KalshiTerminal

A Bloomberg-style command-line trading terminal for Kalshi prediction markets. Built for traders who prefer keyboard-driven workflows and need to analyze multiple markets simultaneously.

Show Image

Show Image

Overview

KalshiTerminal provides a fast, keyboard-driven interface for viewing Kalshi markets, comparing orderbooks, tracking positions, and executing trades—all from your terminal. Inspired by professional trading terminals like Bloomberg and the Godel Terminal.

```
┌──────────────────────────────────────────────────────────────────────────────
│ KALSHI TERMINAL v0.1.0                    Connected │ 2024-01-15 │
├──────────────────────────────────────────────────────────────────────────────
│                                               │
│  Welcome to KalshiTerminal                            │
│  Type 'help' for available commands or 'markets' to browse          │
│                                               │
│  > _                                           │
│                                               │
└──────────────────────────────────────────────────────────────────────────────
```

Features

Core Functionality

- **Market Discovery** — Browse and search all active Kalshi markets by category, keyword, or ticker
- **Live Orderbooks** — Real-time bid/ask depth with volume visualization

- **Multi-Market View** — Compare up to 4 markets side-by-side in split panels
- **Position Tracking** — Monitor open positions, P&L, and exposure
- **Trade Execution** — Place limit and market orders directly from the terminal
- **Watchlists** — Create custom watchlists for quick access to favorite markets
- **Historical Data** — View price history and volume charts (ASCII-rendered)

Terminal Features

- Vim-style keybindings for navigation
- Command history with up/down arrows
- Tab completion for commands and tickers
- Customizable color themes
- Session persistence

Installation

```bash
# Clone the repository
git clone https://github.com/yourusername/kalshi-terminal.git
cd kalshi-terminal

# Install dependencies
npm install

# Configure API credentials
cp .env.example .env
# Edit .env with your Kalshi API credentials

# Start the terminal
npm start
```

Requirements

- Node.js >= 18.0.0
- Kalshi account with API access enabled
- Terminal with 256-color support recommended

Configuration

Create a `.env` file in the project root:

```env
KALSHI_API_KEY=your_api_key_here
KALSHI_API_SECRET=your_api_secret_here
KALSHI_ENV=prod  # or 'demo' for paper trading
```

Commands

Navigation & Discovery

Command	Alias	Description
`markets`	`m`	List all active markets
`markets <category>`	`m <cat>`	Filter by category (politics, economics, weather, etc.)
`search <query>`	`s`	Search markets by keyword
`quote <ticker>`	`q`	Display detailed quote for a market
`book <ticker>`	`b`	Show full orderbook depth

Multi-Panel Views

Command	Description
`split <ticker>`	Add market to split view (max 4)
`unsplit <ticker>`	Remove market from split view
`clear`	Clear all panels, return to single view
`focus <n>`	Focus panel n (1-4) for commands

Trading

Command	Description
`buy <ticker> <qty> @ <price>`	Place limit buy order
`sell <ticker> <qty> @ <price>`	Place limit sell order
`buy <ticker> <qty> market`	Place market buy order
`cancel <order_id>`	Cancel open order
`cancel all`	Cancel all open orders

Portfolio & Analysis

Command	Alias	Description
`positions`	`pos`	View all open positions
`orders`	`ord`	View all open orders
`history`	`hist`	View trade history
`pnl`	—	Display profit/loss summary
`chart <ticker>`	`c`	ASCII price chart
`volume <ticker>`	`vol`	Volume analysis

Watchlists

Command	Description
`watch add <ticker>`	Add to default watchlist
`watch remove <ticker>`	Remove from watchlist
`watch list`	Display watchlist
`watch create <name>`	Create named watchlist

System

Command	Description
`help`	Show all commands
`help <command>`	Detailed help for command
`theme <name>`	Switch color theme
`config`	Edit configuration
`status`	API connection status
`quit` / `exit`	Exit terminal

Usage Examples

Basic Market Analysis

```
> markets politics

┌─────────────────────────────────────────────────────────────┐
│ POLITICS MARKETS                    24 active │
├────────────────┬───────┬───────┬───────┬──────────┬─────────┤
│ Ticker        │ Bid  │ Ask  │ Volume │ Expires   │ Category │
├────────────────┼───────┼───────┼───────┼──────────┼─────────┤
│ PRES-24-DEM     │ 0.42 │ 0.43 │ 125K │ Nov 5 2024 │ Election │
│ PRES-24-REP     │ 0.56 │ 0.57 │ 118K │ Nov 5 2024 │ Election │
│ SENATE-DEM-50   │ 0.38 │ 0.40 │  42K │ Nov 5 2024 │ Congress │
│ ...          │     │     │     │        │       │
└────────────────┴───────┴───────┴───────┴──────────┴─────────┘

> quote PRES-24-DEM

┌──────────────────────────────────────────────────────────────┐
│ PRES-24-DEM          Democrat wins 2024 Presidential Race │
├──────────────────────────────────────────────────────────────┤
│                                      │
│  YES Price:  $0.43      24h Change:  +0.02 (+4.9%)          │
│  NO Price:   $0.57      24h Volume:  $125,420             │
│                                      │
│  Bid: $0.42 (2,500)      Ask: $0.43 (1,800)             │
│  Spread: $0.01 (2.3%)                          │
│                                          │
│  Open Interest: $2.4M    Expires: Nov 5, 2024 11:59 PM ET   │
│                                          │
└──────────────────────────────────────────────────────────────────────────────┘
```

Orderbook View

```
> book PRES-24-DEM

┌──────────────────────────────────────────────────────────────────────────────┐
│ PRES-24-DEM ORDERBOOK                    Depth: 10 levels │
├──────────────────────────────────────────────────────────────────────────────┤
│                                          │
│  BIDS                    │ ASKS                    │
│  ──────────────────────────────────────────────────────────────────    │
│  ████████████████ 2,500 @ $0.42 │ $0.43 @ 1,800 ████████████   │
│  ██████████████   2,100 @ $0.41 │ $0.44 @ 2,200 ██████████████ │
│  ████████████     1,800 @ $0.40 │ $0.45 @ 3,100 ████████████████│
│  ██████████       1,500 @ $0.39 │ $0.46 @ 2,800 ███████████████ │
│  ████████         1,200 @ $0.38 │ $0.47 @ 1,900 ████████████  │
│                                          │
│  Total Bid Depth: $9,100     │ Total Ask Depth: $11,800    │
│                                          │
└──────────────────────────────────────────────────────────────────────────────┘
```

Split View Comparison

```
> split PRES-24-DEM
> split PRES-24-REP

┌──────────────────────────────────────┬──────────────────────────────────────┐
│ PRES-24-DEM               │ PRES-24-REP           │
│ Democrat wins 2024         │ Republican wins 2024       │
├──────────────────────────────────────┼──────────────────────────────────────┤
│                     │                     │
│  YES: $0.43  NO: $0.57      │  YES: $0.57  NO: $0.43      │
│  Spread: 2.3%              │  Spread: 1.8%              │
│                     │                     │
│  ┌─ 24H CHART ─────────────────┐  │  ┌─ 24H CHART ─────────────────┐  │
│  │   ┌─┐           │ │ │      ┌────────┐      │ │
│  │ ┌──┘ └──┐   ┌──      │ │ │ ───┐ ┌────┘    └───    │ │
│  │─┘     └─────┘        │ │ │   └┘           │ │
│  └─────────────────────────────┘  │  └─────────────────────────────┘  │
│                         │                         │
│  Vol: $125K   OI: $2.4M     │  Vol: $118K   OI: $2.3M      │
│                         │                         │
└─────────────────────────────────┴─────────────────────────────────┘
```

Placing Orders

```
> buy PRES-24-DEM 100 @ 0.41

┌──────────────────────────────────────────────────────────────────────┐
│ ORDER CONFIRMATION                                   │
├──────────────────────────────────────────────────────────────────────┤
│                                                      │
│   Action:   BUY YES                                  │
│   Market:   PRES-24-DEM (Democrat wins 2024 Presidential Race)       │
│   Quantity: 100 contracts                            │
│   Price:    $0.41 per contract                       │
│   Total:    $41.00                                   │
│                                                      │
│   Max Profit: $59.00 (143.9%)    Max Loss: $41.00 (100%)         │
│                                                      │
│   Confirm order? (y/n): _                            │
│                                                      │
└──────────────────────────────────────────────────────────────────────┘
```

Keyboard Shortcuts

Key	Action
Ctrl+C	Cancel current input / Exit
Ctrl+L	Clear screen
Tab	Autocomplete command/ticker
↑ / ↓	Navigate command history
Ctrl+R	Reverse search history
Esc	Close current panel/modal
1-4	Quick focus panels (in split view)

Key	Action
?	Quick help overlay

Project Structure

```
kalshi-terminal/
├── src/
│   ├── index.js          # Entry point
│   ├── terminal/
│   │   ├── Terminal.js   # Main terminal class
│   │   ├── Input.js      # Input handling
│   │   ├── Renderer.js   # Screen rendering
│   │   └── themes/       # Color themes
│   ├── commands/
│   │   ├── index.js      # Command router
│   │   ├── markets.js    # Market commands
│   │   ├── trading.js    # Trading commands
│   │   ├── portfolio.js  # Portfolio commands
│   │   └── system.js     # System commands
│   ├── api/
│   │   ├── KalshiClient.js   # API wrapper
│   │   ├── WebSocket.js      # Real-time data
│   │   └── RateLimiter.js    # Rate limiting
│   ├── views/
│   │   ├── Orderbook.js  # Orderbook display
│   │   ├── Chart.js      # ASCII charts
│   │   ├── Quote.js      # Quote display
│   │   └── SplitView.js  # Multi-panel layout
│   └── utils/
│       ├── format.js     # Number/date formatting
│       └── storage.js    # Local persistence
├── tests/
├── .env.example
├── package.json
└── README.md
```

Tech Stack

- **Runtime**: Node.js
- **Terminal UI**: blessed / ink

- **API Client**: Custom Kalshi REST + WebSocket client
- **Charts**: asciichart
- **Storage**: SQLite for local data persistence

API Rate Limits

KalshiTerminal respects Kalshi's API rate limits:

- REST API: 100 requests/minute
- WebSocket: Real-time updates (no polling needed)
- The terminal implements automatic rate limiting and request queuing

Roadmap

□ Core terminal UI framework
□ Market listing and search
□ Real-time orderbook display
□ Split-panel comparison view
□ Order placement and management
□ Position tracking and P&L
□ ASCII price/volume charts
□ Watchlists with alerts
□ Export data to CSV
□ Paper trading mode
□ Custom scripting/macros
□ Mobile-friendly web version

Contributing

Contributions are welcome! Please read the contributing guidelines before submitting PRs.

1. Fork the repository
2. Create a feature branch (`git checkout -b feature/amazing-feature`)
3. Commit changes (`git commit -m 'Add amazing feature'`)
4. Push to branch (`git push origin feature/amazing-feature`)
5. Open a Pull Request

Disclaimer

This software is for educational and informational purposes. Trading on prediction markets involves risk. Past performance does not guarantee future results. Always trade responsibly and never risk more than you can afford to lose.

This project is not affiliated with, endorsed by, or connected to Kalshi, Inc.

License

Built with ☕ by [Your Name]