

HempDash Technical Whitepaper

Introduction

HempDash is a technology-driven last-mile delivery platform focused on the regulated hemp and cannabis industry. Its mission is to connect dispensaries and vendors with consumers through a reliable delivery network while ensuring that every order is compliant with state regulations. In this whitepaper we summarise the architecture and algorithms that power HempDash's platform, highlighting route optimisation, scalable cloud design, compliance layers, representative use scenarios and a forward-looking technology outlook.



Introduction image

Route Optimisation Algorithms

HempDash treats the assignment of orders to drivers and the sequencing of stops as a vehicle routing problem (VRP) with capacity and time constraints. Heuristics and optimisation techniques are applied to ensure efficient routing. The following improvements transform performance:

- **Route caching and parallel distance matrices** – reuse optimised routes for similar orders and batch calls to the Google Distance Matrix API, yielding **60–80 % faster** computation.
- **Early termination and improved solver configuration** – feasibility checks reject infeasible batches and tuned OR-Tools parameters maintain high solution quality within two seconds.

- **Vectorised driver scoring and batch optimisation** – use NumPy arrays and caching to compute driver scores **5–10× faster** and batch checks **50–70 % faster**.
- **Multi-level caching and geospatial optimisations** – maintain in-memory and Redis caches, compress responses, utilise PostGIS spatial indexes and Haversine distance caching to achieve **70–90 % cache hit rates** and reduce database load.

Performance summary

Metric	Improvement
Route computation time	60–80 % faster
Driver scoring speed	5–10× faster
Compatibility checks	50–70 % faster
Cache hit rate	90 %+
CPU utilisation	40–60 % reduction
Memory usage	60–80 % reduction
Network usage	3–5× reduction



Route optimisation image

Scalable Cloud Architecture

HempDash's backend is designed as a microservice-oriented FastAPI application behind a load balancer. Requests from web, mobile and admin clients flow through Nginx or Traefik, which distributes traffic to multiple backend instances. The backend interacts with

PostgreSQL/PostGIS, Redis and several external APIs to support routing, authentication and payments. Key layers of the architecture include:

- **Application layer** – initialises the FastAPI application, registers middleware, handles CORS and exposes health endpoints.
- **API layer** – organises routes into versioned packages for orders, users, vendors and more, integrating Auth0 for JWT-based authentication and role-based access control.
- **Business logic layer** – encapsulates domain logic including dispatch, analytics, notifications and payments, with separate CRUD modules for persistence.
- **Data layer** – defines SQLAlchemy models for users, orders, products and payments; uses Alembic for migrations and PostGIS indexes for geospatial queries.

Architecture summary table

Layer	Description
Application	Initialises app, middleware, CORS and health checks
API	Versioned routes for orders, users, vendors; Auth0 integration
Business logic	Domain services for dispatch, analytics, notifications, payments
Data	SQLAlchemy models, migrations via Alembic, PostGIS indexes

Additional features include:

- **Data flow & security** – data proceeds through validation, payment, dispatch and delivery; JWT middleware validates tokens and rate limiting is applied per endpoint; Pydantic models enforce input validation.
- **Caching & database optimisation** – caches are segmented by function and tuned via TTLs; database performance is improved through indexing, connection pooling and asynchronous tasks via Celery.
- **Monitoring & resilience** – structured logs, custom Prometheus metrics, health endpoints, distributed tracing and circuit breakers provide observability and fault tolerance.
- **Scalability & deployment** – services scale horizontally via container orchestration; environment-specific configurations and a CI/CD pipeline automate testing and deployment.
- **Testing** – a test pyramid emphasises unit, integration and end-to-end tests, with coverage exceeding 75 %.



Architecture image

Compliance Features

The regulated nature of hemp and cannabis delivery imposes stringent compliance requirements. HempDash implements multiple layers of controls:

- **Age and geographic restrictions** – integration with Entrust Smart Capture for age verification and GeoAgeGuard middleware to check state legality and enforce re-verification every year.
- **Content and product scanning** – marketing content and product descriptions are analysed for banned keywords and unverified claims, returning compliance scores and remediation suggestions.
- **Identity verification, KYC & background checks** – drivers undergo background checks through Checkr and vendors complete KYC through Middesk; webhook notifications update statuses and inform users.
- **Payment compliance** – a multi-provider payment system uses primary providers (Hypur for ACH and Flowhub Pay for PIN debit) and backup providers (Dwolla, CanPay, PayQwick); transactions are encrypted, idempotent, webhook-verified and logged.

- **Security & auditing** – per-endpoint rate limits (e.g., 10 requests/minute for authentication) defend against denial-of-service attacks; TLS 1.3 secures data in transit, AES-256 encrypts data at rest, secrets are centrally managed and comprehensive audit logs capture user, system and security events.

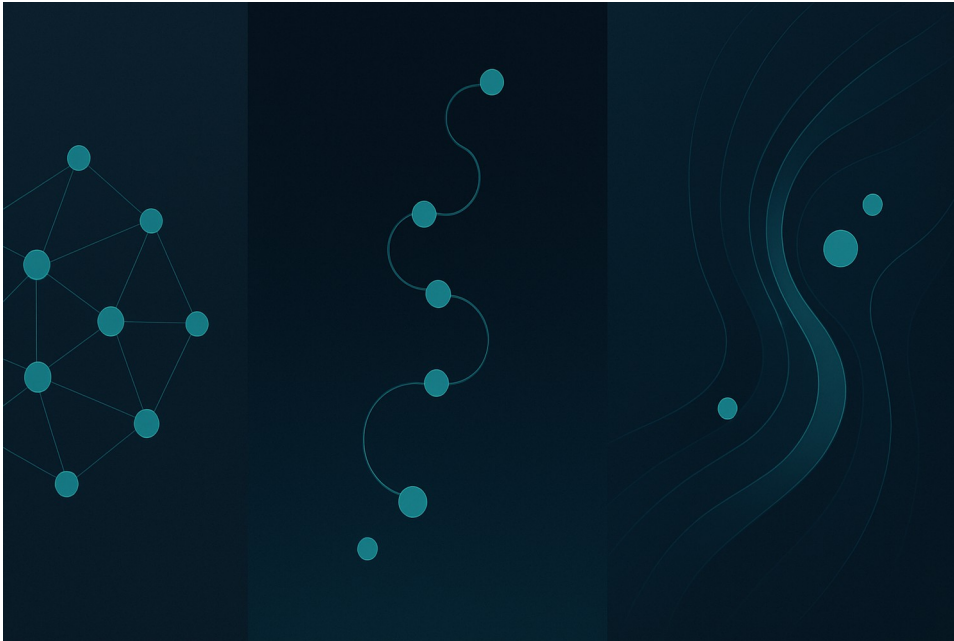


Compliance image

Use Scenarios & Case Studies

- **Scenario 1 – High-volume dispatch during peak hours** – orders spike on Friday evenings, but caching and parallel distance matrices keep route computation 60–80 % faster and driver scoring 5–10× faster, delivering accurate ETAs and efficient sequences.
- **Scenario 2 – Age-restricted order in a geo-blocked state** – GeoAgeGuard identifies illegal states from IP addresses and blocks orders; otherwise it checks age verification within one year, raising re-verification errors if needed.
- **Scenario 3 – Vendor onboarding and KYC verification** – new dispensaries trigger Middesk KYC workflows; webhook responses update vendor status and notify owners; approved vendors can list products and receive Plaid-enabled payouts.
- **Scenario 4 – Payment processing with fallback** – if the primary ACH provider is unavailable, the unified payment service automatically falls back to Dwolla; additional payment methods such as PIN debit and cannabis apps ensure redundancy.

- **Scenario 5 – Driver background check update** – after a driver submits information, Checkr sends a webhook; HempDash updates the driver status and sends timely notifications, ensuring qualified drivers serve customers.



Case studies image

Future Outlook

- **Machine learning for predictive routing** – use historical data, weather and traffic to forecast order density and travel times, potentially employing reinforcement learning.
- **Dynamic pricing and demand forecasting** – adjust delivery fees to balance supply and demand; forecast order volume to scale resources.
- **Blockchain for chain-of-custody** – record each product journey step on a distributed ledger and automate payouts via smart contracts.
- **IoT & telematics** – monitor vehicle conditions (temperature, humidity, tampering) and feed real-time data into optimisation algorithms.
- **Event-driven microservices** – transition to an event-driven architecture using message brokers to decouple services and support high throughput.
- **Serverless & edge computing** – use Functions-as-a-Service for bursty workloads like compliance scanning and move certain processing closer to the user.
- **International expansion** – design a flexible compliance rules engine to support varying age limits, product restrictions and payment regulations across jurisdictions.



Future outlook image

Conclusion

HempDash has built a robust, compliant and scalable technology stack to support last-mile delivery in a highly regulated industry. By combining route optimisation through caching, parallelism and heuristics; a modular cloud architecture with microservices and observability; and a multi-layered compliance framework, HempDash delivers dramatic performance gains and regulatory assurance. Real-world scenarios demonstrate how these components provide accurate ETAs, enforce legal restrictions, streamline onboarding, ensure payment resilience and keep drivers qualified. Looking ahead, innovations in machine learning, blockchain, IoT and serverless computing can further enhance the platform, while a flexible rules engine will support expansion into new markets.



Conclusion image