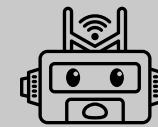


# Huehive

ORGANIZE • RECOMMEND • STYLE

MADE BY -  
Swasti Negi  
Harsh Kumar  
Dhruv Yadav  
Ayush Agarwal



# Huehive : Revolutionizing the Way You Dress



## Introduction

- HueHive is a Next.js-powered virtual wardrobe.
- Helps users organize clothes, manage fashion digitally, and get AI-powered outfit recommendations.
- Designed for modern lifestyles, blending fashion + technology.

## Problem Statement

- People forget what's in their wardrobe.
- Hard to mix & match outfits efficiently.
- Lack of personalized styling advice without stylists.
- Need for a centralized digital wardrobe with smart suggestions.

## Our Solution HueHive

- Digitizes your wardrobe – upload & categorize clothes.
- AI-powered recommendations for stylish outfits.
- Wardrobe statistics & usage tracking.
- Personal styling insights using color theory + AI analysis.
- Simplifies fashion decisions, saves time, boosts confidence.

# User Authentication

## - User Authentication

Purpose: Gate the app with a simple login/registration flow and set up a “session” for personalized features.

### UI & Flow

- Login: Username + password form with primary CTA.
- Register link: Takes users to a sign-up flow (same fields + confirm password in a typical setup).
- Feedback: Disabled button/spinner during submit; error text on invalid credentials.

### -Data & Validations

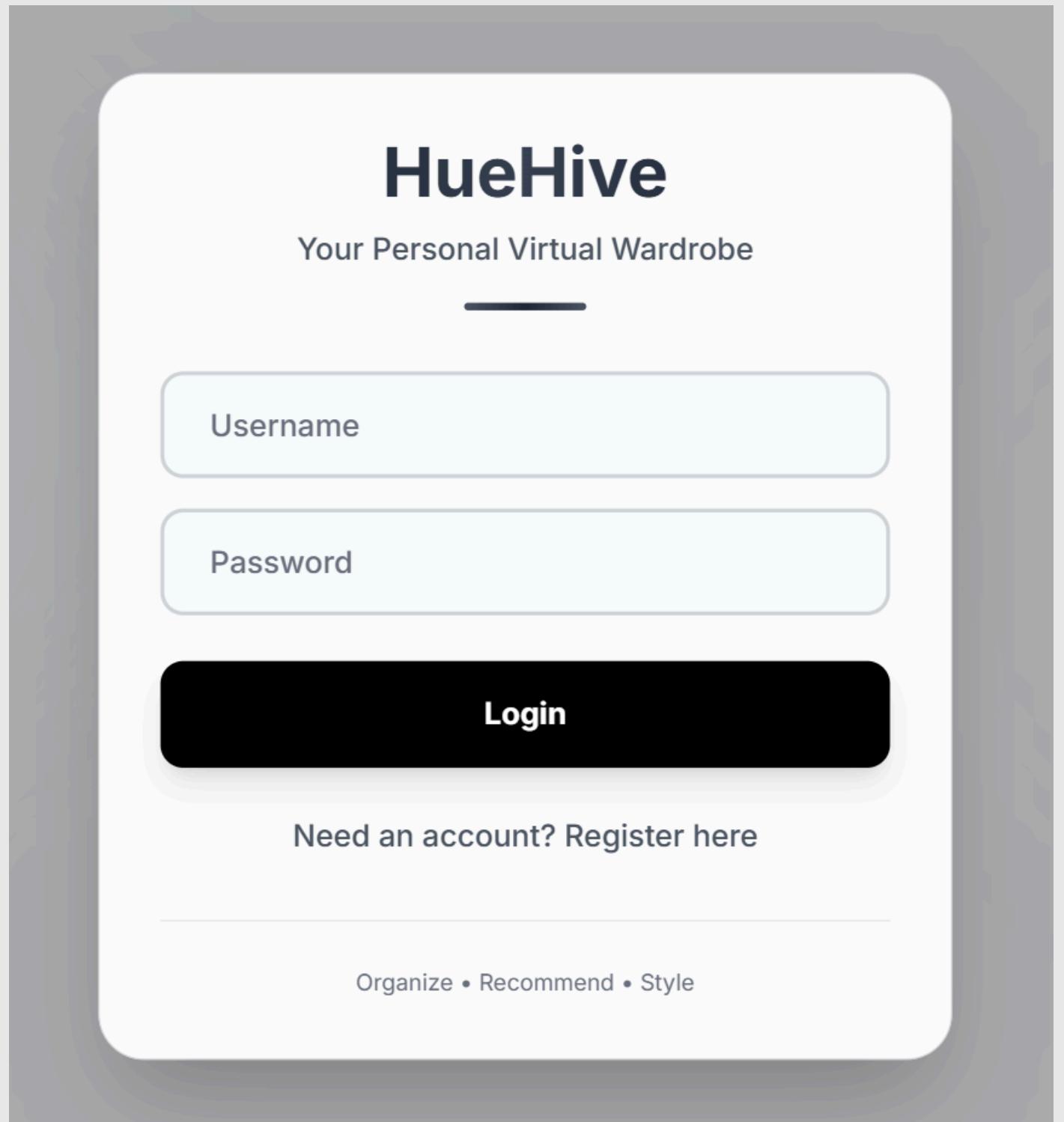
- Required fields, min-length password, basic username checks.
- In the static build, the UI is backend-agnostic: it’s prepped to plug into any auth service (Firebase/Auth0/NextAuth) or a custom API.
- Demo mode may simulate session via in-memory or local storage (so you can show the flow without a server).

### -Behind the Scenes (ready for real auth)

- On production, wire the form to an /api/login or provider SDK.
- Store a short-lived token; protect “app” routes; show user-scoped data (profile, wardrobe).

### -Demo Tips

- Show fast login, then how the sidebar unlocks Profile, Upload, Wardrobe, and AI actions.



# YOUR PROFILE

Purpose: Capture preferences and identity so recommendations feel personal.

## UI & Flow

- Profile avatar, name / age / birth date fields.
- Right-side calendar shows a login streak (gamified engagement).
- Save Changes CTA with success toast.

## Data & Validations

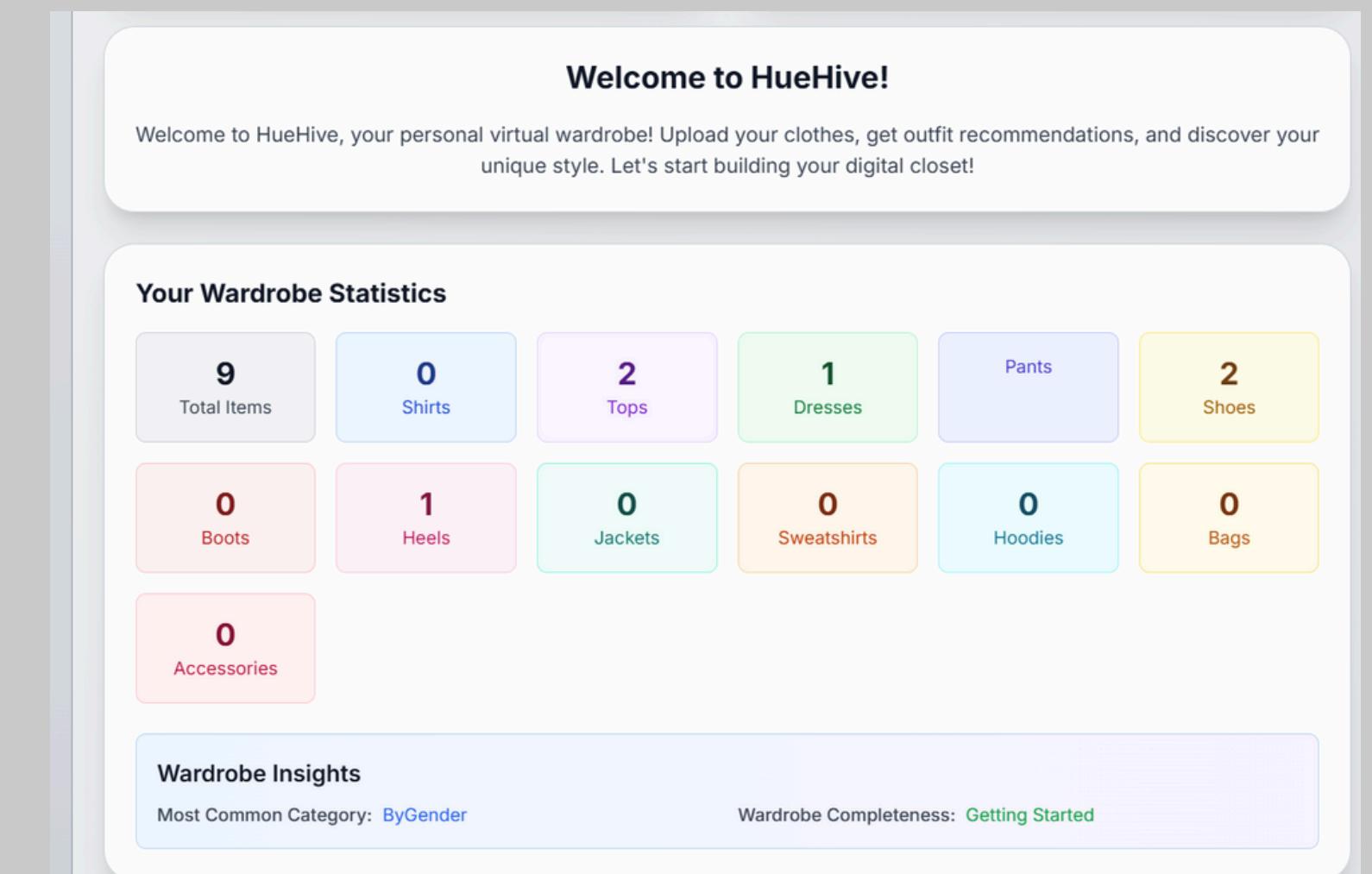
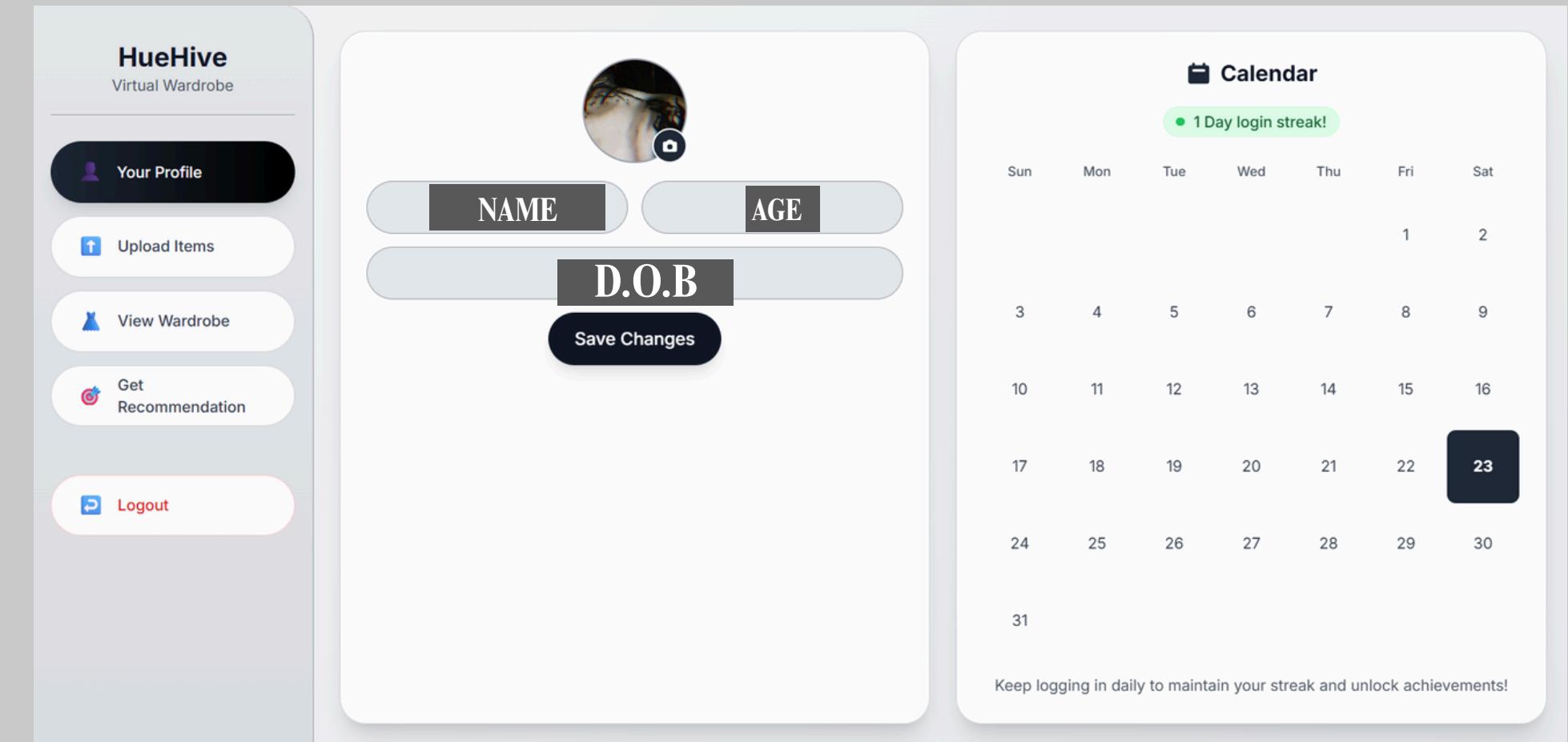
- Sanitized text inputs; sensible ranges (e.g., age); date validation.
- Preferences (like style or gender) can be saved and reused by the AI recommender.

## Behind the Scenes

- In static mode: values persist in local storage.
- In full stack: persist to DB (e.g., users table: id, name, age, dob, preferences), hydrate UI on load.

## -Demo Tips

- Update a field → Save → Refresh → Show it persists.
- Mention that these preferences directly influence AI suggestions.



# Upload Items

-Purpose: Build the digital closet by adding clothing pieces with metadata.

## UI & Flow

- Fields: Type (Top, Pants, Shoes...), Gender, Color (supports multi-color like red/blue), Image upload/preview.
- Actions: Add to Wardrobe and Clear Form.
- Drop area with dashed border for a friendly upload experience.

## Data & Validations

- Required: type + image; normalize color input (lowercase, trimmed).
- Auto-generate a unique item id; store file URL/encoded image (static mode) or blob storage path (prod).

## Behind the Scenes

- Static: items stored in local storage with schema
- { id, type, gender, color, season?, style?, imageUrl }.
- Full stack: send to API -> persist to DB/object storage (S3/GCS/Supabase Storage).

## Demo Tips

- Upload one top, one bottom, one shoe with different colors → immediately visible in Wardrobe.

Let's Add some clothing Item into Your Virtual Wardrobe :

Type

Shoes

Gender

Women's

Color (For multi-color items, use formats like "brown-white" or "red/blue")

black

Upload Image



Click to change image

+ Add to Wardrobe

Clear Form

# Virtual Wardrobe

Purpose: Browse, filter, and manage all items visually.

## UI & Flow

- Grid cards with image, title (type), color chips, badges (gender/season).
- Filters (top bar or sidebar): by category, season, color, gender.
- Per-item delete action (trash icon) to remove from closet.
- “Showing: N / total” counter so users see scope.

## Data & Logic

- Derived stats for dashboard (counts per category).
- Filters applied client-side; fast and responsive.

## Behind the Scenes

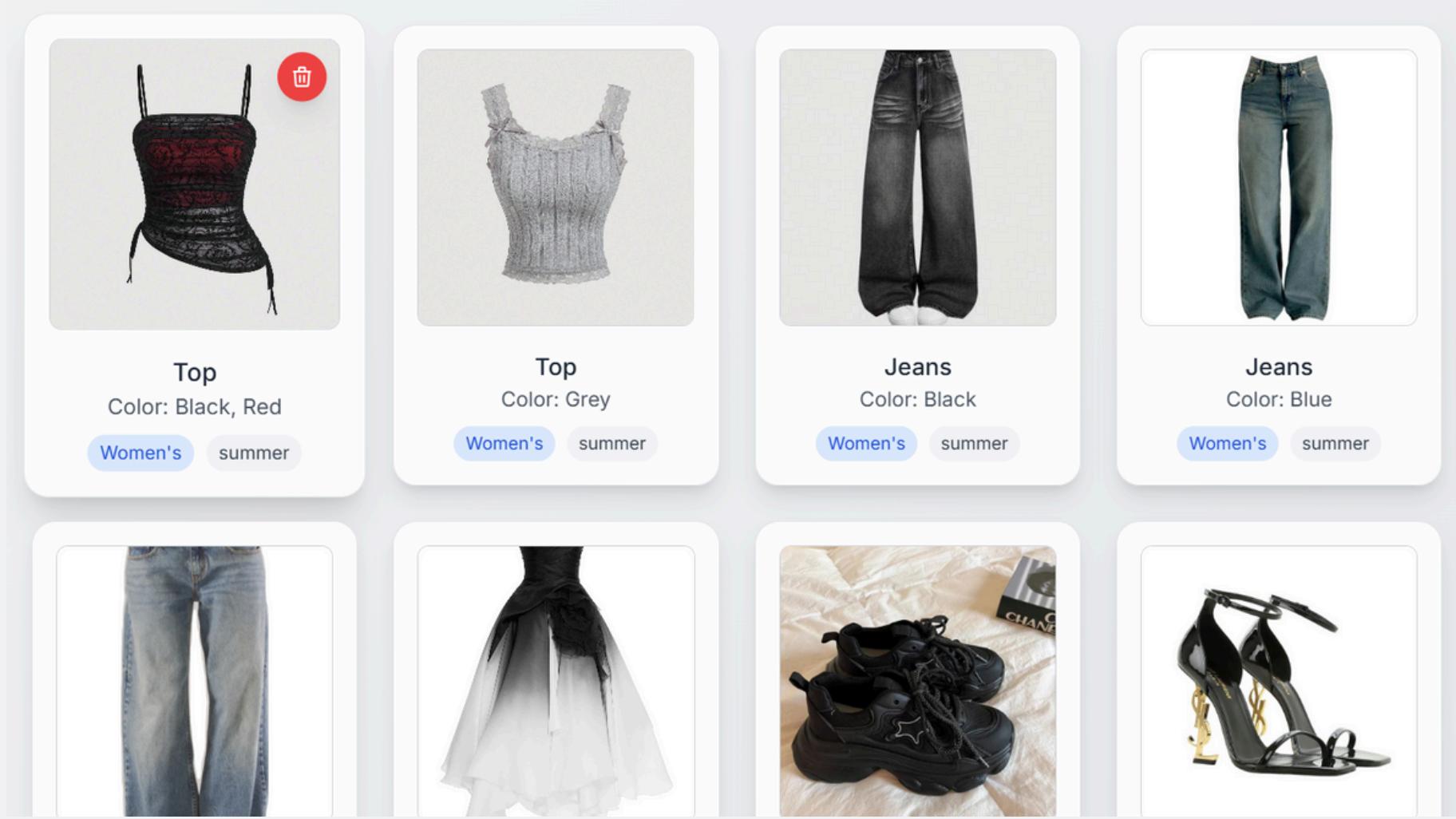
- Static: Read items from local storage; compute stats on the fly.
- Full stack: Query by user id; server or client filtering; support pagination if wardrobe grows large.

## -Demo Tips

- Toggle filters (e.g., “All Items → Women’s → Summer”) to show responsiveness.
- Delete an item and watch counts update live.

Closet Goals ? You Invented them .

All Items ▾ Showing: 9 / 9 items



# GET RECOMMENDATION

Purpose: Generate smart outfits + explain why they work.

## UI & Flow

- Inputs: Gender, Season, Occasion, Style + Color Preference.
- Press Get AI Recommendations → show 3 outfit cards with confidence scores (e.g., 88–90%).
- Each outfit shows components (Top/Bottom/Shoes) + Why this works bullets.
- Detailed Analysis modal:
  - Score breakdown: Color Harmony, Style Coherence, Season Fit, Occasion Fit
  - AI Reasoning paragraph
  - Style Tips + Wardrobe Enhancement Suggestions
- Save Outfit and Generate More Options buttons for iteration.

## Scoring & Heuristics (client-side in static build)

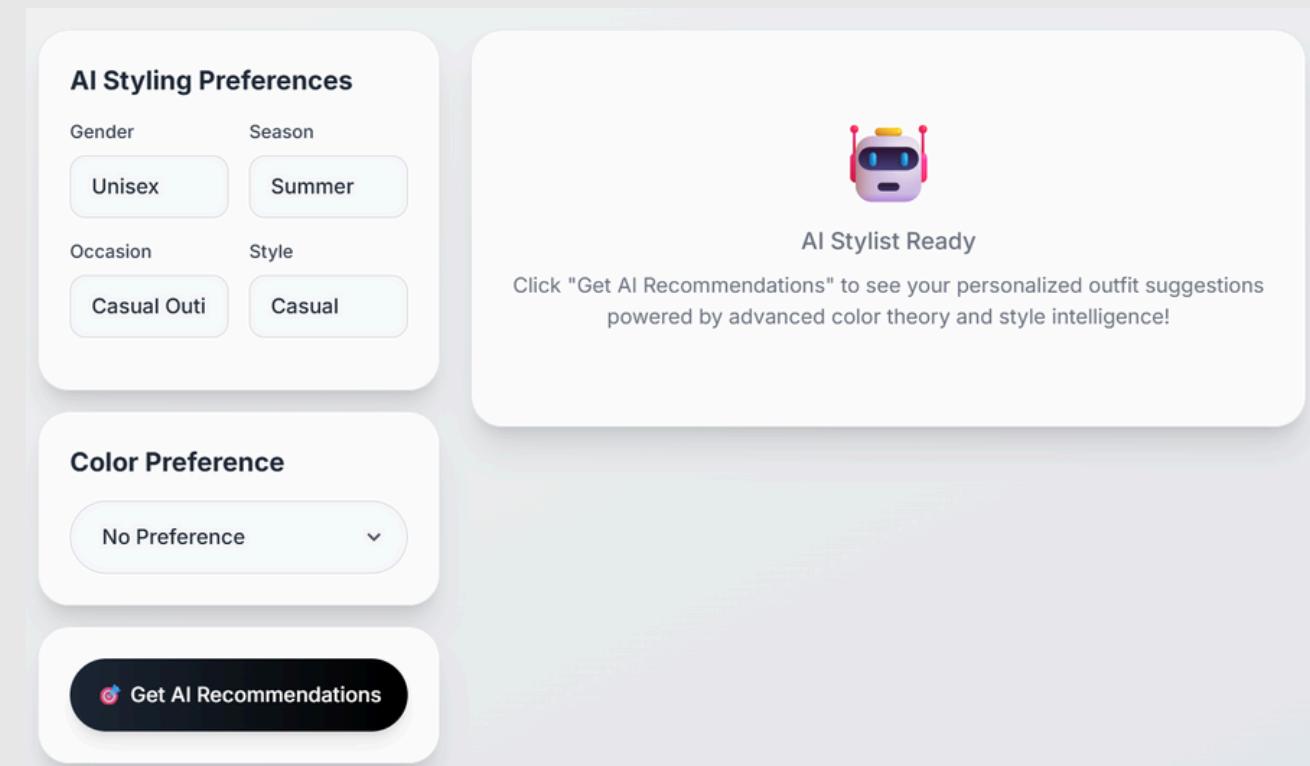
- Color Harmony: complementary/analogous/triadic checks on item colors.
- Style Coherence: style tags across items (e.g., casual x casual).
- Season Fit: season tags vs current preference.
- Occasion Fit: match occasion (date, office, party).
- Weighted to produce the final confidence score.

## Behind the Scenes

- Static: Deterministic heuristics produce stable, fast results entirely on the client.
- With AI API (future/alt): send wardrobe + preferences to an LLM/ML endpoint (OpenAI/HF) for semantic matching, then render the same UI.

## Demo Tips

- Pick Female / Summer / Date / Casual (like your screenshots) → run once → open Detailed Analysis → save outfit.
- Hit Generate More Options to show variety while staying coherent.



**AI Styling Preferences**

Gender: Female, Season: Summer

Occasion: Date, Style: Casual

**Color Preference**

No Preference

**Get AI Recommendations**

Save This Outfit

Or

Generate New Options

**AI-Powered Style Intelligence**

**Your Outfit Recommendations:**

Found 3 perfect outfits for you!

**Outfit 1** (89%): Color 100%, Style 77%, Season 80%. Items: Top (Grey), Jeans (Blue), Shoes (Black). Why this works: Well-balanced top and bottom combination, Excellent color harmony creates a cohesive look.

**Outfit 2** (88%): Color 100%, Style 70%, Season 85%. Items: Dress (Black, White), Jeans (Blue), Heels (Black). Why this works: Well-balanced top and bottom combination, Excellent color harmony creates a cohesive look.

**Outfit 3** (88%): Color 97%, Style 77%, Season 80%. Items: Top (Black, Red), Jeans (Blue), Heels (Black). Why this works: Well-balanced top and bottom combination, Excellent color harmony creates a cohesive look.

**Generate More Outfits**

## Detailed Analysis

**Detailed Outfit Analysis**

**Complete Outfit**

**Score Breakdown**

- Color Harmony: 100%
- Style Coherence: 77%
- Season Appropriateness: 80%
- Occasion Fit: 100%

**AI Analysis & Reasoning**

- Well-balanced top and bottom combination
- Excellent color harmony creates a cohesive look
- Stylish and perfect for a special occasion

**Style Tips & Suggestions**

- Color Theory: This outfit uses complementary color harmony for visual appeal
- Proportions: The silhouette creates a balanced look suitable for your selected occasion
- Versatility: These pieces can be mixed with other items in your wardrobe
- Styling Tip: Consider adding accessories like a watch, necklace, or bag to elevate the look
- Perfect Match: This is an excellent outfit combination that showcases great style sense!

**Overall Score: 89%**

**Save This Outfit**

**Close Analysis**

## Advanced AI Styling Insights

Confidence Score: 90%

**Color Harmony Analysis**

Triadic

**AI Reasoning**

This casual outfit is perfect for date night. The combination of black-red top, black jeans, black-white dress creates a cohesive look. To enhance this outfit, consider adding A statement piece to elevate your outfits.

**Style Tips**

- For date night, aim for medium-high formality level
- summer styling: layer appropriately and choose season-appropriate fabrics

**Wardrobe Enhancement Suggestions**

To create even better outfits, consider adding these items to your wardrobe:

**Statement Accessory** (medium)

A statement piece to elevate your outfits  
Alternatives: bold necklace, colorful scarf...

**Outfit Saved Successfully!**

AI Confidence: 89%

Color Harmony: 100%, Style Match: 77%, Season Fit: 80%, Occasion: 100%

**Your Complete Outfit**

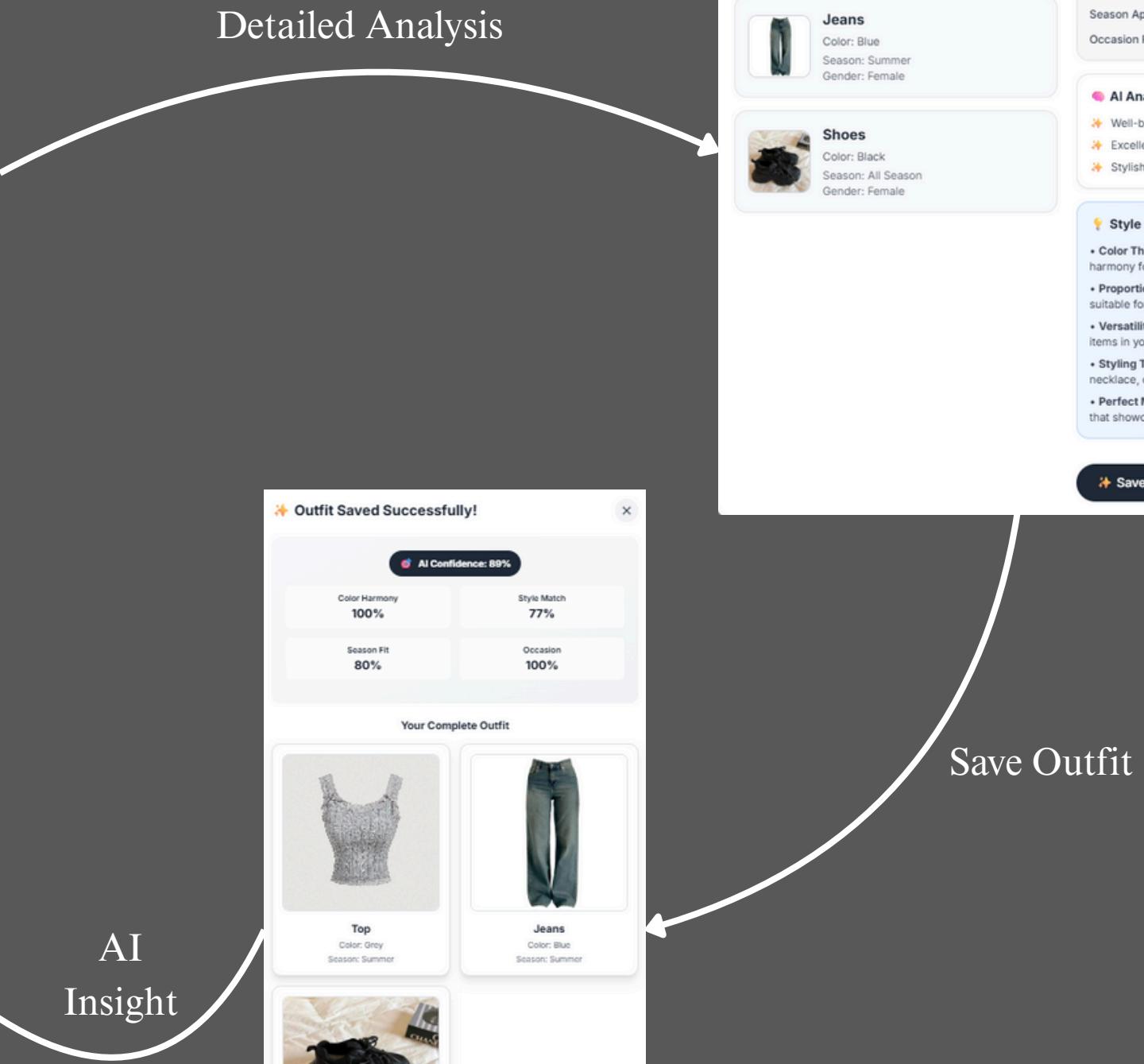
**Top** (Color: Grey, Season: Summer)  
**Jeans** (Color: Blue, Season: Summer)  
**Shoes** (Color: Black, Season: All Season)

**Why This Outfit Works:**

- Well-balanced top and bottom combination
- Excellent color harmony creates a cohesive look
- Stylish and perfect for a special occasion

**Perfect Close**

**Generate More**



# PROJECT STRUCTURE

```
huehive/
  └── app/          # Next.js app router (layouts & pages)
    ├── layout.tsx
    ├── page.tsx
    └── globals.css

  └── components/   # Reusable UI components
    ├── Dashboard.tsx
    ├── LoginPage.tsx
    ├── ProfilePage.tsx
    ├── RecommendationPage.tsx
    ├── UploadPage.tsx
    └── WardrobePage.tsx
      └── ui/        # Atomic UI elements (buttons, inputs, etc.)

  └── public/        # Static assets (images, icons, placeholders)
  └── styles/        # Global styles
  └── docs/          # Deployment-ready static export (GitHub Pages)
  └── out/           # Generated static build

  └── next.config.mjs # Next.js config
  └── package.json   # Dependencies & scripts
  └── README.md      # Documentation
```