NY Sharp Edge - Sportsbook Odds Screener

A React-based sportsbook odds screener and +EV betting tool, focused specifically on New York legal sportsbooks. Built to identify positive expected value bets, arbitrage opportunities, and sharp money signals across traditional sportsbooks, prediction markets, and betting exchanges.

> **Portfolio Project by Jack (@riftfern)**
> Self-taught software engineer demonstrating full-stack React development, API integration, real-time data handling, and algorithmic trading concepts.

Table of Contents

Project Overview

NY Sharp Edge is an odds comparison and +EV betting tool similar to OddsJam, but built from scratch as a personal portfolio project. The screener aggregates odds from all legal NY sportsbooks plus prediction markets (Kalshi, Polymarket) and sweepstakes exchanges (NoVig, ProphetX) to surface:

- **+EV Bets**: Mathematically profitable betting opportunities
- **Arbitrage Opportunities**: Risk-free profit when odds differ across books
- **Sharp Money Signals**: Line movements indicating where professional bettors are placing action
- **Low Hold Markets**: Betting opportunities with minimal house edge
- **Liquidity Making Opportunities**: Bets to post on Kalshi/Polymarket as a market maker

---

Why This Project

For Hiring Managers

This project demonstrates:

- **Real-world problem solving**: Not a tutorial clone—built to solve my actual betting workflow
- **Complex state management**: Real-time odds updates across 10+ data sources
- **API integration at scale**: Handling rate limits, normalization, and caching
- **Algorithmic thinking**: EV calculations, no-vig fair odds, Kelly Criterion
- **Financial/trading concepts**: Order books, liquidity, market making
- **Full-stack capability**: React frontend + Node.js backend + database layer

For Personal Use

I'm an active sports bettor in NYC. This tool consolidates my workflow:

- Stop manually comparing odds across 9 different sportsbook apps
- Systematically find +EV opportunities instead of gut-feel betting
- Tail sharp money from exchanges like NoVig/ProphetX
- Identify arbitrage between traditional books and prediction markets
- Post profitable lines as a liquidity maker on Kalshi/Polymarket

Core Features

Phase 1: Odds Comparison Dashboard

```
┌─────────────────────────────────────────────────────────────────┐
│ NY Sharp Edge                    [NFL ▼] [Today ▼]    │
├─────────────────────────────────────────────────────────────────┤
│ Bills @ Jets - 1:00 PM                           │
│ ───────────────────────────────────────────────────────────────│
│ SPREAD      DK     FD     MGM     CZR     BR     BEST     │
│ Bills -3      -110    -108    -110    -112    -110    FD -108 │
│ Jets +3       -110    -112    -110    -108    -110    CZR -108│
│ ───────────────────────────────────────────────────────────────│
│ TOTAL        DK     FD     MGM     CZR     BR     BEST     │
│ Over 43.5     -110    -110    -105    -110    -110    MGM-105 │
│ Under 43.5    -110    -110    -115    -110    -110    DK -110 │
│ ───────────────────────────────────────────────────────────────│
│ MONEYLINE    DK     FD     MGM     CZR     BR     BEST     │
│ Bills        -155    -152    -155    -160    -155    FD -152 │
│ Jets         +130    +128    +130    +135    +130    CZR +135│
└─────────────────────────────────────────────────────────────────┘
```

Phase 2: +EV Bet Finder

```
┌─────────────────────────────────────────────────────────────────┐
│ +EV OPPORTUNITIES                [Refresh 🔄] [5s ⏱️]   │
├─────────────────────────────────────────────────────────────────┤
│ EV%   Event              Market        Book   Odds    │
│ ───────────────────────────────────────────────────────────────│
│ +4.2%  Bills @ Jets          Bills ML        FD      -152    │
│        Fair: -165 | Edge: $4.20/$100                  │
│ ───────────────────────────────────────────────────────────────│
│ +3.8%  Lakers @ Knicks       LeBron O25.5    DK      -105    │
│        Fair: -115 | Edge: $3.80/$100                  │
│ ───────────────────────────────────────────────────────────────│
│ +2.1%  Yankees vs Red Sox    Over 8.5        MGM     +100    │
│        Fair: -104 | Edge: $2.10/$100                  │
└─────────────────────────────────────────────────────────────────┘
```

Phase 3: Arbitrage Finder

```
┌─────────────────────────────────────────────────────────────────┐
│ ARBITRAGE OPPORTUNITIES                    [Active: 3]          │
├─────────────────────────────────────────────────────────────────┤
│ GUARANTEED PROFIT: 2.3%                                         │
│ Event: Bills @ Jets | Market: Spread                            │
│ ─────────────────────────────────────────────────────────────── │
│ Leg 1: Bills -3 @ FanDuel        -108    Stake: $520.83         │
│ Leg 2: Jets +3 @ Caesars         -108    Stake: $479.17         │
│ ─────────────────────────────────────────────────────────────── │
│ Total Stake: $1,000.00 | Guaranteed Profit: $23.00              │
│                                      [PLACE BETS →]             │
└─────────────────────────────────────────────────────────────────┘
```

Phase 4: Exchange/Prediction Market Integration

```
┌─────────────────────────────────────────────────────────────────┐
│ KALSHI/POLYMARKET ARBITRAGE                                     │
├─────────────────────────────────────────────────────────────────┤
│ Cross-Market Opportunity Found!                                 │
│ ─────────────────────────────────────────────────────────────── │
│ Event: Chiefs to win Super Bowl                                 │
│                                                                 │
│ Kalshi:      YES @ $0.22 (implied 22%)                          │
│ DraftKings:  Chiefs +650 (implied 13.3%)                        │
│                                                                 │
│ Action: BUY on DraftKings, consider SELL on Kalshi              │
│ Edge: 8.7% implied probability difference                       │
└─────────────────────────────────────────────────────────────────┘
```

Phase 5: Liquidity Making Dashboard

```
┌─────────────────────────────────────────────────────────────────┐
│ LIQUIDITY MAKING - Post Profitable Lines                        │
├─────────────────────────────────────────────────────────────────┤
│ Suggested Posts for Kalshi/Polymarket:                          │
│ ─────────────────────────────────────────────────────────────── │
│ Market: Bills to cover -3                                       │
│ Sharp consensus: 52.1% true probability                         │
│ Current Kalshi YES: $0.48 | NO: $0.54                           │
│                                              │
│  Suggested: POST YES @ $0.50 (lock in 2.1% edge)          │
│           Expected fill rate: High (tight spread)        │
│                              [COPY TO KALSHI]  │
└──────────────────────────────────────────────────────────────┘
```

Technical Architecture

```
┌──────────────────────────────────────────────────────────────┐
│                   FRONTEND (React)                │
│  ┌─────────────┐ ┌─────────────┐ ┌─────────────┐ ┌───────────┐  │
│  │ Odds Screen │ │ +EV Finder │ │ Arb Finder │ │ Liquidity │ │
│  └──────┬──────┘ └──────┬──────┘ └──────┬──────┘ └─────┬─────┘  │
│      └──────────────┴──────────────┴──────────────┘   │
│                 │            │
│           ┌─────────▼─────────┐           │
│           │ State Manager  │        │
│           │ (React Query / │        │
│           │ Zustand)   │        │
│           └─────────┬─────────┘           │
└─────────────────────────────┼────────────────────────────────┘
                │ WebSocket / REST
┌─────────────────────────────┼────────────────────────────────┐
│                   BACKEND (Node.js)                │
│           ┌─────────▼─────────┐           │
│           │ API Gateway  │        │
│           └─────────┬─────────┘           │
│    ┌──────────────────┼──────────────────┐      │
│  ┌─────▼─────┐ ┌─────────▼─────────┐ ┌─────▼─────┐     │
│  │ Odds     │ │ Calculation Engine │ │ Alert    │   │
│  │ Aggregator│ │ - EV Calculator   │ │ Service  │   │
│  │      │ │ - No-Vig Fair Odds │ │      │   │
│  │      │ │ - Arb Detector    │ │      │   │
│  └──────────┘ │ - Kelly Criterion │ └──────────┘     │
│    │     └────────────────────┘         │
└──────────┼───────────────────────────────────────────────────┘
     │
┌──────────▼───────────────────────────────────────────────────┐
│             EXTERNAL DATA SOURCES             │
│  ┌─────────────┐ ┌─────────────┐ ┌─────────────┐ ┌───────────┐  │
│  │ The Odds  │ │ Kalshi API │ │ Polymarket │ │ NoVig/ │ │
│  │ API (Free)  │  │ API         │  │ ProphetX    │  │
│  └─────────────┘  └─────────────┘  └─────────────┘  └─────────────┘  │
│                                                                 │
│  ┌─────────────┐  ┌─────────────┐  ┌─────────────┐              │
│  │ SportsData  │  │ Direct Book │  │ Sharp Book  │              │
│  │ .io (Paid)  │  │ Scraping*   │  │ Consensus   │              │
│  └─────────────┘  └─────────────┘  └─────────────┘              │
└─────────────────────────────────────────────────────────────────────┘
                │
┌───────────────────────────────────▼──────────────────────────────────┐
│                DATABASE LAYER                               │
│  ┌─────────────┐  ┌─────────────┐  ┌─────────────┐              │
│  │ PostgreSQL  │  │ Redis Cache │  │ TimescaleDB │              │
│  │ (Bets/Users)│  │ (Live Odds) │  │ (Historical)│              │
│  └─────────────┘  └─────────────┘  └─────────────┘              │
└─────────────────────────────────────────────────────────────────────┘
```

NY Legal Sportsbooks

As of December 2025, New York has **9 licensed online sportsbooks**:

Sportsbook	Status	Notes
FanDuel	✅ Active	Market leader, great SGP builder
DraftKings	✅ Active	Second largest, extensive menu
BetMGM	✅ Active	Partner of Knicks/Rangers/MSG
Caesars	✅ Active	Official Mets partner
BetRivers	✅ Active	Solid software, good promos
Fanatics	✅ Active	Launched Feb 2024, FanCash rewards
Bally Bet	✅ Active	Smaller but decent odds
Bet365	✅ Active	European giant, competitive lines
theScore Bet	✅ Active	Launched Dec 2025 (PENN partner)

Important NY Restrictions

- ❌ Cannot bet on NY college teams (Syracuse, St. John's, Army, etc.)
- ❌ Cannot bet on college events held in NY state
- ❌ 51% state tax rate on operators (highest in US)
- ✅ All major professional sports available
- ✅ Player props, futures, live betting all legal

Exchanges & Prediction Markets

Sharp/No-Vig Sources (For Fair Odds Calculation)

Platform	Type	NY Access	API	Notes
NoVig	Sweepstakes Exchange	✅ Yes	Unofficial	P2P, no vig, ~2% commission on net
ProphetX	Sweepstakes Exchange	✅ Yes	Unofficial	Trading-style interface, custom odds
Pinnacle	Sharp Sportsbook	❌ Offshore	API available	Gold standard for fair odds

Prediction Markets (For Cross-Market Arb & Market Making)

Platform	Type	NY Access	API	Notes
Kalshi	CFTC-Regulated	✅ Yes	✅ Full REST/WS	Sports contracts, NHL partnership
Polymarket	Crypto/Decentralized	⚠️ Limited	✅ Yes	USDC on Polygon, US waitlist

API Documentation Links

- Kalshi: docs.kalshi.com
- Polymarket: Uses CLOB (Central Limit Order Book) API
- The Odds API: the-odds-api.com
- SportsDataIO: sportsdata.io

Data Sources & APIs

Primary Odds Data

API	Cost	Features	Use Case
The Odds API	Free tier (500 req/mo)	US books, h2h/spreads/totals	MVP development
SportsDataIO	Paid (~$50-600/mo)	Full props, alts, live odds	Production
OddsJam API	Paid ($99-299/mo)	Pre-calculated EV, arbs	Validation/backup

Prediction Market APIs

API	Cost	Features
Kalshi	Free	Full order book, historical, WebSocket
Polymarket	Free	CLOB data, market metadata
FinFeedAPI	Paid	Unified Kalshi/Polymarket data

Supplementary Data

API	Use
ESPN API	Schedules, scores, injuries
Sportradar	Premium sports data (expensive)

Development Phases

Phase 1: Foundation (Weeks 1-3)

Goal: Basic odds comparison for NY books

☐ Project setup (Vite + React + TypeScript)

☐ The Odds API integration

- □ Display odds grid for NFL/NBA
- □ Filter by sport, date, market type
- □ Highlight best odds per market
- □ Basic responsive design

Deliverable: Working odds comparison screen

Phase 2: +EV Engine (Weeks 4-6)

Goal: Calculate and surface +EV opportunities

- □ No-vig fair odds calculator
- □ Sharp book consensus (Pinnacle proxy via offshore data)
- □ EV calculation engine
- □ Sort/filter by EV%, edge, confidence
- □ Kelly Criterion stake suggestions
- □ Real-time updates (polling → WebSocket)

Deliverable: +EV bet finder with mathematical edge display

Phase 3: Arbitrage Detection (Weeks 7-8)

Goal: Identify risk-free arbitrage opportunities

- □ Cross-book arbitrage algorithm
- □ Stake calculator for guaranteed profit
- □ Alert system for new arbs
- □ Track arb windows (how long opportunities last)
- □ Middle bet detection (optional lottery tickets)

Deliverable: Arbitrage finder with stake recommendations

Phase 4: Exchange Integration (Weeks 9-11)

Goal: Connect NoVig, ProphetX, Kalshi, Polymarket

- □ Kalshi API integration (authenticated)
- □ Polymarket data feed

- □ NoVig/ProphetX data (scraping or unofficial API)
- □ Cross-market arbitrage detection
- □ Exchange-to-sportsbook comparison view
- □ Order book visualization

Deliverable: Unified view across traditional books + exchanges

---

Phase 5: Liquidity Making Tools (Weeks 12-14)

Goal: Identify profitable lines to post as market maker

- □ Calculate fair odds from aggregate market data
- □ Identify mispriced Kalshi/Polymarket contracts
- □ Suggest bid/ask to post
- □ Track posted orders and fill rates
- □ P&L tracking for market making activity

Deliverable: Market making dashboard for Kalshi/Polymarket

Phase 6: Polish & Deploy (Weeks 15-16)

Goal: Production-ready application

- □ Authentication (save preferences, track bets)
- □ Bet tracker with ROI analytics
- □ Push notifications for high-EV opportunities
- □ Performance optimization
- □ Deploy to Vercel/Railway
- □ Documentation & demo video

Deliverable: Deployed, documented portfolio piece

EV & Arbitrage Calculations

No-Vig Fair Odds Formula

Convert American odds to implied probability, remove vig, convert back:

javascript

```javascript
// American odds to decimal
function americanToDecimal(odds) {
  if (odds > 0) return (odds / 100) + 1;
  return (100 / Math.abs(odds)) + 1;
}

// Decimal odds to implied probability
function decimalToImplied(decimal) {
  return 1 / decimal;
}

// Remove vig (multiplicative method)
function calculateNoVigOdds(odds1, odds2) {
  const implied1 = decimalToImplied(americanToDecimal(odds1));
  const implied2 = decimalToImplied(americanToDecimal(odds2));

  const totalImplied = implied1 + implied2; // This is > 1 due to vig

  const fairImplied1 = implied1 / totalImplied;
  const fairImplied2 = implied2 / totalImplied;

  return {
    fairProb1: fairImplied1,
    fairProb2: fairImplied2,
    vigPercentage: (totalImplied - 1) * 100
  };
}
```

Expected Value Calculation

javascript

```
function calculateEV(betOdds, fairProbability, stake = 100) {
  const decimalOdds = americanToDecimal(betOdds);
  const potentialProfit = stake * (decimalOdds - 1);

  const ev = (fairProbability * potentialProfit) - ((1 - fairProbability) * stake);
  const evPercentage = (ev / stake) * 100;

  return {
    ev,
    evPercentage,
    edge: fairProbability - decimalToImplied(decimalOdds)
  };
}
```

Arbitrage Detection

javascript

```
function detectArbitrage(odds1, odds2) {
  const decimal1 = americanToDecimal(odds1);
  const decimal2 = americanToDecimal(odds2);

  const totalImplied = (1 / decimal1) + (1 / decimal2);

  if (totalImplied < 1) {
    const profit = (1 - totalImplied) * 100;
    const stake1 = (1 / decimal1) / totalImplied;
    const stake2 = (1 / decimal2) / totalImplied;

    return {
      isArbitrage: true,
      profitPercentage: profit,
      stake1Ratio: stake1,
      stake2Ratio: stake2
    };
  }

  return { isArbitrage: false };
}
```

Kelly Criterion

```javascript
function kellyStake(fairProb, decimalOdds, bankroll, fraction = 0.25) {
  const b = decimalOdds - 1; // Net odds
  const q = 1 - fairProb;

  const kelly = (fairProb * b - q) / b;
  const fractionalKelly = kelly * fraction; // Use 1/4 Kelly for safety

  return Math.max(0, fractionalKelly * bankroll);
}
```

Sharp Money Detection

Line Movement Analysis

Sharp money moves lines. Track these signals:

```javascript
```

```javascript
function analyzeLineMovement(openingOdds, currentOdds, publicBetPercentage) {
  const lineMovement = currentOdds - openingOdds;

  // Reverse line movement = sharp indicator
  // If public heavily on one side but line moves opposite
  if (publicBetPercentage > 70 && lineMovement < 0) {
    return {
      signal: 'SHARP_OPPOSITE',
      confidence: 'HIGH',
      side: 'underdog'
    };
  }

  // Steam move = sharp indicator
  // Significant line movement in short time
  if (Math.abs(lineMovement) > 20) { // 20 cents in American odds
    return {
      signal: 'STEAM_MOVE',
      confidence: 'MEDIUM'
    };
  }

  return { signal: 'NONE' };
}
```

NoVig/ProphetX as Sharp Proxy

Since these exchanges are peer-to-peer with sophisticated users:

javascript

```
function compareToExchange(bookOdds, exchangeOdds) {
  const bookImplied = decimalToImplied(americanToDecimal(bookOdds));
  const exchangeImplied = decimalToImplied(americanToDecimal(exchangeOdds));

  const discrepancy = exchangeImplied - bookImplied;

  // Exchange showing higher probability = sharp money believes this
  if (discrepancy > 0.03) { // 3% difference
    return {
      signal: 'EXCHANGE_VALUE',
      expectedValue: discrepancy * 100,
      action: 'Consider taking book odds'
    };
  }

  return { signal: 'ALIGNED' };
}
```

Liquidity Making Strategy

How It Works on Kalshi/Polymarket

1. **Identify mispriced contracts** using aggregated sportsbook data
2. **Post limit orders** at prices that give you edge
3. **Get filled** when other traders take your price
4. **Profit** from the spread between your price and true probability

Example Strategy

javascript

```
function findLiquidityOpportunity(event, kalshiBook, fairProbability) {
  const bestBid = kalshiBook.bids[0]?.price; // Highest buy order
  const bestAsk = kalshiBook.asks[0]?.price; // Lowest sell order

  // If fair prob is 52%, and Kalshi YES is at $0.48
  // Post YES at $0.50 → 2% edge if filled

  if (fairProbability > (bestAsk / 100) + 0.02) {
    return {
      action: 'POST_YES',
      suggestedPrice: Math.round(fairProbability * 100) - 2,
      expectedEdge: ((fairProbability * 100) - bestAsk) / 100
    };
  }

  if ((1 - fairProbability) > (100 - bestBid) / 100 + 0.02) {
    return {
      action: 'POST_NO',
      suggestedPrice: 100 - Math.round(fairProbability * 100) - 2,
      expectedEdge: ((100 - fairProbability * 100) - (100 - bestBid)) / 100
    };
  }

  return { action: 'NONE', reason: 'No edge available' };
}
```

Tech Stack

Frontend

- **React 18** - UI framework
- **TypeScript** - Type safety
- **Vite** - Build tool
- **TanStack Query** - Data fetching & caching
- **Zustand** - Global state management
- **Tailwind CSS** - Styling
- **Recharts** - Data visualization

- **React Router** - Navigation

Backend

- **Node.js + Express** - API server
- **TypeScript** - Type safety
- **PostgreSQL** - Primary database
- **Redis** - Caching layer
- **Bull** - Job queue for background tasks
- **Socket.io** - Real-time updates

DevOps

- **Vercel** - Frontend hosting
- **Railway/Render** - Backend hosting
- **GitHub Actions** - CI/CD
- **Sentry** - Error tracking

Folder Structure

```
ny-odds-screener/
├── apps/
│   ├── web/                    # React frontend
│   │   ├── src/
│   │   │   ├── components/
│   │   │   │   ├── odds/       # Odds grid components
│   │   │   │   ├── ev/         # +EV finder components
│   │   │   │   ├── arb/        # Arbitrage components
│   │   │   │   ├── exchange/   # Exchange integration
│   │   │   │   └── common/     # Shared UI components
│   │   │   ├── hooks/          # Custom React hooks
│   │   │   ├── stores/         # Zustand stores
│   │   │   ├── services/       # API client services
│   │   │   ├── utils/          # Calculation utilities
│   │   │   │   ├── ev.ts       # EV calculations
│   │   │   │   ├── odds.ts     # Odds conversions
│   │   │   │   ├── arb.ts      # Arbitrage detection
│   │   │   │   └── kelly.ts    # Kelly criterion
│   │   │   ├── types/          # TypeScript definitions
│   │   │   └── App.tsx
│   │   └── package.json
│   │
│   └── api/                    # Node.js backend
│       ├── src/
│       │   ├── routes/
│       │   ├── services/
│       │   │   ├── odds-aggregator.ts
│       │   │   ├── kalshi.ts
│       │   │   ├── polymarket.ts
│       │   │   └── ev-calculator.ts
│       │   ├── jobs/           # Background jobs
│       │   └── index.ts
│       └── package.json
│
├── packages/
│   └── shared/                 # Shared types & utils
│       ├── types/
│       └── calculations/
│
├── docs/
│   ├── api-notes.md            # API documentation notes
│   └── calculations.md         # Math explanations
│
├── package.json                # Monorepo root
├── turbo.json                  # Turborepo config
└── README.md
```

Getting Started

Prerequisites

- Node.js 18+
- pnpm (recommended) or npm
- PostgreSQL 14+
- Redis

Installation

```bash
# Clone the repository
git clone https://github.com/riftfern/ny-odds-screener.git
cd ny-odds-screener

# Install dependencies
pnpm install

# Set up environment variables
cp apps/web/.env.example apps/web/.env
cp apps/api/.env.example apps/api/.env

# Add your API keys to .env files
# THE_ODDS_API_KEY=your_key_here
# KALSHI_API_KEY=your_key_here (optional)

# Start development servers
pnpm dev
```

Environment Variables

```env
# apps/api/.env
THE_ODDS_API_KEY=your_key
KALSHI_API_KEY=your_key
KALSHI_API_SECRET=your_secret
DATABASE_URL=postgresql://...
REDIS_URL=redis://...

# apps/web/.env
VITE_API_URL=http://localhost:3001
```

Roadmap

MVP (v0.1) - Target: 4 weeks

☑ Project planning & README

- □ Basic odds comparison for NFL/NBA
- □ +EV calculation engine
- □ Simple UI showing top opportunities

v0.2 - Target: 8 weeks

- □ All NY sportsbooks integrated
- □ Arbitrage detection
- □ Real-time updates
- □ Mobile-responsive design

v0.3 - Target: 12 weeks

- □ Kalshi API integration
- □ Cross-market comparison
- □ Sharp money indicators
- □ Bet tracking

v1.0 - Target: 16 weeks

- □ Full prediction market integration
- □ Liquidity making tools
- □ User authentication
- □ Analytics dashboard
- □ Production deployment

Future Ideas

- □ Telegram bot for alerts (integrate with home server project)
- □ Historical backtesting
- □ Machine learning line prediction
- □ Browser extension for one-click betting
- □ API for other developers

Legal Disclaimer

This tool is for **educational and informational purposes only**.

- You must be 21+ years old and physically located in New York to place bets
- Gambling involves risk of loss - only bet what you can afford to lose

- This project does not encourage gambling or guarantee profits
- Always gamble responsibly: 1-877-8-HOPENY (1-877-846-7369)
- This is a personal project and not affiliated with any sportsbook

---

Contributing

This is a personal portfolio project, but suggestions and feedback are welcome! Feel free to open issues or reach out.

License

MIT License - See LICENSE for details.

Contact

Jack - Self-taught Software Engineer based in NYC

- GitHub: @riftfern
- Portfolio: [Coming Soon]
- Email: [Your email if desired]

Built with ☕ and a passion for beating the vig