

Kakhkhorov Shokhjakhon (Sean)

Full Stack Developer • DevOps Engineer • AI Engineer • Agentic Engineer

Full-stack developer with more than three years of professional experience. I work with MongoDB, PostgreSQL, MySQL and Redis, build typed GraphQL and REST APIs with NestJS and Express, and create responsive interfaces in React and Next.js, then deploy and maintain them myself.



Build end to end

Data models, APIs, auth and real-time features up to the UI



Deploy and operate

Docker, Nginx, DNS and TLS, with CI in GitHub Actions



Integrate AI

Gemini features grounded in site content, built with coding agents

TS TypeScript

React

Next.js

Node.js

NestJS

Express

GraphQL

Socket.io

MongoDB

PostgreSQL

MySQL

Redis

Docker

Nginx

GitHub Actions

Gemini API

Claude Code

Suwon, South Korea · kakhkhorov97@gmail.com · github.com/kakhkhorov97



● Latest project · creatorhub.site

What's inside .

01



About me

Background and how I work

p. 03

03



Skills

Core stack and other tools

p. 06

05



Interface

Discovery, profiles, services, events and sign-in

p. 10

07



AI finder

Gemini assistant grounded in site content

p. 22

09



Case studies

Duplicate requests, AI limits and self-healing releases

p. 29

11



Sofiya Bookstore

Earlier project: React and Express bookstore

p. 35

02



Experience

IT Local and Atlantis Investment

p. 04

04



CreatorHub

Latest project: overview and features

p. 07

06



Workflows

Inquiries, admin area, moderation and the worker

p. 17

08



Engineering

Architecture, security, testing, CI/CD and deployment

p. 24

10



Data model

How CreatorHub's data is structured

p. 32

12



Contact

Email, phone and links

p. 38

Full-stack development, from code to production •

I started as a backend developer building Express and MongoDB APIs, then worked as a full-stack developer at IT Local from June 2023 to January 2026, in a team of four or more developers serving more than 20 client companies. There I looked after everything from database design and authentication to deployments, DNS and outages, which taught me to care about how code behaves after release, not just how it is written.

My latest project, **CreatorHub**, is live at creatorhub.site. I built both the frontend and the backend on my own, and added a Gemini-powered assistant that answers visitors' questions using the site's public content.

I use coding agents such as Claude Code and Codex within written repository rules: I approve each change, and automated checks run before anything is merged.



Reliability

Monitoring, logs and quick outage fixes



Security

Auth, input validation and code review



Documentation

APIs and deploy steps written down



AI workflow

Coding agents with review gates



Kakhkhorov Shokhjakhon (Sean)

Full Stack Developer · DevOps Engineer
AI Engineer · Agentic Engineer

— EXPERIENCE

Full-stack Web Developer

IT Local · 2023.06 – 2026.01

Backend Developer

Atlantis Investment · 2022.05 – 2022.12

— BASED IN

Suwon, Gyeonggi-do, South Korea

— FIND ME

github.com/kakhkhorov97

kakhkhorov97@gmail.com

Where I have worked .

Full-stack Web Developer

2023.06 - 2026.01

IT Local · Uzbekistan

IT services company offering web development, server and computer assembly, network configuration and security-camera installation.

- ◆ Built and maintained websites for 20+ client companies in construction, retail, food service, corporate and public sectors, in a team of 4+ developers
- ◆ Developed APIs and server-side business logic with authentication, authorization and input validation
- ◆ Designed database schemas; improved page and API response times through indexing and query optimization
- ◆ Set up and secured 5–15 Ubuntu servers, including SSH access and firewall rules
- ◆ Ran Node.js services with PM2 and Docker behind Nginx, using Cloudflare for DNS and TLS
- ◆ Moved deployments from manual steps to CI-based automated deployment
- ◆ Monitored application health, analyzed logs and resolved outages and performance issues
- ◆ Reviewed teammates' code; documented APIs and deployment procedures
- ◆ Built responsive pages with React, Bootstrap and Tailwind CSS
- ◆ Onboarded my replacement and handed over every system before leaving

Backend Developer

2022.05 - 2022.12

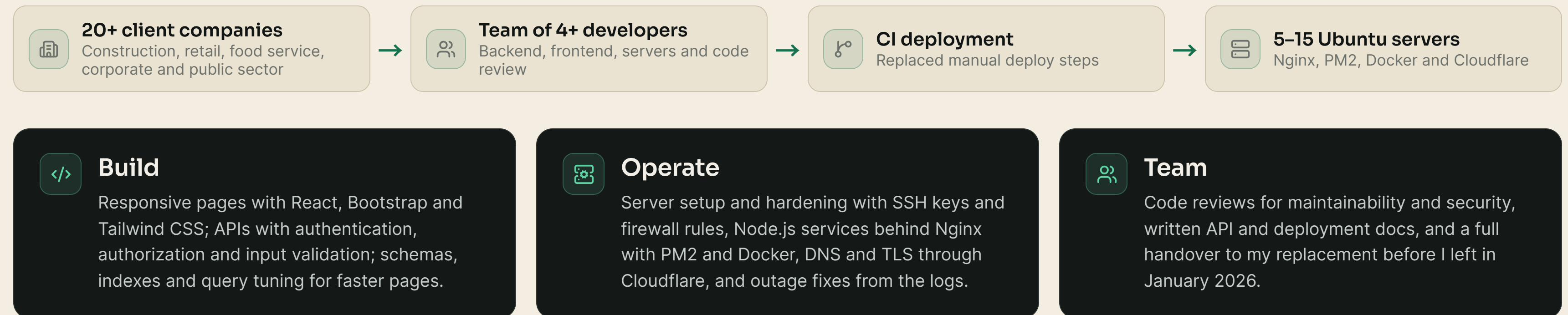
Atlantis Investment · Uzbekistan

Alongside my university studies.

- ◆ Built RESTful APIs with Express.js between the frontend and backend
- ◆ Designed and maintained MongoDB databases, using Mongoose for data modeling and access
- ◆ Applied MVC architecture to separate concerns and keep the code easy to maintain and scale
- ◆ Implemented sign-up and login with sessions and JWT, hashing passwords with bcryptjs
- ◆ Built file upload endpoints with Multer for images and documents
- ◆ Validated request data and added consistent error handling and status codes
- ◆ Wrote MongoDB queries and indexes for search, filtering and pagination
- ◆ Tested and documented endpoints in Postman and agreed request and response formats with the frontend

Client websites, from build to servers .

IT Local builds and runs websites for other businesses. In a team of four or more developers I worked across the whole path, from the first page to the server it runs on.



Client sites are not shown here: they belong to IT Local's customers.

My stack

CORE, USED DAILY ALSO USED

 Frontend

React

Next.js

TypeScript

JavaScript

Apollo Client

Redux Toolkit

MUI

Tailwind CSS

SCSS

HTML

CSS

jQuery

Axios

TanStack Query

React Router

Socket.IO Client

Framer Motion

Three.js

Swiper

Anime.js

Typed.js

TUI Editor

Bootstrap

Chakra UI

shadcn/ui

Figma

 Backend

Node.js

NestJS

Express

GraphQL

REST APIs

Socket.io

JWT & OAuth

Role-based access

Python

EJS

express-session

bcryptjs

Multer

 Databases

MongoDB

Mongoose

PostgreSQL

MySQL

Redis

Indexing & query tuning

Transactions

MongoDB Compass

 DevOps

Docker

Docker Compose

Nginx

GitHub Actions

Cloudflare

DNS & TLS

PM2

Firewall configuration

Git & GitHub

GitHub Pages

Render

 AI & agents

Gemini API

OpenAI API

Claude Code

Codex

Cursor

Content retrieval

 Testing & tools

Jest

Supertest

ESLint

Prettier

Integration tests

API contract tests

PostHog

Postman

Yarn

npm

NVM

— LATEST PROJECT

CreatorHub

🧑 Solo project: I designed, built and deployed the frontend, backend and infrastructure.

A portfolio-discovery platform where visitors find independent creators across Korea, explore their projects and services, and send private project inquiries.

- ◆ Search by category, location and remote work
- ◆ Creator workspace with admin approval
- ◆ Private inquiries with live chat

FRONTEND

Next.js React TypeScript

BACKEND

NestJS GraphQL Socket.io

DATA

MongoDB Mongoose

INFRA

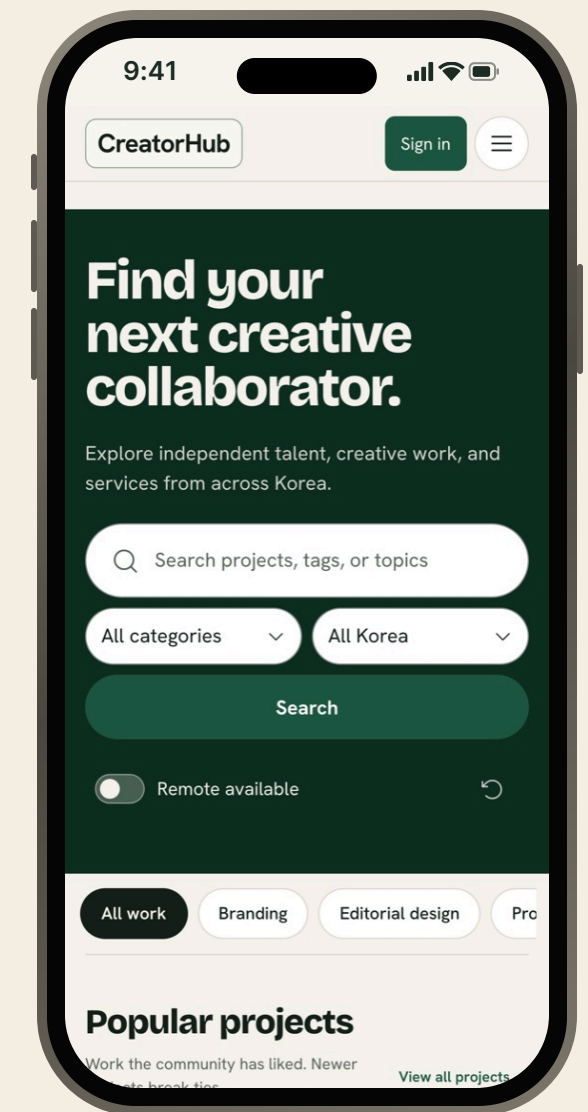
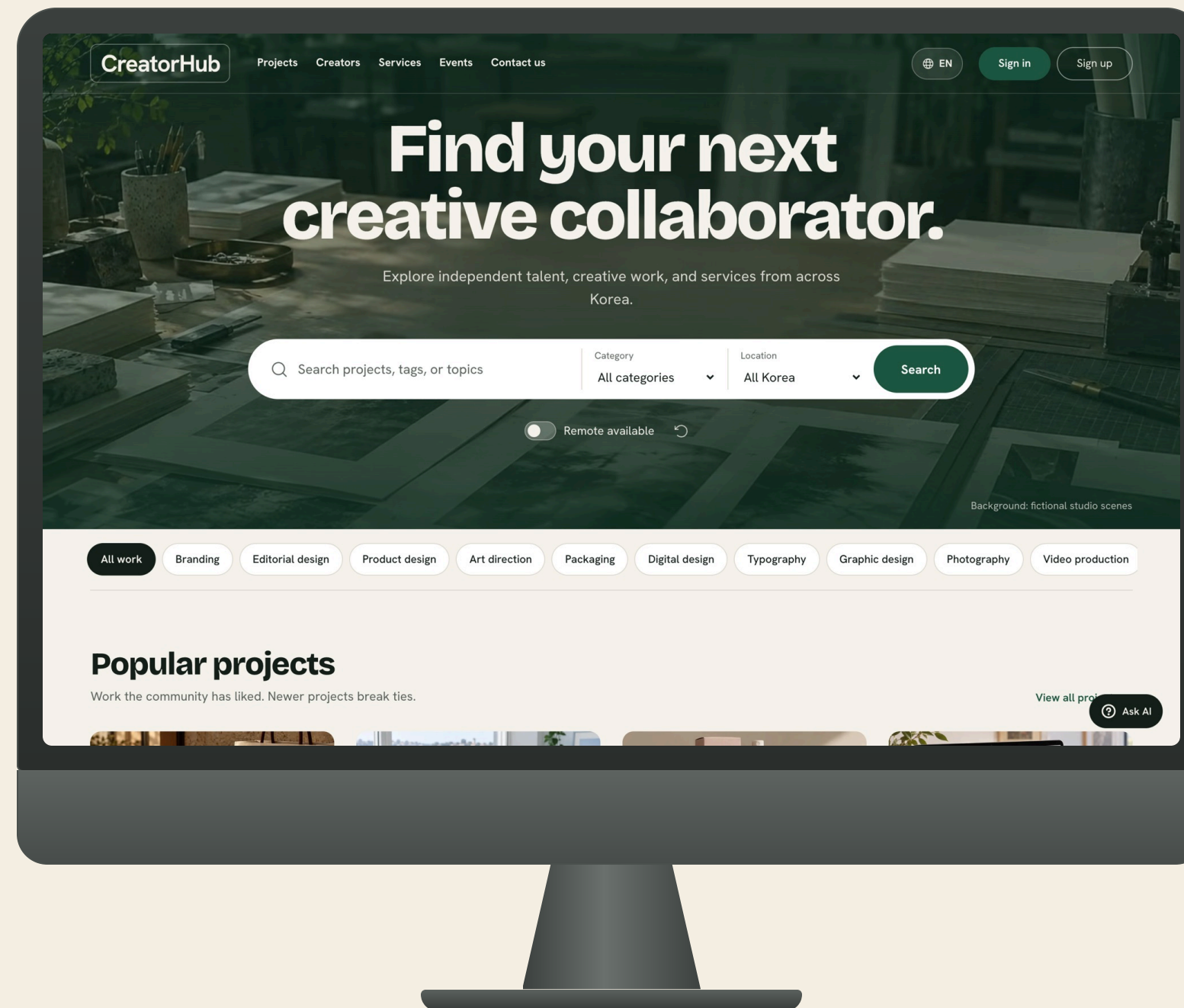
Docker Nginx GitHub Actions

AI

Gemini API

[Visit live site ↗](#)

Repos: [CreatorHub](#) · [CreatorHub-next](#)



A tour of the live site

creatorhub.site/projects

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

Motion

Explore creative work

Q Search projects, tags, or topics

Category

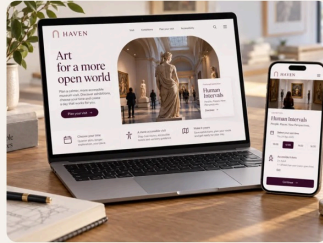
All categories

Location

All Korea

Search

☐ Remote available



Projects

Search and filter all published work

creatorhub.site/creators/leila-hassan

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

Motion

Creators / Leila Hassan



Leila Hassan

Clear digital experiences built around everyday needs

Incheon

Remote available

Available

Follow

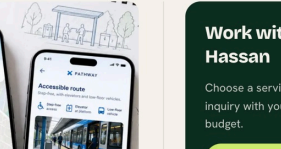
Send inquiry

Work

Services

About

Reviews



Work with Leila Hassan

Choose a service or send a general inquiry with your brief, deadline and budget.

Creator profile

Portfolio, services and an inquiry button

creatorhub.site/services/...

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

Motion

←

Motion direction and product visualization



Oliver Reed

About this service

A clearly scoped creative offering covering Motion design, 3D visualization, Lighting. Scope, schedule and final pricing are discussed through a private inquiry. Fictional portfolio service.

What's included

Visual treatment

Preview frames

Final high-resolution images

From ₩750,000

Discuss the scope and timing directly with the creator.

Send inquiry

Service page

Pricing, scope and what is included

creatorhub.site/events

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

Motion

Meet through creativity

Find creative events across the community. Fictional event concepts are labeled and are not open for registration.

Upcoming events

Workshops, conversations, and opportunities to meet.



Events

Workshops and community events

creatorhub.site

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

Motion

Find your next creative collaborator.

Explore independent talent, creative work, and services from across Korea.

Q Search projects, tags, or topics

Category

All categories

Location

All Korea

Search

☐ Remote available

All work

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Help

Find creative work and get answers

Hi! What can I help you find?

I can help you discover creators, projects and services, and explain how to use CreatorHub. What would you like to do?

Find a creator

Explore projects

Publish my work

Answers are generated automatically and may be incomplete. Check the linked profiles and services.

AI finder

Gemini assistant that answers from site content

creatorhub.site/_admin

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

Motion

Administration / Overview

Admin Administrator

MANAGE

Overview

Users

Creators

Projects

Services

Categories

Events

Content approval

Comments

Website content

Website overview

Manage the people, work and content behind CreatorHub.

Users

27

Manage users

Creators

20

Manage creators

Projects

37

Manage projects

Services

29

Manage services

Comments

1

Manage comments

Categories

16

Manage categories

Events

6

Manage events

Recently updated projects

All projects -->

Morrow Cafe — Website direction

Botanical Studio

Published

Dayplan — Activity interface

Botanical Studio

Published

Fern — Botanical identity direction

Studio Fern

Published

HAVEN — Museum visit planning experience

Leila Hassan

Published

Publishing & settings

Keep public content current. Draft, hidden and removed records remain available in their management pages.

Publish or archive projects

Manage community events

Edit website copy & contact

Review account access

Admin area

Moderation and site management

SEAN K. — 08

What the platform does

For visitors

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

More

Explore creative work

Q Search projects, tags, or topics

Category

All categories

Location

All Korea

Search

☐ Remote available

Discovery

Search by category, location and remote work

AI finder

Answers drawn from the site's own content

Community

Like, save, comment and follow creators

Sign-in options

Email, Google, Telegram or Kakao

Three languages

English, Korean and Uzbek

For creators

CreatorHub

Projects

Creators

Services

Events

Contact us

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

More

Creators / Leila Hassan

Leila Hassan

Clear digital experiences built around everyday needs

Incheon

Remote available

Available

Send inquiry

Follow

Work

Services

About

Reviews

Work with Leila Hassan

Choose a service or send a general inquiry with your brief, deadline and budget.

Workspace

Manage profile, projects and services

Private inquiries

Client briefs with live chat for two

Reviews

One rating per completed inquiry

Follower alerts

Followers hear about new work

For admins

CreatorHub

Administration

EN

Sign in

Sign up

Branding

Editorial design

Product design

Art direction

Packaging

Digital design

Typography

Graphic design

Photography

Video production

Illustration

Web development

UI/UX design

More

Administration / Overview

Admin

Administrator

Website overview

Manage the people, work and content behind CreatorHub.

Open website →

Users

27

Manage users →

Creators

20

Manage creators →

Projects

37

Manage projects →

Services

29

Manage services →

Comments

1

Manage comments →

Categories

16

Manage categories →

Events

6

Manage events →

Recently updated projects

All projects →

Morrow Cafe — Website direction

Artesian Studio

Published

Dayplan — Activity interface

Artesian Studio

Published

Fern — Botanical identity direction

Studio Fern

Published

HAVEN — Museum visit planning experience

Artesian Studio

Published

Publishing & settings

Keep public content current. Draft, hidden and removed records remain available in their management pages.

Publish or archive projects →

Manage community events →

Edit website copy & contact →

Review account access →

Approval

New work goes live after review

Safe edits

Changes stay hidden until approved

Users and roles

Manage accounts, roles and comments

Site content

Categories, events and site copy

Explore creative work

Visitors search all published work and narrow it by category, city or remote availability.

— HOW IT'S BUILT

- ◆ One MongoDB aggregation filters, sorts and pages the results, and returns the total count in the same query
- ◆ Only work from published creators with active accounts is shown
- ◆ Search text is escaped before it reaches a case-insensitive match
- ◆ Search, filters and page number live in the URL, so Back returns to the same spot

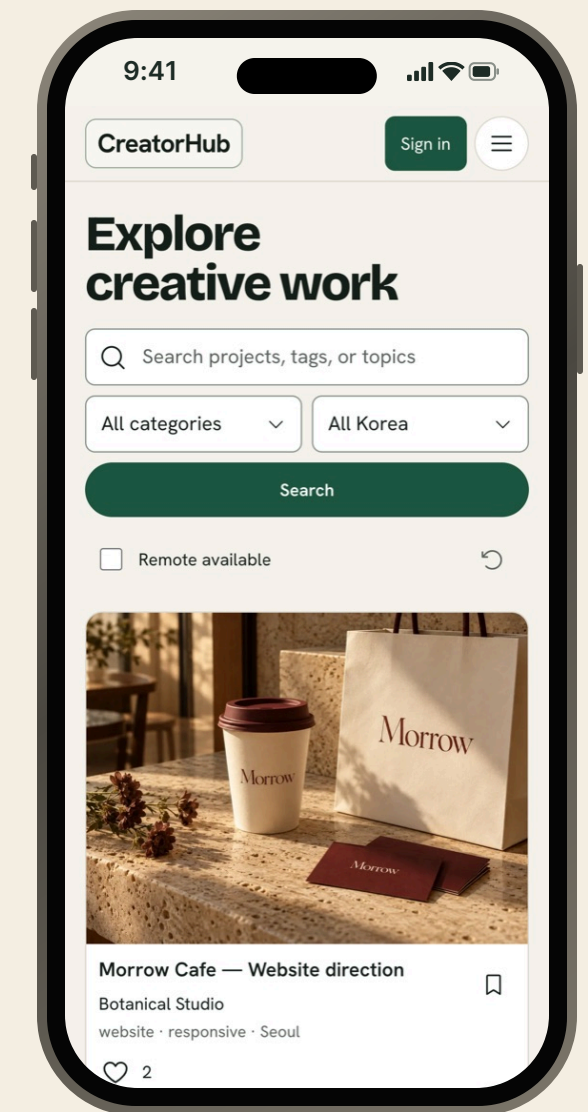
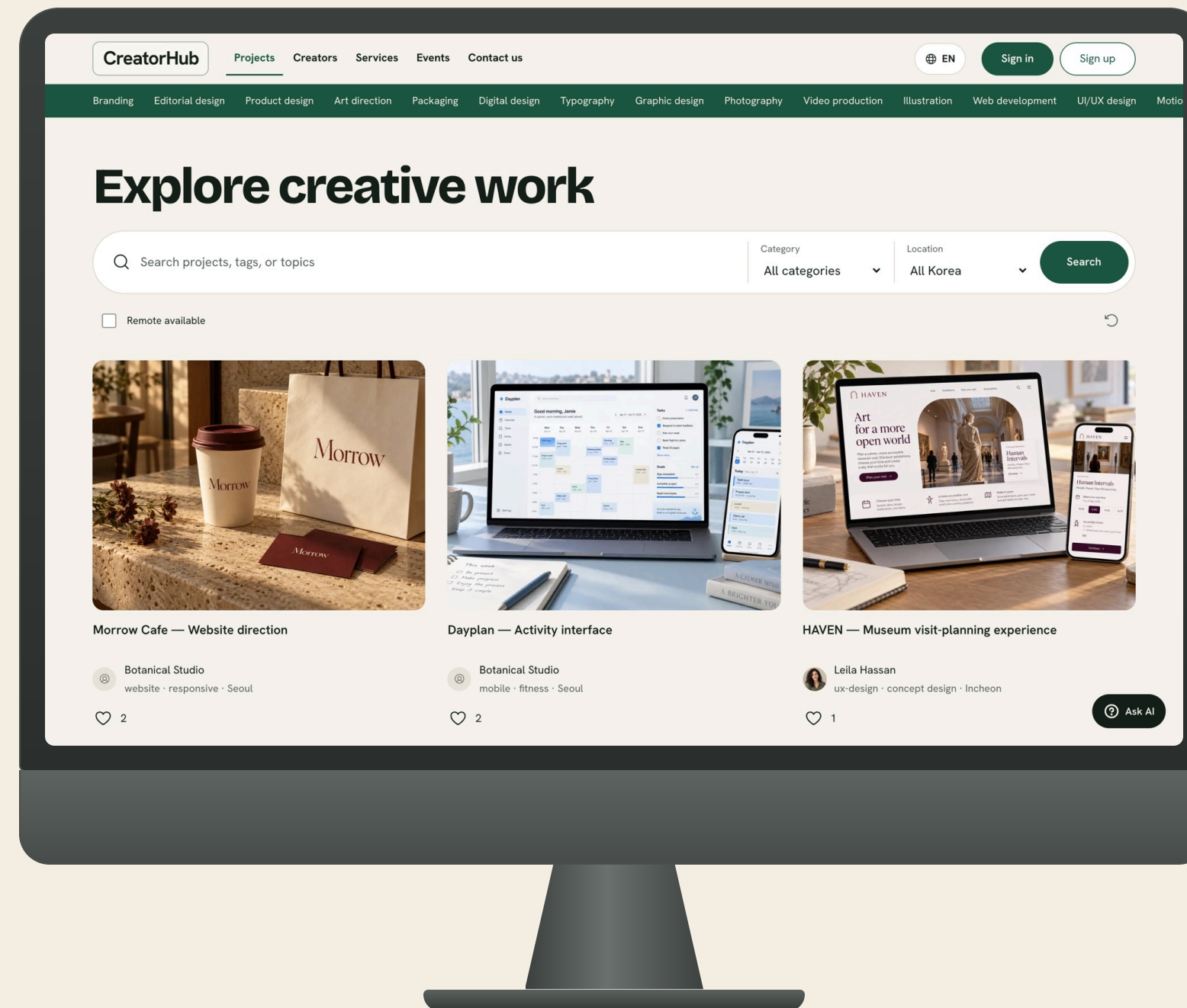
🌿 MongoDB aggregation

🔗 GraphQL

🔗 Apollo Client

🔗 Next.js

creatorhub.site/projects ↗



Project page

Each project has its own page with a gallery, tags, likes and a button to send the creator a private inquiry.

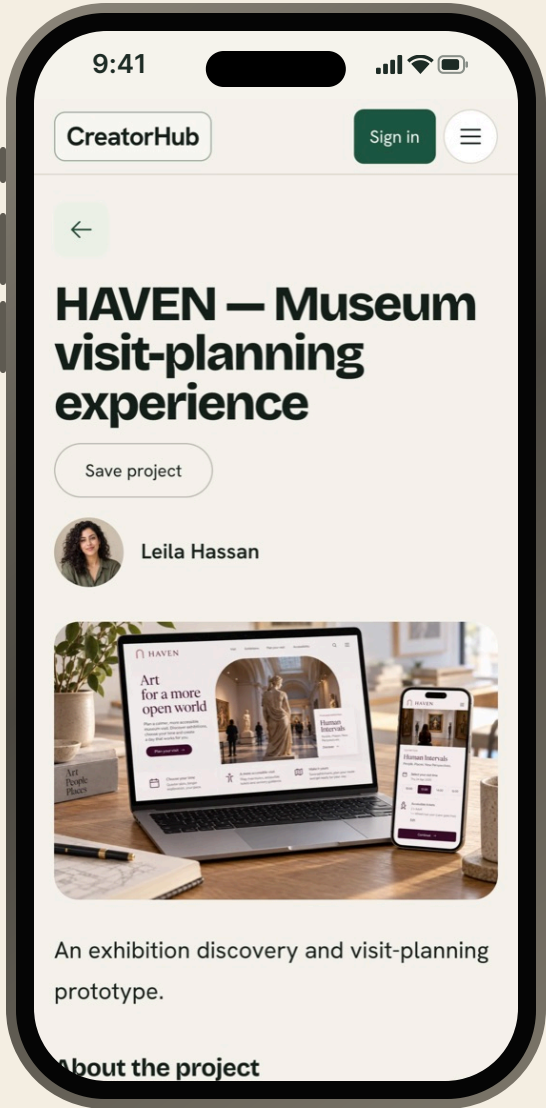
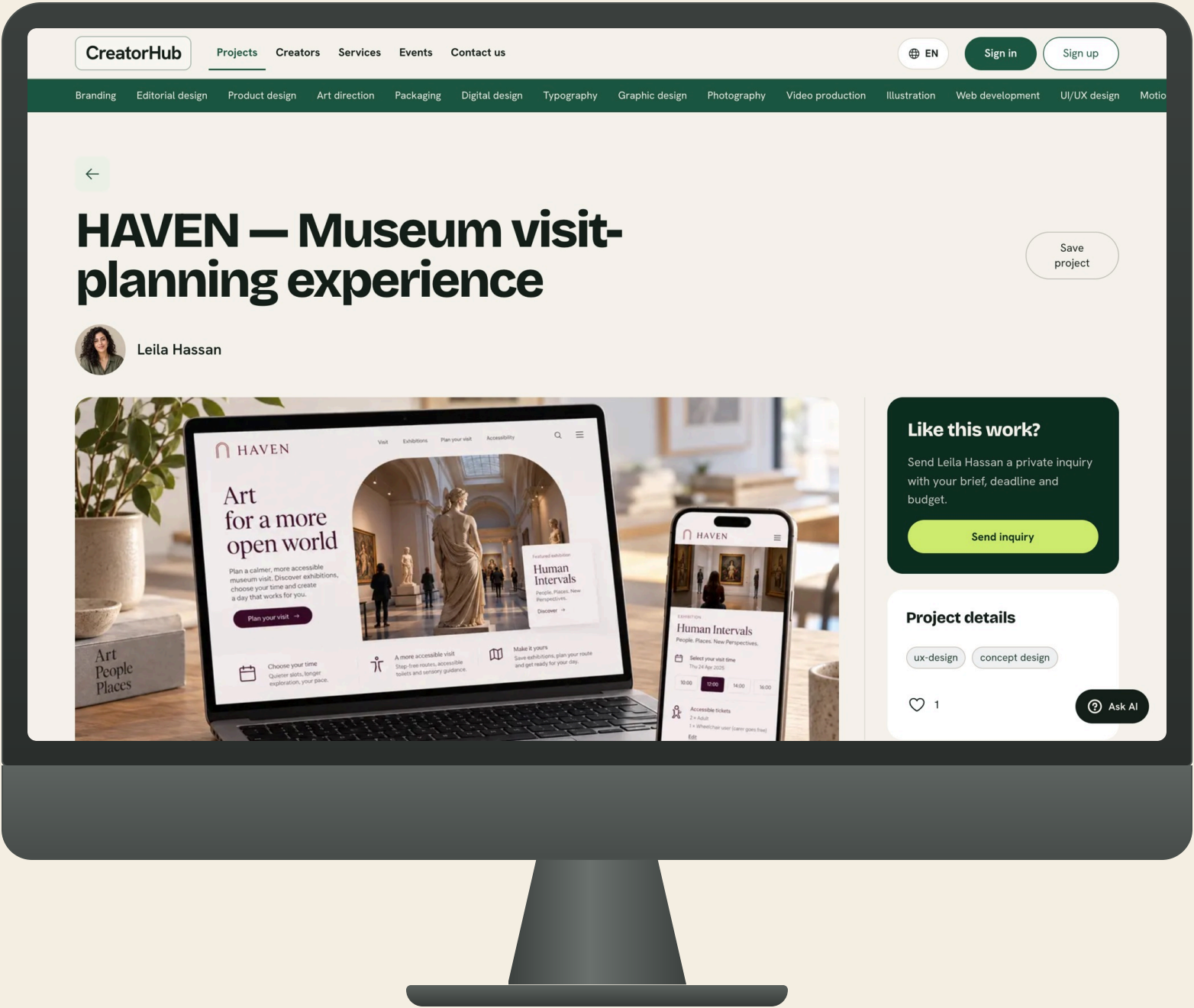
— HOW IT'S BUILT

- ◆ The server renders preview tags (title, summary, cover) for shared links, and unknown IDs return a real 404
- ◆ Likes are stored once per member and project, and recorded in the same transaction as the activity event
- ◆ Members cannot like their own work

Next.js SSR GraphQL

MongoDB transactions

Open this project ↗



Creator profile

A public profile with headline, city and availability, plus tabs for work, services, reviews and about.

— HOW IT'S BUILT

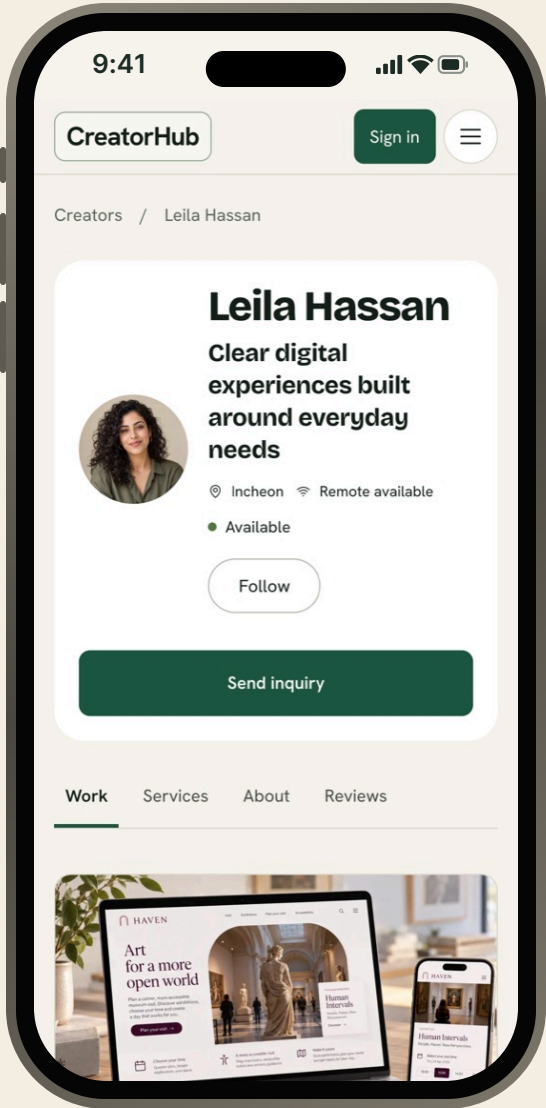
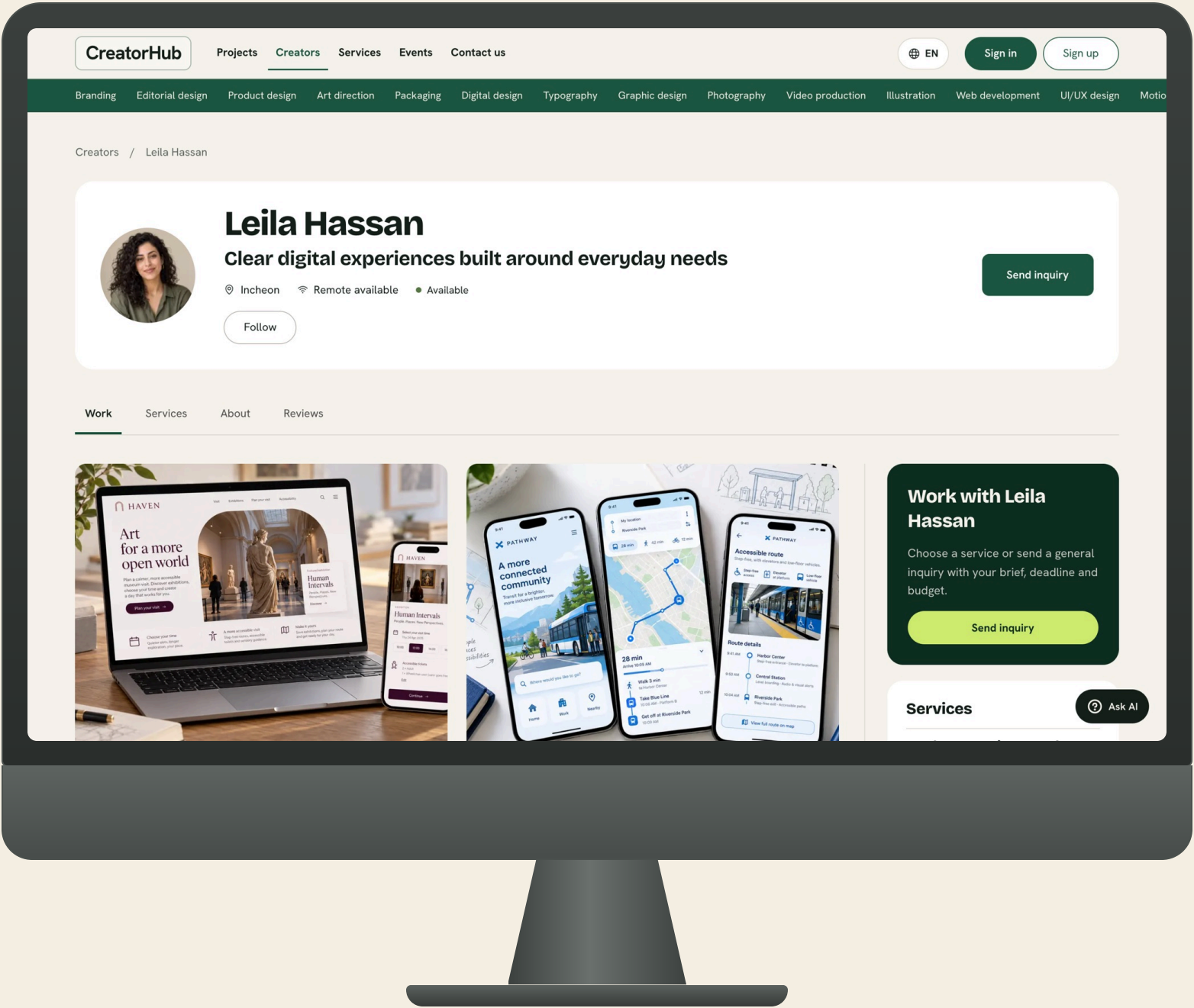
- ◆ Profiles are found by a readable slug instead of a database ID
- ◆ Only published profiles of active accounts can be opened
- ◆ Following uses a transaction and reactivates an old follow instead of duplicating it
- ◆ Signed-out visitors are sent to sign-in and brought back to the same profile

 GraphQL

 MongoDB

 Next.js SSR

[Open this profile ↗](#)



Creators and services

A directory of creators and a marketplace of services, filtered by category, city, remote work and budget.

— HOW IT'S BUILT

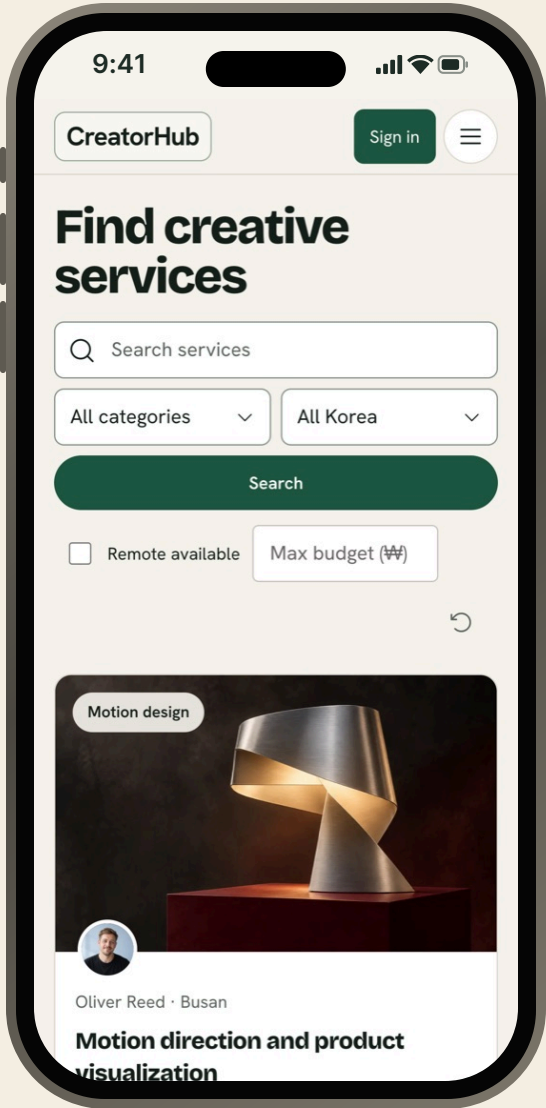
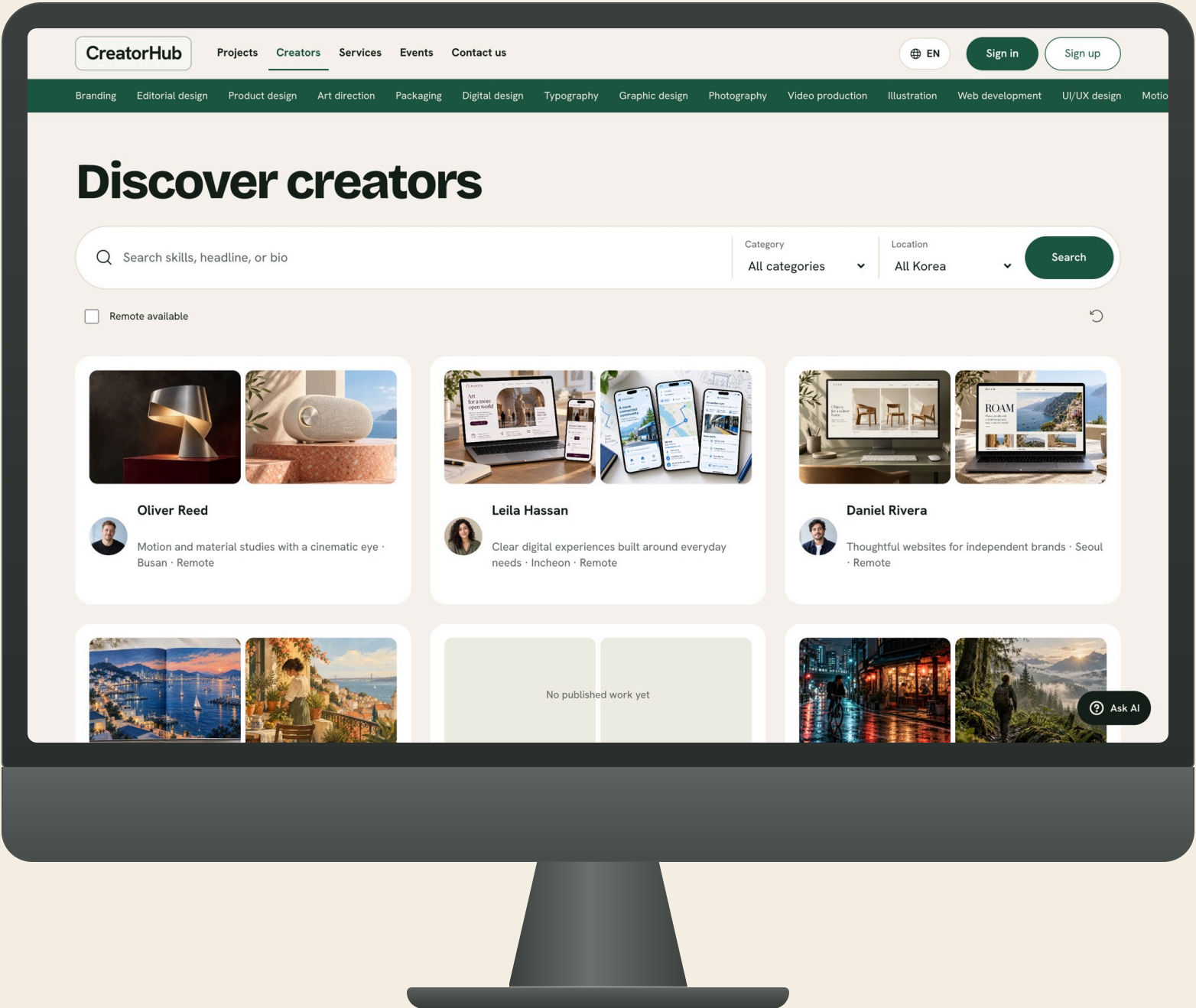
- ◆ The budget filter keeps services under the limit and every custom-quote service
- ◆ Creator cards show the three newest covers, looked up only for the rows on the current page
- ◆ Both lists share the same search and visibility rules as the projects page

MongoDB aggregation

GraphQL

i18n

creatorhub.site/creators



Service page

Scope, deliverables and a starting price in won, or a custom quote. The final terms are agreed through a private inquiry.

— HOW IT'S BUILT

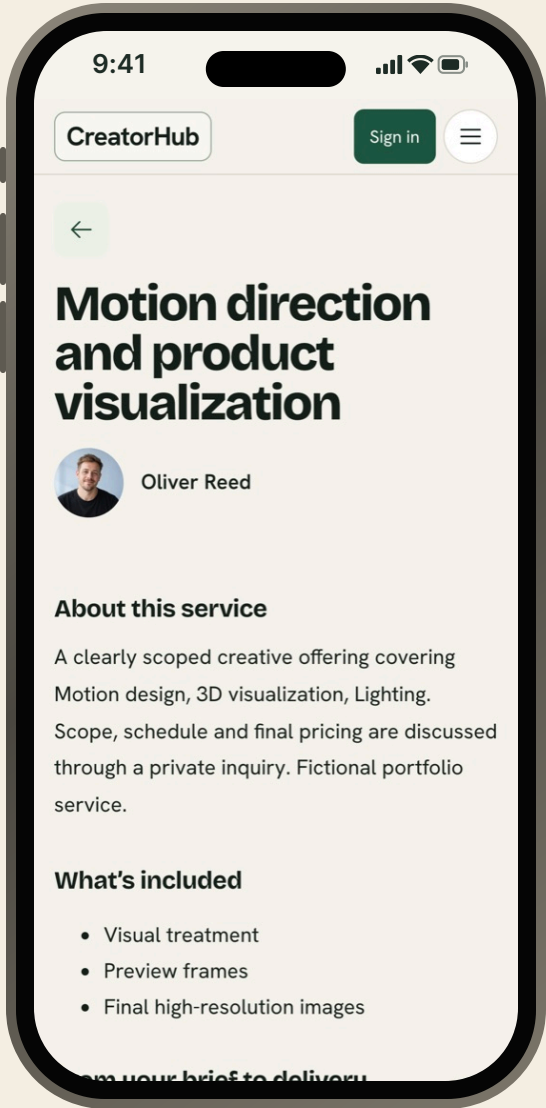
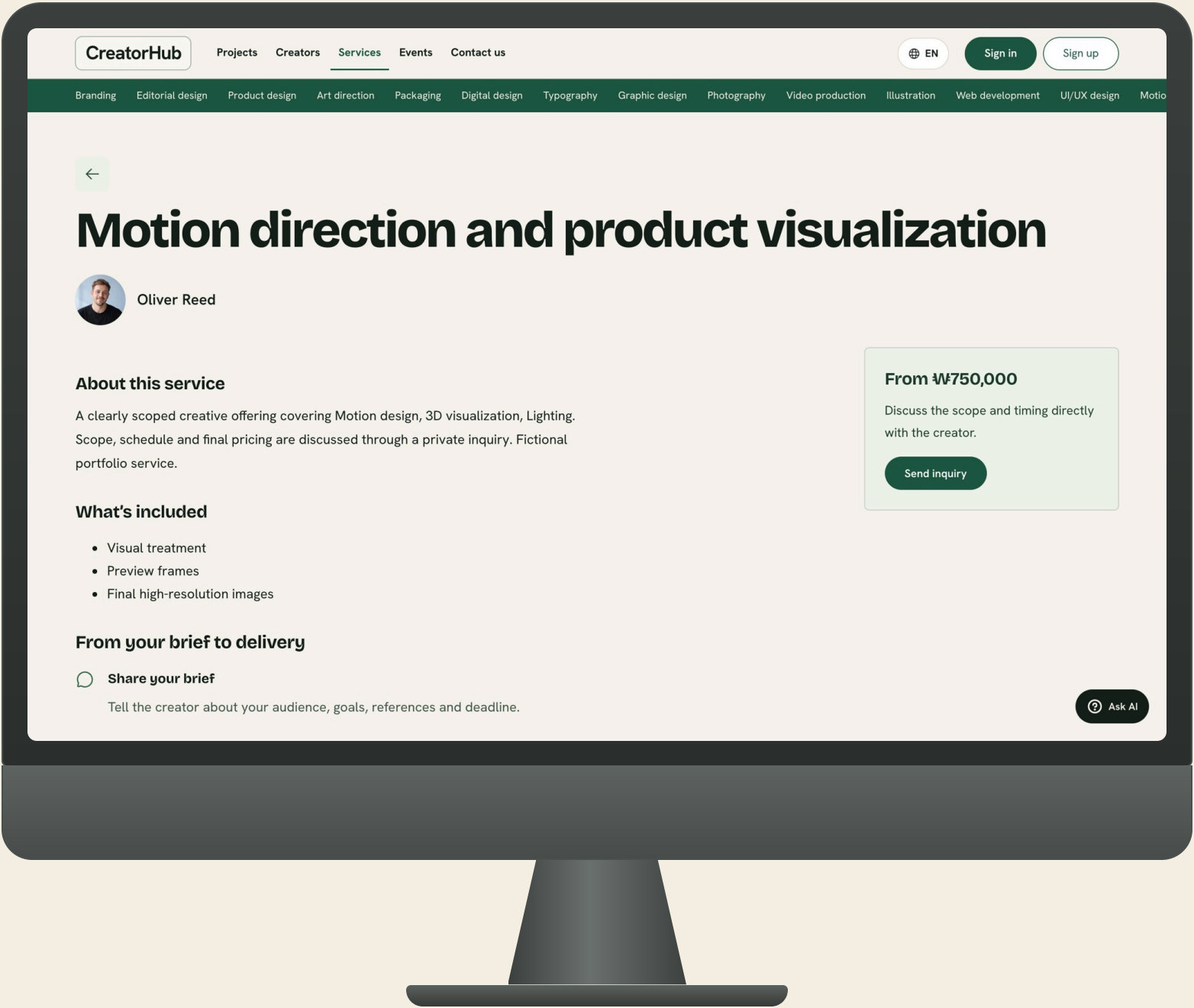
- ◆ Two pricing types: a fixed starting price or a custom quote
- ◆ When an inquiry is submitted, the service title, price, timeline and deliverables are copied into it
- ◆ Later edits to the service never change terms a client already sent

 GraphQL

 MongoDB

Embedded snapshot

Open this service ↗



Creative events

Workshops, talks and meetups for the creator community, managed by admins. Demo events are labelled as fictional.

— HOW IT'S BUILT

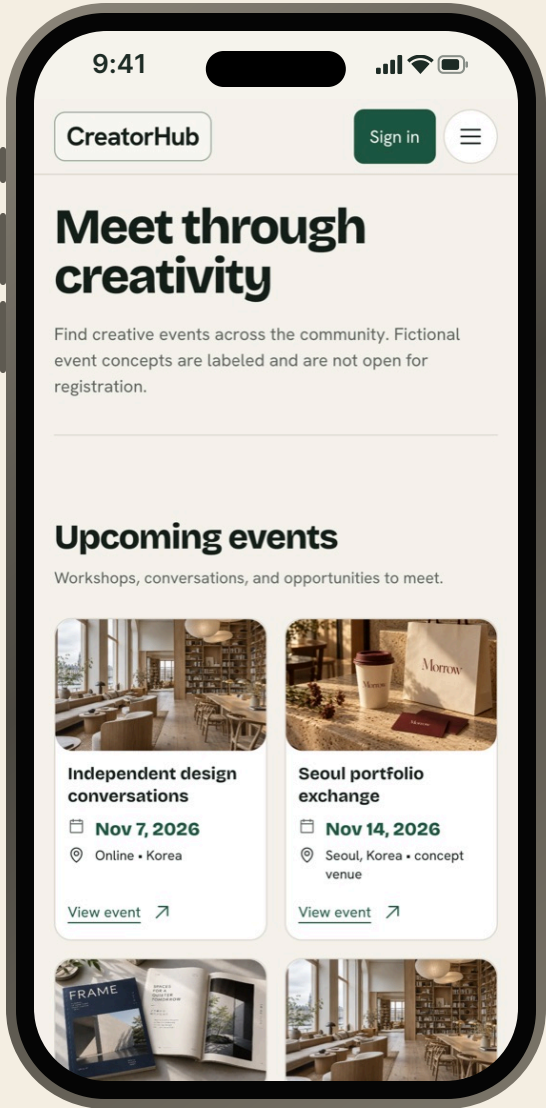
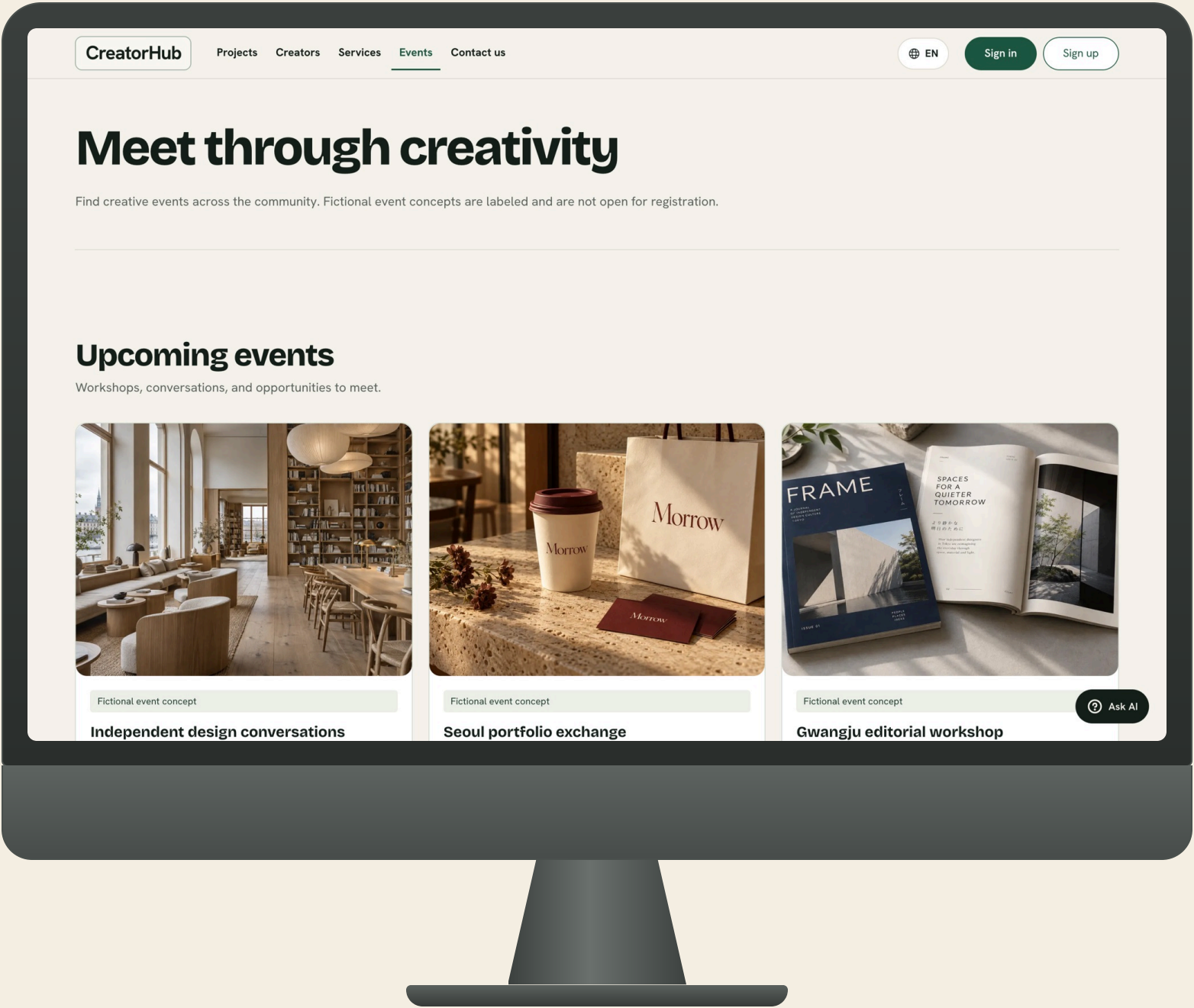
- ◆ Only published, upcoming events in a Korean city or online are listed, soonest first
- ◆ Admins must give an end time after the start and a plain HTTPS link
- ◆ A demo flag marks fictional sample events

GraphQL

MongoDB

Admin area

creatorhub.site/events ↗



Sign-in options

Email and password, or continue with Google, Kakao or Telegram. New social accounts add a name and phone to finish.

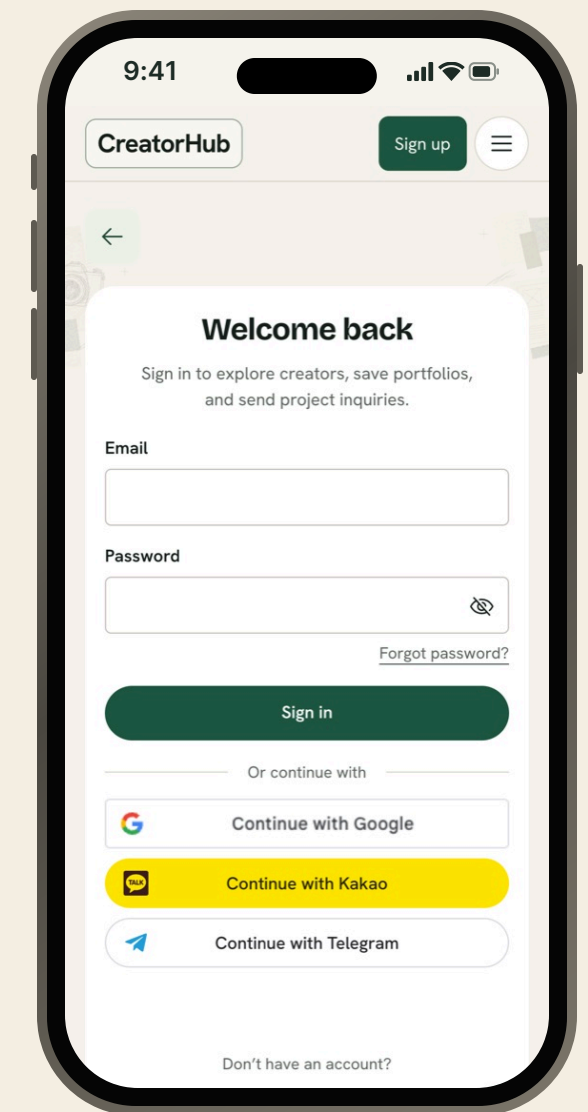
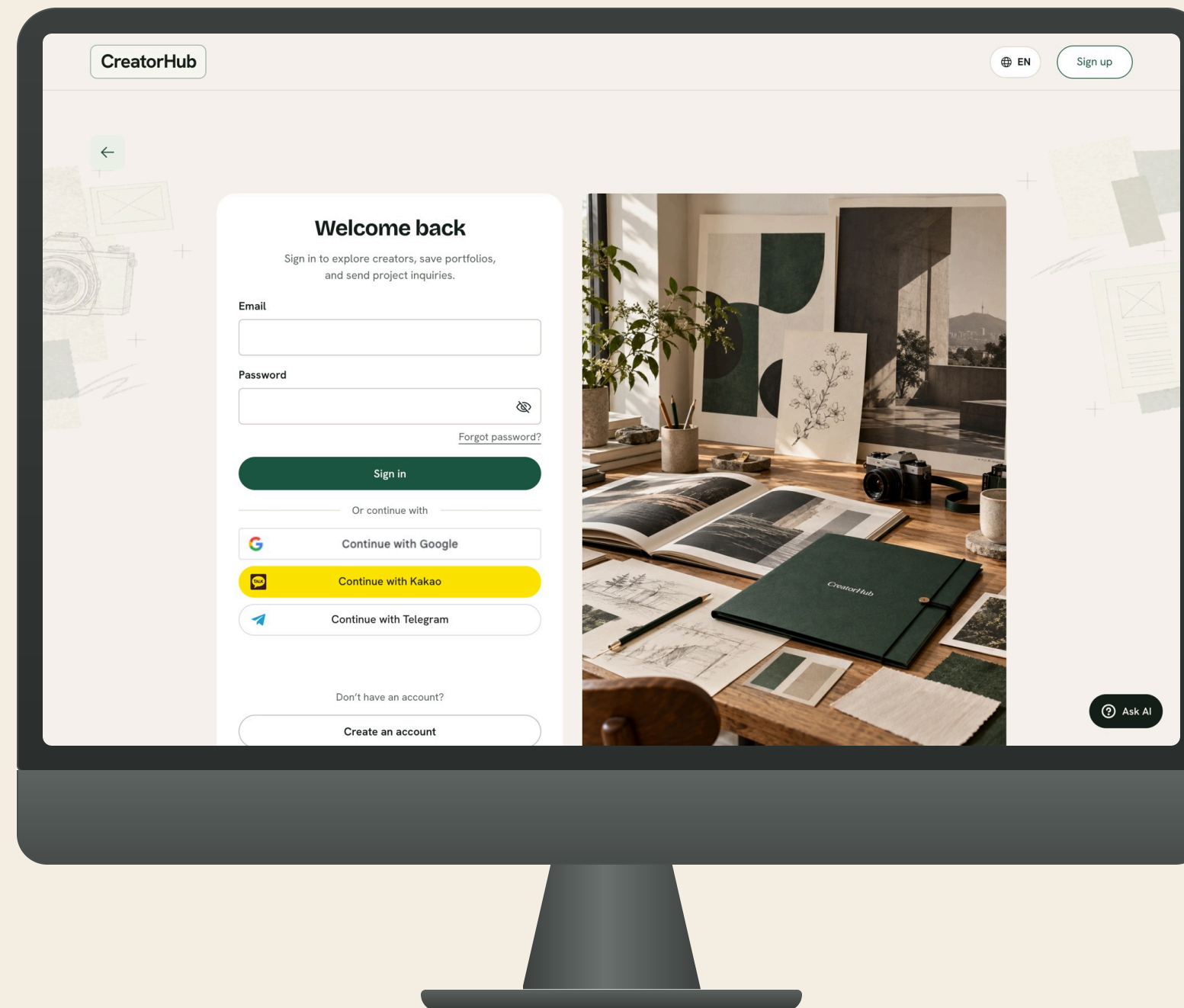
— HOW IT'S BUILT

- ◆ Google, Kakao and Telegram sign-ins are checked against each provider's public keys; Kakao also uses one-time state and nonce values
- ◆ Passwords are hashed with bcrypt; access tokens last 15 minutes
- ◆ Refresh tokens rotate on every use and are stored only as hashes

JWT OAuth / OpenID bcrypt Kakao

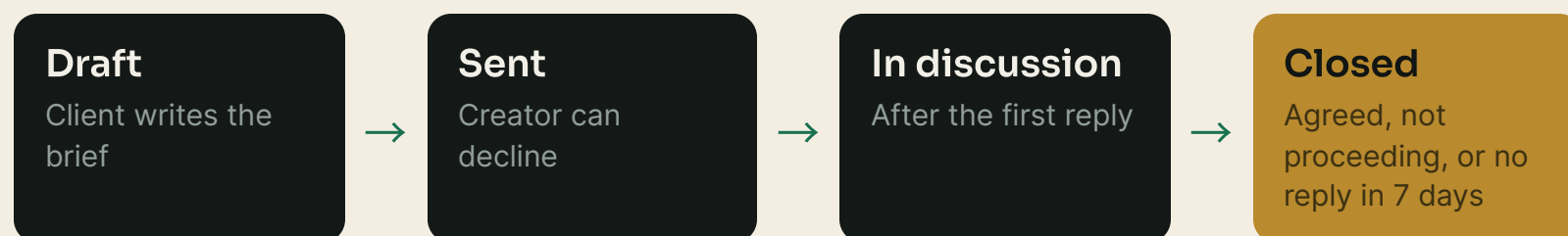
Telegram

creatorhub.site/account/sign-in ↗



Private inquiries and live chat

Clients send a creator a brief with budget and deadline, then talk it through in a private chat.



Private by design

Only the client and the creator can open an inquiry. Creators never see drafts.

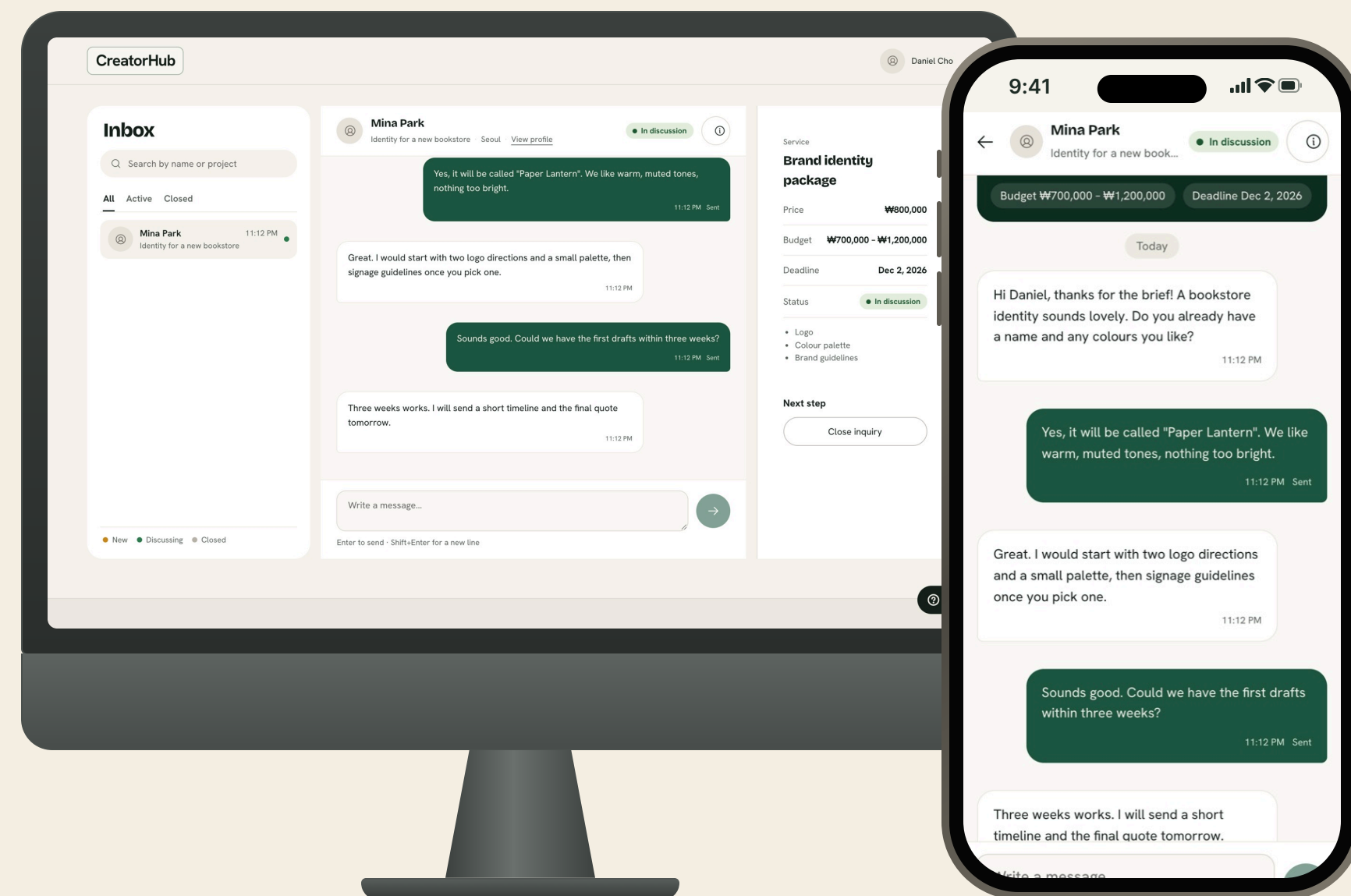
No duplicates

A double-click or network retry never creates a second inquiry or message.

Live chat

Socket.io rooms use the same access check. The database stays the source of truth.

Captured from the current app running locally, with fictional accounts and messages.



Admin: overview and users

A separate admin area shows totals for every record type and lets admins manage accounts and roles.

— HOW IT'S BUILT

- ◆ Every admin query is behind a server-side role guard, not just a hidden menu
- ◆ Suspending or deleting an account ends all of its sessions in the same transaction
- ◆ Admins cannot change their own role or status, so an admin cannot lock themselves out
- ◆ Edits check the record has not changed since it was opened, so two admins cannot overwrite each other



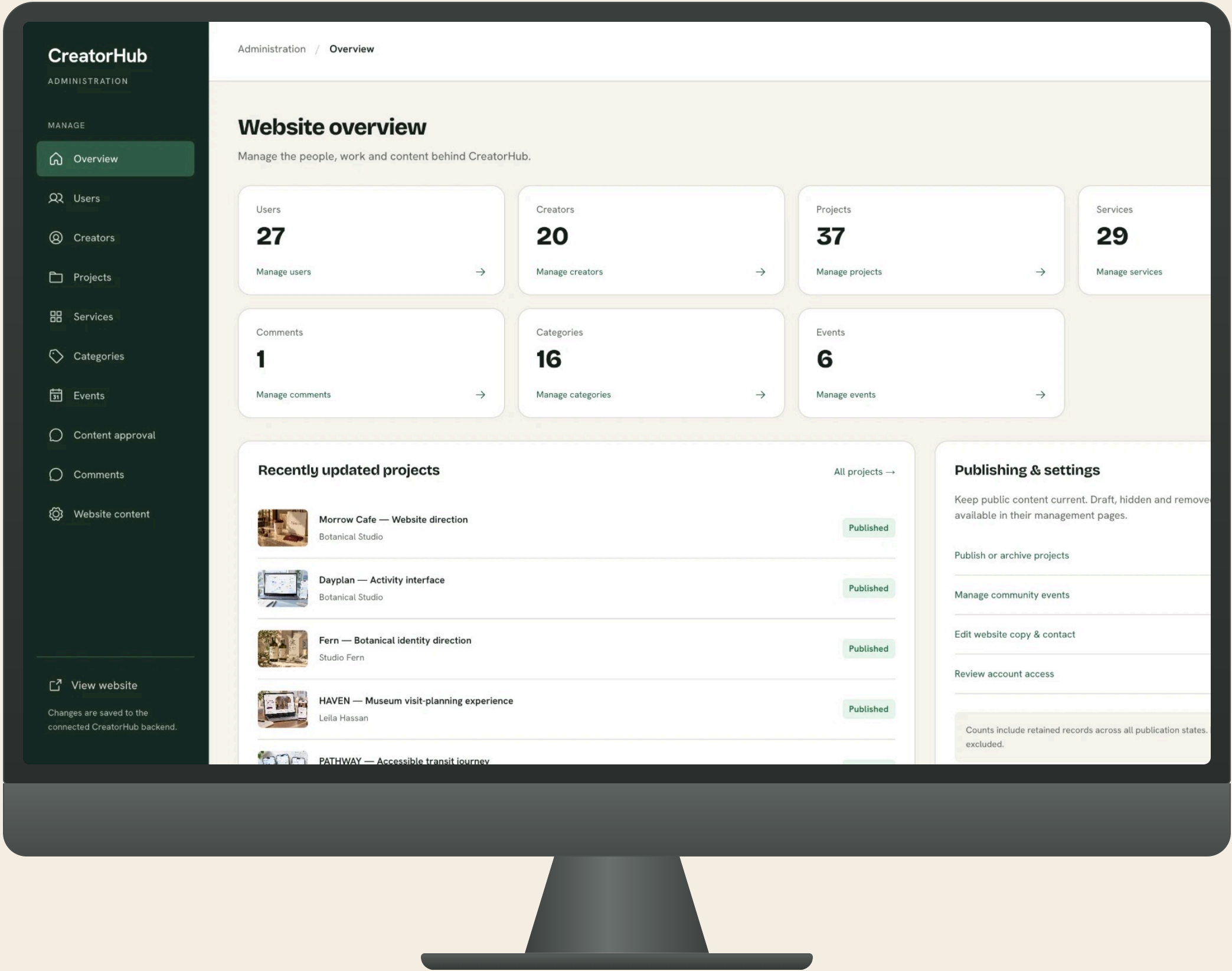
Role guard

Session revocation

Optimistic concurrency



MongoDB transactions



Admin: moderation and content

Admins review every creator, project and service, including hidden and removed ones, and edit the site copy.

— HOW IT'S BUILT

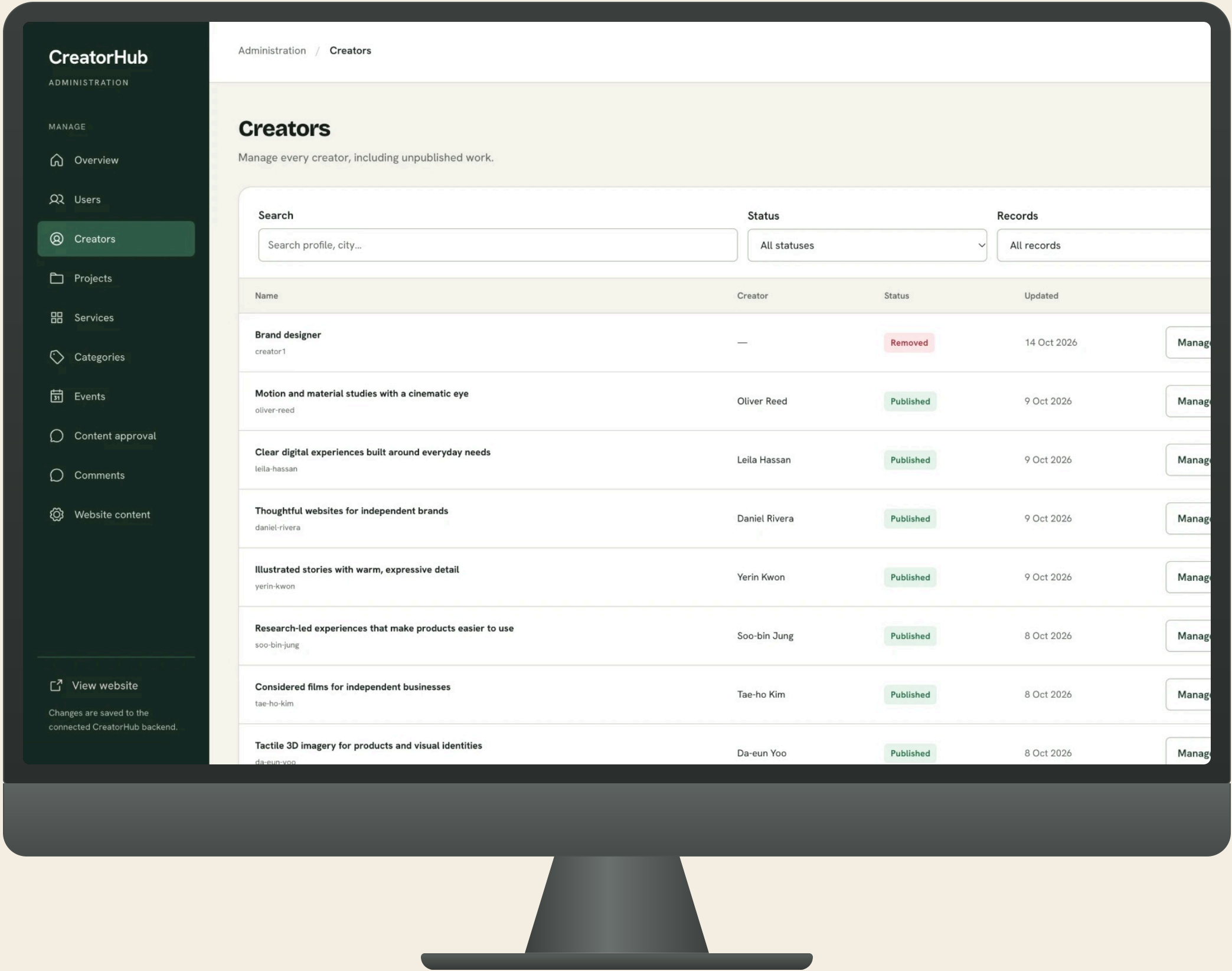
- ◆ Admins can hide a comment and show it again later; comments deleted by their author stay deleted
- ◆ Removing is a soft delete, and restoring never republishes anything automatically
- ◆ Homepage, About and footer text are stored in the database, so copy changes go live without a redeploy

Content review

Soft delete

 GraphQL

 MongoDB



Creator workspace and moderation

Creators edit their work in a private workspace. Nothing goes public until an admin approves it.

- 1 Creator saves an edit.** The live version stays online, untouched.
- 2 A pending review is stored,** one per item. Saving again replaces it.
- 3 Admin approves or rejects.** A rejection needs a written reason.

Publishing checklist

Category, description, cover and media are required, and checked again on approval.

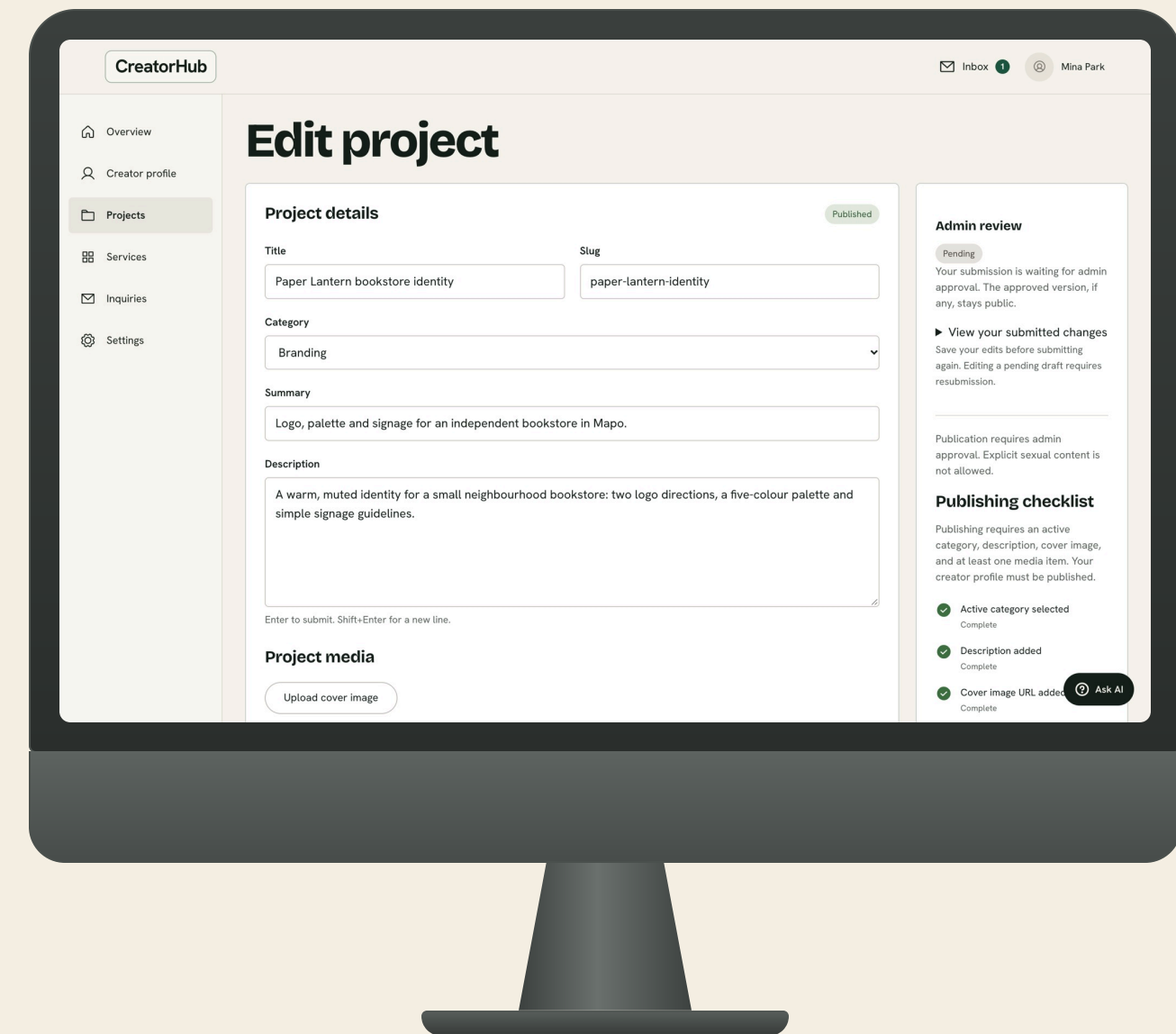
Admin-only decisions

The admin role is checked on the server, not only in the interface.

No stale approvals

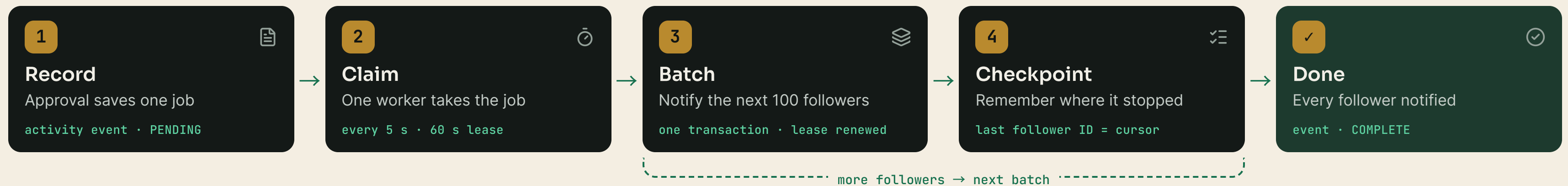
Approval fails if the item changed after the review was saved.

Captured from the current app running locally, with fictional accounts and content.



Follower notifications that survive a worker restart

Problem. When a creator publishes new work, every follower gets a notification. If the worker crashes halfway through, some followers could be notified twice, or not at all.



⚡ Crash or error? An error releases the job for a retry in 30 s; after a crash the 60 s lease simply runs out. Either way, the next claim resumes from the saved cursor.



No duplicates

Each notification is unique per follower and event, so a repeated batch cannot notify anyone twice.



Known limit

Polling MongoDB avoids a separate queue service but adds a few seconds of delay. Retries have no cap yet; next step is to stop after a set number and flag the event for an admin.



Verified

Integration tests cover a rolled-back batch that retries without duplicates, and followers notified only on first publication.

Scope: in-app notifications only, at most one per follower per event. No email or push on this path.

AI finder with Gemini .

An assistant on every page. Visitors describe what they need in plain words and get an answer with links to matching creators and services.

💬 *"I need a UI/UX designer who can work remotely. Who should I look at?"*

- ◆ Answers in English, Korean or Uzbek, matching the site
- ◆ Each answer links to the matching search results
- ◆ Side panel on desktop, full screen on phones

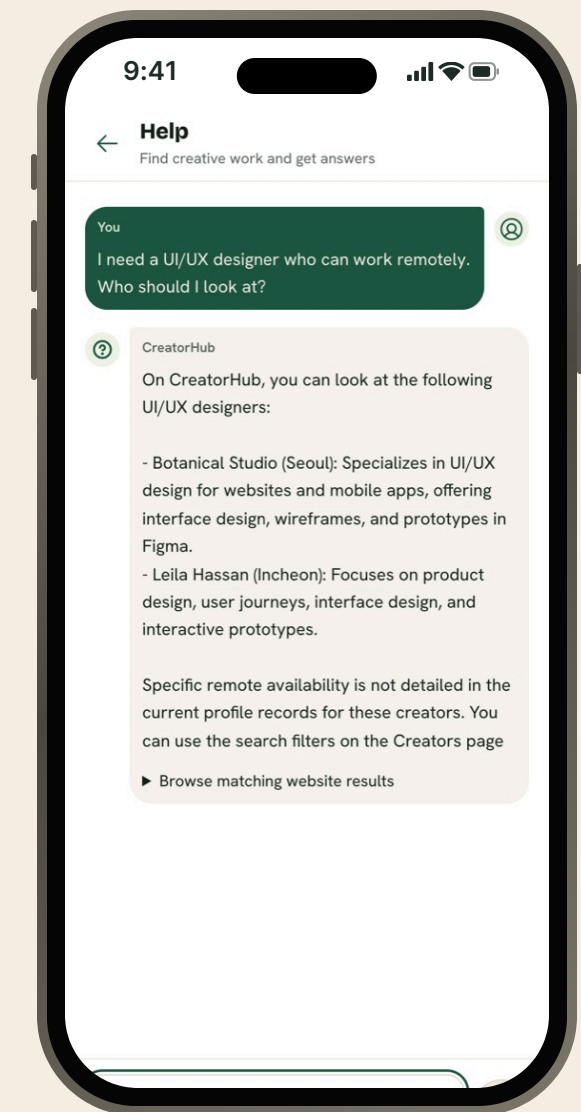
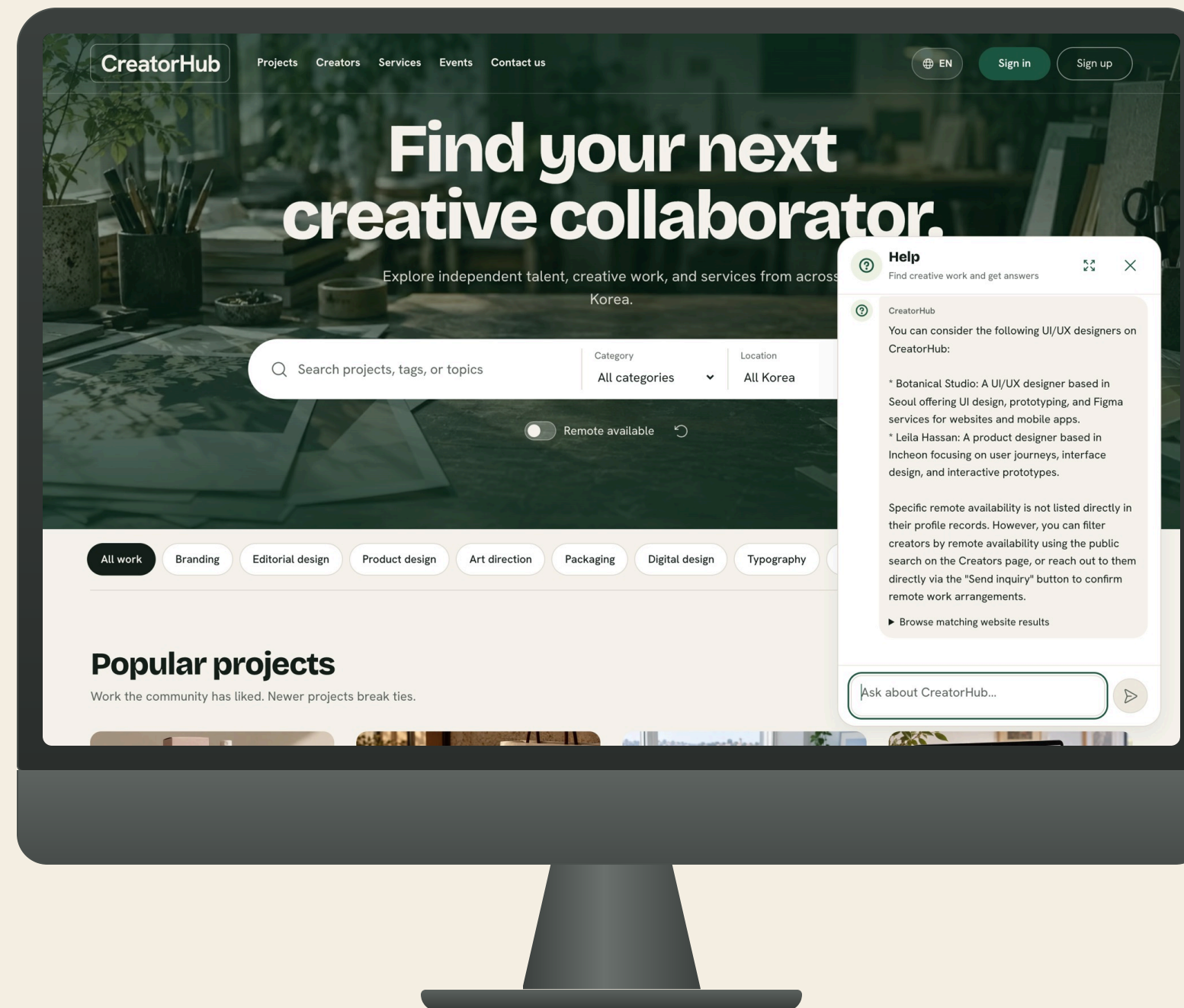
✦ Gemini API

Next.js API route

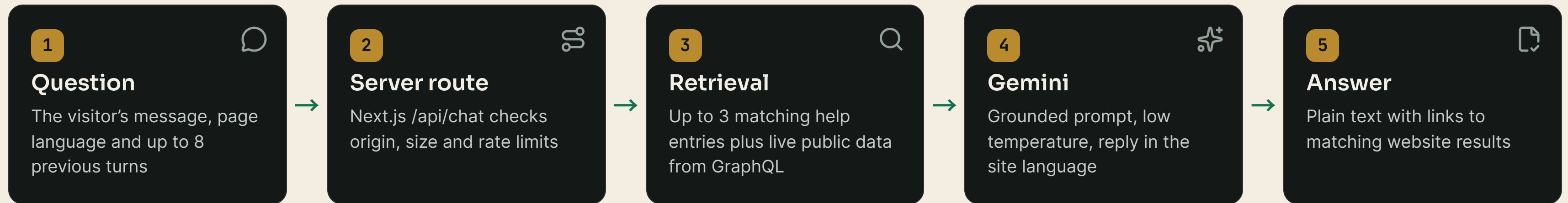
🔗 GraphQL

Open the AI finder ↗

Live answer captured on creatorhub.site. The creators it names are demo profiles on the site.



How the AI finder answers .



Safety

- ◆ The API key stays on the server
- ◆ Only public, approved data is retrieved
- ◆ Sources and chat history are treated as data, not instructions

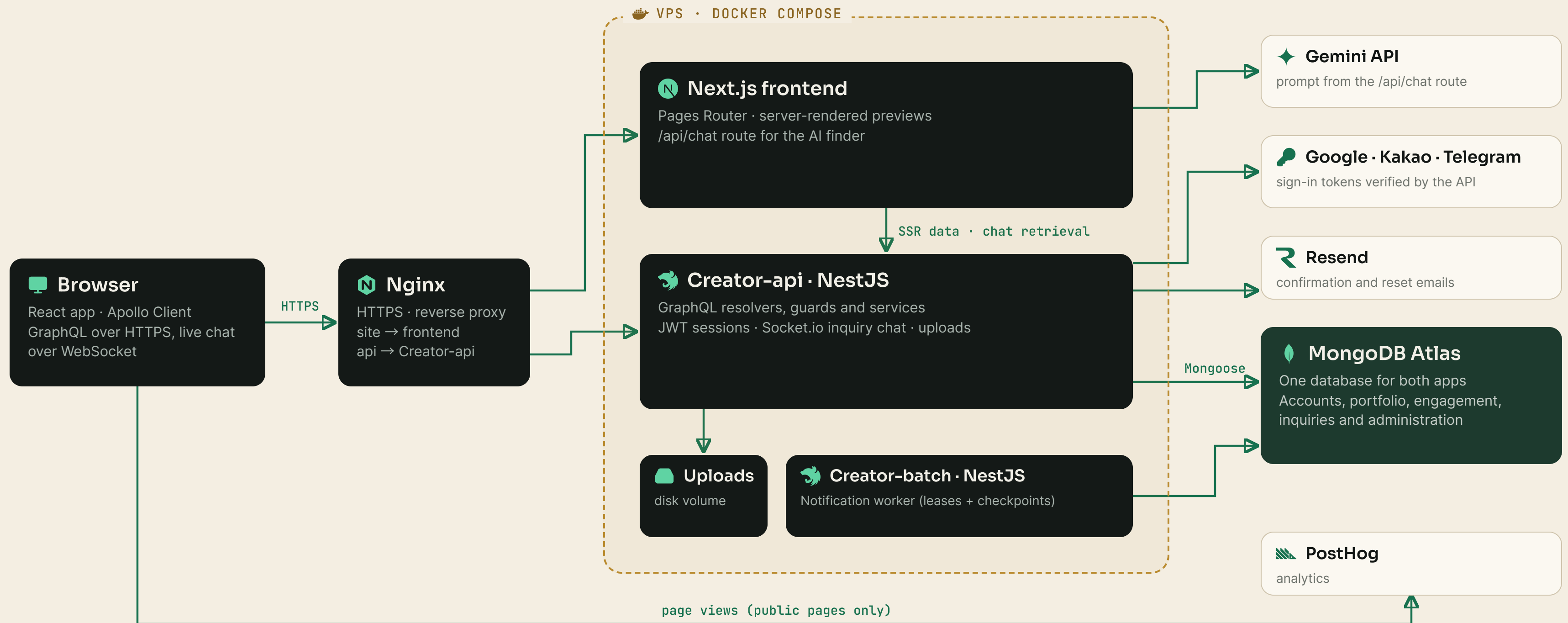
Limits

- ◆ Same-site requests only
- ◆ Messages up to 1,500 characters
- ◆ 8 messages per minute per visitor, plus a daily cap

Resilience

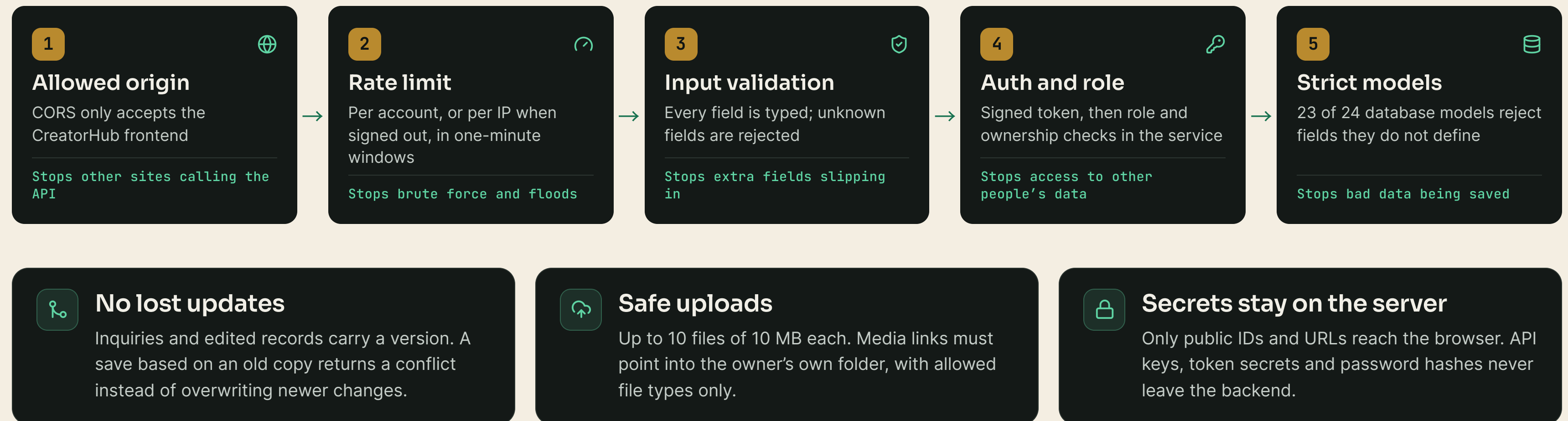
- ◆ Retrieval gives up after 3.5 s
- ◆ A busy model gets one retry on a fallback model
- ◆ Without a key, it points visitors to Help

Three applications, one data model



How every request is protected .

Each GraphQL request passes the same checks, in this order, before it can touch the database.



Sign-in sessions, private inquiries and AI limits are covered on their own pages.

Testing and CI/CD

Both repositories are tested on every push. Releases go through a separate, manually started workflow.

— WHAT IS TESTED

Backend unit · Jest

Business rules: inquiry states, passwords, rate limits

API integration · Jest

Real GraphQL on a throwaway MongoDB replica set

Frontend · Node.js

Sessions, sign-in, the AI chat route, translations

Static checks

TypeScript, ESLint, every query checked against the schema

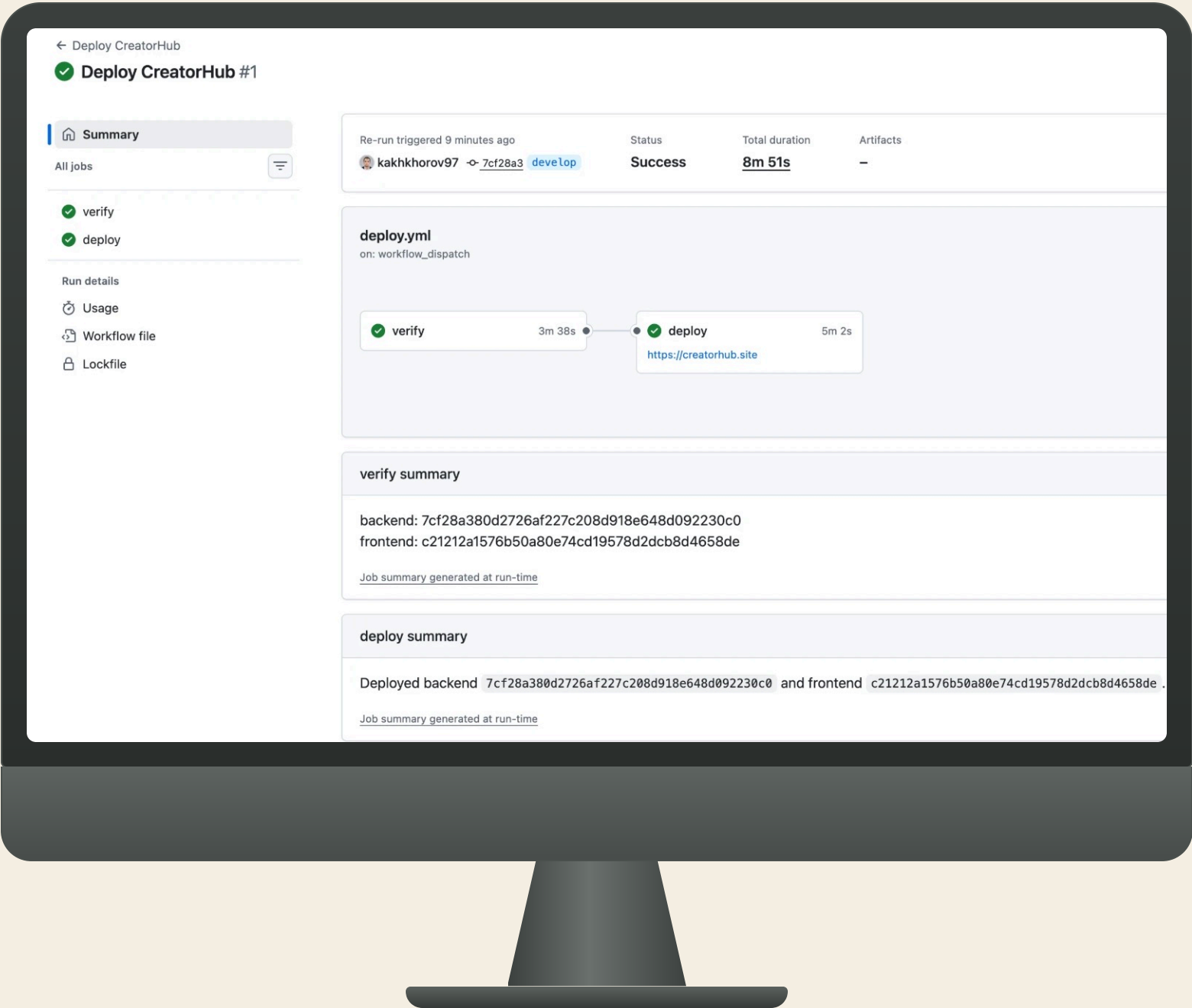
```
local re-run · 17 Oct 2026

$ backend jest ✓ 162/162 tests · 16 suites
$ backend deploy scripts ✓ 23/23
$ frontend node --test ✓ 108/108
$ both typecheck · lint · contract (119 ops) ✓ passed
```

— HOW A RELEASE SHIPS

- 1 **Every push** to develop or main runs install, typecheck, lint, tests and build.
- 2 **Deploy workflow**, started by hand, re-runs all checks for one backend + frontend commit pair.
- 3 **Restricted SSH** to a deploy-only account that accepts nothing but “deploy” and two commit IDs.
- 4 **Health-checked switch**: new images start, wait up to 180 s for health; on failure the previous images return.

Screen: the first Deploy CreatorHub run, 18 Oct 2026. Verify and deploy both passed and released to creatorhub.site.



Running it in production

The frontend, API and worker run as Docker Compose containers on a Linux VPS behind Nginx and HTTPS, with the database on MongoDB Atlas.



Hardened containers

Non-root user, read-only filesystem, every Linux capability dropped and no privilege escalation. Secrets come from env files at runtime, never from the image.



Health checks

Each container is checked every 30 s, three failures mark it unhealthy, and crashed containers restart. A health check proves the app answers, not that every feature works.



Releases and rollback

Each build gets a fixed tag and must pass health checks; the GitHub deploy path returns to the previous images on failure. Database changes are never rolled back.

Checked on 17–18 Oct 2026

- ✓ creatorhub.site serves pages over HTTPS
- ✓ API health endpoint returns ok
- ✓ Container settings read from the Compose file
- ✓ First GitHub deploy run passed (18 Oct 2026)

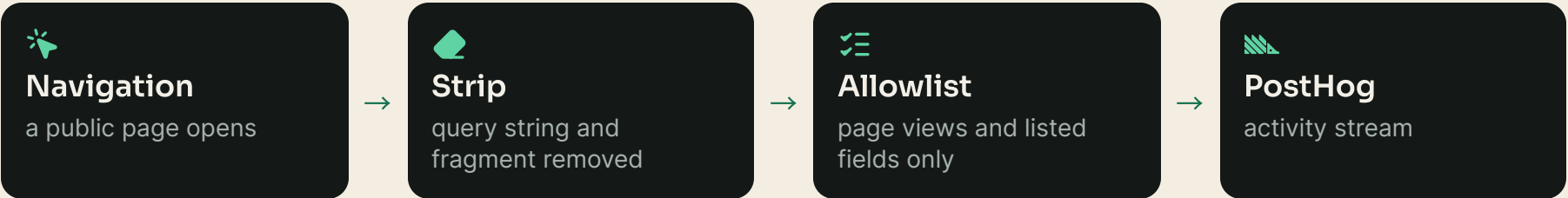
Next steps

- Docker-aware backups with an off-server copy and a test restore
- External uptime alerts

Public-page analytics with PostHog

Page views are recorded on the 15 public pages only. Account, workspace, inquiry and admin pages are never tracked.

- ◆ No IP address and no location lookup
- ◆ No session recordings, heatmaps, surveys or person profiles
- ◆ Analytics errors are caught, so navigation is never interrupted

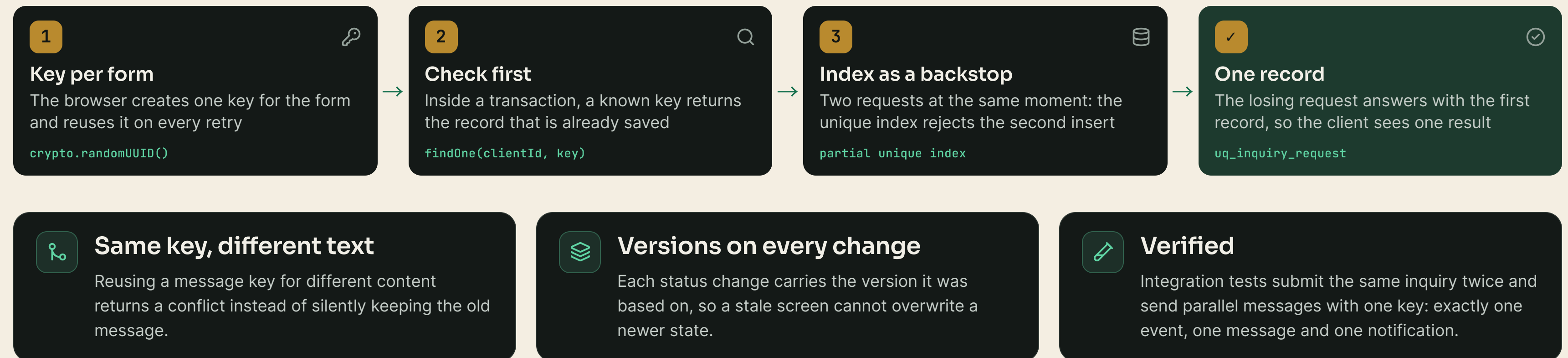


PostHog · Activity · recorded page views on creatorhub.site							
EVENT	PERSON	URL / SCREEN	LIBRARY	TIME			
Pageview	01a144d0-b072-77a1-a99f-7b53868d3100	https://creatorhub.site/	web	17 minutes ago			
Pageview	01a14024-d5e3-72fc-a711-2da1f4583527	https://creatorhub.site/	web	18 minutes ago			
Pageview	01a14455-106d-71dc-99f8-b01e27eaa469	https://creatorhub.site/	web	3 hours ago			
Pageview	01a14392-656b-70bf-ab20-81cb7031588c	https://creatorhub.site/	web	6 hours ago			
Pageview	01a1438d-0dd6-7ecb-a865-97c03a9eb366	https://creatorhub.site/	web	6 hours ago			
Pageview	01a1437a-24e0-73eb-bc57-540ae9c46a2f	https://creatorhub.site/	web	7 hours ago			
Pageview	01a142a3-448e-70bb-b111-1e40cd50e0cc	https://creatorhub.site/	web	10 hours ago			
October 15, 2026							
Pageview	01a143b2-be7f-7851-b968-543a727c341f	https://creatorhub.site/faq	web	16 hours ago			
Pageview	01a143b2-be7f-7851-b968-543a727c341f	https://creatorhub.site/	web	16 hours ago			

Captured 16 Oct 2026. Counts are page views, not unique visitors, and may include my own testing.

One click, one inquiry: retry-safe requests .

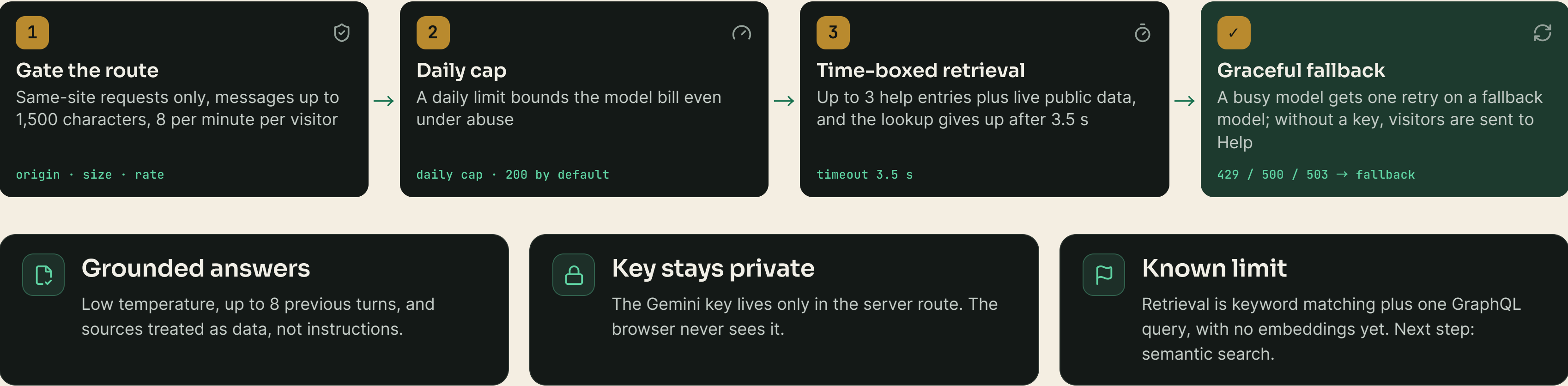
Problem. A double-click or a dropped connection could send the same brief or chat message twice. The server saw two new requests and would save both, with two notifications for the creator.



The same pattern covers likes, saves, follows and comment retries.

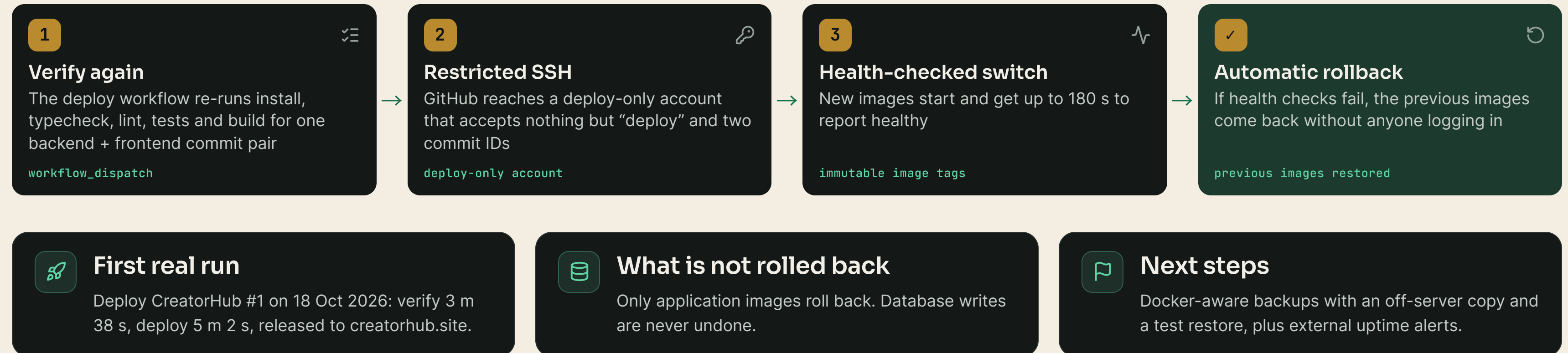
Keeping the AI finder fast, affordable and safe .

Problem. The AI finder is open to every visitor, and each answer is a paid Gemini call. Without limits, one script could flood it, slow lookups would keep visitors waiting, and a busy model would show errors.



Releases that roll themselves back .

Problem. CreatorHub runs on one VPS. A broken release would take the live site down until someone noticed and fixed it by hand.



CreatorHub ER model

All collections expanded · MongoDB · lines connect each parent _id to the referencing field

SOLID LINE = ObjectId REFERENCE

DASHED = EMBEDDED SCHEMA

ACCOUNTS

users		
PK	_id	objectId
	displayName	string
	phoneNumber	string
	avatarUrl	string
	role	enum
	status	enum
	preferredLocale	enum
	lastLoginAt	date
	deletedAt	date
	createdAt	date
	updatedAt	date

authIdentities		
PK	_id	objectId
FK	userId	objectId
	provider	enum
	providerSubject	string
	passwordHash	string
	verifiedAt	date
	revokedAt	date
	lastLoginAt	date
	createdAt	date
	updatedAt	date

authSessions		
PK	_id	objectId
FK	userId	objectId
FK	authIdentityId	objectId
	refreshTokenHash	string
	expiresAt	date
	revokedAt	date
	lastUsedAt	date
	createdAt	date
	updatedAt	date

authTokens		
PK	_id	objectId
FK	authIdentityId	objectId
	purpose	enum
	tokenHash	string
	previousEmail	string
	replacementEmail	string
	expiresAt	date
	revokedAt	date
	consumedAt	date
	createdAt	date

PORTFOLIO

creatorProfiles		
PK	_id	objectId
FK	userId	objectId
	slug	string
	headline	string
	bio	string
	city	string
	remoteAvailable	bool
	availability	enum
FK	categoryIds	objectId[]
	skills	string[]
	socialLinks	SocialLink[]
	status	enum
	publishedAt	date
	deletedAt	date
	createdAt	date
	updatedAt	date

SocialLink		
	label	string
	uri	string

categories		
PK	_id	objectId
FK	slug	string
	name	string
	description	string
	sortOrder	int
	isActive	bool
	createdAt	date
	updatedAt	date

portfolioProjects		
PK	_id	objectId
FK	creatorId	objectId
FK	categoryId	objectId
	title	string
	slug	string
	summary	string
	description	string
	coverImageUrl	string
	media	PortfolioMedia[]
	tags	string[]
	externalUri	string
	status	enum
	publishedAt	date
	deletedAt	date
	createdAt	date
	updatedAt	date

PortfolioMedia		
	type	enum
	uri	string
	altText	string
	sortOrder	int

services		
PK	_id	objectId
FK	creatorId	objectId
FK	categoryId	objectId
	title	string
	description	string
	imageUrl	string
	pricingType	enum
	startingPrice	int
	currency	enum
	estimatedDays	int
	deliverables	string[]
	status	enum
	deletedAt	date
	createdAt	date
	updatedAt	date

ENGAGEMENT

creatorFollows		
PK	_id	objectId
FK	followerId	objectId
FK	creatorId	objectId
	isActive	bool
	followedAt	date
	createdAt	date
	updatedAt	date

projectLikes		
PK	_id	objectId
FK	userId	objectId
FK	projectId	objectId
	createdAt	date

savedProjects		
PK	_id	objectId
FK	userId	objectId
FK	projectId	objectId
	createdAt	date

projectComments		
PK	_id	objectId
FK	userId	objectId
FK	projectId	objectId
	clientCommentKey	string
	content	string
	status	enum
	editedAt	date
	deletedAt	date
	createdAt	date
	updatedAt	date

activityEvents		
PK	_id	objectId
FK	performedById	objectId
FK	creatorId	objectId
FK	projectId	objectId
FK	commentId	objectId
	actionKey	string
	deliveryStatus	enum
	deliveryCursor	string
	leaseUntil	date
	nextAttemptAt	date
	attempts	number
	completedAt	date
	createdAt	date
	updatedAt	date

INQUIRIES

inquiries		
PK	_id	objectId
	clientRequestKey	string
FK	clientId	objectId
FK	creatorId	objectId
FK	servicId	objectId
	subject	string
	brief	string
	budgetMin	int
	budgetMax	int
	currency	enum
	desiredDeadline	date
	serviceSnapshot	InquiryServiceSnapshot
	status	enum
	outcome	enum
FK	closedById	objectId
	closedAt	date
	submittedAt	date
	lastMessageAt	date
	version	int
	createdAt	date
	updatedAt	date

InquiryServiceSnapshot		
	title	string
	pricingType	enum
	startingPrice	int
	currency	enum
	estimatedDays	int
	deliverables	string[]

inquiryMessages		
PK	_id	objectId
FK	inquiryId	objectId
FK	senderId	objectId
	clientMessageKey	string
	body	string
	readAt	date
	createdAt	date

inquiryEvents		
PK	_id	objectId
FK	inquiryId	objectId
FK	performedById	objectId
	type	enum
	fromStatus	enum
	toStatus	enum
FK	messageId	objectId
	version	int
	createdAt	date

serviceFeedback		
PK	_id	objectId
FK	inquiryId	objectId
FK	clientId	objectId
FK	creatorId	objectId
FK	servicId	objectId
	rating	int
	feedback	string
	createdAt	date
	updatedAt	date

ADMINISTRATION

contentReviews		
PK	_id	objectId
	kind	enum
	targetId	objectId
FK	ownerId	objectId
	title	string
	snapshot	mixed
	baseUpdatedAt	date
	status	enum
	reason	string
FK	reviewedById	objectId
	createdAt	date
	updatedAt	date

notifications		
PK	_id	objectId
FK	recipientId	objectId
FK	inquiryEventId	objectId
FK	activityEventId	objectId
	inboxReadAt	date
	readAt	date
	createdAt	date

creativeEvents		
PK	_id	objectId
	title	string
	description	string
	startsAt	date
	endsAt	date
	location	string
	eventUrl	string
	status	enum
	isDemo	bool
	createdAt	date
	updatedAt	date

siteContent		
PK	_id	objectId
	key	enum
	homeHeading	string
	homeSubheading	string
	aboutText	string
	contactEmail	string
	contactText	string
	footerCopy	string
	createdAt	date
	updatedAt	date

Inquiries, field by field

PK primary key FK → referenced collection  embedded schema

inquiries		
PK	_id	objectId
	clientRequestKey	string
FK	clientId	→ users
FK	creatorId	→ creatorProfiles
FK	serviceId	→ services
	subject	string
	brief	string
	budgetMin	int
	budgetMax	int
	currency	enum
	desiredDeadline	date
	serviceSnapshot	embedded
	status	enum
	outcome	enum
FK	closedById	→ users
	closedAt	date
	submittedAt	date
	lastMessageAt	date
	version	int
	createdAt	date
	updatedAt	date

InquiryServiceSnapshot		
embedded in inquiries.serviceSnapshot		
	title	string
	pricingType	enum
	startingPrice	int
	currency	enum
	estimatedDays	int
	deliverables	string[]

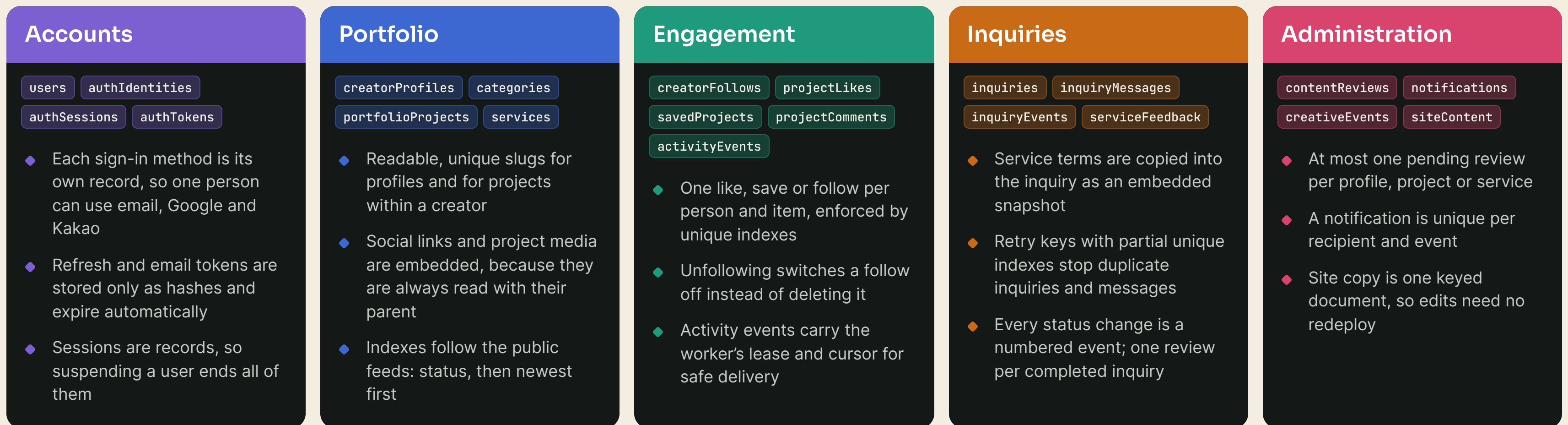
inquiryMessages		
PK	_id	objectId
FK	inquiryId	→ inquiries
FK	senderId	→ users
	clientMessageKey	string
	body	string
	readAt	date
	createdAt	date

inquiryEvents		
PK	_id	objectId
FK	inquiryId	→ inquiries
FK	performedById	→ users
	type	enum
	fromStatus	enum
	toStatus	enum
FK	messageId	→ inquiryMessages
	version	int
	createdAt	date

serviceFeedback		
PK	_id	objectId
FK	inquiryId	→ inquiries
FK	clientId	→ users
FK	creatorId	→ creatorProfiles
FK	serviceId	→ services
	rating	int
	feedback	string
	createdAt	date
	updatedAt	date

How the data is designed .

21 collections in one MongoDB database, shared by the API and the worker. References are checked in the services, and unique indexes enforce the rules that must never break.



Every field of every collection is on the previous page.

— EARLIER PROJECT

Sofiya Bookstore

An online bookstore where readers browse books by genre and format, fill a basket and follow their orders. A separate admin panel manages the catalogue.

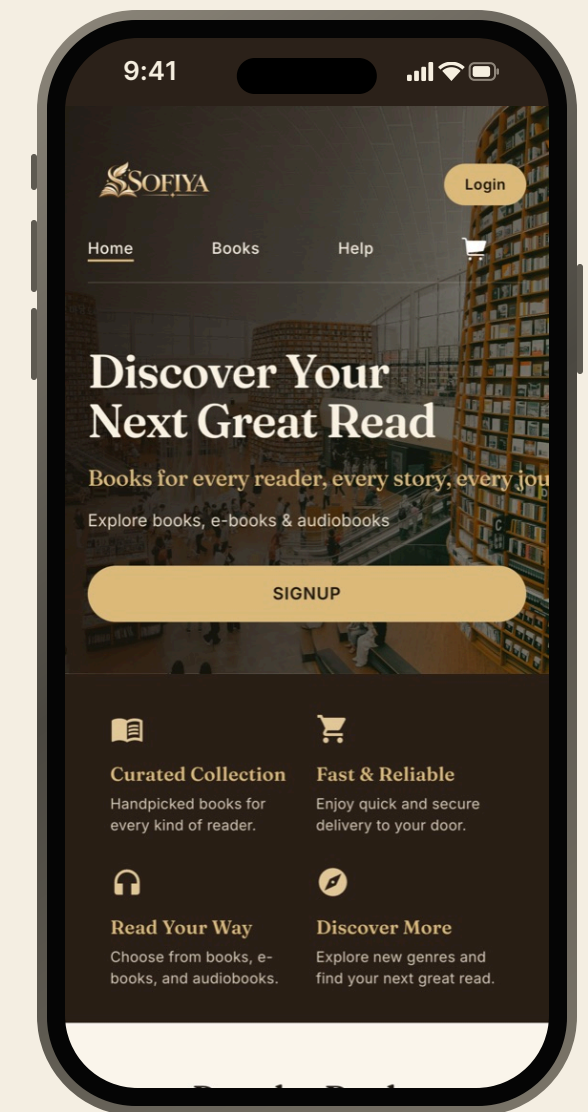
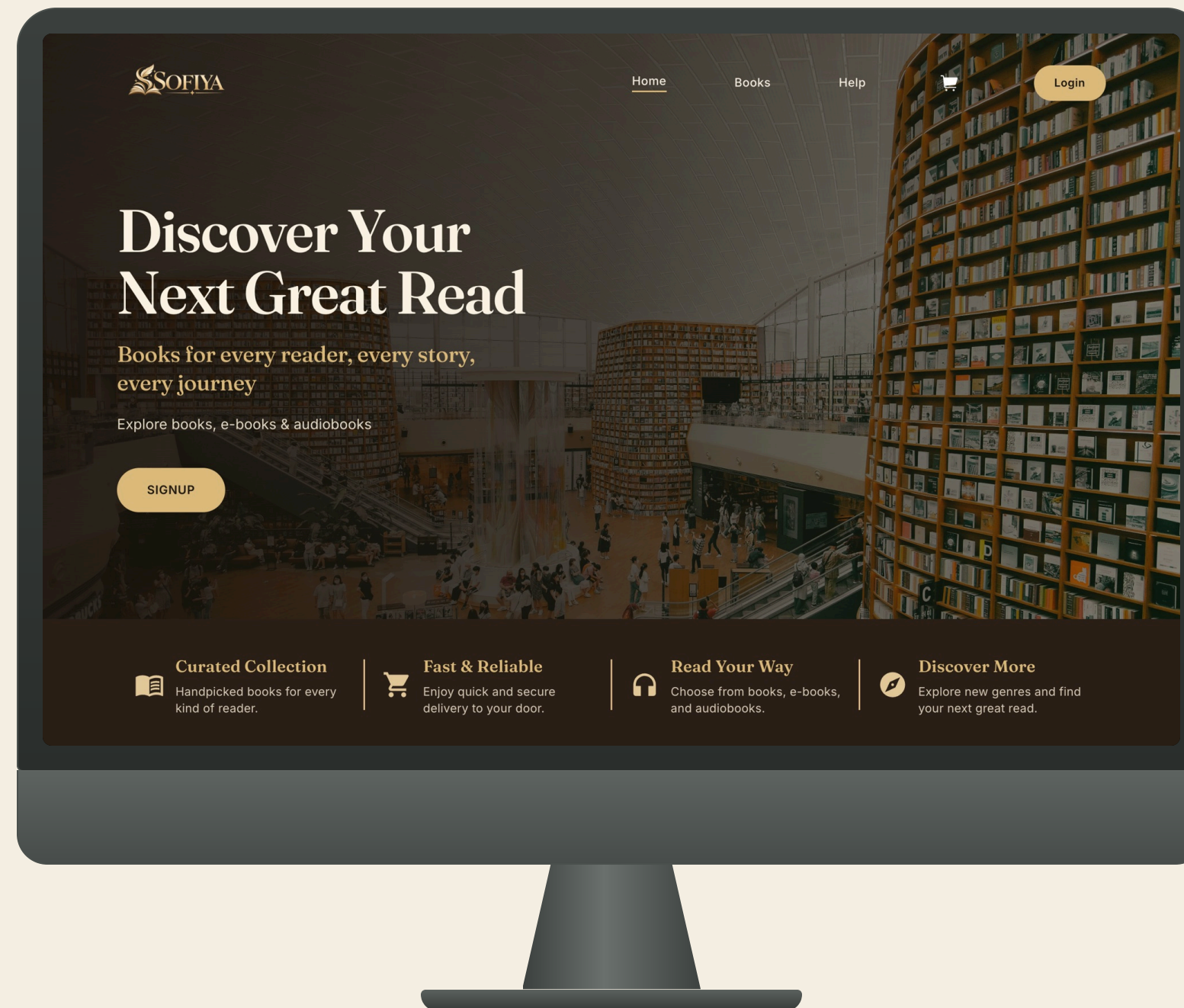
— HOW IT'S BUILT

- ◆ React and Redux Toolkit storefront talking to an Express REST API
- ◆ Two sign-in paths: JWT cookie for readers, MongoDB-stored sessions for the admin panel
- ◆ Admin panel rendered on the server with EJS; passwords hashed with bcrypt

React Redux Toolkit MUI Express

MongoDB

Repos: [Sofiya-react](#) · [Sofiya-Bookstore](#)



Browse, search and order

Nine genres and four formats with search and pagination, a product page with a slider and basket, and the same screens adapted for phones.

— HOW IT'S BUILT

- ◆ Orders are saved with their line items and move through paused, in-process and delivered
- ◆ A product view is counted once per member
- ◆ Cover and profile image uploads handled with Multer

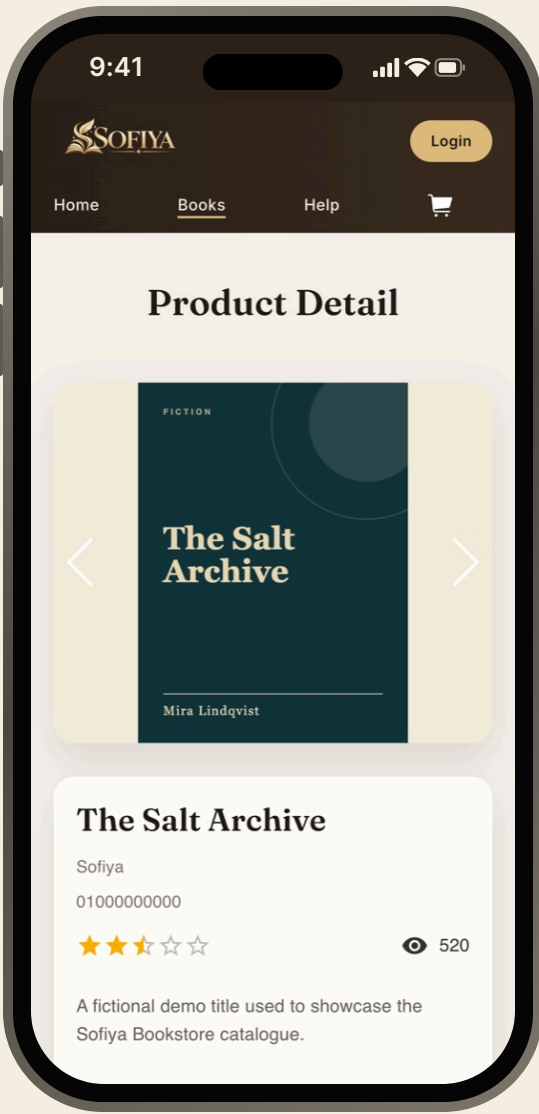
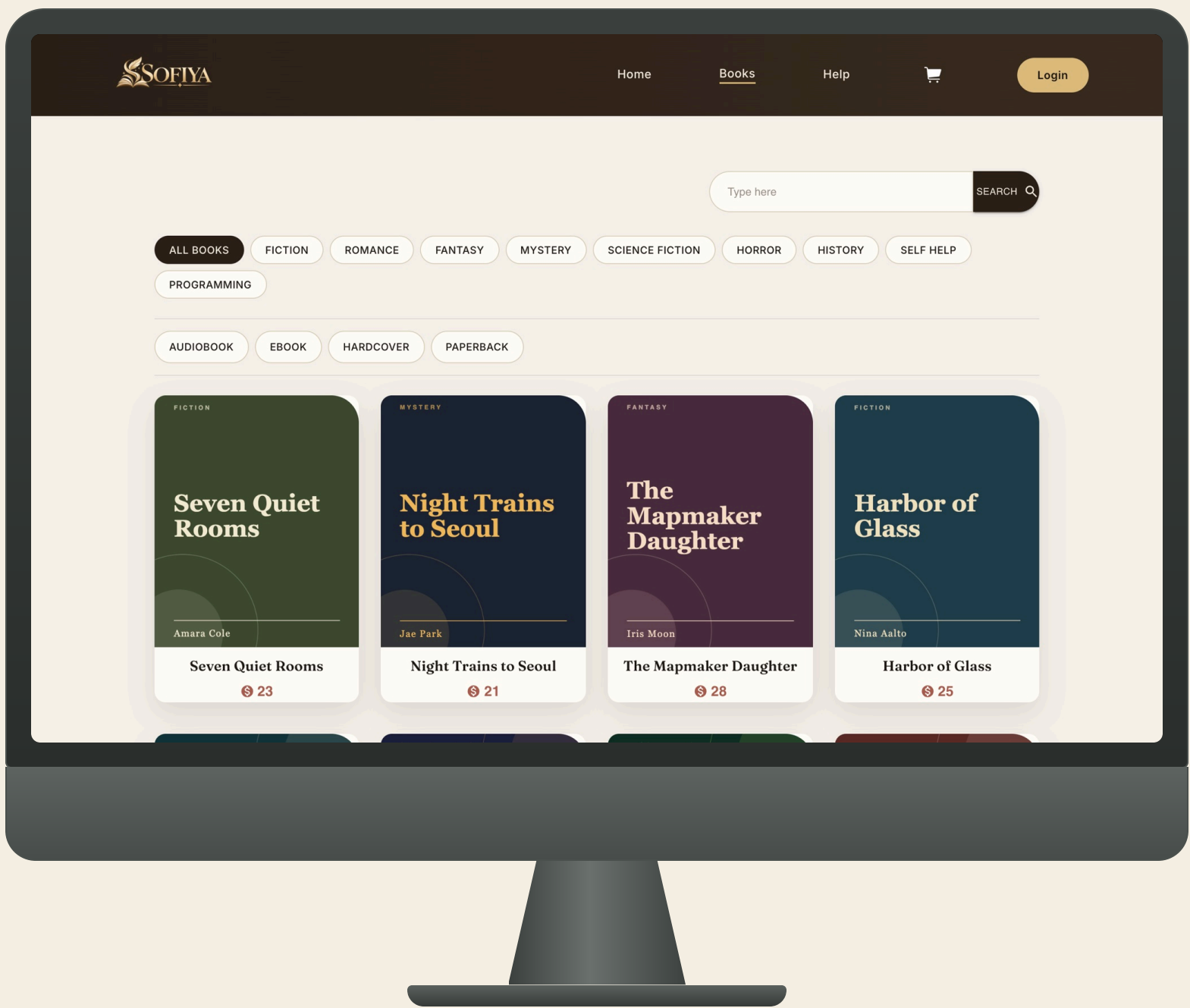
Mongoose

Axios

Multer

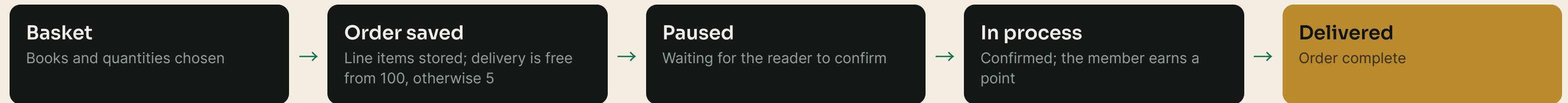
React Router

Book titles, authors and covers in these screenshots are fictional demo data from a local run.



Admin panel, orders and sign-in •

Every order query is filtered by the signed-in member, so readers only ever see and change their own orders.



Admin panel

Server-rendered EJS pages: sign-in, the product list, new products with up to 5 images, product edits, and a user list where the admin can change a member's status.



Two sign-in paths

Readers get a JWT in a cookie for 24 hours, checked on every API call. The admin panel uses sessions stored in MongoDB for 6 hours, and only store accounts pass its guard.



What I would change now

Item prices come from the browser and the payment step is a mock-up. Today I would recompute totals on the server and connect a real payment provider.

— CONTACT

From idea to production

✉ EMAIL

kakhkhorov97@gmail.com

☎ PHONE

[010-7901-7555](tel:010-7901-7555)

💬 KAKAOTALK

[kakhkhorov_A](https://kakaotalk.com/kakhkhorov_A)

🐙 GITHUB

github.com/kakhkhorov97

🌐 LIVE PROJECT

creatorhub.site

📍 ADDRESS

Suwon, Gyeonggi-do, South Korea



Live project
creatorhub.site



GitHub
github.com/kakhkhorov97