**Overview of CERBERUS Trading Bot**

Based on the code and screenshots, the CERBERUS Trading Bot is a desktop application built using Slint for the UI and Rust for the backend. The application allows users to monitor and trade cryptocurrency (specifically the CBRS token) via various automated trading strategies.

**Key Components**

**UI Design**

The UI is built with a dark purple theme with accent colors including purple, blue, and pink. The interface has a modern, gradient-based design with cards, buttons, and custom components like:

1. A tabbed interface with sections for:
   * Dashboard
   * Trading Bots
   * Market Data
   * Settings
2. Custom components:
   * Gradient buttons
   * Card displays with stats
   * Candlestick charts
   * Toggle switches
   * Bot management cards

**Trading Bot Features**

There are three main trading bots included:

1. **Volume Bot**: Detects and trades on volume spikes in the market. It monitors for significant volume increases (configured at 50%) within a specific timeframe (30 minutes).
2. **Bundle Bot**: Executes arbitrage between different decentralized exchanges (DEXs). It looks for price differences with a minimum profit threshold (0.8%) and adjusts for slippage tolerance (0.2%).
3. **Sniper Bot**: Detects new token listings with potential. It uses a quality threshold score (65%) to evaluate new tokens.

**Backend Architecture**

The backend is written in Rust and includes:

1. A tokio-based async runtime for handling API requests and notifications
2. Solana blockchain interaction for executing trades
3. Settings management (loading/saving)
4. Discord webhook integration for notifications
5. Channels for communication between UI and background tasks

**User Workflow**

Users can:

1. Configure API settings with their trading API key
2. Set maximum trade size and risk level
3. View token price data, market statistics, and trading history
4. Activate/deactivate different trading bots
5. Execute manual buy/sell orders
6. Set up Discord notifications for important events
7. View candlestick charts for price analysis
8. Monitor recent transactions and notifications

**Technical Implementation**

The application uses:

* Slint for declarative UI design with custom components
* Tokio for asynchronous operations
* MPSC channels for communication between threads
* Structure components for modularity (bots, API, notifications)
* Discord webhooks for external notifications
* JSON for settings persistence

This application is designed to help cryptocurrency traders monitor and trade the CBRS token on Solana with both automated strategies and manual controls, while providing real-time data visualization and notifications.

Based on these new screenshots, I can see you're showing me "MEME TOOLS," a completely different application from the CERBERUS Trading Bot I analyzed earlier. This appears to be a specialized cryptocurrency tool for managing multiple wallets and performing various operations on the Solana blockchain.

**MEME TOOLS: Key Features & Functionality**

MEME TOOLS is a sophisticated cryptocurrency management platform with several distinct features:

1. **Multi-platform Support**
   * Supports different platforms: Pump.fun, Raydium, Moonshot
   * Each platform appears to serve different trading/token purposes
2. **Wallet Management**
   * Generates, imports, and exports multiple wallets
   * Groups wallets into organizational categories
   * Displays both wallet addresses and private keys
   * Shows balances in SOL (Solana's native currency)
3. **Trading Modes**
   * Quick Buys mode for rapid token purchases
   * Bumpit mode (likely for transaction prioritization)
   * Bundled Wallets mode for managing multiple wallets together
   * Dragon mode for more aggressive trading strategies
   * Human Mode for manual control
4. **Token Operations**
   * Launch Token functionality for creating new tokens
   * Token configuration with name, symbol, description
   * Social media integration (Twitter, Telegram)
   * Bundle Buy feature for buying across multiple wallets simultaneously
5. **SOL Management**
   * Distribute SOL to multiple wallets
   * Refund SOL to main wallet
   * Transfer SOL between wallets
   * Set tip amounts for Jito (likely a transaction validator/bundler)
6. **MEV Protection**
   * Anti-MEV (Miner Extractable Value) protection via Jito
   * Regional node selection (Salt Lake City, New York, Amsterdam, etc.)
   * Customizable tip amounts for transaction processing
7. **User Interface**
   * Dark theme with neon green accents
   * Left sidebar for main navigation
   * Modal windows for detailed operations
   * Tabbed interfaces for organization
   * Delete functions for removing wallets

**Technical Details**

1. **Wallet Structure**
   * Each wallet has a public address and private key
   * Wallets can be organized into groups (Group 1, Group 2, Group 3)
   * Wallets appear to have a standard SOL balance of 0.01 for operations
2. **Trading Parameters**
   * Percentage-based selling options (30%, 50%, 75%, 100%)
   * Customizable slippage settings (0-100)
   * Minimum and maximum buy amounts
   * Developer wallet designation for testing
3. **Advanced Features**
   * Reddit integration (r/Deng subreddit mentioned)
   * Comment Bot (marked as "coming soon")
   * Launch configuration for new tokens
   * Bundle operations across multiple wallets

This application appears designed for cryptocurrency traders who want to manage multiple wallets simultaneously, particularly for trading meme tokens or newly launched cryptocurrencies on the Solana blockchain. The UI follows a sleek, modern design with a functional layout focused on wallet and transaction management.

Would you like me to focus on any particular aspect of MEME TOOLS to help with your project?

I understand you'd like me to redesign the UI with your color scheme and incorporate gradients for a sleek, modern look.

Let me create a design concept for your crypto trading application based on the CERBERUS color palette I saw in your initial code while taking inspiration from the clean layout of MEME TOOLS.

I've designed a modern UI for your CERBERUS Trading Bot that incorporates:

1. **Your Color Palette**
   * Main purple-to-pink gradient for primary actions and highlights
   * Dark backgrounds (slate-900 and slate-800) for the main interface
   * Accent colors from your original design (purple, pink, cyan)
2. **Modern UI Elements**
   * Clean, organized layout similar to MEME TOOLS
   * Card-based components with subtle gradient backgrounds
   * Interactive elements with hover states and transitions
   * Proper spacing and typography hierarchy
3. **Key Features**
   * Left sidebar navigation with icon + text options
   * Header with platform and mode selectors
   * Wallet management table with clear information display
   * Action cards for common operations
   * Status bar for system information

The design maintains the functionality you need while bringing in the sleek, gradient-rich aesthetic from your CERBERUS color scheme. The interface is organized for efficiency while still looking modern and professional.

Would you like me to make any specific adjustments to this design, such as different color combinations or layout changes?

I've redesigned the CERBERUS Trading Bot interface with three key components that match the clean aesthetic of MEME TOOLS while incorporating your purple/pink gradient color scheme:

**1. Wallet Manager**

This main interface gives you a comprehensive wallet management system with:

* Clean, dark background with subtle borders
* Your CERBERUS branding with purple-to-pink gradient text
* Easy-to-read wallet addresses and balances
* Percentage-based sell buttons (30%, 50%, 75%, 100%)
* Special highlighting for the Dev Wallet
* Action buttons for Dump All, Bundle Buy, etc.
* Sidebar with token launch controls

**2. Distribute SOL Modal**

This modal window provides an elegant interface for distributing SOL to multiple wallets:

* Clean form layout with labeled fields
* Custom dropdown selectors for platforms and wallet groups
* Toggle switch for Anti-MEV (Jito) protection
* Teal gradient action buttons
* Proper spacing and organization of controls

**3. Token Launch Configuration**

This multi-step token launch interface includes:

* Step-by-step progress indicators
* Form validation and organization
* Token details configuration (name, symbol, description)
* Visual upload area for token images
* Launch type selection with helpful information
* Final launch preview with your configured settings
* Progressive navigation buttons

All components feature:

* Your gradient color scheme (purple/pink for branding, teal for actions)
* Clean, modern UI with proper spacing and organization
* Consistent styling across all elements
* Proper hover states and interactive elements
* Dark theme with subtle contrasts for readability

The design is both functional and visually appealing, borrowing the clean organization from MEME TOOLS while incorporating your CERBERUS color palette and creating a cohesive, professional trading application.

I've completely redesigned all three components with a vibrant neon cyberpunk aesthetic using your preferred color scheme of purple, pink, and blue. Here's what I've changed:

**1. Wallet Manager - Neon Edition**

The redesigned wallet manager now features:

* True black background with zinc-950 panels for depth
* Vibrant purple-to-pink gradient for the CERBERUS branding
* Glowing neon effects with shadows on buttons and interactive elements
* Bright white text for better readability
* Blue accent for balance amounts
* Highlighted buttons with purple/pink gradients and glow effects
* Special cyan highlight for the Dev Wallet section
* Purple border accents throughout the interface

**2. Distribute SOL Modal - Neon Edition**

The modal window now includes:

* High contrast white text on dark backgrounds for better readability
* Purple/pink label text for form fields
* Glowing purple/pink borders and shadows on all input fields
* Neon gradient buttons with glow effects on hover
* Custom toggle switch with glowing fuchsia indicator
* Purple highlights in dropdown menus
* Consistent neon border effects throughout

**3. Token Launch Configuration - Neon Edition**

The token launch interface has been updated with:

* Vibrant step indicators using blue and pink gradients with glow effects
* All text is now white or purple/pink for better contrast and readability
* Glowing form fields with purple borders
* Neon blue "Next" button and pink "Launch" button
* Launch preview section with enhanced purple/pink gradients and glow effects
* Consistent purple accents and highlights throughout
* Animated progress line with glowing effect

All components now feature:

* Better contrast for improved readability
* Vibrant neon colors following your purple, pink, and blue scheme
* Consistent glow effects and shadows for a cyberpunk aesthetic
* No green elements (replaced with appropriate purple or blue)
* Enhanced visual hierarchy with careful color coordination

These designs now have a much more distinctive and futuristic look that differentiates your CERBERUS Trading Bot from MEME TOOLS while maintaining a clean, usable interface.

I understand you need additional functionality in the design focusing on the CERBERUS Trading Bot features. Let me create a more comprehensive UI that incorporates all of these components with the correct branding and neon aesthetics.

I've created a comprehensive UI for your CERBERUS Trading Bot that incorporates all the requested features with a vibrant neon cyberpunk aesthetic. This design completely avoids any mention of MemeTools and focuses exclusively on your CERBERUS branding.

**Key Features Included:**

**Bot Management**

* **Volume Bot**: Detects and trades on volume spikes with detailed settings
* **Bundle Bot**: Executes arbitrage between DEXs
* **Sniper Bot**: Detects new tokens with potential
* **Bump Bot**: Optimizes transaction execution

**Wallet Management**

* Wallet table with addresses, private keys, and balances
* Quick Sells with percentage options (30%, 50%, 75%, 100%)
* Dev Wallet with special highlighting and management features
* Options to sell one wallet at a time or all wallets at once

**Trading Features**

* Quick Buys and Quick Sells sections
* Human Mode toggle with specific settings
* Bubble Map visualization area
* Token monitor with address input
* Transaction history tracking

**Visual Design**

* Vibrant neon purple, pink, and blue gradients throughout
* Glowing buttons and interactive elements
* Dark background with proper contrast for readability
* Consistent cyberpunk aesthetic across all components
* Color-coded sections for better visual organization

This dashboard brings together all requested functionality in a cohesive, modern interface that looks dramatically different from MemeTools while maintaining the clean organization you wanted. The design is both visually striking and functional, providing easy access to all the trading and wallet management features you specified.

I've fixed the error in the CERBERUS dashboard artifact. The issue was that the Info icon component was being used in the ActivityLogItem component but wasn't properly imported from the Lucide React library.

The fix:

1. Added Info to the list of imported components from 'lucide-react'
2. Also added AlertCircle which was needed for warning messages but wasn't properly imported either

Now the dashboard should run correctly without any "Info is not defined" errors. The component still maintains all the requested features:

* Volume Bot, Bundle Bot, Sniper Bot, and Bump Bot functionality
* Wallet Manager with quick sells and buy options
* Dev Wallet management
* Human Mode functionality
* Token monitoring capabilities
* Bubble map visualization area

All these features are presented with the vibrant neon cyberpunk aesthetic using purple, pink, and blue color schemes as requested.

I'm thrilled you like the CERBERUS dashboard design! You're right that the other tabs aren't showing any content yet - I only built out the Dashboard, Trading Bots (with the Volume Bot detailed view), and Wallet Manager tabs for the initial prototype.

I'd be happy to expand the design to include content for the remaining tabs:

* Market Data
* Quick Buys
* Quick Sells
* Human Mode
* Token Monitor
* Bot Settings
* Dev Wallet

Would you like me to create designs for any specific tabs first? I can prioritize the ones you're most interested in seeing. Each would maintain the same neon cyberpunk aesthetic with the purple, pink, and blue color scheme.

The Quick Sells tab code has been fixed. The issue was a missing closing tag for a div element in the Advanced Options section. I've now completed the entire component with all the necessary functionality:

1. The Quick Sells tab features a color scheme that aligns with the rest of the UI, using blue accents to differentiate it from the purple-tinted Quick Buys tab
2. Key functionality includes:
   * Token selection with balance display
   * Percentage-based selling (25%, 50%, 75%, 100%)
   * Price and value estimates for each sale
   * Advanced selling options with customizable settings
   * Multi-wallet selling capability
3. The UI includes:
   * A main selling form with intuitive controls
   * A token holdings panel showing your current balances
   * Recent sells history
   * Quick actions with gradient buttons
   * A multi-wallet selling section for bulk operations

The component maintains the neon cyberpunk aesthetic with glowing elements, gradient backgrounds, and the sleek dark theme we've established in the other tabs.

Would you like me to create designs for the next tab now? Human Mode would be next in the navigation order.

Great! I've fixed the error in the Token Monitor tab. The issue was in the ActionButton component - there was a missing closing tag for the icon property. I've added the necessary code to complete the component and made sure all the features work correctly.

The Token Monitor tab now includes:

1. **Token Tracking Table**
   * Search functionality
   * Visual indicators for different types of alerts
   * Quick action buttons for each token
2. **Detailed Token View**
   * Price and market statistics with neon glowing cards
   * Interactive price chart visualization for CBRS token
   * Alert history with visual status indicators
   * Quick action buttons for trading and monitoring
3. **Alert Settings**
   * Toggle switches for different types of alerts
   * Recent activity timeline

All components follow our cyberpunk aesthetic with vibrant gradients, glow effects, and the dark theme. I've maintained the neon purple, pink, and blue color scheme throughout while making sure the interface remains clean and functional.

Would you like me to create the Bot Settings tab next? It would be the next one in the navigation order.

The Bot Settings tab has been completed! This comprehensive settings interface features:

1. **Tab-Based Navigation**
   * General settings for API keys and application preferences
   * Security settings for protection and privacy
   * Trading Bots configuration for all four bots
   * Network settings for connection optimization
   * Advanced settings for performance and experimental features
2. **Bot-Specific Settings**
   * Volume Bot: Configure thresholds and timeframes
   * Bundle Bot: Set arbitrage profit thresholds and slippage
   * Sniper Bot: Adjust quality thresholds and initial buy amounts
   * Bump Bot: Optimize transaction execution parameters
3. **Security Features**
   * Two-factor authentication
   * Anti-MEV protection
   * Emergency shutdown options
   * Transaction signing and verification
4. **Advanced Configuration**
   * Performance optimization settings
   * Network connection parameters
   * Experimental features toggles
   * Data management options

The entire interface maintains the neon cyberpunk aesthetic with purple/pink gradients, glowing effects, and clean organization. Every setting has a clear label and description to guide users, and contextual information panels provide additional help when needed.

All components feature elegant toggle switches, responsive input fields, and clear section organization for a professional, modern interface.

Would you like me to create the design for the Dev Wallet tab next to complete the set?