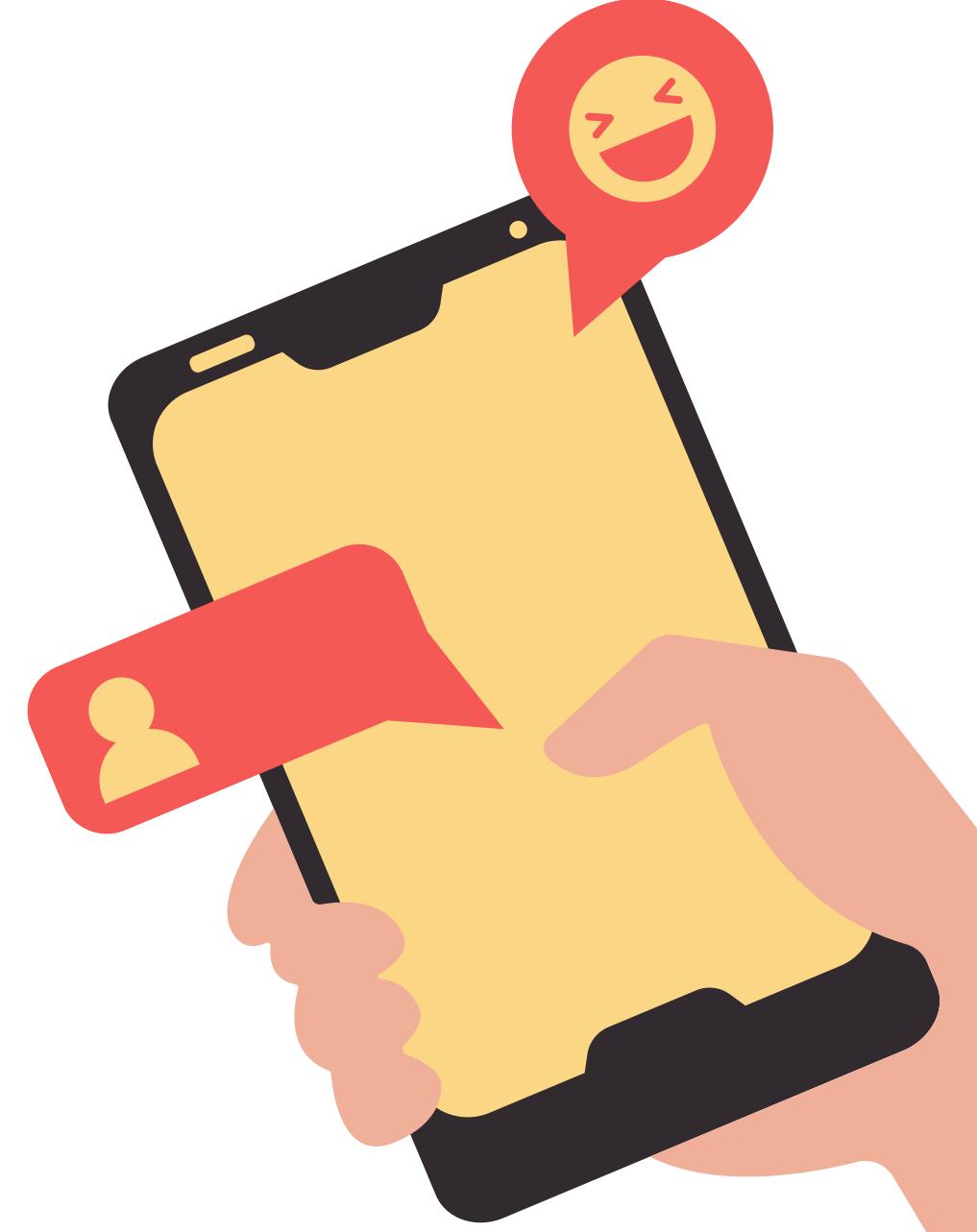




# Campaign Analytics Dashboard



Visualizing KPIs & Forecasting ROI with AI

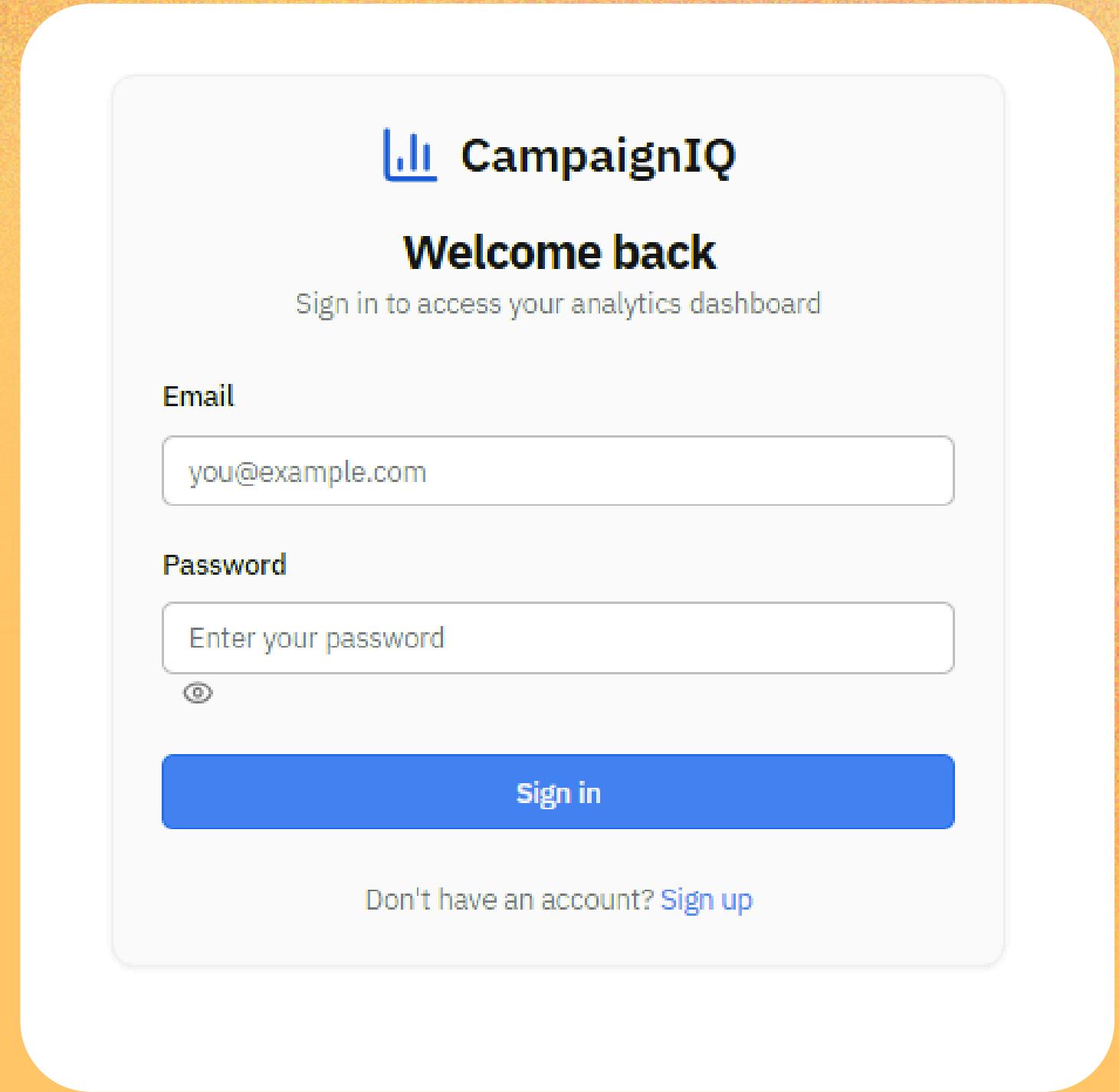
---

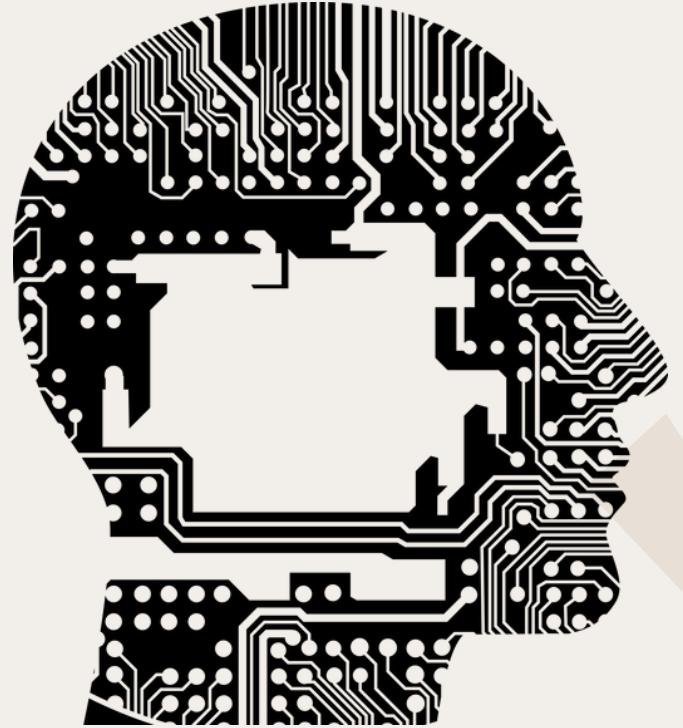
Presented by :

*Aryan Sharma ,Emil Tom Thomas, Prabhat Sharma, Vishwa Singh Kushwah*

# Why This Dashboard?

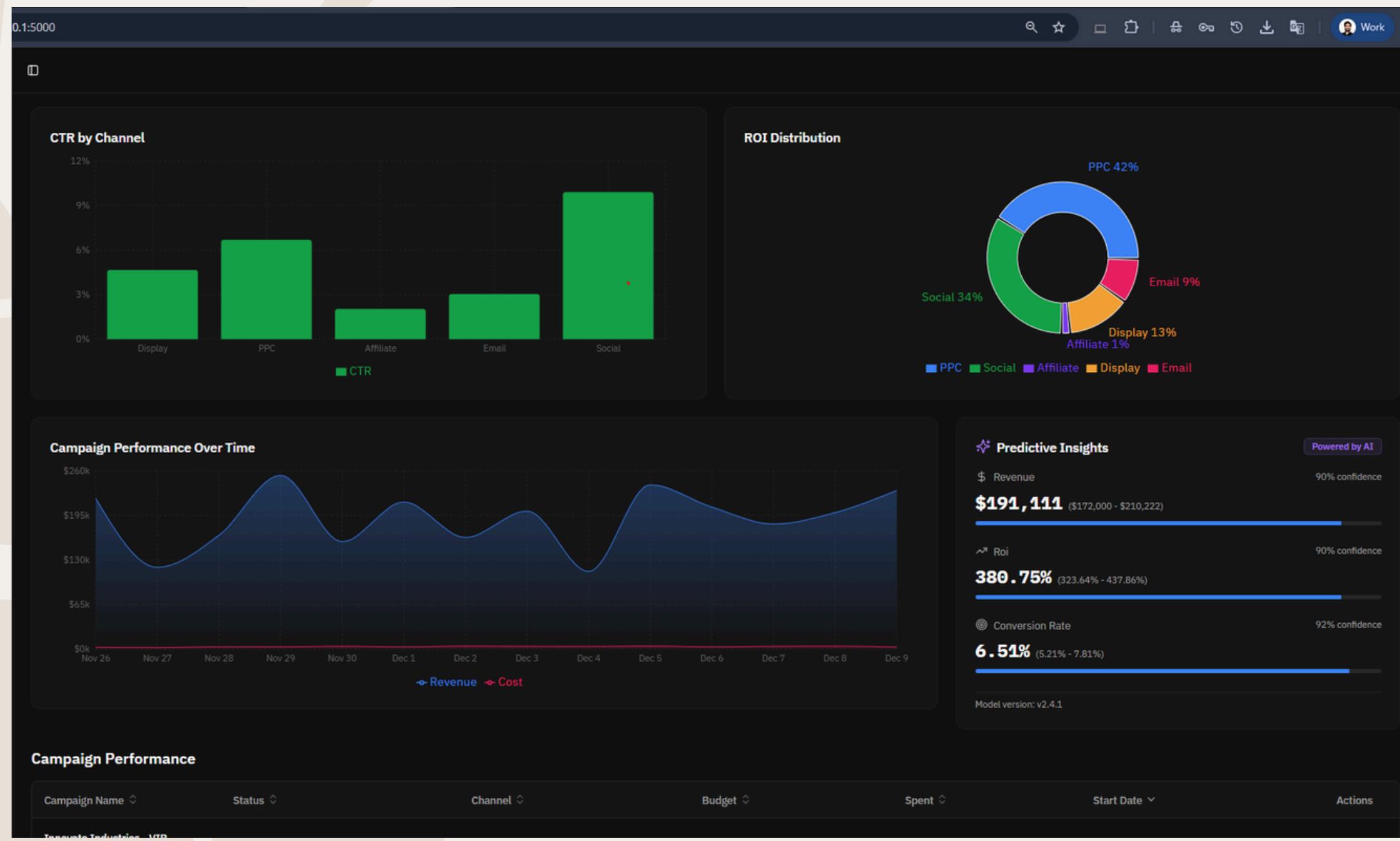
- **Problem:** Marketing teams need a single source of truth to track campaign performance and predict future success.
- **Solution:** A comprehensive dashboard for:
  - **KPI Visualization:** Real-time tracking of Conversion Rate, CTR, ROI, and Impressions.
  - **Forecasting:** Integration of a predictive model for AI-powered conversion and ROI forecasts.
  - **Operational Efficiency:** Full CRUD for campaign management.
- **Key Components:**
  - Interactive Charts (Recharts)
  - KPI Cards & Forecast Card
  - Paginated, sortable Campaign Table





# Technology & Architecture

- **Frontend** :- Interactive Dashboard (React, Recharts, Tailwind CSS) for visualization and UX.
- **Backend** :- RESTful API (Express) handling business logic, authentication, and data retrieval.
- **Data Layer** :- PostgreSQL for persistence, managed by Drizzle ORM for type safety.
- **Integration** :- External Predictive Model API resilience ensured via Circuit Breaker pattern.



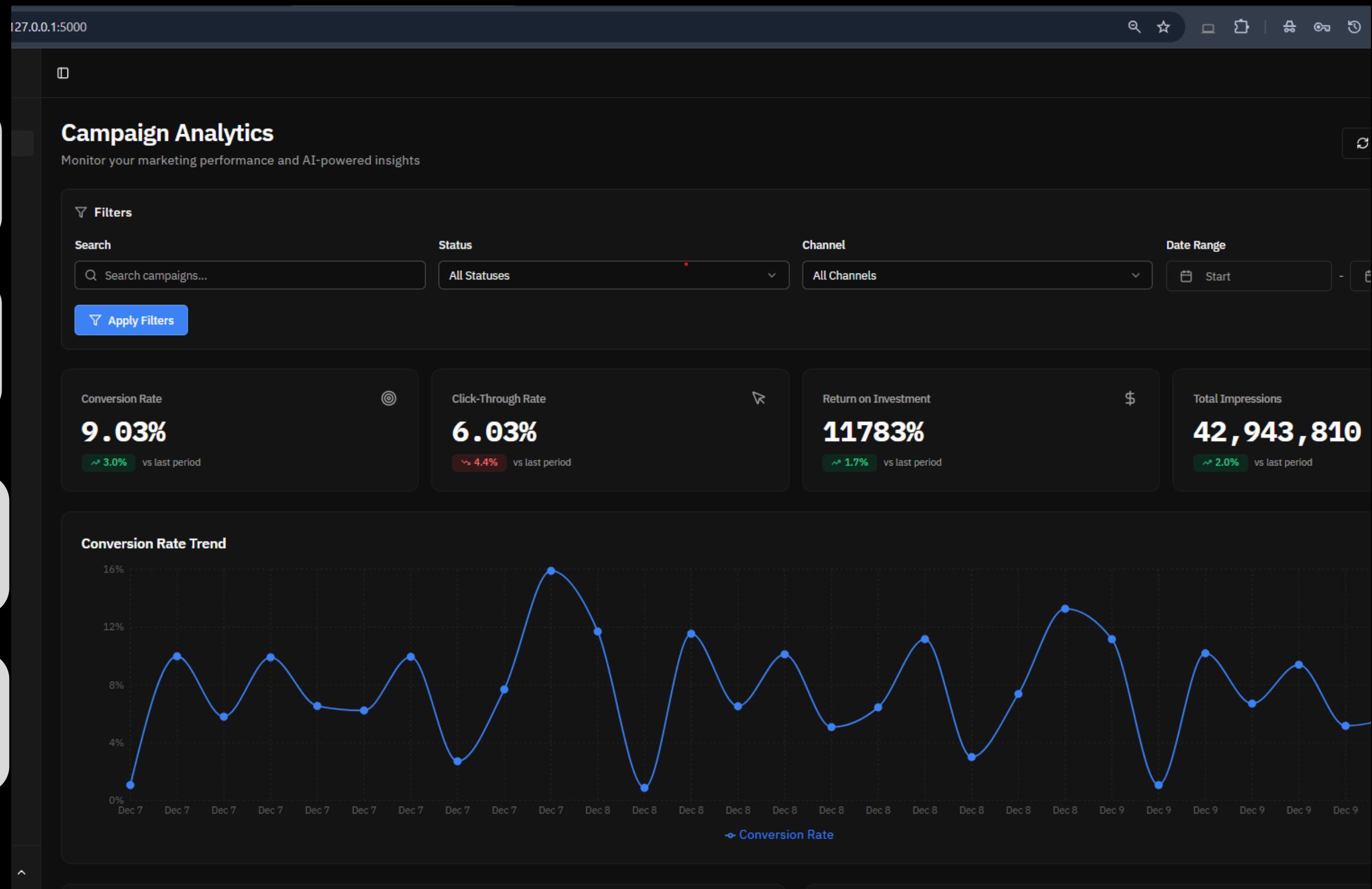
# Interactive Dashboard & Visualization

**Key Metrics:-** Interactive charts (Line, Bar, Area) and KPI cards for Conversion Rate, CTR, & ROI.

**Predictive Layer:-** Dedicated Forecast Card with AI-powered insights from the Model API integration.

**UX/Design:-** Modern design with Dark/Light theme support and adherence to accessibility standards.

**Data Alignment:-** Schema aligns with the Kaggle Marketing Campaign Performance Dataset structure.



# Data & API Structure

 **Database Schema:**- PostgreSQL schema includes campaigns, campaign\_metrics, forecasts, and pre-aggregated kpi\_summaries.

 **CRUD Operations:**- Full CRUD for campaign management with clear RESTful endpoints (/api/campaigns).

 **Documentation:**- Comprehensive Swagger/OpenAPI documentation exposed at /api/docs.

 **Efficiency:**- Implemented Pagination, Sorting, and Filtering on campaign listings

Campaign Name	Status	Channel	Budget
Innovate Industries - VIP Member Exclusive	Active	Display Ads	\$65,128
TechCorp - PPC Optimization	Completed	PPC	\$16,770
Innovate Industries - Holiday Gift Guide	Paused	PPC	\$42,298
TechCorp - Social Media Contest	Draft	Affiliate	\$97,918
Alpha Innovations - Early Bird Discount	Completed	Display Ads	\$6,737
Innovate Industries - Influencer Partnership	Active	Display Ads	\$60,906
DataTech Solutions - Back to School Promotion	Completed	Email	\$72,013
NexGen Systems - Seasonal Clearance	Active	PPC	\$69,597
Alpha Innovations - Referral Program Boost	Active	Social Media	\$63,753
TechCorp - Newsletter Signup Drive	Active	PPC	\$32,796

# Performance & Resilience

## Caching

In-memory cache with TTL (Time-To-Live) support for frequently accessed data (e.g., dashboard summaries).

## Pre-Aggregation

Used kpi\_summaries table to store pre-aggregated data for fast dashboard loading.

## Model Resilience

Circuit Breaker pattern implemented for the external Model API to prevent cascading failures.

## Validation

Utilized Zod schemas for strict, fast server-side input validation.



# Enterprise-Grade Security

## Authentication

---

**JWT Authentication** flow with short-lived **Access Tokens** (15 min) and longer **Refresh Tokens** (7 days).

## Authorization

---

**Role-Based Access Control (RBAC)** implemented: Admin, Analyst, and Viewer roles.

## Token Security

---

Refresh tokens stored securely using **HttpOnly Secure Cookies** and server-side storage in refresh\_tokens table.

## Revocation

---

Token revocation supported via the token\_blacklist table (maintaining a blacklist).



# Robust Validation & Logging

**Server Validation:**- Strict API input validation using Zod (fulfilling the JSON schema requirement).

**Error Handling:**- Centralized error handling middleware to categorize errors (400, 401, 403, 404, 500).

**Traceability:**- Structured logging includes Correlation IDs (request id) for tracing across services.

**Security Log:**- Sensitive data is masked before logging to ensure PII compliance.

**Unit Tests:-** Jest used for unit testing cache, circuit breaker, and validation logic.

**Integration Tests:-** Jest tests validate API endpoint structure and data flow with the database.

**Coverage:-** Achieved the target of 50% coverage for core logic.

**External Mocks:-** External dependencies (Model API) are mocked during unit testing.



# Automated CI/CD with GitLab

## Containerization

Dockerfile implemented using a Multi-stage Build for optimized production images.

## Code Quality

**ESLint** configuration ensures consistent code quality and style checks.

## GitLab CI

CI/CD pipeline automates stages for Linting, Unit/Integration Testing, Building, and Deployment.

## Security Scan

Automated npm audit integrated for **vulnerability scanning**.

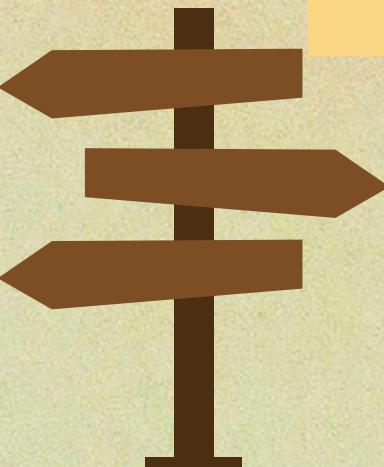
# Summary & Demonstrated Success

Fulfillment:- Successfully delivered all Expected Deliverables including forecasting, optimization, security, and CI/CD.

Key Takeaway:- Platform is secure, highly performant, and ready for enterprise deployment.

Next Steps:- Implement E2E tests (Cypress/Selenium) and establish a key rotation strategy for RS256.

Call to Action:- Demo: Viewing Campaign Dashboard and API Documentation.



THONK  
YOU

