

FASHION AND LIFESTYLE PLATFORM

CoolAndLovely

One place where a brand's
piece becomes a creator's
look, a shopper's board, and
something actually worn.

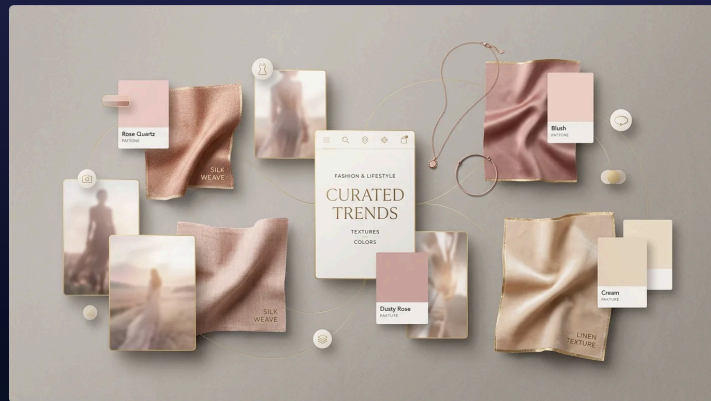
BRAND STOREFRONTS

DIGITAL WARDROBE

LOOKBOOKS

CREATOR COLLABORATIONS

GOVERNED AI



Pre-launch, waitlist open. Prepared for brands, creators and the technical reviewers who must approve them.

COOLANDLOVELY.COM

01 The short version

CoolAndLovely puts shoppers, brands and creators on one shared record of style, so the piece you discover, save, style and wear is the same object the whole way through.

Most fashion lives in pieces. Discovery happens on a social feed, the catalogue sits on a dozen brand sites, the clothes you already own live in your camera roll, and the relationship between a brand, a creator and a shopper is a thread of messages plus a spreadsheet of payments. Nothing joins up, so somebody is always re-typing something.

CoolAndLovely joins it up. A brand opens a storefront and lists real pieces into a seasonal collection. A creator builds those exact pieces into a published look with a styling note on each one. A shopper clips that look into a themed board, adds the piece to a digital wardrobe, and taps to log every time they wear it. No screenshots, no copies. One record, moving through its own life.

Underneath, all of it is workspace-native. Every storefront, collection, piece, board, look and partnership belongs to a workspace, so permissions, tenant isolation and an audit trail are there from the first record rather than retro-fitted when an enterprise buyer asks. That is why a shopper on the free plan and a fashion house running fifteen labels are on the same platform rather than two different products.



THE PROMISE

Style should be one connected object across its whole life - listed, styled, saved, owned and worn - not five apps and a folder of screenshots.

One platform, four structural facts

3

Sides, one data model

Shoppers, brands and creators write into the same record set. A catalogue piece becomes a look item, a board entry and a wardrobe piece without ever being copied.

9

Governed resource types

Storefronts, collections, pieces, looks, collaborations, boards, wardrobe items, style profiles and follows are each permission-checked in their own right.

Free

A real starting tier

The full product surface is reachable on the free plan. Editions differ by catalogue scale, collaboration throughput and governance depth, not by which features are switched on.

40

Tools for your own agent

The fashion domain is published over the Model Context Protocol, so your assistant works the catalogue with your login and exactly your permissions.

Who it is for

Style-conscious shoppers

You want one place that understands your taste, sizes and palette, keeps saved looks organised instead of scattered, and remembers what you already own so shopping fills real gaps rather than buying near-duplicates.

Brand marketing and merchandising teams

You staff storefronts, plan drops by season and run creator partnerships from one workspace, with roles that deliberately separate catalogue work from partnership terms.

Independent and emerging brands

You need a discoverable storefront, seasonal drops and a proper catalogue without standing up a commerce stack, an audience and a creator programme from scratch, and without renting your reach from a social platform.

Marketplace and community operators

You want catalogue, wardrobe and social as the backbone of a fashion experience, with a review step, curation levers and platform-wide oversight built into the lifecycle rather than bolted on later.

Established fashion houses and retail groups

You are consolidating storefront, collection planning and creator partnerships into one governed surface, with several labels under one workspace and one audit trail across all of them.

IT, identity and data protection teams

Single sign-on with the identity provider you already run, one identity holding shopper, creator and administrator roles across workspaces, authorisation decided before a request reaches the product, and every write on an audit trail.

Creators and stylists

You publish looks built from real catalogue pieces, keep credit that travels with you, and want brand partnerships to carry a brief, a role, compensation and a status rather than a handshake and a hope.

03 What breaks today

None of the problems below are exotic. They persist because the shopper's side, the brand's side and the creator's side have always lived in different systems, and every join between them is a person re-typing something.

Discovery is an endless, impersonal scroll

Shoppers wade past thousands of pieces in the wrong size, wrong palette and wrong budget, and give up before finding the one thing that suits them. Brands pay to reach an audience that was never a fit.

- Taste is declared, not inferred. A style profile carries preferred silhouettes, colours, fits, sizes and the occasions someone dresses for, and discovery ranks the real catalogue against it by attribute overlap you can see on the piece itself.

Inspiration dies in a camera roll

Saved posts, screenshots and open tabs pile up in three different places, and none of them stay linked to the piece, the price or the brand. The idea is remembered and the item is lost.

- Themed boards clip the live catalogue piece or the saved look itself, with a note and an order you control. A board entry stays connected to the real record, so it never decays into a picture of something you can no longer find.

A full wardrobe and nothing to wear

People own far more than they wear, forget most of it, and keep buying near-duplicates of things already hanging up. The money goes on the wrong gaps.

- A digital wardrobe holds the pieces someone actually owns, with source, colour, size, photo and notes. Logging a wear increments a count and stamps a date, and recording what a piece cost turns that into cost-per-wear.

Emerging brands cannot get seen without renting the audience

A small label depends on paid social to reach anyone, watches its organic reach get re-priced, and never owns the relationship with the people who buy from it.

- A storefront carries the brand's own handle, identity, aesthetics and hero collection, and is discoverable to shoppers whose declared taste actually fits it. The customer relationship stays attached to the brand's own record.

What breaks today, continued

Creator partnerships run on messages and goodwill

Briefs live in a chat thread, fees live in a spreadsheet, and credit disappears the moment a post scrolls away. Neither side has a record to point at when it matters.

- A collaboration is a real object with a brief, a participant role, compensation and currency, date windows and a checked lifecycle from proposed through accepted, in progress and published. Published looks are attributed back to it.

Everyone claims AI, almost nobody will say what it cannot do

Buyers cannot tell a real capability from a demonstration, and a confidently wrong styling suggestion or a fabricated search result costs more trust than saying nothing at all would have.

- Every AI capability here carries a status, and the assistant carries two written refusals: it will not invent styling advice and it will not invent search results. Where a control is a convention rather than a lock, this document says so.

Assembling a whole outfit is the actual hard part

Finding one nice item is easy. Knowing which shoes, which layer and which bag go with it is the bit people get wrong, so the piece is bought and then never worn.

- Looks are composed from real catalogue pieces with a styling note on each one, so a viewer sees the whole outfit and can shop any part of it. Boards mix things to buy with things already owned.

Fashion tools have no idea who is allowed to do what

Consumer fashion apps have no roles at all, and commerce tools have roles that were never designed for a merchandiser, a stylist and a shopper being three hats on one person.

- One identity holds several roles, and every action is checked against the role that holds it. A merchandiser builds a catalogue without seeing partnership fees; a brand lead publishes a storefront without any route into a shopper's private wardrobe.

What you get

The product is nine building blocks sharing one set of records. Three belong to the brand, three to the shopper, two to the social side, and one joins them together. Each is a real, permission-checked record rather than a screen.

SUPPLY

Brand storefronts

A brand's home: a handle unique in the workspace, a tagline and bio, aesthetic descriptors, logo and cover imagery, an optional website, and a checked lifecycle running draft, pending review, published, suspended and archived. Illegal jumps are rejected rather than quietly applied.

SUPPLY

Seasonal collections

Drops and curated edits grouping pieces by season - spring through winter, plus resort and all-season - each with a status, a cover and publish and end windows, so a collection goes live when the drop is ready rather than when someone remembers.

SUPPLY

Style items

The shoppable piece itself: category, status, stock code, primary image, price and currency, plus free-text colourways, sizes and materials. Categories span ready-to-wear, outerwear, footwear, bags, accessories, jewellery and beauty. Pieces are archived, never destroyed.

PERSONAL

Style profile

One per shopper per workspace: display name, bio, visibility, style preferences, a colour palette, a sizing record kept per garment type, and favourite brands. It is the taste vector discovery ranks against, and it belongs to the person who wrote it.

PERSONAL

Digital wardrobe

The pieces someone owns or wants: name, category, source, brand, colour, size, their own photo, notes, purchase price, an optional link to the catalogue piece it came from, and a wear log counting wears and stamping the last-worn date.

PERSONAL

Lookbooks

Themed boards - summer, work, wedding guest - each with a description, a cover and a visibility setting. Every entry clips a live catalogue piece or a saved look, carries a note, and can be re-ordered.

SOCIAL

Looks

A published outfit built from real catalogue pieces, with a title, caption, occasion, visibility, styled photo, like and save counts, and a styling note on every tagged piece so a viewer can shop any part of it.

SOCIAL

Collaborations

A brand-to-creator partnership carrying a title, brief, participant role, compensation and currency, start, end and published dates, and a checked lifecycle. It can produce a collection, and published looks can be credited to it.

DISCOVERY

Discover and Trends

Discover ranks the real catalogue against a declared style profile and the people you follow. Trends counts what is genuinely listed and engaged with - categories, colourways, brand aesthetics, occasions - and labels itself activity, never a forecast.

Put to work

Fifteen situations the platform was built for, written as scenarios rather than customer stories. CoolAndLovely is pre-launch, so nothing below describes a real deployment, and where a step depends on something still on the roadmap the scenario says so.

SHOPPER

A feed that fits you, built on taste you actually declared

The bind. Every fashion feed claims to know you, but it learned by watching what you nearly bought at one in the morning, and you cannot see or correct any of it.

- A feed you can audit, correct and explain - and a profile that sharpens everything downstream, from boards to the styling engine when it lands.

SHOPPER

Boards that hold the piece, not a picture of the piece

The bind. Your inspiration is spread across screenshots, saved posts and open tabs, and by the time you want to buy something you cannot find where it came from.

- Inspiration you can act on months later, still attached to the brand, the price and the piece.

SHOPPER

Know what you own, and what it has actually cost you

The bind. You own far more than you wear, keep forgetting half of it, and have no honest way of telling which purchases earned their keep.

- A durable, private record of what you have, how hard each piece works, and where the genuine gaps are.

SHOPPER

Dress for the occasion, not just buy another single piece

The bind. Finding one good item is easy. Knowing the shoes, the layer and the bag that go with it is the part that goes wrong, so the item hangs unworn.

- A complete, shoppable outfit for a real occasion - assembled by you, with the automated version honestly marked as not yet here.

BRAND OWNER

Launch a storefront that reads as your label, not a listing page

The bind. You make good things and need them seen, but the only route is renting an audience from a platform that re-prices your reach whenever it likes.

- A discoverable, on-brand front door that reaches shoppers whose declared taste actually matches your label.

Put to work: the brand side

MERCHANDISER

Plan a drop as a season, not as a spreadsheet upload

The bind. Your catalogue reads like an inventory export, when the entire point of a drop is that it has a point of view.

- A catalogue that browses like an edit, and a drop that goes live on the day it was planned for.

CREATOR

Publish looks you keep the credit for, and partnerships with a paper trail

The bind. Your styling moves product and nobody can prove it. Briefs live in a chat thread, fees live in a spreadsheet, and credit vanishes when the post scrolls away.

- A credited portfolio that travels with you, and partnerships that exist as records rather than as goodwill.

COMMUNITY MEMBER

Ask which one, from people whose taste you actually rate

The bind. Fashion is social, but the places to be social about it are all-public by default, so you either overshare or say nothing at all.

- A social layer that is genuinely opt-in per item, rather than an all-or-nothing public profile.

BRAND STRATEGIST

Read what is actually rising, and never a fabricated forecast

The bind. Trend reports are either expensive and generic or free and invented. You need evidence you can trace before committing a season's production to it.

- Directional evidence you can trace back to real listings and real engagement, with the limits of it printed on the surface.

PLATFORM OPERATOR

Run a curated marketplace instead of an open flood

The bind. A marketplace that grows without curation loses the point of view that made it worth joining, and there is no lever left to pull once the quality has gone.

- A quality bar enforced by the lifecycle rather than by remembering to check, with a full record of who changed what.

Put to work: governance and AI

WORKSPACE ADMIN

One person, several hats, no accidental access

The bind. On a style platform the same person is a shopper, a creator and the owner of a small label. Most permission models make you pick one and live with it.

- Roles that map to how fashion teams really work, with the sharp edges written down rather than discovered later.

BRAND MERCHANDISER

Ask on the screen you are already on, and have it done

The bind. Assistants in most products can describe the app but cannot touch it, so you get a paragraph explaining where the publish button lives instead of a published collection.

- Words become the same operation the buttons run, under your account and your permissions, with no click-by-click narration in between.

BRAND OPERATIONS LEAD

Point your own agent at the catalogue

The bind. Your team already works in other tools, and an AI product that assumes its own chat window is the only place a model will ever run is a product you have to leave in order to use.

- Your agent works your catalogue with your login and gets exactly the refusals you would get, because there is no privileged route for AI.

SHOPPER

See why something matched, not just that it did

The bind. Recommendation scores are unexplainable by design, so when a feed gets you wrong there is nothing to correct, and no reason to trust it when it gets you right.

- Ranking you can check by eye, and a grounded, structured substrate the styling engine can cite once it exists.

PLATFORM ADMINISTRATOR

Every AI action authorised the way a person's is

The bind. The frightening sentence in any AI roadmap is that the agent can also delete things - usually written shortly after someone gave AI its own permission model.

- One permission gate, one audit path, and an honest note that the destructive-action confirmation is still a convention rather than a code-level lock.

06 Built AI-native

AI that acts on your catalogue, and refuses outright the two things it cannot honestly do.

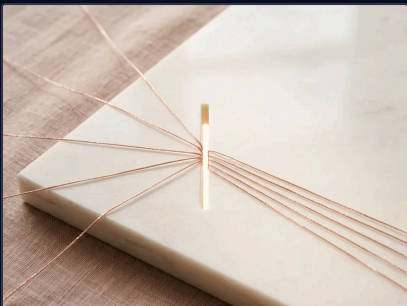
Bolting a chat box onto a fashion catalogue is easy and almost worthless. The assistant here is wired to the same contract the screens are, so publish this collection is the actual operation rather than a paragraph explaining where the publish button lives. It runs under your account, with your permissions and your plan limits, and every change it makes lands in the same activity record an administrator already reads for human ones. There is no AI-specific permission model and no privileged route to your data.

What makes it feel native is context. Before your prompt leaves the browser, the app resolves the screen you are on into a typed record - a storefront, a collection, a catalogue piece, a published look, a collaboration, a board, or a style profile - including on each of their edit screens, because an earlier version silently dropped the reference there and every question about this item became a guessing game. On list surfaces with no single record, such as your wardrobe, Discover and Trends, it passes the section instead, so what am I looking at still has an answer.

The third principle is that it is fenced. It ships with a written operational brief covering the entity model, all four lifecycle state machines, the self-scoped set and the pagination shape, plus two hard prohibitions: never fabricate styling advice and present it as the platform's recommendation, and never fabricate search results. There is no styling engine and no aesthetic search here yet, so an assistant that quietly improvised either would be the single most damaging thing it could do to a shopper's trust. Refusing is the feature, not a gap in the feature.



Four principles it is built on



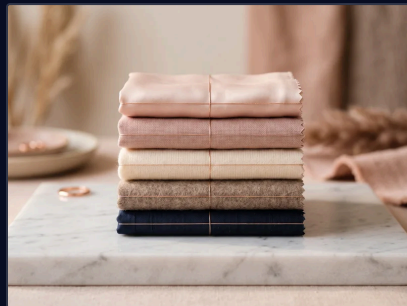
No special access for AI

The assistant and any agent you connect work with your permissions, checked at the same edge as the browser. If you cannot publish a collection, neither can they, and they report the refusal plainly instead of trying another route around it.



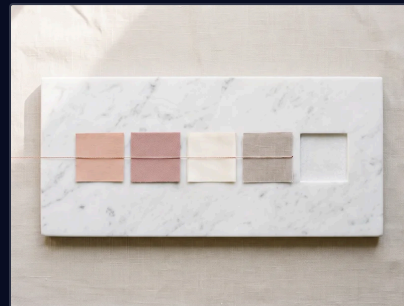
It already knows what you have open

Open a storefront, a collection, a piece, a look, a board or a profile and just say publish this. The screen you are on is resolved into the exact record before the prompt is sent, so this has an answer and you never paste a name or a reference.



Grounded before clever

Your taste is what you declare, not what you were watched doing. A match is explained by the colours, materials, tags and brands a piece genuinely shares with your profile, and every ranked row is a real row on the platform.



Refusal is a feature

Until real styling and aesthetic search exist, it will not improvise an outfit or a search result. A confident wrong answer costs a shopper more than a plain no, and costs the platform more than the feature was ever worth.

What the AI can actually do

Screen-aware page context

Resolves the storefront, collection, piece, look, collaboration, board or style profile you have open into a typed record, including on its edit screen.

Available today. It is why this works as a pronoun instead of forcing you to paste names or identifiers.

Lifecycle-aware actions

It is briefed on the four state machines - storefront, collection, look and collaboration - and reports the platform's own rejection when a step is skipped.

Available today. You get the real refusal rather than an invented success message.

Runs the real operation

Publish this collection publishes it. Take this storefront live walks the actual lifecycle. Nothing is simulated and nothing is merely described back at you.

Available today. Your words become the same action the buttons run, under your account.

A written operational brief

An authored guide to the entity model, the lifecycles, the self-scoped set and the pagination shape, rather than general knowledge about shopping apps.

Available today. It answers from this product, not from the category.

Named-board resolution

Add this look to my Summer board finds the board by name and clips the look, and asks which one you meant rather than silently creating a second board.

Available today. Ambiguity is surfaced rather than resolved by inventing a new record.

Two hard refusals

Never fabricate styling advice as the platform's recommendation, and never fabricate search results. Free-text aesthetic requests are declined rather than quietly degraded.

Available today. The two failure modes most likely to destroy trust are prohibited in writing.

The agent interface, and grounding

About forty fashion tools over MCP

Storefronts, collections, pieces, looks, collaborations, boards, wardrobe and style profile published as typed tools any compatible client can call.

Available today. Each tool is a thin wrapper over a real operation; there are no convenience endpoints that exist only for an agent.

Read-only grounding resources

Resources expose the live schema plus current brands, looks and style items, so an agent reads reality before acting on it.

Available today. Field names are read rather than guessed.

Twenty-one shared platform tools

Files, tags, conversations, notifications, permissions, workspaces and billing exposed to the same agent alongside the fashion domain.

Available today. An agent can finish a workflow without dropping out to a second integration.

A declared, editable taste vector

Palette, style preferences, favourite brands, sizes and who may see them are yours to set and change, and Discover ranks against them.

Available today. Nothing is inferred from watching your behaviour.

Guided agent prompts

A brand-onboarding sequence covering storefront, collection, catalogue and publication, and a shopper styling sequence, each instructing the model to show results and confirm before publishing.

Available today. The workflow is authored rather than improvised by whichever model is driving.

Transparent overlap ranking

Candidates are scored by counting the colourways, materials, tags and brands a piece genuinely shares with your profile, plus the people you follow.

Available today, and deliberately model-free. It is the legible baseline anything cleverer must beat.

Shipped, in progress, and roadmap

Honest empty states

With no style profile yet, Discover says so and invites you to build one rather than filling the screen with whatever is popular.

Available today. A cold start is named instead of papered over.

A written behaviour test set

Committed cases cover everyday questions, refusing access to other people's records, confirming before deletion, and attempts to talk it into inventing answers.

In progress. The promised behaviour is written down as checkable cases; running them automatically is still being built.

Activity signals, never forecasts

Trends counts the most-listed categories and colourways, the aesthetics brands claim, engagement by occasion and the most-loved looks, and labels itself activity.

Available today. No forecaster exists, so none is implied on the surface.

Outfit styling and wardrobe gap analysis

Planned: head-to-toe outfits built from what you already own first, then the catalogue, inside your palette, sizes and budget, plus the occasions your wardrobe genuinely cannot cover.

Roadmap. When it lands it will cite the pieces behind each slot and hand you an editable draft look, never publish one for you.

Confirmation before destructive work

Ten deleting or hiding operations are enumerated with their real blast radius, and the assistant must restate the target by name before acting on any of them.

In progress. Today that is a rule it follows, not a server-side lock. Enforcement in code is tracked work.

Aesthetic and natural-language search

Planned: a described mood resolving to real catalogue rows by matching colourways, materials, seasons, tags and the styling notes creators attach.

Roadmap. Today only fixed filters exist, and the assistant declines free-text style requests rather than dressing up unrelated results as matches.

And what keeps it in bounds

One permission model, no AI exception

Every operation carries an authorisation rule naming action, resource and scope, enforced at the platform edge from the composed schema before a request reaches the service. The assistant, an external agent, the browser and a raw request are checked identically.

Self-scoped records cannot be redirected

Your style profile, wardrobe, boards and follows are structurally incapable of taking someone else's identity as an argument. There is nothing in the request to point at another person.

Tenant re-filtering on every write

Each workspace write is re-filtered by tenant and asserts exactly one affected row, so an identifier guessed from another workspace matches nothing rather than mutating something it should not.

Attributable by default

Every mutation is recorded with the operation, the actor and the workspace. An assistant-initiated publish is indistinguishable from a human one in the trail except for who the actor was, which is precisely the property an administrator needs.

Redacted, bounded audit payloads

Sensitive keys are stripped and arguments are depth- and size-bounded before anything is recorded, so the audit trail stays useful without becoming a second copy of your data.

An enumerated destructive contract

Archiving a storefront, soft-deleting a collection or a piece, suspending a live brand, deleting a look or a whole board, cancelling an in-flight collaboration, removing a wardrobe piece along with its wear history, unfollowing - ten operations, each listed with its real blast radius.

Guardrails, and where they stop

A vague instruction is not consent

Clean up my catalogue is explicitly not treated as permission to archive anything. The assistant must name the affected record and wait for you to confirm that specific thing.

The confirmation gap, stated plainly

That confirmation is a prompt-level contract plus derived tool annotations today, not a hard server-side gate. The enforced mode exists in the underlying library but is opt-in, and this product has not switched it on.

Behaviour written down as tests

A committed evaluation set covers ordinary questions, cross-account refusals, deletion confirmation and jailbreak attempts, including one naming the wrong kind of record. Running it automatically is still being built.

Metering, honestly partial

Creating a catalogue piece and publishing a look already count against your plan, and the assistant transport is credit-aware. Still missing: a provisioned AI credit allowance and a considered ceiling per action.

No model advertised, and no bring-your-own-key yet

Nothing here needs a model today, because the assistant runs real operations rather than generated advice. Model choice and bring-your-own-key arrive together with spend limits and a per-run record, not before them.

Nothing runs unasked

No scheduled work runs against the fashion domain today and no proposal queue exists. The planned agent loop is blocked on a durable run ledger and on confirmation being enforced in code, which is the honest reason rather than a sequencing excuse.

06 Reachable from, and in numbers

REACHABLE FROM

- The in-app assistant, on every fashion screen - storefronts, collections, pieces, looks, wardrobe, boards, Discover and Trends
- A published Model Context Protocol server, for Claude, a coding agent, or your own automation
- Guided agent prompts for brand onboarding and for the shopper styling journey
- Read-only grounding resources exposing the live schema plus current brands, looks and pieces
- The Discover ranking surface, where the reason for a match is visible on the piece itself
- The Trends surface, counting real listings and real engagement and labelled as activity
- The fashion API, reached through the same gateway and the same permission checks as everything else
- Style Intelligence, the planned home for AI proposals carrying the signal and the exact action on each card
- Scheduled autonomous loops, planned and explicitly gated on run records and enforced confirmation

IN NUMBERS

- About forty typed fashion tools published over the Model Context Protocol, plus twenty-one shared platform tools
- Two hard prohibitions written into the assistant brief: no fabricated styling advice, no fabricated search results
- Ten destructive operations enumerated with their real blast radius before the assistant may touch any of them
- Four lifecycle state machines the assistant is briefed on: storefront, collection, look and collaboration
- Nine governed resource types, each permission-checked in its own right
- Six workspace roles seeded automatically into every workspace and reconciled on every update
- Zero AI-specific permissions: the assistant, an external agent and the browser queue at one gate
- Two catalogue operations already metered against your plan; an AI credit allowance is not yet provisioned
- Zero scheduled agent runs against the fashion domain today, and no proposal queue

What the AI does not do yet

What the AI does not do yet, stated plainly. There is no styling engine: it cannot build you an outfit or tell you what your wardrobe is missing, and it is instructed to refuse rather than improvise.

There is no aesthetic or natural-language search: describe a mood in free text and it will say so instead of listing unrelated pieces and calling them a match. The destructive-action confirmation is a documented contract plus a set of derived tool annotations, not a server-side lock - the enforced mode exists but is not switched on

here, and the archive-shaped tools are not even flagged destructive by the naming heuristic. The behaviour test set is committed but nothing runs it automatically. There is no per-request record of what a model was asked, what it proposed, what it cost and what a person did with it, and no AI credit allowance - which is exactly why the scheduled agent loop is not shipping rather than shipping unaudited. No model is advertised and bring-your-own-key is not available. Every one of these is tracked work, not a claim.

Who does what

Six roles are seeded into every workspace automatically and reconciled on every update, so nobody builds them by hand. Each grants one level per resource across the nine governed types - administrator, editor or viewer - and leaving a resource out of a role grants nothing on it. The omission is the access control, not a hidden setting.

Style Director

Accountable for the catalogue, the creator programme and community content

The single accountable owner of the workspace's fashion surface, and the only role that can archive a storefront outright. It is the one role granted all nine resources, and it keeps a personal wardrobe and boards like any other member.

- Archive a storefront
- Own the full catalogue
- Run the creator programme
- Oversee community content
- Keep a personal wardrobe and boards
- Set workspace-wide standards

Brand Manager

Storefront lifecycle, seasonal drops and partnership terms

Runs a storefront end to end - creating it, driving its whole publish lifecycle, owning the catalogue and committing the brand to paid creator collaborations - while stopping short of archiving the storefront and holding no route into a shopper's private records.

- Create and update a storefront
- Submit, publish and suspend it
- Retire and delist catalogue pieces
- Publish and end collections
- Propose, accept and cancel collaborations
- No access to wardrobes or boards

Merchandiser

Getting the season built and listed correctly

The day-to-day catalogue builder on a brand or merchandising team: creates and edits collections and pieces, but cannot create or archive a storefront, retire a drop, or see partnership terms at all.

- Build seasonal collections
- List and edit style items
- Set colourways, sizes and materials
- Read the storefront and looks
- No collaboration access whatsoever
- No access to personal resources

Who does what, continued

Creator

Publishing work, keeping credit, getting paid

The stylist and micro-influencer role: authors, publishes and retires their own looks, negotiates and delivers the collaborations they take part in, and keeps their own style identity - while only ever browsing the brand catalogue.

- Publish and retire own looks
- Tag catalogue pieces with styling notes
- Negotiate and deliver collaborations
- Keep a wardrobe, boards and a profile
- Follow and be followed
- Read-only on the catalogue

Shopper

Discovering, saving, owning and wearing

The consumer role and the largest population: full control of their own profile, wardrobe, boards and follows, plus the ability to react to and clip community looks, with the brand catalogue read-only and collaboration terms out of reach entirely.

- Build and edit a style profile
- Catalogue a wardrobe and log wears
- Curate and share boards
- Like, save and clip looks
- Follow people and brands
- No collaboration access whatsoever

Style Analyst

Reviewing without the ability to change anything

A deliberately appointed read-only reviewer for merchandising performance, partnership oversight and brand safety - appointed rather than broadly assigned precisely because a collaboration read exposes compensation across the workspace.

- Review storefronts and drops
- Review published looks
- Read collaboration terms
- Read style profiles
- No wardrobe or board access
- No write access anywhere

08 Under the hood

CoolAndLovely is delivered as software as a service and built API-first. The fashion domain - catalogue, wardrobe and social - is a federated GraphQL surface composed into one workspace schema alongside the shared services it reuses, so a client fetches a storefront, its collections, the files behind its imagery, the tags on it and the conversation threads beneath it in a single request rather than five. Clients never talk to the domain service directly.

Authorisation is the architectural decision that shapes everything else. Every query and mutation carries an access rule that the API gateway evaluates before the request reaches the product; the service itself never validates a token and never re-implements authorisation. Once a request is authorised, the resolver filters again by the workspace it arrived from and asserts exactly one affected row, so a permission cannot reach another organisation's records even if the first check were somehow wrong.

Lifecycles are state machines rather than free-text status, so a storefront, collection, look or collaboration cannot enter an illegal state because a client sent an unexpected value. Deletion is soft throughout - records carry a deleted timestamp and archive flows never destroy data - which is what makes the audit trail worth keeping. Imagery is referenced from the shared file service rather than stored in the fashion domain, so a storefront cover, a look photo and a wardrobe shot all inherit one set of storage, access and lifecycle rules.



THE SPLIT THAT MATTERS

Authorisation is decided before the request reaches the product, and the product's own workspace filter is defence in depth rather than the only line. That split is what makes one identity holding many roles across many workspaces safe, and it is why an assistant, an external agent and a browser cannot diverge in what they are allowed to do.

Capability by area

Catalogue

Storefronts with a handle unique per workspace, seasonal collections, and style items carrying category, stock code, price and currency, colourways, sizes and materials. Cursor-paginated listings with status, featured, brand, collection and category filters.

Authorisation

Every query and mutation carries a rule naming action, resource and scope, enforced at the gateway from the composed schema. Nine resource types, three levels, six workspace roles plus organisation and billing presets, all seeded and reconciled automatically.

Wardrobe and identity

One style profile per person per workspace, a digital wardrobe with source, imagery, notes, purchase price and a wear log, and themed boards whose entries clip a live catalogue piece or a saved look with a note and an order.

Audit and events

Every mutation is recorded with operation, actor, workspace and result, with sensitive keys stripped and payloads bounded. Domain changes are emitted as events for the relationship graph that backs shop-the-look connections.

Social

Published looks composed of tagged catalogue pieces with per-piece styling notes and ordering, like and save counters, a follow graph, and brand-to-creator collaborations with a brief, role, compensation, currency and lifecycle.

AI interfaces

A screen-aware in-app assistant bound to the same operations as the interface, and a published Model Context Protocol server exposing about forty typed fashion tools plus twenty-one shared platform tools, guided prompts and read-only grounding resources.

Discovery

Deterministic attribute-overlap ranking across the real catalogue and follow graph, plus platform-wide activity aggregation over listings and engagement. Full-text search is in progress; aesthetic and semantic search are on the roadmap.

Internationalisation and accessibility

Multi-lingual interface with locale-aware date, number and currency formatting and right-to-left support, plus keyboard navigation, a command palette, screen-reader semantics, focus management and a high-contrast theme.

08 Standards it speaks, systems it plugs into

OPEN STANDARDS

- GraphQL, served as one federated API across the fashion domain and the shared platform services
- OpenID Connect for single sign-on with the identity provider you already run
- OAuth 2.0 device-code flow for terminal and headless clients
- OAuth 2.0 client credentials for machine-to-machine integrations
- Model Context Protocol, so AI clients act through the same governed operations a person uses
- Cursor-based pagination on every list surface
- Webhooks for event delivery into your own systems
- SAML and SCIM for enterprise identity and provisioning
- Accessibility semantics, keyboard navigation and high-contrast theming
- Progressive web app install, so the web client runs from a home screen like an application

PLUGS INTO

- Enterprise identity providers over single sign-on, shared across every product in the suite
- Role-based access control and policy, enforced at the gateway edge
- Multi-tenancy, including dedicated tenancy on enterprise agreements
- Billing, plans, regional pricing bands, add-on capacity and usage quotas
- Notifications across in-app, email, push and webhook, with per-user preferences
- Conversations, for comments and threads on looks, pieces and collaborations
- Tagging, one workspace-wide service shared with storefronts, pieces and looks
- Shared file storage for storefront logos and covers, look photos and wardrobe shots
- Activity and audit timelines, searchable and exportable
- Feature flags, developer-portal keys, in-product support and the add-on marketplace

How it deploys

Multi-tenant cloud

The delivery model today. Your workspace is isolated by construction, with shared platform services and storage and request quotas pooled across every product on the same account.

Composed into your workspace

The product experience composes into a shared application shell alongside the other products in the suite, so one sign-in, one set of roles and one activity trail cover all of them.

Dedicated tenancy

Available on enterprise agreements for fashion houses and retail groups that need a separate tenant rather than a separate schema. Scoped per contract rather than self-serve.

Self-hosting and data residency

Contracted options for organisations with residency or sovereignty requirements, scoped per agreement. Not available to buy self-serve.

Web, and installable

The cross-platform web client is the shipping product, and it installs to a home screen and runs like an application. Native iOS and Android shells are in progress and not yet generally available.

Programmable

The fashion API and the Model Context Protocol server are available now. A command-line client and typed Node and Python SDKs are specified and sequenced on the roadmap.

Security and trust

Security here is inherited rather than reinvented, which is the point: identity, authorisation, tenancy, audit and secret handling are shared platform services, so they behave identically here and in every other product on the same workspace. The controls below are the ones a technical reviewer should actually check, and the honest limits follow immediately after.

Authorisation decided before the product sees the request

Every query and mutation carries an access rule naming action, resource and scope. The gateway reads it from the composed schema and decides before the request reaches the domain service, which never validates a token. There is one place to audit rather than one per feature.

Roles seeded and reconciled, never hand-built

Six workspace roles plus organisation and billing presets are created when a scope is created and re-applied on every reconcile, so a role definition cannot drift between environments or be forgotten when a new workspace appears.

Validated lifecycles

Storefront and collaboration transitions are state-machine-checked rather than free text. An illegal transition is rejected with an error, so a record cannot be pushed into a state the rest of the system does not expect.

Workspace filtering as defence in depth

Even after a request is authorised, every write filters by the workspace it arrived from and asserts exactly one affected row. An identifier guessed from another tenant matches zero rows, so the failure mode is a not-found rather than a cross-tenant mutation.

Self-scope enforcement inside the service

Style profiles, wardrobe items, boards and follows are owner-filtered by the calling identity on top of the gateway grant, so a valid identifier belonging to another person resolves to nothing rather than to a permission error.

Nine governed resource types

Storefronts, collections, style items, looks, collaborations, boards, wardrobe items, style profiles and follows are each permission-checked independently at one of three levels. Omitting a resource from a role is the denial - there is no explicit none.

Author and participant filtering on writes

Look updates, publications and deletions are author-filtered, and collaboration writes are filtered to the participating creator or brand owner, so even a broad role level does not reach another person's content.

Security and trust, continued

Mutation audit trail with redaction

Every mutation is recorded with the operation, the actor, the workspace and the result, with sensitive keys stripped and arguments depth- and size-bounded before anything is written.

AI held to the permissions of the person asking

There is no AI-specific permission model. An assistant driven by a shopper account gets the same refusal a shopper would, and reports it rather than trying another route, so AI cannot be used to obtain access a role does not have.

Published subprocessors and disclosure contact

The infrastructure, payment, communication and verification providers that help run the service are published, with any not yet finalised marked as such, alongside a published contact for responsible disclosure.

Soft deletion throughout

Records carry a deleted timestamp and archive flows never destroy data, so a mistaken archive is recoverable and the audit trail continues to resolve the records it references.

Encryption and managed secrets

Data is encrypted in transit and at rest, and platform secrets are held in a managed vault rather than in configuration, with each environment reading only its own set.

Per-item and per-look visibility

Wardrobe items, boards, looks and the style profile each carry their own visibility setting - public, followers only, or private - so sharing is a per-record decision rather than an account-wide switch.

Data minimisation and no data selling

Only the data needed to run discovery, boards and the wardrobe is collected, and personal data is never sold to third parties. You can view, edit, export or delete your information.

Stated plainly

CoolAndLovely is pre-launch and holds no third-party security certification. It is not SOC 2 audited and not ISO 27001 certified; compliance evidence exports and streaming into a security information and event management system are roadmap, and the enterprise compliance packages are contracted commitments rather than certificates in hand. Multi-factor authentication is on the roadmap rather than shipped. Quota enforcement at the action layer is partial: two catalogue operations are metered against your plan and the rest are not yet. The destructive-action confirmation for AI callers is a documented contract plus derived annotations, not a server-side lock. Enforced single sign-on, SAML and provisioning are enterprise

contract items rather than defaults on the self-serve plans. Two access-control limits are worth stating outright: a collaboration read is not participant-filtered, which is exactly why the read-only reviewer role is described as deliberately appointed and why the merchandiser role is given no collaboration access at all; and there is no community-moderation path over other people's looks, because look writes are author-filtered even for the most senior role. Platform data is hosted in India, and residency options beyond that default are scoped per enterprise agreement. Any certification will be claimed only once it has been independently awarded.

One product, many front doors

One product, reachable from several places - and honest about which of them exist today. The web client, the fashion API, the assistant and the agent interface are the product right now. The desktop, terminal and language-SDK surfaces are specified and sequenced rather than shipped, and are labelled as one or the other rather than implied.

Web application

Available. The shipping client: storefronts, collections, pieces, looks, collaborations, wardrobe, boards, the style profile, Discover and Trends, all bound to the live fashion API.

Composed workspace experience

Available. The same product composed into a shared application shell, so a team already on the platform reaches it alongside everything else they run, under one sign-in and one set of roles.

Fashion API

Available. The complete domain - catalogue, wardrobe and social - as a federated GraphQL surface through the workspace gateway, with authorisation enforced at the edge.

In-app assistant

Available. Screen-aware, mounted on every fashion surface, running the same operations the buttons run under your account and your permissions.

MCP server

Published. A versioned Model Context Protocol server exposing about forty typed fashion tools plus twenty-one shared platform tools, guided prompts and read-only grounding resources. Honest gap: enforced confirmation on destructive tools is not switched on.

Installable web client

Available. The web app installs to a home screen and runs like an application, which is what carries the mobile experience until the native shells reach general availability.

In progress and planned

Marketing site and waitlist

Available. Product story, features, the use-case library, comparison pages, the honest AI status page and the waitlist that opens access ahead of the 2026 launch.

Documentation

In progress. Product guides, API reference and integration tutorials, published at the documentation subdomain alongside the launch.

Native mobile applications

In progress. iOS and Android for discovery, wardrobe and looks on the move, and for creators capturing and publishing from a phone. Not yet generally available.

Creator Studio desktop

Planned. A signed native desktop workspace for brands and creators covering storefronts, catalogue, collaborations and looks in one window. Binaries are not published yet.

Command-line interface and SDKs

Planned. A terminal client for bulk catalogue and collaboration work, plus typed Node and asynchronous Python SDKs for integrations and analytics pipelines. Install commands are published; the packages are not on the shelf.

Browser and editor extensions

Planned. A browser companion for clipping a piece into a board from any page, and an editor extension for reaching catalogue records and collaboration briefs without leaving the tool.



11 Plans

Pricing is published rather than quoted, and priced per brand workspace rather than per seat, because the number of merchandisers and stylists in a fashion business moves with the season. Every user-facing feature family is reachable from the free plan; editions differ by catalogue scale, collaboration throughput, governance depth and support. Prices are the planned launch configuration and remain indicative until paid checkout opens.

FREE

\$0

An individual shopper, stylist or single boutique evaluating the platform. No card required.

- 1 brand storefront, 25 published style items, 1 collection
- 50 wardrobe items, 3 boards, 10 looks published a month
- 1 active collaboration, 2 administrative seats
- Style profile, wardrobe, wear log and Discover in full
- 1 GB shared storage, 500 API requests a day
- Community support plus one ticket a month, 7-day audit retention

PRO

\$29/mo, or \$23/mo billed annually

Independent creators, boutiques and emerging brands running a real catalogue and a handful of partnerships.

- 3 brand storefronts, 500 published style items, 10 collections
- Unlimited wardrobe items, 50 boards, 200 looks a month
- 10 active collaborations, 5 administrative seats
- 10 connected integrations, 25 GB shared storage
- 10,000 API requests a day
- Email support under 48 hours, 30-day audit retention

BOUTIQUE

\$149/mo, or \$119/mo billed annually

Established brands, multi-line boutiques and stylist agencies running several storefronts at once.

- 15 storefronts, 10,000 published style items, unlimited collections
- Unlimited looks under fair use, 100 active collaborations
- 25 administrative seats across up to 5 brand workspaces
- 50 integrations, 250 GB pooled storage
- 250,000 API requests a day
- Priority support under 8 hours, 180-day audit retention

ENTERPRISE

Custom, contracted annually

Fashion houses, retail groups and multi-brand marketplaces buying governance and scale rather than seats.

- Custom quotas across storefronts, catalogue and collaborations
- Enforced single sign-on, SAML and provisioning
- Multi-brand organisation structures and dedicated tenancy
- Self-hosting and data-residency options
- Custom audit retention and event streaming into your own tooling
- Dedicated service-level agreement with service credits

How the pricing behaves

Regional pricing, four permanent bands

Prices are banded by purchasing power rather than discounted case by case. India sits in the fourth band, where Pro is Rs 1,499 a month (Rs 1,199 a month billed annually) and Boutique is Rs 6,999 a month (Rs 5,599 a month billed annually). The other bands cover the Gulf, Eastern Europe, Taiwan and Hong Kong, and Latin America, Southeast Asia, Turkey and South Africa.

Add-ons sit on top of a base plan

Capacity add-ons for extra storefronts, published pieces and boards, priced per unit each month. Premium add-ons for advanced analytics, predictive trend insight and AI styling credits. Support add-ons for a premium service level and a dedicated success manager. Marketplace add-ons for featured listings and promoted collections, priced per campaign or as a revenue share.

Pooled across the account, not per product

Storage and shared API request limits are pooled across every product on the same account, so subscribing to more of the suite raises the shared ceiling rather than fragmenting it. Marketplace sales and paid creator collaborations carry a platform take rate, documented separately.

What is not sold as available

AI styling, on-platform checkout and payouts, and advanced predictive analytics appear in the quota tables as the planned launch configuration. They are gated behind feature flags and quotas and are not sold as available before the corresponding capability ships. Prices remain indicative until paid checkout opens.

Why teams choose it

One data model, not four synchronised apps

A catalogue piece flows through storefront, creator look, shopper board and wardrobe wear log as the same record. Nothing is re-keyed and nothing is a screenshot, which is a structural property a competitor assembled from a feed and a checkout cannot retrofit later.

Storefronts with a real lifecycle

Draft, pending review, published, suspended, archived - each transition checked, rather than a single publish toggle. That is what lets a marketplace stay curated as it grows instead of relying on someone remembering to look.

A wardrobe joined to a live catalogue

Wardrobe pieces link back to the catalogue item they came from, so recommendations fill gaps in what you already own. Dedicated closet apps are deliberately disconnected from commerce, and discovery platforms have no ownership layer at all.

Wear logging and cost-per-wear

Every wardrobe piece counts wears and stamps the last-worn date, and recording what you paid turns that into cost-per-wear. Where price or wears are missing the figure is omitted rather than estimated.

Partnerships as records, not messages

A collaboration carries a brief, a role, compensation and currency, date windows and a checked lifecycle, and published work is attributed back to it. Both sides end up with something they can point at.

Why teams choose it, continued

Privacy per item, not per account

Public, followers only, or private, set on each wardrobe piece, board, look and the style profile itself. Social fashion platforms that are all-public by default force an all-or-nothing choice on everybody.

Enterprise governance under a consumer surface

Authorisation on every operation, tenant isolation asserted on every write, and an audit trail from the first record - inherited from the platform rather than built for a later enterprise release. No consumer fashion competitor offers this.

AI that is bounded in writing

Two prohibitions the assistant will not cross, ten destructive operations enumerated with their blast radius, and a published statement of what is a convention rather than a lock. Most products state the capability and leave the boundary undefined.

Explainable ranking, on purpose

Discover is deterministic attribute overlap rather than an unexplainable score. It is the legible baseline any future model has to beat, and it means a bad match is something you can correct rather than merely endure.

Nothing claimed that is not built

Every capability across the site, the specifications and this document carries a status, and roadmap items are named rather than blurred into the shipped list. That is the differentiator that survives a technical evaluation.

Honest answers to hard questions

You are pre-launch. Why would we commit now?

You should not commit as though we had a track record, because we do not have one - there are no customers, no revenue and no usage to point at. What exists is the working software: the fashion domain, its permission model and its audit trail are implemented, and the web client runs against them. The honest reasons to engage now are shaping the roadmap while it is still shapeable and getting early access at launch. The waitlist costs nothing.

Everyone says AI-native. What does yours actually do today?

It acts. On any fashion screen it resolves what you have open into the exact record and runs the real operation - publishing a collection, taking a storefront live, clipping a look into a board you named. Outside the app, about forty typed tools let your own agent do the same with your login. What it does not do is style an outfit or answer a free-text mood search, and it is instructed to say so rather than improvise. That refusal is the honest half of the answer.

Your competitors are far bigger and have real catalogues.

They do, and we do not out-aggregate them. A search aggregator carries tens of thousands of retailers, and a visual discovery platform has an index we will not match. What none of them have is a single record that survives from listing through styling to a logged wear, or governed storefronts and structured creator partnerships alongside it. We compete on integration and governance, not on breadth.

AI styling is the feature we actually want, and you say it is not built.

Correct, and we would rather you knew now. What is built is the substrate a styling engine needs: a declared taste vector, a wardrobe with wear counts, catalogue pieces with real colourways and materials, looks with per-piece styling notes, and engagement counters. The engine is blocked on the shared model runtime and the run ledger, because shipping an unlogged recommendation engine hands you something nobody can govern afterwards.

There is no checkout. How does anyone buy anything?

On-platform checkout, carts, orders and inventory are roadmap, and the quota tables mark them as planned rather than selling them. Today the platform owns discovery, ownership and the creator relationship, and the purchase completes with the brand. If checkout is your deciding requirement, this is not the release to buy - that is a straight answer rather than a deflection.

Honest answers, continued

You are not SOC 2 or ISO 27001 certified.

We are not, and the compliance evidence exports and event streaming that support those programmes are on the roadmap. What is in place today is authorisation enforced at the gateway on every operation, tenant isolation asserted per write, self-scope filtering, soft deletes, an audit trail with redaction, encryption in transit and at rest, and a published subprocessor list. For a regulated buyer that is a starting position rather than a finished one, and we would rather say so than imply otherwise.

Can an agent delete our catalogue?

It cannot do anything you cannot do, because there is no AI-specific permission model. Ten destructive operations are enumerated with their real blast radius, and the assistant must restate the target by name first - clean up my catalogue is explicitly not consent. Being straight about the gap: that is a rule it follows rather than a server-side lock today. Enforcement in code is tracked work, and it is one of the two things blocking scheduled agent runs.

Native mobile matters to our audience and you do not have it.

The web client is cross-platform and installs to a home screen, and the native iOS and Android shells are in progress but not generally available. We describe that as in progress rather than shipped, because a fashion audience is mobile-first and overstating it would be found out on the first day.

We already run a commerce stack. Why add another system?

Because the parts CoolAndLovely owns are the parts a commerce stack does not: a shopper's declared taste, a wardrobe of what they actually own and wear, creator looks built from your real pieces, and partnerships carrying a brief and a lifecycle. It is deliberately not trying to be your order management system, and integration is through the API, webhooks and the agent interface rather than a migration.

How do we know the pricing will not move before launch?

It may. The published prices are the planned launch configuration and are marked indicative until paid checkout opens and finance sign-off completes. The structure is settled - four editions, priced per brand workspace, banded regionally, with capacity and premium add-ons on top - and we would rather flag the caveat than let you plan a budget on a number that changes quietly.

Where it sits against the alternatives

VS FASHION SEARCH AGGREGATORS

They win on catalogue breadth, with tens of thousands of brands and stores and a search intent we do not match and are not trying to. Their journey ends at an affiliate link. Ours continues into a board, a wardrobe entry and a logged wear, and adds governed storefronts and structured creator partnerships they have no equivalent for. Cross-retailer price comparison, which is their genuine strength, is on our roadmap and is not built.

VS SOCIAL RESALE MARKETPLACES

They have a genuinely engaged community, very low seller fees and a real circular-fashion story we respect. They are also resale-only, all-public by default, and carry no official brand storefronts or professional tooling. We ship the storefront lifecycle, seasonal collections, structured collaborations and per-item visibility. Peer-to-peer resale, which is their entire model, is on our roadmap and does not exist here today.

VS LUXURY E-COMMERCE DESTINATIONS

Their editorial authority and white-glove logistics are excellent and we do not compete on either. They are also one-way: no user-generated looks, no wardrobe, no creator economy, and a price floor that excludes most people. We open curated storefronts to brands at every price point and add community and ownership. Editorial and magazine content, one of their strongest assets, is roadmap for us.

VS PERSONAL STYLING SUBSCRIPTIONS

Their algorithm-plus-human styling is real, proven and ahead of ours - we have no styling engine at all yet and say so throughout this document. What we offer instead is free, open discovery with no styling fee, a wardrobe you own and control, and a social layer a private styling box has no equivalent for. If having a stylist choose for you is the requirement, they are currently the better answer.

VS CLOSET-ORGANISER APPS

The mature ones have far more wardrobe features than we do and a one-off price rather than a subscription. They are also deliberately disconnected from commerce and community, and single-platform. We link wardrobe pieces to live catalogue records so you can shop real gaps, add looks and a follow graph, and run cross-platform. On pure wardrobe depth they are still deeper than we are.

VS VISUAL INSPIRATION PLATFORMS

Their visual search index, scale and recommendation quality are in a different league, and visual search and model-driven recommendations are roadmap for us rather than shipped. Their weakness is that a saved image is where the journey stops: no ownership, no wear tracking, no storefront lifecycle, no structured partnerships. We turn the save into a record you own.

START TODAY

Join the waitlist before the 2026 launch

CoolAndLovely is pre-launch and in waitlist mode. If you are a brand, a creator or an enterprise with a specific requirement, tell us what it is while the roadmap is still being shaped.

- Join the waitlist and be first in line when access opens
- Book a walkthrough of the storefront, wardrobe and creator surfaces
- Bring your hardest requirement - checkout, styling, compliance - and get a straight answer on whether it exists yet
- For a fashion house or marketplace, start an enterprise conversation on quotas, tenancy and residency

JOIN THE WAITLIST

coolandlovely.com

BOOK A DEMO

coolandlovely.com/contact

READ THE USE CASES

coolandlovely.com/use-cases

HOW THE AI WORKS

coolandlovely.com/ai

TALK TO SALES

sales@coolandlovely.com
