



PRODUCTION-GRADE PLATFORM

Bharadwaj.ly

A Cloud-Native, Full-Stack URL Shortening Platform built with **React**, **Spring Boot**, **JWT Security**, and **PostgreSQL**. Stateless. Secure. Fully deployed.



Not a Demo. A Production System.

Bharadwaj.ly is a complete, working URL shortening system engineered with real-world architecture and deployed across cloud infrastructure — not a toy project, but a fully operational platform.

Stateless JWT Auth

Secure, scalable token-based authentication

REST API Design

Clean, protected, and well-structured endpoints

Cloud-Native Deploy

Frontend, backend, and DB on separate cloud services

Environment-Driven

Zero hardcoded secrets, fully configurable

LIVE DEPLOYMENT

Where It Lives

Frontend – Vercel Edge Network

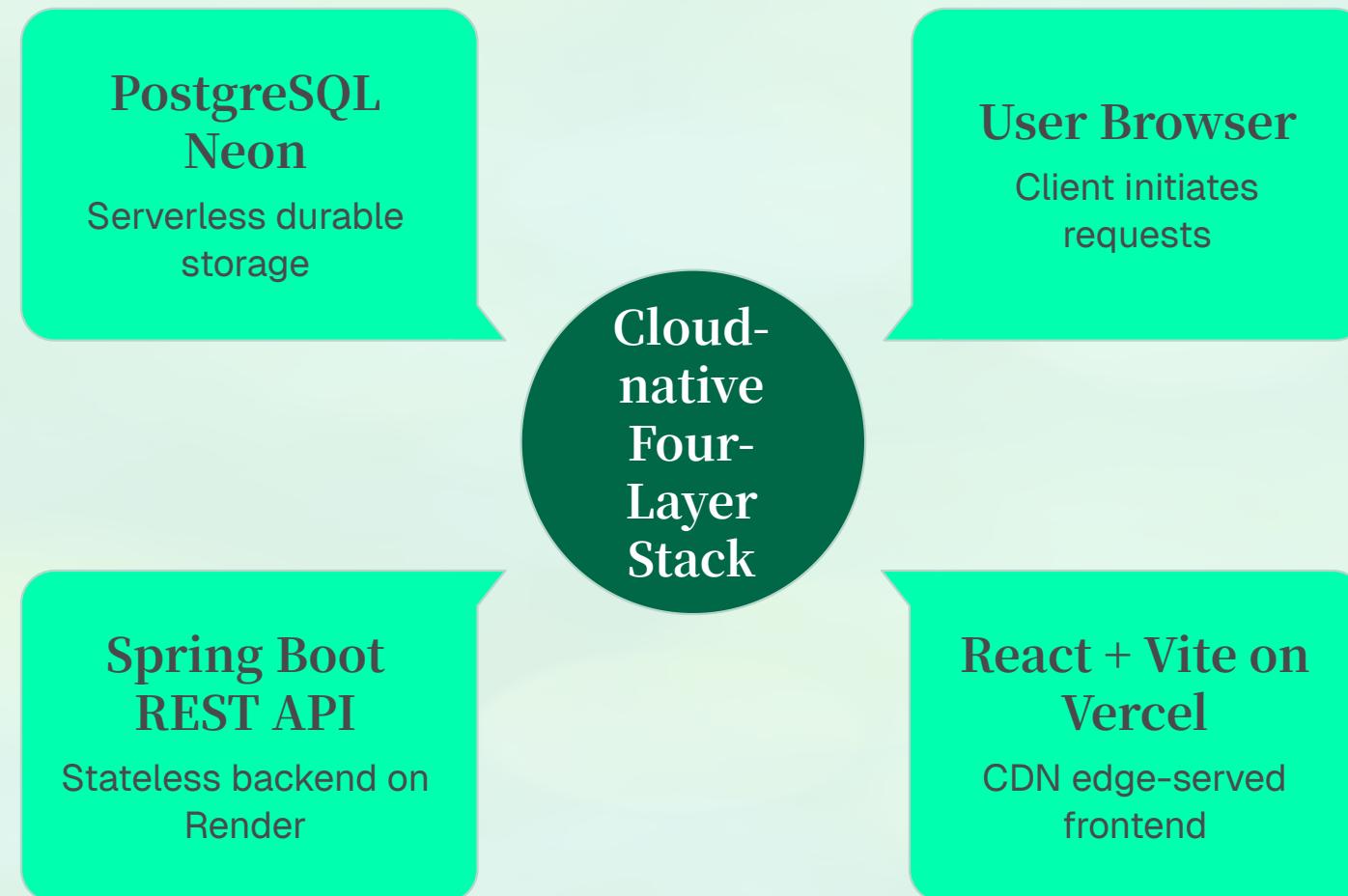
Backend API – Render Cloud

Database – Neon PostgreSQL

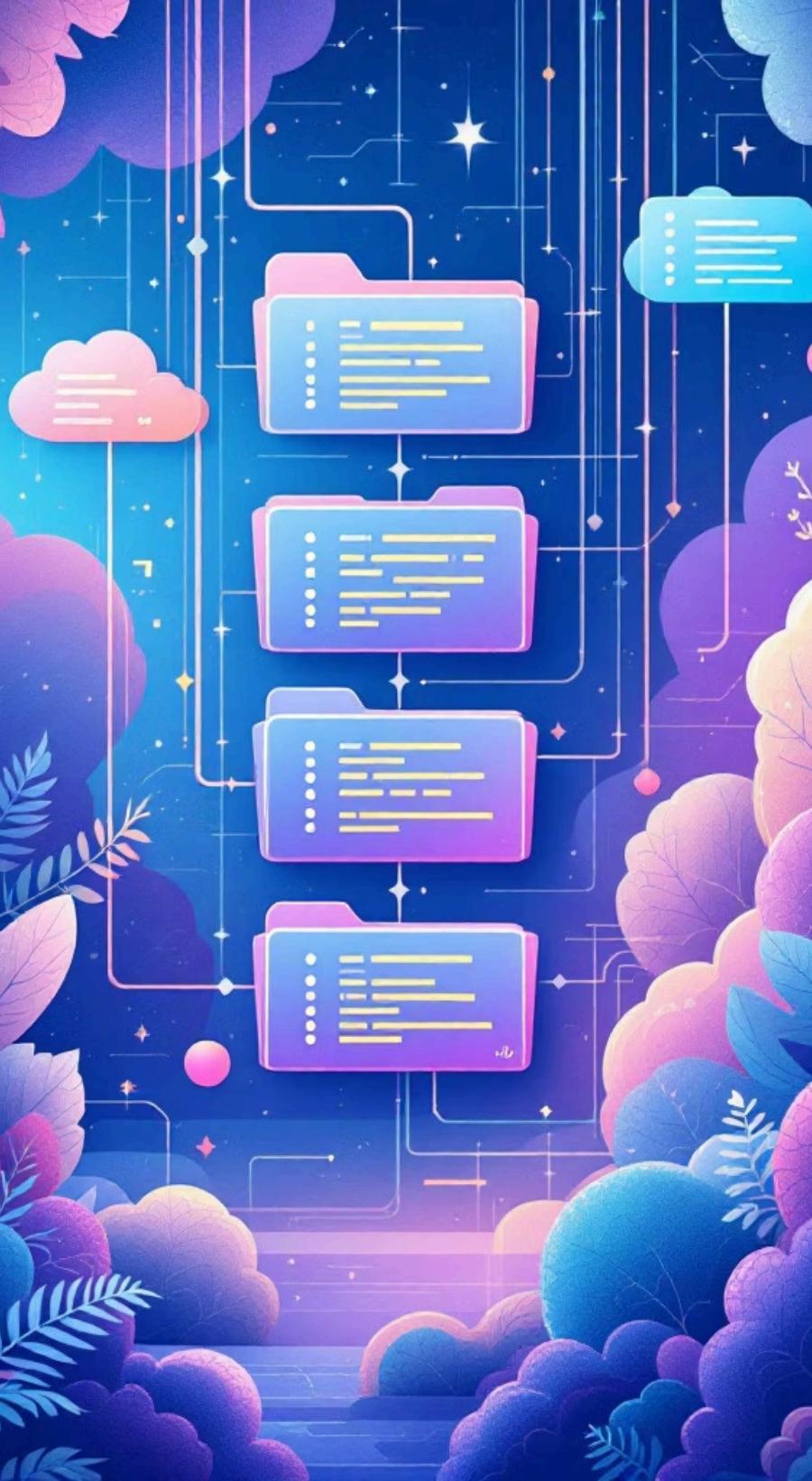
Serverless, managed, always-on PostgreSQL



System Architecture Overview



A clean, vertically separated stack — each layer independently deployed and horizontally scalable. Stateless by design, secure by default, cloud-native throughout.



Monorepo Project Structure

Backend /backend

- `controller/` — REST endpoints (Auth, URL APIs)
- `service/` — Business logic layer
- `repository/` — JPA repositories
- `model/` — Entity classes (User, Url)
- `security/` — JWT filter, CORS, security setup
- `Dockerfile` — Container-ready deployment

Frontend /frontend

- `api/` — Centralized Axios configuration
- `components/` — Reusable UI components
- `contextApi/` — Global auth state
- `pages/` — Login, Register, Dashboard
- `routes/` — Protected routing logic
- `vite.config.js` — Optimized build config

❑ Structured. Modular. Maintainable — every directory has a single, clear responsibility.

Spring Boot Backend Stack

Core Dependencies

- spring-boot-starter-web
- spring-boot-starter-data-jpa
- spring-boot-starter-security
- spring-boot-starter-validation
- JWT (io.jsonwebtoken)
- PostgreSQL Driver + Hibernate
- Lombok

Architecture Highlights

- Stateless JWT authentication with custom filter
- Secure CORS configuration per origin
- Protected API routes with Spring Security
- Layered: Controller → Service → Repository
- Docker-ready with environment variable injection

React + Vite Frontend Stack



React & Routing

React with `react-router-dom` for protected client-side routing and Context API for global auth state management.



API & Forms

Centralized Axios instance with `react-hook-form` for validation and `react-hot-toast` for user feedback.



UI & Styling

Material UI components paired with Tailwind CSS for a responsive, production-grade interface with minimal overhead.

Build & Deploy

Vite for fast development and optimized production builds, deployed instantly to Vercel's global edge network.

Security Model

Security is a first-class citizen — not an afterthought. Every layer of the stack enforces strict access control.



JWT Stateless Auth

Bearer token validation on every protected request



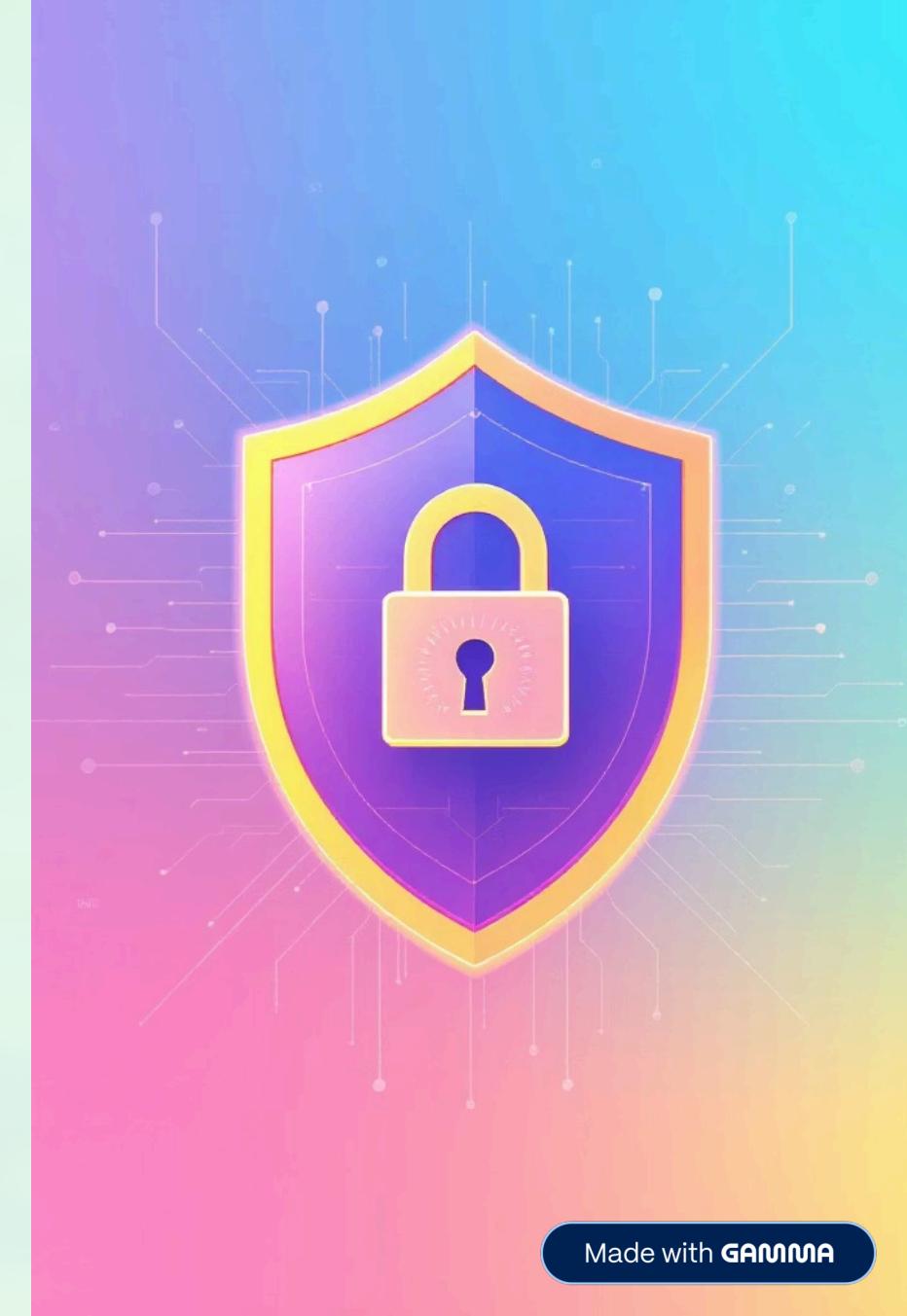
CORS Origin Control

Only trusted origins permitted to communicate with the API



Zero Hardcoded Secrets

All credentials injected at runtime via environment variables



Running It Locally

Clone & Setup

```
git clone https://github.com/manu577228/  
url-shortener-fullstack.git  
cd url-shortener-fullstack
```

Run Backend

```
cd backend  
.mvnw spring-boot:run
```

Run Frontend

```
cd frontend  
npm install  
npm run dev
```

Environment Variables

Backend

```
DATABASE_URL=  
DATABASE_USERNAME=  
DATABASE_PASSWORD=  
JWT_SECRET=  
FRONTEND_URL=
```

Frontend

```
VITE_BACKEND_URL=  
VITE.REACT_SUBDOMAIN=
```

Built by Manu Bharadwaj

Designed with intent. Engineered for scale. Deployed with confidence.

01

Clean Architecture

Separation of concerns across every layer

02

Stateless & Secure

JWT auth, CORS control, environment-driven secrets

03

Cloud Native

Vercel + Render + Neon — independently scalable

04

Production Ready

Not a demo — a live, fully operational system

LINKEDIN

MANU BHARADWAJ

YOUTUBE

CODE WITH BHARADWAJ

 [Live Demo: url-shortener-frontend-by-manu.vercel.app](https://url-shortener-frontend-by-manu.vercel.app) |  Star

the repo if you find it useful.



Made with 