{

"appName": "TradePro",

"version": "1.0.0",

"description": "Comprehensive day-trading and currency trading platform with advanced data collection, self-check systems, and GitHub integration",

"architecture": {

"frontend": {

"framework": "React",

"platforms": ["Web", "Desktop", "Mobile"],

"desktopTechnologies": ["Electron"],

"mobileTechnologies": ["React Native"],

"designSystem": {

"goldenRatio": {

"enabled": true,

"baseSpacing": 8,

"baseRadius": 4,

"baseFontSize": 16,

"proportions": {

"primary": 0.618,

"secondary": 0.382

}

},

"adaptiveTheming": {

"darkMode": true,

"lightMode": true,

"colorBlindMode": true,

"highContrastMode": true,

"tradingFocusedThemes": ["Day", "Night", "Power Trader"]

},

"components": {

"TradingCards": {

"variants": ["asset", "watchlist", "portfolio", "analysis"],

"aspectRatios": {

"primary": 1.618,

"compact": "auto"

}

},

"ChartContainer": {

"customizable": true,

"layouts": ["single", "dual", "quad", "custom"],

"savedLayouts": true

}

},

"colorCoding": {

"up": "#25a764",

"down": "#e53935",

"neutral": "#757575",

"alert": "#f9a825",

"success": "#43a047",

"error": "#d32f2f"

}

},

"userExperience": {

"navigationPattern": "Sidebar with Tabs",

"accessibilityFeatures": {

"screenReaderSupport": true,

"keyboardNavigation": true,

"voiceCommands": true,

"gestureControl": true

},

"onboarding": {

"interactive": true,

"skillLevelAssessment": true,

"customizablePath": true,

"featureHighlights": ["charts", "watchlists", "orders", "analysis"]

}

},

"components": {

"navigation": {

"primary": [

{

"name": "Dashboard",

"icon": "dashboard",

"prominence": "high"

},

{

"name": "Charts",

"icon": "trending-up",

"prominence": "high"

},

{

"name": "Watchlists",

"icon": "list",

"prominence": "high"

},

{

"name": "Orders",

"icon": "shopping-cart",

"prominence": "high"

},

{

"name": "Portfolio",

"icon": "account-balance",

"prominence": "high"

},

{

"name": "Analysis",

"icon": "assessment",

"prominence": "medium"

},

{

"name": "News",

"icon": "article",

"prominence": "medium"

},

{

"name": "Education",

"icon": "school",

"prominence": "medium"

},

{

"name": "Settings",

"icon": "settings",

"prominence": "low"

}

],

"contextual": {

"quickActionBar": {

"alwaysAccessible": true,

"position": "top",

"customizable": true

}

}

},

"screens": [

{

"name": "Dashboard",

"description": "Customizable dashboard with key market insights and portfolio overview",

"components": [

{

"name": "MarketOverview",

"props": {

"showIndices": true,

"showCommodities": true,

"showForex": true,

"showCrypto": true,

"customizable": true

}

},

{

"name": "PortfolioSummary",

"props": {

"showPnL": true,

"showAllocation": true,

"showPerformance": true,

"timeFrames": ["daily", "weekly", "monthly", "yearly"]

}

},

{

"name": "WatchlistSummary",

"props": {

"showPercentChange": true,

"showVolume": true,

"showAlerts": true

}

},

{

"name": "NewsWidget",

"props": {

"filterByWatchlist": true,

"filterByPortfolio": true,

"customKeywords": true

}

},

{

"name": "TradingJournal",

"props": {

"quickEntry": true,

"tradeStatistics": true,

"githubSync": true

}

}

]

},

{

"name": "ChartScreen",

"description": "Advanced charting with multiple layouts and technical indicators",

"components": [

{

"name": "AdvancedChartComponent",

"props": {

"timeframes": ["1min", "5min", "15min", "30min", "1H", "4H", "1D", "1W", "1M"],

"chartTypes": ["candlestick", "line", "bar", "area", "heikin-ashi", "renko"],

"drawingTools": true,

"indicators": {

"momentum": ["RSI", "MACD", "Stochastic", "CCI"],

"trend": ["Moving Averages", "Bollinger Bands", "Ichimoku", "Parabolic SAR"],

"volatility": ["ATR", "Standard Deviation", "Bollinger Bands Width"],

"volume": ["Volume", "OBV", "Money Flow Index"],

"custom": true

},

"layoutOptions": {

"multiChart": true,

"overlays": true,

"customizableGrids": true,

"savedLayouts": true

}

},

"methods": [

"addIndicator(type, params)",

"saveLayout(name)",

"loadLayout(name)",

"exportChart(format)",

"setTimeframe(interval)",

"addDrawing(type, params)",

"toggleVisibility(element)",

"compareSymbols(symbols)"

]

},

{

"name": "TickerSearch",

"props": {

"multiMarket": true,

"recentSearches": true,

"popularAssets": true

}

},

{

"name": "TradingPanel",

"props": {

"orderTypes": ["market", "limit", "stop", "stop-limit", "trailing-stop"],

"positions": {

"size": true,

"risk": true,

"potentialReward": true,

"marginRequirement": true

},

"quickOrder": true,

"orderPresets": true

}

}

]

},

{

"name": "WatchlistScreen",

"description": "Customizable watchlists with detailed market data",

"components": [

{

"name": "WatchlistManager",

"props": {

"multipleWatchlists": true,

"filterOptions": true,

"sortOptions": true,

"customColumns": true,

"exportToGithub": true

}

},

{

"name": "MarketScanner",

"props": {

"predefinedScans": true,

"customScans": true,

"technicalFilters": true,

"fundamentalFilters": true,

"saveScans": true

}

},

{

"name": "AlertManager",

"props": {

"priceAlerts": true,

"percentageAlerts": true,

"volumeAlerts": true,

"indicatorAlerts": true,

"patternAlerts": true,

"multiDeviceSync": true

}

}

]

},

{

"name": "OrderScreen",

"description": "Complete order management system with advanced order types",

"components": [

{

"name": "OrderEntry",

"props": {

"orderTypes": [

"market", "limit", "stop", "stop-limit",

"trailing-stop", "OCO", "bracket", "iceberg"

],

"timeInForce": ["day", "GTC", "IOC", "FOK"],

"routingOptions": true,

"previewCommissions": true,

"riskAnalysis": true

}

},

{

"name": "OrderBook",

"props": {

"depth": true,

"histogram": true,

"liquidityVisualization": true

}

},

{

"name": "OrderHistory",

"props": {

"filtering": true,

"sorting": true,

"statistics": true,

"exportOptions": ["CSV", "PDF", "GitHub Gist"]

}

},

{

"name": "RiskCalculator",

"props": {

"positionSizing": true,

"stopLossSuggestions": true,

"riskRewardVisualizer": true,

"marginRequirements": true

}

}

]

},

{

"name": "PortfolioScreen",

"description": "Comprehensive portfolio management and analysis",

"components": [

{

"name": "PortfolioOverview",

"props": {

"assetAllocation": true,

"sectorExposure": true,

"riskMetrics": true,

"performanceMetrics": true

}

},

{

"name": "PositionManager",

"props": {

"openPositions": true,

"closedPositions": true,

"pnlCalculation": true,

"costBasis": true,

"taxLotAnalysis": true

}

},

{

"name": "PerformanceAnalytics",

"props": {

"historicalPerformance": true,

"benchmarkComparison": true,

"drawdownAnalysis": true,

"volatilityMetrics": true

}

}

]

},

{

"name": "AnalysisScreen",

"description": "Technical and fundamental analysis tools",

"components": [

{

"name": "TechnicalAnalysis",

"props": {

"patternRecognition": true,

"signalDetection": true,

"customIndicators": true,

"backtesting": true,

"aiSuggestions": true

}

},

{

"name": "FundamentalAnalysis",

"props": {

"financialStatements": true,

"ratioAnalysis": true,

"earningsHistory": true,

"analystRecommendations": true,

"companyNews": true

}

},

{

"name": "MacroeconomicDashboard",

"props": {

"economicCalendar": true,

"interestRates": true,

"inflationData": true,

"marketSentiment": true,

"correlationMatrices": true

}

},

{

"name": "AIMarketInsights",

"props": {

"trendAnalysis": true,

"sentimentAnalysis": true,

"anomalyDetection": true,

"predictiveModels": true,

"explainabilityFeatures": true

}

}

]

},

{

"name": "NewsScreen",

"description": "Comprehensive financial news and updates",

"components": [

{

"name": "NewsFeed",

"props": {

"customFilters": true,

"relevanceRanking": true,

"sentimentAnalysis": true,

"watchlistSync": true,

"portfolioSync": true

}

},

{

"name": "EconomicCalendar",

"props": {

"upcomingEvents": true,

"eventImpact": true,

"historicalResults": true,

"notifications": true

}

},

{

"name": "SocialSentiment",

"props": {

"redditAnalysis": true,

"twitterAnalysis": true,

"stocktwitsIntegration": true,

"sentimentTrends": true

}

},

{

"name": "InsiderActivity",

"props": {

"recentTransactions": true,

"historicalPatterns": true,

"alertSystem": true

}

}

]

},

{

"name": "EducationScreen",

"description": "Comprehensive learning resources and trading academy",

"components": [

{

"name": "LearningCenter",

"props": {

"strategyGuides": true,

"videoTutorials": true,

"interactiveCoursework": true,

"skillLevelFiltering": true

}

},

{

"name": "PaperTradingSimulator",

"props": {

"realMarketData": true,

"strategyTesting": true,

"performanceTracking": true,

"skillDevelopmentPrompts": true

}

},

{

"name": "CommunityHub",

"props": {

"discussionForums": true,

"strategySharing": true,

"mentorships": true,

"githubIntegration": true

}

},

{

"name": "TradingJournal",

"props": {

"templateOptions": true,

"performanceAnalytics": true,

"githubExport": true,

"aiInsights": true

}

}

]

}

],

"shared": [

{

"name": "DataSyncService",

"description": "Cross-platform data synchronization",

"features": {

"realTimeSync": true,

"offlineSupport": true,

"conflictResolution": true,

"encryptedSync": true,

"githubBackup": true

},

"methods": [

"syncData(dataType, direction)",

"resolveConflicts(conflictData)",

"exportToGithub(dataType, repository, path)",

"importFromGithub(repository, path)",

"getLastSyncStatus()"

]

},

{

"name": "NotificationService",

"description": "Multi-channel trading alerts and notifications",

"features": {

"priceAlerts": true,

"orderExecutionAlerts": true,

"newsAlerts": true,

"technicalSignals": true,

"multiDeviceDelivery": true,

"priorityLevels": true

},

"methods": [

"createAlert(type, conditions, channels)",

"manageAlerts(actions)",

"setAlertPreferences(preferences)",

"testAlert(alertConfig)",

"getNotificationHistory()"

]

},

{

"name": "MarketDataService",

"description": "Real-time and historical market data management",

"features": {

"realTimeDataStreaming": true,

"historicalDataCache": true,

"multiSourceIntegration": true,

"dataQualityMonitoring": true,

"customDataFeeds": true

},

"methods": [

"subscribeToRealtime(symbols, fields)",

"getHistoricalData(symbol, timeframe, start, end)",

"addCustomDataFeed(config)",

"monitorDataQuality(sources)",

"exportMarketData(format, selections)"

]

}

]

},

"state": {

"redux": {

"slices": [

{

"name": "marketData",

"actions": [

"FETCH\_REALTIME\_DATA",

"FETCH\_HISTORICAL\_DATA",

"UPDATE\_PRICE",

"UPDATE\_ORDERBOOK",

"UPDATE\_MARKET\_DEPTH",

"SET\_TIMEFRAME",

"SET\_CHART\_TYPE"

]

},

{

"name": "watchlists",

"actions": [

"CREATE\_WATCHLIST",

"DELETE\_WATCHLIST",

"RENAME\_WATCHLIST",

"ADD\_SYMBOL",

"REMOVE\_SYMBOL",

"REORDER\_WATCHLIST",

"EXPORT\_TO\_GITHUB"

]

},

{

"name": "portfolio",

"actions": [

"UPDATE\_POSITIONS",

"UPDATE\_BALANCE",

"UPDATE\_MARGIN",

"CALCULATE\_PNL",

"SET\_CURRENCY",

"EXPORT\_PORTFOLIO"

]

},

{

"name": "orders",

"actions": [

"PLACE\_ORDER",

"CANCEL\_ORDER",

"MODIFY\_ORDER",

"UPDATE\_ORDER\_STATUS",

"GET\_ORDER\_HISTORY",

"CLEAR\_ORDERS"

]

},

{

"name": "user",

"actions": [

"LOGIN",

"LOGOUT",

"UPDATE\_PREFERENCES",

"CONNECT\_BROKER",

"UPDATE\_PROFILE",

"CONNECT\_GITHUB"

]

},

{

"name": "alerts",

"actions": [

"CREATE\_ALERT",

"DELETE\_ALERT",

"TRIGGER\_ALERT",

"DISMISS\_ALERT",

"GET\_ALERT\_HISTORY"

]

},

{

"name": "analysis",

"actions": [

"RUN\_TECHNICAL\_ANALYSIS",

"RUN\_FUNDAMENTAL\_ANALYSIS",

"SAVE\_ANALYSIS",

"LOAD\_ANALYSIS",

"EXPORT\_ANALYSIS",

"SHARE\_TO\_GITHUB"

]

}

],

"middleware": [

"thunkMiddleware",

"socketMiddleware",

"persistMiddleware",

"analyticsMiddleware",

"githubIntegrationMiddleware"

],

"enhancers": [

"devToolsEnhancer",

"persistEnhancer",

"memoizationEnhancer"

]

}

}

},

"backend": {

"server": {

"technology": "Node.js",

"framework": "Express",

"scaling": "Kubernetes"

},

"githubIntegration": {

"repositories": {

"frontend": {

"name": "tradepro-frontend",

"cicd": {

"provider": "GitHub Actions",

"workflowFiles": [

"build-test.yml",

"deploy-staging.yml",

"deploy-production.yml"

]

},

"branchStrategy": {

"main": "production-ready code",

"staging": "pre-production testing",

"develop": "active development",

"feature/\*": "new features"

},

"issueTemplates": [

"bug-report.md",

"feature-request.md",

"ui-improvement.md"

],

"automatedTesting": {

"unit": "Jest",

"integration": "Cypress",

"e2e": "Playwright"

}

},

"backend": {

"name": "tradepro-backend",

"cicd": {

"provider": "GitHub Actions",

"workflowFiles": [

"test-lint.yml",

"build-deploy.yml",

"database-migration.yml"

]

},

"branchStrategy": {

"main": "production-ready code",

"staging": "pre-production testing",

"develop": "active development",

"feature/\*": "new features"

},

"issueTemplates": [

"bug-report.md",

"api-change.md",

"performance-issue.md"

],

"automatedTesting": {

"unit": "Mocha",

"integration": "Supertest",

"load": "k6"

}

},

"dataRepository": {

"name": "tradepro-data",

"description": "Repository for trading strategies, backtesting results, and market analysis",

"access": {

"public": false,

"collaborators": true

},

"structure": {

"strategies": {

"description": "Trading strategies with metadata",

"format": "JSON"

},

"backtests": {

"description": "Backtest results with performance metrics",

"format": "JSON and CSV"

},

"marketAnalysis": {

"description": "Market analysis reports and models",

"format": "Markdown, Jupyter Notebooks"

},

"tradingJournals": {

"description": "User trading journals with performance metrics",

"format": "Markdown, CSV"

}

}

},

"shared": {

"name": "tradepro-shared",

"description": "Shared types, constants, and utilities",

"packages": [

"api-types",

"constants",

"validation-schemas",

"market-calculations"

]

}

},

"projectManagement": {

"tool": "GitHub Projects",

"boards": [

"Development Roadmap",

"Bug Tracking",

"Feature Requests",

"Release Planning"

],

"automations": [

"Auto-assign new issues",

"Move issues on PR creation",

"Close issues on PR merge"

]

},

"documentation": {

"tool": "GitHub Wiki",

"sections": [

"API Documentation",

"Development Setup",

"Architecture Overview",

"Trading Strategies Guide",

"Data Collection Documentation"

],

"automatedUpdates": {

"apiDocsGeneration": "swagger-to-github-wiki",

"componentDocumentation": "storybook-to-github-pages"

}

},

"userContent": {

"tradingStrategies": {

"sharing": true,

"versioning": true,

"permissions": {

"private": true,

"public": true,

"team": true

}

},

"tradingJournals": {

"templates": true,

"statistics": true,

"exportOptions": ["PDF", "CSV", "GitHub Markdown"]

},

"backtestResults": {

"versioning": true,

"comparisons": true,

"optimization": true

}

}

},

"database": {

"primary": {

"type": "MongoDB",

"collections": [

{

"name": "users",

"schema": {

"userId": "String",

"email": "String",

"passwordHash": "String",

"name": "String",

"brokerConnections": "Array",

"preferences": "Object",

"githubIntegration": "Object",

"createdAt": "Date",

"lastLogin": "Date"

},

"indexes": ["userId", "email"]

},

{

"name": "watchlists",

"schema": {

"watchlistId": "String",

"userId": "String",

"name": "String",

"symbols": "Array",

"createdAt": "Date",

"updatedAt": "Date",

"settings": "Object"

},

"indexes": ["watchlistId", "userId"]

},

{

"name": "portfolios",

"schema": {

"portfolioId": "String",

"userId": "String",

"brokerId": "String",

"name": "String",

"balance": "Number",

"currency": "String",

"positions": "Array",

"history": "Array",

"createdAt": "Date",

"updatedAt": "Date"

},

"indexes": ["portfolioId", "userId", "brokerId"]

},

{

"name": "orders",

"schema": {

"orderId": "String",

"userId": "String",

"portfolioId": "String",

"brokerId": "String",

"symbol": "String",

"type": "String",

"side": "String",

"quantity": "Number",

"price": "Number",

"stopPrice": "Number",

"timeInForce": "String",

"status": "String",

"createdAt": "Date",

"executedAt": "Date",

"cancelledAt": "Date",

"metadata": "Object"

},

"indexes": ["orderId", "userId", "symbol", "status"]

},

{

"name": "strategies",

"schema": {

"strategyId": "String",

"userId": "String",

"name": "String",

"description": "String",

"rules": "Array",

"indicators": "Array",

"createdAt": "Date",

"updatedAt": "Date",

"backtestResults": "Array",

"isPublic": "Boolean",

"githubUrl": "String"

},

"indexes": ["strategyId", "userId", "isPublic"]

},

{

"name": "alerts",

"schema": {

"alertId": "String",

"userId": "String",

"symbol": "String",

"type": "String",

"conditions": "Object",

"status": "String",

"createdAt": "Date",

"triggeredAt": "Date",

"notifications": "Array"

},

"indexes": ["alertId", "userId", "symbol", "status"]

},

{

"name": "backtests",

"schema": {

"backtestId": "String",

"userId": "String",

"strategyId": "String",

"symbol": "String",

"timeframe": "String",

"startDate": "Date",

"endDate": "Date",

"parameters": "Object",

"results": "Object",

"createdAt": "Date",

"githubUrl": "String"

},

"indexes": ["backtestId", "userId", "strategyId"]

},

{

"name": "marketData",

"schema": {

"symbol": "String",

"exchange": "String",

"type": "String",

"timeframe": "String",

"date": "Date",

"open": "Number",

"high": "Number",

"low": "Number",

"close": "Number",

"volume": "Number",

"adjustedClose": "Number",

"additionalData": "Object"

},

"indexes": ["symbol", "timeframe", "date"],

"timeSeriesConfig": {

"enabled": true,

"granularity": "hours"

}

},

{

"name": "tradingJournals",

"schema": {

"entryId": "String",

"userId": "String",

"tradeIds": "Array",

"entryDate": "Date",

"symbols": "Array",

"strategy": "String",

"notes": "String",

"attachments": "Array",

"performance": "Object",

"lessons": "String",

"tags": "Array",

"githubUrl": "String"

},

"indexes": ["entryId", "userId", "entryDate", "symbols"]

}

]

},

"timeSeries": {

"type": "TimescaleDB",

"tables": [

{

"name": "price\_ticks",

"schema": {

"symbol": "String",

"exchange": "String",

"timestamp": "Timestamp",

"price": "Number",

"volume": "Number",

"side": "String"

},

"indexes": ["symbol", "timestamp"],

"retention": "30 days"

},

{

"name": "orderbook\_snapshots",

"schema": {

"symbol": "String",

"exchange": "String",

"timestamp": "Timestamp",

"bids": "Array",

"asks": "Array",

"spread": "Number"

},

"indexes": ["symbol", "timestamp"],

"retention": "7 days"

}

]

},

"cache": {

"type": "Redis",

"keys": [

{

"pattern": "market:realtime:\*",

"ttl": 60

},

{

"pattern": "user:session:\*",

"ttl": 3600

},

{

"pattern": "orderbook:\*",

"ttl": 30

},

{

"pattern": "analytics:\*",

"ttl": 600

}

]

}

},

"apis": {

"rest": {

"documentation": {

"tool": "Swagger/OpenAPI",

"autoGeneration": true,

"publishToGitHub": true

},

"design": {

"versionedEndpoints": true,

"consistentNaming": true,

"descriptiveErrorMessages": true,

"rateLimit": {

"enabled": true,

"dynamic": true

}

},

"endpoints": [

{

"path": "/api/auth",

"methods": [

{

"type": "POST",

"route": "/register",

"description": "Register new user account"

},

{

"type": "POST",

"route": "/login",

"description": "Authenticate user"

},

{

"type": "POST",

"route": "/logout",

"description": "End user session"

},

{

"type": "POST",

"route": "/refresh-token",

"description": "Refresh authentication token"

},

{

"type": "POST",

"route": "/github-login",

"description": "Login using GitHub OAuth"

},

{

"type": "POST",

"route": "/broker-auth",

"description": "Connect to brokerage account"

}

]

},

{

"path": "/api/market",

"methods": [

{

"type": "GET",

"route": "/quotes/:symbol",

"description": "Get real-time quote for symbol"

},

{

"type": "GET",

"route": "/history/:symbol",

"description": "Get historical data for symbol"

},

{

"type": "GET",

"route": "/orderbook/:symbol",

"description": "Get order book for symbol"

},

{

"type": "GET",

"route": "/options-chain/:symbol",

"description": "Get options chain for symbol"

},

{

"type": "GET",

"route": "/forex/rates",

"description": "Get forex exchange rates"

},

{

"type": "GET",

"route": "/crypto/rates",

"description": "Get cryptocurrency rates"

}

]

},

{

"path": "/api/watchlists",

"methods": [

{

"type": "GET",

"route": "/",

"description": "Get all user watchlists"

},

{

"type": "POST",

"route": "/",

"description": "Create new watchlist"

},

{

"type": "GET",

"route": "/:watchlistId",

"description": "Get specific watchlist"

},

{

"type": "PUT",

"route": "/:watchlistId",

"description": "Update watchlist"

},

{

"type": "DELETE",

"route": "/:watchlistId",

"description": "Delete watchlist"

},

{

"type": "POST",

"route": "/:watchlistId/symbols",

"description": "Add symbols to watchlist"

},

{

"type": "DELETE",

"route": "/:watchlistId/symbols/:symbol",

"description": "Remove symbol from watchlist"

},

{

"type": "POST",

"route": "/:watchlistId/github-export",

"description": "Export watchlist to GitHub repository"

}

]

},

{

"path": "/api/portfolio",

"methods": [

{

"type": "GET",

"route": "/",

"description": "Get user portfolio summary"

},

{

"type": "GET",

"route": "/positions",

"description": "Get current positions"

},

{

"type": "GET",

"route": "/history",

"description": "Get portfolio history"

},

{

"type": "GET",

"route": "/performance",

"description": "Get portfolio performance metrics"

},

{

"type": "POST",

"route": "/github-export",

"description": "Export portfolio data to GitHub repository"

}

]

},

{

"path": "/api/orders",

"methods": [

{

"type": "POST",

"route": "/",

"description": "Place new order"

},

{

"type": "GET",

"route": "/active",

"description": "Get active orders"

},

{

"type": "GET",

"route": "/history",

"description": "Get order history"

},

{

"type": "PUT",

"route": "/:orderId",

"description": "Modify existing order"

},

{

"type": "DELETE",

"route": "/:orderId",

"description": "Cancel order"

},

{

"type": "GET",

"route": "/:orderId/status",

"description": "Check order status"

}

]

},

{

"path": "/api/strategies",

"methods": [

{

"type": "GET",

"route": "/",

"description": "Get user's trading strategies"

},

{

"type": "POST",

"route": "/",

"description": "Create new strategy"

},

{

"type": "GET",

"route": "/:strategyId",

"description": "Get specific strategy"

},

{

"type": "PUT",

"route": "/:strategyId",

"description": "Update strategy"

},

{

"type": "DELETE",

"route": "/:strategyId",

"description": "Delete strategy"

},

{

"type": "POST",

"route": "/:strategyId/backtest",

"description": "Run backtest on strategy"

},

{

"type": "POST",

"route": "/:strategyId/github-export",

"description": "Export strategy to GitHub repository"

},

{

"type": "GET",

"route": "/public",

"description": "Get public trading strategies"

},

{

"type": "POST",

"route": "/github-import",

"description": "Import strategy from GitHub repository"

}

]

},

{

"path": "/api/alerts",

"methods": [

{

"type": "GET",

"route": "/",

"description": "Get user alerts"

},

{

"type": "POST",

"route": "/",

"description": "Create new alert"

},

{

"type": "GET",

"route": "/:alertId",

"description": "Get specific alert"

},

{

"type": "PUT",

"route": "/:alertId",

"description": "Update alert"

},

{

"type": "DELETE",

"route": "/:alertId",

"description": "Delete alert"

},

{

"type": "GET",

"route": "/history",

"description": "Get alert history"

}

]

},

{

"path": "/api/analytics",

"methods": [

{

"type": "GET",

"route": "/technical/:symbol",

"description": "Get technical analysis for symbol"

},

{

"type": "GET",

"route": "/fundamental/:symbol",

"description": "Get fundamental analysis for symbol"

},

{

"type": "GET",

"route": "/screener",

"description": "Run market screener with filters"

},

{

"type": "POST",

"route": "/custom",

"description": "Run custom analytics query"

},

{

"type": "GET",

"route": "/sentiment/:symbol",

"description": "Get sentiment analysis for symbol"

},

{

"type": "POST",

"route": "/github-export",

"description": "Export analysis results to GitHub repository"

}

]

},

{

"path": "/api/journal",

"methods": [

{

"type": "GET",

"route": "/entries",

"description": "Get trading journal entries"

},

{

"type": "POST",

"route": "/entries",

"description": "Create new journal entry"

},

{

"type": "GET",

"route": "/entries/:entryId",

"description": "Get specific journal entry"

},

{

"type": "PUT",

"route": "/entries/:entryId",

"description": "Update journal entry"

},

{

"type": "DELETE",

"route": "/entries/:entryId",

"description": "Delete journal entry"

},

{

"type": "GET",

"route": "/statistics",

"description": "Get trading statistics from journal"

},

{

"type": "POST",

"route": "/github-export",

"description": "Export journal to GitHub repository"

},

{

"type": "GET",

"route": "/templates",

"description": "Get journal entry templates"

}

]

},

{

"path": "/api/github",

"methods": [

{

"type": "POST",

"route": "/connect",

"description": "Connect GitHub account"

},

{

"type": "GET",

"route": "/status",

"description": "Check GitHub connection status"

},

{

"type": "GET",

"route": "/repositories",

"description": "List accessible GitHub repositories"

},

{

"type": "POST",

"route": "/export",

"description": "Export data to GitHub repository"

},

{

"type": "POST",

"route": "/import",

"description": "Import data from GitHub repository"

},

{

"type": "GET",

"route": "/gists",

"description": "List trading-related GitHub Gists"

}

]

}

]

},

"websocket": {

"events": [

{

"name": "market\_data",

"description": "Real-time market data updates",

"data": {

"symbol": "String",

"price": "Number",

"change": "Number",

"volume": "Number",

"timestamp": "Number"

}

},

{

"name": "orderbook\_update",

"description": "Order book updates",

"data": {

"symbol": "String",

"bids": "Array",

"asks": "Array",

"timestamp": "Number"

}

},

{

"name": "order\_update",

"description": "Order status updates",

"data": {

"orderId": "String",

"status": "String",

"filledQuantity": "Number",

"remainingQuantity": "Number",

"timestamp": "Number"

}

},

{

"name": "trade\_execution",

"description": "Trade execution notifications",

"data": {

"tradeId": "String",

"orderId": "String",

"symbol": "String",

"quantity": "Number",

"price": "Number",

"side": "String",

"timestamp": "Number"

}

},

{

"name": "portfolio\_update",

"description": "Portfolio value and position updates",

"data": {

"balance": "Number",

"equity": "Number",

"positions": "Array",

"pnl": "Object",

"timestamp": "Number"

}

},

{

"name": "alert\_triggered",

"description": "Alert triggering notifications",

"data": {

"alertId": "String",

"symbol": "String",

"condition": "String",

"value": "Number",

"timestamp": "Number"

}

},

{

"name": "news\_update",

"description": "Breaking news updates",

"data": {

"symbols": "Array",

"headline": "String",

"source": "String",

"url": "String",

"sentiment": "String",

"timestamp": "Number"

}

},

{

"name": "github\_sync",

"description": "GitHub synchronization events",

"data": {

"type": "String",

"status": "String",

"repositoryName": "String",

"path": "String",

"timestamp": "Number"

}

}

]

}

},

"services": [

{

"name": "DataCollectionService",

"description": "Comprehensive market data collection and processing",

"components": [

{

"name": "MultiSourceAggregator",

"description": "Aggregates data from multiple providers for validation and completeness",

"features": {

"dataSourceRedundancy": true,

"discrepancyDetection": true,

"qualityScoring": true,

"sourceFailoverHandling": true

}

},

{

"name": "HistoricalDataProcessor",

"description": "Processes and normalizes historical market data",

"features": {

"dateRangeNormalization": true,

"adjustmentFactorsHandling": true,

"timeframeConversion": true,

"dataGapDetection": true,

"dataNormalization": true

}

},

{

"name": "RealTimeProcessor",

"description": "Handles real-time data streams",

"features": {

"lowLatencyProcessing": true,

"dataValidation": true,

"streamMultiplexing": true,

"quoteBroadcasting": true,

"dataBuffering": true

}

},

{

"name": "AltDataCollector",

"description": "Collects alternative data sources",

"features": {

"socialMediaSentiment": true,

"newsAnalysis": true,

"economicIndicators": true,

"insiderTrading": true,

"searchTrendAnalysis": true

}

}

],

"methods": [

"fetchHistoricalData(symbol, timeframe, startDate, endDate, options)",

"subscribeToRealTime(symbols, dataTypes, options)",

"aggregateMultiSourceData(symbol, dataType, sources)",

"validateDataQuality(dataSegment, qualityCriteria)",

"detectAnomalies(dataStream, detectionParams)",

"storeProcessedData(dataType, data, storageOptions)",

"exportToGitHub(dataType, data, repository, path)"

]

},

{

"name": "TradingService",

"description": "Manages trading operations across multiple brokers",

"components": [

{

"name": "OrderRouter",

"description": "Routes orders to appropriate brokers",

"features": {

"smartRouting": true,

"costOptimization": true,

"executionQualityAnalysis": true,

"splitOrderCapability": true

}

},

{

"name": "RiskManager",

"description": "Manages trading risk parameters",

"features": {

"positionSizeCalculation": true,

"riskRewardAnalysis": true,

"exposureMonitoring": true,

"drawdownProtection": true,

"volatilityAdjustment": true

}

},

{

"name": "ExecutionAnalyzer",

"description": "Analyzes execution quality",

"features": {

"slippageCalculation": true,

"executionSpeedAnalysis": true,

"costAnalysis": true,

"fillQualityMetrics": true,

"brokerPerformanceTracking": true

}

}

],

"methods": [

"placeOrder(orderDetails, routingPreferences)",

"modifyOrder(orderId, modifications)",

"cancelOrder(orderId)",

"getOrderStatus(orderId)",

"calculateOptimalPositionSize(symbol, strategy, riskParameters)",

"analyzeExecutionQuality(tradeId, benchmarks)",

"monitorRiskExposure(portfolioId, riskMetrics)"

]

},

{

"name": "AnalyticsService",

"description": "Market and portfolio analytics engine",

"components": [

{

"name": "TechnicalAnalyzer",

"description": "Performs technical analysis on market data",

"features": {

"indicatorCalculation": true,

"patternRecognition": true,

"signalGeneration": true,

"multiTimeframeAnalysis": true,

"customIndicatorSupport": true

}

},

{

"name": "FundamentalAnalyzer",

"description": "Analyzes fundamental company data",

"features": {

"financialStatementAnalysis": true,

"ratioCalculation": true,

"peerComparison": true,

"valuationModeling": true,

"earningsAnalysis": true

}

},

{

"name": "PortfolioAnalyzer",

"description": "Analyzes portfolio performance and characteristics",

"features": {

"performanceAttribution": true,

"riskMetrics": true,

"correlationAnalysis": true,

"scenarioTesting": true,

"assetAllocationOptimization": true

}

},

{

"name": "PredictiveModels",

"description": "Machine learning-based prediction models",

"features": {

"pricePrediction": true,

"volatilityForecasting": true,

"trendClassification": true,

"anomalyDetection": true,

"sentimentImpactModeling": true

}

}

],

"methods": [

"calculateIndicators(symbol, timeframe, indicators, params)",

"detectPatterns(symbol, timeframe, patternTypes)",

"analyzeFundamentals(symbol, metrics, compareWith)",

"evaluatePortfolio(portfolioId, metrics, benchmarks)",

"generateSignals(symbol, strategy, timeframe)",

"runPredictiveModel(symbol, modelType, parameters)",

"exportAnalysisToGitHub(analysisId, repository, path)"

]

},

{

"name": "BacktestingService",

"description": "Strategy backtesting and optimization engine",

"components": [

{

"name": "HistoricalSimulator",

"description": "Simulates trading strategies on historical data",

"features": {

"accurateOrderExecution": true,

"multiAssetTesting": true,

"customSlippageModels": true,

"transactionCostModeling": true,

"dividendHandling": true

}

},

{

"name": "StrategyOptimizer",

"description": "Optimizes strategy parameters",

"features": {

"parameterSweep": true,

"geneticAlgorithms": true,

"machineLearningSuggestions": true,

"walkForwardTesting": true,

"robustnessAnalysis": true

}

},

{

"name": "PerformanceAnalyzer",

"description": "Analyzes backtest performance",

"features": {

"standardMetrics": true,

"drawdownAnalysis": true,

"tradingStatistics": true,

"benchmarkComparison": true,

"riskadjustedMetrics": true

}

}

],

"methods": [

"runBacktest(strategyId, symbols, timeframe, dateRange, parameters)",

"optimizeStrategy(strategyId, parameterRanges, optimizationGoal)",

"analyzeBacktestResults(backtestId, analysisParameters)",

"compareBacktests(backtestIds, comparisonMetrics)",

"generateBacktestReport(backtestId, reportFormat)",

"exportBacktestToGitHub(backtestId, repository, path)"

]

},

{

"name": "AlertService",

"description": "Trading alerts and notifications",

"methods": [

"createAlert(userId, alertConfig)",

"checkAlertConditions(alertId, currentData)",

"triggerAlert(alertId, triggerData)",

"notifyUser(userId, notification, channels)",

"getAlertHistory(userId, timeRange, status)"

]

},

{

"name": "JournalService",

"description": "Trading journal and performance tracking",

"methods": [

"createJournalEntry(userId, entryData)",

"attachTradeData(entryId, tradeIds)",

"generatePerformanceStats(userId, timeframe)",

"exportJournalToGitHub(userId, dateRange, repository, path)",

"importJournalFromGitHub(userId, repository, path)"

]

},

{

"name": "GitHubIntegrationService",

"description": "Manages GitHub connectivity for data sharing and backup",

"methods": [

"connectUserAccount(userId, githubCode)",

"getRepositories(userId)",

"exportDataToRepository(userId, dataType, data, repository, path)",

"importDataFromRepository(userId, dataType, repository, path)",

"createTradingGist(userId, content, description, isPublic)",

"syncUserSettings(userId, settings)"

]

}

],

"selfCheckSystems": [

{

"name": "DataQualityMonitor",

"description": "Monitors and validates market data quality",

"checks": [

{

"name": "PriceDataConsistency",

"description": "Checks for gaps, spikes, and inconsistencies in price data",

"frequency": "real-time",

"severity": "high",

"actions": [

"flagInconsistentData",

"switchToAlternativeSource",

"notifyAdministrator"

]

},

{

"name": "CrossExchangeValidation",

"description": "Compares data across exchanges for consistency",

"frequency": "every minute",

"severity": "medium",

"actions": [

"calculateDeviationMetrics",

"identifyOutlierSources",

"applyWeightedAveraging"

]

},

{

"name": "HistoricalDataCompleteness",

"description": "Verifies completeness of historical datasets",

"frequency": "daily",

"severity": "medium",

"actions": [

"identifyGaps",

"scheduleFillOperations",

"reportDataQualityMetrics"

]

},

{

"name": "FundamentalDataFreshness",

"description": "Ensures fundamental data is up-to-date",

"frequency": "daily",

"severity": "low",

"actions": [

"checkLastUpdateTimestamp",

"scheduleUpdates",

"flagStaleData"

]

}

]

},

{

"name": "TradingStrategyValidator",

"description": "Validates trading strategies before execution",

"checks": [

{

"name": "BacktestPerformanceValidation",

"description": "Ensures strategies meet minimum performance criteria",

"parameters": {

"minimumSharpeRatio": 0.8,

"maximumDrawdown": 20,

"minimumProfitFactor": 1.5,

"minimumWinRate": 40

},

"actions": [

"flagUnderperformingStrategies",

"suggestOptimizations",

"requireManualApproval"

]

},

{

"name": "OverfittingDetection",

"description": "Detects potentially overfit strategies",

"parameters": {

"walkForwardValidation": true,

"outOfSampleTesting": true,

"parameterSensitivityAnalysis": true

},

"actions": [

"calculateOverfitScore",

"warnUser",

"suggestSimplification"

]

},

{

"name": "RiskParameterValidation",

"description": "Validates risk parameters are within acceptable limits",

"parameters": {

"maximumPositionSize": "5% of portfolio",

"maximumCorrelation": 0.7,

"minimumRiskRewardRatio": 1.5

},

"actions": [

"enforceRiskLimits",

"suggestAdjustments",

"logValidationResults"

]

},

{

"name": "MarketConditionCheck",

"description": "Checks if strategy is appropriate for current market conditions",

"parameters": {

"volatilityChecks": true,

"trendAnalysis": true,

"correlationShifts": true

},

"actions": [

"assessMarketRegime",

"warnIfMismatch",

"suggestAdaptations"

]

}

]

},

{

"name": "OrderExecutionVerifier",

"description": "Verifies order execution quality and compliance",

"checks": [

{

"name": "SlippageMonitor",

"description": "Monitors execution slippage against expectations",

"parameters": {

"acceptableSlippagePercent": 0.1,

"highVolumeAdjustment": 0.2,

"volatilityAdjustment": true

},

"actions": [

"calculateActualVsExpectedSlippage",

"flagExcessiveSlippage",

"adjustRoutingPreferences"

]

},

{

"name": "ExecutionTimeMonitor",

"description": "Monitors order execution time",

"parameters": {

"maximumLatencyMs": 500,

"criticalLatencyMs": 2000

},

"actions": [

"trackExecutionTimes",

"alertOnDelays",

"switchRoutingOnFailure"

]

},

{

"name": "CostVerification",

"description": "Verifies trading costs against expectations",

"parameters": {

"commissionTolerance": 0.05,

"spreadTolerance": 0.1

},

"actions": [

"calculateActualCosts",

"compareWithEstimates",

"reportDiscrepancies"

]

},

{

"name": "RegulatoryCompliance",

"description": "Ensures trades comply with regulatory requirements",

"parameters": {

"washSalePrevention": true,

"patternDayTraderRules": true,

"marginRequirements": true

},

"actions": [

"preTradeComplianceCheck",

"blockNoncompliantOrders",

"logComplianceIssues"

]

}

]

},

{

"name": "PortfolioRiskMonitor",

"description": "Monitors portfolio risk metrics in real-time",

"checks": [

{

"name": "ExposureMonitor",

"description": "Monitors overall market exposure",

"parameters": {

"maximumNetExposure": 100,

"maximumSectorExposure": 30,

"maximumSingleAssetExposure": 10

},

"actions": [

"calculateCurrentExposures",

"alertOnThresholdBreaches",

"suggestHedgingStrategies"

]

},

{

"name": "CorrelationMonitor",

"description": "Monitors inter-position correlations",

"parameters": {

"correlationThreshold": 0.7,

"minimumDiversification": "4 uncorrelated positions"

},

"actions": [

"updateCorrelationMatrix",

"identifyConcentrationRisks",

"suggestDiversification"

]

},

{

"name": "VolatilityMonitor",

"description": "Monitors portfolio volatility",

"parameters": {

"targetVolatility": 15,

"maximumVolatility": 25,

"volatilityWindow": "20 days"

},

"actions": [

"calculateRealizedVolatility",

"forecastExpectedVolatility",

"suggestVolatilityAdjustments"

]

},

{

"name": "DrawdownMonitor",

"description": "Monitors portfolio drawdowns",

"parameters": {

"warningThreshold": 5,

"actionThreshold": 10,

"emergencyThreshold": 15

},

"actions": [

"trackDrawdownDepth",

"implementDrawdownProtocols",

"executeHedgingStrategies"

]

}

]

},

{

"name": "SystemHealthMonitor",

"description": "Monitors overall system health and performance",

"checks": [

{

"name": "DataFeedStatus",

"description": "Monitors status of all data feeds",

"frequency": "every minute",

"severity": "critical",

"actions": [

"checkConnectionStatus",

"measureLatency",

"activateBackupSources"

]

},

{

"name": "APIPerformance",

"description": "Monitors API performance and response times",

"frequency": "continuous",

"severity": "high",

"actions": [

"trackResponseTimes",

"identifyBottlenecks",

"scaleResourcesIfNeeded"

]

},

{

"name": "DatabasePerformance",

"description": "Monitors database performance",

"frequency": "every 5 minutes",

"severity": "high",

"actions": [

"monitorQueryTimes",

"checkIndexEfficiency",

"optimizeSlowQueries"

]

},

{

"name": "GithubIntegrationStatus",

"description": "Verifies GitHub integration functionality",

"frequency": "hourly",

"severity": "medium",

"actions": [

"verifyAuthentication",

"checkPermissions",

"testSampleOperations"

]

}

]

}

]

},

"thirdPartyIntegrations": [

{

"name": "BrokerageAPIs",

"description": "Integration with multiple brokerage platforms",

"integrations": [

{

"name": "Interactive Brokers",

"features": [

"Trading API",

"Market Data",

"Account Information",

"Order Management",

"Position Tracking"

],

"apiType": "REST and WebSockets"

},

{

"name": "TD Ameritrade",

"features": [

"Trading API",

"Market Data",

"Option Chains",

"Account Information",

"Streaming Data"

],

"apiType": "REST and Streaming"

},

{

"name": "Alpaca",

"features": [

"Trading API",

"Market Data",

"Paper Trading",

"Fractional Shares",

"Websocket Streams"

],

"apiType": "REST and WebSockets"

},

{

"name": "FOREX.com",

"features": [

"FX Trading API",

"Market Data",

"Technical Indicators",

"Account Management",

"Order Execution"

],

"apiType": "REST and FIX"

}

]

},

{

"name": "DataProviders",

"description": "Market data providers for comprehensive coverage",

"integrations": [

{

"name": "Polygon.io",

"dataTypes": [

"Real-time Quotes",

"Historical Data",

"Options Data",

"Reference Data",

"WebSocket Streams"

]

},

{

"name": "Alpha Vantage",

"dataTypes": [

"Stock Time Series",

"Forex Data",

"Crypto Data",

"Technical Indicators",

"Fundamental Data"

]

},

{

"name": "IEX Cloud",

"dataTypes": [

"Real-time Data",

"Historical Data",

"Corporate Actions",

"News",

"Alternative Data"

]

},

{

"name": "Finnhub",

"dataTypes": [

"Real-time Data",

"Company Fundamentals",

"Economic Data",

"Sentiment Analysis",

"Insider Transactions"

]

}

]

},

{

"name": "NewsAndSentiment",

"description": "News and sentiment analysis providers",

"integrations": [

{

"name": "Benzinga",

"features": [

"Financial News",

"Corporate Calendars",

"Analyst Ratings",

"Dividend Data"

]

},

{

"name": "Sentifi",

"features": [

"Social Media Sentiment",

"Market Buzz Analytics",

"Trend Detection",

"Impact Analysis"

]

},

{

"name": "StockTwits",

"features": [

"Social Sentiment",

"Message Streaming",

"Trending Symbols",

"User Engagement Metrics"

]

}

]

},

{

"name": "TradingViewIntegration",

"description": "Integration with TradingView for advanced charting",

"features": [

"Embedded Charts",

"Indicator Sharing",

"Custom Scripts",

"Chart Synchronization",

"Drawing Tool Synchronization"

]

},

{

"name": "GitHub",

"description": "GitHub integration for code management, data sharing, and collaboration",

"features": [

"OAuth Authentication",

"Repository Management",

"Content Management",

"Gist Creation and Management",

"Webhook Integration",

"Issue Management",

"Pull Request Handling",

"User Content Synchronization"

],

"apiEndpoints": [

"GitHub REST API",

"GitHub GraphQL API"

]

}

]

},

"security": {

"authentication": {

"methods": [

"JWT tokens",

"Refresh tokens",

"Two-factor authentication",

"Biometric authentication (for mobile)",

"OAuth (GitHub, brokerage accounts)"

],

"passwordPolicy": {

"minLength": 12,

"requireUppercase": true,

"requireLowercase": true,

"requireNumbers": true,

"requireSpecialChars": true,

"preventReuseCount": 5

}

},

"dataEncryption": {

"atRest": "AES-256",

"inTransit": "TLS 1.3",

"sensitiveBrokerageData": "Double encryption with separate keys"

},

"accessControl": {

"rbacEnabled": true,

"mfaRequiredForSensitiveOperations": true,

"ipRestrictionOption": true,

"sessionTimeout": 30,

"concurrentSessionLimit": 3

},

"auditLogging": {

"enabled": true,

"loggedActions": [

"Authentication attempts",

"Order submissions",

"Settings changes",

"API key management",

"Brokerage connections",

"Data exports"

],

"retentionPeriod": "1 year"

},

"complianceFeatures": {

"tradingRestrictions": {

"patternDayTraderProtection": true,

"marginRequirementChecks": true,

"washSalePrevention": true

},

"dataRetention": {

"tradeRecords": "7 years",

"accountStatements": "7 years",

"communicationRecords": "5 years"

}

}

},

"deployment": {

"environments": {

"development": {

"infrastructure": "AWS",

"compute": "EC2 (t3.large)",

"database": {

"primary": "MongoDB Atlas (M10)",

"timeSeries": "TimescaleDB (2 vCPU)",

"cache": "Redis Cloud (1GB)"

},

"containerization": "Docker"

},

"staging": {

"infrastructure": "AWS",

"compute": "EKS (3 nodes)",

"database": {

"primary": "MongoDB Atlas (M30)",

"timeSeries": "TimescaleDB (4 vCPU)",

"cache": "Redis Cloud (5GB)"

},

"containerization": "Kubernetes"

},

"production": {

"infrastructure": "AWS",

"compute": "EKS (auto-scaling 5-20 nodes)",

"database": {

"primary": "MongoDB Atlas (M60, sharded)",

"timeSeries": "TimescaleDB (8 vCPU, high availability)",

"cache": "Redis Enterprise (20GB, cluster)"

},

"containerization": "Kubernetes",

"cdn": "CloudFront",

"loadBalancing": "ALB with WAF"

}

},

"cicd": {

"provider": "GitHub Actions",

"workflows": {

"backend": [

"test-lint.yml",

"build-deploy.yml",

"database-migration.yml"

],

"frontend": [

"build-test.yml",

"browser-tests.yml",

"deploy.yml"

],

"desktop": [

"electron-build.yml",

"electron-release.yml"

],

"mobile": [

"react-native-build.yml",

"app-store-deploy.yml",

"play-store-deploy.yml"

]

},

"qualityGates": {

"codeCoverage": "80%",

"testPassing": "100%",

"performanceBaseline": true,

"securityScan": true

}

},

"monitoring": {

"services": [

{

"name": "DataDog",

"metrics": [

"API response time",

"Database query performance",

"Market data latency",

"Order execution time",

"User session analytics"

]

},

{

"name": "New Relic",

"metrics": [

"Application performance",

"Transaction tracing",

"Error tracking",

"Availability monitoring"

]

},

{

"name": "Sentry",

"features": [

"Error tracking",

"Exception handling",

"User impact analysis",

"Release tracking"

]

}

],

"customDashboards": [

"System Health Overview",

"Market Data Quality",

"Trading Performance",

"User Engagement",

"API Performance"

],

"alerts": {

"dataFeedDisruption": "Critical",

"orderExecutionDelay": "Critical",

"databaseLatency": "High",

"apiErrorRate": "High",

"securityEvents": "Critical"

}

},

"backupStrategy": {

"databases": {

"frequency": "Continuous with point-in-time recovery",

"retention": "30 days"

},

"userContent": {

"frequency": "Daily",

"retention": "90 days"

},

"githubSync": {

"enabled": true,

"frequency": "Daily",

"encryptedBackups": true

}

}

},

"platformSupport": {

"web": {

"browsers": [

"Chrome (last 2 versions)",

"Firefox (last 2 versions)",

"Safari (last 2 versions)",

"Edge (last 2 versions)"

],

"responsiveDesign": {

"desktop": {

"multiColumnLayouts": true,

"advancedCharting": true,

"multipleWorkspaces": true

},

"tablet": {

"adaptiveLayouts": true,

"touchOptimized": true,

"reducedFeatureSet": false

},

"mobile": {

"simplifiedInterface": true,

"prioritizedContent": true,

"gestureControls": true

}

},

"offlineCapabilities": {

"caching": true,

"localDataStorage": true,

"syncOnReconnect": true

}

},

"desktop": {

"platforms": [

"Windows 10/11",

"macOS 11+",

"Linux (Ubuntu, Debian)"

],

"features": {

"enhancedPerformance": true,

"multiMonitorSupport": true,

"systemNotifications": true,

"hotkeys": true,

"localDataStorage": true

},

"distributions": {

"directDownload": true,

"appStore": true,

"autoUpdates": true

}

},

"mobile": {

"platforms": [

"iOS 14+",

"Android 9+"

],

"formFactors": [

"Phone",

"Tablet"

],

"features": {

"biometricAuthentication": true,

"pushNotifications": true,

"offlineMode": true,

"widgetSupport": true,

"backgroundRefresh": true

},

"optimizations": {

"batteryUsage": true,

"dataUsage": true,

"storageUsage": true

},

"distributions": {

"appStore": true,

"playStore": true,

"enterpriseDistribution": true

}

}

}

}