

# Sports Arbitrage Platform API - Simple Guide

**Version:** 2.0.0

**Developer:** Shoaib

**Status:** Production Ready

---

## What This API Does

This is a complete sports betting arbitrage platform that helps you find profitable betting opportunities across 4 categories:

1. **Arbitrage** - Guaranteed profit by betting on all outcomes
  2. **Middles** - Potential big wins when result lands in the “middle”
  3. **EV (Expected Value)** - Bets with long-term profit potential
  4. **Lows** - Markets with low bookmaker margins (best value)
- 

## Quick Start (5 Minutes)

*# 1. Install dependencies*

```
pip install -r requirements.txt
```

*# 2. Generate test data*

```
python -m app.services.demo_seed
```

*# 3. Start server*

```
uvicorn app.main:app --reload
```

*# 4. Open browser*

```
http://localhost:8000/docs
```

*# 5. Try first API call*

```
curl -X POST http://localhost:8000/opportunities/compute
```

That's it! You're ready to use the API.

---

## What's New in v2.0

### New Features

- **3 New Detection Types:** Middles, EV, and Lows (previously only had Arbitrage)
- **Unified Endpoint:** One `/opportunities/feed` for all opportunity types
- **Better Filtering:** Filter by category, sport, ROI, and more

- **Auto Expiry:** Opportunities automatically expire after 2 hours
- **Event Linking:** Proper data structure with event IDs
- **Manual Refresh:** `/admin/refresh` to update opportunities

#### New Files

- `app/services/detection_service.py` - All detection algorithms
  - `app/services/opportunity_service.py` - Opportunity management
  - `app/routes/opportunities.py` - Main opportunities API
  - `data/opportunities.csv` - Unified storage for all opportunities
- 

## Complete Feature List

### Core Features (v1.0 + v2.0)

#### Odds Management

- Get live odds from multiple bookmakers
- Filter by sport, market, bookmaker
- WebSocket streaming for real-time updates
- Support for 5 sports: Football, Basketball, Tennis, Ice Hockey, Baseball
- 10+ bookmakers included

#### Opportunity Detection

- **Arbitrage:** Find risk-free profit opportunities (2-way and 3-way markets)
- **Middles:** Find overlapping spread/total lines
- **EV:** Calculate positive expected value bets
- **Lows:** Identify low-hold markets ( $< 2.5\%$  margin)

#### Betting System

- Create betting sessions with unique IDs
- Place bets with automatic stake calculation
- Track all bets by session
- Calculate expected profit for all bet types

#### Admin Tools

- Comprehensive statistics dashboard
- Generate demo/test data
- Manual refresh of opportunities
- System health monitoring
- Reset all data option

## API Features

- Swagger UI documentation at /docs
  - RESTful API design
  - JSON responses
  - CORS enabled
  - Input validation with Pydantic
  - Thread-safe CSV storage
- 

## API Endpoints Reference

### 1 Opportunities (Main Feature)

#### Get All Opportunities

GET /opportunities/feed

*# Filters:*

?category=arbitrage|middle|ev|low

?sport=Football|Basketball|Tennis|etc

?min\_roi=2.0 *# Minimum ROI %*

?min\_profit=1.5 *# Minimum profit %*

?top\_n=20 *# Limit results*

?status=active *# active, expired, removed*

#### Examples:

*# All opportunities*

curl http://localhost:8000/opportunities/feed

*# Only arbitrage*

curl http://localhost:8000/opportunities/feed?category=arbitrage

*# EV bets with 3%+ edge*

curl http://localhost:8000/opportunities/feed?category=ev&min\_roi=3

*# Football middles*

curl http://localhost:8000/opportunities/feed?category=middle&sport=Football

*# Top 10 by ROI*

curl http://localhost:8000/opportunities/feed?top\_n=10

#### Response:

```
{
  "success": true,
  "count": 15,
  "opportunities": [
```

```

    {
      "id": "uuid",
      "type": "arbitrage",
      "sport": "Football",
      "match": "Liverpool vs Man City",
      "roi": 2.4,
      "profit_percentage": 2.4,
      "status": "active"
    }
  ]
}

```

### Get Single Opportunity

GET /opportunities/{id}

### Compute Opportunities

POST /opportunities/compute

Runs all 4 detection algorithms and saves results.

### Response:

```

{
  "success": true,
  "breakdown": {
    "arbitrage": 12,
    "middles": 8,
    "ev": 20,
    "lows": 5,
    "total": 45
  }
}

```

### Get Opportunity Stats

GET /opportunities/stats/summary

### Remove Opportunity

DELETE /opportunities/{id}

## 2 Odds

### Get Live Odds

GET /odds/live?sport=Football&market=Match Winner

### Response:

```
{
  "success": true,
  "count": 1247,
  "odds": [
    {
      "sport": "Football",
      "match": "Liverpool vs Man City",
      "market": "Match Winner",
      "bookmaker": "Bet365",
      "outcome": "Liverpool",
      "odds": 2.10
    }
  ]
}
```

### Get Sports List

GET /odds/sports

### Get Markets List

GET /odds/markets?sport=Football

### WebSocket (Live Streaming)

ws://localhost:8000/odds/ws

---

## 3 Sessions

### Create Session

POST /session/init

Body:

```
{
  "user_name": "John Doe"
}
```

Response:

```
{
  "success": true,
  "session": {
    "session_id": "abc-123",
    "created_at": "2024-01-15T10:30:00"
  }
}
```

```
}  
}
```

#### Get Session

GET /session/{session\_id}

#### Get All Sessions

GET /session/

---

### 4 Bets

#### Place Bet

POST /bets/place

Body:

```
{  
  "session_id": "abc-123",  
  "opportunity_id": "xyz-789",  
  "opportunity_type": "arbitrage",  
  "total_stake": 1000  
}
```

Response:

```
{  
  "success": true,  
  "bets": [  
    {  
      "bookmaker": "Bet365",  
      "outcome": "Liverpool",  
      "stake": 493.00,  
      "odds": 2.10  
    },  
    {  
      "bookmaker": "William Hill",  
      "outcome": "Man City",  
      "stake": 507.00,  
      "odds": 2.05  
    }  
  ],  
  "summary": {  
    "total_stake": 1000.00,  
    "expected_profit": 24.00,  
    "profit_percentage": 2.4  
  }  
}
```

```
}  
}
```

#### Get Bets by Session

GET /bets/session/{session\_id}

#### Get Single Bet

GET /bets/{bet\_id}

#### Get All Bets

GET /bets/

---

## 5 Admin

#### Get Statistics

GET /admin/stats

Response:

```
{  
  "odds": {  
    "total_odds": 1247,  
    "sports": ["Football", "Basketball", ...],  
    "events": 42  
  },  
  "opportunities": {  
    "total_opportunities": 45,  
    "by_type": {  
      "arbitrage": 12,  
      "middle": 8,  
      "ev": 20,  
      "low": 5  
    },  
    "avg_roi": 3.45  
  },  
  "bets": {  
    "total_bets": 25,  
    "total_stake": 25000.00,  
    "total_expected_profit": 625.50  
  }  
}
```

### Refresh Opportunities

POST /admin/refresh

Cleans up expired opportunities and recomputes all.

### Generate Demo Data

POST /admin/seed?num\_events\_per\_sport=5

### System Health

GET /admin/health

### Reset Everything (Dangerous!)

DELETE /admin/reset

---

## Understanding the 4 Opportunity Types

### 1. Arbitrage (Risk-Free)

**What it is:** Bet on all outcomes and guarantee profit.

**Example:**

Liverpool vs Man City  
Bet365: Liverpool @ 2.10  
William Hill: Man City @ 2.05

Bet \$493 on Liverpool → Returns \$1035 if wins  
Bet \$507 on Man City → Returns \$1040 if wins  
Total stake: \$1000  
Guaranteed profit: ~\$37 (3.7%)

**When to use:** Always! Zero risk.

---

### 2. Middles (High Reward)

**What it is:** Bet on overlapping lines. If result lands in middle, both bets win!

**Example:**

Lakers vs Warriors (Spread)  
Bet365: Lakers +3.5 @ 1.91  
Betway: Warriors -2.5 @ 1.91

Bet \$500 on each



If Lakers lose by exactly 3:  
Both bets WIN! → \$1910 return (91% profit!)

If Lakers lose by 2 or less, or 4+:  
One bet wins → \$955 return (-4.5% loss)

**When to use:** When you think result might hit the middle.

---

### 3. EV (Long-Term Profit)

**What it is:** Bookmaker odds better than “fair” odds = long-term profit.

**Example:**

Djokovic vs Alcaraz  
Fair odds (market average): 2.00 (50% chance)  
Betway offers: 2.20

You have 10% edge!  
Over many bets, you'll profit 10% on average.

**When to use:** Professional betting strategy.

---

### 4. Lows (Best Value)

**What it is:** Markets where bookmaker takes small margin.

**Example:**

Nadal vs Federer  
Bet365:  
Nadal: 1.95  
Federer: 2.00

Total implied probability: 101.28%  
Bookmaker margin: Only 1.28% (excellent!)  
Normal margin: 5%+

This is sharp pricing = good value

**When to use:** Finding the best odds/value.

---

## Complete Workflow Example

```
# 1. Generate test data
curl -X POST http://localhost:8000/admin/seed

# 2. Compute opportunities
curl -X POST http://localhost:8000/opportunities/compute

# Response: Found 45 opportunities (12 arb, 8 middle, 20 ev, 5 low)

# 3. View top arbitrage opportunities
curl "http://localhost:8000/opportunities/feed?category=arbitrage&top_n=5"

# Response: Shows 5 best arbitrage opportunities

# 4. Create betting session
curl -X POST http://localhost:8000/session/init -H "Content-Type: application/json" -d '{"us

# Response: session_id = "abc-123"

# 5. Place bet on first opportunity
curl -X POST http://localhost:8000/bets/place \
  -H "Content-Type: application/json" \
  -d '{
    "session_id": "abc-123",
    "opportunity_id": "opp-001",
    "opportunity_type": "arbitrage",
    "total_stake": 1000
  }'

# Response: Shows optimal stakes for each bookmaker

# 6. Check session stats
curl http://localhost:8000/session/abc-123

# Response: Shows total bets, stake, expected profit

# 7. View overall stats
curl http://localhost:8000/admin/stats

# Response: Complete system statistics
```

---

## Data Storage (CSV Files)

All data stored in data/ folder:

### odds.csv

Stores all odds from bookmakers

Columns: event\_id, sport, match, market, bookmaker, outcome, odds, timestamp

### opportunities.csv (NEW in v2.0)

Stores all detected opportunities (arbitrage, middles, ev, lows)

Columns: id, type, event\_id, sport, match, roi, status, timestamp, details

### bets.csv

Stores all placed bets

Columns: bet\_id, session\_id, opportunity\_id, bookmaker, stake, odds, profit

### sessions.csv

Stores betting sessions

Columns: session\_id, created\_at, total\_bets, total\_stake, total\_profit

---

## Configuration

Edit .env file:

```
# Server
HOST=0.0.0.0
PORT=8000

# WebSocket update interval (seconds)
WS_BROADCAST_INTERVAL=5

# Opportunity expiry (hours)
OPPORTUNITY_EXPIRY_HOURS=2
```

---

## Testing

### Run Complete Test Suite

```
python test_all_opportunities.py
```

Tests: - Health check - Demo data generation - All 4 detection types -  
Filtering by category, sport, ROI - Session creation - Bet placement -  
Statistics - Manual refresh

## Quick Manual Test

### # 1. Check health

```
curl http://localhost:8000/health
```

### # 2. Seed data

```
curl -X POST http://localhost:8000/admin/seed
```

### # 3. Compute

```
curl -X POST http://localhost:8000/opportunities/compute
```

### # 4. View results

```
curl http://localhost:8000/opportunities/feed
```

---

## Troubleshooting

Problem	Solution
No opportunities found	Run: <code>POST /admin/seed</code> then <code>POST /opportunities/compute</code>
All opportunities expired	Run: <code>POST /admin/refresh</code>
Server won't start	Check if port 8000 is free: <code>lsof -i :8000</code>
Import errors	Run: <code>pip install -r requirements.txt</code>
File permission errors	Check <code>data/</code> folder permissions

---

## API Documentation

**Interactive Docs:** <http://localhost:8000/docs>

**Alternative Docs:** <http://localhost:8000/redoc>

**OpenAPI JSON:** <http://localhost:8000/openapi.json>

---

## Project Structure

```
sports-arbitrage-platform/  
  app/  
    main.py           # FastAPI app  
    models/schemas.py # Data models  
    routes/  
      opportunities.py # NEW: Main API
```

```

odds.py
bets.py
session.py
admin.py
services/
  detection_service.py # NEW: 4 algorithms
  opportunity_service.py # NEW: Management
  odds_service.py
  bet_service.py
  demo_seed.py
utils/
  csv_io.py # Thread-safe I/O
data/
  odds.csv
  opportunities.csv # NEW: Unified storage
  bets.csv
  sessions.csv
requirements.txt
test_all_opportunities.py # NEW: Test suite
README.md # This file

```

---

## Key Points

### What Makes This Platform Special?

1. **4 Detection Types** - Not just arbitrage, but middles, EV, and lows too
2. **Unified API** - One endpoint for all opportunity types
3. **Auto Expiry** - Opportunities automatically expire after 2 hours
4. **Thread-Safe** - Safe for production use with file locking
5. **Complete** - From odds to bets, everything you need
6. **Well-Documented** - Swagger UI + detailed README

### Best Practices

**Always refresh** before betting: `POST /admin/refresh`

**Filter by ROI** to find best opportunities

**Create sessions** to track your bets

**Check expiry** - opportunities expire after 2 hours

**Use demo data** for testing first

### Important Notes

**Demo Data:** Includes intentional variance to create opportunities

**Thread Safety:** Uses file locking for production safety

**Expiry:** Opportunities auto-expire after 2 hours  
**ROI Sorting:** Results always sorted by ROI (highest first)

---

### Need Help?

1. **API Docs:** <http://localhost:8000/docs>
  2. **Check Health:** GET /admin/health
  3. **View Stats:** GET /admin/stats
  4. **Test Everything:** `python test_all_opportunities.py`
- 

### Version History

#### v2.0.0 (Current)

- Added Middles detection
- Added EV detection
- Added Lows detection
- Unified opportunities endpoint
- Opportunity expiry system
- Enhanced filtering
- Manual refresh endpoint
- Event-based data structure

#### v1.0.0

- Arbitrage detection
  - Odds management
  - Session & bet tracking
  - WebSocket streaming
  - Admin tools
- 

**That's it! Start using the API and find profitable betting opportunities!**

**Version:** 2.0.0

**Status:** Production Ready

**Developer:** Shoaib

**Last Updated:** Saturday EOD