

Capsule Wardrobe Generator

A Business Case for Personalized
Styling Using AI

Stats and Research

Metric	Women	Men	Source
<u>Time Per weekday morning</u>	16–17 minutes	~13 minutes	<u>Link</u>
<u>Total time over working life (18-60)</u>	~6 months	~4 months	<u>Link</u>

- 62% struggle with outfit planning
- 70% wear only 20% of their wardrobe
- Capsule wardrobe trends rising (TikTok/Pinterest)
- Personal styling cost = \$80–\$200/session → unsustainable

01

WHY Statement

Why Capsule Wardrobe AI?

- People struggle with “What should I wear today?” and repeat the same few outfits.
- Our AI helps users get more value from their existing wardrobe with effortless daily styling.
- Saves Time

Mission

We believe that outfit planning should be effortless, empowering, and sustainable.
Our mission is to help users fall in love with their existing wardrobe using smart, personalized AI.

The Problem



01

- Decision fatigue

02

- Repeating outfits

03

- Under-utilizing their wardrobe

04

- Time wasted choosing clothes



This problem represents a high-value opportunity for retailers and digital fashion platforms to drive personalization and increase user engagement.

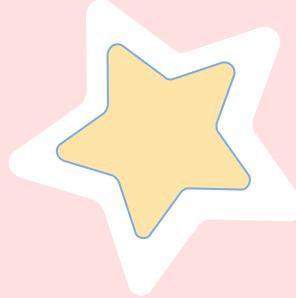
Challenge & Objective

Build an AI system
that can understand
a user's wardrobe
and generate
diverse, compatible
outfits automatically.

Deliver a balanced
30-day capsule
wardrobe using
visual AI, color logic,
and style similarity.



+Solution Overview



Customer

B2C

- Students
- Working professionals
- Fashion enthusiasts

B2B

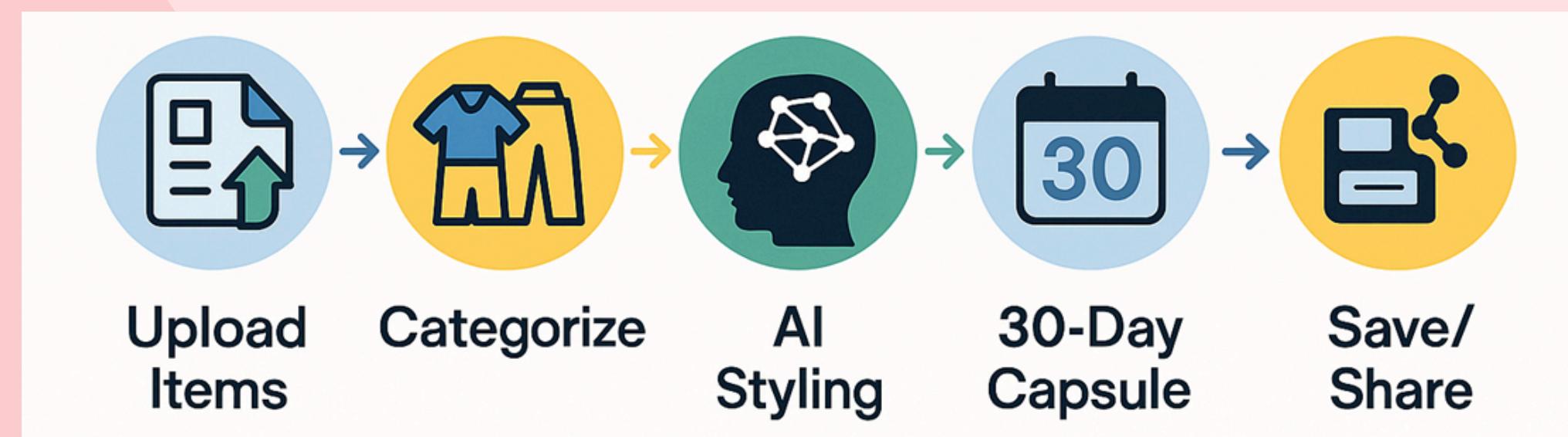
- Apparel retailers
- Fashion e-commerce platforms
- Styling apps

Competitor Landscape

Smart Closet		Botika	Pinterest Outfits
<u>Strengths</u> Basic Wardrobe Catalogue <u>Weaknesses</u> Manual Tagging	<u>Strengths</u> Retail Focused AI <u>Weaknesses</u> Expensive, B2B only	<u>Strengths</u> Outfit Generation <u>Weaknesses</u> Not Consumer facing	<u>Strengths</u> Inspiration <u>Weaknesses</u> Not Personalised
Our zero-shot AI auto-classifies	Our model serves both B2C + B2B	We target consumers directly	We generate personalized capsules

How it Works / Architecture

- Upload wardrobe images
- + • AI detects category + dominant color
- Generates compatible outfits
- Ranks them using style + color scoring
- + • Creates a personalized 30-day capsule calendar



Value Proposition

For B2C Users

- Saves time
- Reduces decision fatigue
- Creates new outfit combinations
- Increases wardrobe usage
- Inspires consistent, stylish dressing
- Emotional Value
- Users feel organized, confident, and stylish every day.

Value Proposition

For B2B Retailers

- Increase average order value
- Sell complete outfits instead of single items
- Improved retention & personalization
- Lower product returns

Cost Breakdown & Financial Overview

Development Costs

35k-55k

500-1200

80-150

Development Cost
Covers building the AI model, front-end UI, and backend system.

Operational Cost
Covers servers, storage, AI inference, and maintenance.

Scaling Cost
If the product gains users or partners (retailers).

Cost Breakdown & Financial Overview

Revenue Models

\$3-\$7

\$300-\$700

3%-10%
per sale

B2C: Premium Subscription

- Unlimited outfit generation
- Seasonal capsules
- Save/share calendar

B2B: Retail Partnerships

- “Shop the Look” integration
- Personalized product recommendations
- Capsule-building for their items

Commission-Based Sales

Earn commission when users buy clothing items from partner brands.

ROI Analysis

User ROI

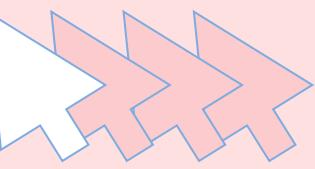
- ~60 hours saved annually
- ~\$200–500 saved in reduced shopping

Business ROI

- Retailers using outfit recommendation engines gain:
 - 12–30% increase in Average Order Value
 - Higher engagement & lower returns

Risks & Mitigation

Risk	Likelihood	Impact	Mitigation
Misclassification	Medium	High	Improve prompts, manual override
Poor Image Quality	High	Medium	Auto-enhance images
Retailers Slow to Adapt	Medium	Medium	Pilot B2C first



Team Structure

Hitakshi
Chugh

Model
architecture

Khushi
Navadiya

UI/UX design,
frontend

Navyasri
Edara

Documentation

Jacob Raj

Backend
(Flask),
deployment



Thank You!

Cmd

A

+

C

Ctrl

Z