ENTERPRISE INTEGRATION GUIDE

NeoStrike

Fast Track-Native Integration Platform

- 100% Serverless
- RabbitMQ-Powered
- Launch in 48 Hours

Version 4.1

February 2026

Table of Contents

1. Executive Summary

NeoStrike is a serverless integration middleware for Fast Track-native casinos. We provide game provider adapters (BYOC), payment orchestration (BYOC), RabbitMQ-based Fast Track event streaming, Game Catalog API, and an open-source React reference frontend with SDK.

> **100% Serverless Architecture**
>
> NeoStrike runs entirely on Vercel Edge Functions with managed services (Supabase PostgreSQL, CloudAMQP RabbitMQ, Upstash Redis). Everything scales automatically from zero to thousands of concurrent users with no server management required.

What NeoStrike Provides

- Game provider adapters: Evolution, Pragmatic Play, NetEnt (BYOC)
- Payment orchestration: Adyen + Stripe with smart routing
- Fast Track integration: RabbitMQ async event streaming
- Auto Duty of Care: Rule-based behavioral monitoring (UKGC/MGA compliant)
- Game Catalog API: 847+ games, single endpoint
- Reference Frontend + SDK: Open-source React starter kit
- Regulatory reporting: Automated UKGC + MGA compliance

2. What is NeoStrike?

NeoStrike is integration orchestration for Fast Track-powered casinos. We sit between your frontend and third-party services, handling the complexity of game providers, payment processors, and CRM event streaming.

2.1 Core Middleware (BYOC Model)

- Game provider adapters: Evolution, Pragmatic Play, NetEnt (you own contracts)
- Payment orchestration: Adyen + Stripe with smart routing (99.2% approval)

- Fast Track integration: RabbitMQ event streaming (10 event types)
- Auto Duty of Care: Rule-based behavioral monitoring (UKGC/MGA compliant)
- Regulatory reporting: Automated UKGC MRR + MGA quarterly reports

2.2 Low-Risk Accelerators

- Game Catalog API: Single endpoint for 847+ games across all providers
- Reference Frontend: React + Tailwind template (MIT licensed)
- Frontend SDK: @neostrike/sdk with React hooks (npm published)

2.3 What We Don't Touch

- Player data custody (you're the GDPR Data Controller)
- Wallet custody (no financial liability for NeoStrike)
- Bonus engines (campaigns managed via Fast Track)
- Campaign logic (Fast Track's domain)

3. Quick Start: Launch Timelines

3.1 The 48-Hour Advantage (Backend Only)

NeoStrike's backend integration can be deployed and configured in 48 hours. This includes connecting your game providers (Evolution, Pragmatic, NetEnt), payment processors (Adyen, Stripe), and Fast Track CRM via RabbitMQ. After 48 hours, your backend APIs are live and ready to accept requests from any frontend.

3.2 The 2-Week Launch (Full Platform)

A complete, branded casino platform (backend + frontend) can be launched in 2 weeks. This includes the 48-hour backend setup plus an additional 10-12 days for frontend customization using our reference template. Operators fork our React starter kit, customize branding (logo, colors, terms & conditions), and deploy to Vercel. The result is a fully branded, player-facing casino live in 2 weeks instead of the industry standard 6-8 weeks.

Timeline Breakdown

Phase	Duration	What Gets Done
Backend Setup	48 hours	Deploy Vercel APIs, configure providers, connect Fast Track
Frontend Customization	10-12 days	Fork React template, customize branding, deploy
Total	**2 weeks**	**Fully branded casino live with backend + frontend**

3.3 Week 1: Backend Configuration

The first 48 hours can complete the backend configuration:

```
STEP 1: Deploy NeoStrike Backend
```

```
git clone https://github.com/neostrike/backend
cd backend
vercel deploy

# Auto-provisions: Supabase, CloudAMQP, Upstash
# Your API is now live at: https://your-api.vercel.app
```

```
STEP 2: Configure Game Providers (Tenant Portal)
```

```
# Portal > Game Providers > Add Provider

# Evolution: Enter casino key + API token
# Pragmatic: Enter casino ID + secret key
# NetEnt: Enter merchant ID + API key

# Test connections (2 min per provider)
```

```
STEP 3: Configure Payment Providers
```

```
# Portal > Payments > Add Provider

# Adyen: Merchant account + API key
# Stripe: Publishable + secret key

# Test connections + enable smart routing
```

STEP 4: Connect Fast Track CRM

```
# Portal > Integrations > Fast Track

# Enter Fast Track API URL + credentials
# RabbitMQ connection auto-configured
# Test event delivery

# Backend APIs ready in 48 hours
```

3.4 Week 2: Frontend Deployment

Using the Reference Template:

STEP 1: Scaffold with Starter Kit

```
npx @neostrike/create-casino my-casino
cd my-casino
npm install
```

STEP 2: Customize Branding

```
# Edit theme.config.js:

# Update logo, colors, fonts
# Modify landing page content
# Add terms & conditions
```

STEP 3: Deploy to Vercel

```
vercel deploy

# Production URL live in 2 minutes
# Fully branded casino live in 2 weeks total
```

4. Fast Track Integration via RabbitMQ

NeoStrike uses RabbitMQ for Fast Track integration, providing async event streaming with guaranteed delivery and zero impact on API performance.

4.1 How It Works

Player places 10 EUR bet

```
POST /api/v1/debit
{
  "player_id": "player_123",
  "amount": 10.00,
  "game_id": "evolution:lightning-roulette"
}
```

Your Vercel Edge Function

```
# 1. Updates PostgreSQL balance (15ms)
# 2. Publishes to RabbitMQ (5ms, async)
# 3. Returns response to player (35ms total)

Response:
{
  "balance": 90.00,
  "transaction_id": "txn_abc123",
  "alerts": [],
  "notifications": []
}
```

Meanwhile, RabbitMQ

```
# 4. Stores event in durable queue
# 5. Fast Track polls queue at their own pace
# 6. Fast Track processes for campaigns (no impact on your API)

# Key benefit: Your API returns in 35ms instead of 100-200ms
```

4.2 Event Types Sent to Fast Track

Event	Trigger	Fast Track Use Case
REGISTRATION	Player signs up	Welcome campaign
LOGIN	Player authenticates	Session tracking
BET	Wager placed	Activity segmentation

WIN	Payout received	Win celebration
DEPOSIT	Funds added	Deposit bonus trigger
WITHDRAWAL	Funds requested	VIP notification
BONUS_AWARDED	Bonus credited	Campaign tracking
KYC_UPDATE	Verification status	Compliance workflow
CASINO	Casino game played	Casino activity tracking
SPORTSBOOK	Sports bet placed	Sportsbook activity tracking

4.3 Benefits of RabbitMQ Integration

- **Async:** API returns in 35ms (Fast Track processing happens separately)
- **Reliable:** Messages never lost (durable queues + persistence)
- **Resilient:** Fast Track downtime doesn't affect your platform
- **Simpler:** No retry logic needed in your code
- **Guaranteed Delivery:** ACK-based consumption

5. Real-Time Player Updates

NeoStrike delivers real-time balance and alert updates through stateless API responses. Every player action already triggers an API call (placing a bet, checking balance, making a deposit). We include state updates directly in these response payloads, eliminating the need for separate push infrastructure.

How It Works

When a player interacts with your casino, their action hits your NeoStrike API. The API processes the request, updates the database, and returns a response that includes not just the result of the action, but also the player's current balance, any pending alerts (Duty of Care warnings), and notifications (bonus awards, achievements).

Example: Player Places a Bet

Player clicks "Spin 10 EUR" in the frontend

```
# Frontend sends:
POST /api/v1/debit
{
  "player_id": "player_123",
  "amount": 10.00,
  "game_id": "evolution:lightning-roulette",
  "round_id": "round_xyz"
}
```

NeoStrike API response (35ms)

```
{
  "success": true,
  "balance": 90.00,
  "bonus_balance": 10.00,
  "currency": "EUR",
  "transaction_id": "txn_abc123",
  "alerts": [
    {
      "type": "session_time",
      "message": "You've been playing for 90 minutes. Take a break?",
      "severity": "warning",
      "action": "reality_check"
    }
  ],
  "notifications": [
    {
      "type": "bonus_awarded",
      "message": "You received a 10 EUR welcome bonus!",
      "amount": 10.00,
      "timestamp": "2026-02-17T15:30:00Z"
    }
  ]
}
```

Frontend automatically

```
# 1. Updates balance display (90.00 EUR)
# 2. Shows alert modal (session time warning)
# 3. Shows notification toast (bonus awarded)
```

Benefits of Stateless Responses

- **Simpler Architecture:** No separate push infrastructure to manage
- **Lower Cost:** No always-on connection servers required
- **Reliable:** Updates are guaranteed (part of the API response)
- **Sufficient Latency:** ~100ms is acceptable for iGaming (players act every few seconds)
- **Auto-Scaling:** 100% serverless, scales from 0 to thousands of users

6. Game Catalog API

The Game Catalog API aggregates game metadata from all enabled providers into a single, searchable endpoint. This eliminates 2-3 weeks of boilerplate work for every operator.

Query Games

```
GET                                                    /api/v1/games/catalog?
provider=evolution,pragmatic&category=slots&search=roulette&page=1&limit=50
```

Response


```
{
  "games": [
    {
      "id": "evolution:lightning-roulette",
      "name": "Lightning Roulette",
      "provider": "evolution",
      "category": "live-casino",
      "thumbnail": "https://cdn.evolution.com/games/lightning-roulette.jpg",
      "rtp": 97.3,
      "min_bet": 0.20,
      "max_bet": 5000.00,
      "has_demo": false,
      "tags": ["roulette", "live", "featured"]
    },
    {
      "id": "pragmatic:gates-of-olympus",
      "name": "Gates of Olympus",
      "provider": "pragmatic",
      "category": "slots",
      "thumbnail": "https://cdn.pragmaticplay.com/games/gates-olympus.jpg",
      "rtp": 96.5,
      "min_bet": 0.20,
      "max_bet": 100.00,
      "has_demo": true,
      "tags": ["slots", "high-volatility", "multipliers"]
    }
  ],
  "total": 847,
  "page": 1,
  "limit": 50
}
```


Launch Game

POST `/api/v1/games/evolution:lightning-roulette/launch`

Request


```
{
  "player_id": "player_123",
  "mode": "real"
}
```

Response

```
{
  "game_url": "https://game.evolution.com/frontend/evo/r2?token=eyJhb...",
  "session_id": "evo_session_abc123",
  "expires_at": "2026-02-17T16:30:00Z"
}

# Frontend loads game_url in iframe
```

Benefits

- Single API for all games across providers
- Searchable and filterable
- Cached via Upstash Redis (5ms response time)
- Provider metadata synced nightly
- Low risk (information only, no player data)

7. Reference Frontend + SDK

The NeoStrike Starter Kit provides an open-source React template and npm SDK to eliminate 6-8 weeks of frontend boilerplate.

@neostrike/sdk (React Hooks)

Install SDK

```
npm install @neostrike/sdk
```

Use hooks in your components

```
import { useBalance, useGames, usePayments } from '@neostrike/sdk';

function WalletBalance({ playerId }) {
  const { balance, bonusBalance, loading, error } = useBalance(playerId);

  if (loading) return <Spinner />;
  if (error) return <Error message={error.message} />;

  return (
    <div>
      EUR {balance.toFixed(2)}
      <small>Bonus: EUR {bonusBalance.toFixed(2)}</small>
    </div>
  );
}
```

Available Hooks

Hook	Returns
`useBalance(playerId)`	{ balance, bonusBalance, loading, error, refresh }
`useGames(filters)`	{ games, total, loading, search, filter, loadMore }
`usePayments(playerId)`	{ deposit, withdraw, history, loading }
`useAlerts(playerId)`	{ alerts, dismiss, loading }
`useSession()`	{ player, login, logout, register, isAuthenticated }

Reference Frontend Scaffold

Scaffold new casino project

```
npx @neostrike/create-casino my-casino
cd my-casino
npm install

# Project structure:
my-casino/
  src/
    components/
      GameLobby/        # Game grid with search/filter
      WalletDrawer/     # Balance, deposit, withdraw
      AuthModal/        # Login, register, password reset
      AlertModal/       # Auto Duty of Care popups
      PlayerProfile/    # Limits, history, KYC status
    hooks/              # From @neostrike/sdk
    theme.config.js     # Customize: logo, colors, fonts
    App.jsx
  .env.example          # API keys and config
  README.md             # Customize branding
```

Edit theme.config.js

```
export default {
  logo: '/assets/logo.png',
  colors: {
    primary: '#0066cc',
    secondary: '#10b981'
  },
  fonts: {
    heading: 'Inter',
    body: 'Roboto'
  }
}
```

Deploy to Vercel

```
vercel deploy

# Your casino is live at https://my-casino.vercel.app
```

8. Auto Duty of Care

NeoStrike provides rule-based behavioral monitoring for responsible gaming compliance (UKGC/MGA). The system detects 5 patterns and automatically triggers interventions.

5 Behavioral Patterns

Pattern	Detection Rule	Auto-Intervention
Chasing Losses	5+ consecutive deposits after net loss	Reality check + cooling-off offer
Velocity Spike	Bet frequency > 3x baseline	Session time alert + mandatory break
Rapid Escalation	Bet size increases 10x in 30 min	Mandatory 5-minute break
Prolonged Session	Continuous play > 90 minutes	Reality check notification
Affordability Breach	Deposits exceed 1,000 EUR in 30 days	Affordability questionnaire required

How Interventions Work

Auto Duty of Care detects pattern

```
# Player has been playing for 95 minutes (prolonged session)
# Next API call returns:

POST /api/v1/debit
{
  "player_id": "player_123",
  "amount": 10.00,
  "game_id": "pragmatic:gates-of-olympus"
}

# Response includes alert:
{
  "balance": 80.00,
  "transaction_id": "txn_xyz",
  "alerts": [
    {
      "type": "prolonged_session",
      "message": "You've been playing for 95 minutes. Your balance is EUR 80.00.",
      "severity": "warning",
      "action": "reality_check",
      "buttons": [
        { "label": "Take a Break", "action": "cooling_off" },
        { "label": "Continue Playing", "action": "acknowledge" }
      ]
    }
  ]
}

# Frontend shows modal, player must respond before continuing
```

Why Rule-Based (Not ML)

- **Explainability:** You can tell regulators exactly why a player was flagged
- **Audit Defense:** Clear rules pass compliance audits immediately
- **No Training Data:** Works from day one without historical player data
- **UKGC/MGA Compliant:** Meets all current regulatory requirements
- **ML Upgrade Available:** After 10 operators (sufficient training data)

9. Security & BYOC Model

Legal Positioning: NeoStrike is a Technical Service Provider (TSP). We process events but never own player data or hold funds.

What You Own

- Game provider contracts (Evolution, Pragmatic, NetEnt)
- Payment processor contracts (Adyen, Stripe)
- Player data (you're GDPR Data Controller)
- Wallet custody (we route, you control)
- Merchant of Record status

What NeoStrike Does

- API translation (your credentials to provider APIs)
- Event routing (your backend to RabbitMQ to Fast Track)
- Payment orchestration (smart routing, retry)
- Game metadata aggregation (catalog API)

Credential Security

Security Measure	Implementation
Encryption at rest	AES-256
Encryption in transit	TLS 1.3
Isolation	Per-operator vaults (never co-mingled)
Rotation	90-day reminders via Tenant Portal
Access logging	Full audit trail of credential usage

10. API Reference

Debit (Place Bet)

```
POST  /api/v1/debit
```

Request

```
Content-Type: application/json
Authorization: Bearer YOUR_API_KEY

{
  "player_id": "player_123",
  "amount": 10.00,
  "game_id": "evolution:lightning-roulette",
  "round_id": "round_xyz"
}
```

Response 200 OK

```
{
  "success": true,
  "balance": 90.00,
  "bonus_balance": 10.00,
  "transaction_id": "txn_abc123",
  "alerts": [],
  "notifications": []
}
```

Credit (Win Payout)

```
POST  /api/v1/credit
```

Request

```
Content-Type: application/json
Authorization: Bearer YOUR_API_KEY

{
  "player_id": "player_123",
  "amount": 25.00,
  "game_id": "evolution:lightning-roulette",
  "round_id": "round_xyz"
}
```

Response 200 OK

```
{
  "success": true,
  "balance": 115.00,
  "bonus_balance": 10.00,
  "transaction_id": "txn_def456"
}
```

Balance Query

```
GET  /api/v1/balance?player_id=player_123
```

Response 200 OK

```
{
  "balance": 115.00,
  "bonus_balance": 10.00,
  "currency": "EUR",
  "last_updated": "2026-02-17T15:45:00Z"
}
```

Deposit

```
POST  /api/v1/deposit
```

Request

```
Content-Type: application/json
Authorization: Bearer YOUR_API_KEY

{
  "player_id": "player_123",
  "amount": 50.00,
  "method": "card",
  "return_url": "https://my-casino.com/deposit/complete"
}
```

Response 200 OK

```
{
  "payment_url": "https://checkout.adyen.com/...",
  "transaction_id": "txn_ghi789",
  "expires_at": "2026-02-17T16:00:00Z"
}
```

11. Pricing & Business Model

Tier	Monthly Fee	Transaction Fee	Target Segment
Startup	EUR 0	2.5% of GGR	< EUR 50K GGR/month
Growth	EUR 2,500	2.0% of GGR	EUR 50K-500K GGR/month
Enterprise	Custom	1.5% of GGR	> EUR 500K GGR/month

What You're Paying For

- Serverless API infrastructure (Vercel Edge Functions)
- Game provider adapters (Evolution, Pragmatic, NetEnt - BYOC)
- Payment orchestration (Adyen + Stripe, 99.2% approval)
- RabbitMQ Fast Track integration (guaranteed delivery)
- Game Catalog API (847+ games)
- Auto Duty of Care (rule-based, UKGC/MGA compliant)

- Regulatory reporting automation (UKGC + MGA)
- Reference frontend + SDK (MIT licensed, open-source)
- Tenant Portal (self-service configuration)
- 24/7 monitoring + support