Firebase Hosting + GitHub Actions (build & deploy)
- Analytics/Logs: Google Analytics (basic events), Firebase logs

# LOGICAL FLOW DIAGRAM

## EXAMPLE FLOW DIAGRAMS:

### Flow 1 — Buyer: Browse → Rent/Buy → Checkout

Login/Sign up → choose Women/Men → browse → open Product → toggle Rent / Buy (enter days if Rent) → Add to Cart → Checkout (shipping form) → Payment method → Order confirmation (email receipt)

### Flow 2 — Seller: List Item → Verification → Live

Login → Sell → choose Category → enter Title, Brand, Size, Condition, Price (USD), Rent/day, Availability dates → Upload images → Submit → Verification (quality/brand check) → Listing appears in selected category

### Flow 3 — Wishlist & Reviews

Login → browse → ❤️ add to Wishlist → open from Wishlist → Rent/Buy → after return/purchase → Leave star rating & review

# LOGICAL FLOW DIAGRAM

## EXAMPLE FLOW DIAGRAMS:

### Flow 4 — Country & Currency Selection

Click  → choose USA / EUR / UK / Pakistan / India / Bangladesh → prices convert (USD default) → size/labels adjust to region → continue shopping

### Flow 5 — Quick “Buy Now” (no rental)

Login → search/filters → Product (Buy selected) → Add to Cart / Buy Now → Checkout → Payment → Confirmation

# From Sketch to Screens

- Started with hand-sketched wireframes for the EcoWear homepage, sell flow, and messaging screens.
- Refined the sketches into a clearer paper prototype, iterating on layout: hero banner, category tiles, footer, and navigation.
- Used the sketches as a blueprint while coding the UI in HTML/CSS, keeping the same sections but improving spacing, typography, and button styles.
- Implemented a responsive card grid (rounded corners, soft shadows, pill badges) so the final screens still feel like the original sketches, but polished and clickable.

# UI MOCKUPS



## Rent Elegance. Resell Style

Designer bridal, traditional wear, bags and heels - at a fraction of retail

(Search designer, item, size...)

Search

Browse

Browse All Sections

All

Women

Men

Shop by Category



Designer Bridal Dresses

Explore curated designer  
bridal dresses



Designer Traditional  
wedding Dresses

Explore curated designer  
traditional wedding  
dresses



Designer Bags

Explore curated groom  
suits and sherwanis



Designer shoes/heels

Explore curated designer  
shoes/heels



Groom Suits and  
Sherwanis

Explore curated groom  
suits and sherwanis

chat with sales

→ 1(929) 283-8043

✉ sales@ecowear.com

Help

FAQ

Shipping and returns

Payment Methods

Sizing Guide

Careers

Connect  
Join our mailing list for updates

Email Address

Join

## List Your Designer Item

Upload details, choose category and Men / Women section, specify sizes, and set availability for rents.  
(Note : Designer traditional wedding dresses is for women only).

Title

Brand

Choose Category



Section

 Women Men

Price (USD)

mm / dd / yyyy

mm / dd / yyyy

Size (e.g., 38EU, M, 40R)

Rent per day (USD)

Images : Upload in next step • Verification required for designer labels • Secure payments & deposits for rentals

 chat with sales

→ 1(929) 233-8043

✉ sales@ecowear.com

Help

FAQ

Shipping and returns

Payment Methods

Sizing Guide

Careers

Connect

Join our mailing list for updates

Email Address

Join



## How Renting Works

1. Choose Men or Women section, then pick a category.
2. Select dates, pay a refundable deposit, then check out securely.
3. Receive, wear, return - ratings keep the community trusted.

chat with sales

1(929) 233-8043

sales@ecowear.com

[Help](#)

[FAQ](#)

[Shipping and returns](#)

[Payment Methods](#)

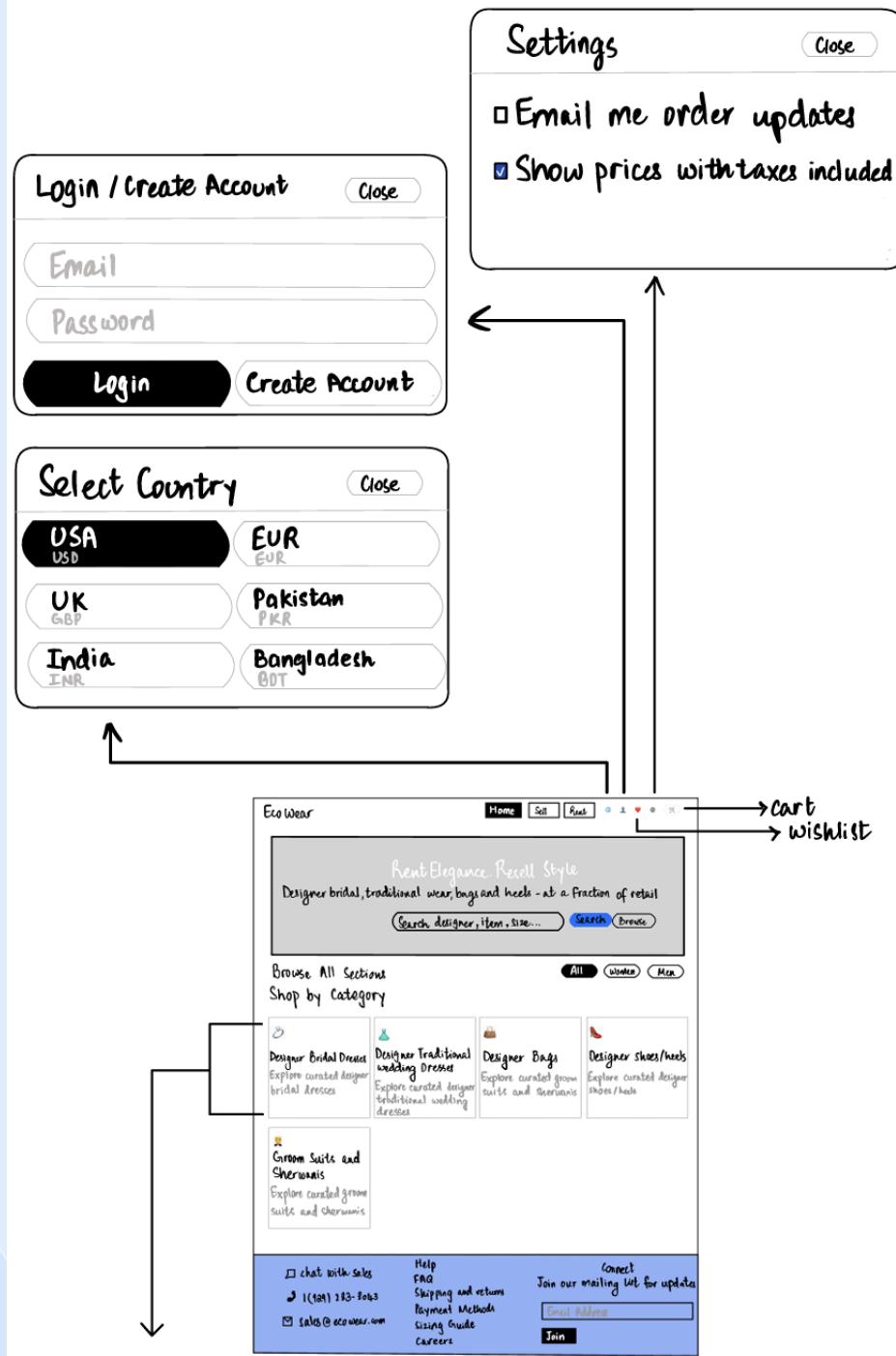
[Sizing Guide](#)

[Careers](#)

**Connect**  
Join our mailing list for updates

Email Address

[Join](#)



## Browse All Sections

### Designer Bridal Dresses

[All](#)[Women](#)[Men](#)[Filter Results...](#)

Picture

Women

Sabyasachi Bridal Lehenga (Maroon)

Buy: \$ 2,200      Rent/day: \$ 160.00

★★★★★ 4.9

Picture

Women

Hussain Rehar Bridal Lehenga

Buy: \$ 2,500      Rent/day: \$ 200.00

★★★★★ 4.5

chat with sales

→ 1(929) 233-8043

✉ sales@ecowear.com

Help

FAQ

Shipping and returns

Payment Methods

Sizing Guide

Careers

Connect

Join our mailing list for updates

Email Address

Join

# Sabyasachi Bridal Lehenga (Maroon)

**Close**

**Picture**

Women

Sabyasachi Bridal Lehenga (Maroon)

**Sub Total : \$1480.00**

Brand : Sabyasachi

Section : Women

Size : Small

★★★★★ 4.9

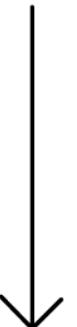
**Rent**

**Buy**

Days :  x \$ 160.00 /day

Refundable deposit : \$ 1000.00

**Add to Cart**



# Your Cart

[Close](#)

[Photo](#)

Sabyasachi Bridal Lehenga(Maroon)

Rent: 3 day(s)

\$1480.00

1 item

Total : \$ 1480

[Proceed To Checkout](#)



# Checkout

Close

## Shipping

Full Name

Address

City

Postal Code

Phone

Proceed To Payment

Fill in Payment information

Checkout

Order confirmation

# Learning Curve & New Technologies

- Frontend foundations from scratch
- Learned HTML for semantic structure (sections, nav, forms, inputs, buttons).
- Learned CSS to style EcoWear: responsive grid layouts, flexbox header/footer, pill buttons, card layout, hover states, and a consistent color system using CSS variables.
- Built the UI with JavaScript: DOM selection, event listeners, dynamic rendering of product cards, and updating the view when the URL hash changes.
- **Single-page application structure**
- Implemented a simple SPA router in JavaScript using location.hash (e.g. #/, #/category/heels, #/product/id).
- Created separate render functions: renderHome, renderCategory, renderProduct, renderSell, renderCart, etc.
- Centralized global state in a STATE object and persisted safe parts to localStorage (auth status, cart, wishlist, messages, orders).
- **Backend & DevOps – completely new to me**
- Backend concepts were new: I learned about authentication, hosting, and client–server separation from scratch.
- **Set up Firebase:**
- Configured a Firebase project, created a web app, and added the Firebase config to my frontend.
- Implemented Firebase Authentication (email/password) to sign users in and out, and wired it to UI states (e.g., “Sign up / Log in” vs “Sign out”, gating the Sell page, and showing “Welcome back!”).
- Used Firebase Hosting to deploy the SPA with the Firebase CLI (npx firebase-tools init hosting and deploy --only hosting).
- **Payments & Stripe exploration**
- Researched how Stripe Checkout / Payment Intents work: publishable vs secret keys, client vs server responsibilities.
- Experimented with how the frontend cart data would be sent to a Stripe backend endpoint, and what a checkout call would look like, even though the full production integration is still in progress.
- **Confidence gained**
- Before this project I had never built both frontend + backend pieces myself.
- Now I feel confident I can design, code, and deploy an end-to-end website solo: UI, routing, state management, auth, hosting, and a path toward real payments.

# ECOWEAR LANDING PAGE — SIGNED-OUT STATE

- When a user first visits **EcoWear**, they see the public landing page: hero banner, value proposition, and category cards (bridal, traditional, bags, heels, groom).
- This view is rendered by my vanilla-JS single-page app using a hash-based router (#/route).
- The header shows “Sign up / Log in” and Sell/Rent links, but protected features (Sell form, Messages, Account) all check a shared **STATE.signedIn** flag before rendering.
- All of this is just HTML + CSS + JavaScript — no frameworks — so I had to implement my own layout system (CSS grid/flex), card components, and navigation styling.



## Rent Elegance. Resell Style.

A curated marketplace to rent or resell designer wedding wear, bags, and heels—luxury made accessible.

High fashion, low footprint.

### Categories



**Designer Bridal Dress**  
Explore curated designer  
bridal dresses.



**Designer Traditional  
Wedding Dress**  
Explore curated designer  
traditional wedding  
dresses.



**Designer Bags**  
Explore curated luxury  
bags.



**Designer Shoes/Heels**  
Explore curated designer  
shoes & heels.



**Groom Suits &  
Sherwanis**  
Explore HSY & MNR  
sherwanis, suits,  
waistcoats.

### Designed for Every Occasion

Key features that make EcoWear inclusive, easy to use, and perfect for renting or reselling designer pieces.

## Designed for Every Occasion

Key features that make EcoWear inclusive, easy to use, and perfect for renting or reselling designer pieces.



### Curated & Authentic

- Verified designers
- Clear condition and honest photos



### Simple & Intuitive

- List or rent in minutes
- Clean, fast search & filters
- Live availability dates



### Secure & Trusted

- Safe checkout + protections
- Reviews and verified profiles
- Dispute support



### Sustainable & Circular

- Extend garment life
- High fashion, low footprint

[About Us](#)

[FAQs](#)

[Return and Exchange Policy](#)

[Privacy Policy](#)

[Shipping Policy](#)

[Contact Us](#)

**Join our mailing list for updates**

Email Address

Join

# EMAIL/PASSWORD SIGNUP MODAL WIRED TO FIREBASE AUTHENTICATION

- Clicking “Sign up / Log in” opens a modal I built in pure JS. It traps focus for accessibility and can be dismissed via the close button, background click, or Esc.
- The form collects email and password, validates the email format, and then calls my Firebase Authentication code to either log in or create an account using the Email/Password provider.
- On success, the client:
- Sets `STATE.signedIn = true`
- Stores a minimal profile (email) in `localStorage` as "eco\_state"
- Re-renders the home route and updates the header button text to “Sign out”
- This is where the **frontend and backend meet**: the browser is using the Firebase JS SDK to create a real user in my Firebase project, not just a fake local login.



# Rent Elegance. Resell Style.

A curated marketplace to rent or resell designer wedding wear, bags, and heels—luxury made accessible.  
High fashion, low footprint.

## Categories



### Designer Bridal Dress

Explore curated designer  
bridal dresses.



### Designer Groom Wear

Explore curated designer  
groom wear.



### Groom Suits & Sherwanis

Explore HSY & MNR  
sherwanis, suits,  
waistcoats.

## Login / Create Account

[Close](#)

.....

[Login](#)[Create Account](#)

## Designed for Every Occasion

Key features that make EcoWear inclusive, easy to use, and perfect for renting or reselling designer pieces.



### Curated & Authentic

- Verified designers
- Clear condition and honest photos



### Simple & Intuitive

- List or rent in minutes
- Clean, fast search & filters
- Live availability dates



### Secure & Trusted

- Safe checkout + protections
- Reviews and verified profiles
- Dispute support



### Sustainable & Circular

- Extend garment life
- High fashion, low footprint



Firebase

Project Overview



Project shortcuts

Authentication

Hosting

Product categories

Build

Run

Analytics

AI

Related development tools

Firebase Studio

Spark  
No-cost (\$0/month) Upgrade NEW

EcoWear ▾

# Authentication

[Users](#)[Sign-in method](#)[Templates](#)[Usage](#)[Settings](#)

Extensions



The following Authentication features will stop working when Firebase Dynamic Links shuts down soon: email link authentication for mobile apps, as well as Cordova OAuth support for web apps.



Search by email address, phone number, or user UID

Add user



Identifier	Providers	Created	Signed In	User UID
qwerty312@gmail.com		Dec 7, 2025	Dec 7, 2025	fYXOJaClKlOn7XGO3fvMjXE4...
finaltesting123@gmail....		Dec 7, 2025	Dec 7, 2025	VTf8cePJNvRu2TcdSee2zXfQ...
testing5thtime@gmail....		Dec 7, 2025	Dec 7, 2025	tQI5J15FmXclErdBdGG3iyVF5...
sehar@gmail.com		Dec 7, 2025	Dec 7, 2025	ZC7oPEA5o2ONyDcLGNlayh...
seharali31@gmail.com		Dec 7, 2025	Dec 7, 2025	3u9X1arDVBRz55sTi7MNMF1I...
test123@gmail.com		Dec 7, 2025	Dec 7, 2025	KFKvuuXSODS9jeDA7dJFEHL...
qwerty31@gmail.com		Dec 7, 2025	Dec 7, 2025	sVeTys5wUuPonF2SpCU3No6...
nishat123@gmail.com		Dec 7, 2025	Dec 7, 2025	d7wDlxIgyOPeeV6NyFrBxJRd...

Rows per page:

50



1 – 8 of 8



E



# FIREBASE AUTHENTICATION CONSOLE

- The previous screenshot is from the **Firebase Authentication dashboard** for my EcoWear project.
- Every test account I created from the EcoWear login modal appears here with:
  - Email identifier
  - Provider: **password** (email/password auth)
  - Timestamps for “Created” and “Signed in”
  - A unique Firebase UID
- This confirms my **frontend → Firebase** wiring is correct: when I submit the form on the website, the request goes to Firebase’s Auth API, which persists the user and issues an ID token back to the client.
- Even though my MVP doesn’t yet use secure server-side APIs, this gives me a real, production-ready identity layer I can plug into future features (orders per user, wishlists per user, etc.).

EcoWear

Home

Sell

Rent



Sign out

## Welcome back!

A curated marketplace to rent or resell designer wedding wear, bags, and heels—luxury made accessible.  
High fashion, low footprint.

### Categories



#### Designer Bridal Dress

Explore curated designer  
bridal dresses.



#### Designer Traditional Wedding Dress

Explore curated designer  
traditional wedding  
dresses.



#### Designer Bags

Explore curated luxury  
bags.



#### Designer Shoes/Heels

Explore curated designer  
shoes & heels.

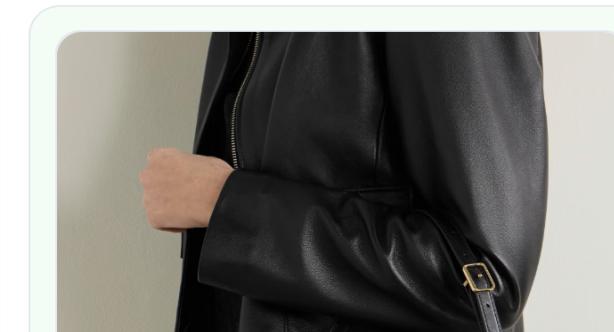
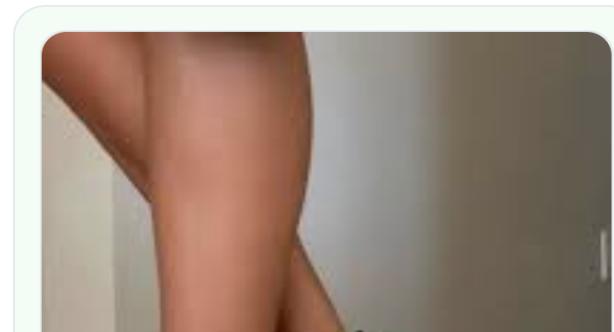


#### Groom Suits & Sherwanis

Explore HSY & MNR  
sherwanis, suits,  
waistcoats.

### Recommended for you

A few pieces to start your EcoWear wishlist.





## Recommended for you

A few pieces to start your EcoWear wishlist.



### Saint Laurent — Opyum Slingback

YSL heel logo, sleek slingback.

\$995.00

Rent \$60.00/day



### Tory Burch — Shoulder Bag

Polished shoulder bag that goes from daytime plans to wedding events effortlessly.

\$650.00

Rent \$45.00/day

# PERSONALIZED SIGNED-IN EXPERIENCE WITH CONDITIONAL UI

- After a successful login, the homepage re-renders with “Welcome back!” instead of the original marketing headline.

- In code, this is just:

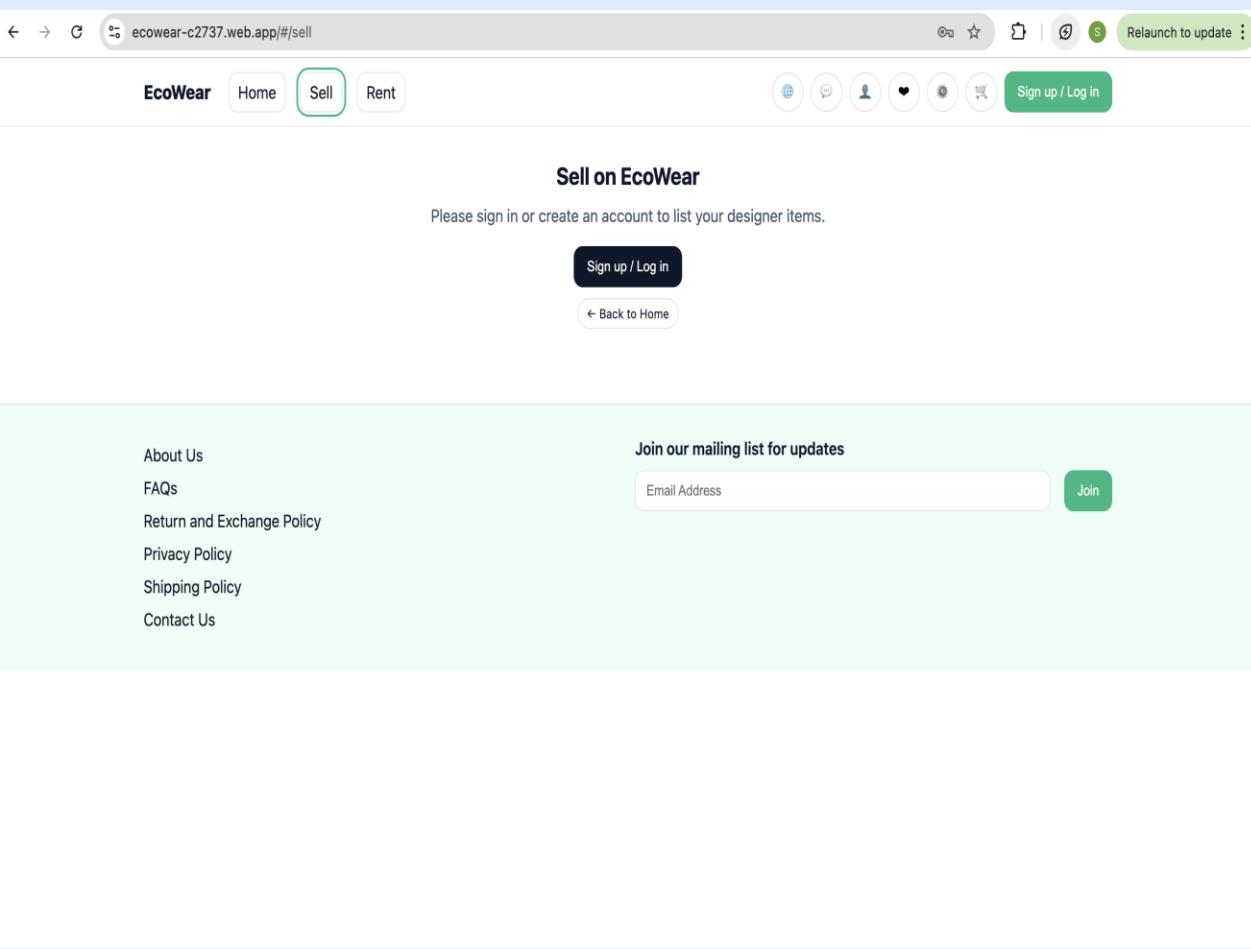
```
const heroTitle = STATE.signedIn ? "Welcome back!" : "Rent Elegance.  
Resell Style.;"
```

- I also toggle which section appears under Categories:

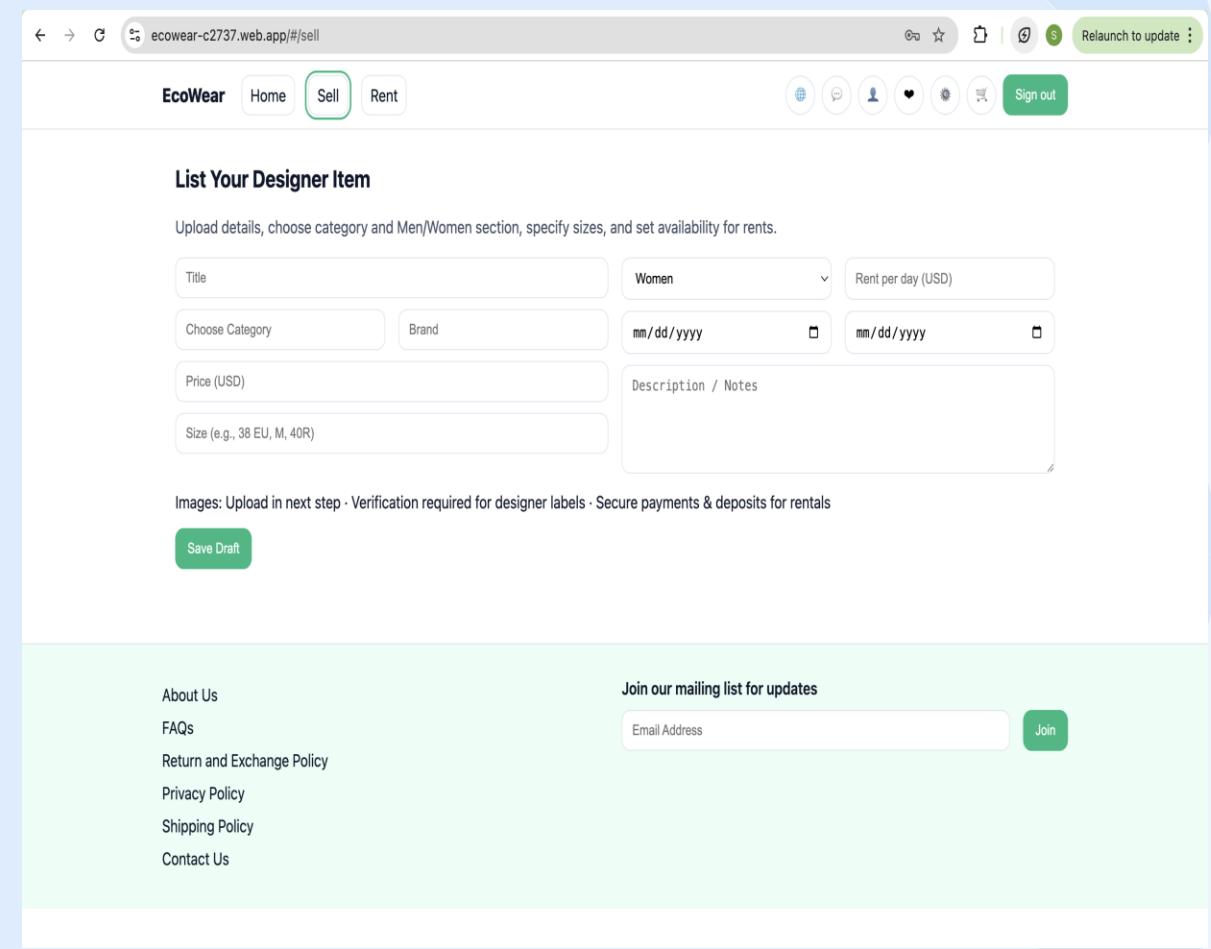
- Signed-out: “Designed for Every Occasion” feature cards

- Signed-in: “Recommended for you” section

- This switch is handled in my `renderHome()` function with a conditional block that checks `STATE.signedIn`. The UI is therefore **state-driven**, even though I’m not using React or Vue.



The screenshot shows the 'Sell' section of the EcoWear website. At the top, there's a navigation bar with 'EcoWear' and 'Relaunch to update' buttons. Below it, a secondary navigation bar includes 'Home', 'Sell' (which is highlighted with a green border), and 'Rent'. A row of icons follows, and a 'Sign up / Log in' button is on the right. The main content area is titled 'Sell on EcoWear' and contains a message: 'Please sign in or create an account to list your designer items.' It features a 'Sign up / Log in' button and a 'Back to Home' link. On the left, a sidebar lists links: 'About Us', 'FAQs', 'Return and Exchange Policy', 'Privacy Policy', 'Shipping Policy', and 'Contact Us'. On the right, there's a 'Join our mailing list for updates' form with an 'Email Address' input field and a 'Join' button.



This screenshot shows the 'List Your Designer Item' page. The top navigation bar and sidebar are identical to the previous screen. The main area is titled 'List Your Designer Item' and contains instructions: 'Upload details, choose category and Men/Women section, specify sizes, and set availability for rents.' It features several input fields: 'Title' (with a dropdown menu for 'Women'), 'Choose Category' (dropdown), 'Brand' (dropdown), two date pickers for 'mm/dd/yyyy', 'Price (USD)' (input field), 'Description / Notes' (text area), and 'Size (e.g., 38 EU, M, 40R)' (input field). A note at the bottom says 'Images: Upload in next step · Verification required for designer labels · Secure payments & deposits for rentals'. A 'Save Draft' button is located below the form. The right sidebar contains the same links as the first screenshot: 'About Us', 'FAQs', 'Return and Exchange Policy', 'Privacy Policy', 'Shipping Policy', and 'Contact Us'. A 'Join our mailing list for updates' form is also present on the right.

# SELL PAGE

## signed-out view

*Sell on EcoWear is gated behind authentication. If a visitor isn't logged in, the Sell tab shows a very simple guard screen that explains the feature and pushes them back into the login flow. This is all driven by a `STATE.signedIn` flag in my front-end state store, so the same URL conditionally renders either the guard screen or the full listing form.*

## signed-in view – listing form

*Once authenticated, the Sell page switches to a structured listing form. I collect the core listing data on the client: title, category, brand, sale price, rent-per-day, size, gender (Men/Women), available-from / available-to dates, and free-form notes. The layout mirrors a real marketplace listing flow and is wired to my central `STATE` object, so later I can plug this into Firestore to actually persist drafts.*

**EcoWear**

Home

Sell

Rent



Sign out

## How Renting Works

1. Choose Men or Women section, then pick a category.
2. Select dates, pay the rental fee (and a refundable deposit if required), and check out securely.
3. Receive, wear, return — ratings keep the community trusted.

[About Us](#)[FAQs](#)[Return and Exchange Policy](#)[Privacy Policy](#)[Shipping Policy](#)[Contact Us](#)

### Join our mailing list for updates

Join

# RENT PAGE

The Rent tab is a lightweight ‘How renting works’ page. Instead of throwing users into forms immediately, I use a three-step explainer component that reuses the same card layout as the rest of the site. It walks through choosing men/women + category, selecting dates and paying rental + refundable deposit, and then the receive–wear–return loop. Technically it’s powered by the same hash-based router as the rest of the app (`#/rent` route), so it feels like a proper page in the SPA even though it’s mostly content right now.



Relaunch to update :

EcoWear

Home

Sell

Rent



Sign out

## Designer Bridal Dress

[← Back to Home](#)



### Hussain Rehar — Bridal Couture '24

Iconic fuchsia bridal set, worn twice, perfect condition.

\$2,500.00

Rent \$250.00/day

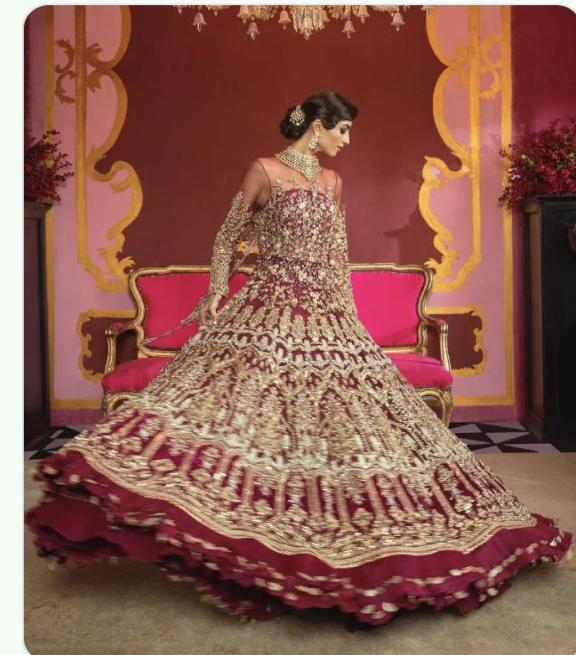


### Élan — Ivory Crystal Bridal

Ivory hand-embellished lehenga with crystal work; mint dupatta.

\$3,000.00

Rent \$300.00/day



### Sana Safinaz — Classic Red Bridal

Classic red with antique zardozi; lightweight skirt, rich look.

\$2,200.00

Rent \$220.00/day

## Designer Traditional Wedding Dress

← Back to Home



**Nomi Ansari — Mehndi Lehenga**

Vibrant multi-color lehenga, perfect for mehndi.

\$1,200.00

Rent \$130.00/day



**Faiza Saqlain — Pastel Pishwas**

Pastel pishwas with handwork; dreamy twirl.

\$950.00

Rent \$110.00/day



**Maria B — Tea Pink Formal**

Tea pink net with pearls; great for dholki/engagement.

\$800.00

Rent \$95.00/day



Sign out

## Designer Bags

[← Back to Home](#)



**Valentino Garavani — Rockstud Shoulder**  
Grain leather with signature studs.

\$1,900.00

Rent \$90.00/day



**Gucci — GG Marmont Matelassé**  
Matelassé chevron leather with GG flap.

\$1,700.00

Rent \$85.00/day



**Chanel — Classic Quilted Flap (Medium)**  
Timeless quilt with chain strap.

\$5,400.00

Rent \$150.00/day



← → ⌂ ecowear-c2737.web.app/#/category/heels

Relaunch to update :

EcoWear Home Sell Rent

Sign out

Designer Shoes/Heels

← Back to Home

A pair of light beige Valentino Rockstud pumps featuring multiple horizontal rows of gold studded straps across the toe and ankles.

**Valentino — Rockstud Pump 100**  
Pointed toe with signature studs.

\$980.00 Rent \$55.00/day

A woman's legs wearing Jimmy Choo Romy Glitter pumps, which are silver glittery pointed-toe pumps.

**Jimmy Choo — Romy Glitter**  
Classic glitter pump for receptions.

\$650.00 Rent \$40.00/day

A woman's legs wearing Saint Laurent Opyum Slingback pumps, which are black leather pumps with a prominent yellow YSL heel logo.

**Saint Laurent — Opyum Slingback**  
YSL heel logo, sleek slingback.

\$995.00 Rent \$60.00/day

About Us

FAQs

Return and Exchange Policy

Join our mailing list for updates

seharfatimaali6@gmail.com

Join

EcoWear

Home

Sell

Rent



Sign out

## Groom Suits & Sherwanis

← Back to Home



**HSY — Gold Sherwani**

Classic gold sherwani; regal, fitted silhouette.

\$1,400.00

Rent \$160.00/day



**MNR — Emerald Sherwani**

Emerald brocade; ideal for barat.

\$1,300.00

Rent \$150.00/day



**Michael Kors — Navy Suit**

Tailored two-piece; perfect for reception.

\$600.00

Rent \$70.00/day

## CATEGORY PAGES – CURATED SHOPPING FLOWS

- I designed five dedicated category pages: Designer Bridal Dress, Designer Traditional Wedding Dress, Designer Bags, Designer Shoes/Heels, and Groom Suits & Sherwanis. Each page shows a focused “rack” of curated items with large imagery, title, short blurb, price, and rent-per-day badge so users can compare at a glance.
- All category pages share the same reusable card layout: a JS **CATALOG** object stores items with IDs, names, prices, rent rates, blurbs, and image paths. The router reads the URL hash (e.g., `#/category/bridal`, `#/category/bags`) and filters the catalog by category to render the correct set of cards dynamically.
- The layout is built with flexbox/grid CSS, consistent padding, and the light mint background from the home page, plus a “Back to Home” button in the header so users never get stuck. This keeps the UX consistent while letting each category feel like its own curated mini-boutique.

EcoWear

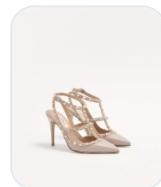
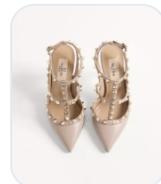
Home

Sell

Rent



Sign out



# Valentino — Rockstud Pump 100

**\$980.00**

In stock · prices include tax

## Description

Pointed toe with signature studs.

[Add to Cart](#)[Buy Now](#)[★ Add to Wishlist](#)[Message Seller](#)

## Rent this item

Start date

mm/dd/yyyy

End date

mm/dd/yyyy

No blackout dates for this item

Rate: \$55.00/day

1

Total: \$55.00

+\$1,000.00 refundable deposit

[Add Rental to Cart](#)

## Seller Information

Sold by: Kiran

Email: kiran.j@example.com

**Reviews (4.7)**

EcoWear Home Sell Rent Sign out Relaunch to update

**Valentino — Rockstud Pump 100**

**\$980.00**  
In stock · prices include tax

**Description**  
Pointed toe with signature studs.

Add to Cart Buy Now Add to Wishlist Message Seller

**Rent this item**

Start date: 12/09/2025 End date: mm/dd/yyyy December 2025

Rate: \$55.00/day 1  
+\$1,000.00 refundable deposit

**Seller Information**  
Sold by: Kiran Email: kiran.j@example.com

Reviews (4.7)

EcoWear Home Sell Rent Sign out Relaunch to update

Click to go back, hold to see history

**Cart**

Item	Quantity	Price
Valentino — Rockstud Pump 100 (Rent x 4d)	1	\$220.00
<b>Subtotal</b>		\$220.00
<b>Refundable deposit (rentals)</b>		\$1,000.00
Deposit is added at payment and refunded after acceptable return.		

Checkout Clear Cart

About Us FAQs Return and Exchange Policy Privacy Policy Shipping Policy Contact Us

Join our mailing list for updates Email Address Join

EcoWear Home Sell Rent Sign out Relaunch to update

**Order Confirmed**

Order #229232 · 12/8/2025, 6:16:34 AM

Payment method: CARD

Item	Price
Valentino — Rockstud Pump 100 (Rent x 4d)	\$220.00
<b>Subtotal</b>	\$220.00
Tax (2%)	\$4.40
Shipping	\$10.00
Refundable deposit (rentals)	\$1,000.00
<b>Total</b>	<b>\$1,234.40</b>

Home Messages View in Account

About Us FAQs Return and Exchange Policy Privacy Policy Shipping Policy Contact Us

Join our mailing list for updates seharfatimaali6@gmail.com Join

# CHECKOUT & RENTAL FLOW

- Product page – choose rental dates & price is calculated
  - On the product detail screen (e.g., *Valentino — Rockstud Pump 100*), the user selects a **start** and **end** date with HTML date pickers.
  - My JavaScript calculates the **number of rental days**, multiplies by the daily rate, and adds a **refundable security deposit**.
  - The UI updates in real time (e.g., “Rent × 4d”) and the user can add the item to the cart with one click.
- Cart – review rental summary
  - The Cart page shows each line item with its rental duration, subtotal, and deposit.
  - Cart data is stored on the front end, so users can navigate around the site without losing their selections.
  - From here the user can clear the cart, continue shopping, or go straight to Checkout.
- Checkout – capture shipping details
  - At Checkout, I collect the customer’s name, address, and phone number in a validated form.
  - An Order Summary panel on the right shows a live breakdown: item total, shipping, and refundable deposit.
- Payment step & order confirmation
  - The “Continue to Payment” action takes the user to a confirmation flow that is structured for Stripe integration (order object, totals, and metadata are ready to be sent to a payment API).
  - For the demo, I simulate a successful card payment and generate an **Order Confirmed** screen with order ID, timestamp, payment method, and full price breakdown (subtotal, tax, shipping, deposit).
  - This page is the final step in the rental journey and is designed with a real Stripe checkout session.

# Project Timeline & Milestones

## August – Framing the project

- Got oriented with the course, met with professor, and locked in EcoWear as my capstone idea.
- Finalized the concept (luxury bridal/rental marketplace), wrote user stories and scenarios, and did hand-drawn low-fidelity prototypes of the main screens (home, categories, product, cart, auth).

## September – Design + core frontend foundations

- Turned the initial ideas into a concrete outline and schedule, especially once I decided to run the project solo.
- Deepened my frontend skills: refreshed HTML, got more comfortable with semantic structure and forms, and studied TypeScript patterns.
- Designed full screen flows for Sell, Wishlist, Checkout, and drew data-flow and ERD sketches for how rentals, users, and orders would work.
- Started building the first coded version: category grids, product cards, and basic navigation, and learned how much work it is to own both design and implementation myself.

## October – Turning it into a single-page app + documentation

- Reorganized everything into a single-page app with hash routing and one central `app.js`, plus shared `styles.css` and a single HTML shell.
- Built out the full flow: Home → Category → Product → Cart → Checkout → Confirmation, with cart math, date picker for rentals, shipping, wishlist, and messages all handled in JavaScript state.
- Iterated on UX details: availability validation, better messages UI, accessibility tweaks, and consistent spacing/typography.
- Treated it like a real product: pushed to GitHub, added README, license, issues, project board, architecture diagrams, sitemaps, and created the first draft slide deck and demo GIF.

## November – Backend phase: Firebase + auth + data

- Shifted focus to the backend: researched options, then created a Firebase project, enabled email/password auth, and wired the SDK into my existing SPA.
- Replaced my fake “signed in” flag with real Firebase Authentication, wiring login and sign-up into my custom modal and updating global state based on `onAuthStateChanged`.
- Started writing orders to Firestore so checkout isn’t just local state anymore.
- Updated the UX to react to auth: added the signed-out Sell gate, the signed-in Sell form, and swapped the homepage feature cards to a “Recommended for you” section for logged-in users.
- Spent time hardening this: end-to-end testing of navigation, auth flows, and conditional UI, plus more research on Stripe and hosting to understand security constraints.

## December – Payments experiments, deployment, and wrap-up

- Experimented with integrating Stripe with my cart totals and learned in practice why secure payments require a backend (secret keys, webhooks, PCI concerns).
- Set up Firebase Hosting with the CLI, deployed the live version of EcoWear, and debugged asset paths and caching so all the real product images (bags, heels, bridal, groom) loaded correctly.
- Did a final testing pass on the hosted site: routing, cart/checkout, and the Stripe sandbox limitations.
- Closed out with documentation: updated the time log, wrote a feature/architecture summary, captured final screenshots of the live app, and recorded the presentation + demo video walking through auth, browsing, product pages, cart, and checkout.

# DATA SOURCES

# DATA SOURCES

- **User-Generated Content:** Sellers upload and list designer items (bridal lehengas, women's traditional wear, groom sherwanis/waistcoats/suits, luxury bags, heels) for rent or resale with photos and availability dates.
- **Manual Input / Admin Uploads:** Admin curates brand catalog (HSY, MNR, LV, MK, DKNY...), category taxonomy, size charts, verification flags, and takedown reasons (counterfeit/poor quality).
- **External APIs:** Stripe (payments, test mode now), currency-conversion API for country/currency switcher (USD default), optional address autocomplete (e.g., Places-style service) for shipping forms.
- **Platform Stores:** Firebase (Auth, Firestore for documents, Storage for images, Cloud Functions for webhooks/logs) as the primary data backbone.

# NATURE OF THE DATA SETS

- Product listings: title, designerBrand, category (Women: Bridal/Traditional; Men: Groom; Accessories: Bags/Heels), condition, size (with measurements), color, location, priceUSD, rentPerDayUSD, availabilityStart, availabilityEnd, securityDeposit (optional), images[], sellerId, createdAt, status (draft/active flagged).
- User profiles: uid, name, email, role (buyer/seller), country, currencyPreference, addresses[], wishlist[], orders[] (references), verificationStatus, createdAt/updatedAt. Passwords are handled by Auth and not stored in Firestore.
- Orders / rentals: orderId, buyerId, sellerId, items[] (listingId, rent or buy, quantity), rentalDates (start/end), totals (items, fees, deposit, tax, shipping), shippingAddress, fulfillmentStatus, createdAt. Payments: store Stripe paymentIntentId and outcome only; no raw card data.
- Reviews and ratings: itemId, authorId, stars (1–5), comment, optional photos, createdAt, moderationStatus; derived ratingAverage stored on listing.
- Cart / session data: temporary cart lines, selected currency at calculation time, discounts (future).
- Search/filter metadata: brand and category indexes, price-range buckets, size tags, popularity counters; used to speed queries and sorting.
- Audit / moderation / analytics: login and listing actions, report reasons, webhook logs, anonymized event metrics (views, add-to-cart, checkout starts).

# USER CASES

# USER STORY 1

User Story: As a buyer, I want to rent a designer bridal lehenga for a specific weekend so I don't have to buy an expensive outfit for one day.

Scenario: The user logs in, goes to Women → Bridal, filters by HSY and size M, opens a product, switches to Rent, selects Oct 18–20, adds to cart, enters shipping, and pays.

Outputs: A rental order is created, the item's calendar blocks Oct 18–20, and the user sees a confirmation page and receives a confirmation email.

# USER STORY 3

As a buyer, I want to purchase a luxury bag outright instead of renting.

Scenario: The user logs in, goes to Accessories → Bags, filters brand Louis Vuitton, opens a product, leaves it on Buy, adds to cart, checks out, and pays (prices shown in the selected currency with USD as base).

Outputs: An order is created with fulfillment status “processing,” the listing is marked sold or quantity reduced, and a receipt is shown and emailed.

# USER STORY 2

As a seller, I want to list my Mohsin Naveed Ranjha groom sherwani so I can earn by renting or selling it.

Scenario: The user logs in, opens Sell, chooses Men → Groom, fills title, size 40, condition “like new,” price and rent/day, sets availability dates, uploads photos, and submits.

Outputs: The listing is saved with status “pending verification,” then appears in Men → Groom and in the seller’s dashboard once approved.

# USER STORY 4

As a user, I want to save items to a wishlist and be notified when prices or dates change.

Scenario: The user taps the heart icon on a DKNY suit to add it to Wishlist, later returns to Wishlist to view it, and receives an in-app badge when the seller drops the price or opens new dates.

Outputs: The item is stored in the user's wishlist, notifications appear when changes occur, and the user can rent or buy directly from the wishlist item page.

# USER STORY 5

As a recent renter, I want to leave a rating and short review after my rental ends.

Scenario: The user opens Orders, selects the completed rental, chooses “Leave review,” sets 4 stars, writes a brief comment, and submits.

Outputs: The review is saved, the product’s average rating updates, and the review appears on the product page (subject to moderation if flagged).

# Impact & Value of EcoWear

- **Problem Space**
- Designer occasion-wear is extremely expensive, worn a few times, then sits in closets.
- There's no simple, trusted way to rent verified designer pieces or safely resell them.
- Users need something that feels like a real marketplace, not a random Facebook group.
- **How EcoWear Adds Value**
- Turns idle high-end pieces into income-generating assets via rentals and resale.
- Makes luxury more accessible: users can rent a \$3,000 bridal lehenga for a single event at rental price instead of full retail.
- Built-in refundable deposits + clear policies reduce risk for both renters and owners.
- Clean UX (categories, recommendations, clear checkout flow) lowers friction so users can go from browsing → booking in a few clicks.
- **My Role & Contribution**
- Solo designer + developer: hand-sketched the marketplace, then implemented the full frontend (HTML/CSS/JS) and Firebase-backed authentication.
- Connected the UI to real data (catalog, product pages, cart, checkout), making it a working web application, not just mockups.
- Deployed on Firebase Hosting so EcoWear can be demoed, user-tested, and extended into a production marketplace later.

- GITHUB REPOSITORY: <https://github.com/sali-31/EcoWear>
- PROJECT MANAGEMENT BOARD: <https://github.com/users/sali-31/projects/2>
- HOSTED WEB APP: <https://ecowear-c2737.web.app>

# Thank you