



Buriza Wallet: UI/UX Design and Front-End Development

Clark Alesna¹ Stan Kiefer Gallego² Caitlin Lindsay³

2025-10-06

clark@saib.dev
stan.gallego@saib.dev
caitlin.lindsay@saib.dev
SAIB Inc.

Table of Contents

I. Introduction	4
II. Buriza Front-End	6
ii.1 Design	6
ii.1.1 Material Design 3	7
ii.2 Development	9
ii.2.1 Project Architecture	9
ii.2.2 Platform Implementations	9
ii.2.3 Front-End Implementation	10
III. Implementation	12
iii.1 Mobile	12
iii.2 Browser	12
iii.3 Desktop	12
IV. User Interface	14
iv.1 Mobile	14
iv.1.1 Send Assets	14
iv.1.2 Receiving Assets	28
iv.1.3 View Balance and Assets	32
iv.1.4 Transaction History	36
iv.1.5 Full Node Setup	40
iv.1.6 dApp Access	44
iv.1.7 Wallet Creation and Account Management	48
iv.2 Browser	68
iv.2.1 Send Assets	68
iv.2.2 Receiving Assets	78
iv.2.3 View Balance and Assets	82
iv.2.4 Asset Selection	86
iv.2.5 Transaction History	88
iv.2.6 Full Node Setup	92
iv.2.7 dApp Access	98
iv.3 Desktop	102
iv.3.1 Send Assets	102
iv.3.2 Receiving Assets	108
iv.3.3 View Balance and Assets	110
iv.3.4 Transaction History	111
iv.3.5 Full Node Setup	114
iv.3.6 dApp Access	116
iv.3.7 Wallet Creation and Account Management	118
V. Platform Development Setup	126
v.1 Prerequisites	127
v.2 Installation	127
v.3 Platform Development Instructions	127
v.3.1 Desktop Application (macOS)	127
v.3.2 Desktop Application (Windows)	127

v.3.3 iOS Simulator	127
v.3.4 Physical iPhone	129
v.3.5 Android Development	129
v.3.6 Browser Extension	130
v.3.7 Progressive Web App	130
VI. Conclusion	131
VII. Links	132
VIII. References	133

I. Introduction

Buriza is a comprehensive Cardano wallet software suite spanning mobile, desktop, and browser platforms. The name Buriza is inspired by the Japanese interpretation of “blizzard,” symbolizing power, resilience, and control. Reflecting this spirit, Buriza empowers users with full sovereignty over their wallet infrastructure through open-source development and complete decentralization.

Funded under Project Catalyst Fund 13, the initiative is structured across three complementary proposals:

- [Buriza.Mobile](#) — focused on delivering a flexible and responsive mobile application,
- [Buriza.Browser](#) — the browser extension interface, and
- [Buriza.Desktop](#) — a desktop implementation featuring full-node capabilities.

Since Milestone 1, Buriza’s design has undergone significant evolution. The team adopted Material Design 3 (MD3) principles to create a more refined, accessible, and cohesive user experience. A major visual shift includes the introduction of Buriza Frostling, a bold, yeti-inspired mascot that replaces the earlier soft, gem-like designs. Frostling embodies Buriza’s core values—security, trust, and power—and appears throughout the interface to reinforce brand consistency.

The updated design also features enhanced iconography, a standardized color system based on Buriza’s three original core colors, and new accent shades that provide stronger visual contrast. Clean mockups and improved UI patterns further highlight the wallet’s sleek, modern aesthetic.

On the technical side, Buriza is built on a modular .NET 9 architecture, emphasizing shared components and platform independence. At the core of the system is the Buriza.UI shared component library, enabling a single codebase to serve multiple targets—browser extension, progressive web app (PWA), and cross-platform MAUI applications. This approach reduces redundant development and ensures a consistent user experience across all platforms.

The front-end implementation utilizes:

- Blazor for application logic and rendering,
- MudBlazor for MD3-compliant UI components, and
- Tailwind CSS for utility-first styling.

The browser extension is powered by Blazor WebAssembly and follows Manifest V3 standards. The PWA achieves near-native performance through WebAssembly, while the MAUI app leverages BlazorWebView to combine native device capabilities with a unified web-based UI.

With these design and architectural foundations in place, this document outlines the progress and deliverables completed for Milestone 2. It details the evolution of the user interface and system architecture since Milestone 1, explains the project setup and tools used, and documents the front-end implementation methodology. Screenshots, videos, and working examples are provided to demonstrate Buriza’s evolving functionality. The report also highlights the challenges encountered and the solutions developed by the SAIB Inc. team, reflecting its ongoing commitment to building a secure, performant, and user-centered Cardano wallet.

This report has been converted to Word format for Project Catalyst submission. The original version may be found in the Buriza project's GitHub repository: <https://github.com/SAIB-Inc/Buriza/tree/main/src/Buriza.Docs>

II. Buriza Front-End

This section covers the branding and UI/UX design of the Buriza wallet, describing the updates made since the first submission. Additionally, it tackles the implementation of the design - detailing the project set up and various tools utilized.

ii.1 Design

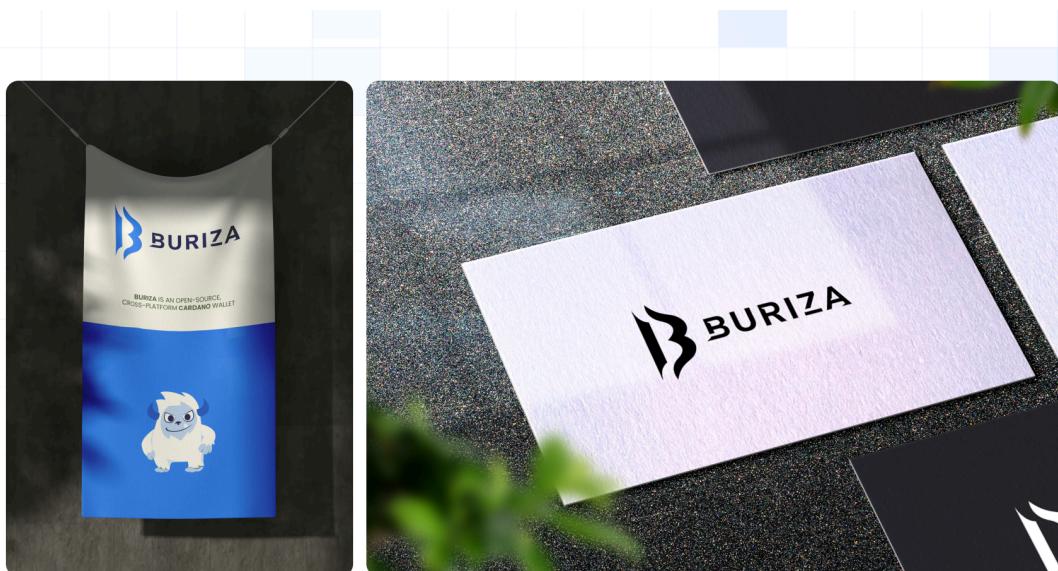
This section outlines Buriza's design evolution in alignment with the principles and best practices of Google's Material Design 3.

Buriza's design has significantly evolved since the submission of Project Catalyst's Milestone 1, with the initial branding and UI/UX designs detailed in the following [report](#).

A key transformation in Buriza's branding is the introduction of its new mascot, Buriza Frostling. Inspired by the yeti, Frostling's energetic expressions and bold, resilient personality stand in contrast to the earlier delicate, pastel mascots. This strong character is present throughout the application's design, reinforcing Buriza's core values of security, trust, and power.

The updated branding report also features enhanced iconography, an expanded color palette, and clean mockups that showcase Buriza's modern and sleek aesthetic. These comprehensive changes can be viewed in the revised [branding report](#), which includes detailed descriptions and illustrations of the logo and typography.

Figure 1 – Buriza Mockup: featuring the mascot, colors, and logo



Title:
Logo

Brand
Guidelines

Produced by
SAIB

Page
09

These developments are reflected in the overall UI and UX of the wallet, resulting in a more refined and visually cohesive application. The improvements stem from the opportunity the SAIB Inc. design team had to explore Material Design 3, using its guidelines to further elevate the project's look and feel. These refinements can be seen in detail in the following [Figma file](#).

ii.1.1 Material Design 3

Material Design is a design system created by Google designers and developers that provides UX foundations, guidance, and UI components across Android, Flutter, and the Web. Its most recent update, Material Design 3, highlights emotional impact. Placing great significance on expressiveness, dynamism, and accessibility, Material Design 3 expands the system with additional shapes, typography guidelines, intuitive motions, and more ([Material Design 3](#), 2025).

SAIB's design team studied the foundations of this update, noting and implementing changes based on Material Design 3's guidelines on color, elevation, icons, shape, motion, and typography. The impact of this study can be seen in the most recent design. Buriza's colors have been evaluated for contrast, and the initial color scheme has been updated to a more vibrant palette with accents and a greater variety of shades established for uniformity. Elevation is utilized to provide visual hierarchy and interaction guides for users, and labels have been added to the primary menu icons. Similar considerations have been applied to the wallet's shapes, with Material Design 3 button group designs added to many main pages and expressive animations applied in the main header. While Buriza's initial typography has not changed, revisions have been made to consider type roles within pages.

Figure 2 – Buriza Mobile UI - Home: design evolution from first to third iteration (left to right)

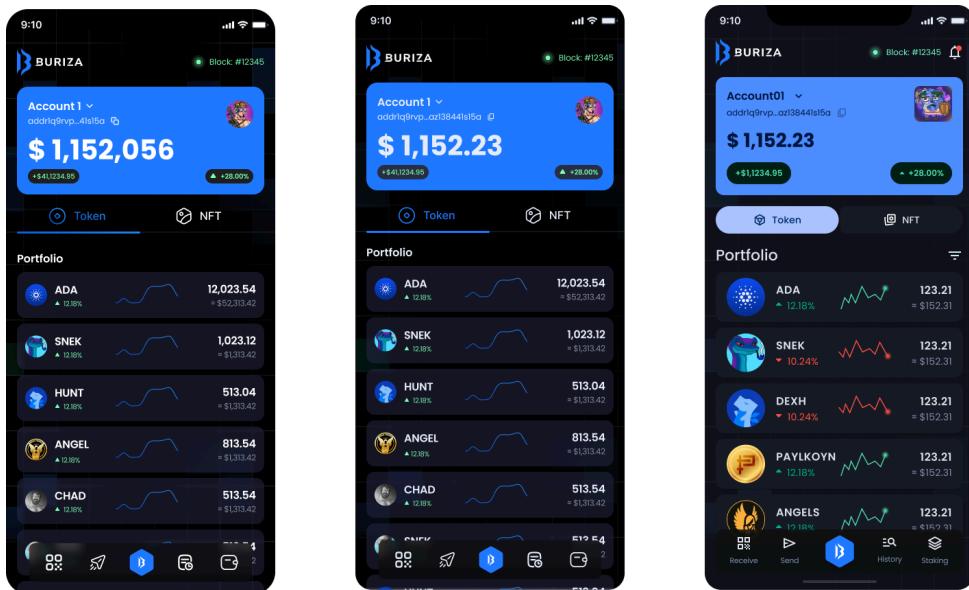


Figure 3 – Buriza Mobile UI - History: design evolution from first to third iteration (left to right)

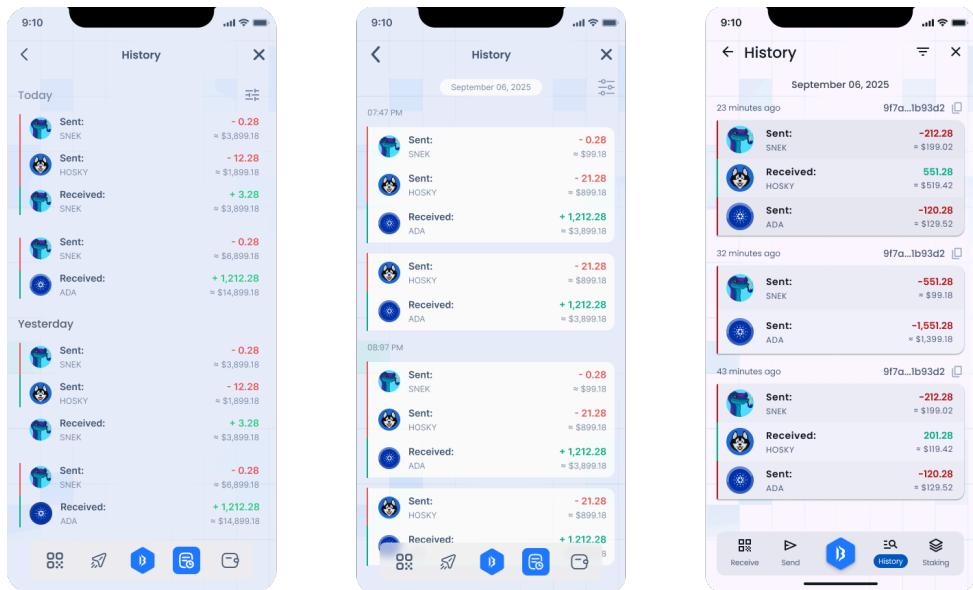
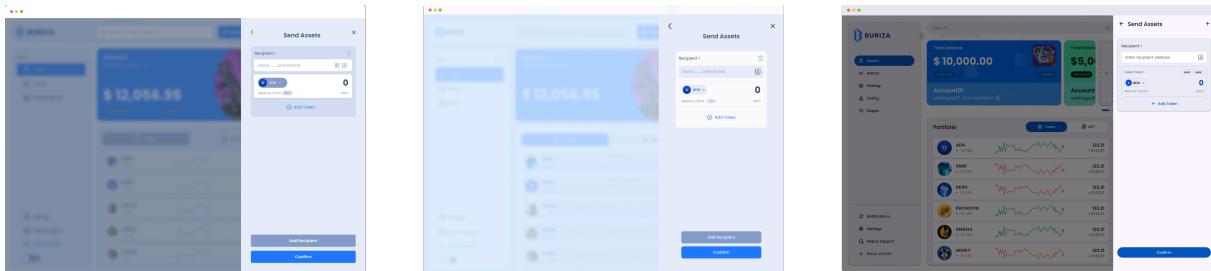


Figure 4 – Buriza Desktop UI - Send Assets: design evolution from first to third iteration (left to right)



Buriza's latest design update embodies SAIB's desire to create an impactful application that continues to evolve and improve. The team strives for continuous design iteration and improvement. With the future release of prototypes or mockups to the public, SAIB will continue to evolve through community feedback, validating users' needs and experiences while augmenting improvements with design solutions backed by research and best practices.

ii.2 Development

This section covers the technical implementation of Buriza's wallet suite, including architecture decisions, platform implementations, and front-end technologies.

ii.2.1 Project Architecture

Buriza's architecture is built around component sharing and platform independence, enabling a single codebase to serve multiple deployment targets.

ii.2.1.1 Solution Structure

Buriza uses a modular architecture with 5 .NET 9 projects designed for scalability and maintainability:

- **Buriza.UI** - Shared Blazor component library containing all user interface elements
- **Buriza.Data** - Core data models and services for wallet functionality
- **Buriza.Extension** - Browser extension for seamless dApp interaction
- **Buriza.Web** - Progressive web app for universal access
- **Buriza.App** - Cross-platform MAUI app for mobile and desktop

This architecture eliminates the need to implement separate UI layers for each platform. Instead of building distinct interfaces for the browser extension, web app, and mobile app, all platforms consume the same **Buriza.UI** components. This approach ensures consistent user experience while dramatically reducing development effort and maintenance overhead.

ii.2.1.2 Shared Component Library (**Buriza.UI**)

The component library provides a comprehensive set of reusable Blazor components organized into a hierarchical structure:

- **Common** - Foundational UI primitives like buttons, text fields, and navigation tabs
- **Controls** - Advanced interactive elements including asset cards and search functionality
- **Layout** - Application structure components for sidebars and main content areas
- **Pages** - Complete page/screen implementations for wallet operations like assets, transaction history, dapp access, send, and receive

The library leverages MudBlazor's Material Design 3 implementation alongside Tailwind CSS for utility-first styling and responsive design patterns. This combination ensures both visual consistency and flexible customization across all platform implementations.

ii.2.2 Platform Implementations

Each platform implementation leverages the shared **Buriza.UI** components while providing platform-specific features and deployment mechanisms.

ii.2.2.1 Browser Extension

The browser extension represents Buriza's most integrated approach to decentralized web interaction, embedding wallet functionality directly into the user's browsing experience. Built with Blazor WebAssembly and the Blazor.BrowserExtension framework, the extension follows Manifest V3 standards to ensure compatibility with modern browser security requirements and enhanced performance through service workers.

ii.2.2.2 Progressive Web App

The progressive web application serves as Buriza's most accessible deployment target, requiring only a modern web browser to provide full wallet functionality. Built with Blazor WebAssembly, the PWA delivers near-native performance by compiling C# code to WebAssembly bytecode that executes directly in the browser's runtime environment.

The PWA architecture enables installation directly from the browser without requiring app store distribution, creating a native-like experience while maintaining web-based deployment advantages.

ii.2.2.3 Cross-Platform App

The [MAUI](#) application targets iOS, Android, macOS, and Windows through a hybrid architecture that combines native platform capabilities with web-based UI components. At the core of this implementation is BlazorWebView, a native control that hosts Blazor content within a webview container while providing seamless integration with platform-specific APIs.

BlazorWebView acts as a bridge between the native MAUI shell and the Blazor UI layer, allowing the same **Buriza.UI** components to render within a native application context. This approach eliminates the need to rebuild the entire user interface using platform-specific controls like XAML, while still providing access to native device features such as secure storage, biometric authentication, and push notifications.

The hybrid model significantly reduces development complexity by leveraging existing web technologies and shared component libraries. Rather than maintaining separate native codebases for each platform, the application shares a single UI implementation while the MAUI framework handles platform-specific compilation and native API integration automatically.

ii.2.3 Front-End Implementation

The front-end implementation combines modern web technologies with component-driven architecture to deliver consistent user experiences across all platform deployments.

ii.2.3.1 Blazor Components

[Blazor](#) is Microsoft's web framework that serves as the foundational UI technology for Buriza, enabling C# development for web interfaces through WebAssembly compilation. This Microsoft-developed framework allows the entire Buriza application stack to use a single programming language, eliminating context switching between backend and frontend development.

The component architecture follows a hierarchical structure where complex UI elements are composed of smaller, reusable primitives. Blazor's two-way data binding simplifies form interactions and real-time updates, particularly important for wallet operations that require immediate visual feedback on transaction states and balance changes.

ii.2.3.2 MudBlazor Design System

[MudBlazor](#) v8.11.0 provides the Material Design 3 foundation for Buriza's visual language, delivering pre-built components with accessibility standards and smooth animations. The theming system integrates seamlessly with CSS custom properties, allowing Buriza to maintain brand identity while leveraging Material Design's proven usability patterns.

Buriza extends this foundation with Buriza-styled components:

Common Components:

- **BurizaButton** - Standardized button with consistent styling and interaction states
- **BurizaTextField** - Custom text input with validation and wallet-specific formatting
- **BurizaTabs** - Navigation tabs with custom styling for asset type switching
- **BurizaSelect** - Dropdown selection with wallet account and asset filtering
- **BurizaHeader** - Consistent header layout across all pages

Control Components:

- **BurizaAssetCard** - Asset display with balance, price changes, and interactive actions
- **BurizaSearchBar** - Universal search functionality for assets and transactions
- **BurizaDappCard** - dApp connection interface with authorization controls

ii.2.3.3 Tailwind CSS Integration

[Tailwind CSS](#) v4 complements MudBlazor by providing utility-first styling capabilities for custom layouts and responsive behavior. The integration uses a Bun build pipeline for rapid CSS compilation and automatic purging of unused styles, ensuring optimal bundle sizes.

The utility-first approach enables precise control over spacing, positioning, and responsive breakpoints, particularly valuable for wallet interfaces that require exact alignment for transaction details and balance displays. Custom CSS variables bridge the gap between Tailwind utilities and MudBlazor's theming system.

ii.2.3.4 Responsive Design Patterns

Buriza implements a mobile-first responsive design strategy that adapts to various screen sizes and interaction methods. The design system accommodates everything from mobile browser extensions to desktop applications while maintaining usability and visual hierarchy.

The responsive approach extends beyond screen size to consider platform-specific interaction patterns. Touch targets are appropriately sized for mobile interfaces, while desktop versions support keyboard navigation and hover states, ensuring consistent functionality regardless of how users access Buriza.

III. Implementation

This section outlines the implementation progress and highlights challenges encountered by the development team.

Buriza's unique infrastructure significantly streamlined the development process. Rather than maintaining separate codebases for each platform and operating system, the unified Buriza.UI codebase ensures compatibility across mobile, desktop, and web platforms—spanning Windows, macOS, Linux, Android, and iOS. As a result, platform-specific development primarily focused on using Tailwind's responsive directives to adapt the user interface across varying screen sizes.

To ensure both code quality and visual fidelity, the development process established had code peer-reviewed prior to being pushed to the repository. The team maintained a strong focus on building a beautiful, high-quality application through continuous collaboration and refinement.

iii.1 Mobile

Development began with a mobile-first approach, with the goal of closely following the design system established by the design team. The focus was on replicating the intended user experience, layout behavior, and visual responsiveness on mobile devices.

During this process, the team encountered a challenge related to how image assets were handled. Since the Buriza.App project (used to compile the mobile application) did not have direct access to image files stored in the shared Buriza.UI library, assets such as icons and illustrations could not be referenced at runtime.

To solve this, the team developed an internal utility that enables asset embedding directly into the application code. This ensured that all necessary visual elements remained accessible regardless of platform-specific limitations—eliminating the need for runtime file system access and enabling smooth, consistent rendering across platforms.

iii.2 Browser

Browser extension development leveraged the shared codebase, with only minor layout adjustments needed to fit the extension's popup dimensions. When fully expanded, the extension mirrors the desktop interface.

Similar to the mobile application, the extension faced challenges with accessing shared image assets once deployed in the browser environment. Since browser extensions operate in isolated environments with limited file system access, the same asset management solution used in the mobile app was also applied here. This unified approach allowed the team to maintain consistency in asset rendering without duplicating resources or code.

iii.3 Desktop

The desktop implementation also leveraged Blazor's flexible layout system. Developers were able to implement the drawer-based interface with minimal overhead, using dynamic layouts that adjusted based on page context.

Additionally, core features such as Send and Receive reused mobile UI patterns within drawers, allowing the team to focus on small adjustments rather than re-implementing entire views.

IV. User Interface

iv.1 Mobile

iv.1.1 Send Assets

Figure 5 – Buriza UI - Mobile: Send assets

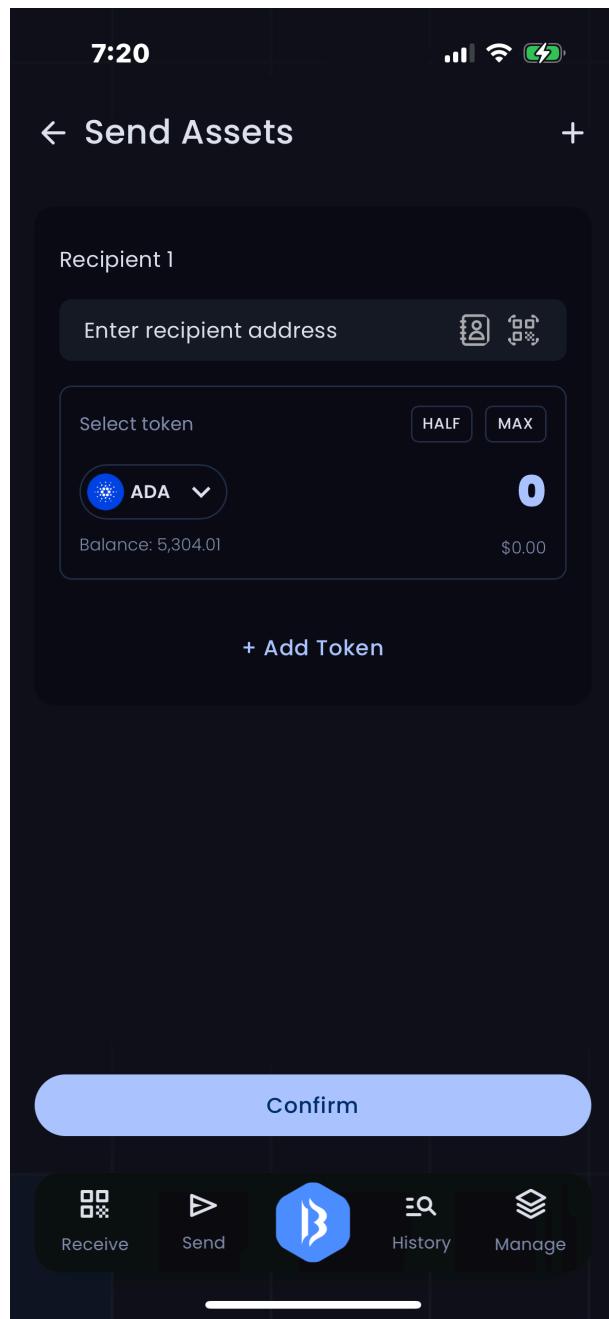


Figure 6 – Buriza UI - Mobile: Send assets

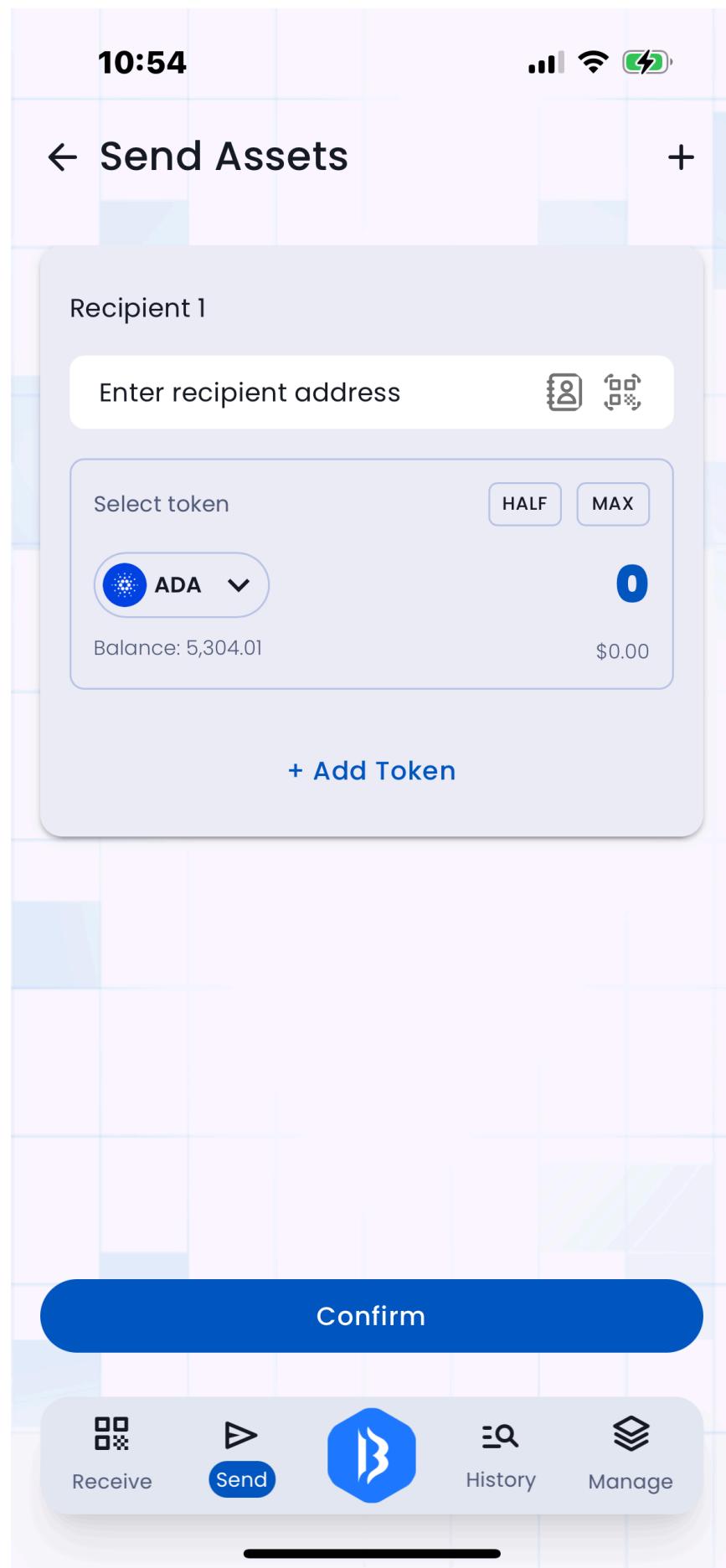


Figure 7 – Buriza UI - Mobile: Send assets with an additional recipient

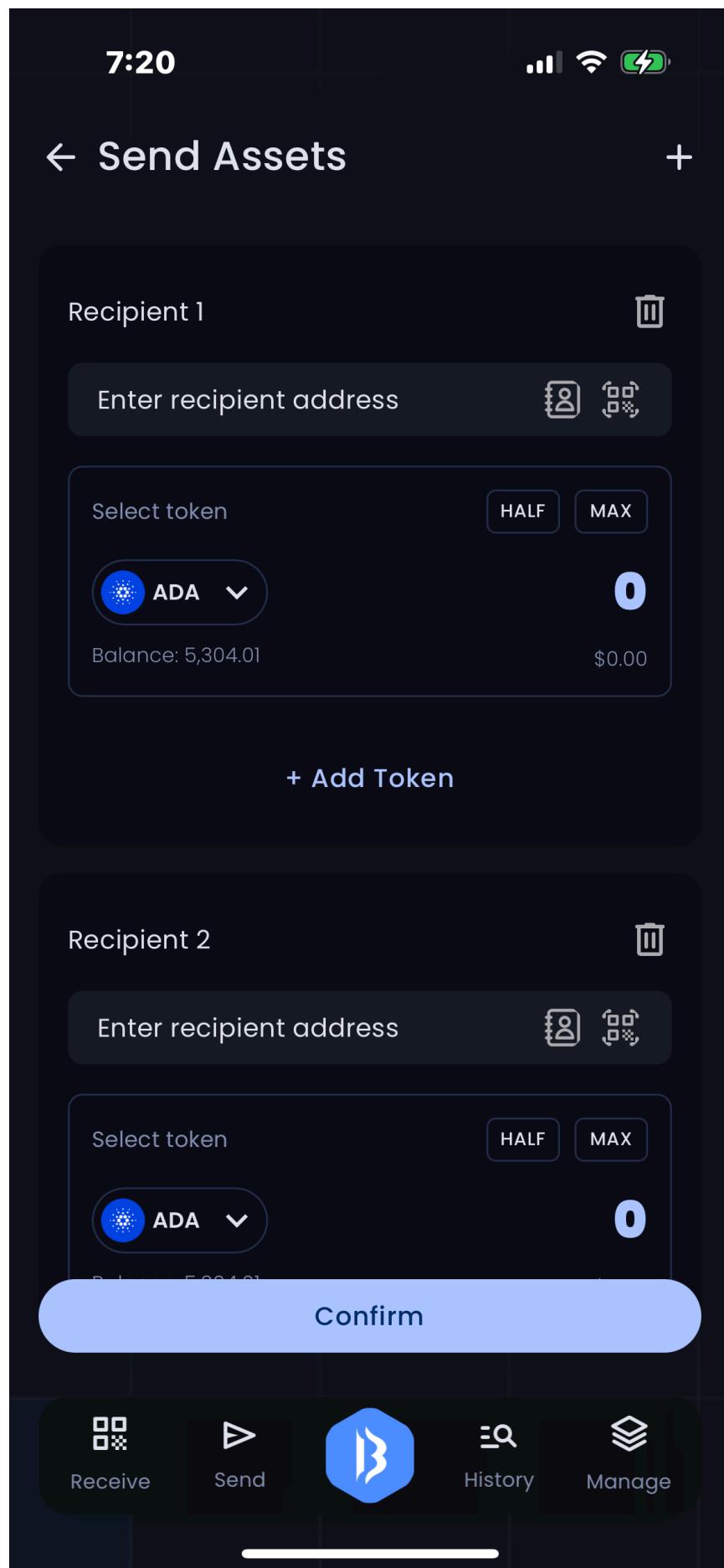


Figure 8 – Buriza UI - Mobile: Send assets with an additional recipient

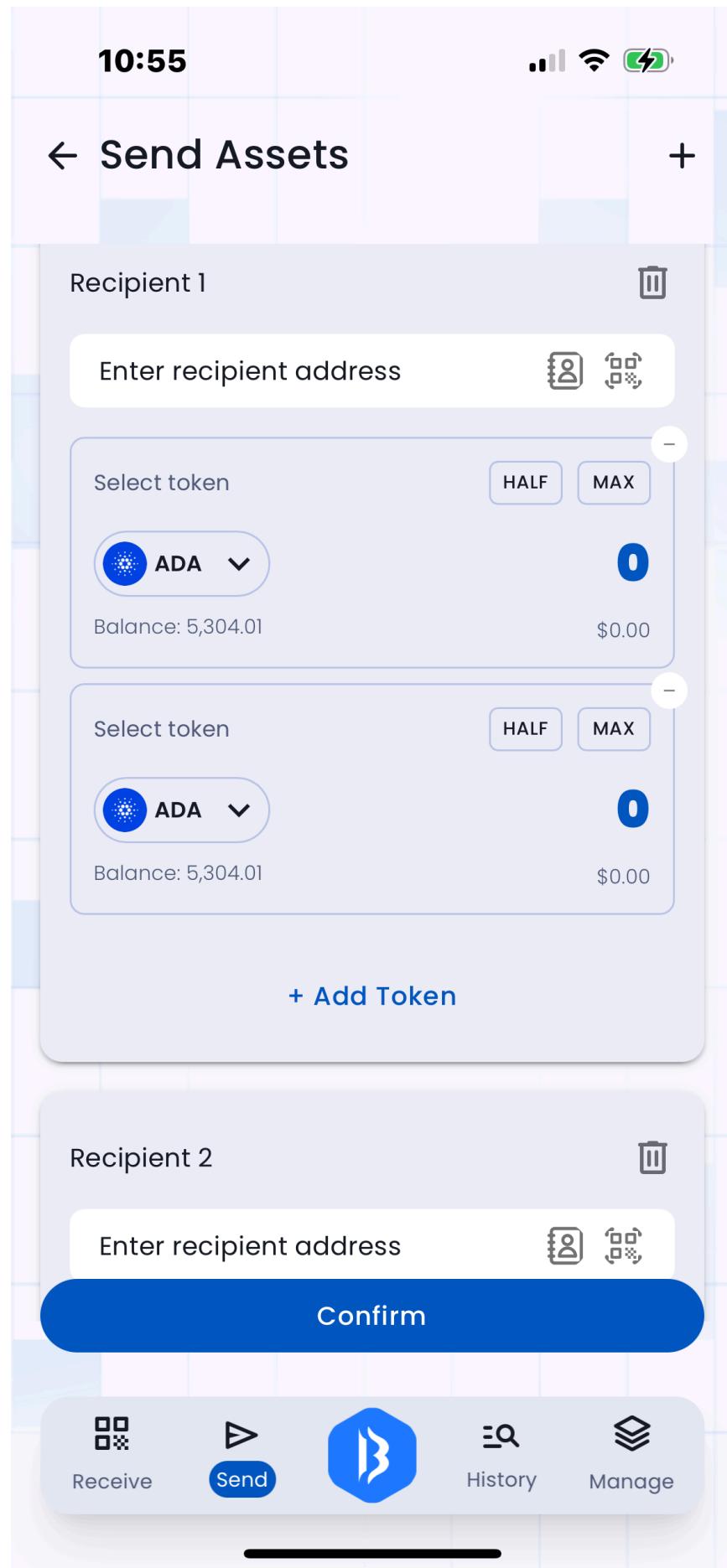


Figure 9 — Buriza UI - Mobile: Send assets with an additional token

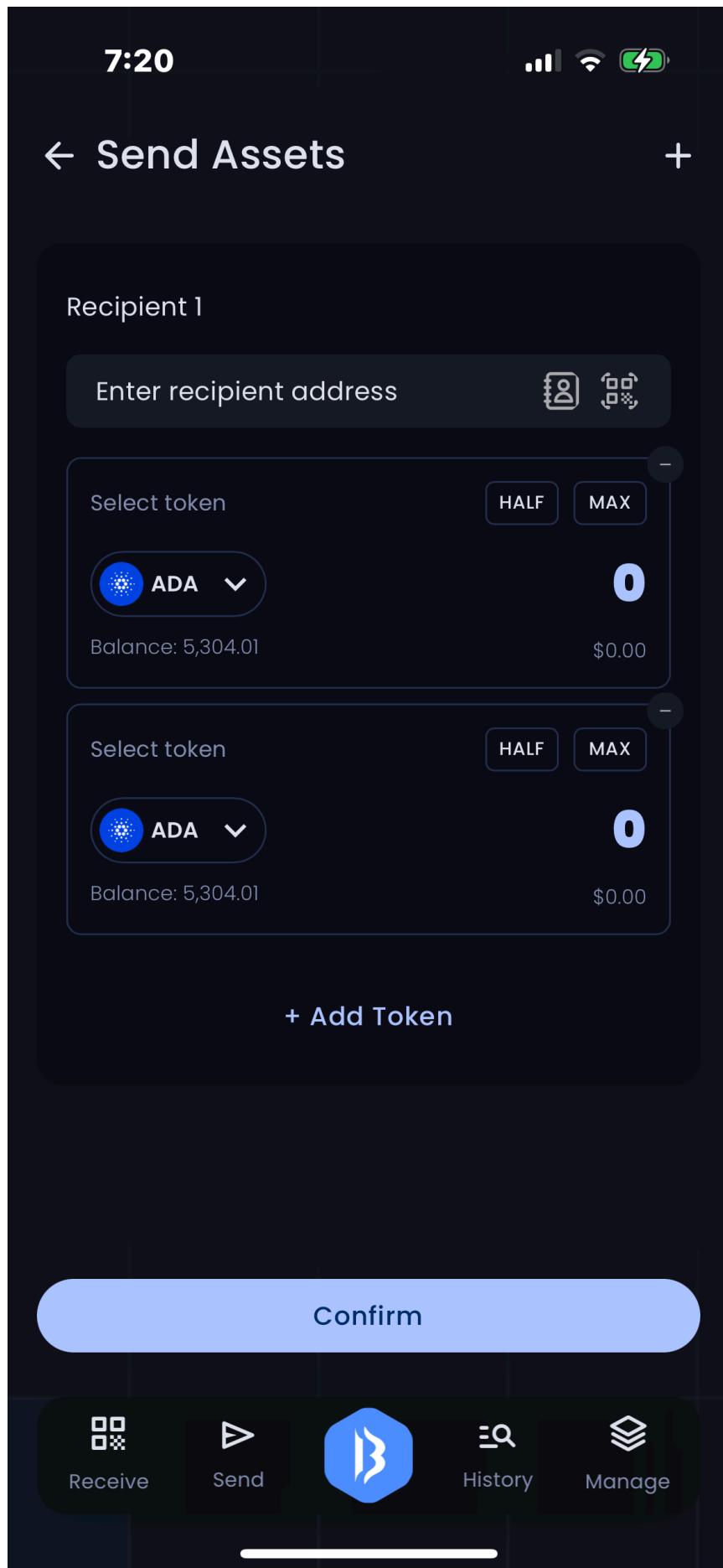


Figure 10 – Buriza UI - Mobile: Send assets with an additional token

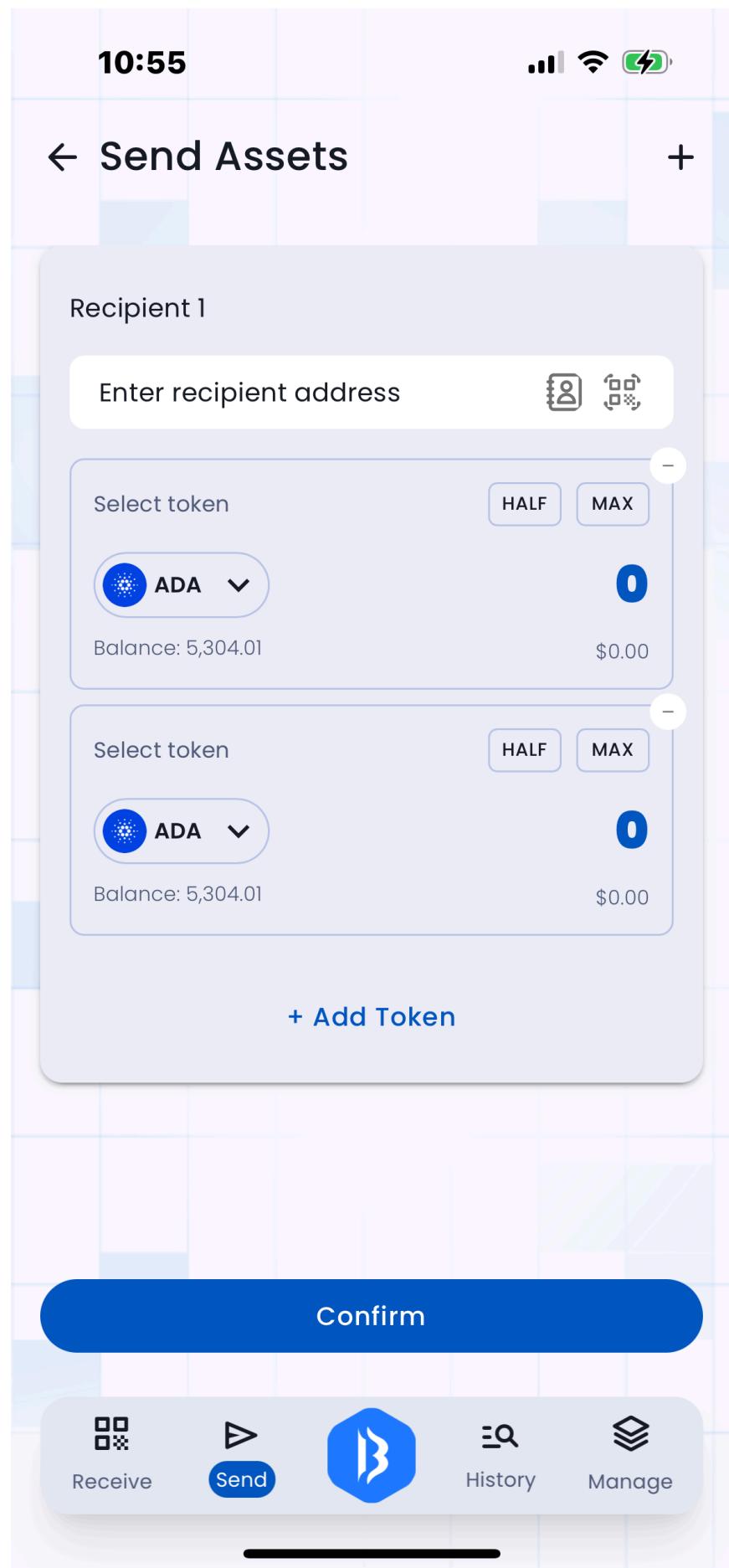


Figure 11 — Buriza UI - Mobile: Send assets transaction summary

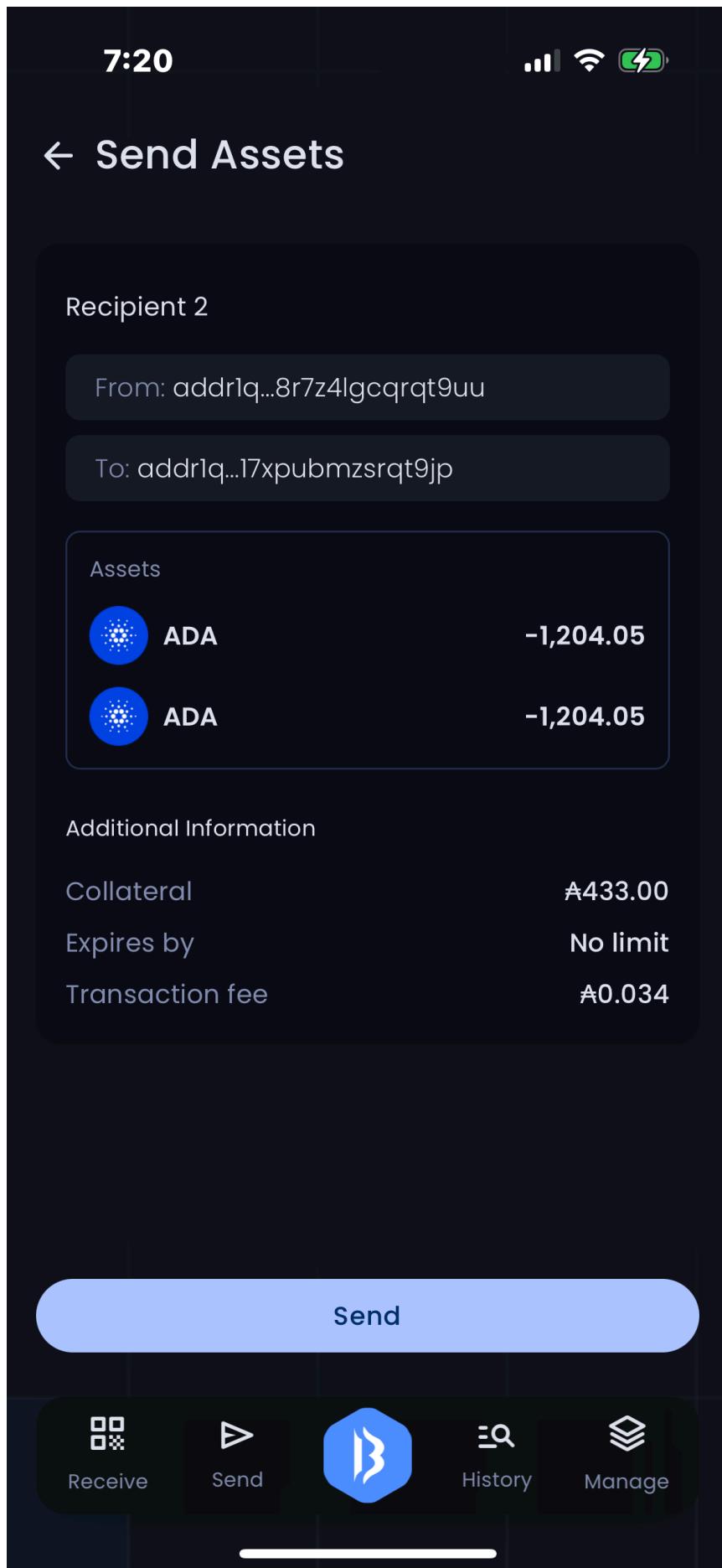


Figure 12 – Buriza UI - Mobile: Send assets transaction summary

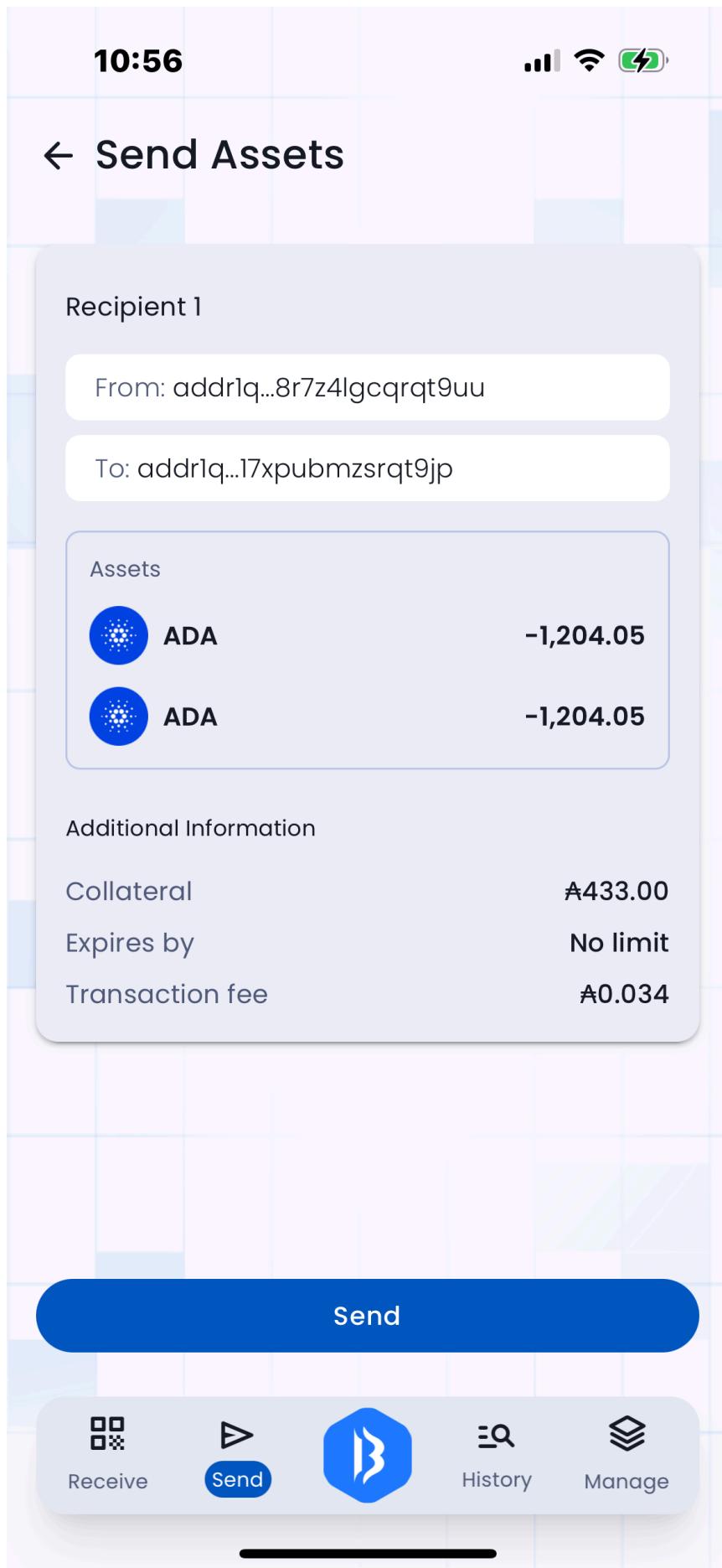


Figure 13 – Buriza UI - Mobile: Send assets transaction summary with additional recipient

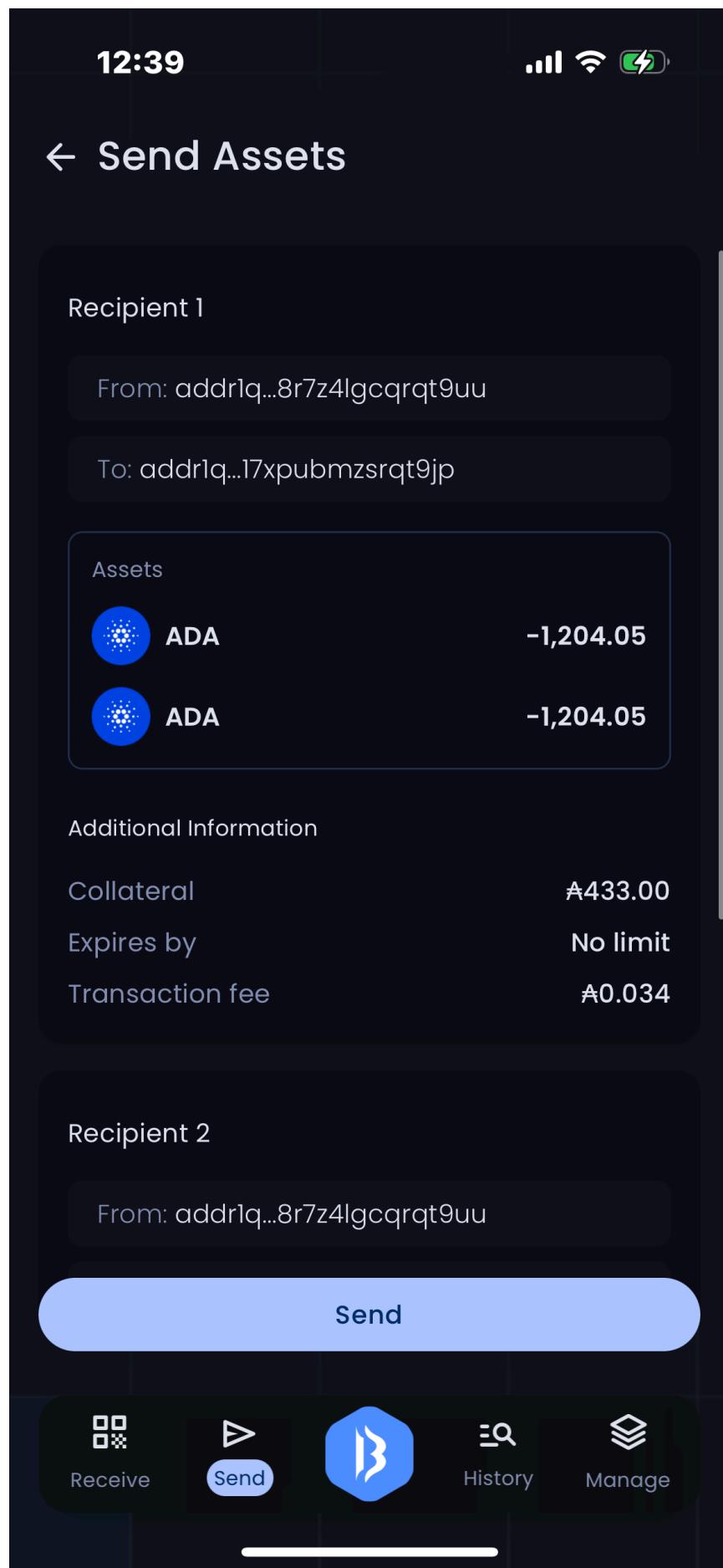


Figure 14 – Buriza UI - Mobile: Send assets transaction summary with additional recipient

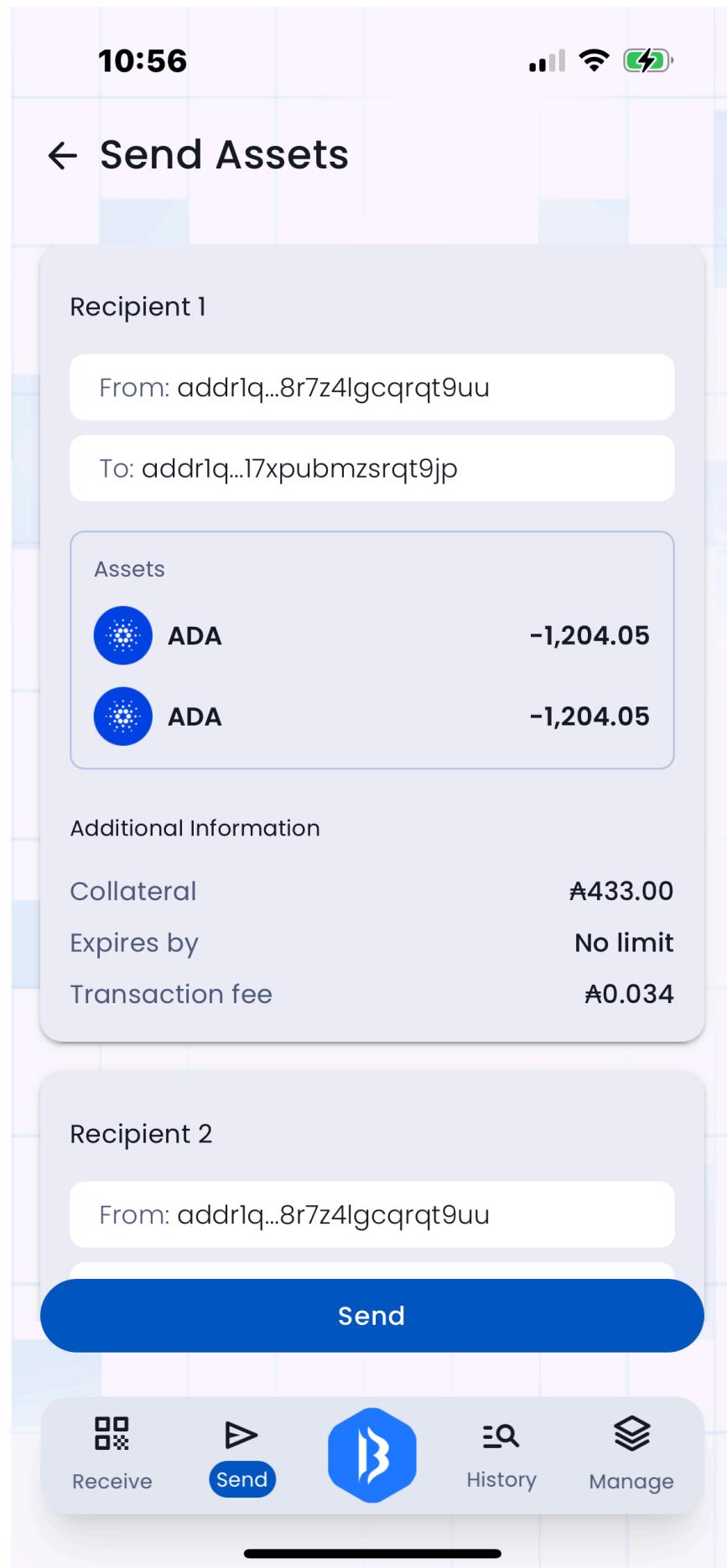


Figure 15 — Buriza UI - Mobile: Face ID

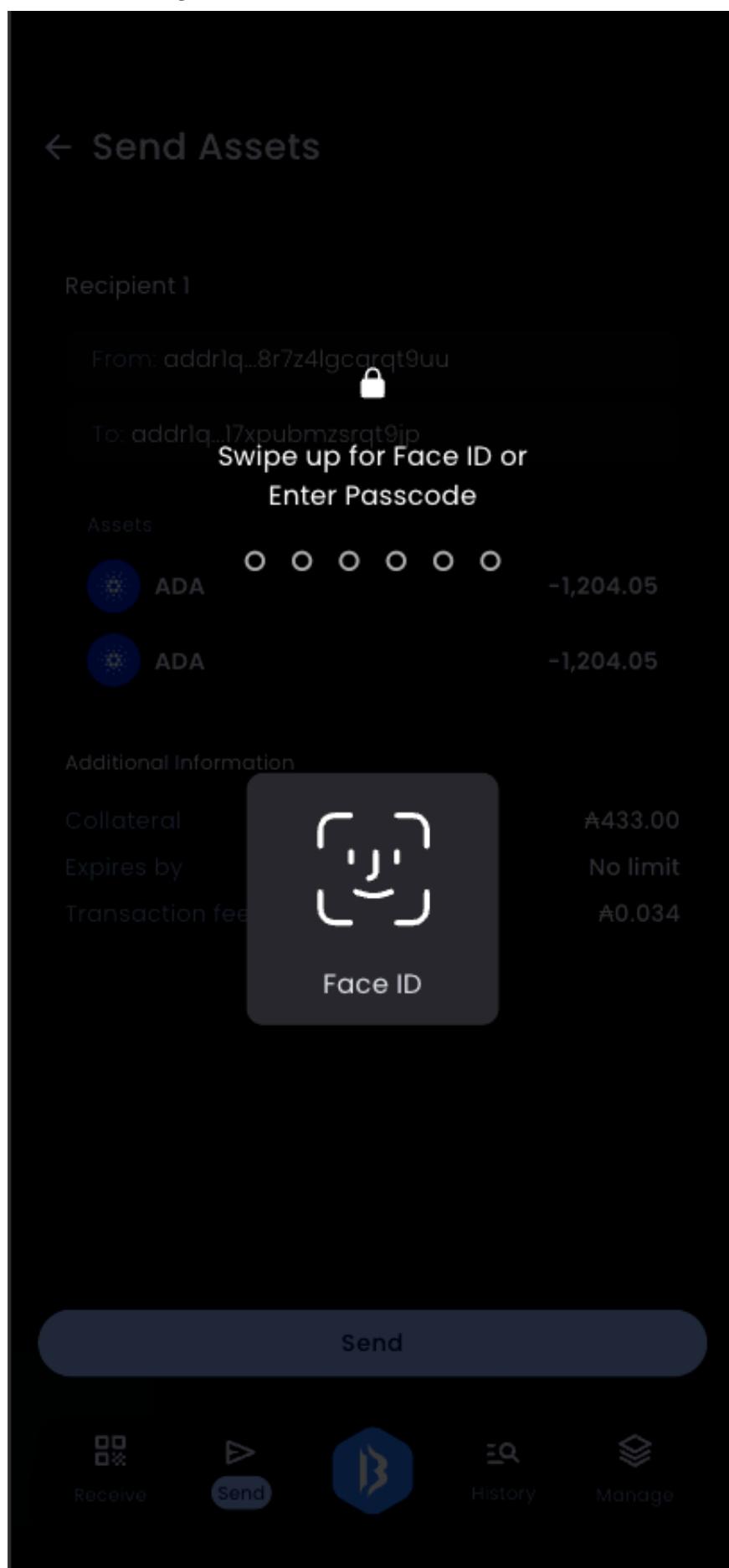


Figure 16 — Buriza UI - Mobile: Face ID

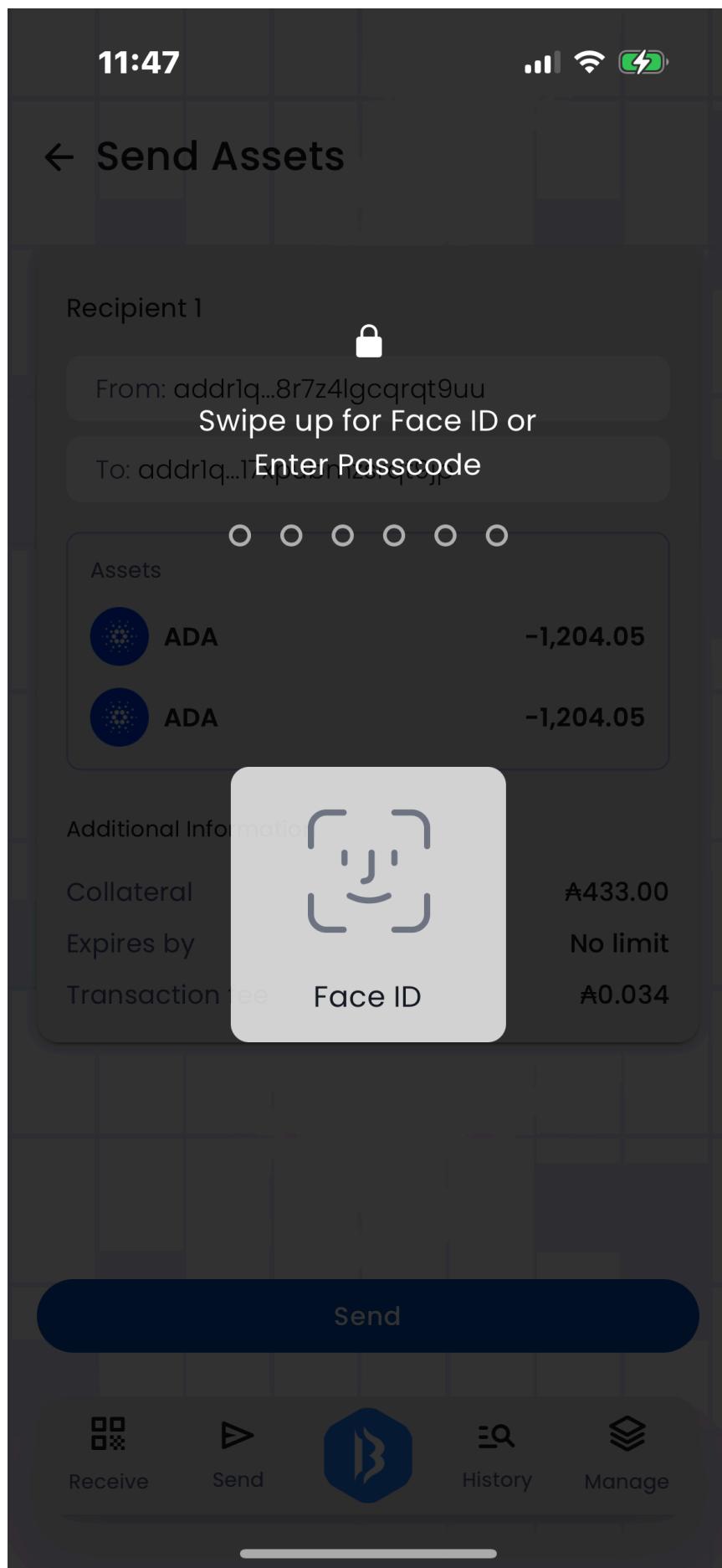


Figure 17 – Buriza UI - Mobile: Send assets success screen

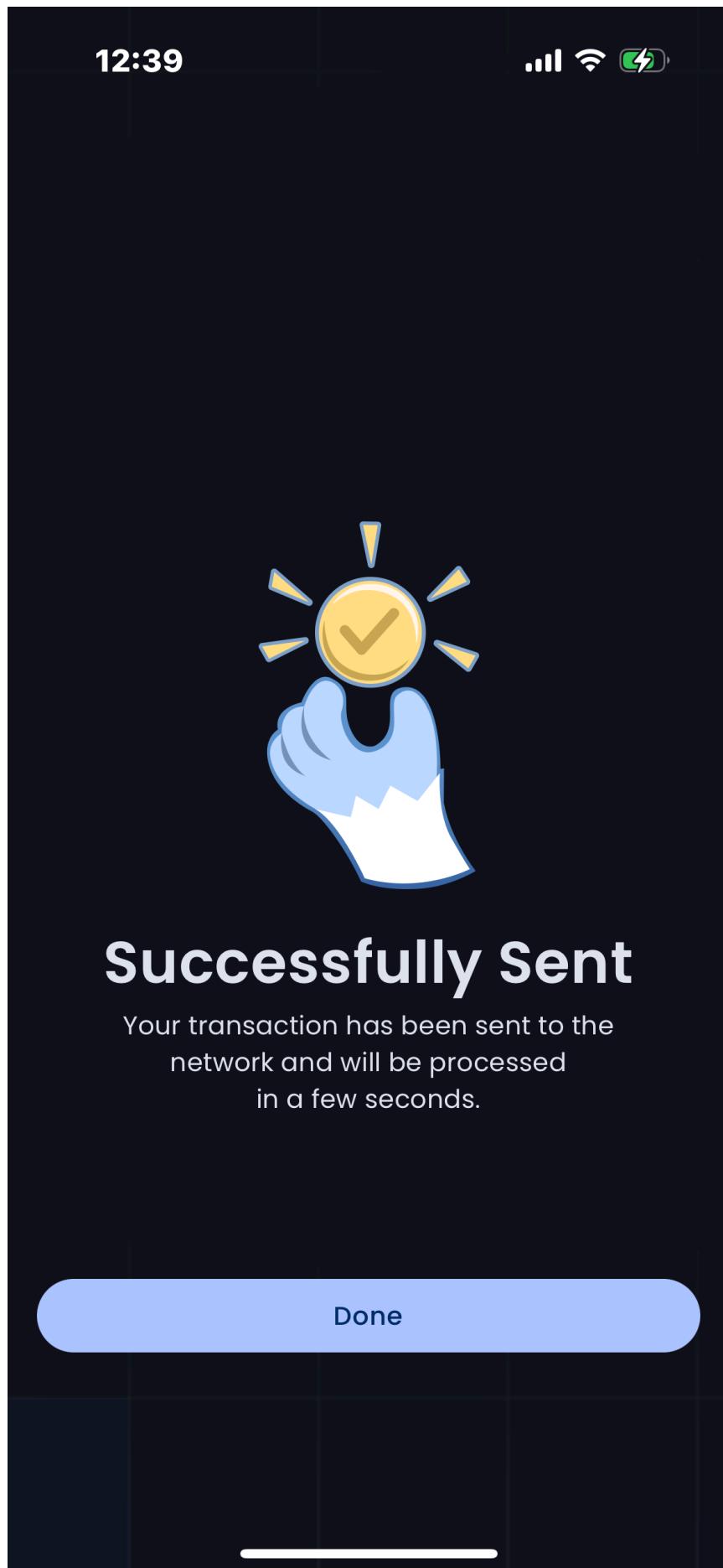
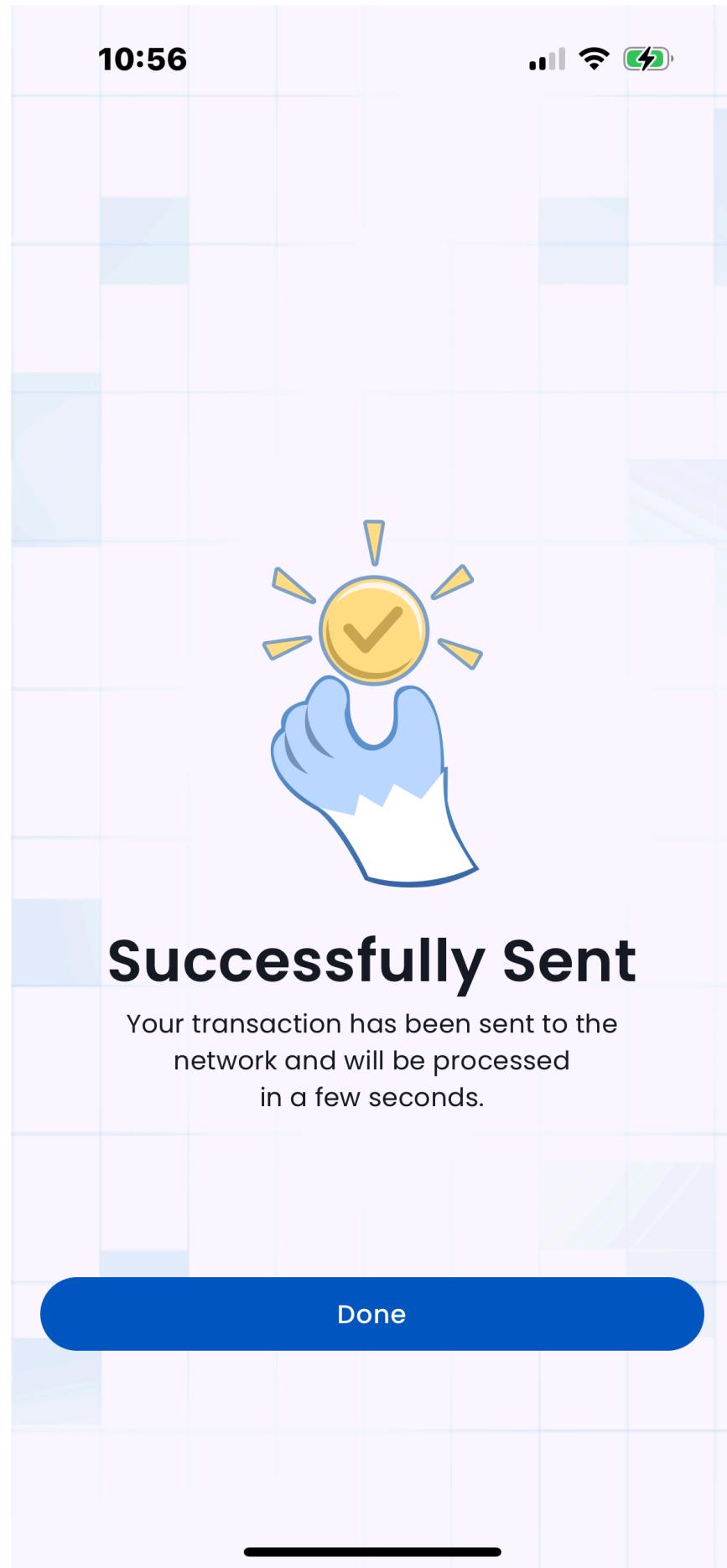


Figure 18 – Buriza UI - Mobile: Send assets success screen



iv.1.2 Receiving Assets

Figure 19 — Buriza UI - Mobile: Receive assets

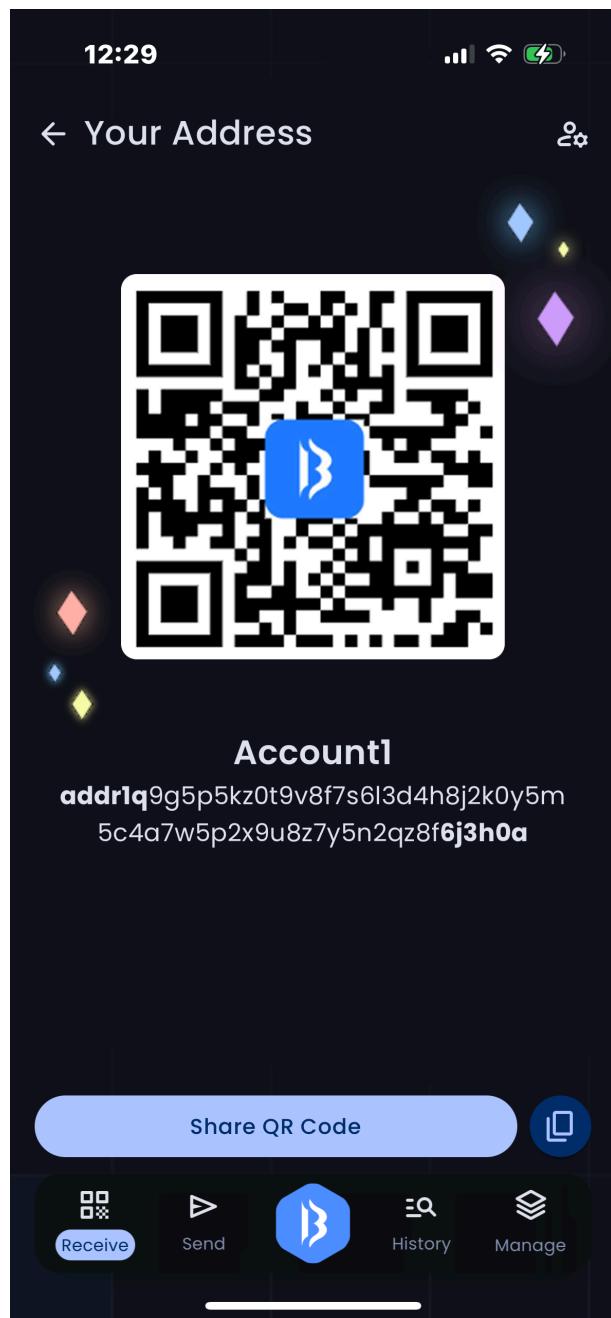


Figure 20 — Buriza UI - Mobile: Receive assets

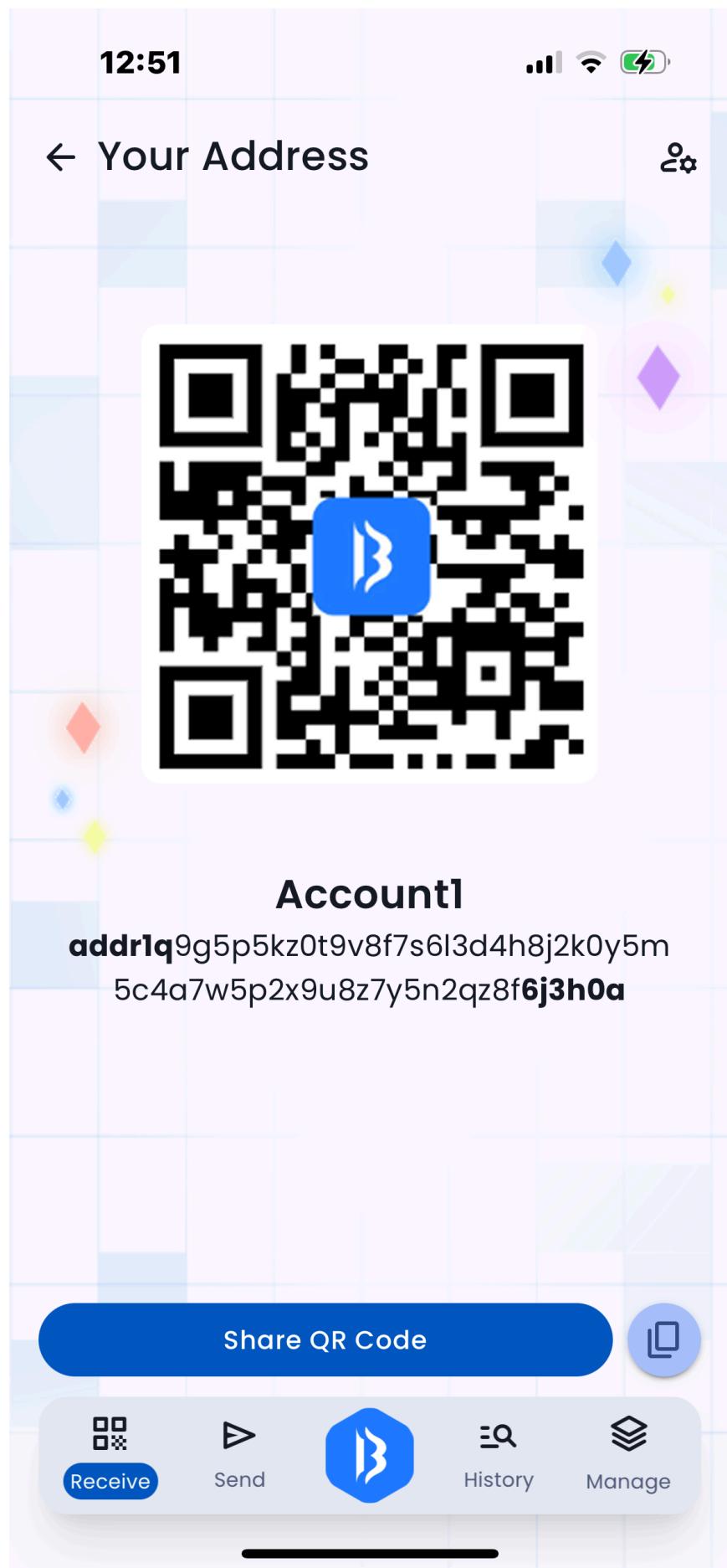


Figure 21 — Buriza UI - Mobile: Receive assets advanced mode

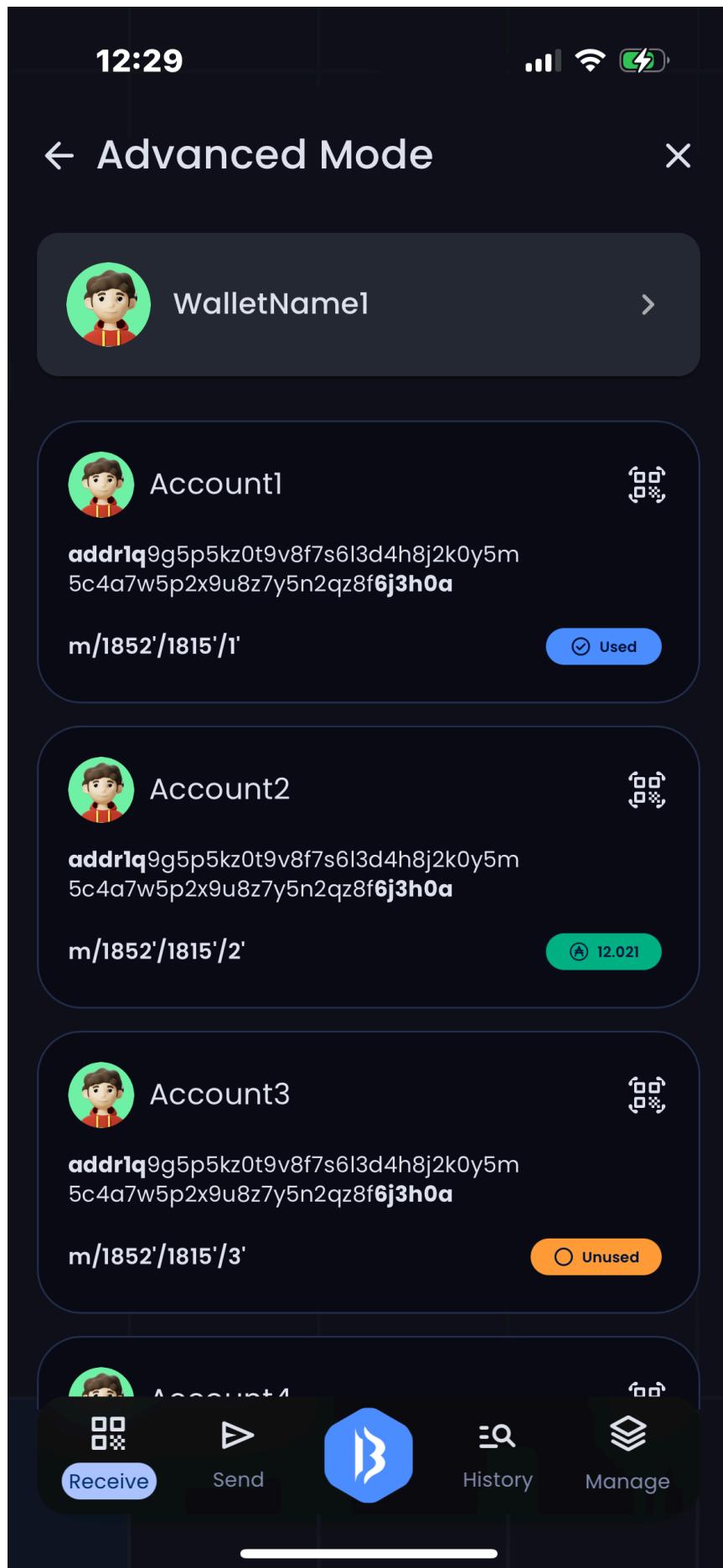
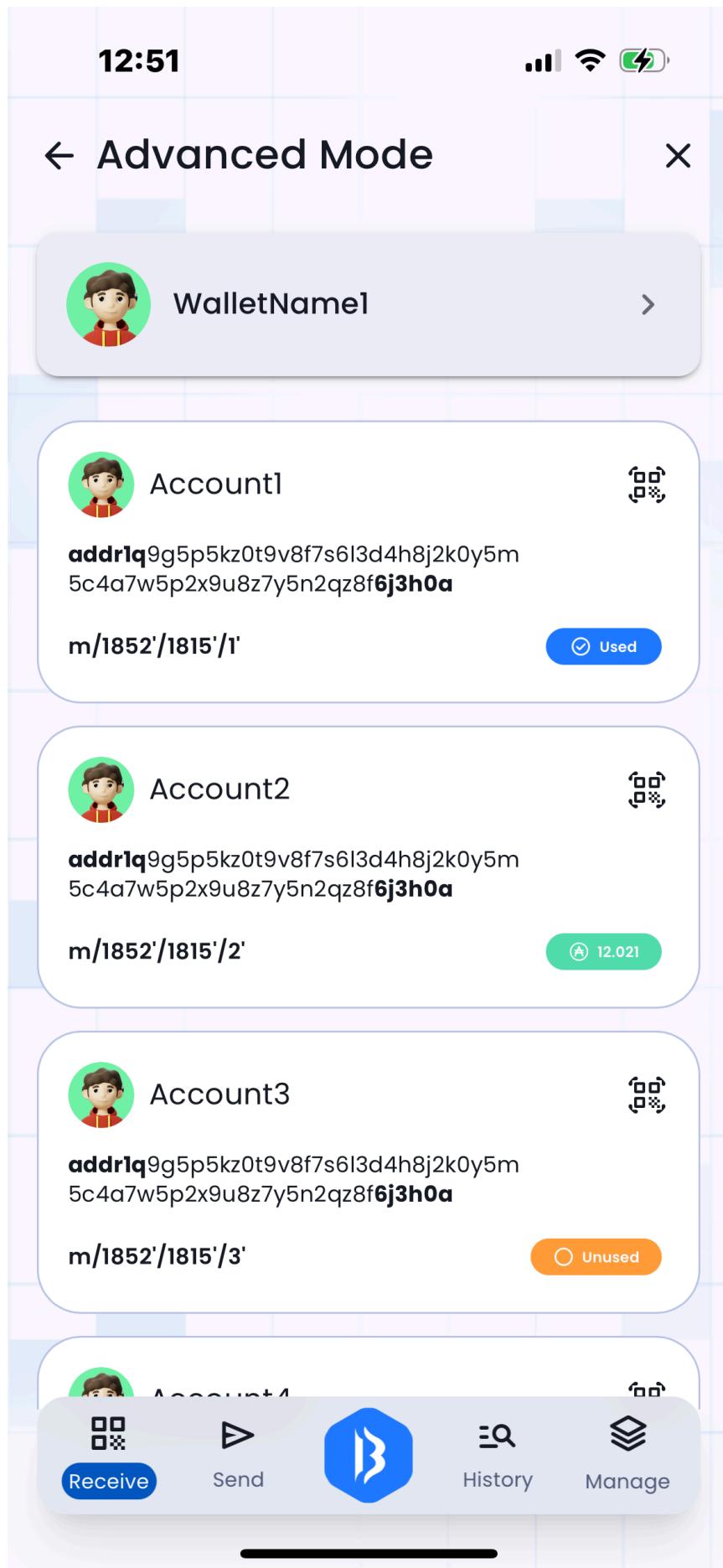


Figure 22 — Buriza UI - Mobile: Receive assets advanced mode



iv.1.3 View Balance and Assets

Figure 23 – Buriza UI - Mobile: View balance and assets



Figure 24 — Buriza UI - Mobile: View balance and assets

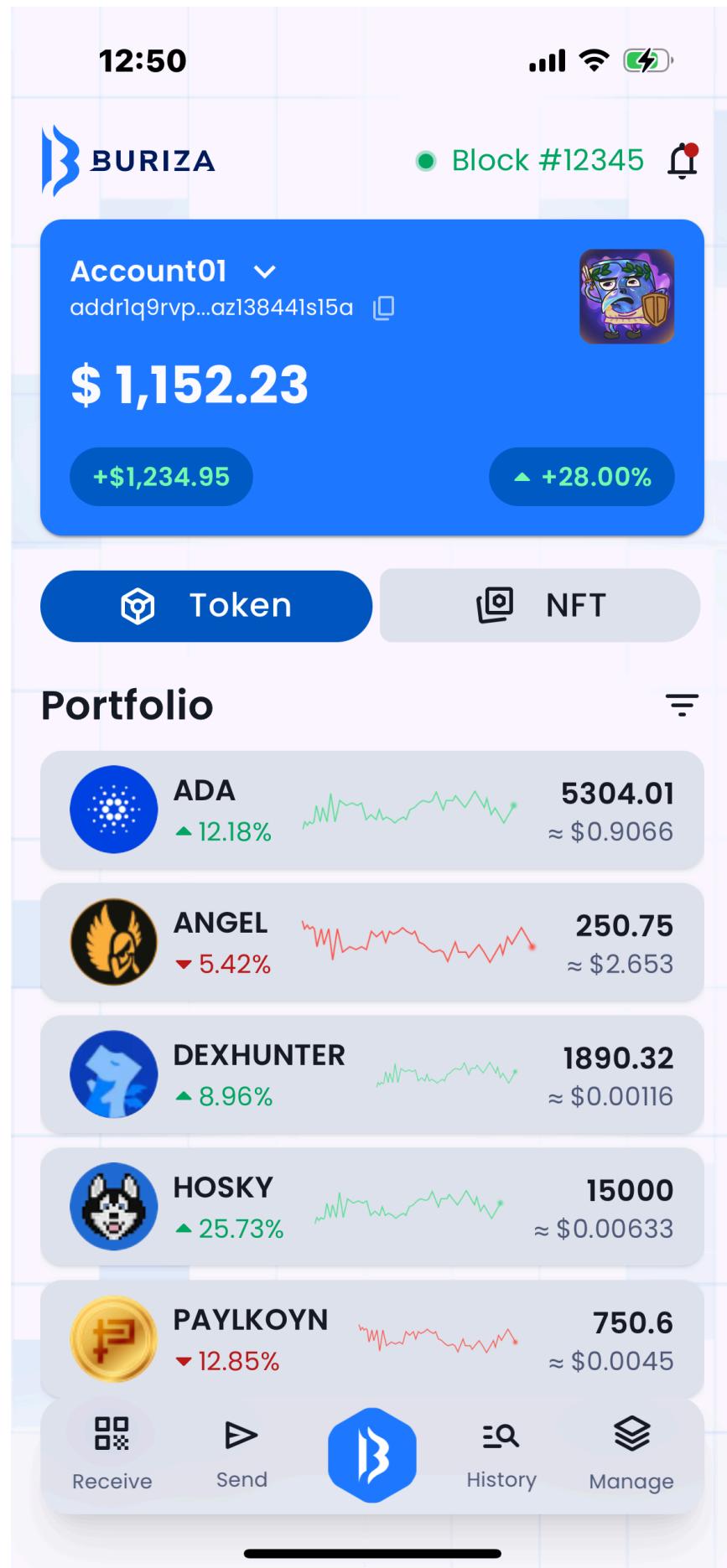


Figure 25 — Buriza UI - Mobile: Header animation

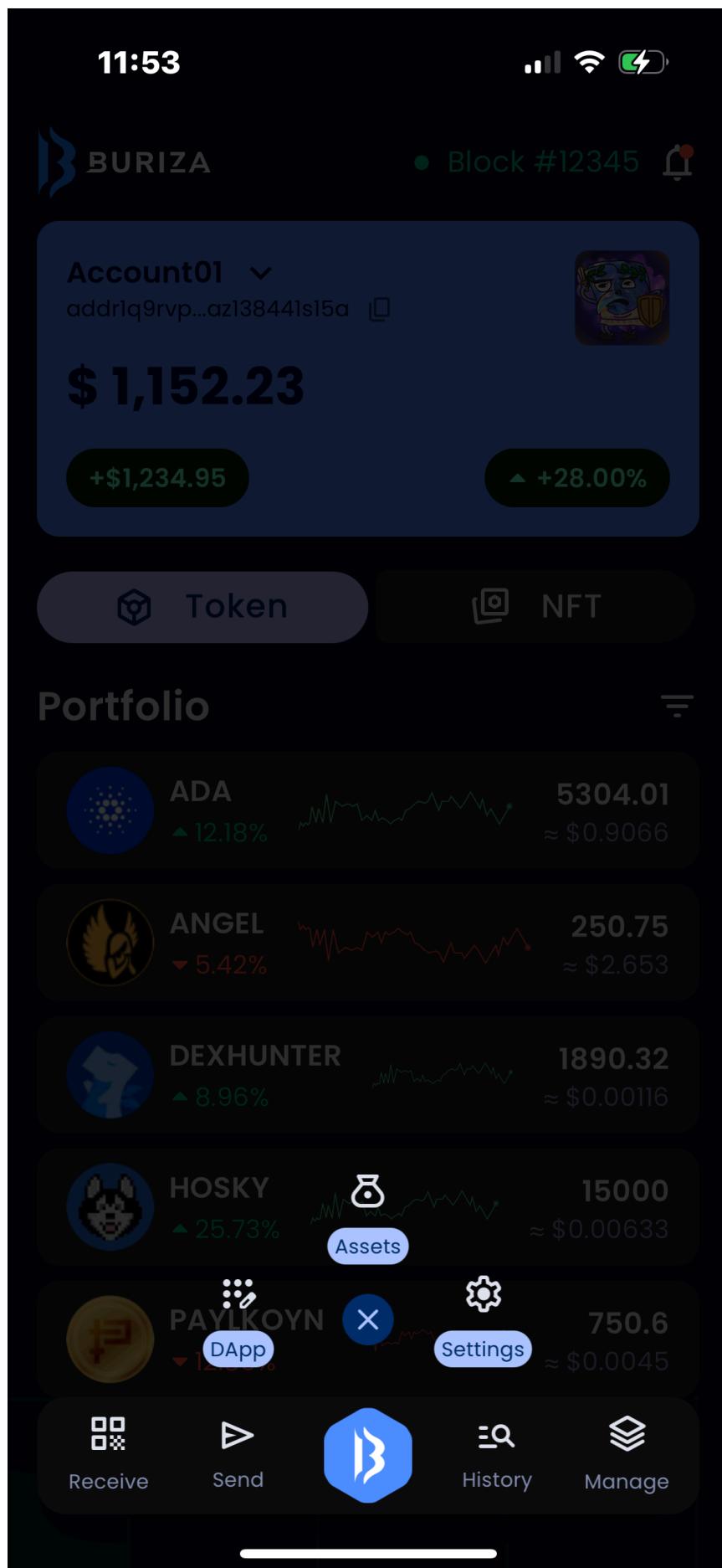
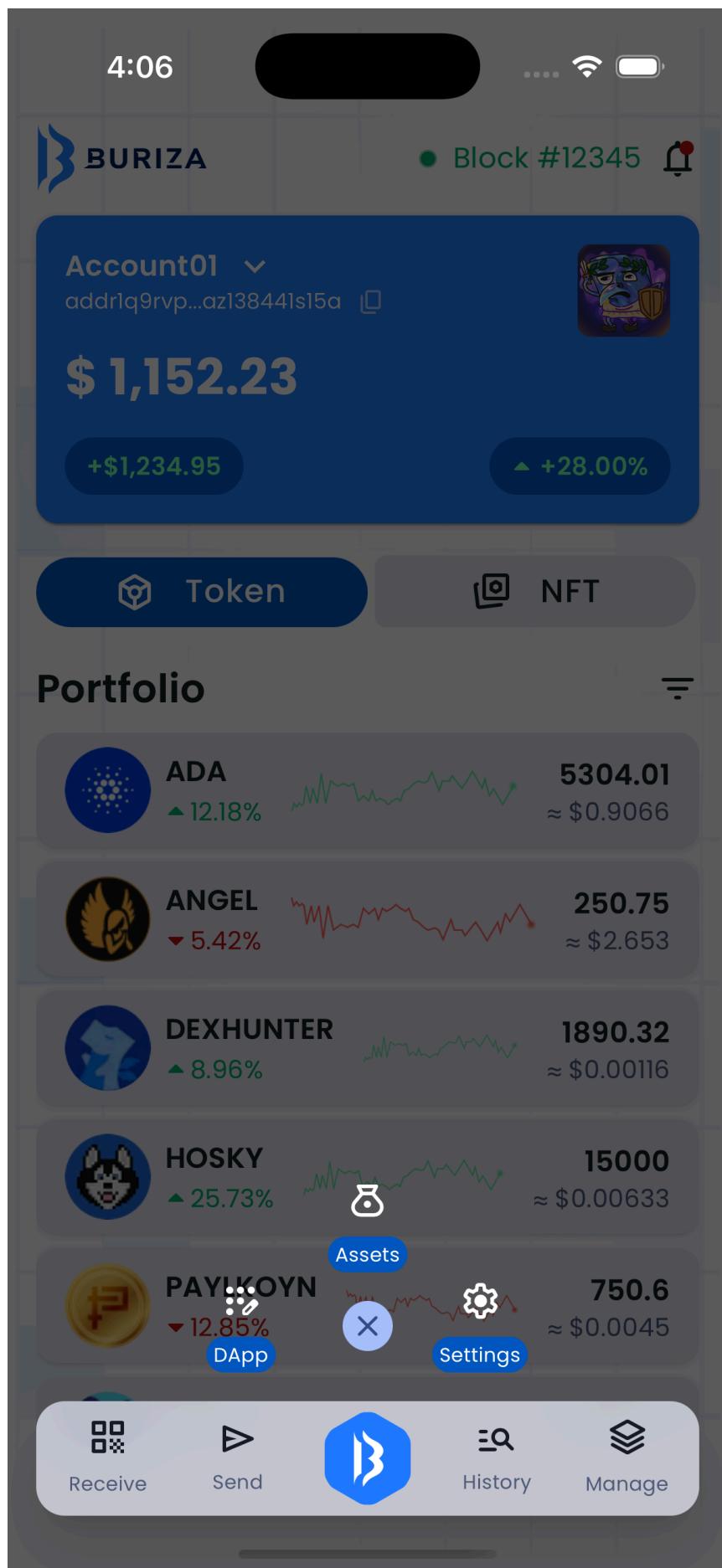


Figure 26 — Buriza UI - Mobile: Header animation



iv.1.4 Transaction History

Figure 27 – Buriza UI - Mobile: Transaction history

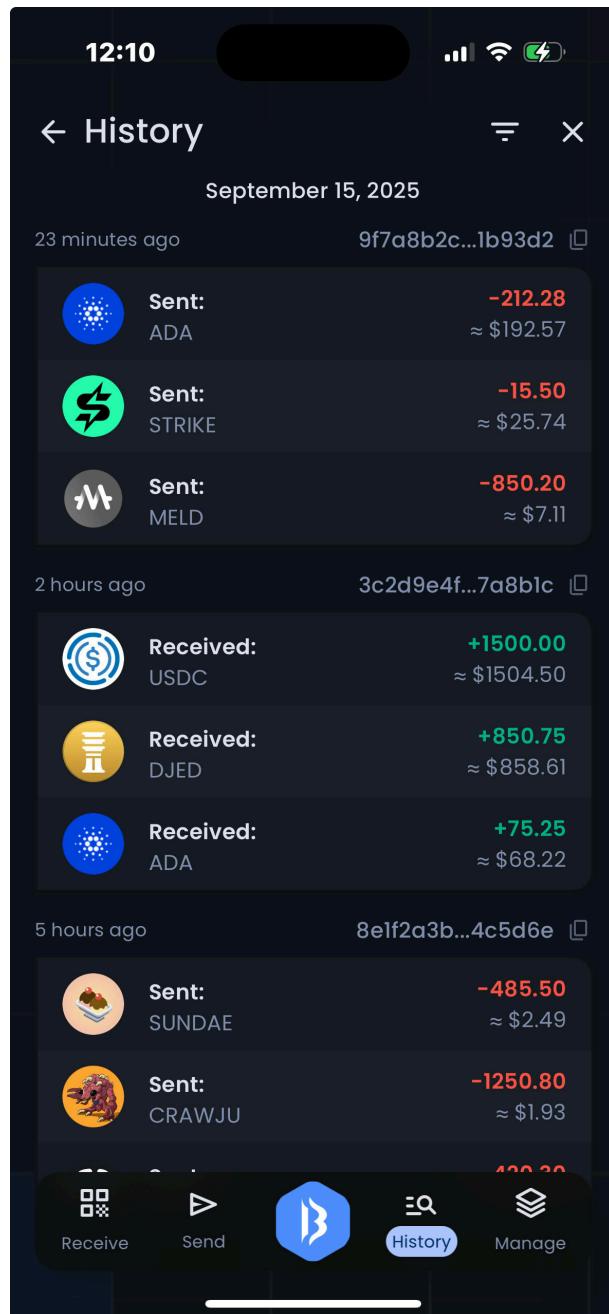


Figure 28 – Buriza UI - Mobile: Transaction history

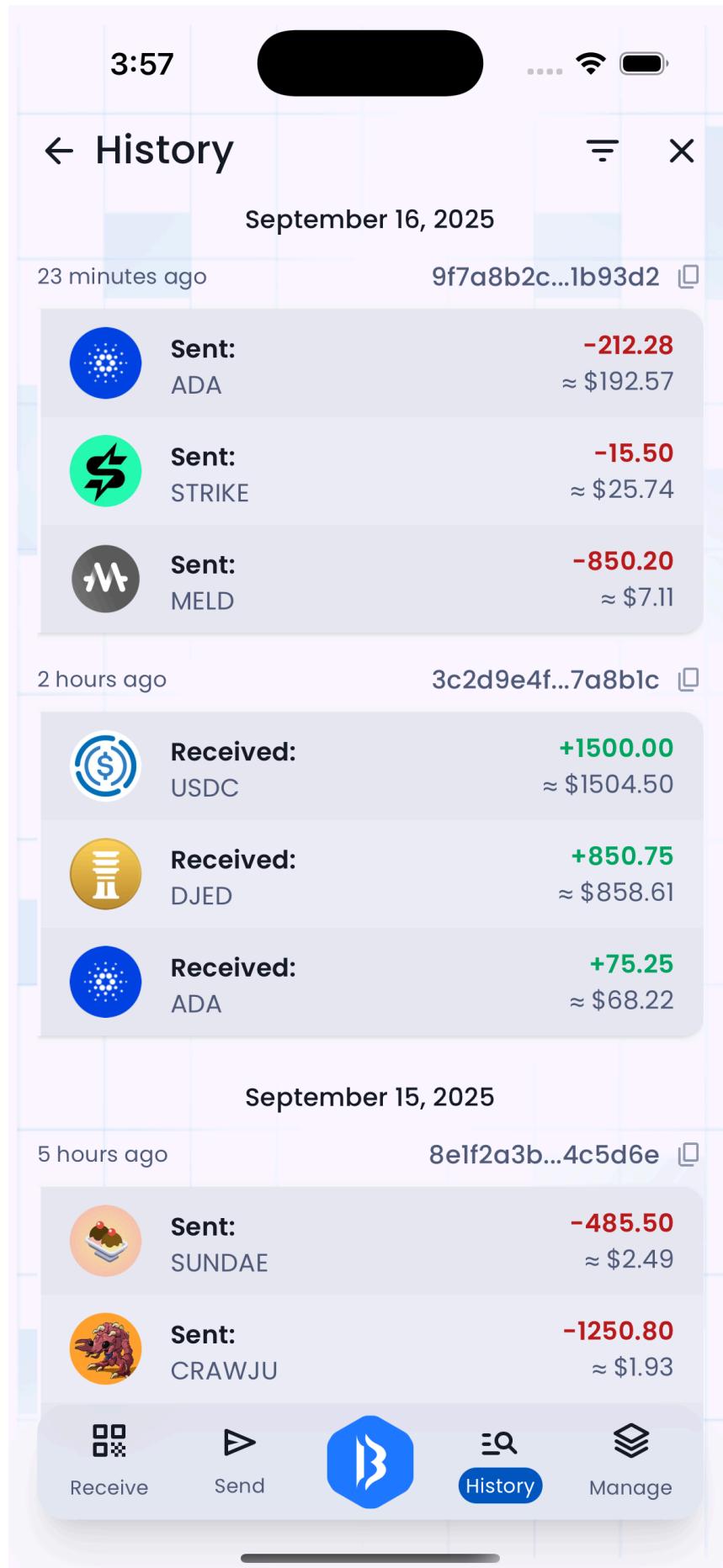


Figure 29 — Buriza UI - Mobile: Transaction summary

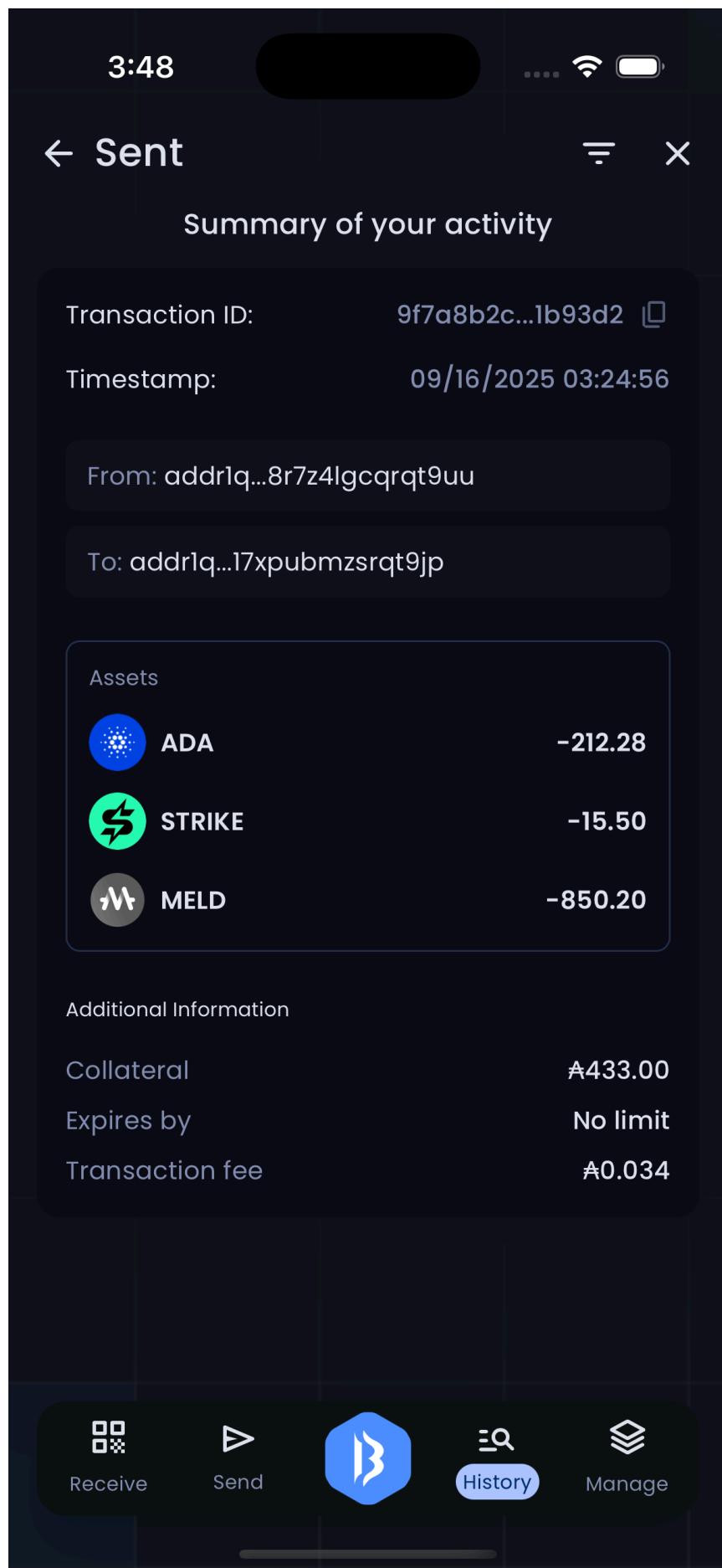
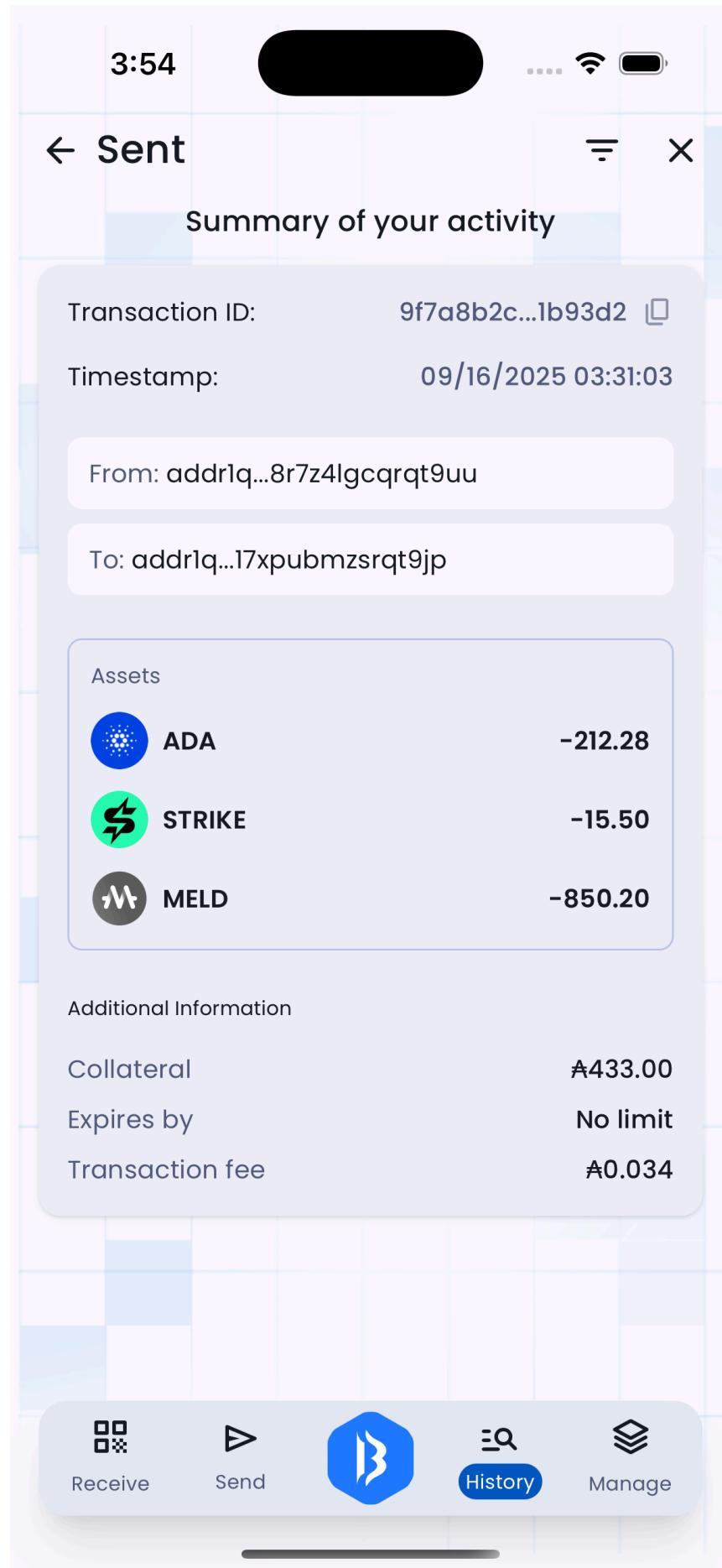


Figure 30 — Buriza UI - Mobile: Transaction summary



iv.1.5 Full Node Setup

Figure 31 – Buriza UI - Mobile: Settings

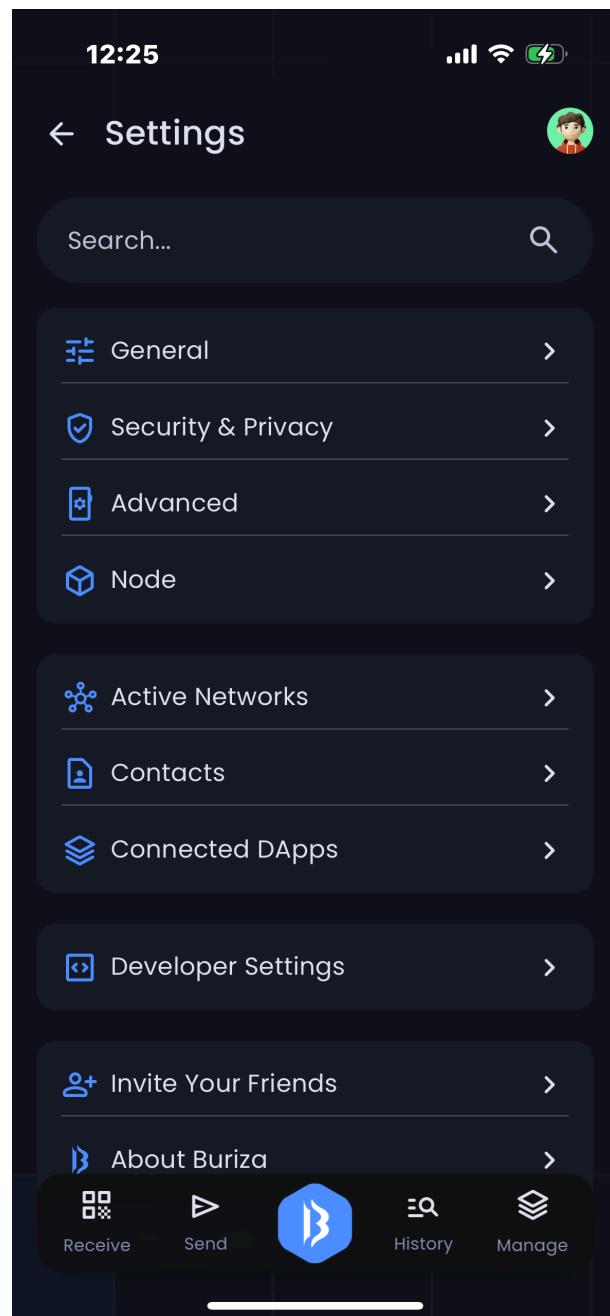


Figure 32 – Buriza UI - Mobile: Settings

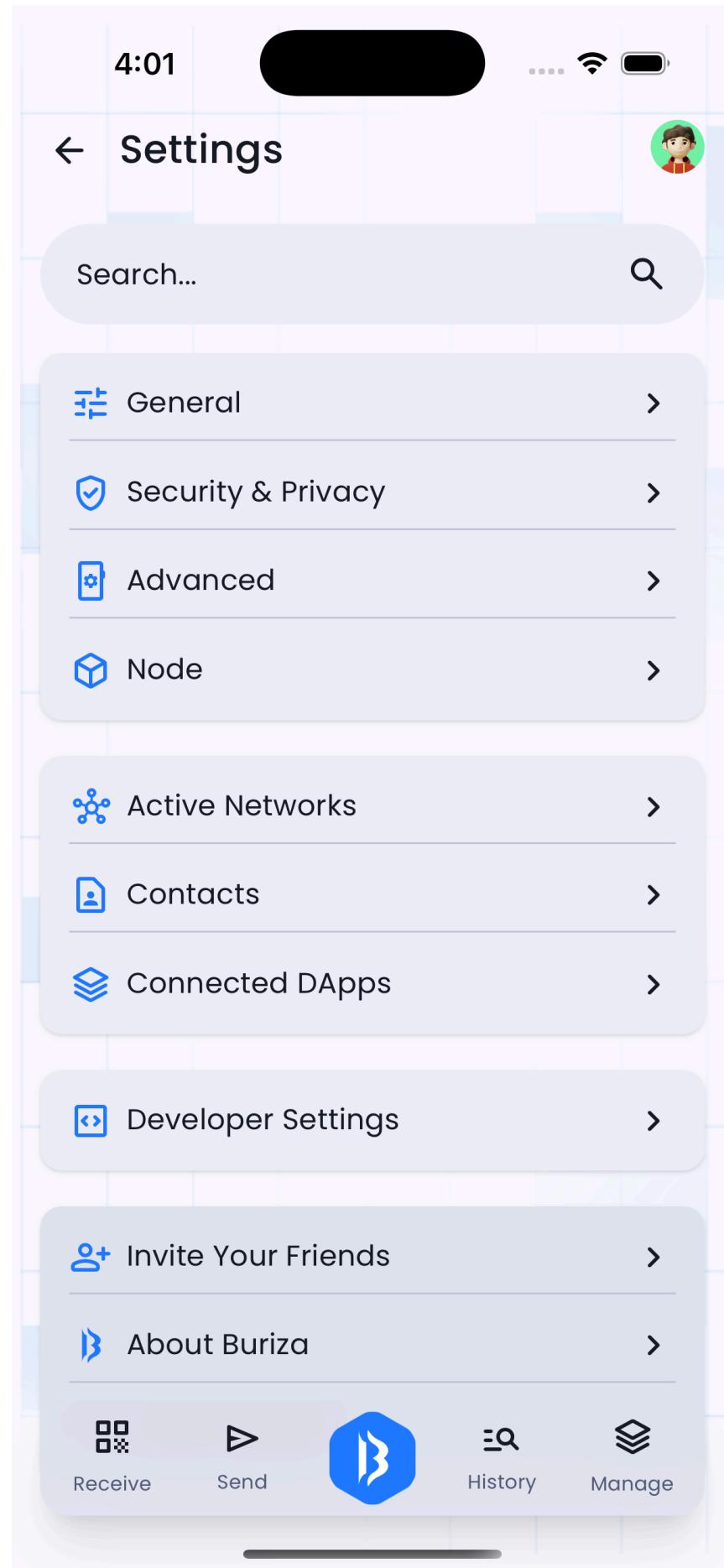


Figure 33 — Buriza UI - Mobile: Node setup

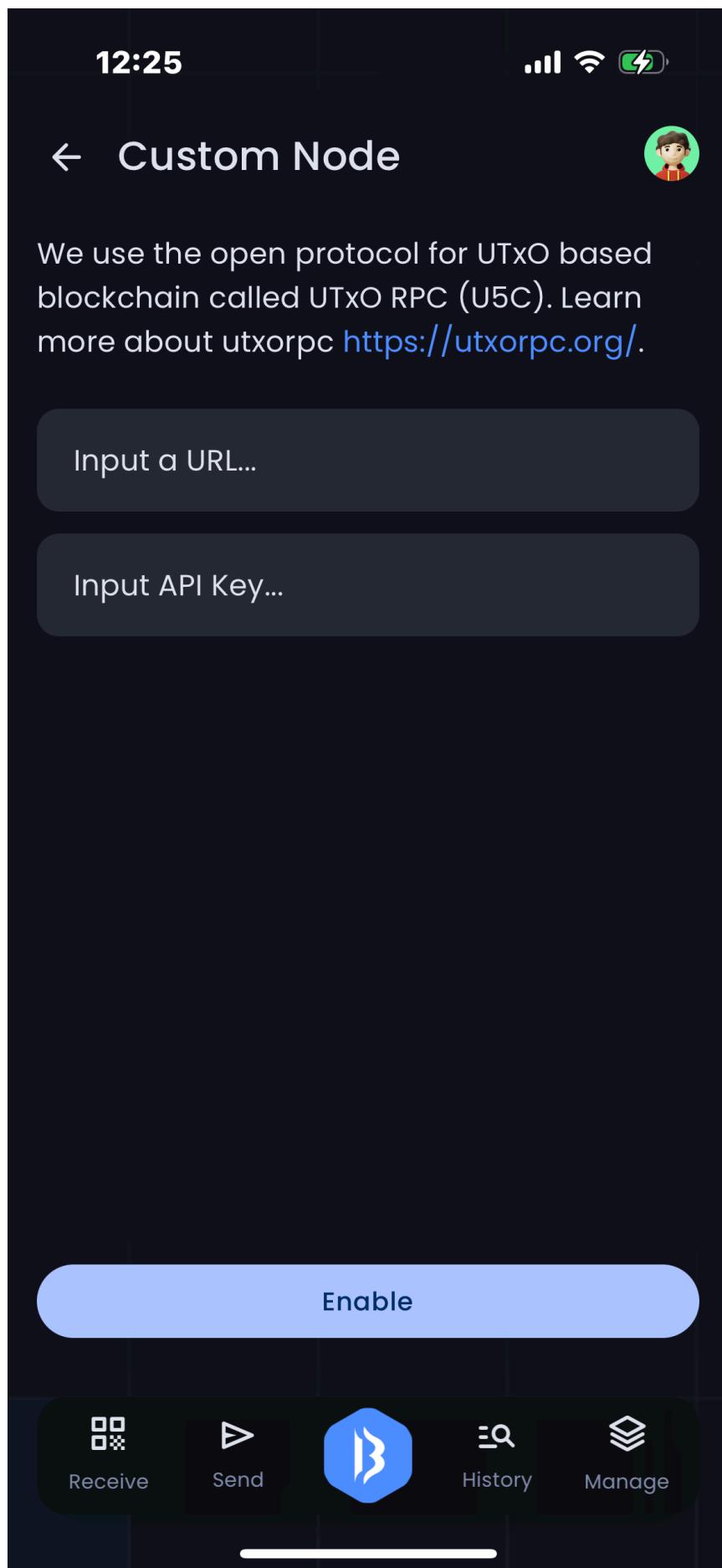
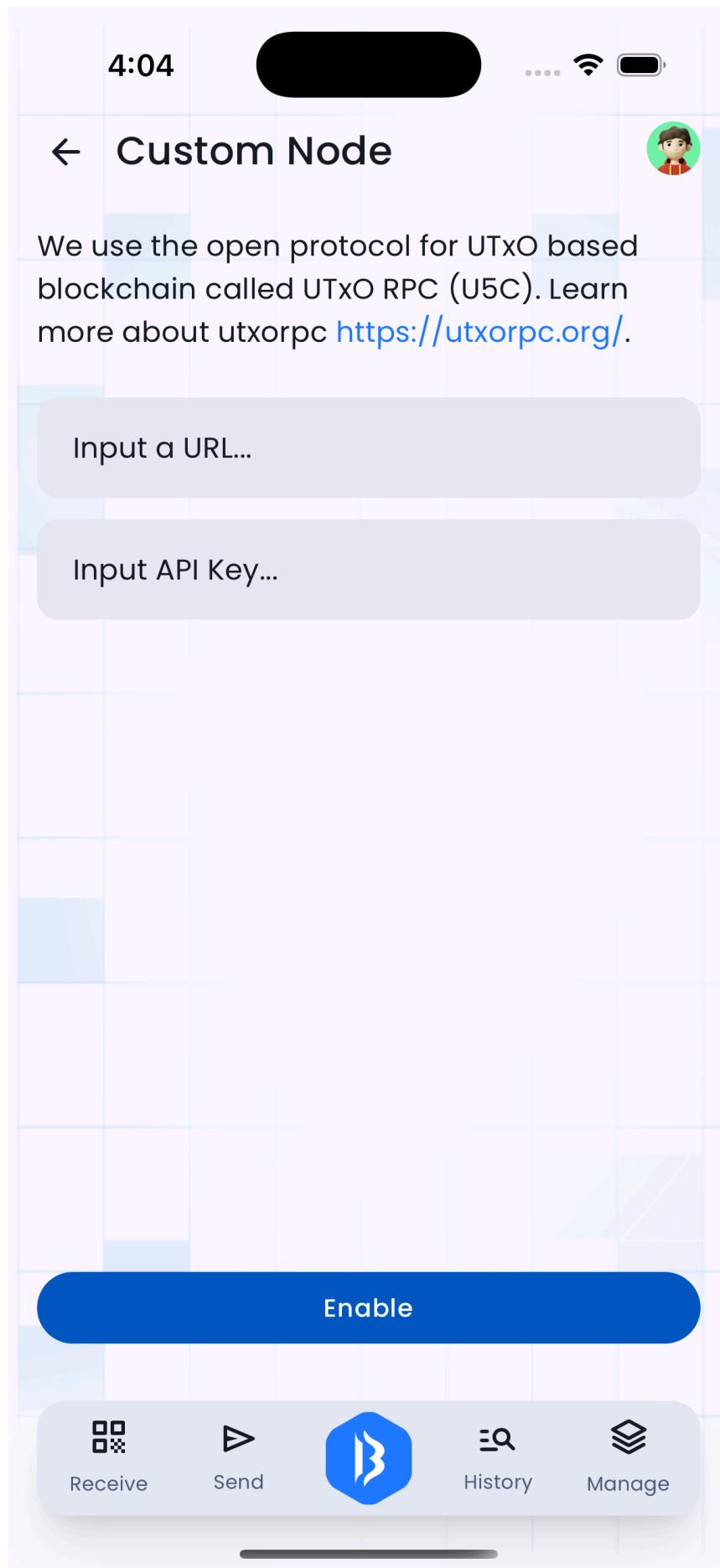


Figure 34 — Buriza UI - Mobile: Node setup



iv.1.6 dApp Access

Figure 35 – Buriza UI - Mobile: dApp explorer

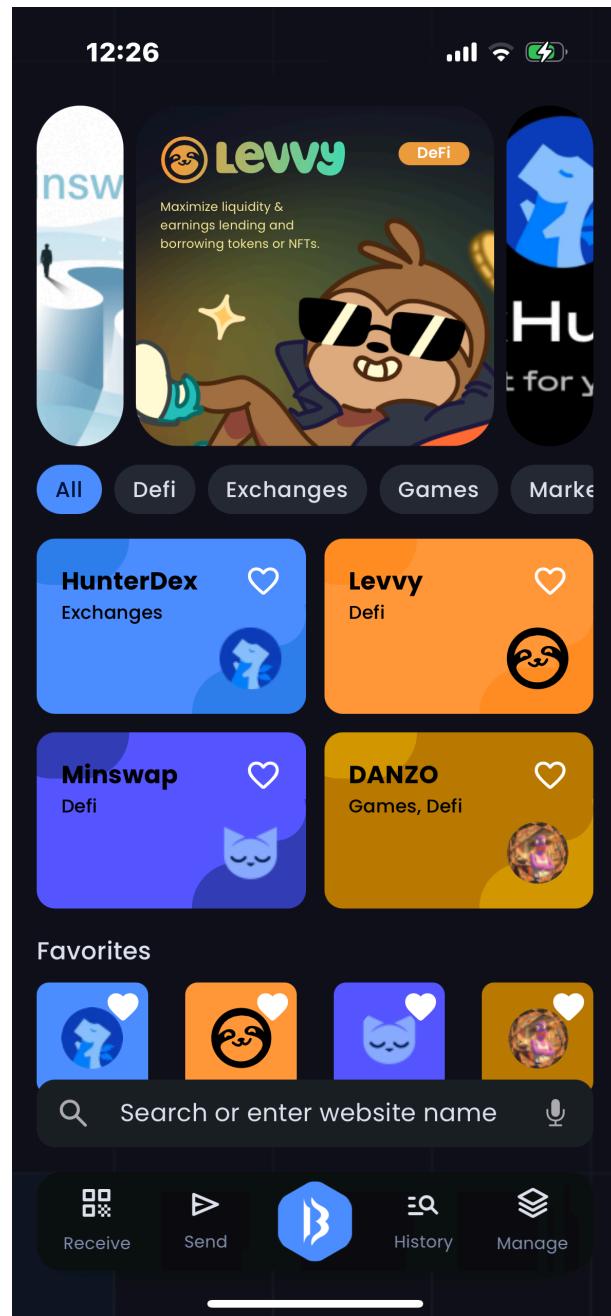


Figure 36 – Buriza UI - Mobile: dApp explorer

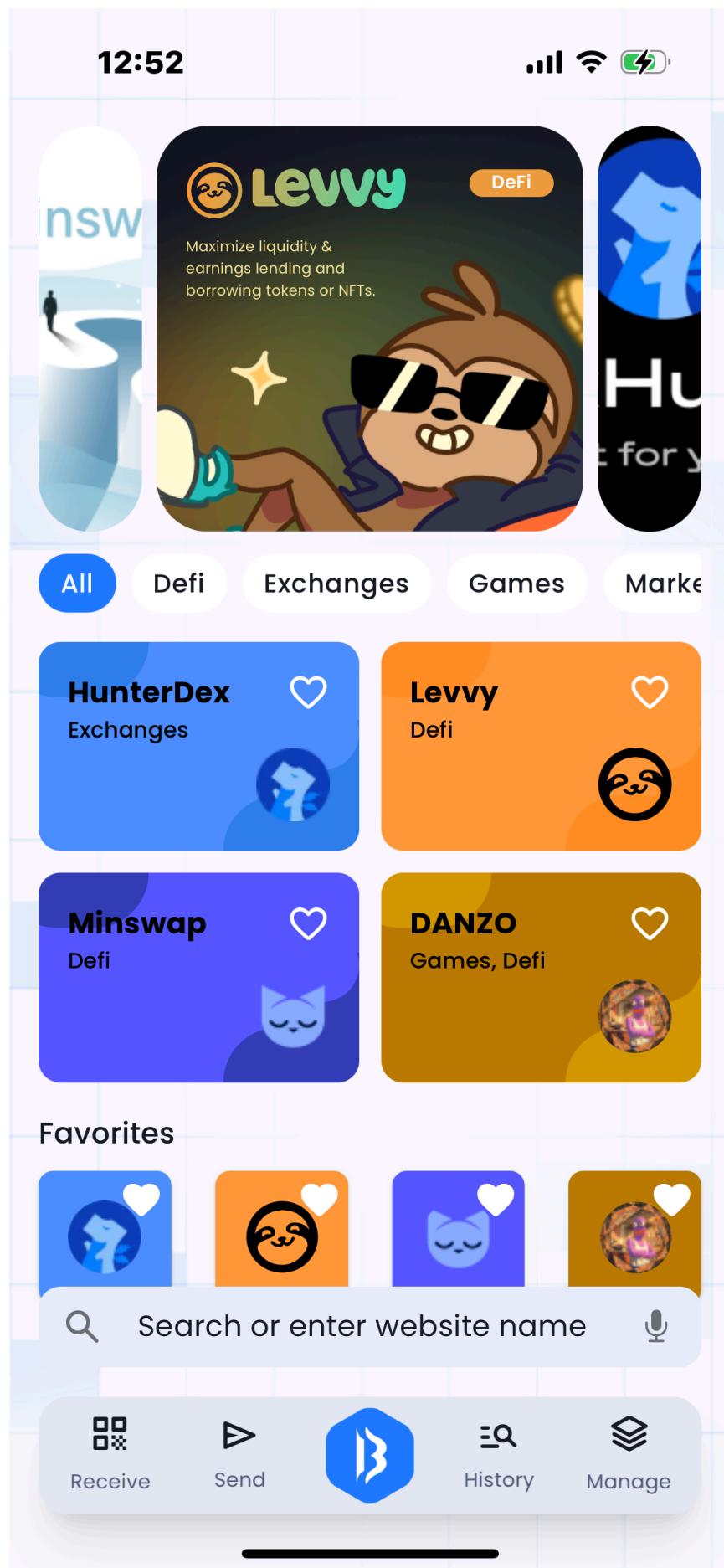


Figure 37 – Buriza UI - Mobile: dApp explorer authorization request

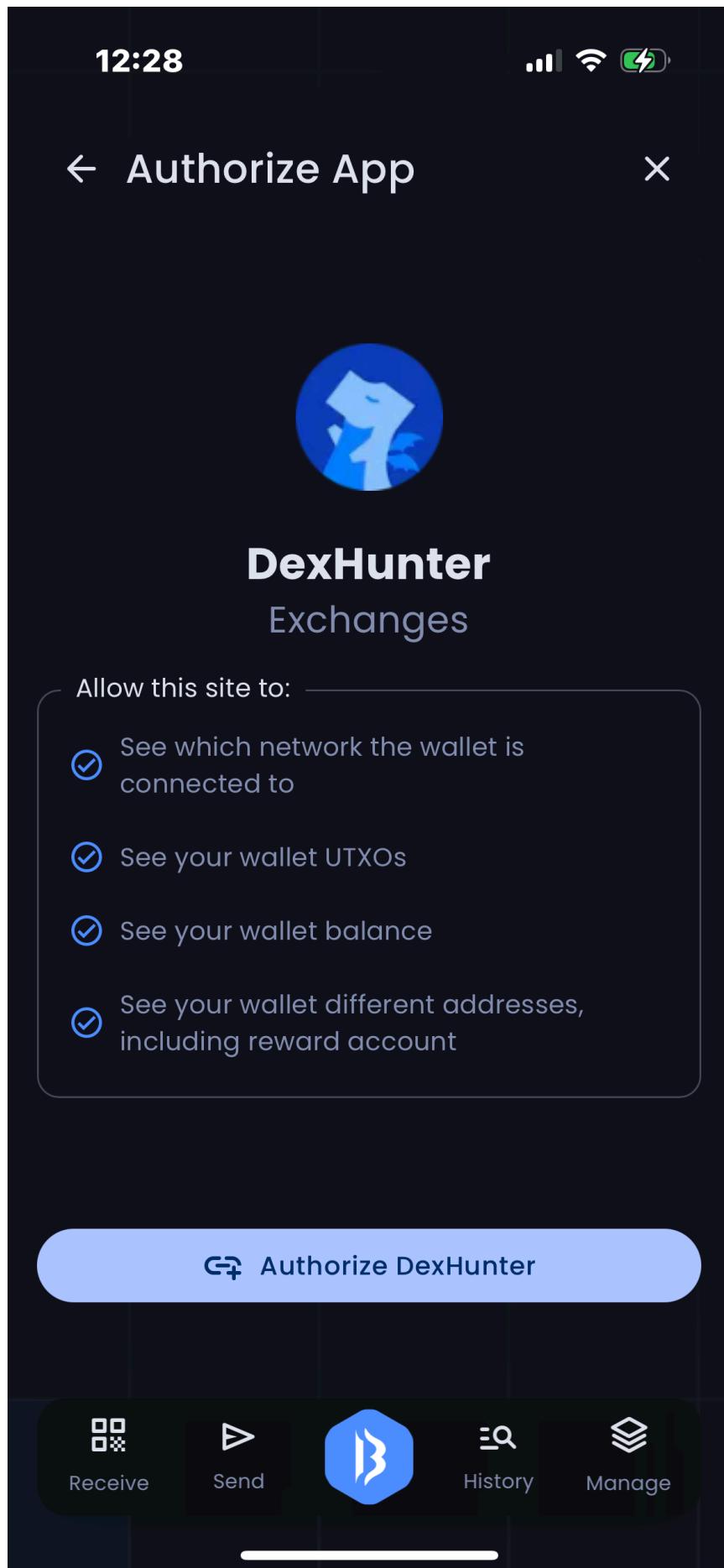
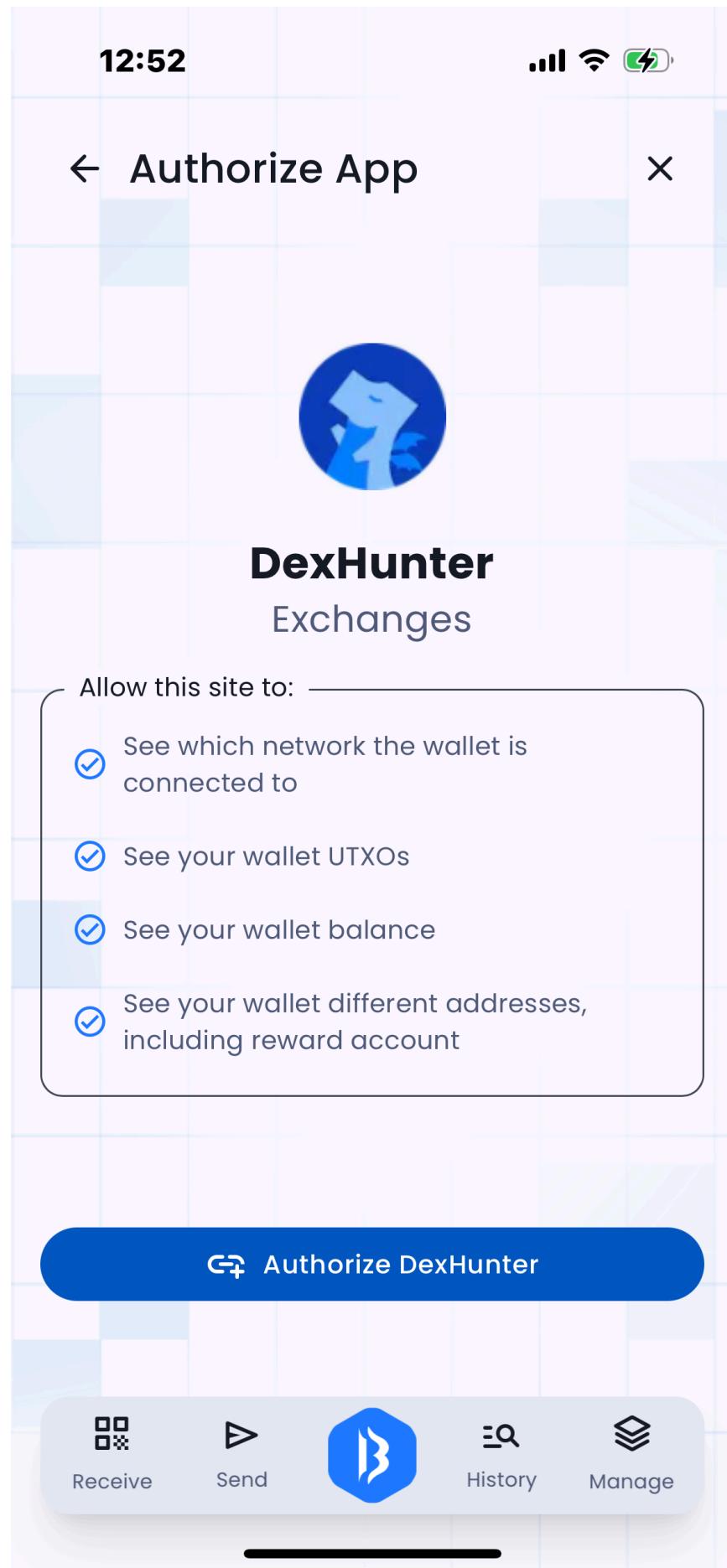


Figure 38 — Buriza UI - Mobile: dApp explorer authorization request



iv.1.7 Wallet Creation and Account Management

Figure 39 – Buriza UI - Mobile: Manage wallet

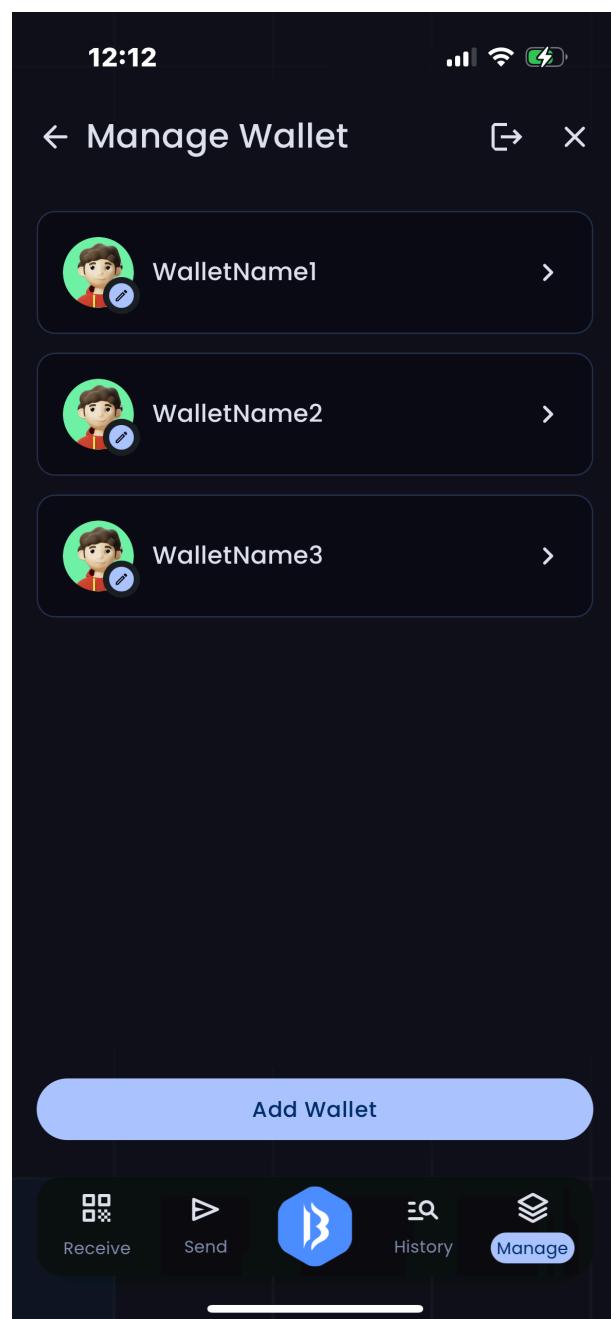


Figure 40 – Buriza UI - Mobile: Manage wallet

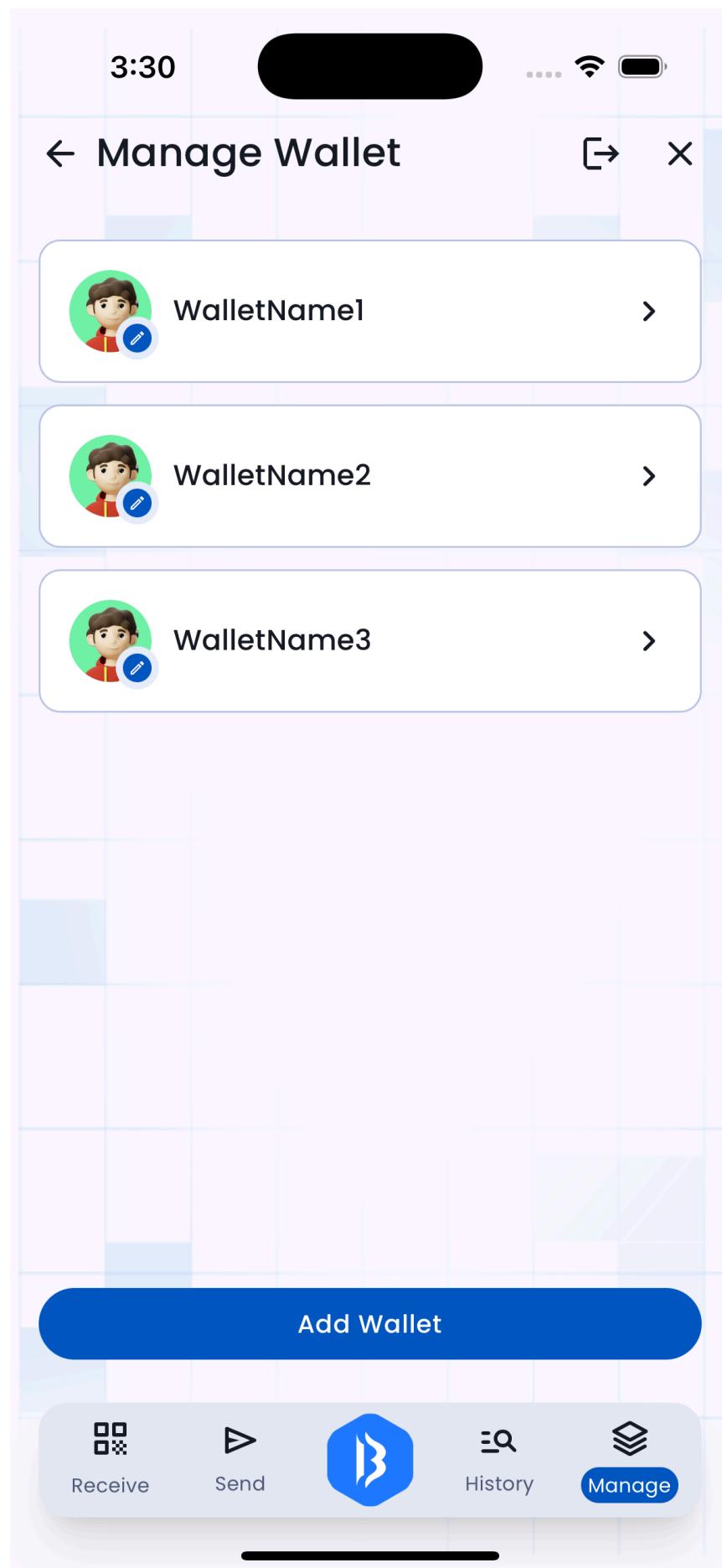


Figure 41 — Buriza UI - Mobile: Manage wallet

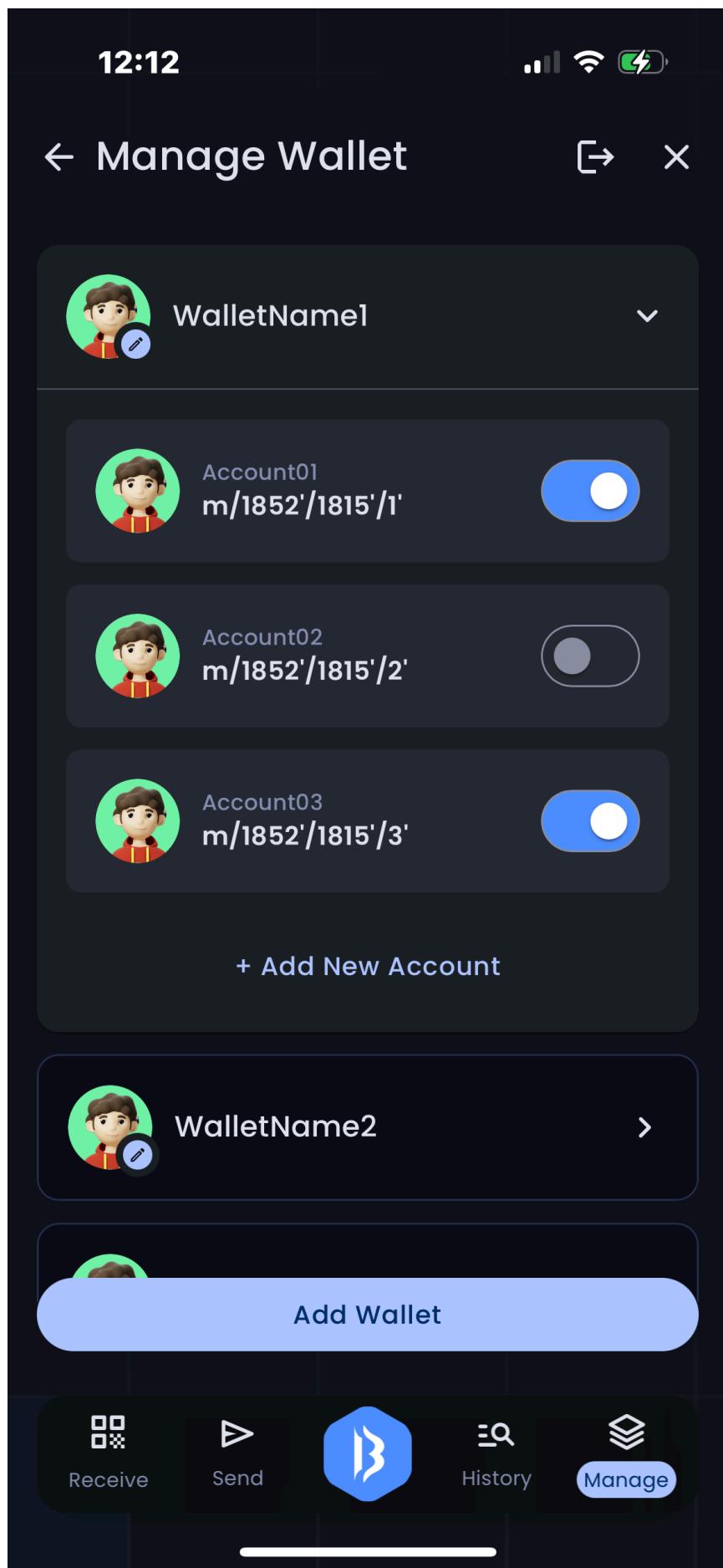


Figure 42 – Buriza UI - Mobile: Manage wallet

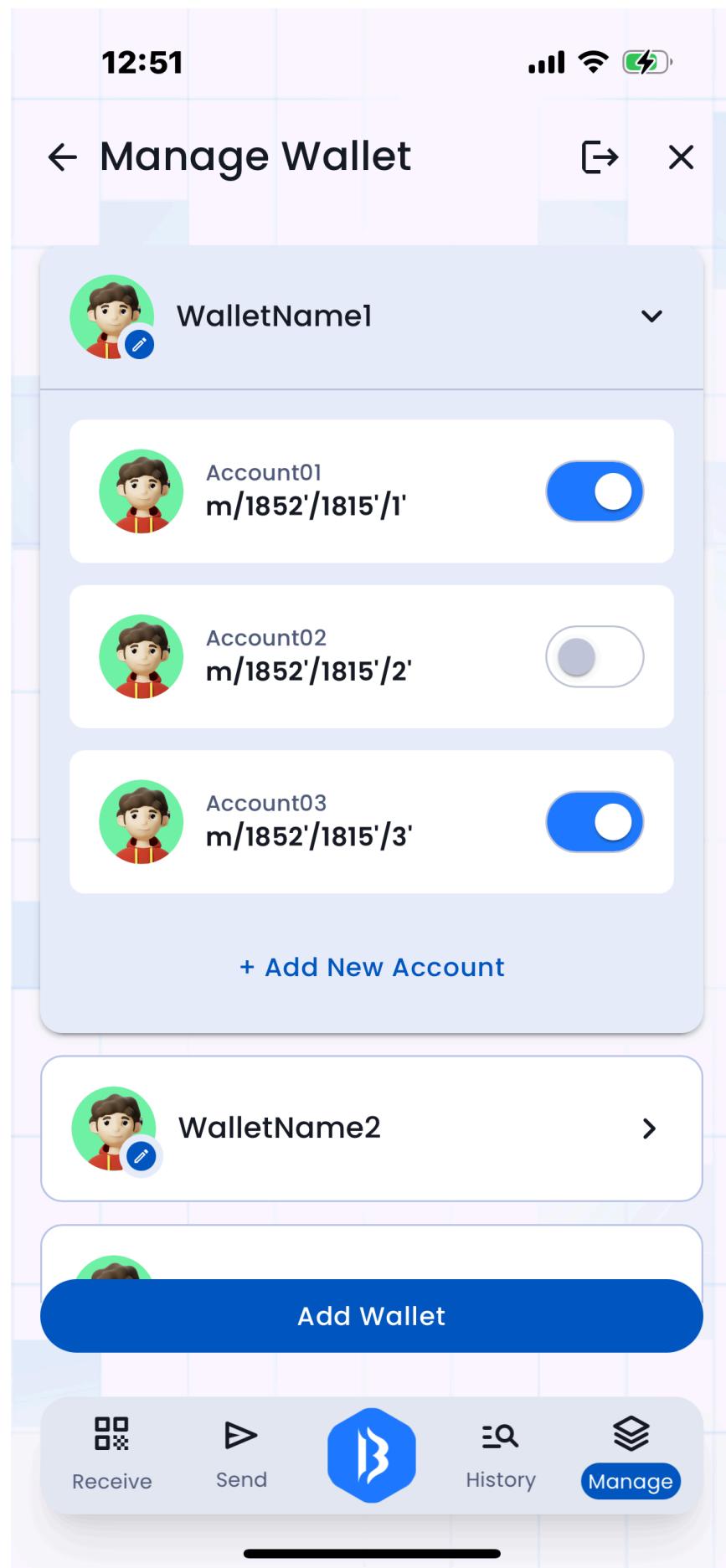


Figure 43 — Buriza UI - Mobile: Manage wallet - add new account



Figure 44 – Buriza UI - Mobile: Manage wallet - add new account



Figure 45 — Buriza UI - Mobile: Manage wallet - edit account



Figure 46 — Buriza UI - Mobile: Manage wallet - edit account



Figure 47 – Buriza UI - Mobile: Manage wallet - add new wallet

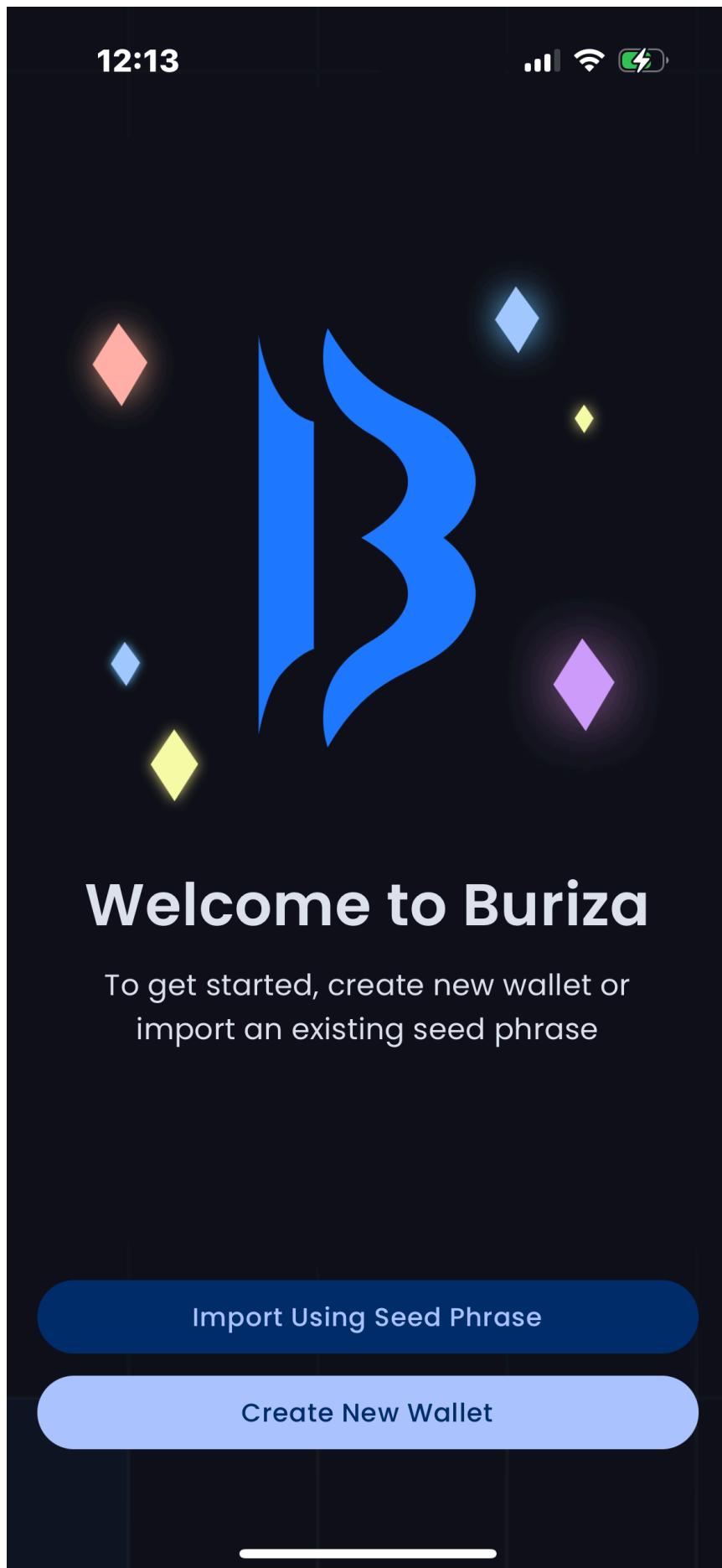


Figure 48 – Buriza UI - Mobile: Manage wallet - add new wallet

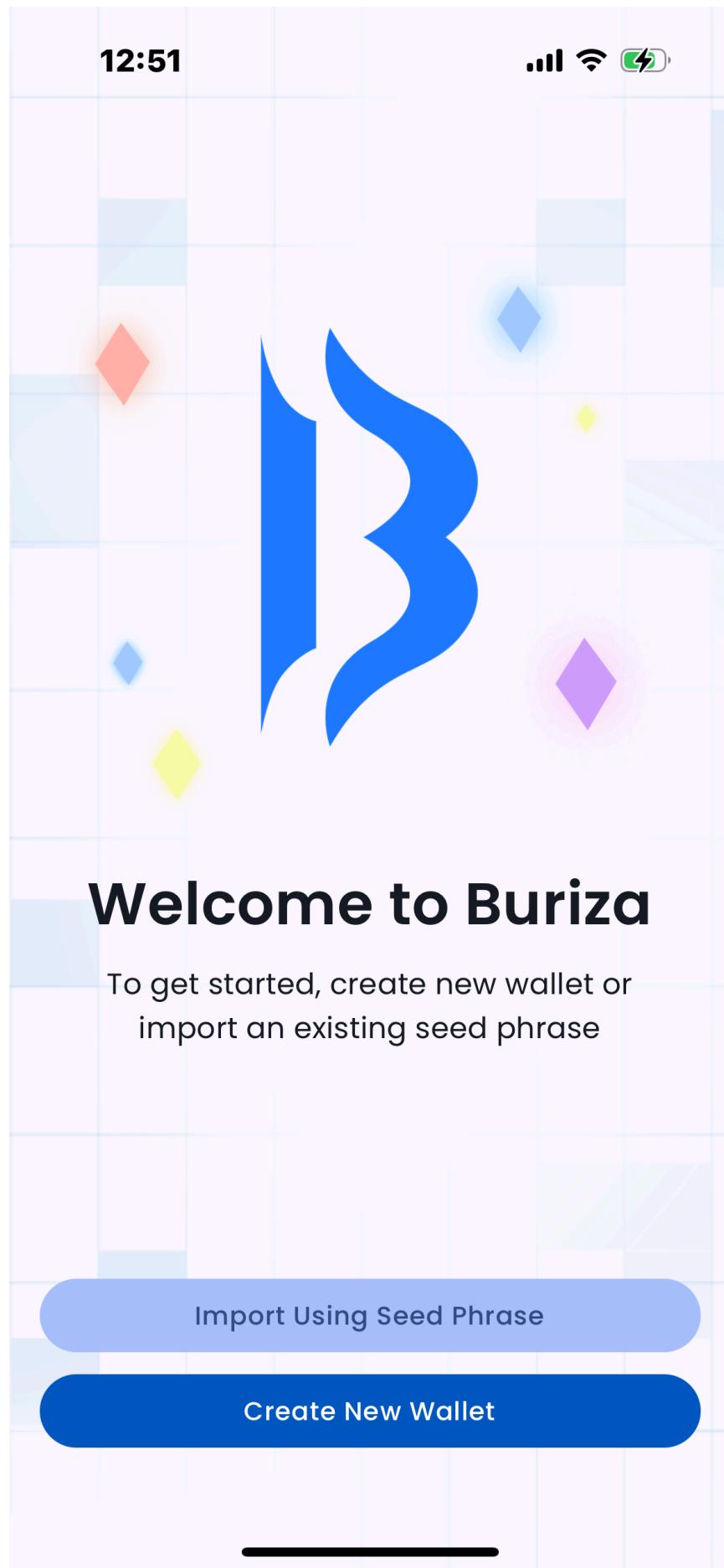


Figure 49 — Buriza UI - Mobile: Create new wallet - save recovery phrase

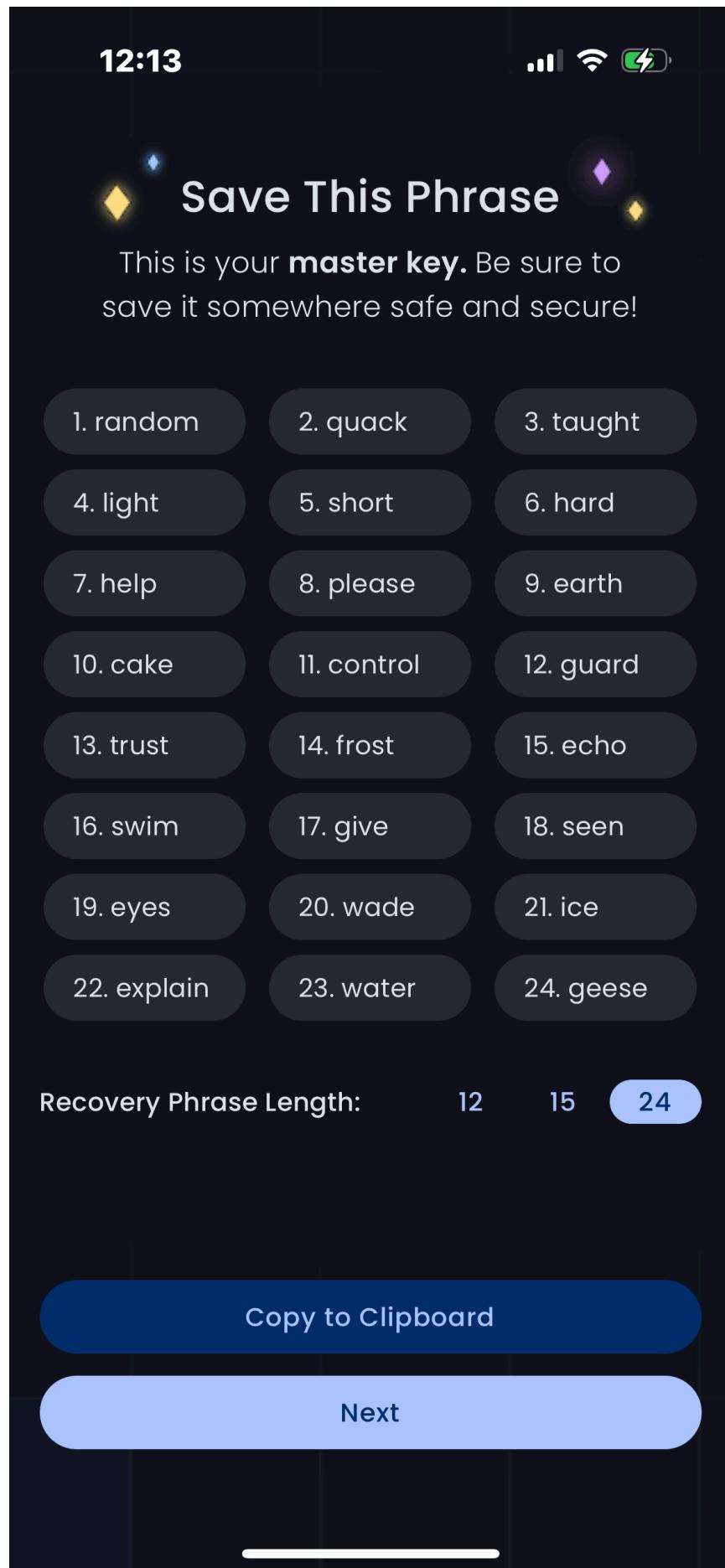


Figure 50 — Buriza UI - Mobile: Create new wallet - save recovery phrase

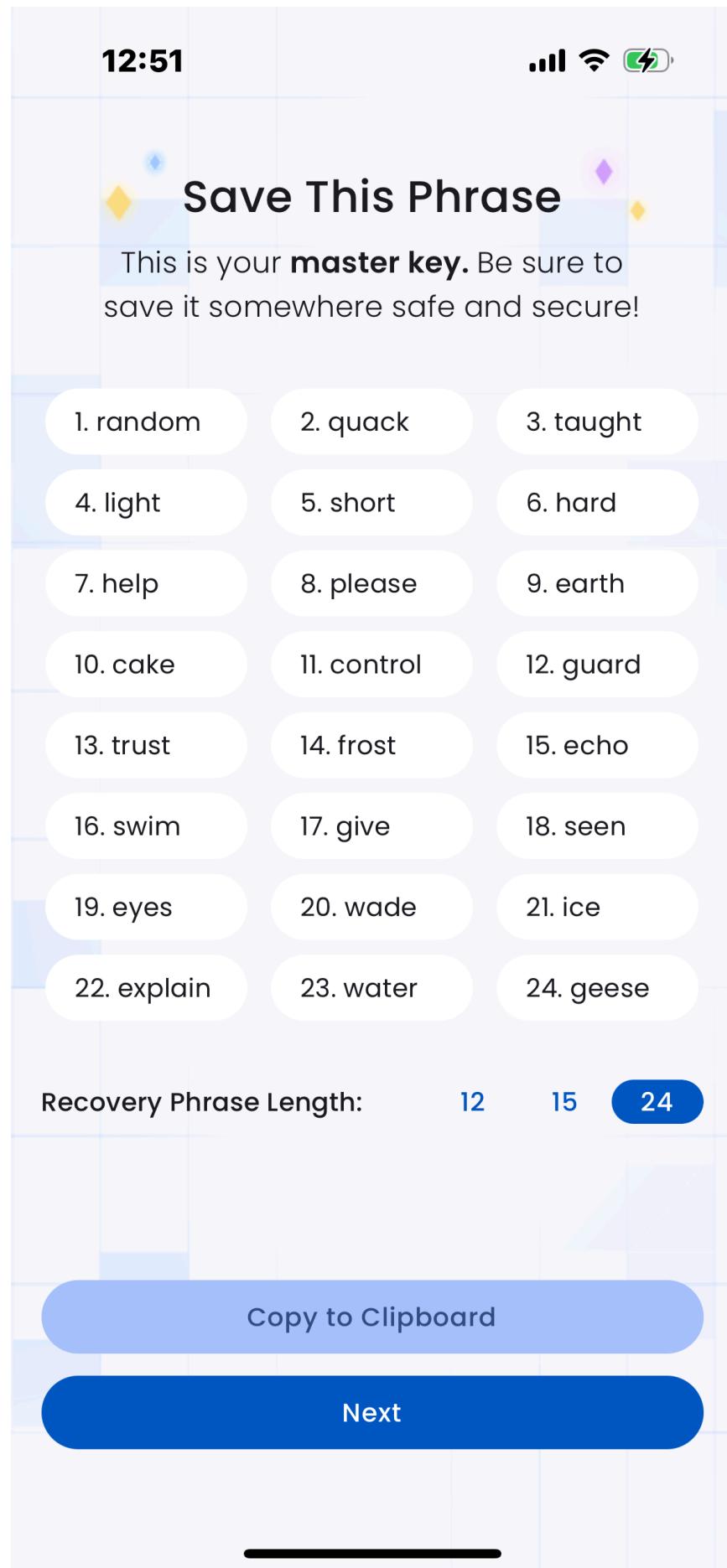


Figure 51 – Buriza UI - Mobile: Create new wallet - verify recovery phrase

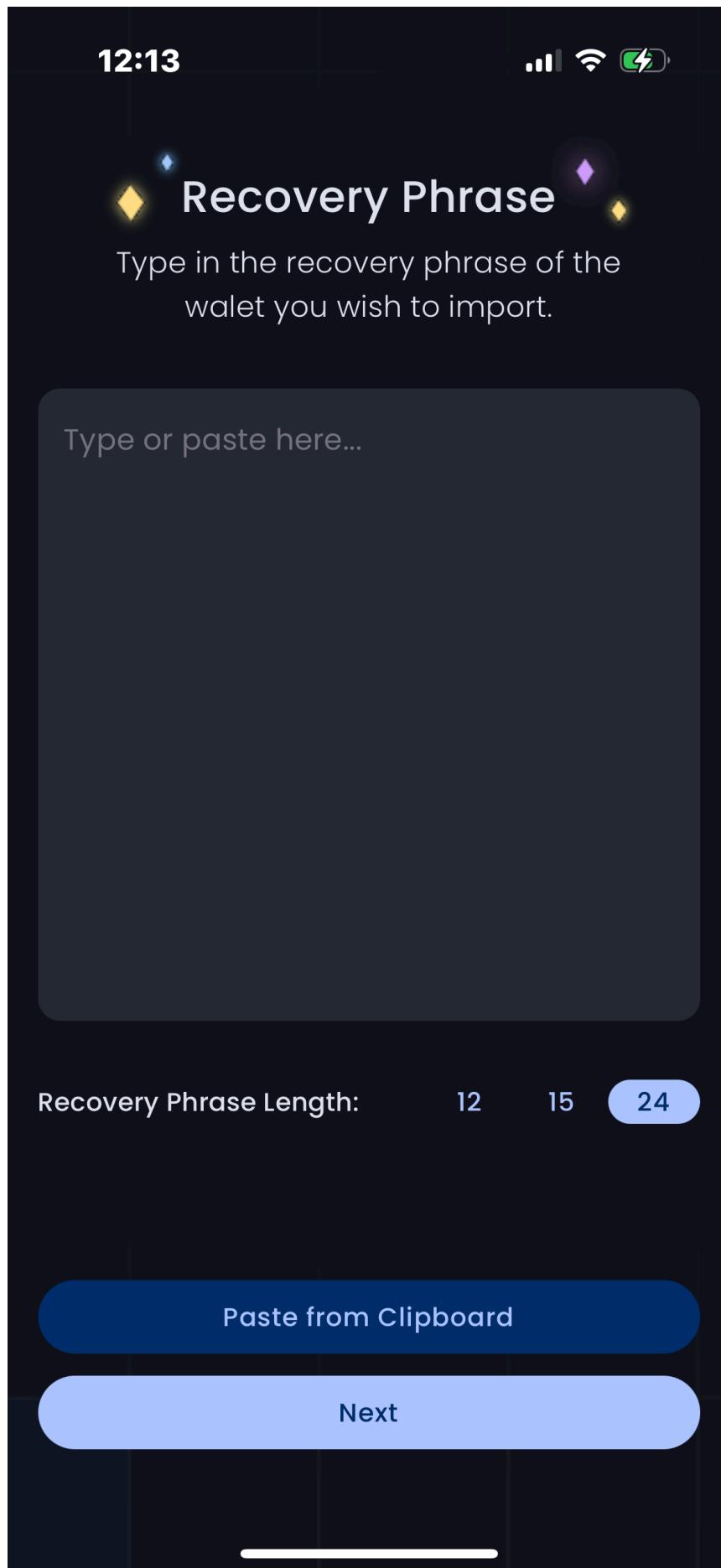


Figure 52 – Buriza UI - Mobile: Create new wallet - verify recovery phrase

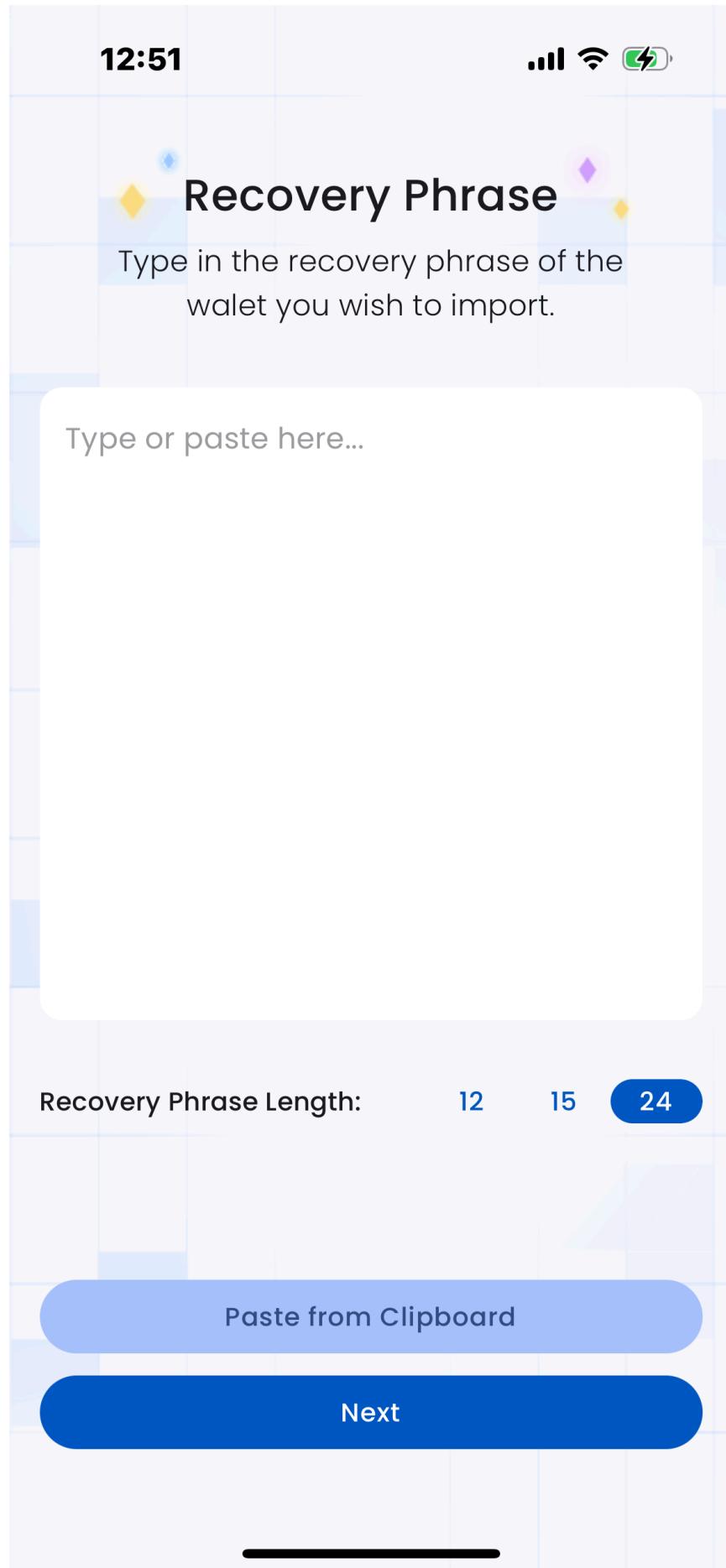


Figure 53 – Buriza UI - Mobile: Create new wallet - setup

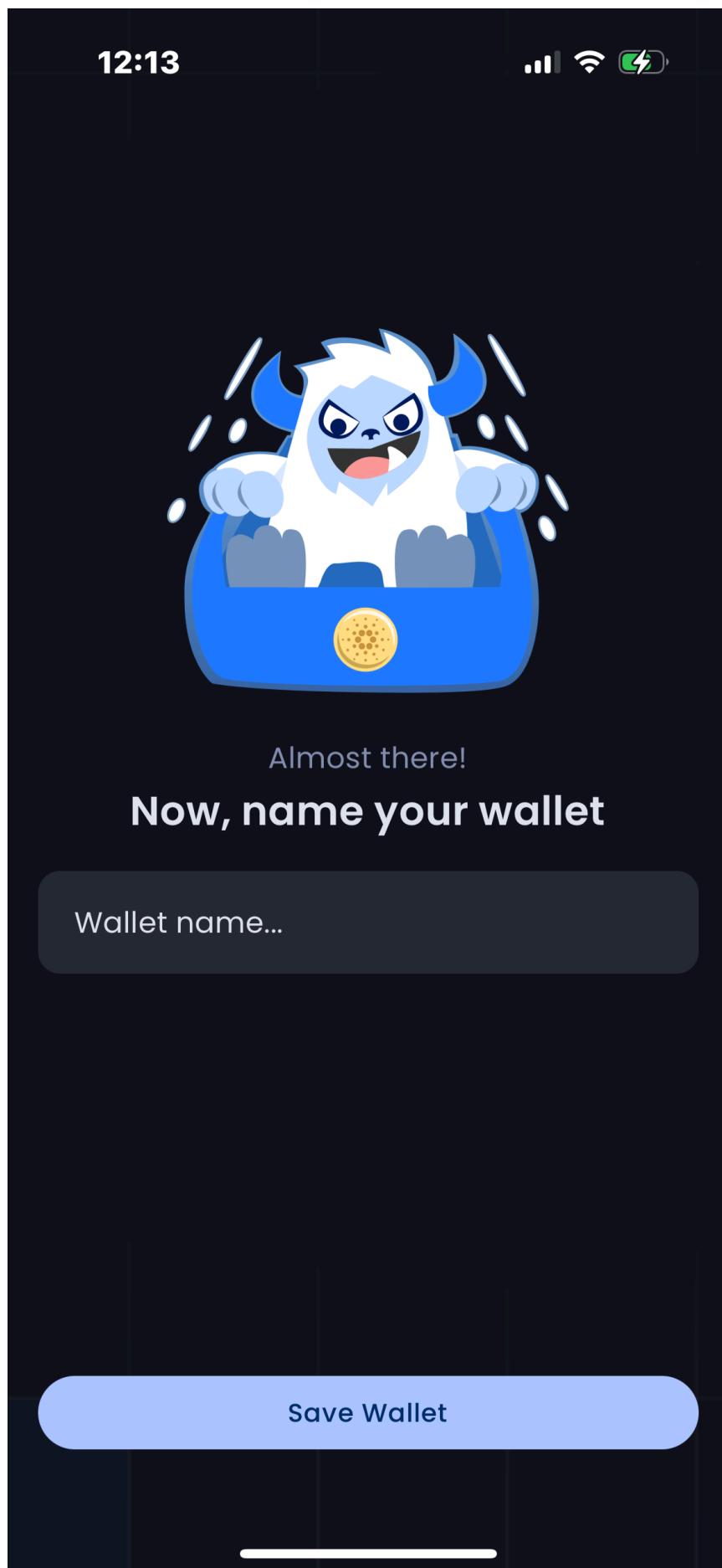


Figure 54 – Buriza UI - Mobile: Create new wallet - setup

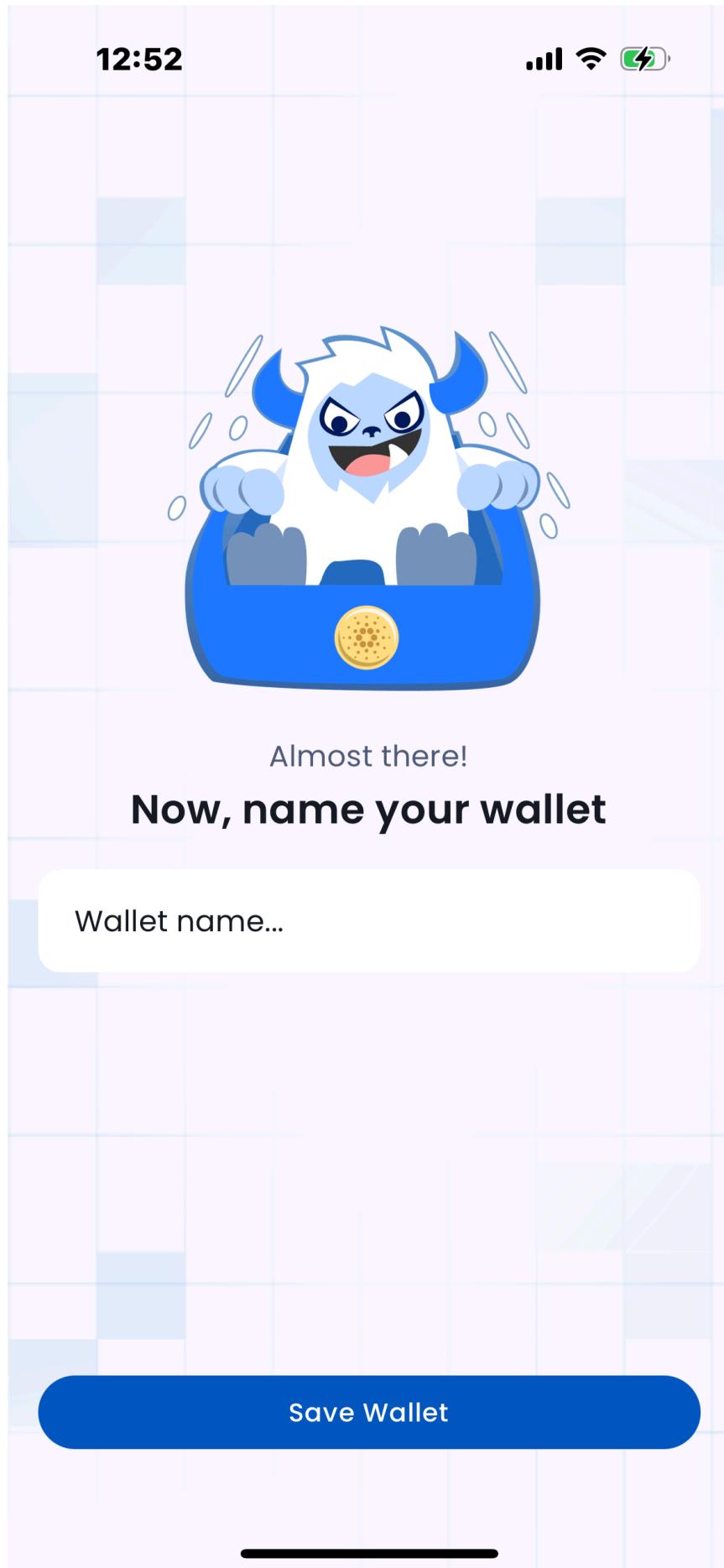


Figure 55 — Buriza UI - Mobile: Create new wallet - authentication

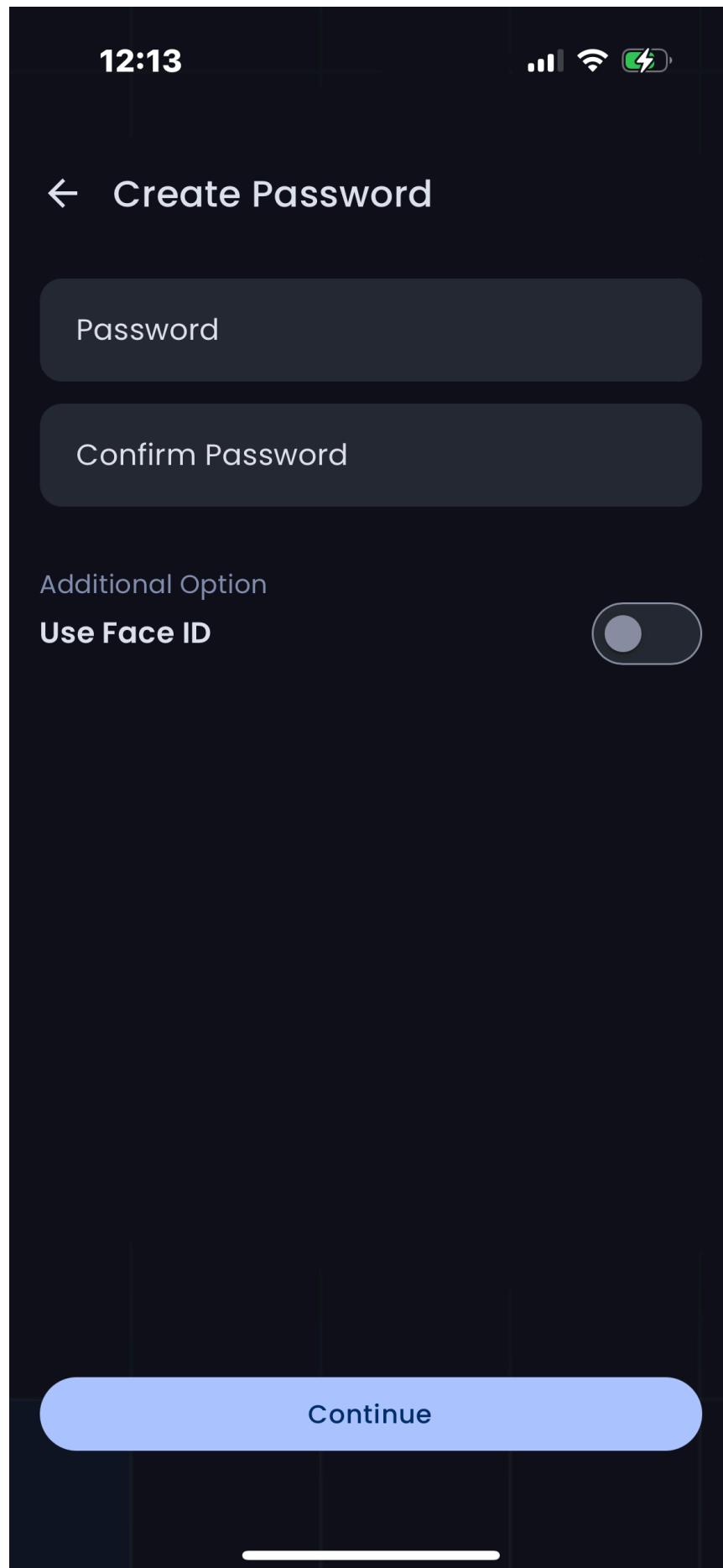


Figure 56 — Buriza UI - Mobile: Create new wallet - authentication

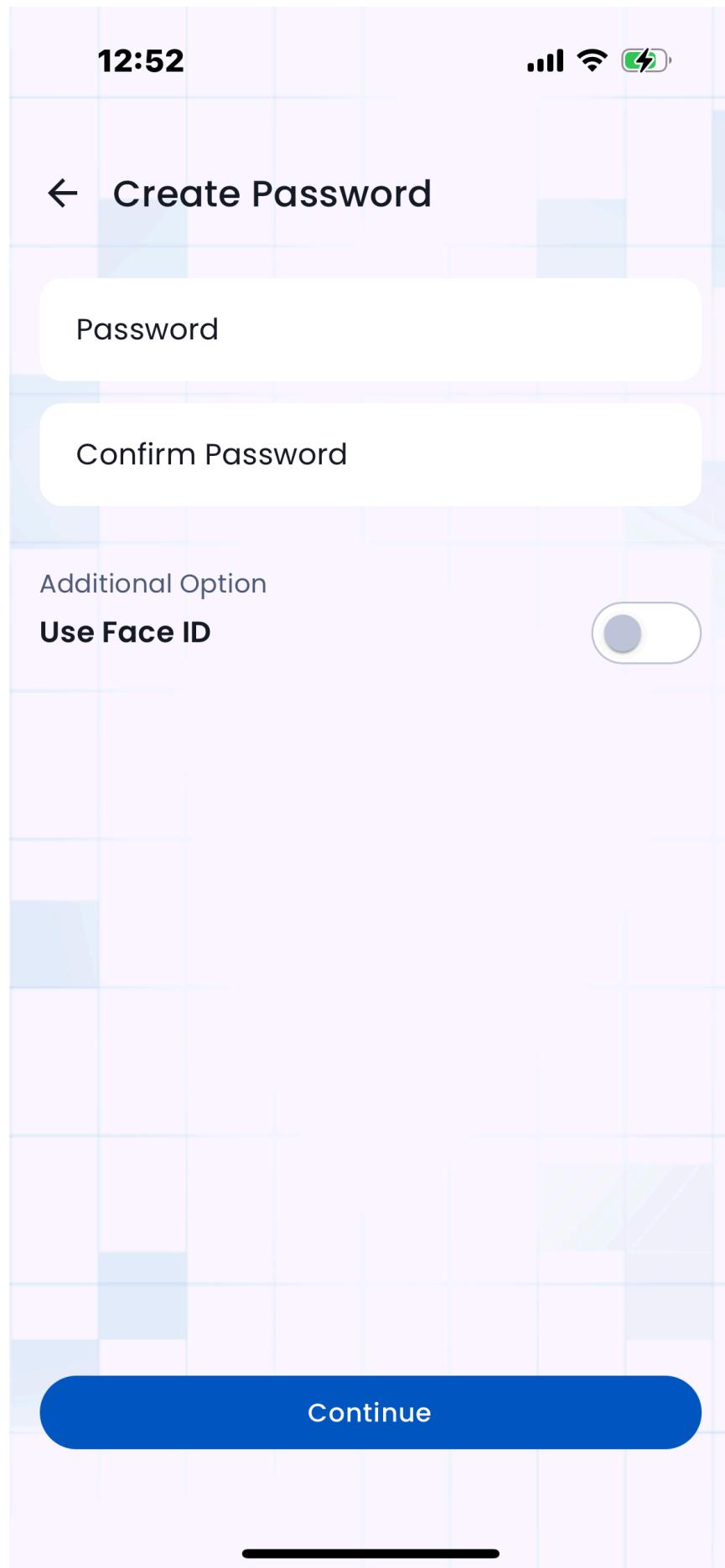


Figure 57 — Buriza UI - Mobile: Create new wallet - success screen

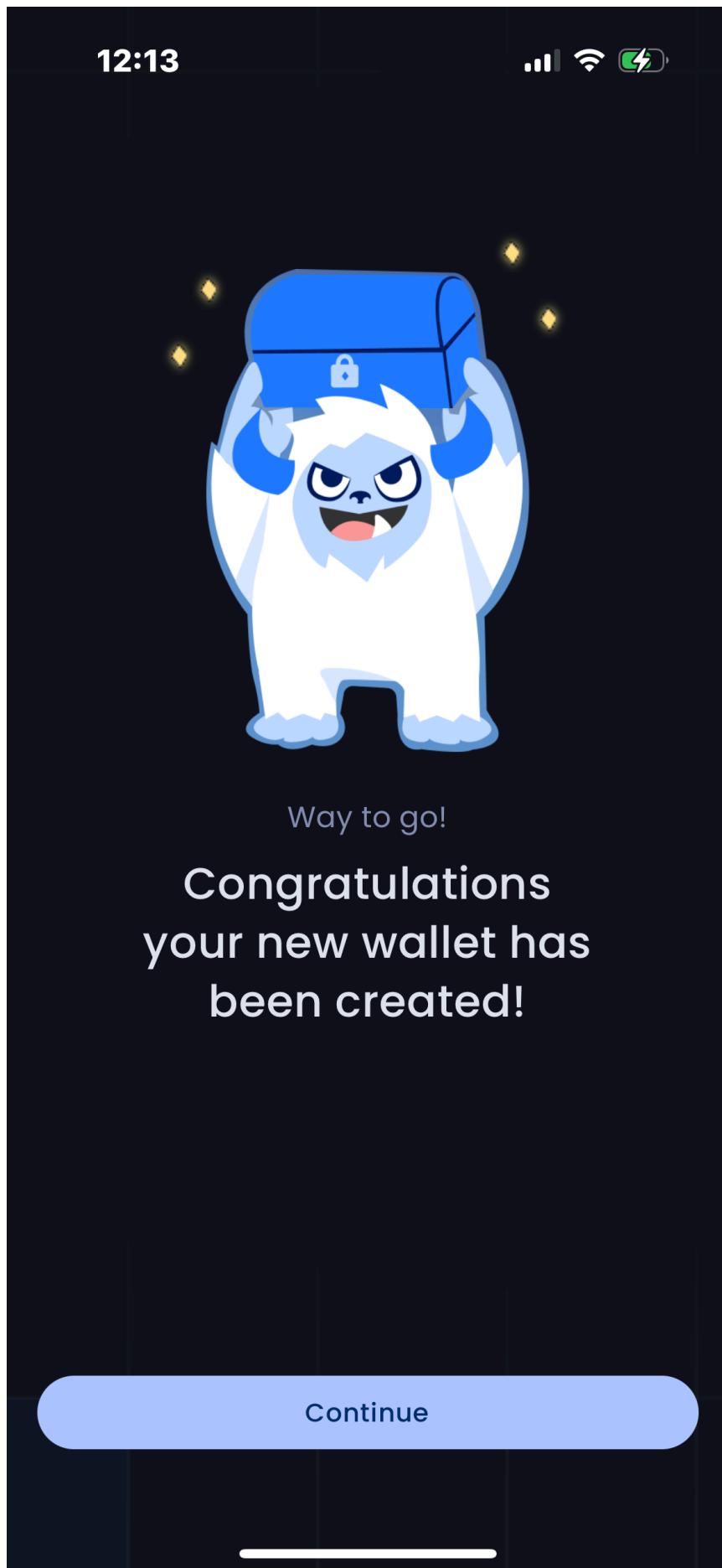
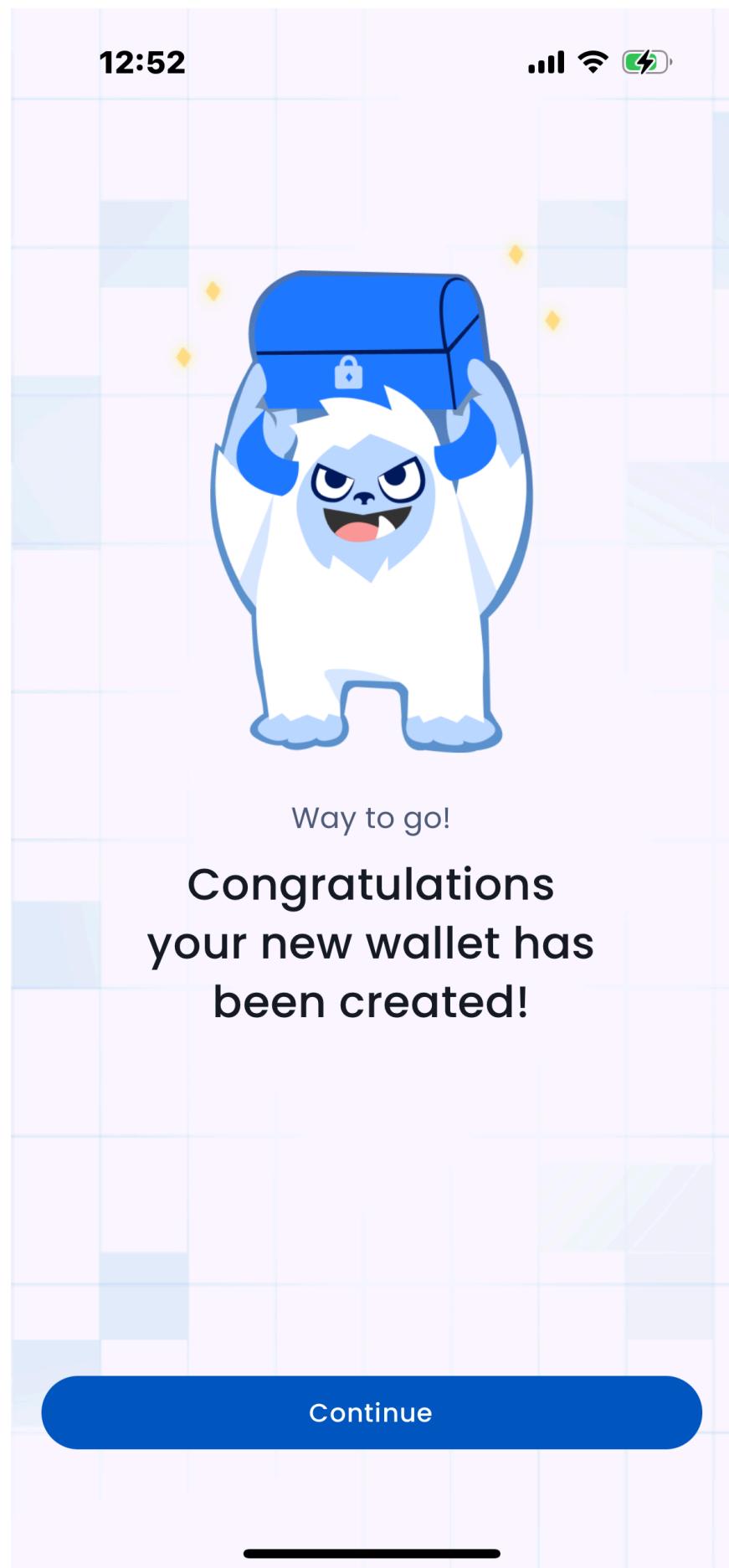


Figure 58 — Buriza UI - Mobile: Create new wallet - success screen



iv.2 Browser

iv.2.1 Send Assets

Figure 59 — Buriza UI - Browser: Send assets

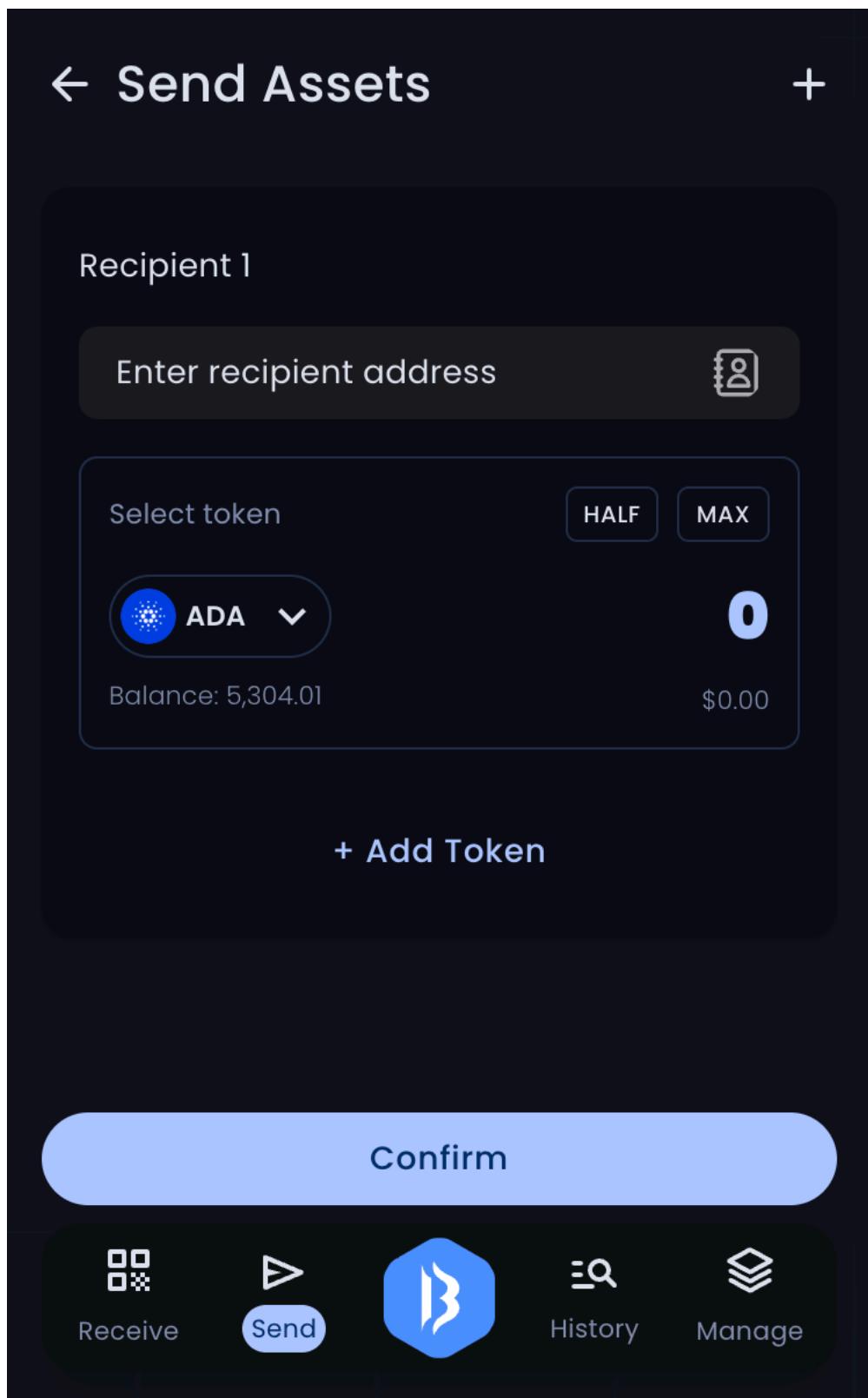


Figure 60 — Buriza UI - Browser: Send assets

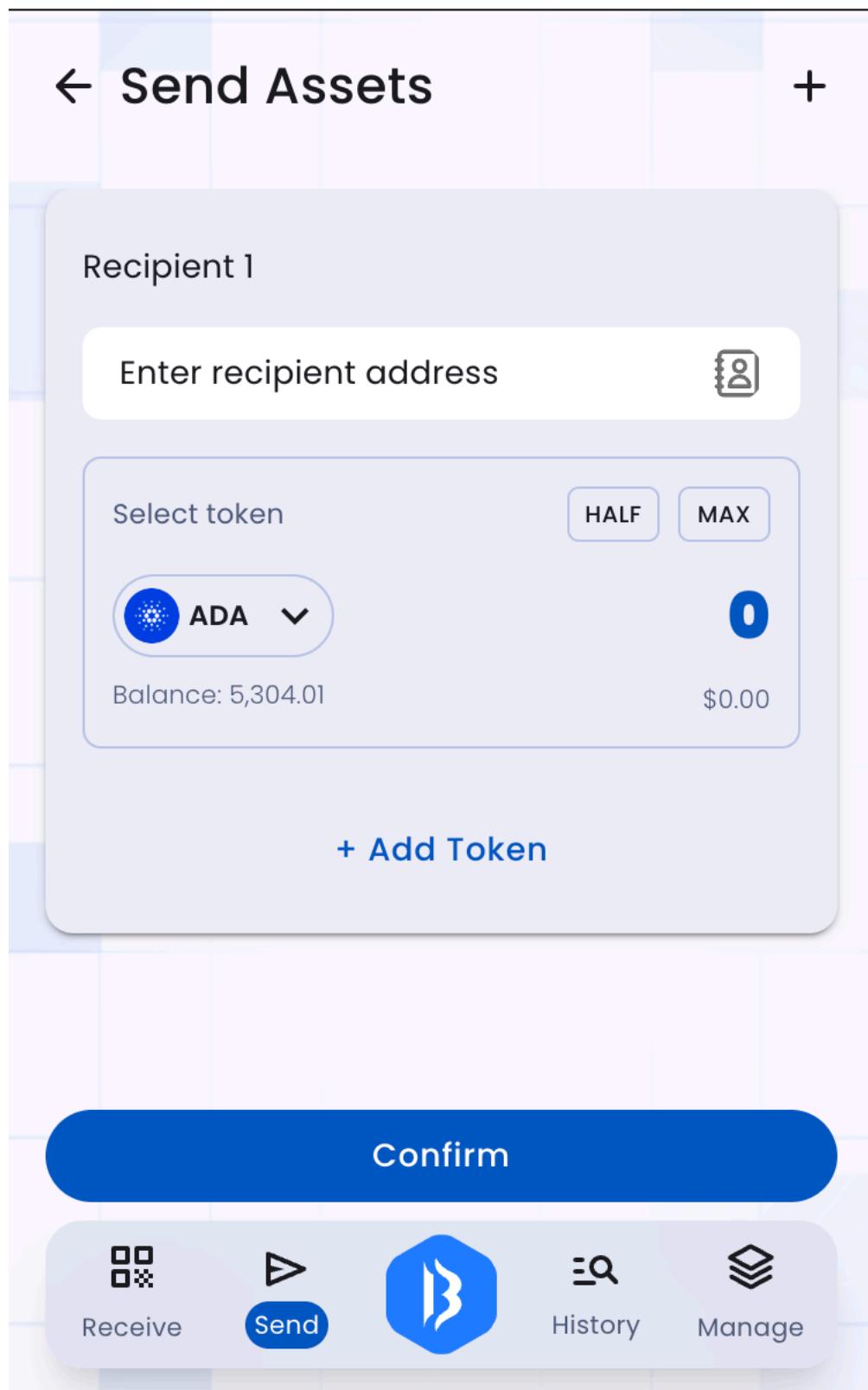


Figure 61 — Buriza UI - Browser: Send assets with an additional recipient

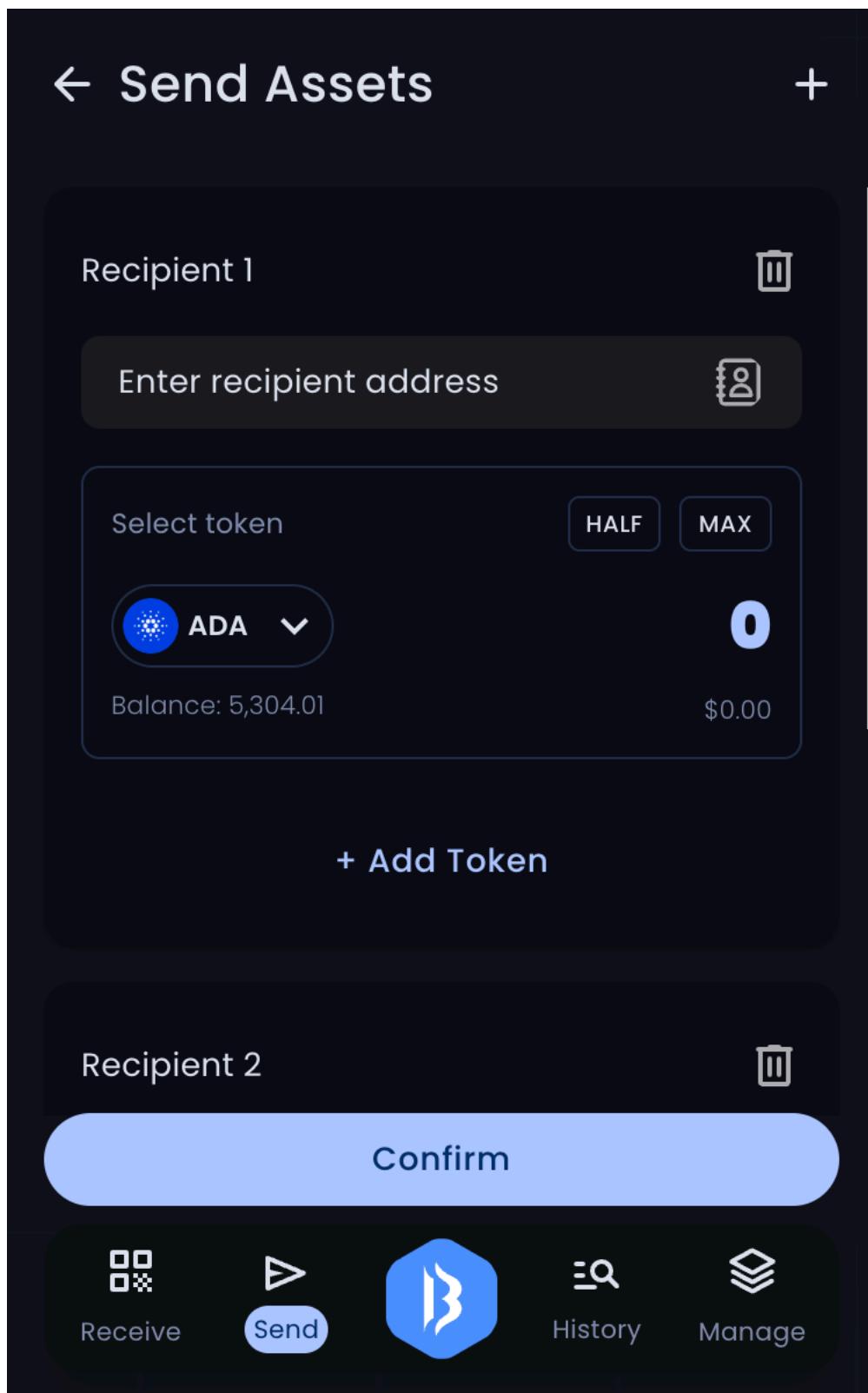


Figure 62 – Buriza UI - Browser: Send assets with an additional recipient

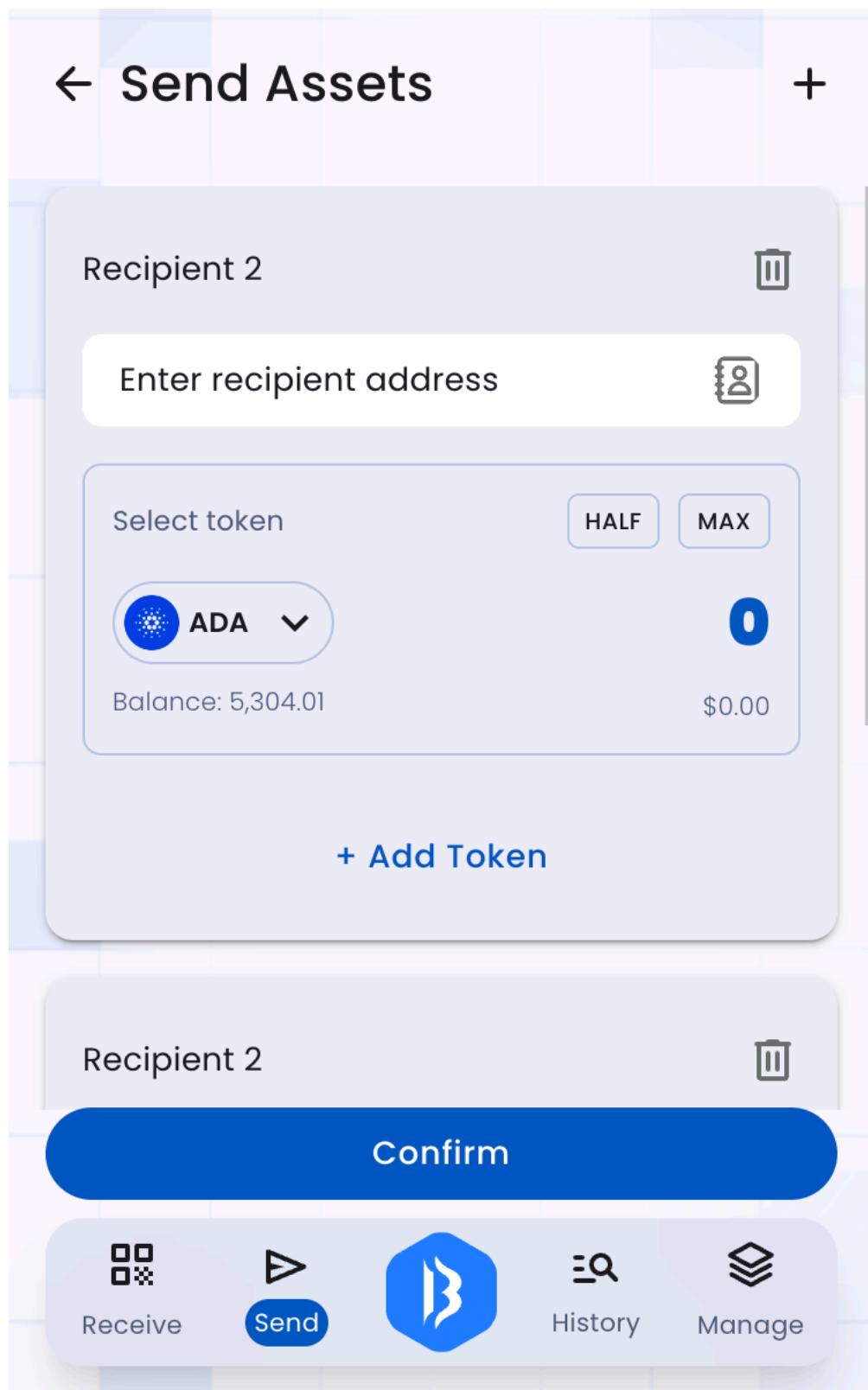


Figure 63 – Buriza UI - Browser: Send assets with an additional token

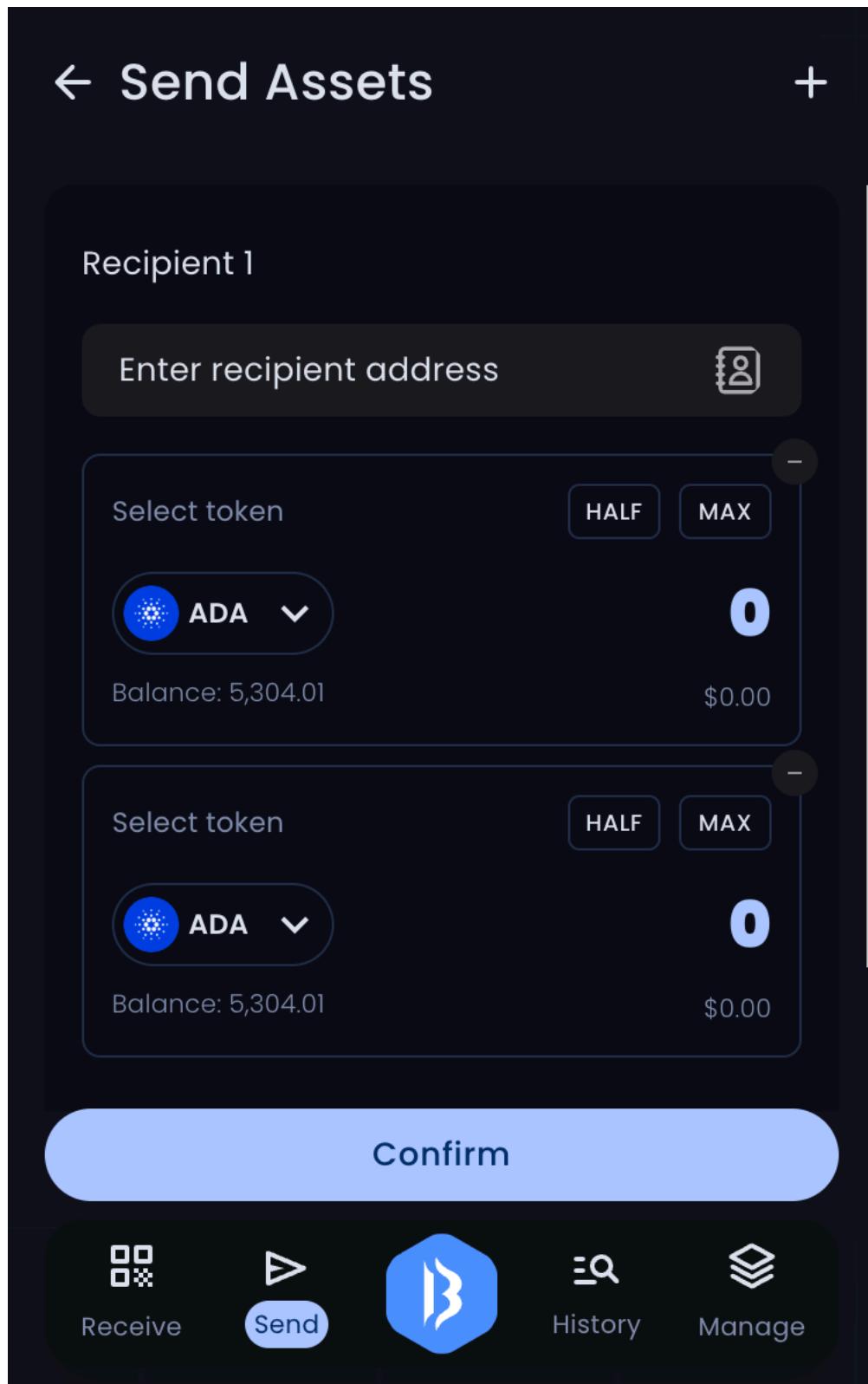


Figure 64 – Buriza UI - Browser: Send assets with an additional token

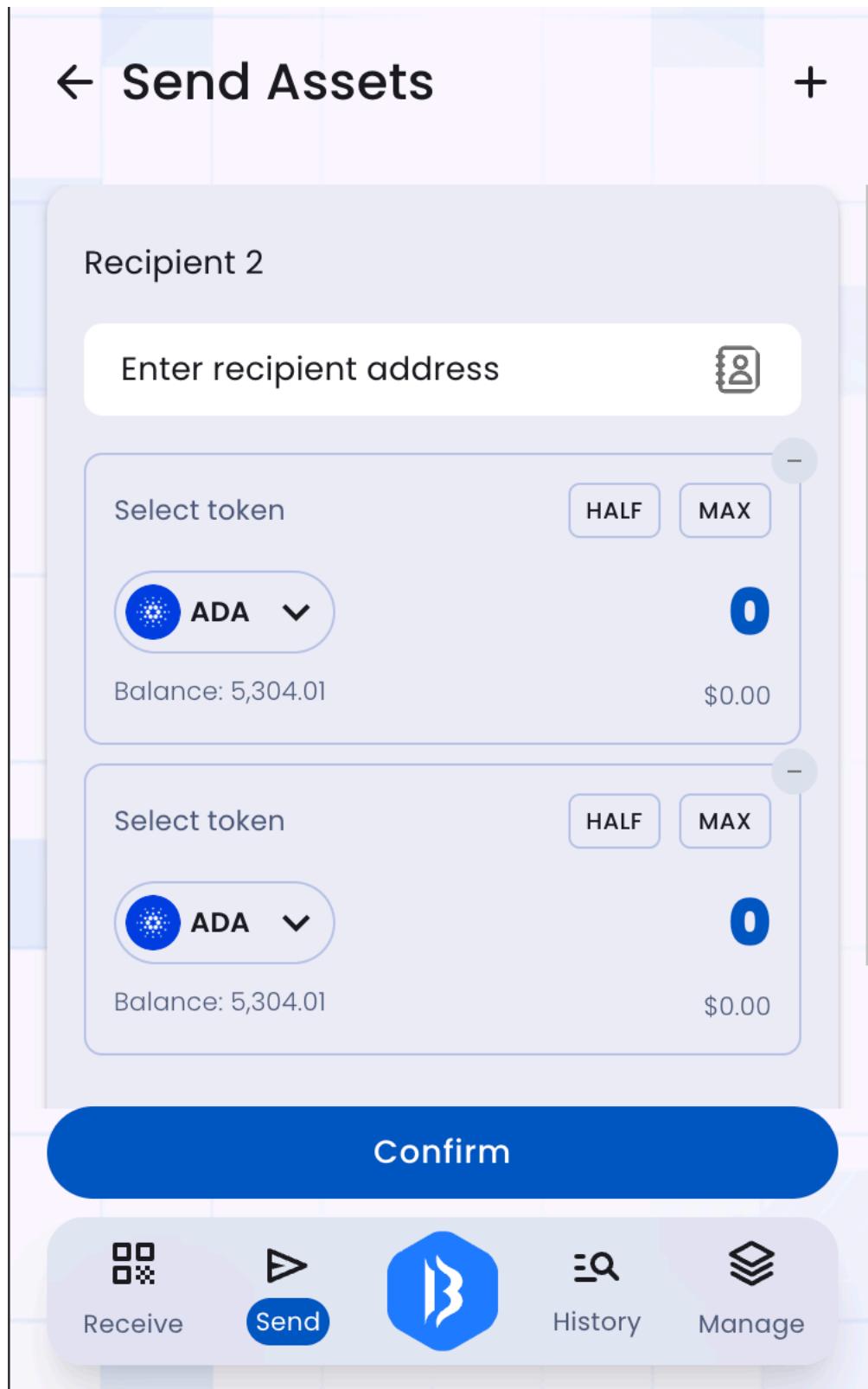


Figure 65 — Buriza UI - Browser: Send assets transaction summary

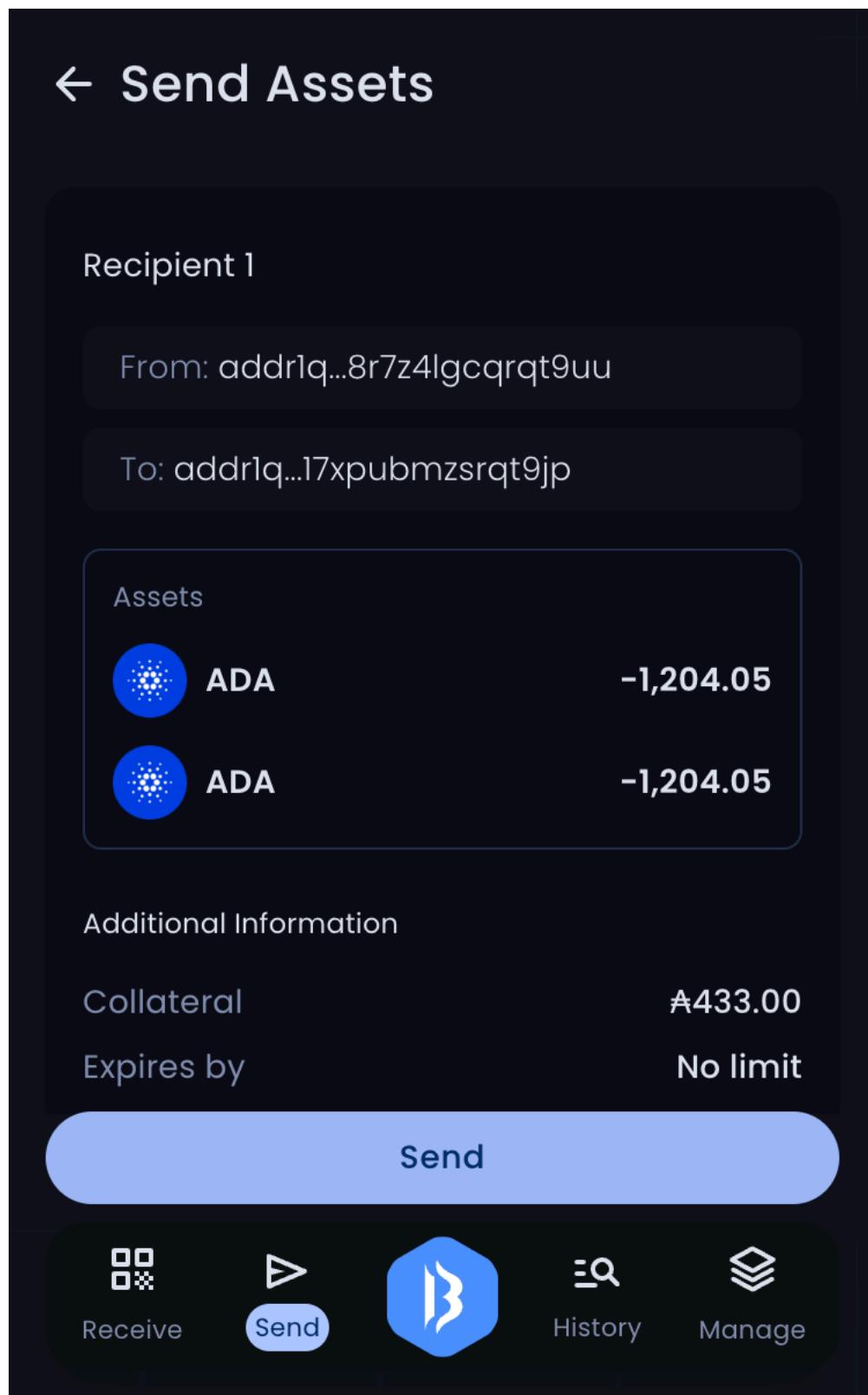


Figure 66 — Buriza UI - Browser: Send assets transaction summary

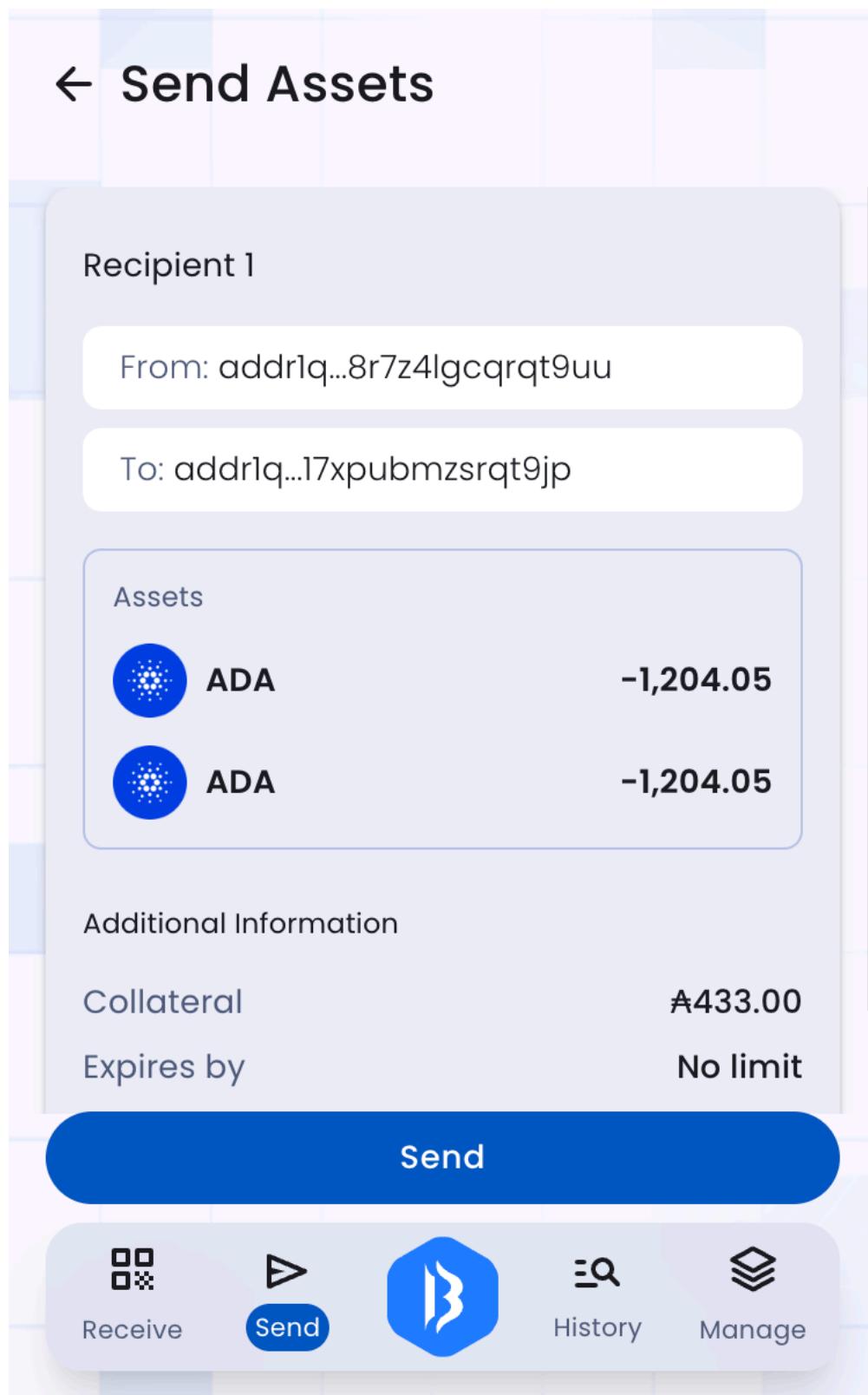


Figure 67 — Buriza UI - Browser: Send assets success screen

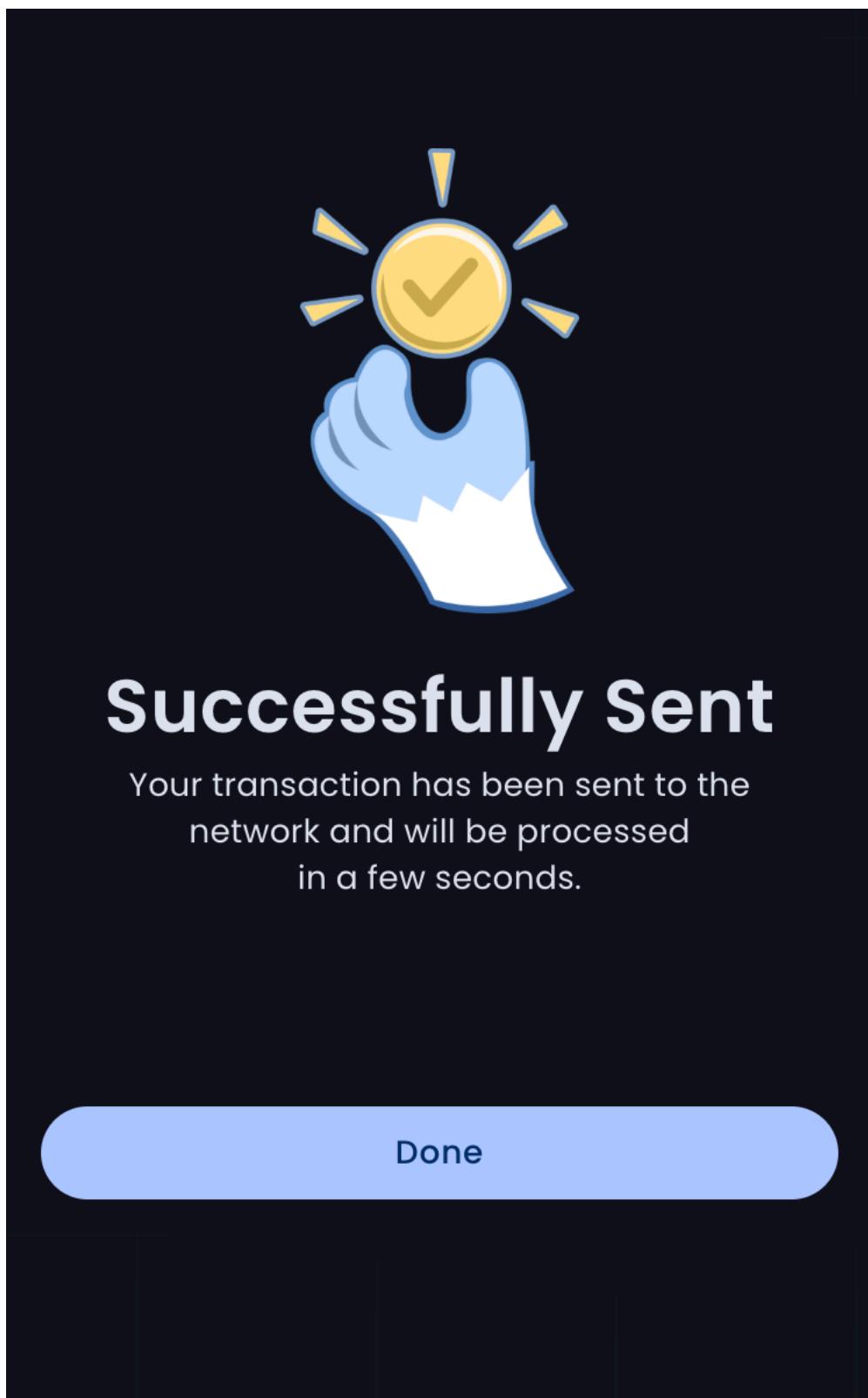
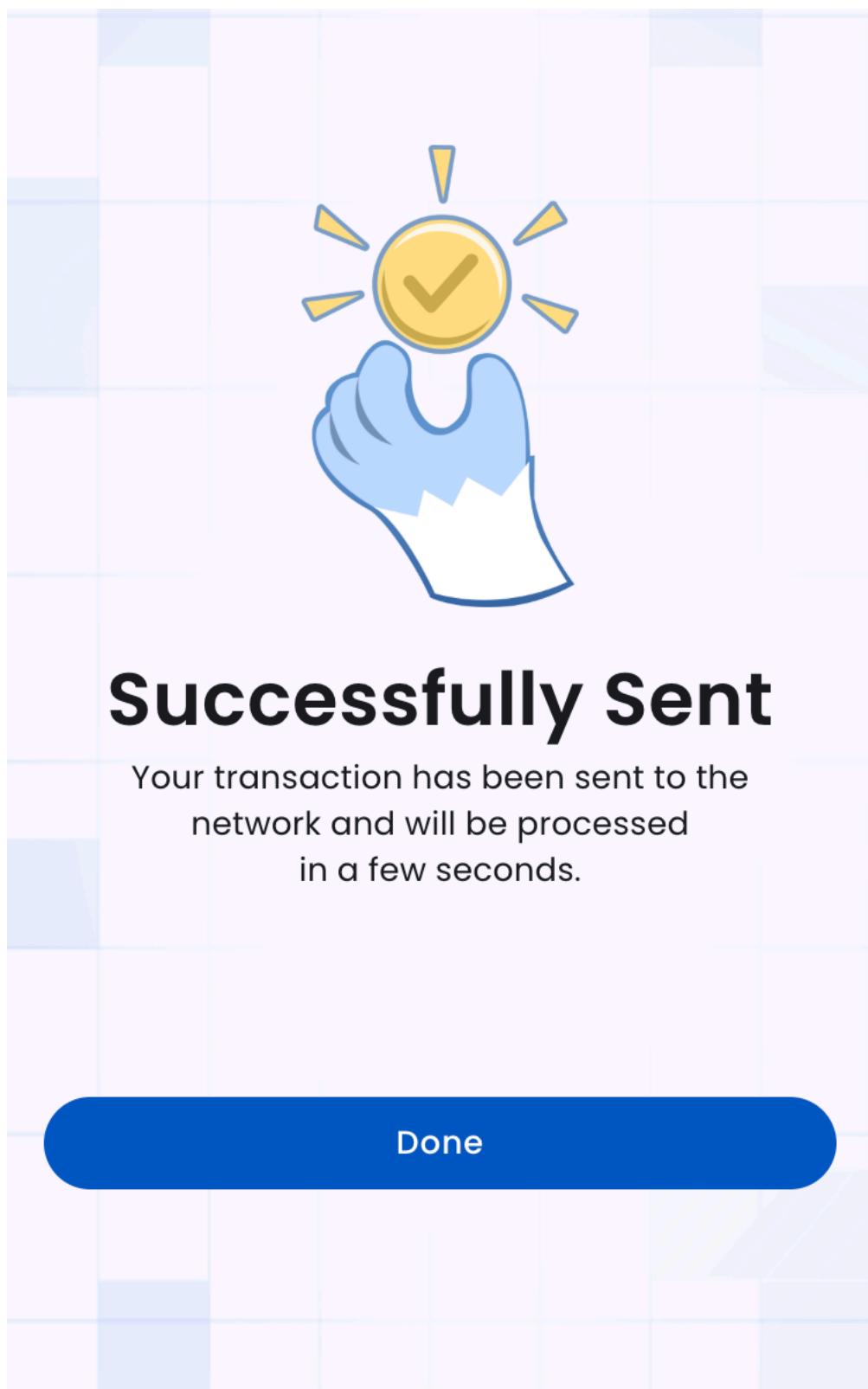


Figure 68 — Buriza UI - Browser: Send assets success screen



iv.2.2 Receiving Assets

Figure 69 – Buriza UI - Browser: Receive assets

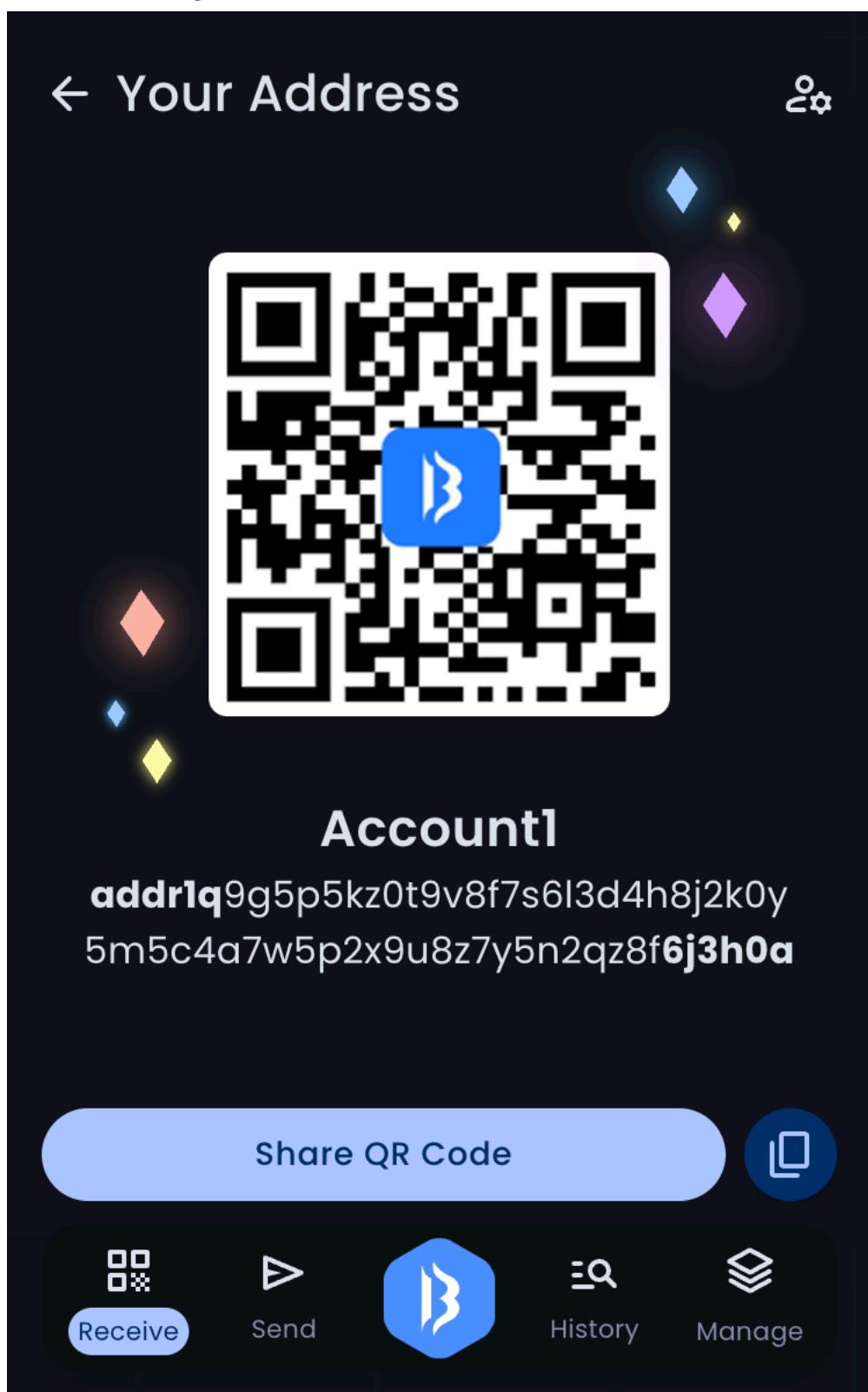


Figure 70 — Buriza UI - Browser: Receive assets

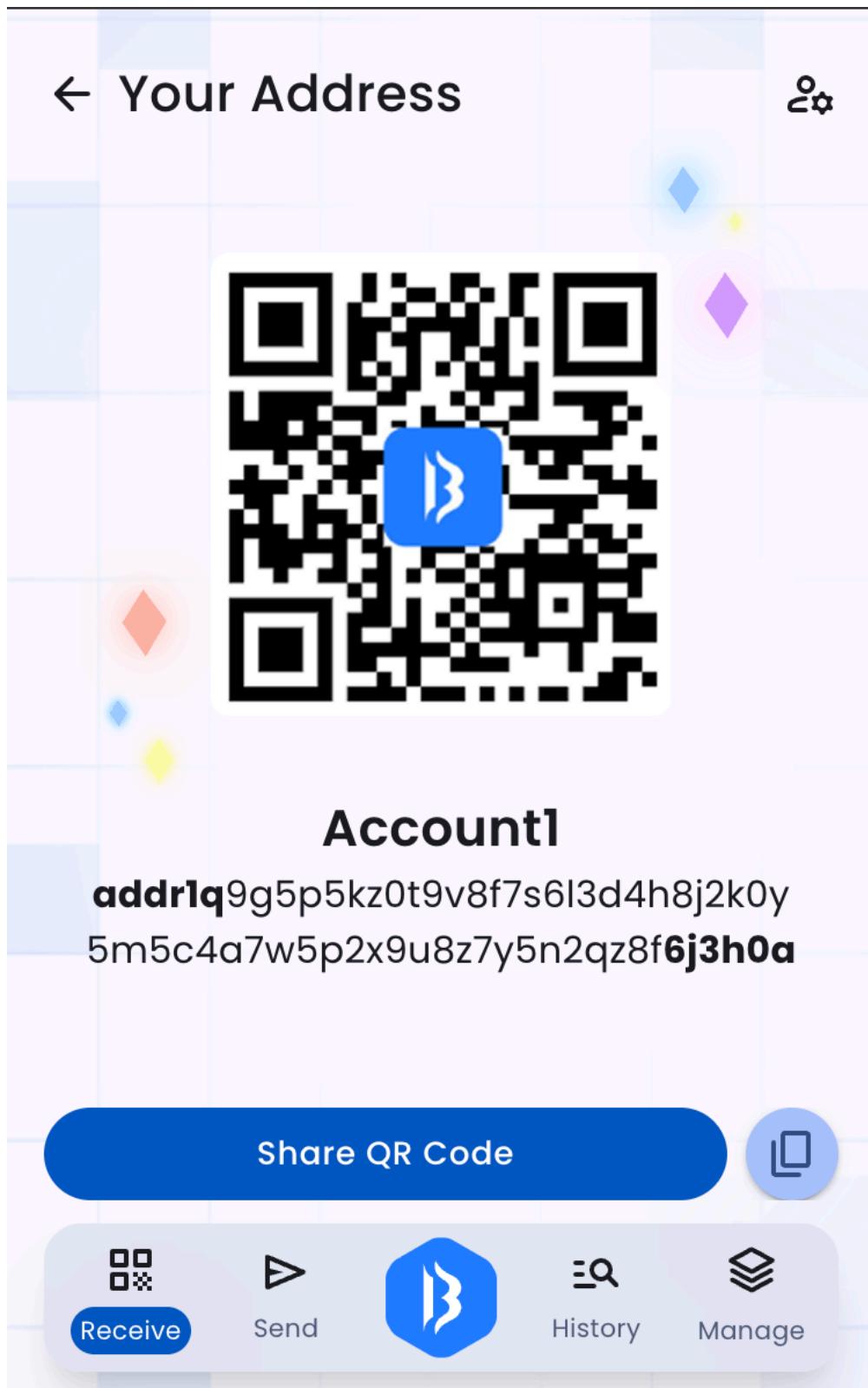


Figure 71 — Buriza UI - Browser: Receive assets advanced mode

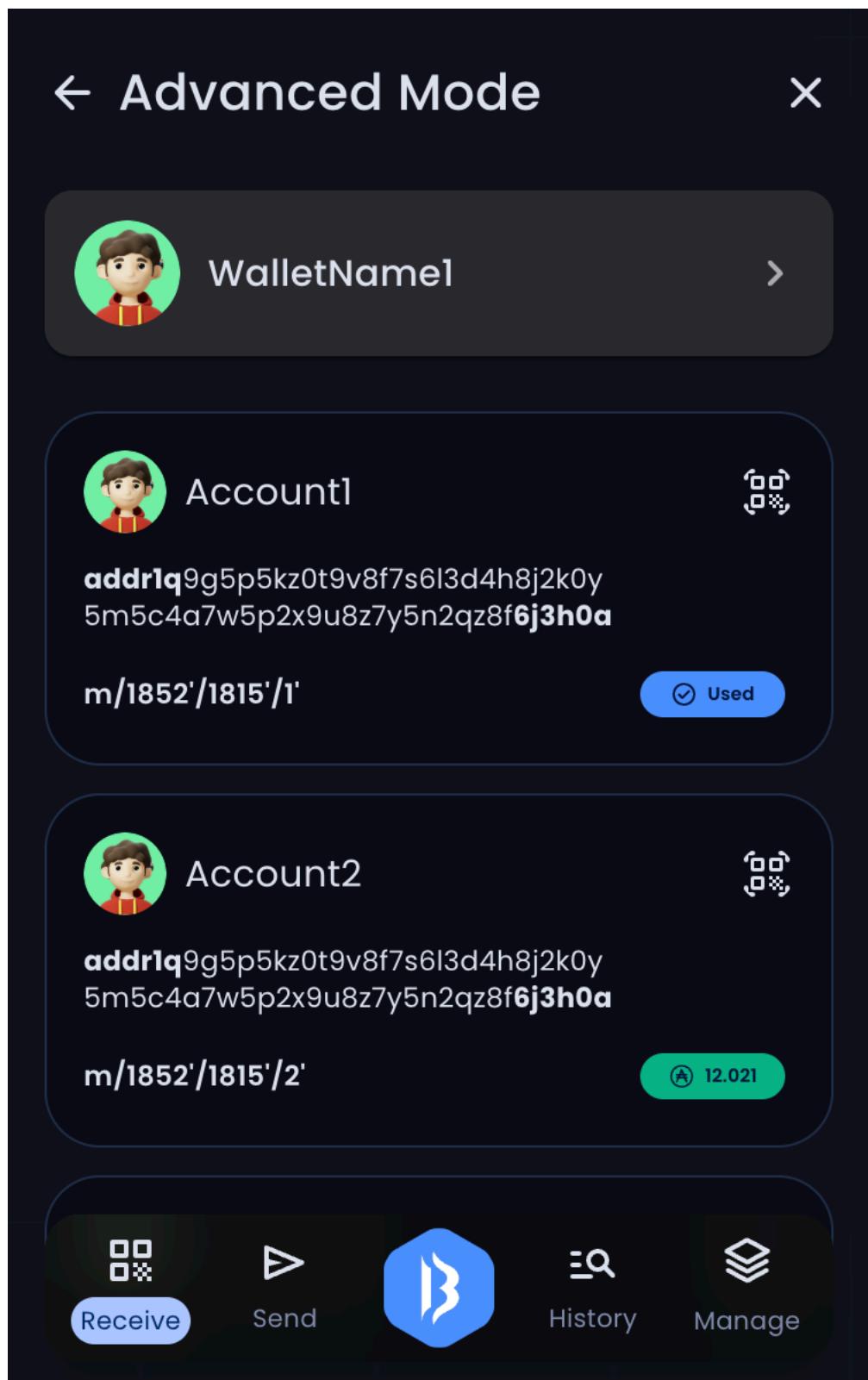
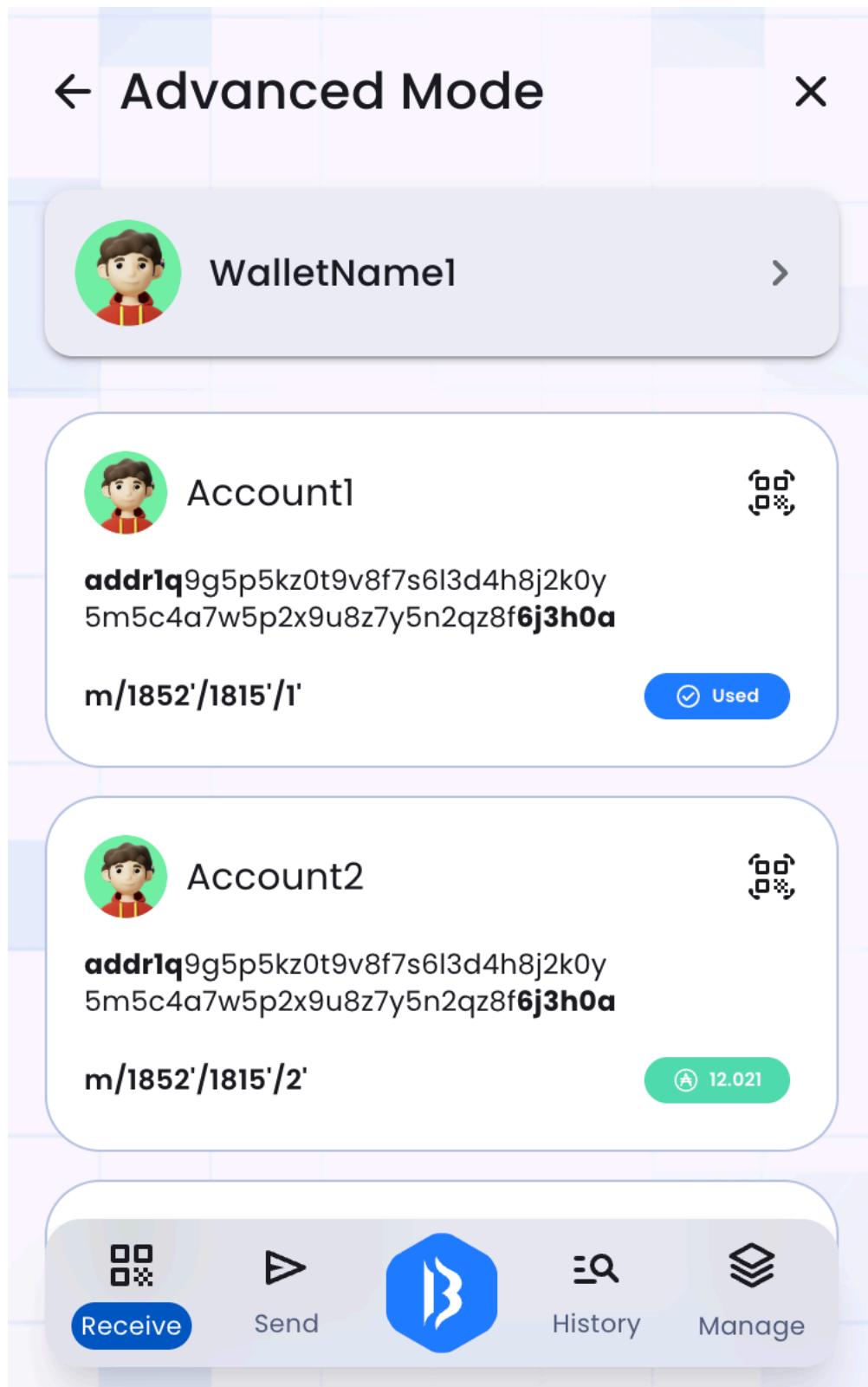


Figure 72 – Buriza UI - Browser: Receive assets advanced mode



iv.2.3 View Balance and Assets

Figure 73 – Buriza UI - Browser: View balance and assets

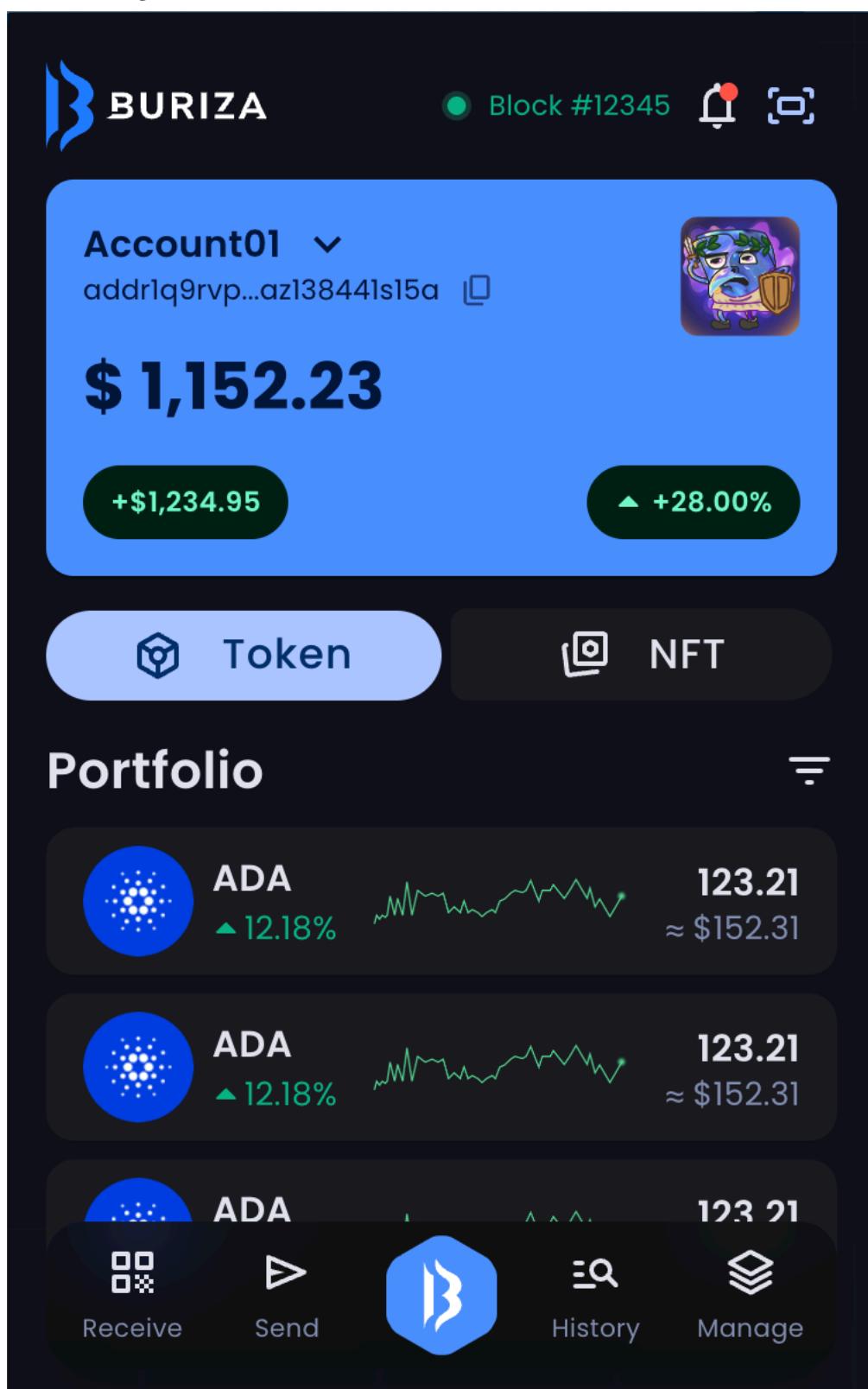


Figure 74 – Buriza UI - Browser: View balance and assets

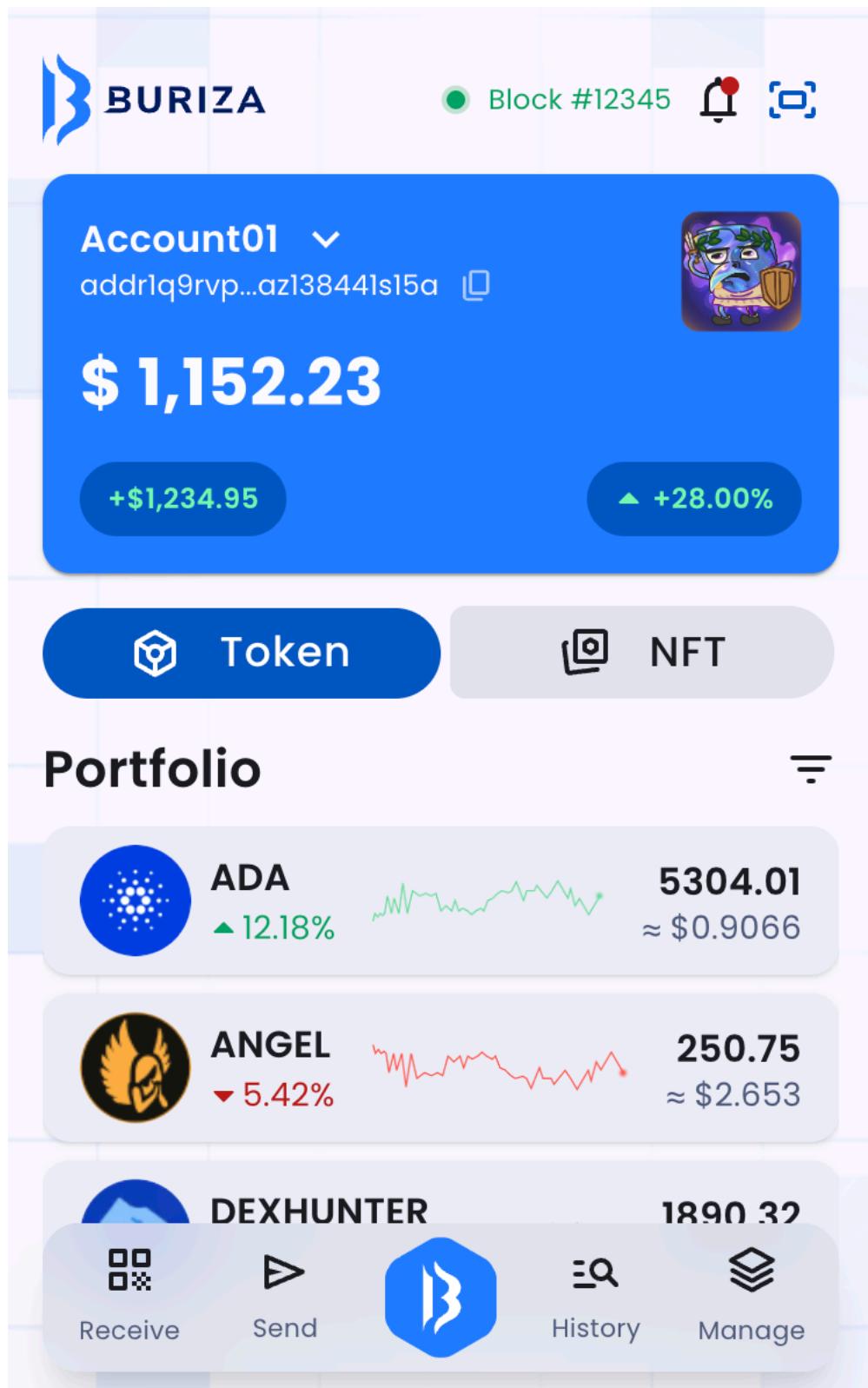


Figure 75 — Buriza UI - Browser: Header animation

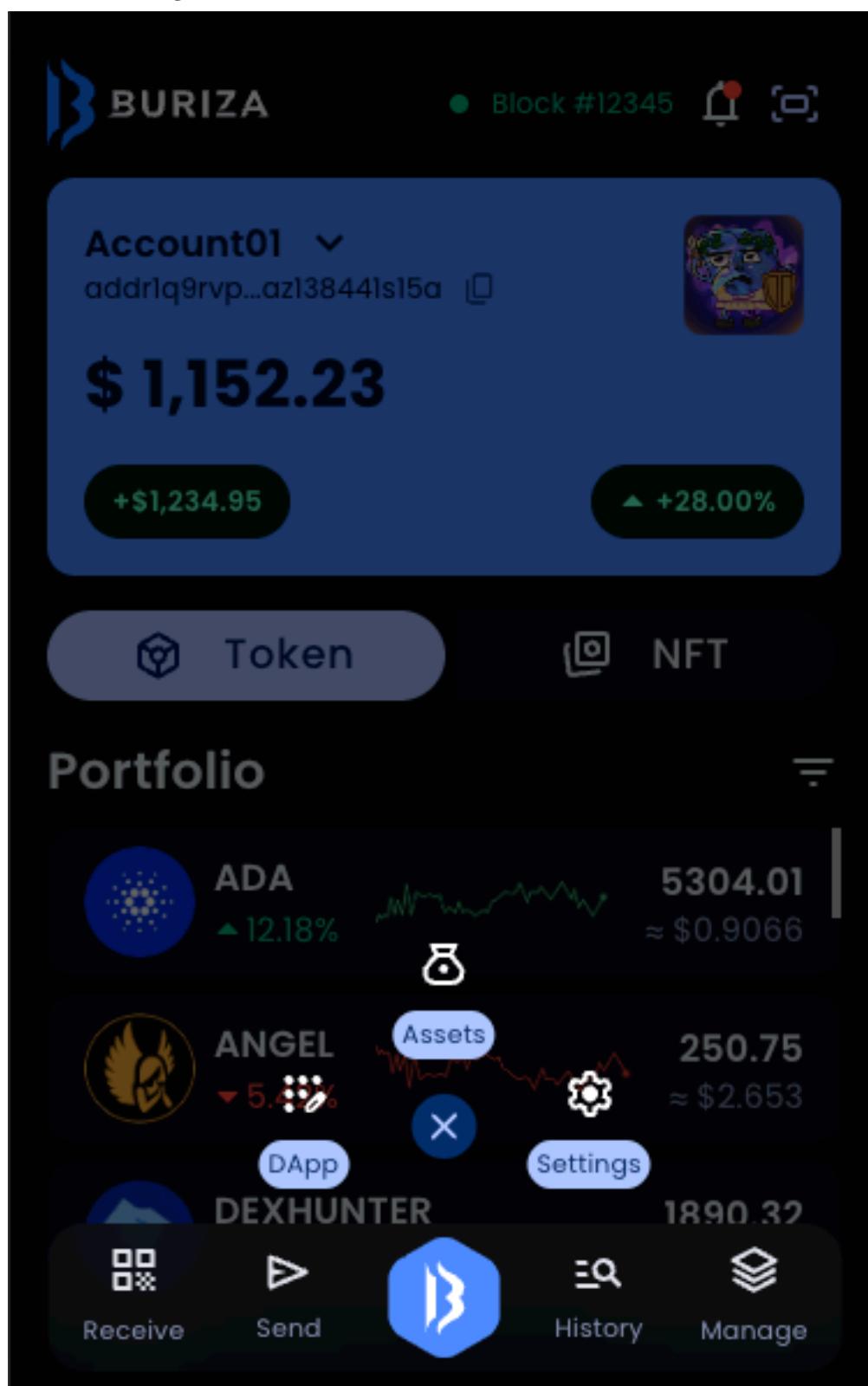
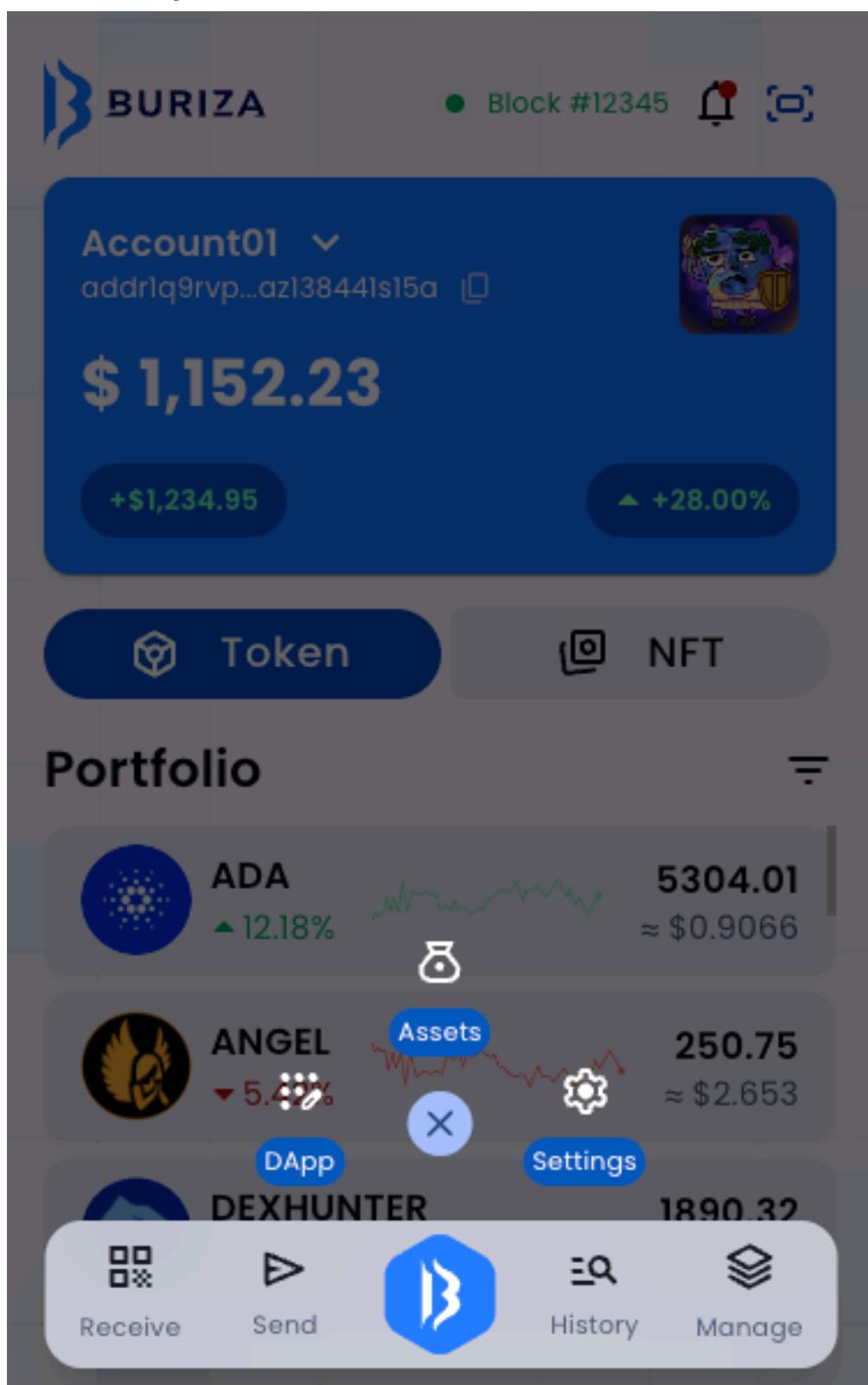


Figure 76 — Buriza UI - Browser: Header animation



iv.2.4 Asset Selection

Figure 77 – Buriza UI - Browser: Asset selection

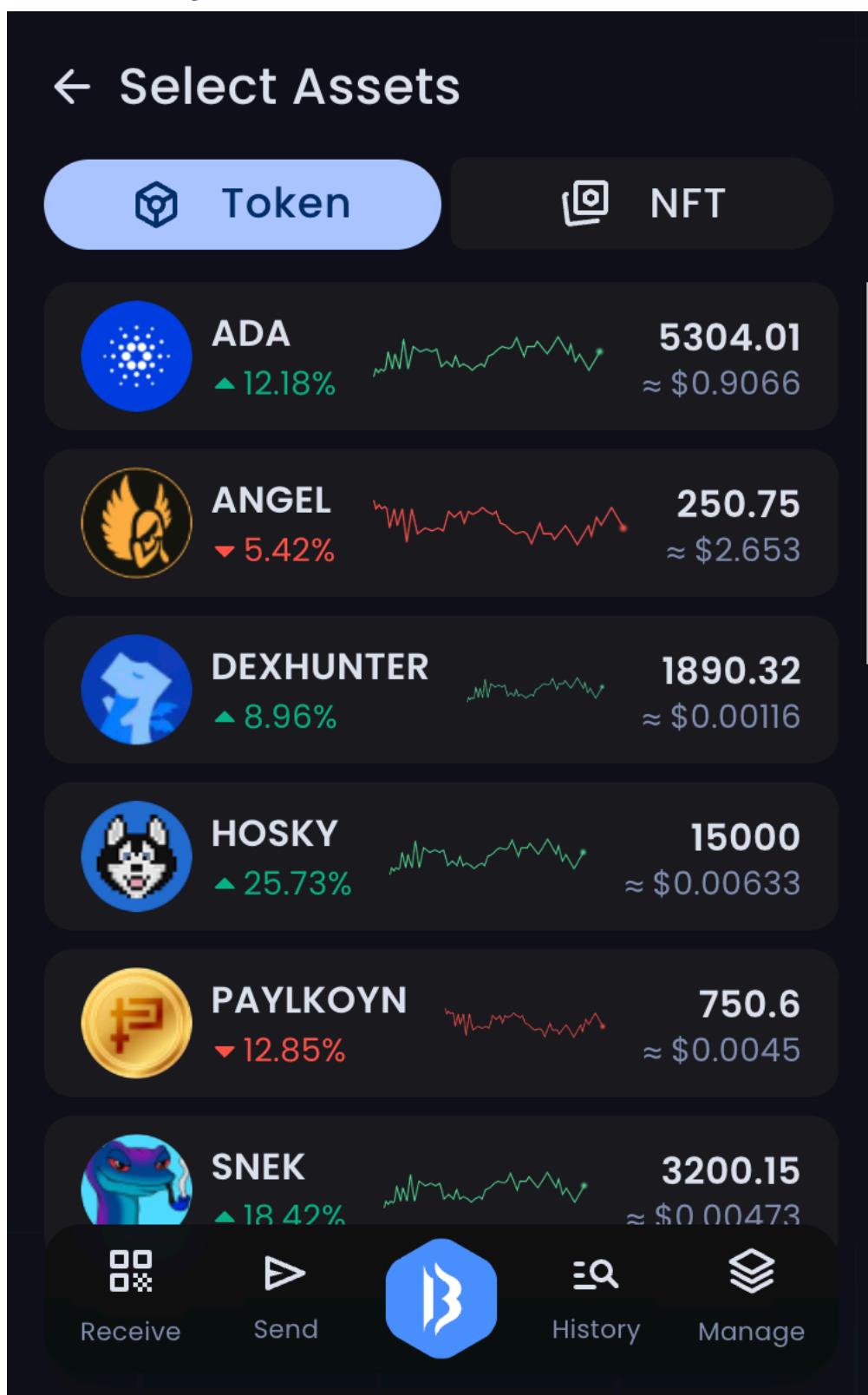
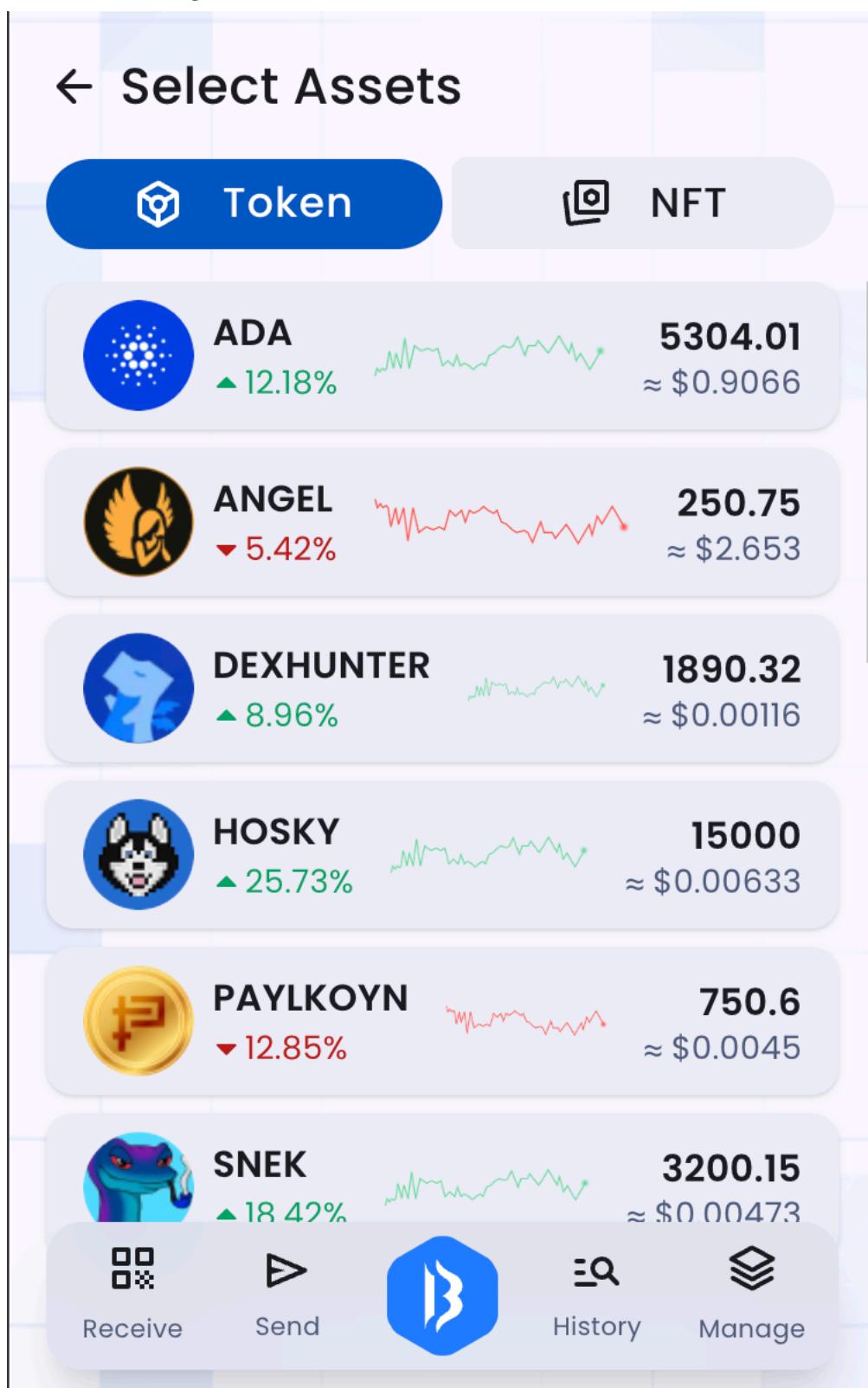


Figure 78 – Buriza UI - Browser: Asset selection



iv.2.5 Transaction History

Figure 79 – Buriza UI - Browser: Transaction history

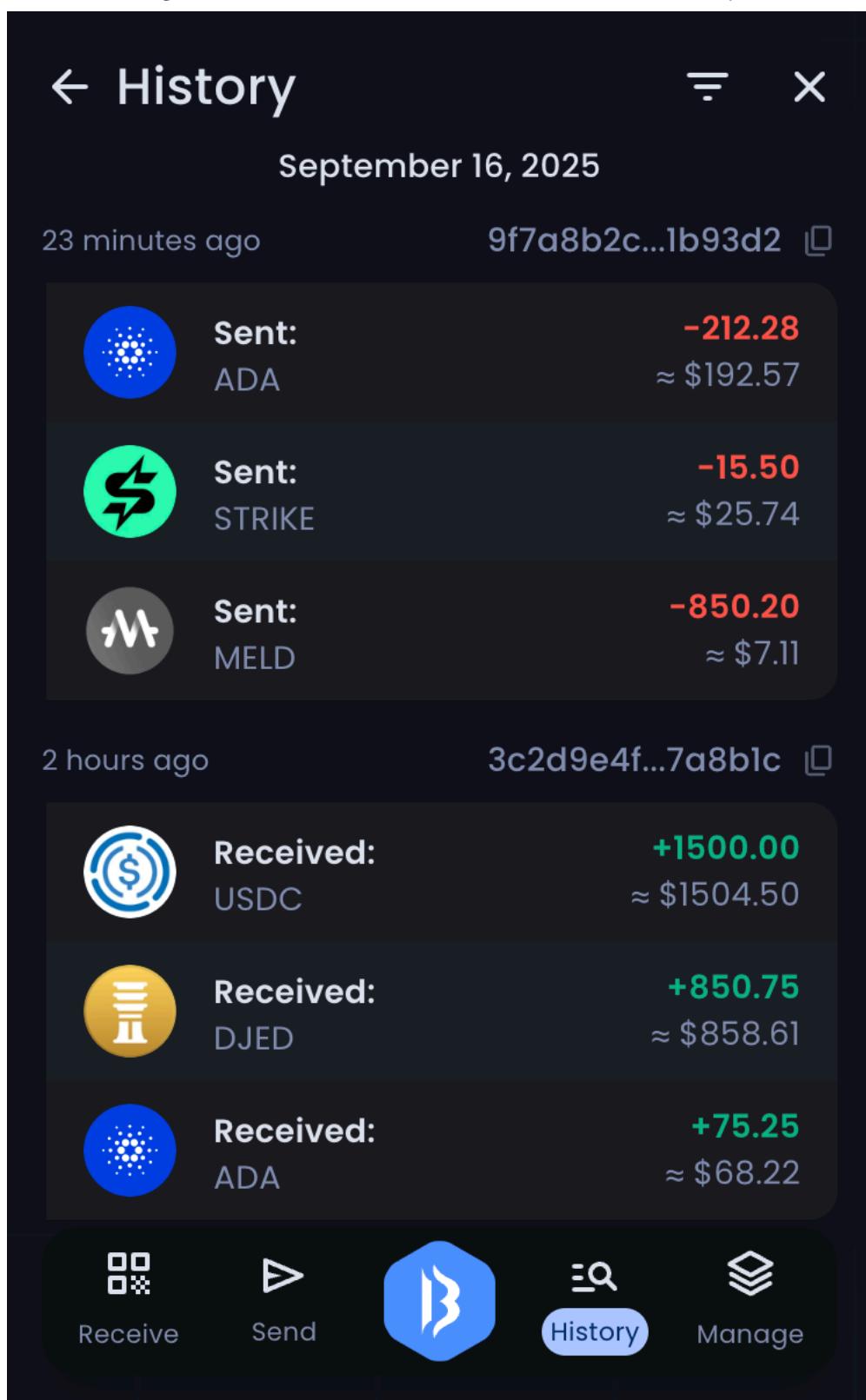


Figure 80 – Buriza UI - Browser: Transaction history

The screenshot shows the transaction history screen of the Buriza Wallet. At the top, there's a header with a back arrow, the word "History", a menu icon, and a close button. Below the header, the date "September 16, 2025" is displayed. The history is divided into two sections by time: "26 minutes ago" and "2 hours ago".

26 minutes ago:

- Sent:** ADA (9f7a8b2c...1b93d2) -212.28 ≈ \$192.57 (Icon: blue circle with dots)
- Sent:** STRIKE (9f7a8b2c...1b93d2) -15.50 ≈ \$25.74 (Icon: green lightning bolt)
- Sent:** MELD (9f7a8b2c...1b93d2) -850.20 ≈ \$7.11 (Icon: grey circle with M)

2 hours ago:

- Received:** USDC (3c2d9e4f...7a8b1c) +1500.00 ≈ \$1504.50 (Icon: blue circle with dollar sign)
- Received:** DJED (3c2d9e4f...7a8b1c) +850.75 ≈ \$858.61 (Icon: gold pyramid)
- Received:** ADA (3c2d9e4f...7a8b1c) +75.25 ≈ \$68.22 (Icon: blue circle with dots)

At the bottom, there are five navigation buttons: "Receive" (qr code icon), "Send" (play icon), a central logo (blue hexagon with a white stylized letter B), "History" (blue rounded rectangle with white text), and "Manage" (stacked squares icon).

Figure 81 – Buriza UI - Browser: Transaction summary

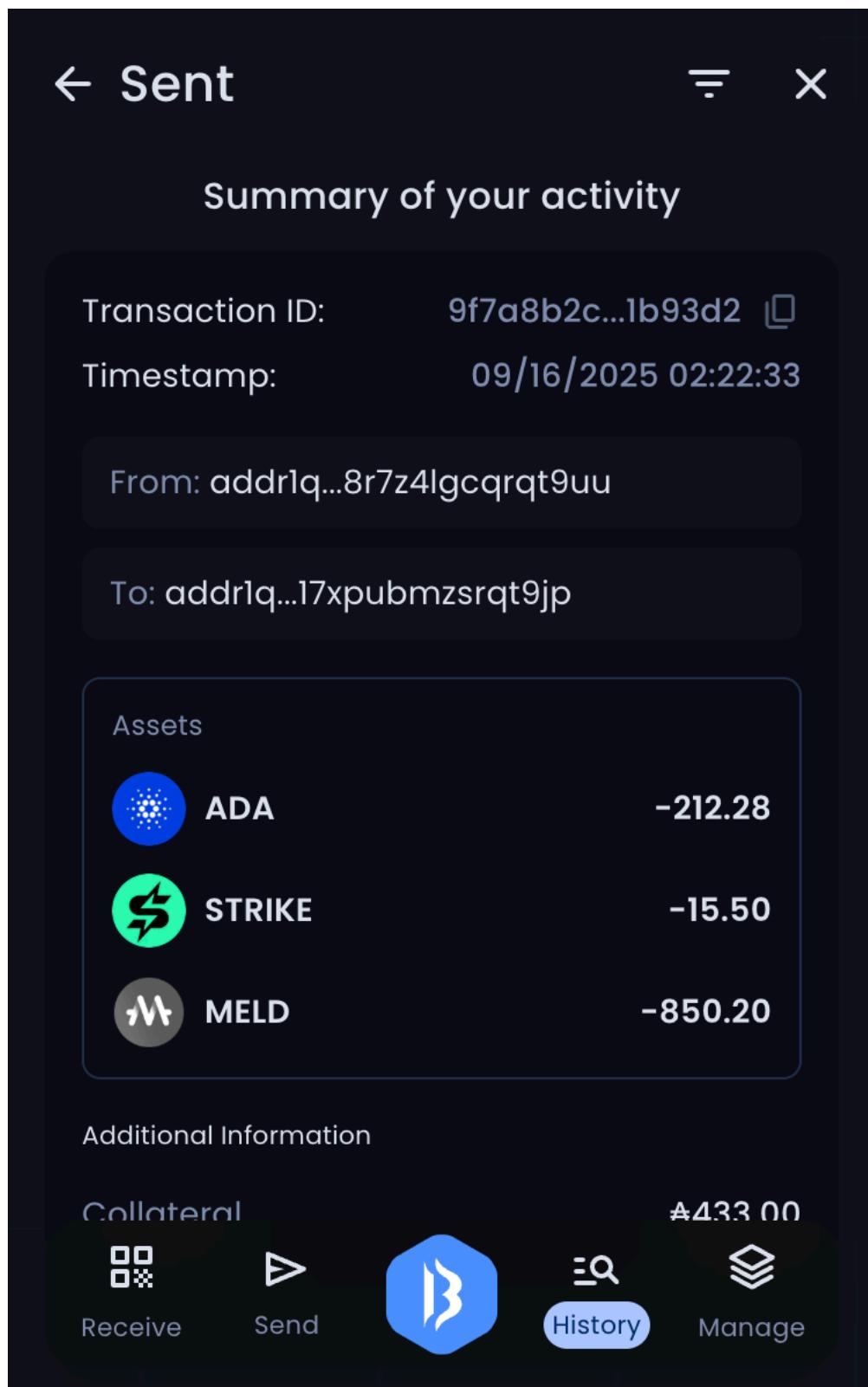
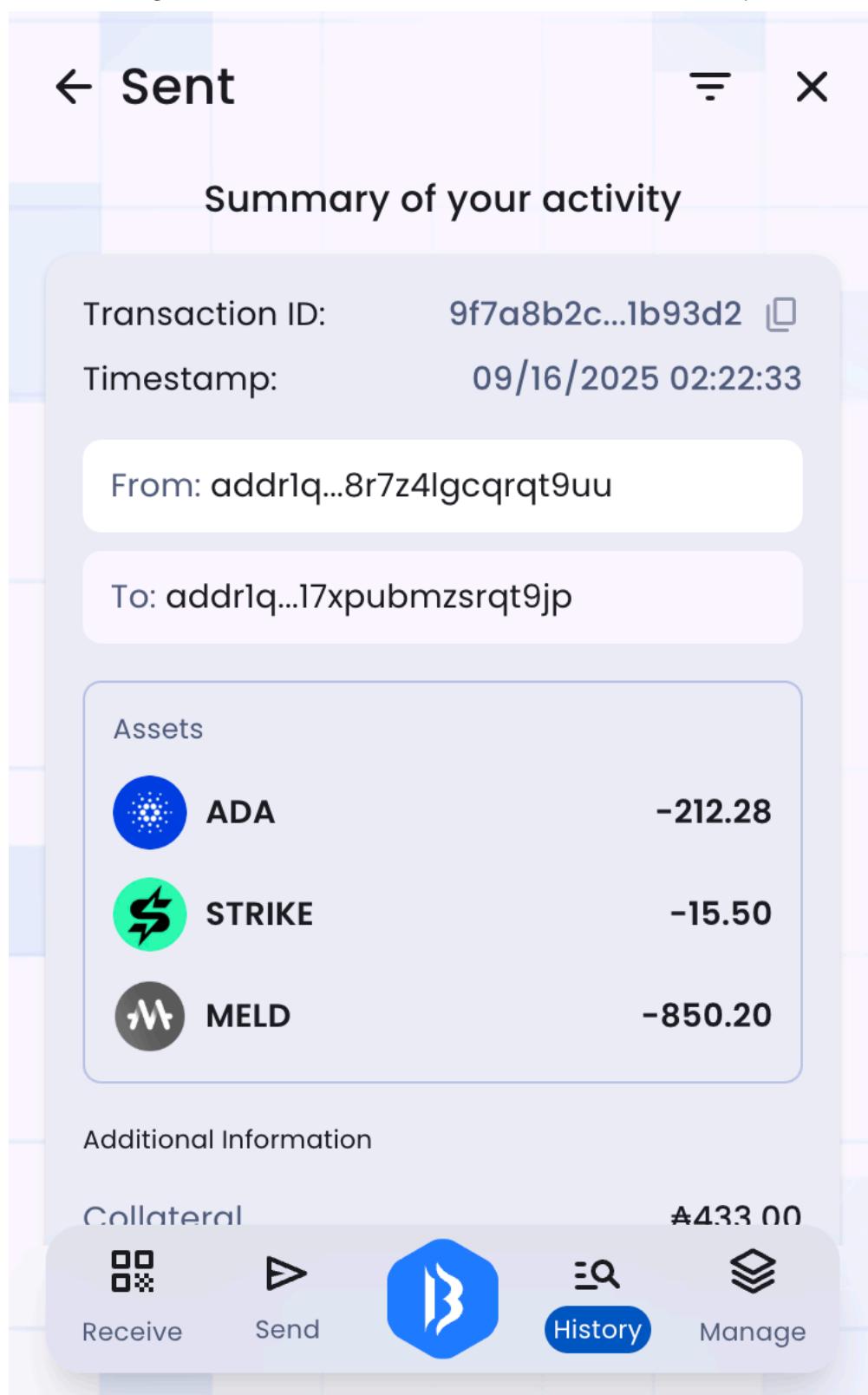


Figure 82 – Buriza UI - Browser: Transaction summary



iv.2.6 Full Node Setup

Figure 83 – Buriza UI - Browser: Settings

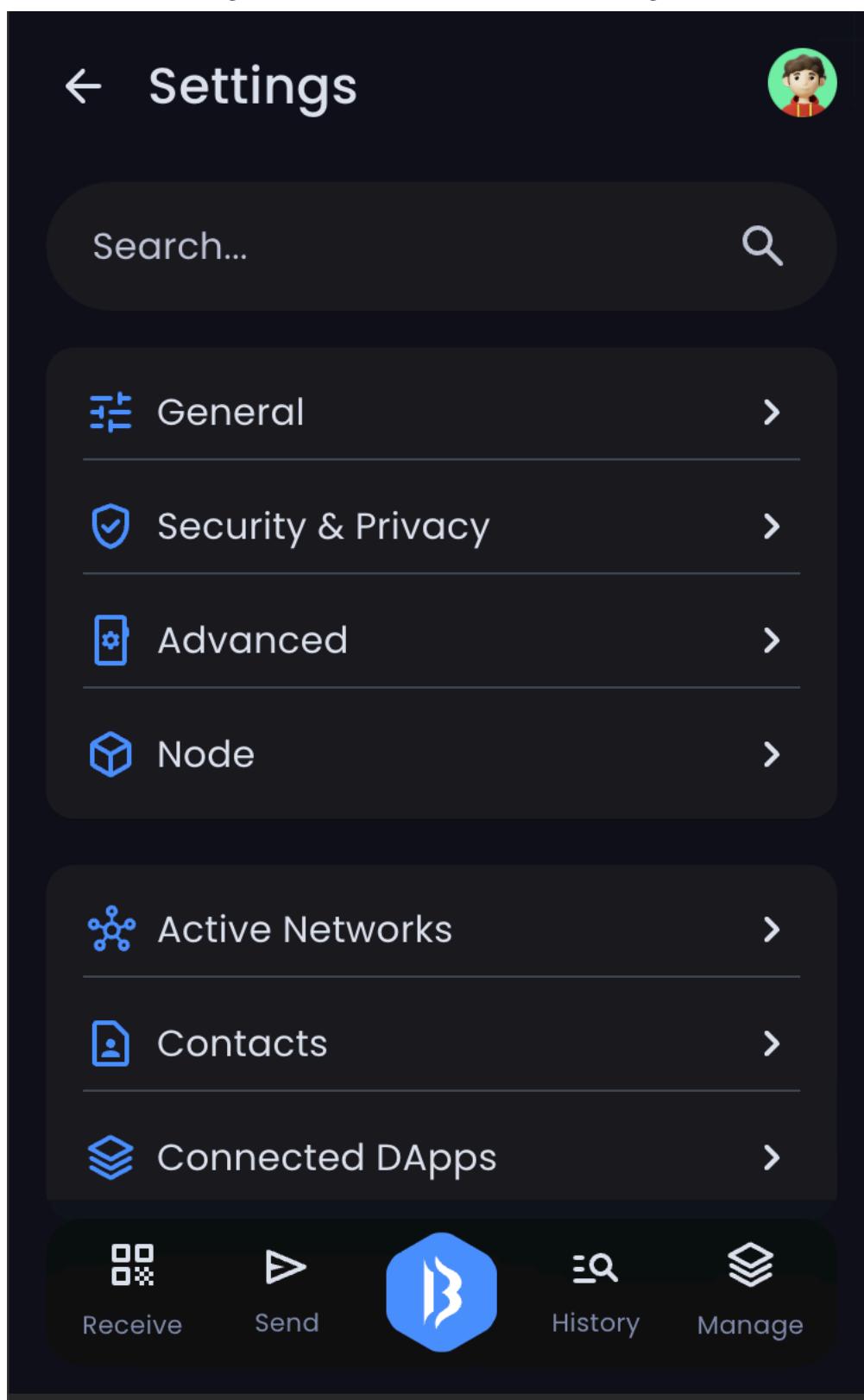


Figure 84 – Buriza UI - Browser: Settings

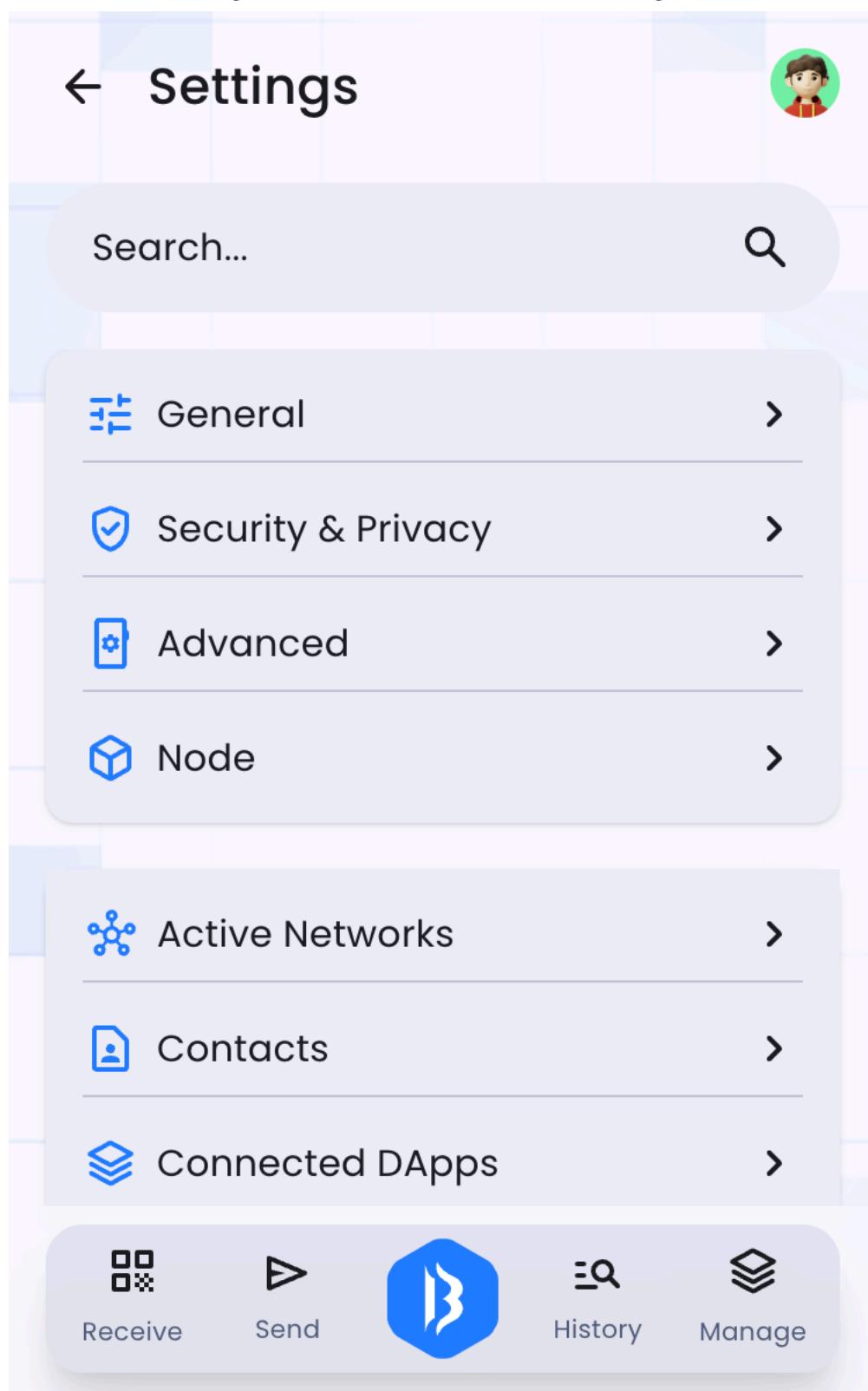


Figure 85 — Buriza UI - Browser: Node setup with full node

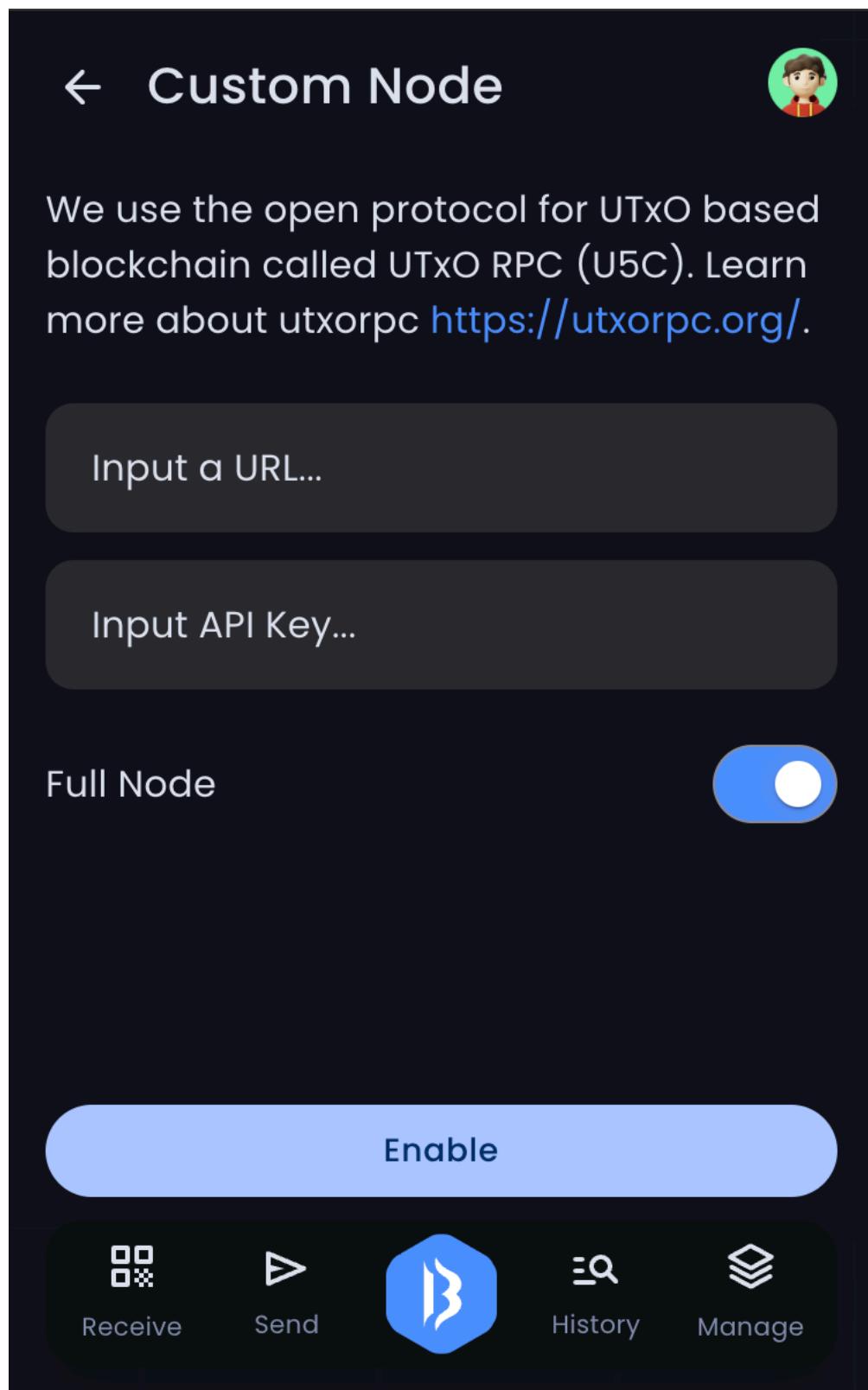


Figure 86 — Buriza UI - Browser: Node setup with full node

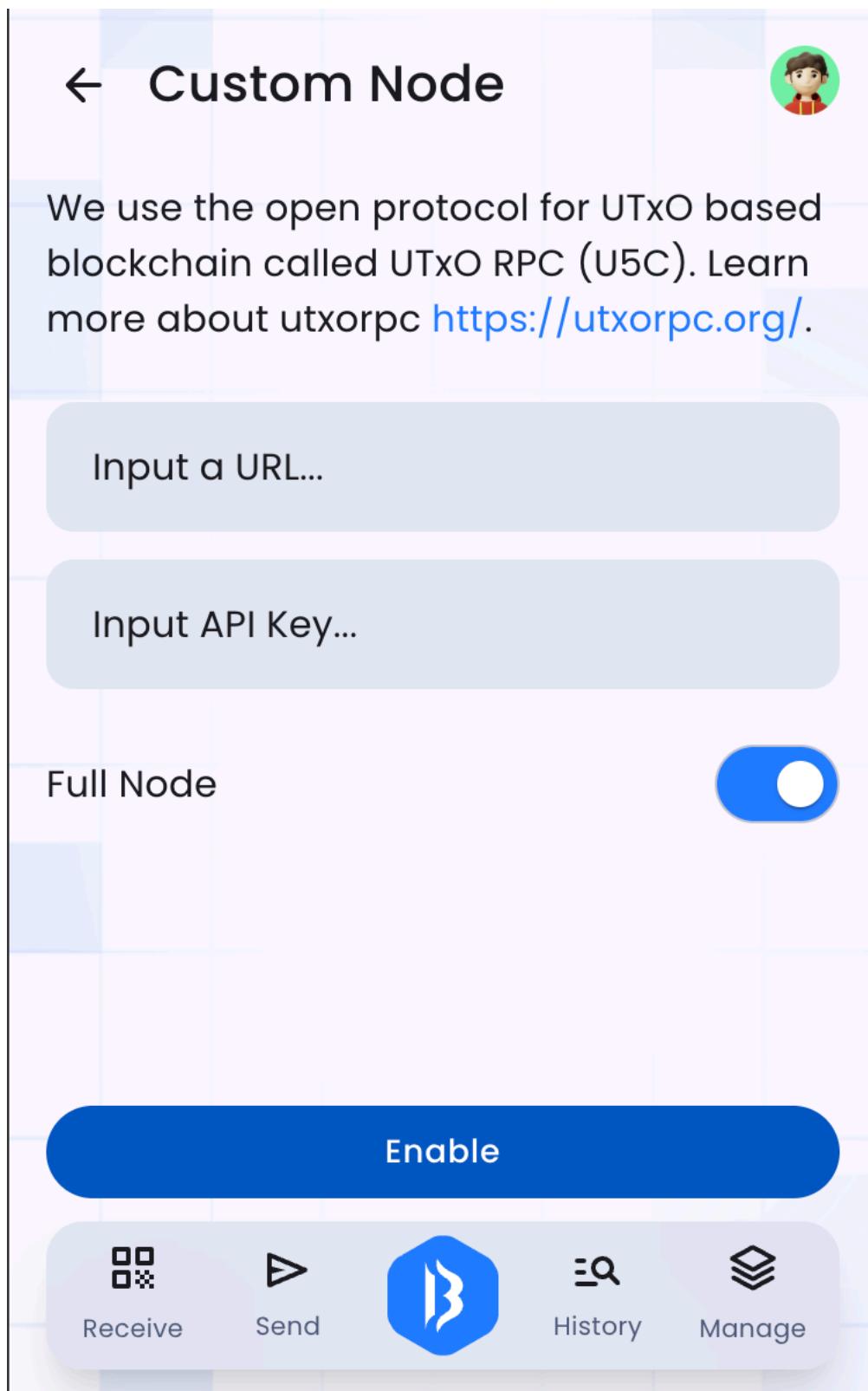


Figure 87 – Buriza UI - Browser: Node setup

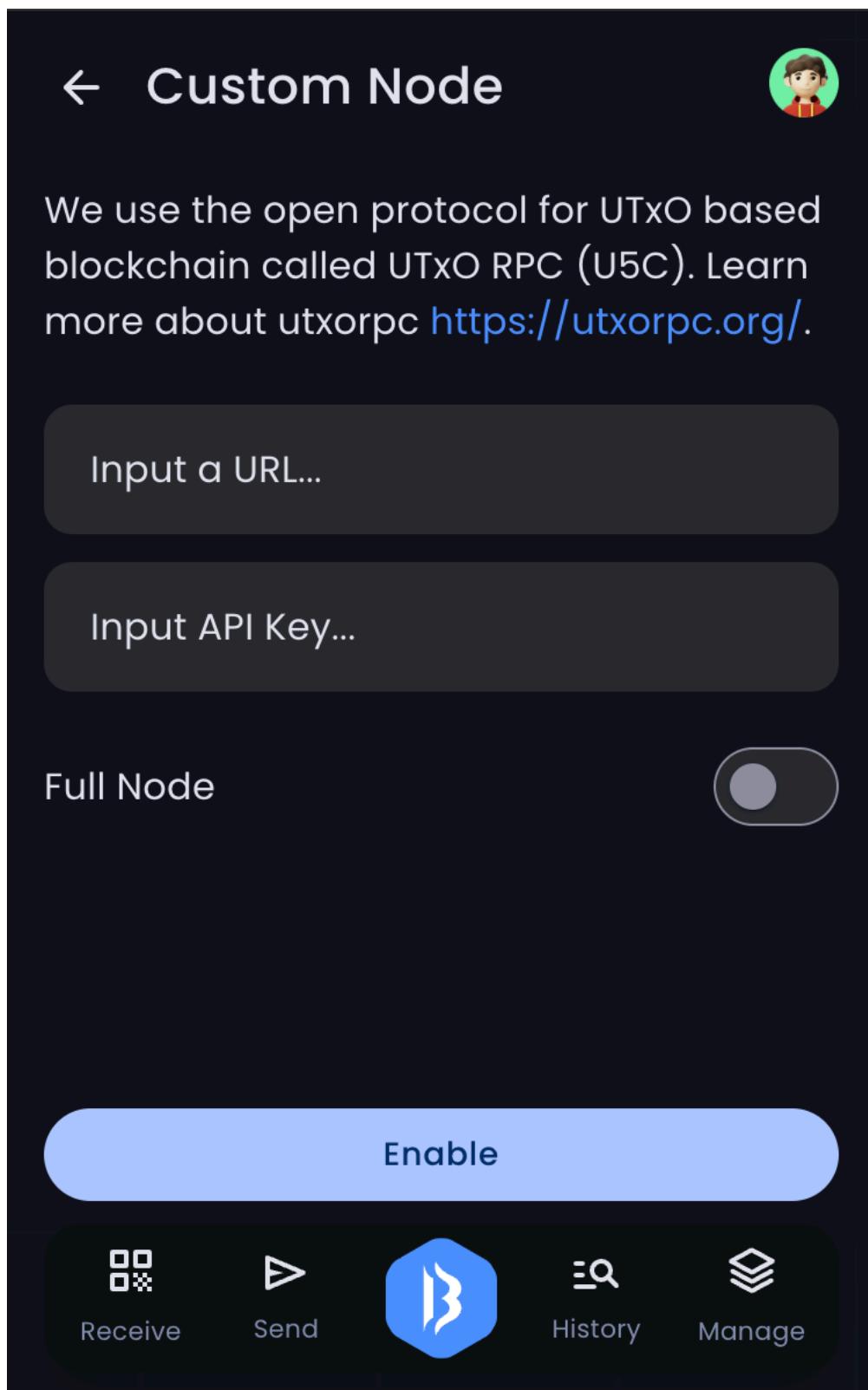
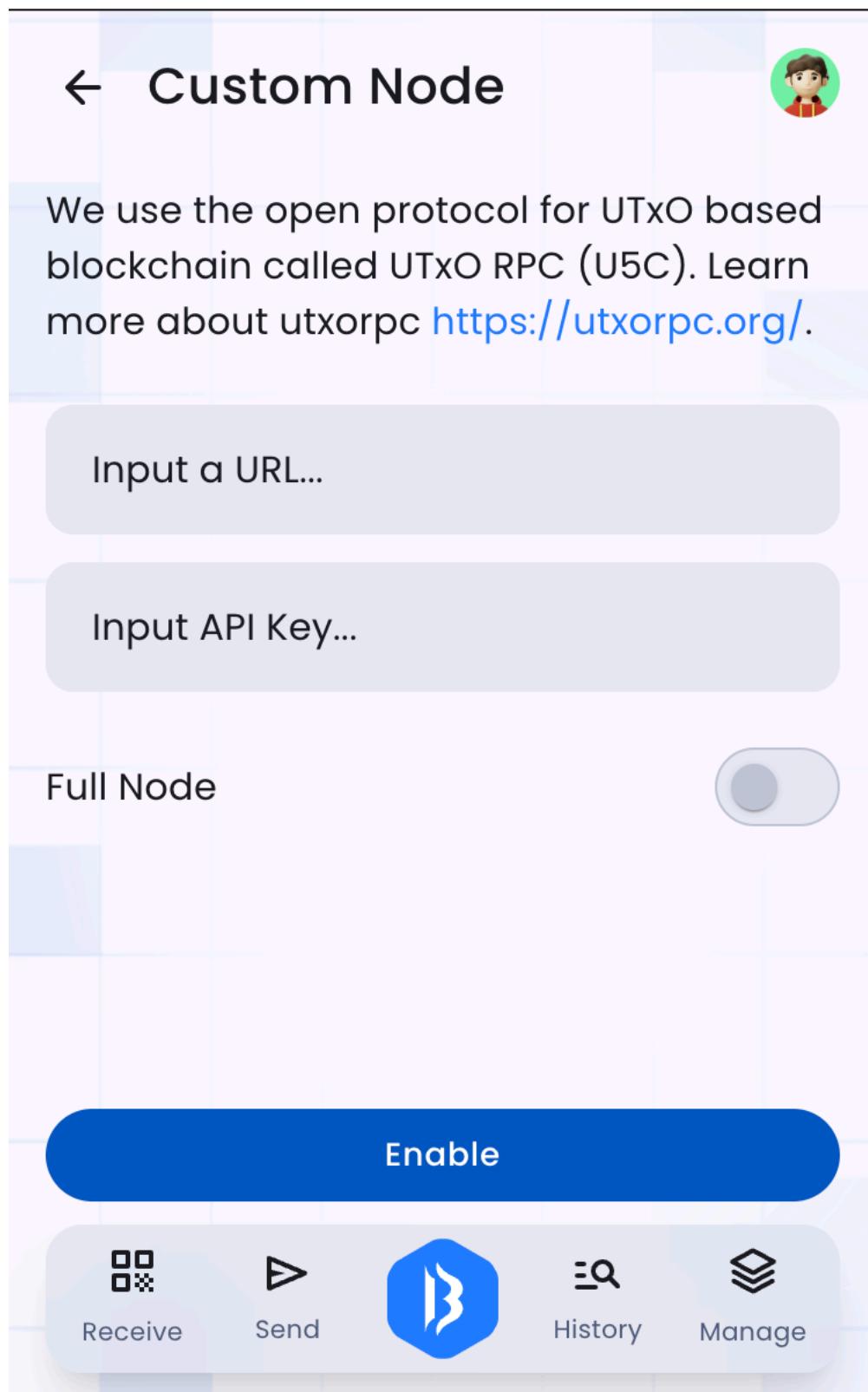


Figure 88 – Buriza UI - Browser: Node setup



iv.2.7 dApp Access

Figure 89 – Buriza UI - Browser: dApp explorer

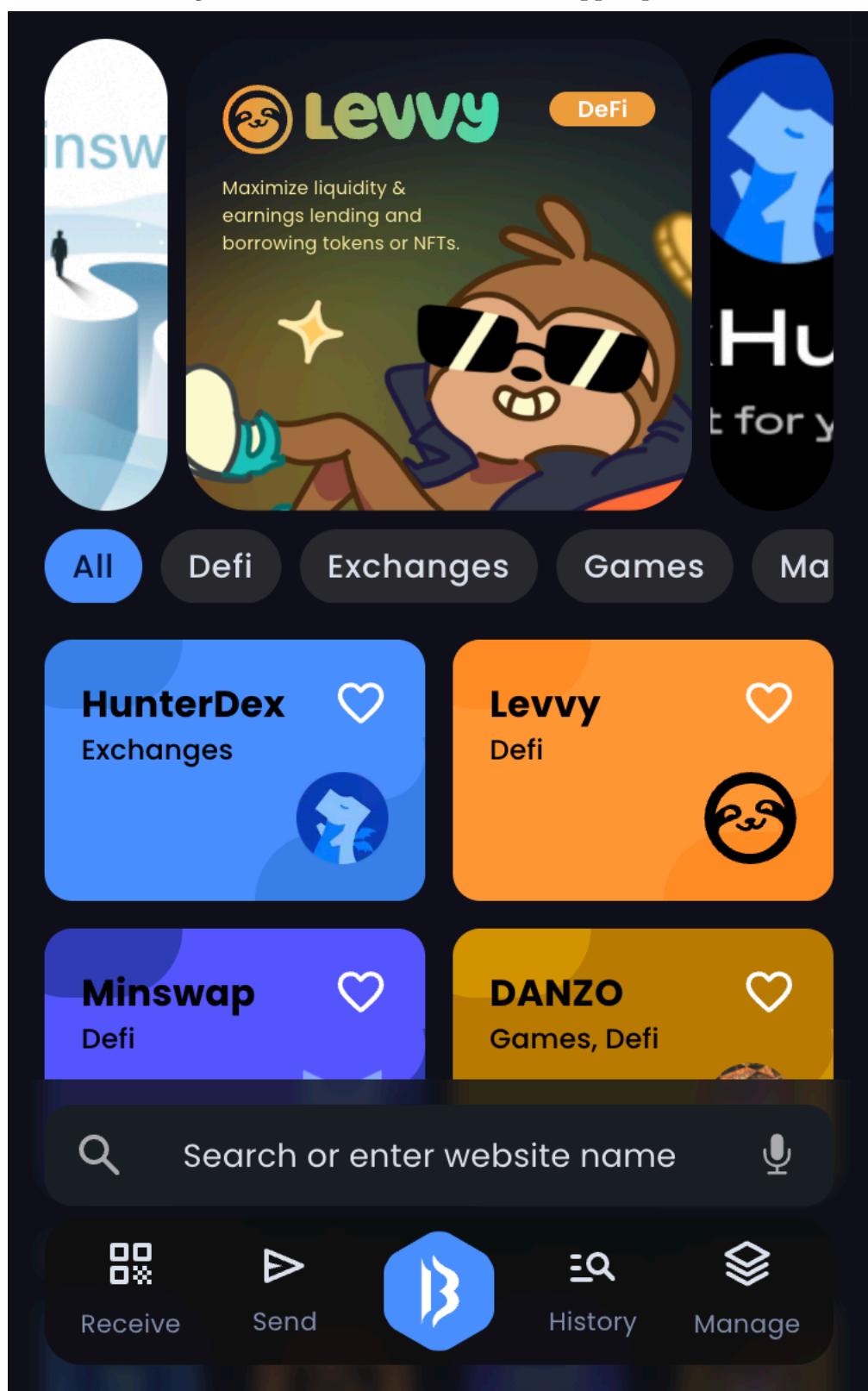


Figure 90 – Buriza UI - Browser: dApp explorer

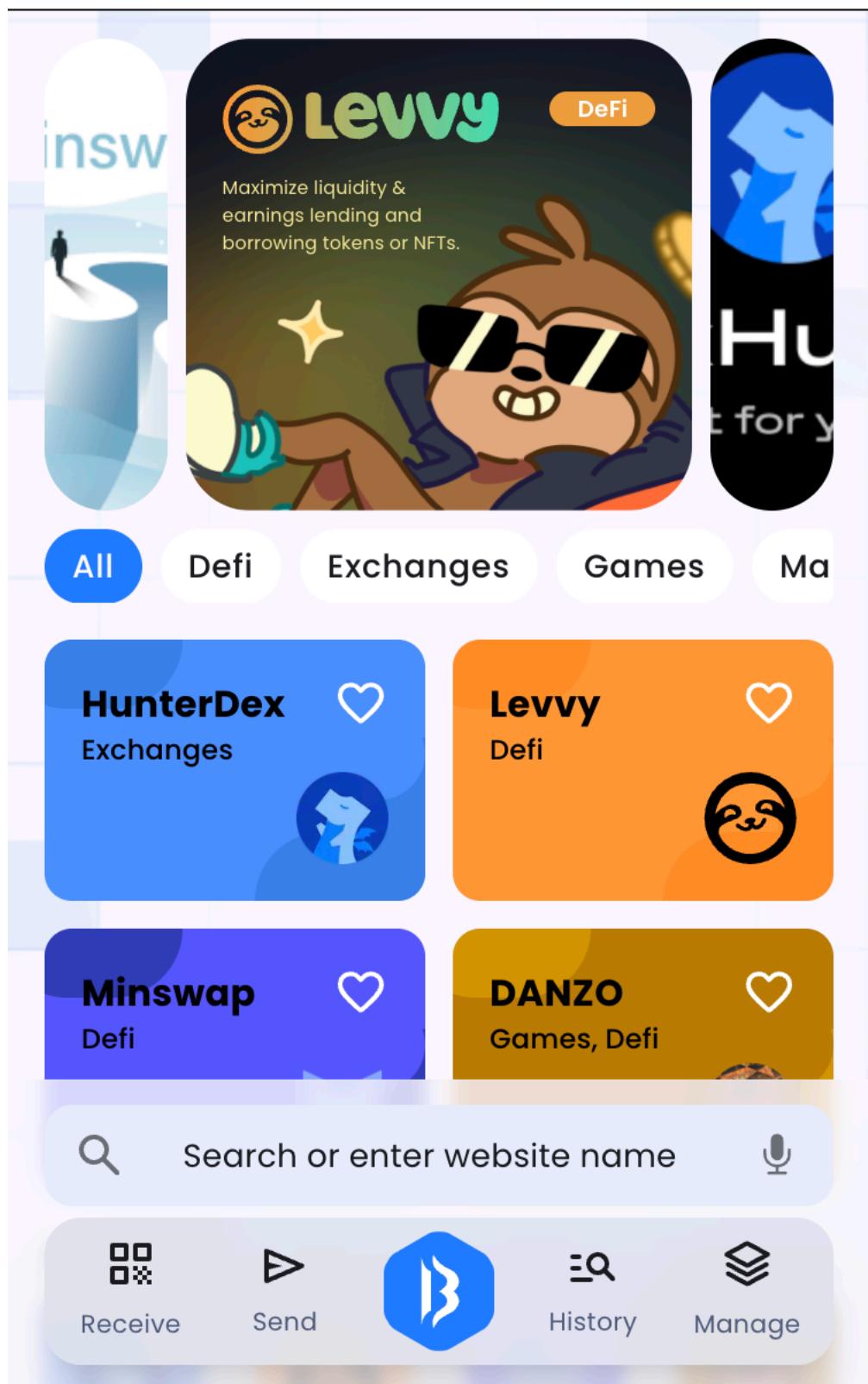


Figure 91 — Buriza UI - Browser: dApp authorization request

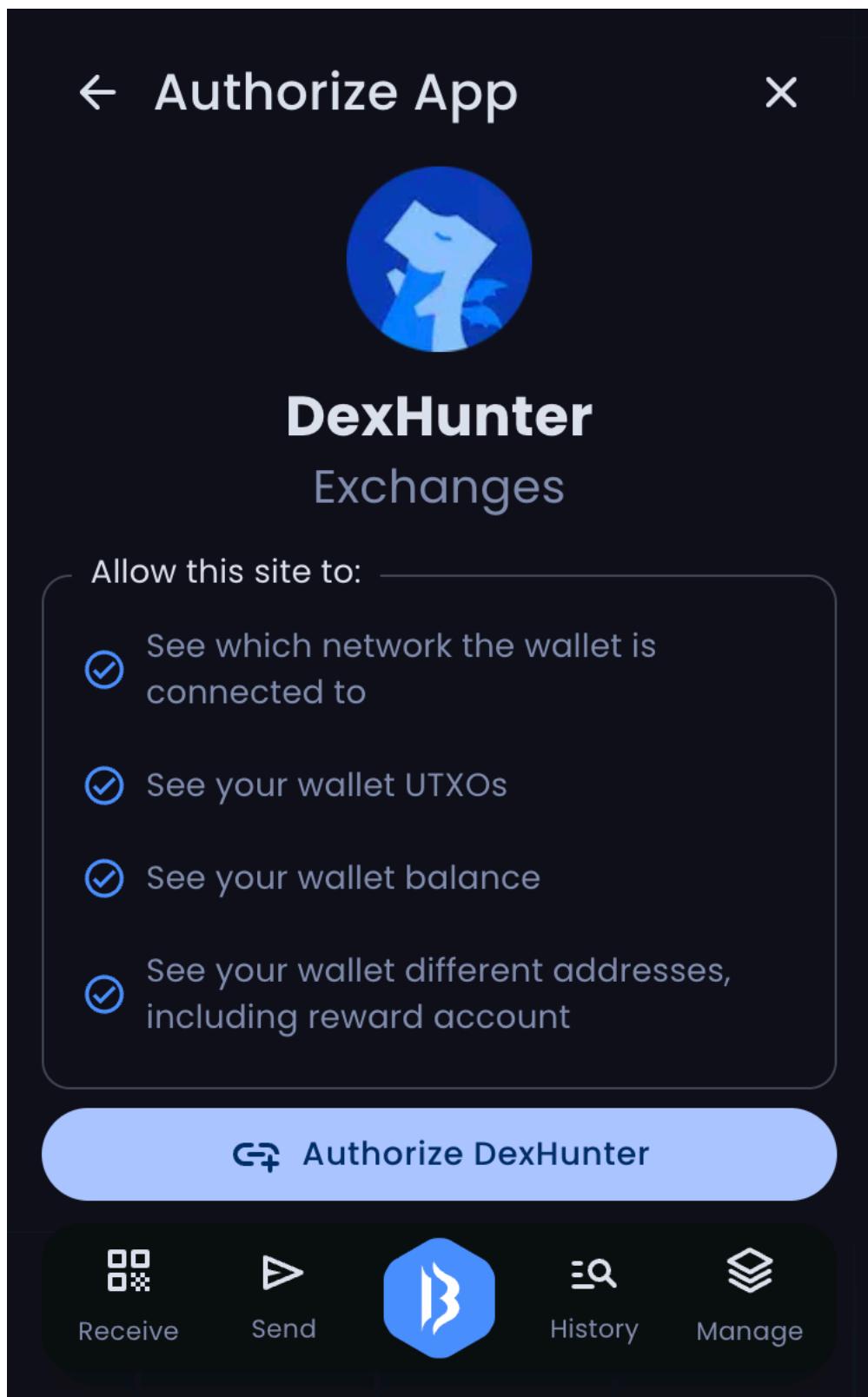
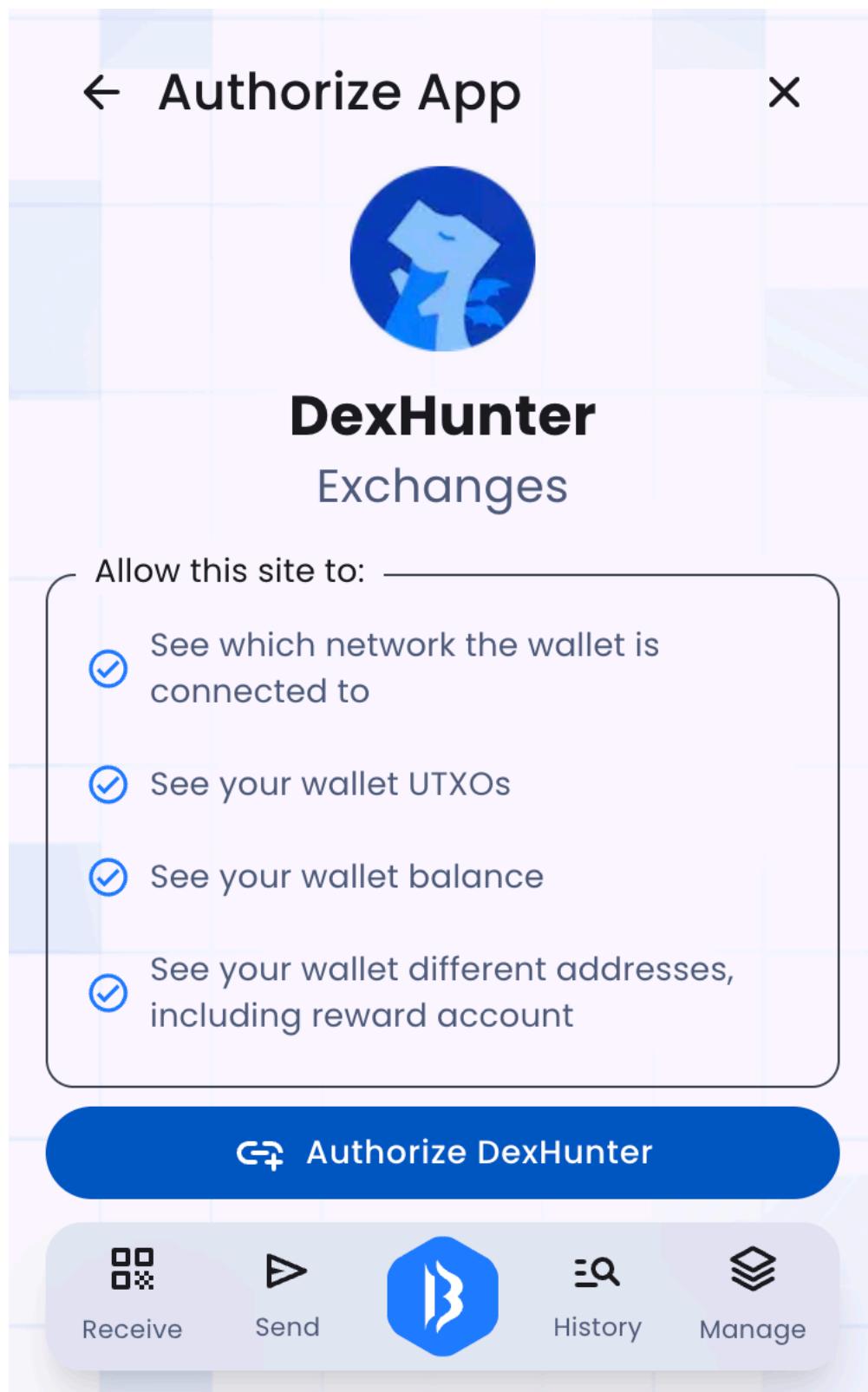


Figure 92 – Buriza UI - Browser: dApp authorization request



iv.3 Desktop

iv.3.1 Send Assets

Figure 93 – Buriza UI - Desktop: Send assets

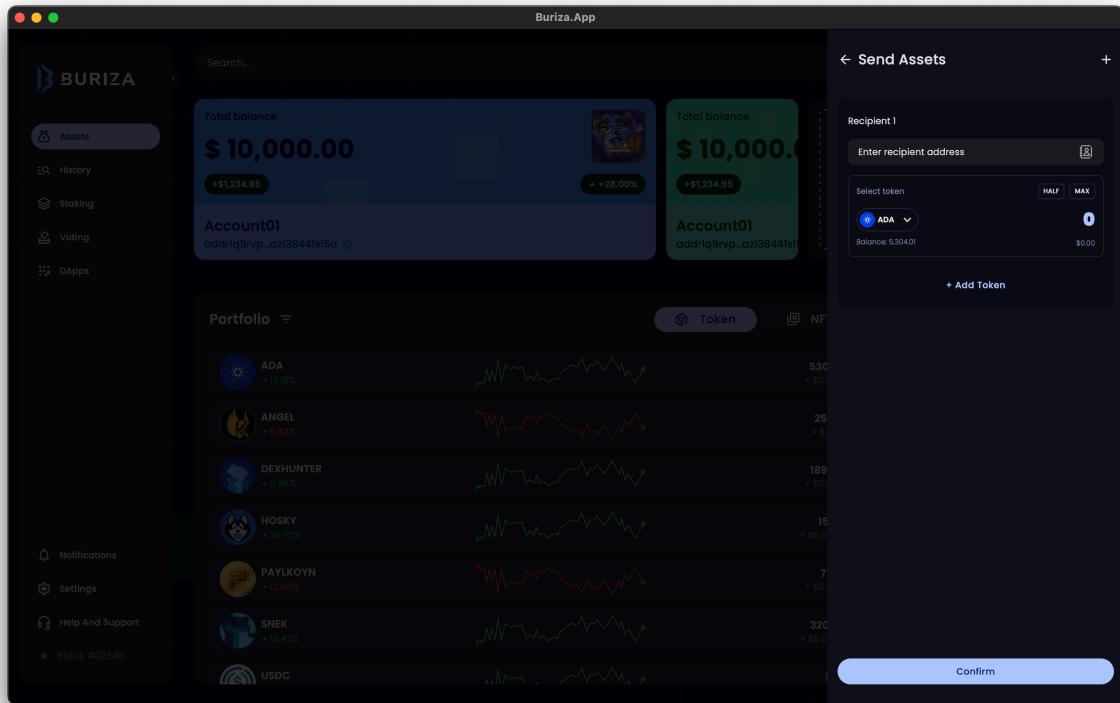


Figure 94 – Buriza UI - Desktop: Send assets

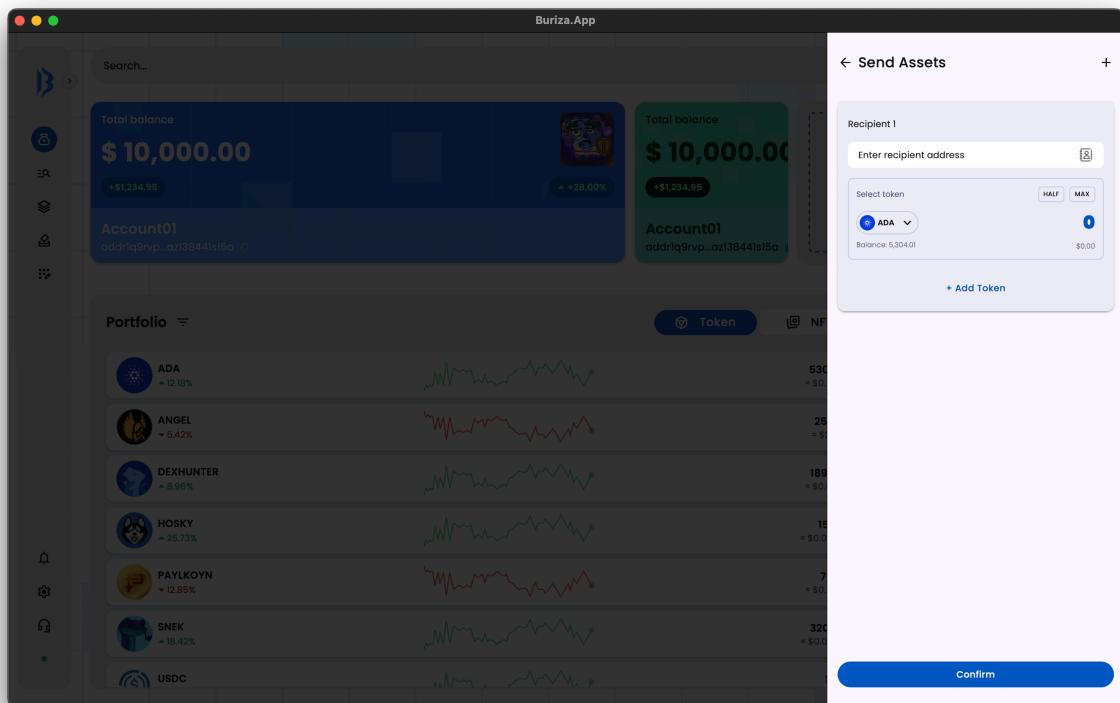


Figure 95 – Buriza UI - Desktop: Send assets with an additional recipient

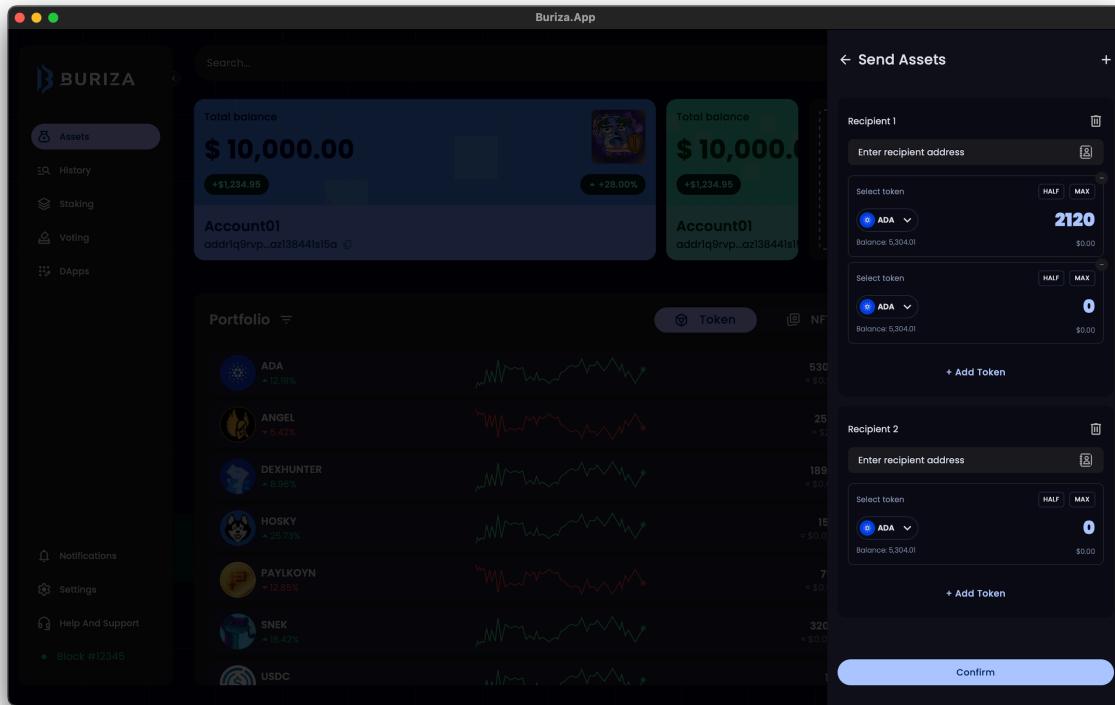


Figure 96 – Buriza UI - Desktop: Send assets with an additional recipient

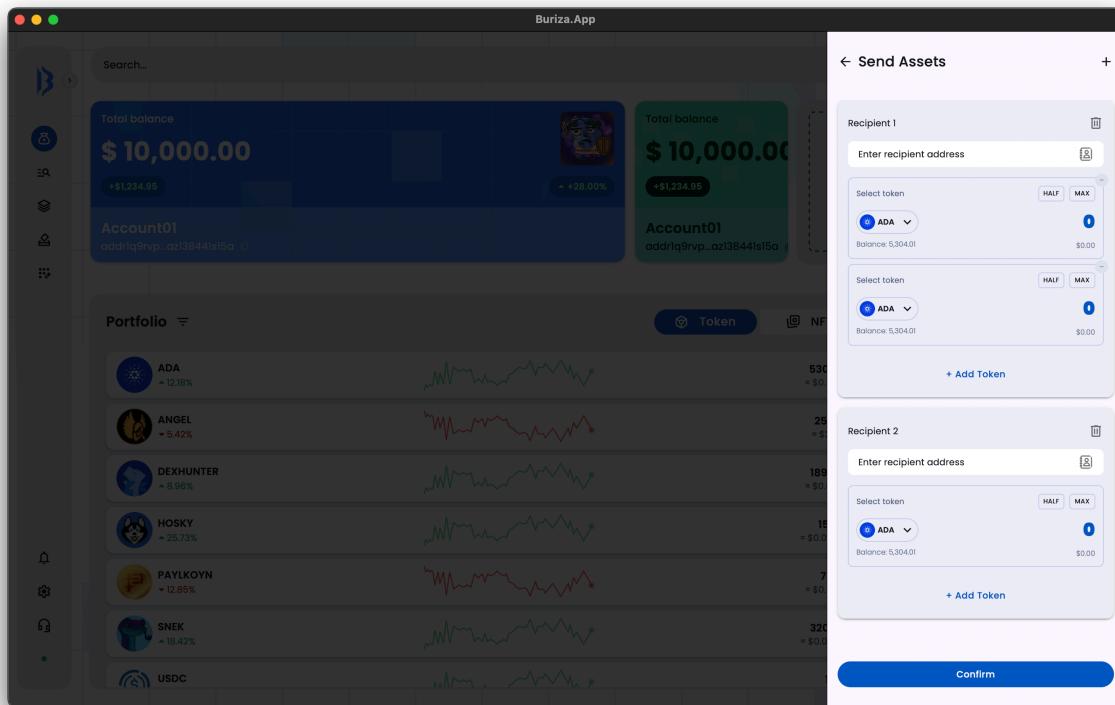


Figure 97 – Buriza UI - Desktop: Send assets with an additional token

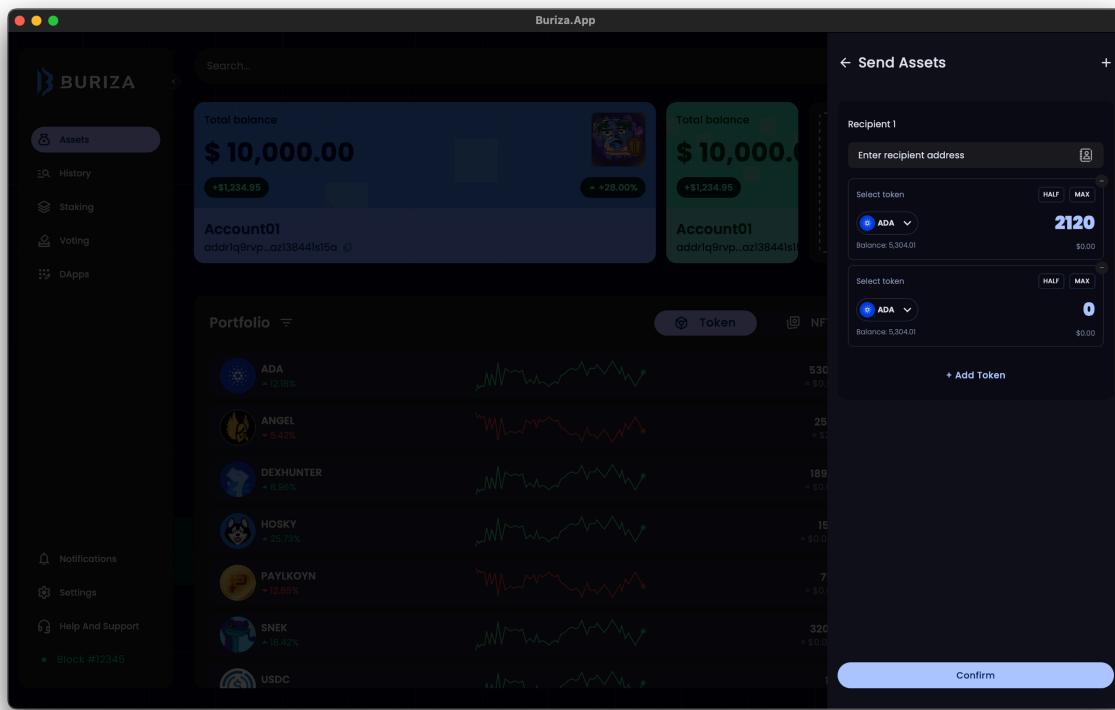


Figure 98 – Buriza UI - Desktop: Send assets with an additional token

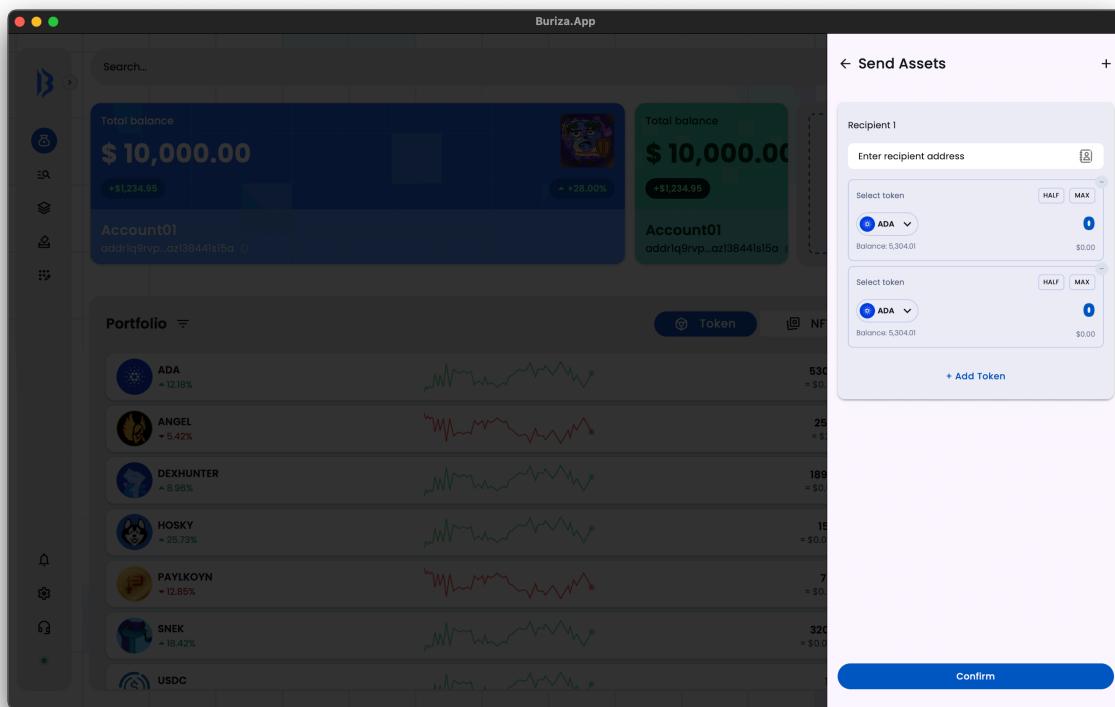


Figure 99 – Buriza UI - Desktop: Send assets transaction summary with added recipient

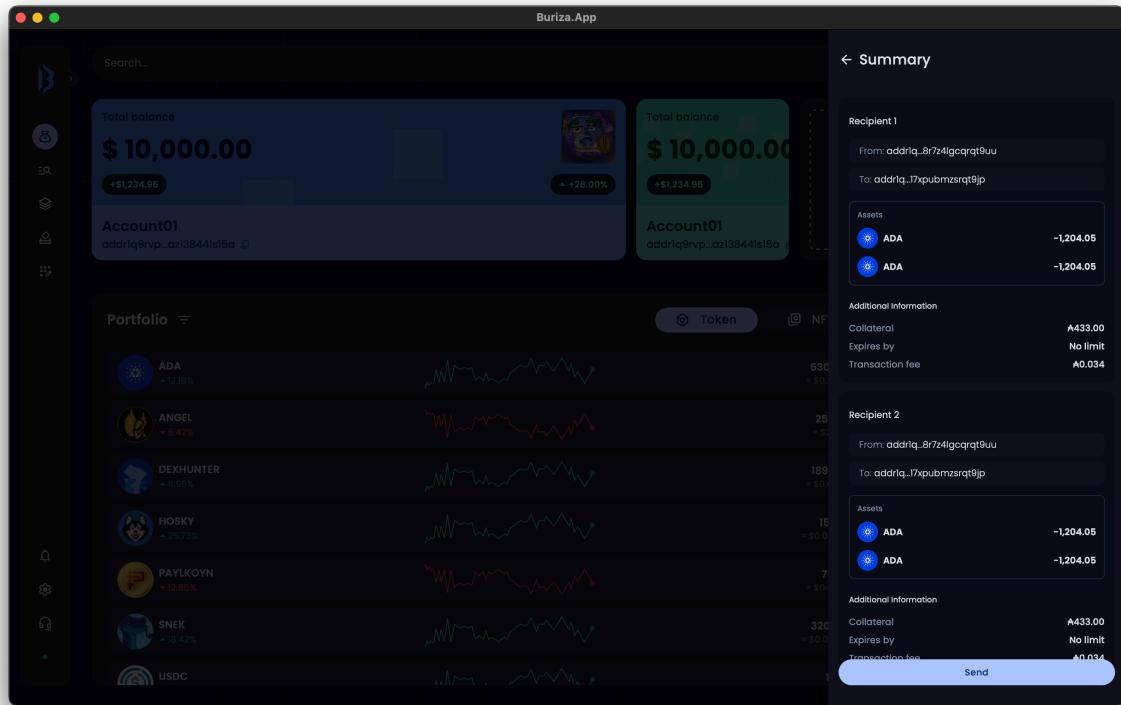


Figure 100 – Buriza UI - Desktop: Send assets transaction summary with added recipient

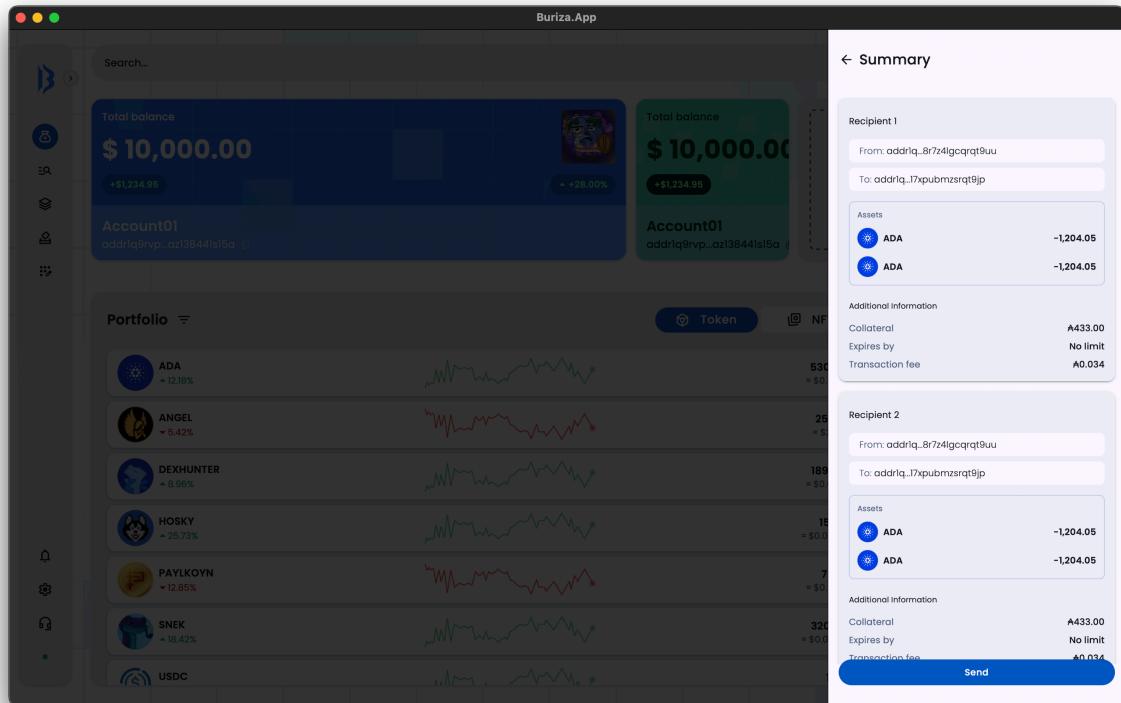


Figure 101 – Buriza UI - Desktop: Send assets transaction summary

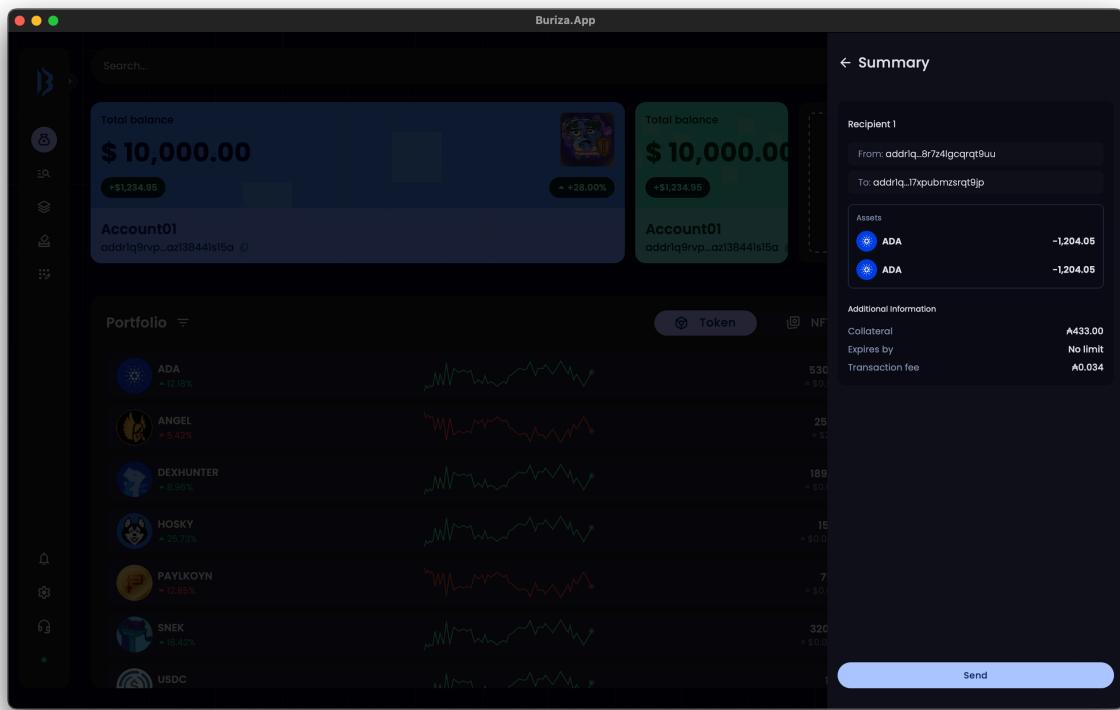


Figure 102 – Buriza UI - Desktop: Send assets transaction summary

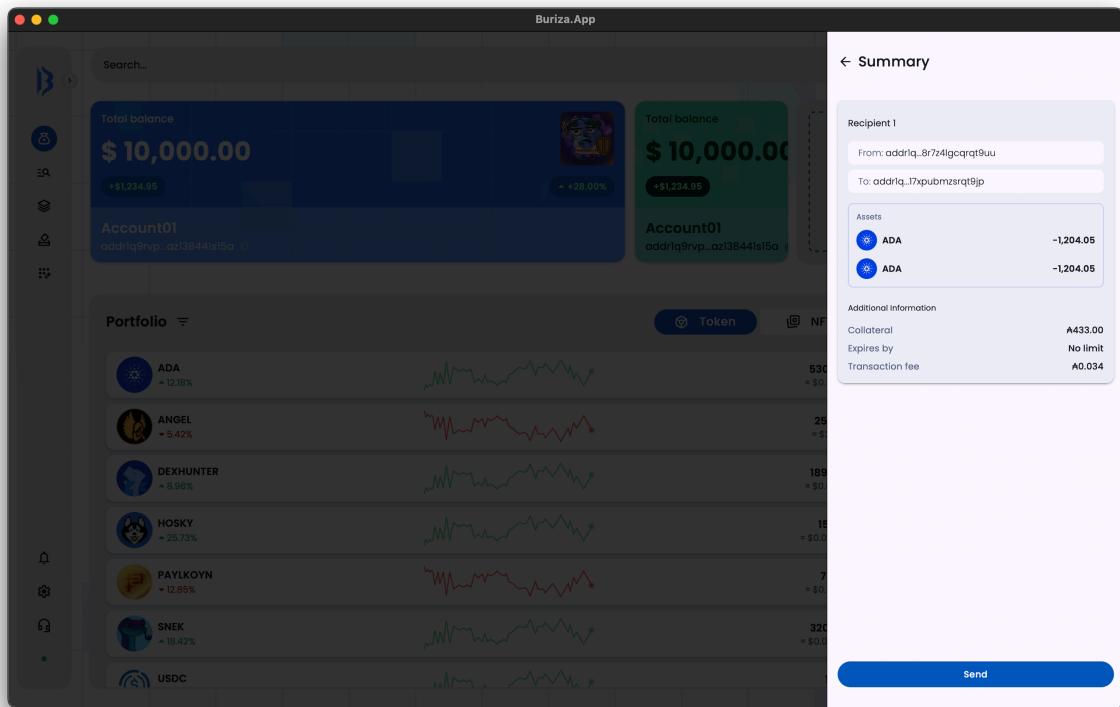


Figure 103 – Buriza UI - Desktop: Send assets success screen

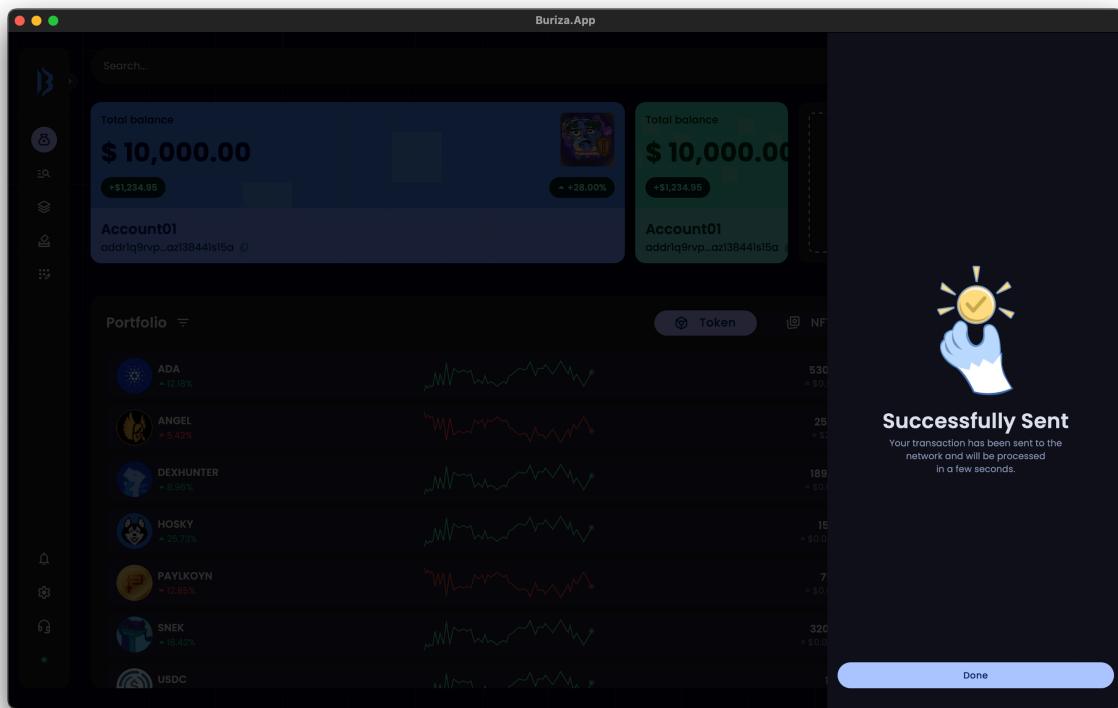
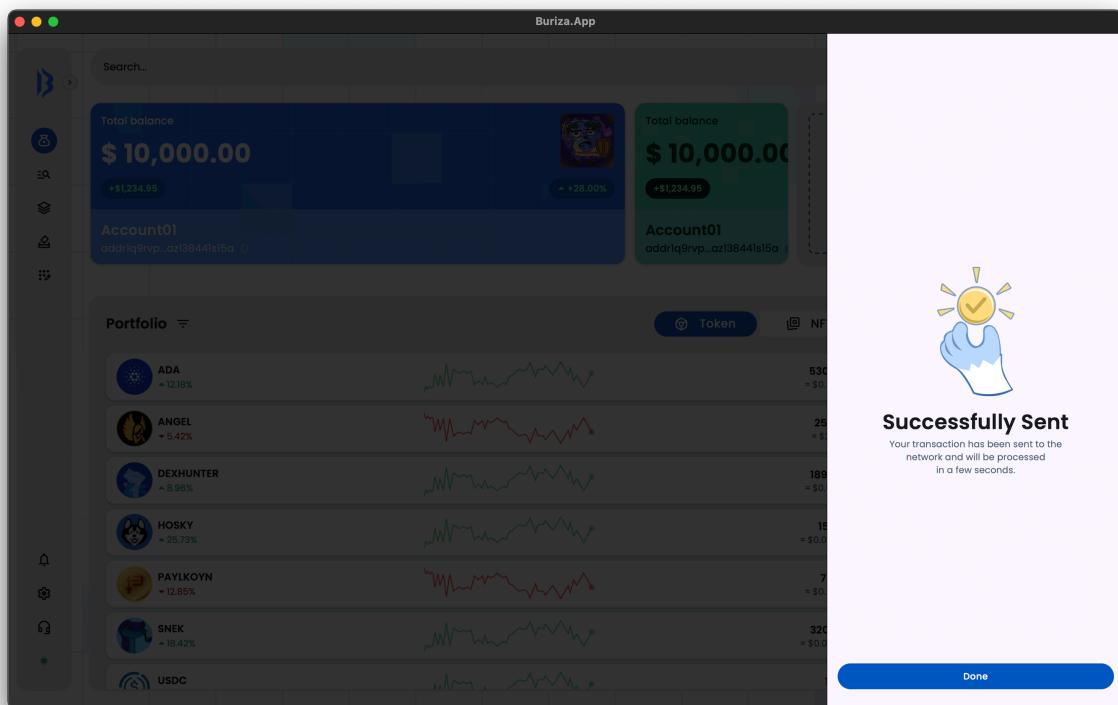


Figure 104 – Buriza UI - Desktop: Send assets success screen



iv.3.2 Receiving Assets

Figure 105 – Buriza UI - Desktop: Receive assets

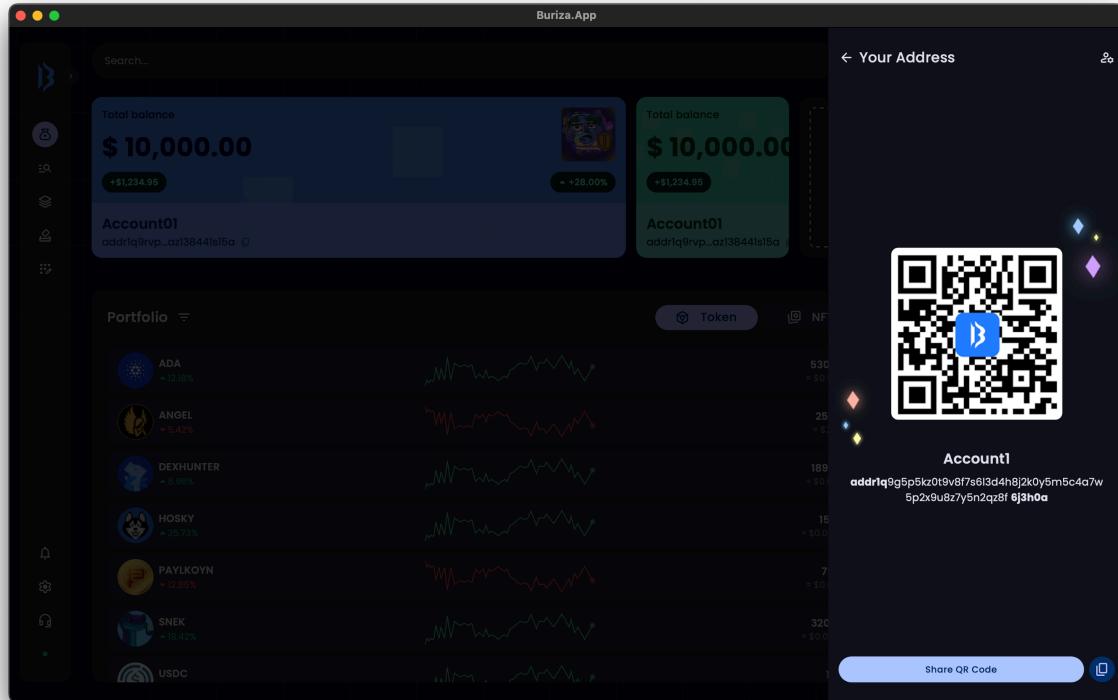


Figure 106 – Buriza UI - Desktop: Receive assets

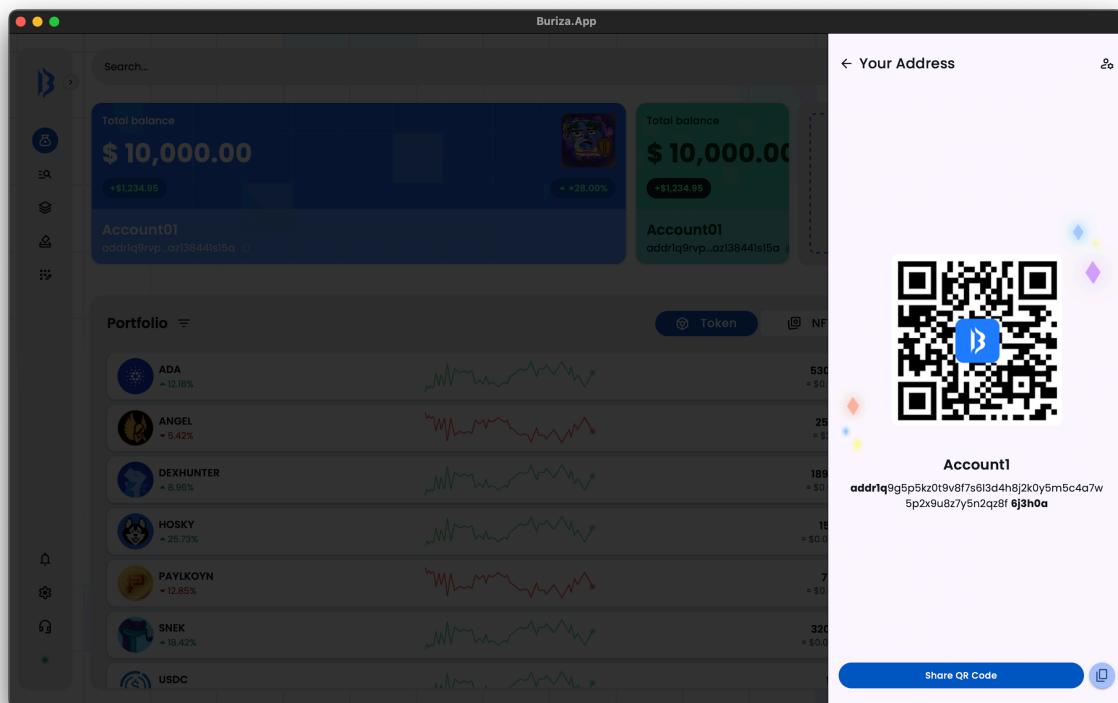


Figure 107 – Buriza UI - Desktop: Receive assets advanced mode

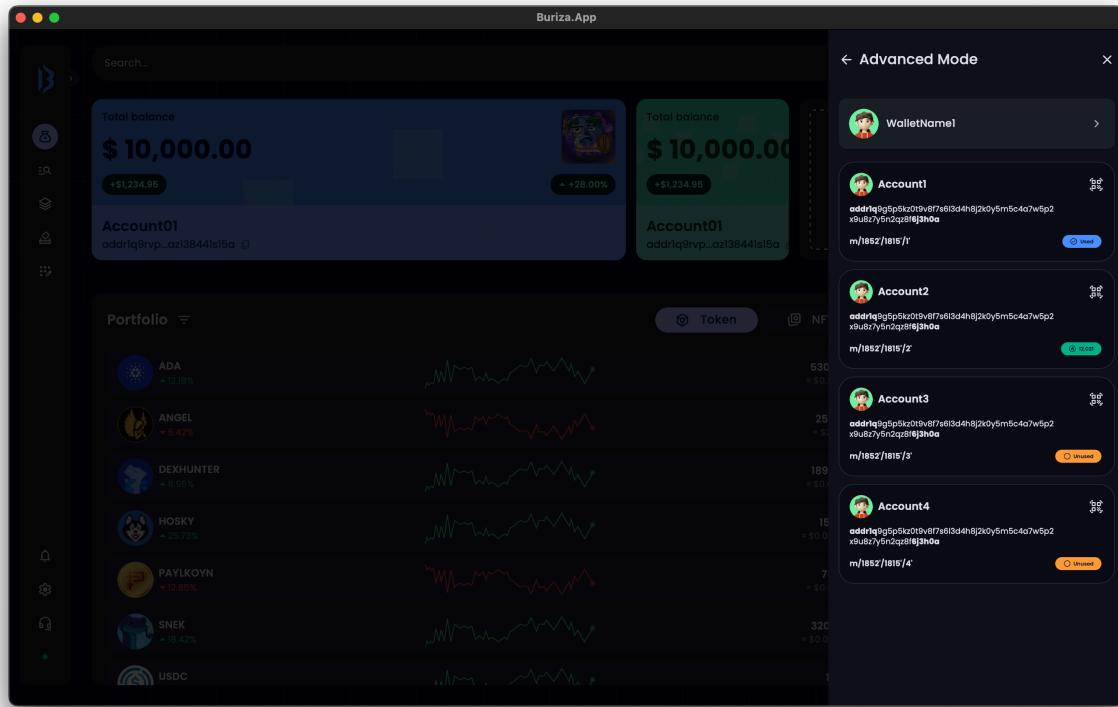
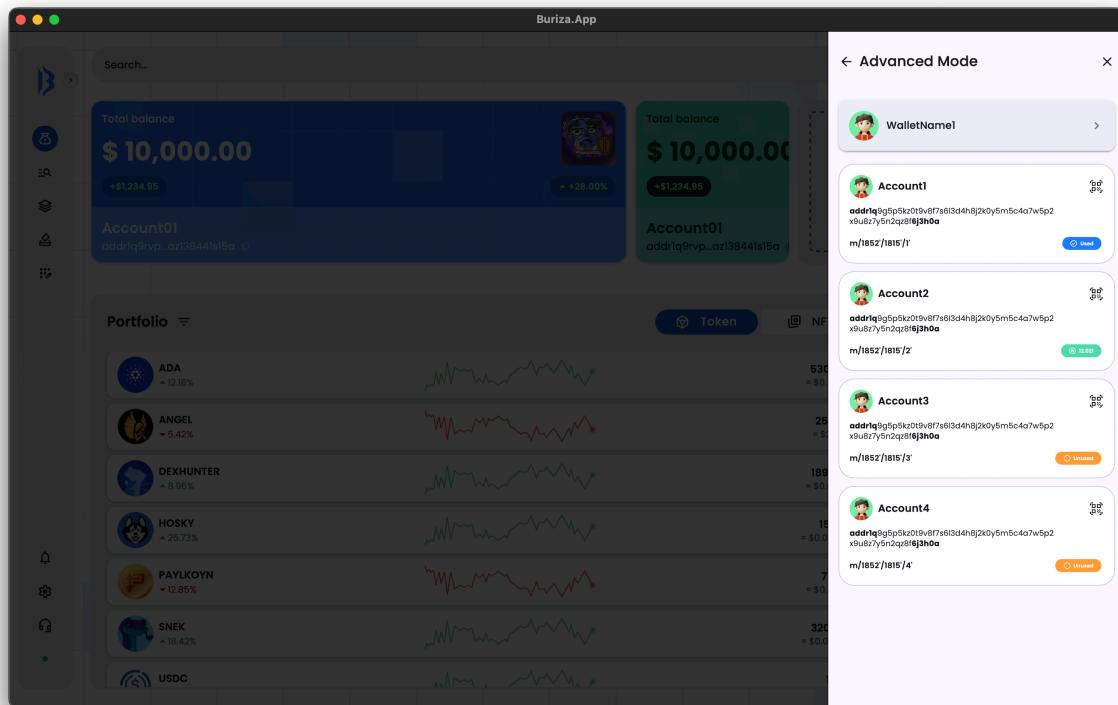


Figure 108 – Buriza UI - Desktop: Receive assets advanced mode



iv.3.3 View Balance and Assets

Figure 109 – Buriza UI - Desktop: View balance and assets

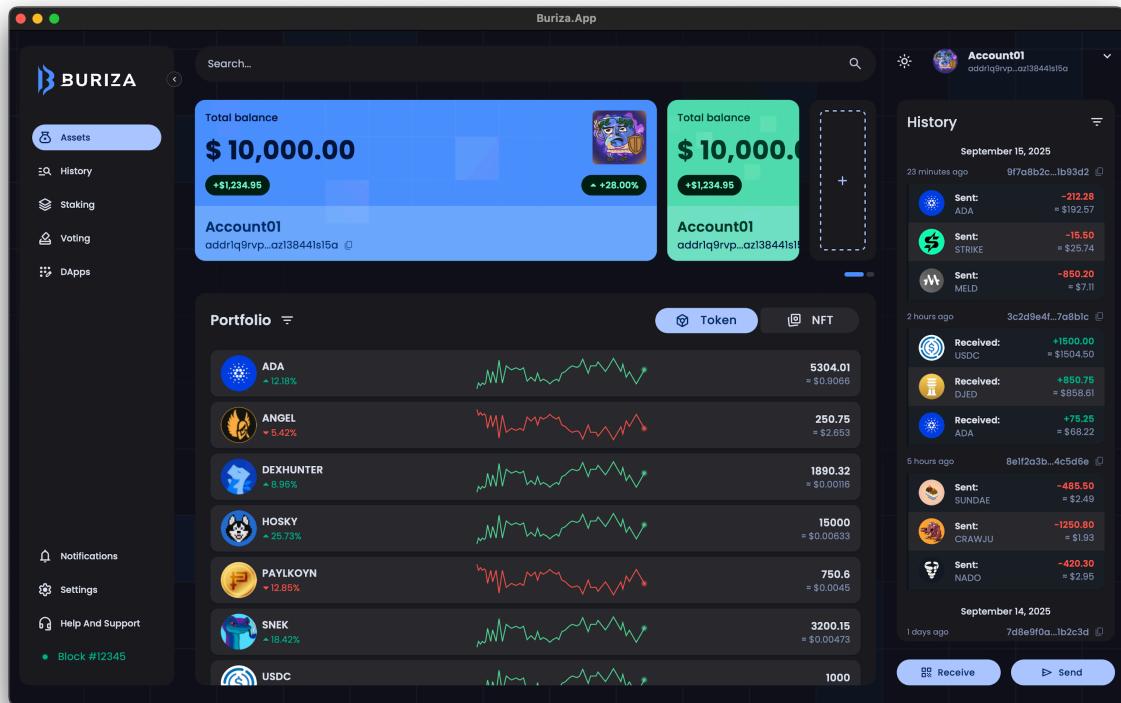
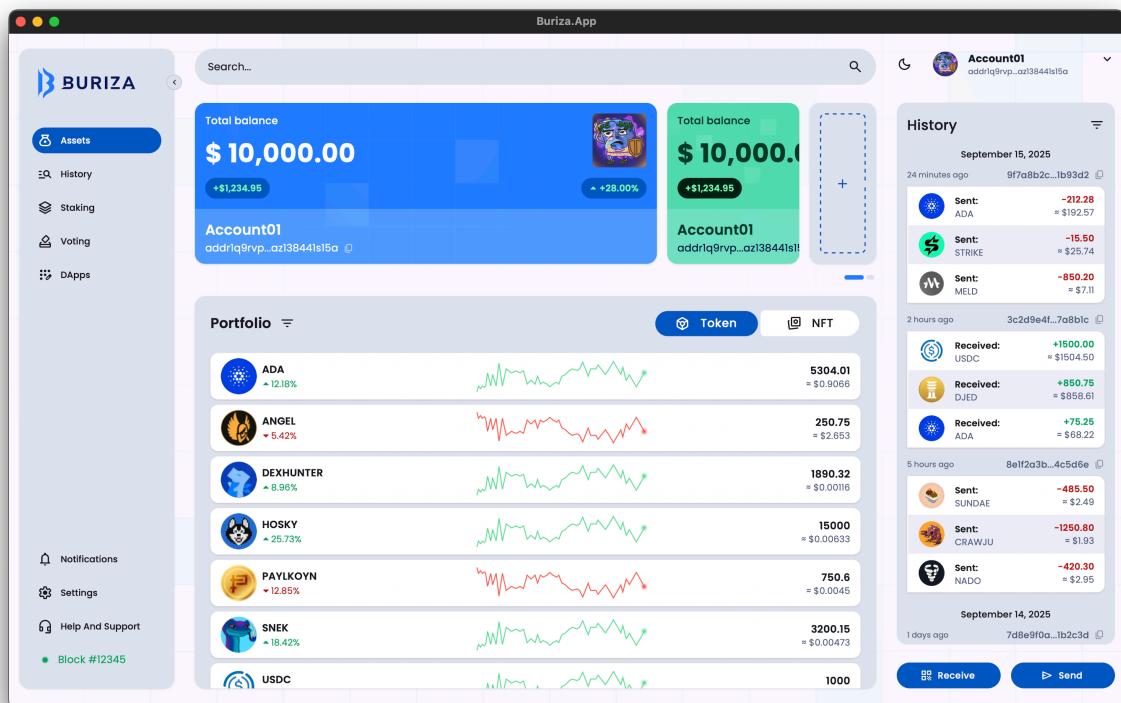


Figure 110 – Buriza UI - Desktop: View balance and assets



iv.3.4 Transaction History

Figure 111 – Buriza UI - Desktop: Transaction history with NFT

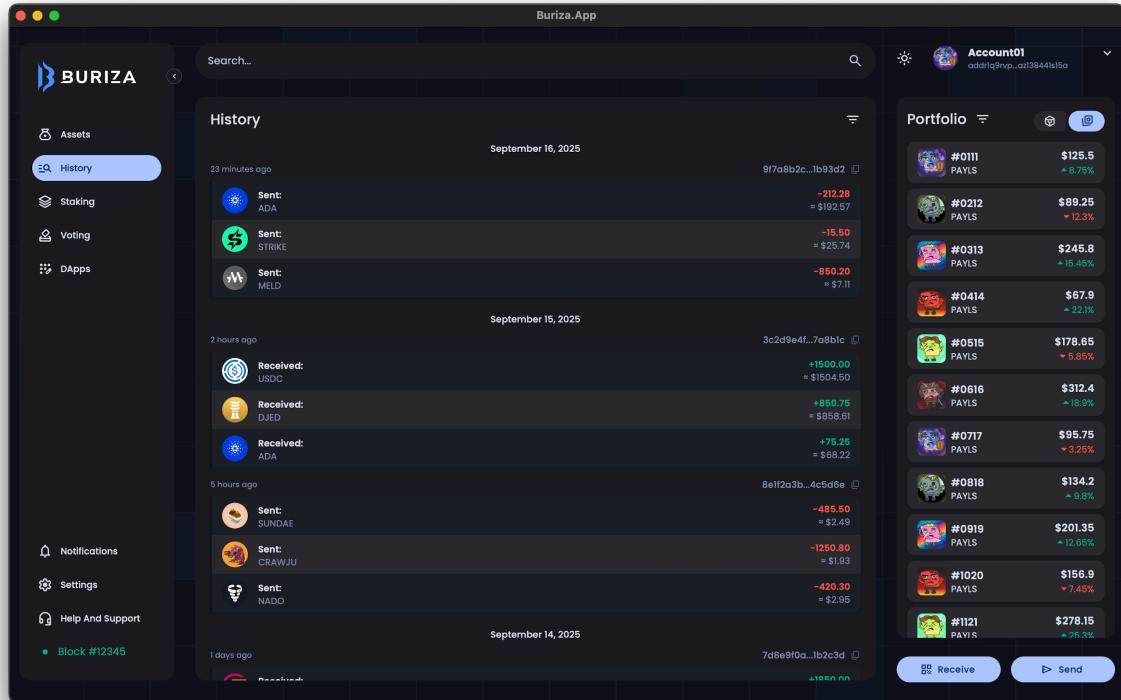


Figure 112 – Buriza UI - Desktop: Transaction history with NFT

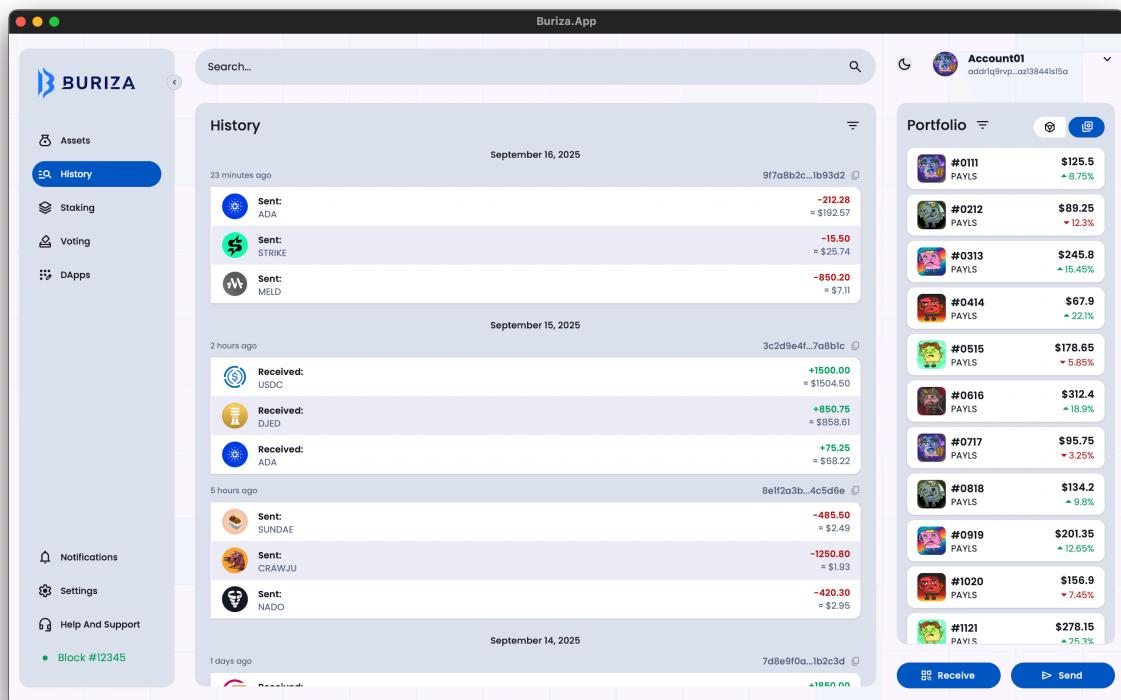


Figure 113 – Buriza UI - Desktop: Transaction history with tokens

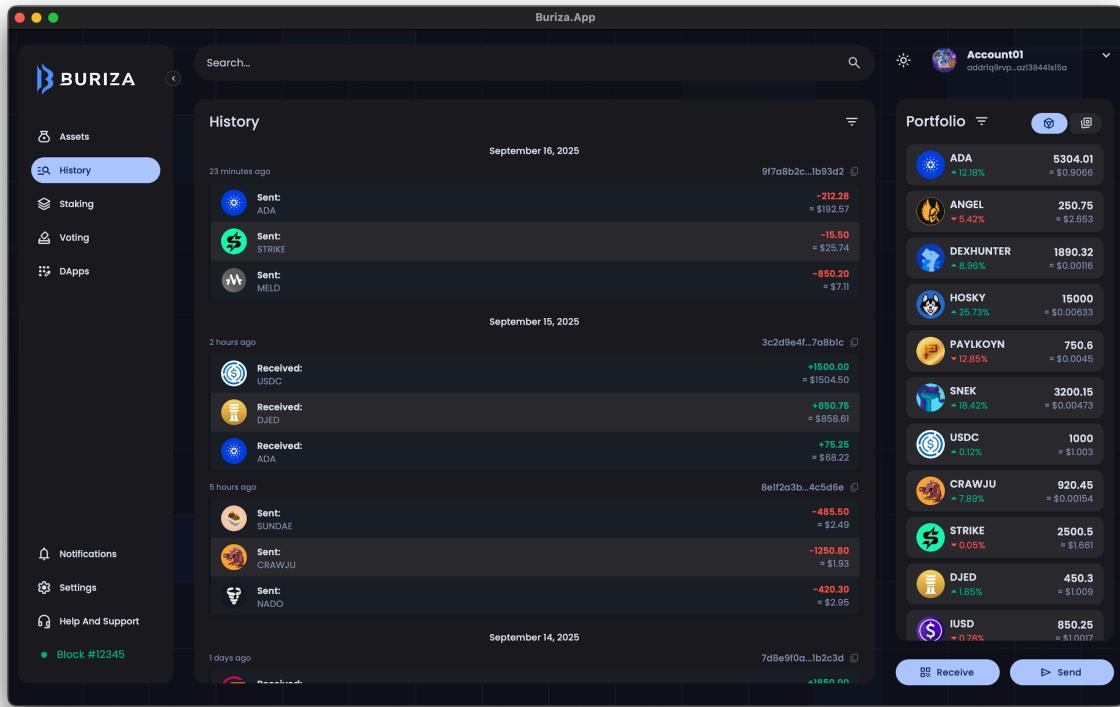


Figure 114 – Buriza UI - Desktop: Transaction history with tokens

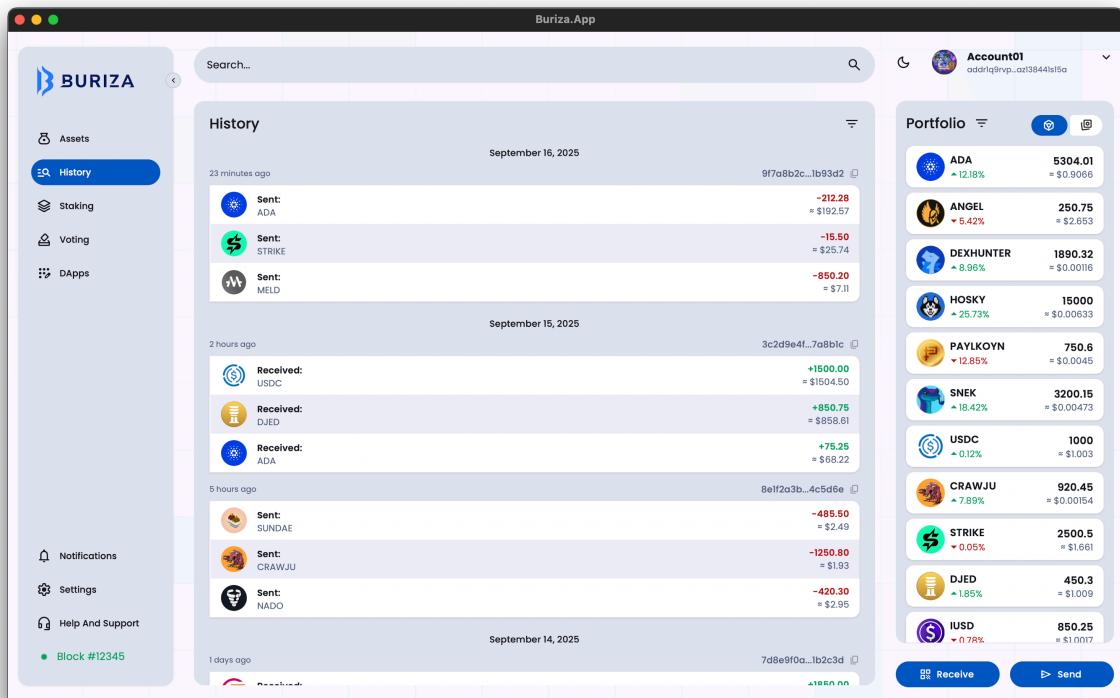


Figure 115 – Buriza UI - Desktop: Transaction summary

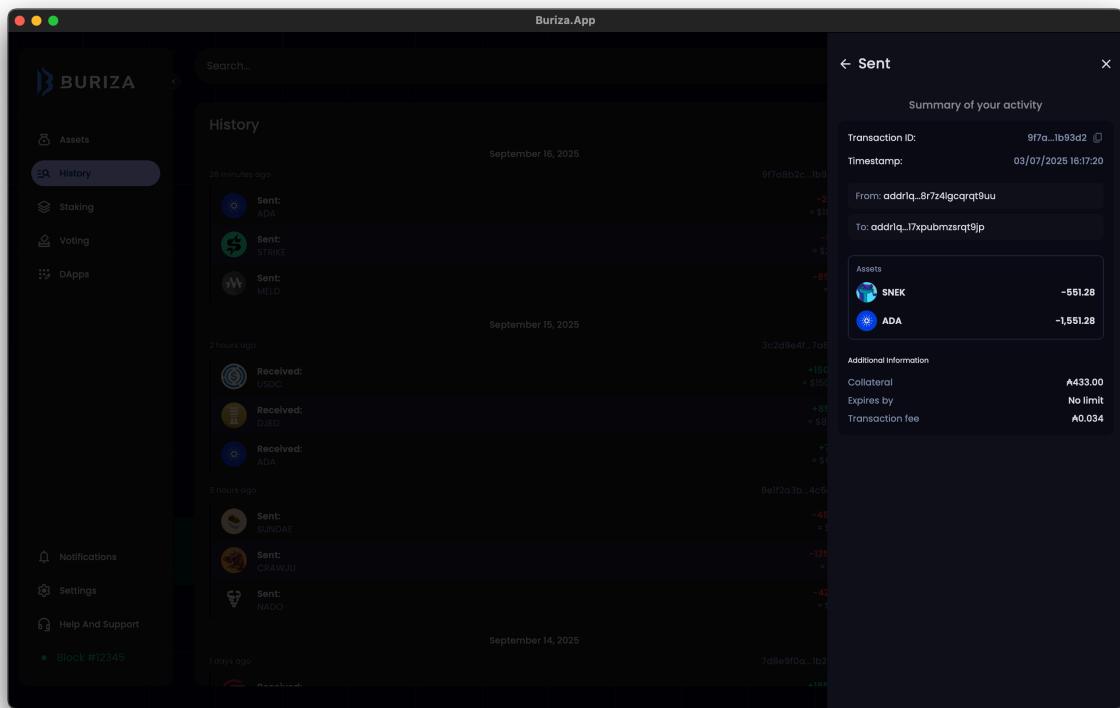
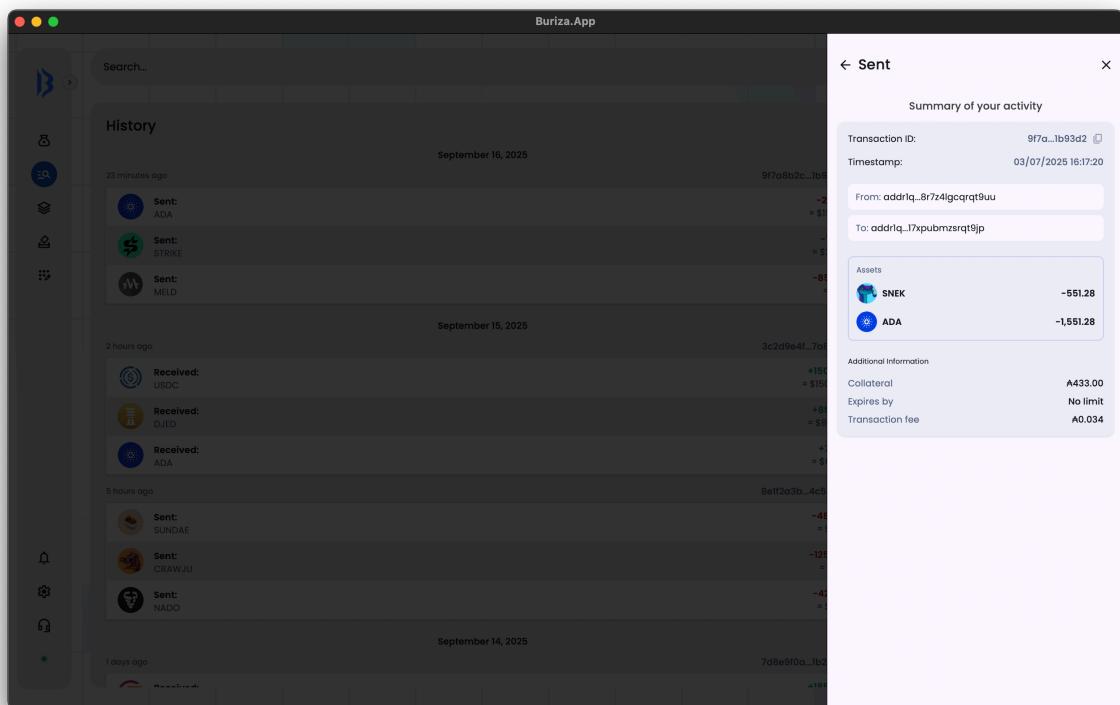


Figure 116 – Buriza UI - Desktop: Transaction summary



iv.3.5 Full Node Setup

Figure 117 – Buriza UI - Desktop: Settings

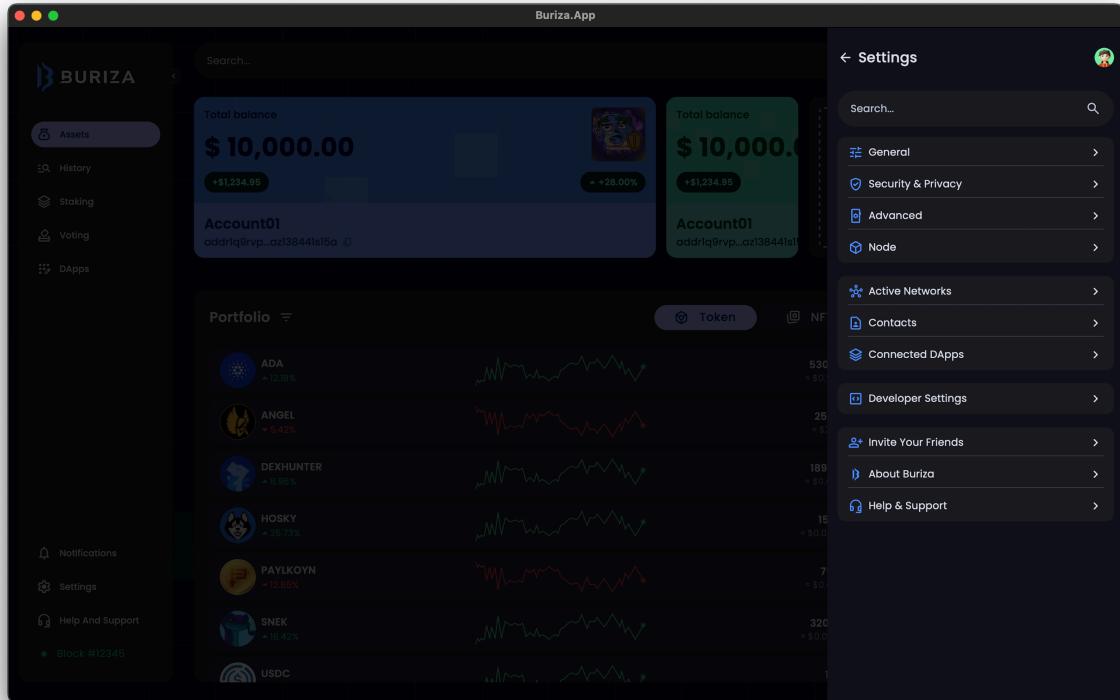


Figure 118 – Buriza UI - Desktop: Settings

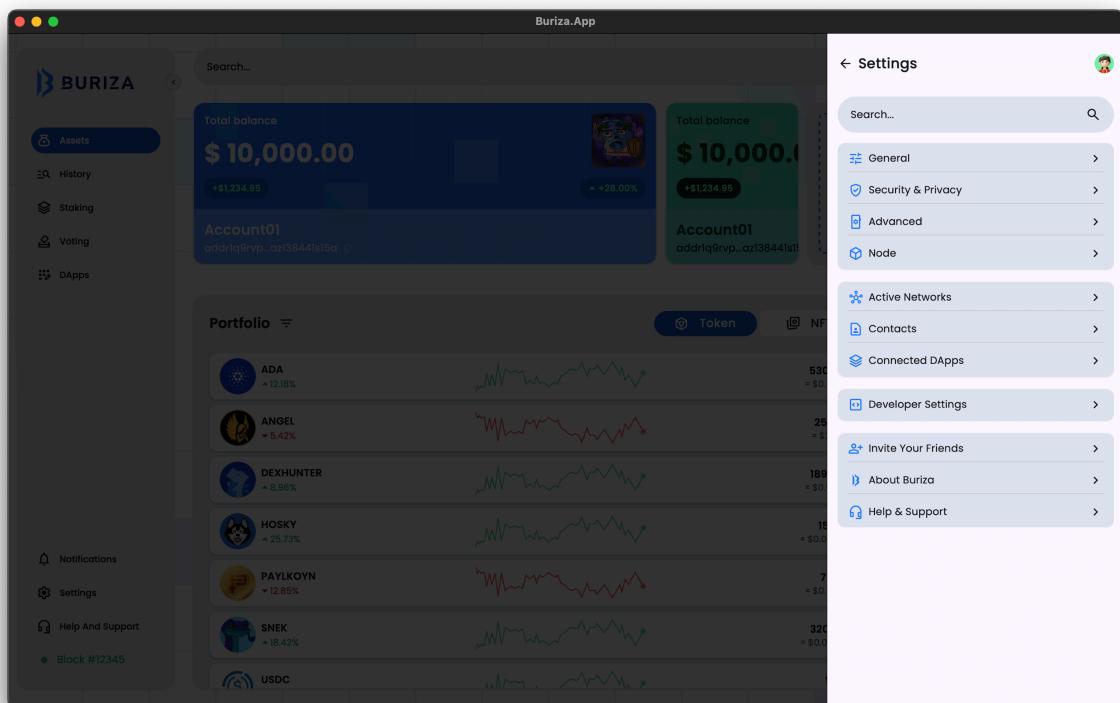


Figure 119 — Buriza UI - Desktop: Node setup

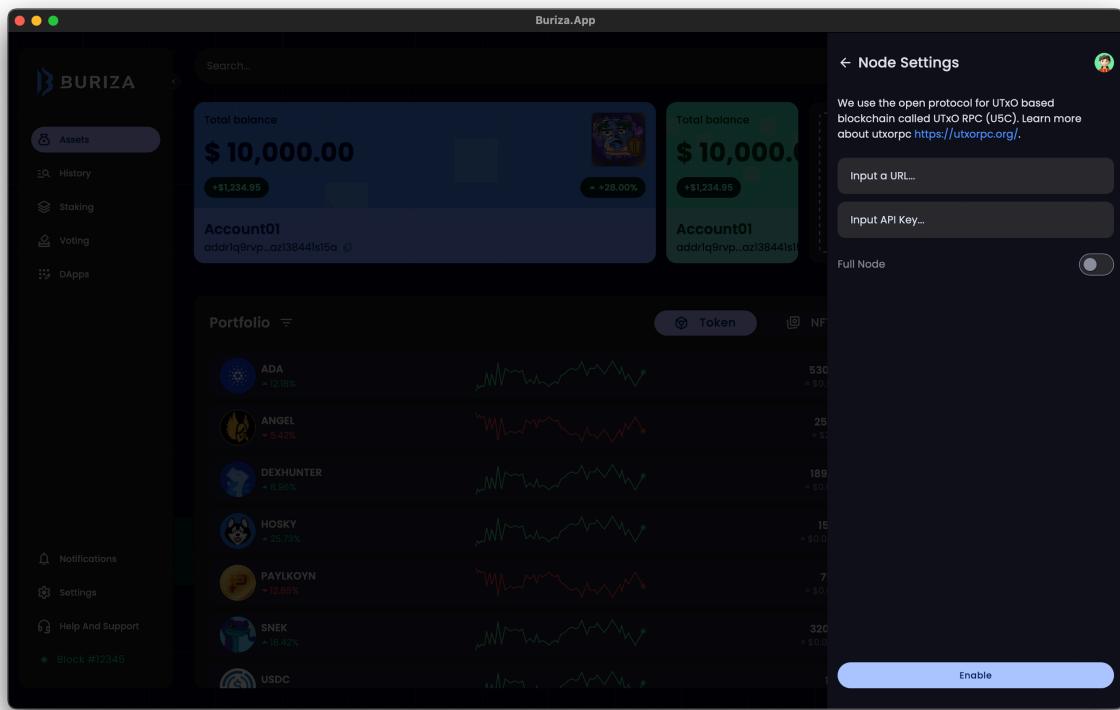
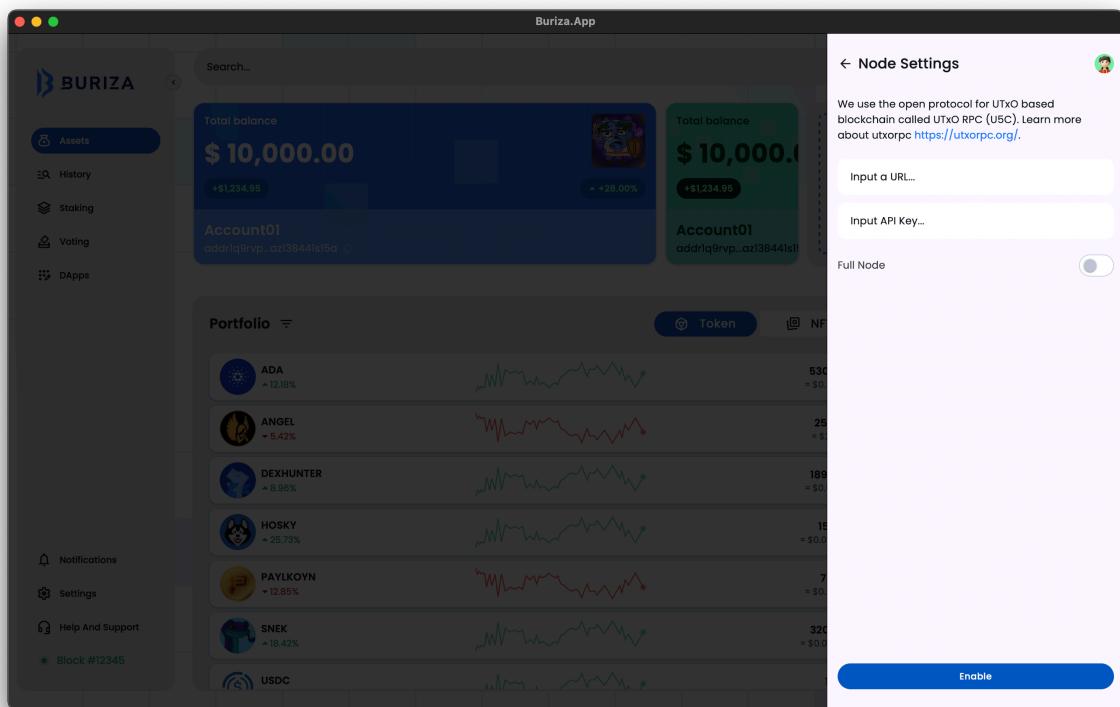


Figure 120 — Buriza UI - Desktop: Node setup



iv.3.6 dApp Access

Figure 121 – Buriza UI - Desktop: dApp explorer



Figure 122 – Buriza UI - Desktop: dApp explorer

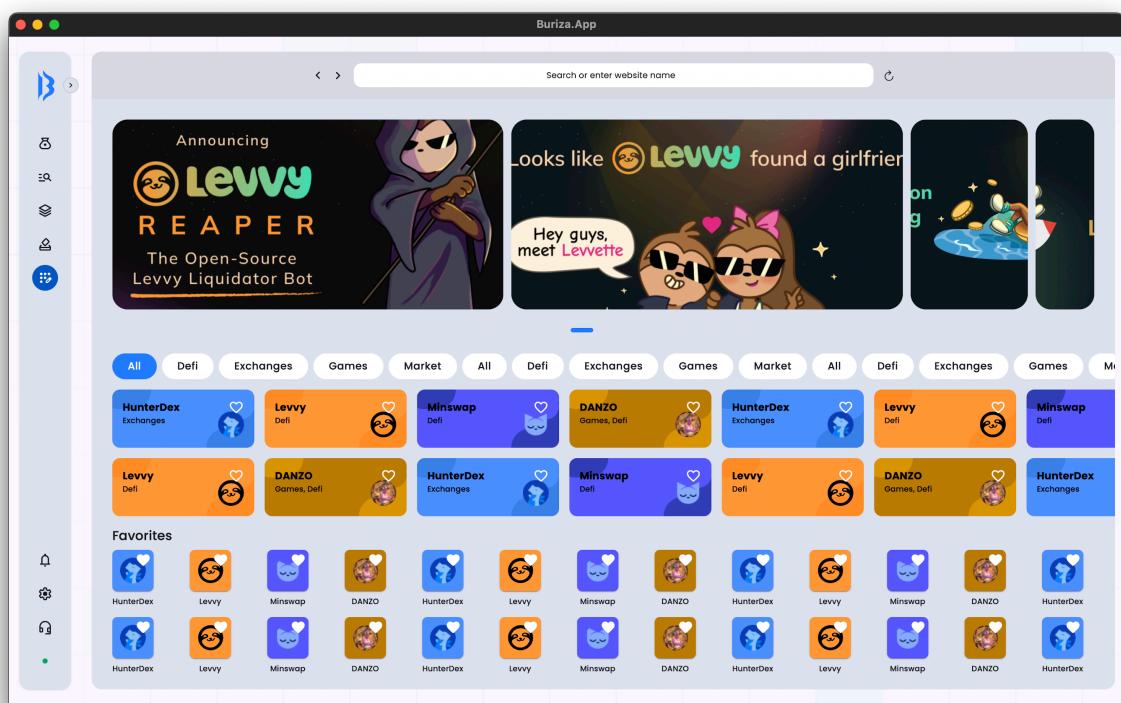


Figure 123 – Buriza UI - Desktop: dApp authorization request

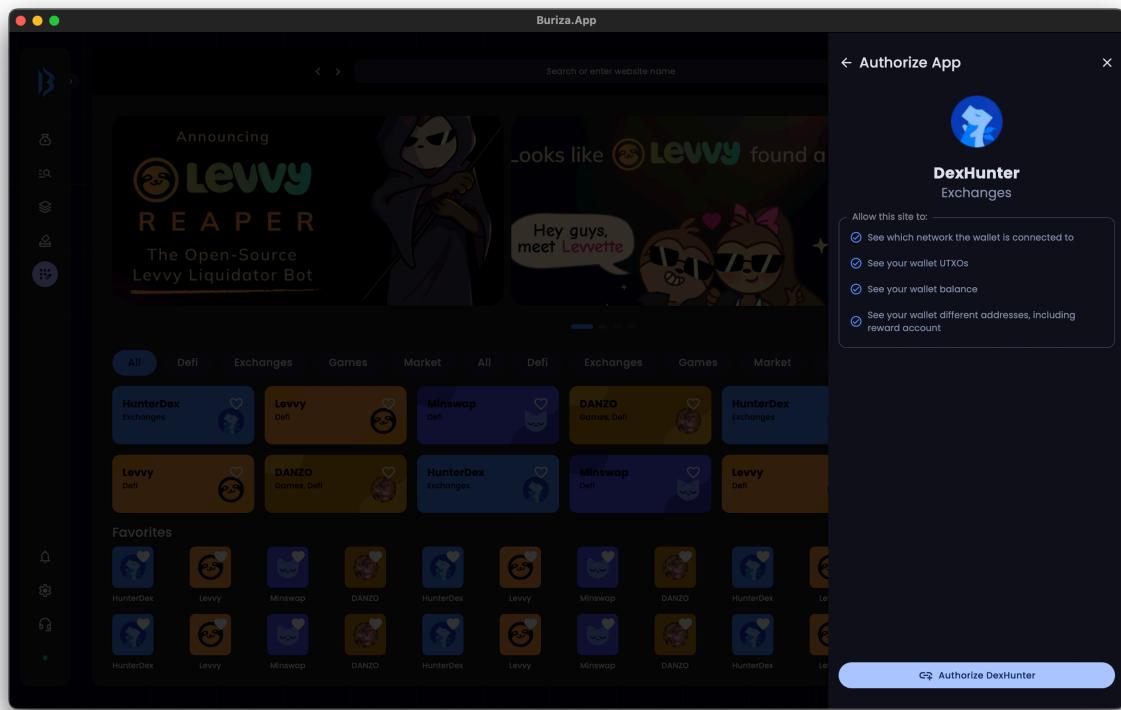
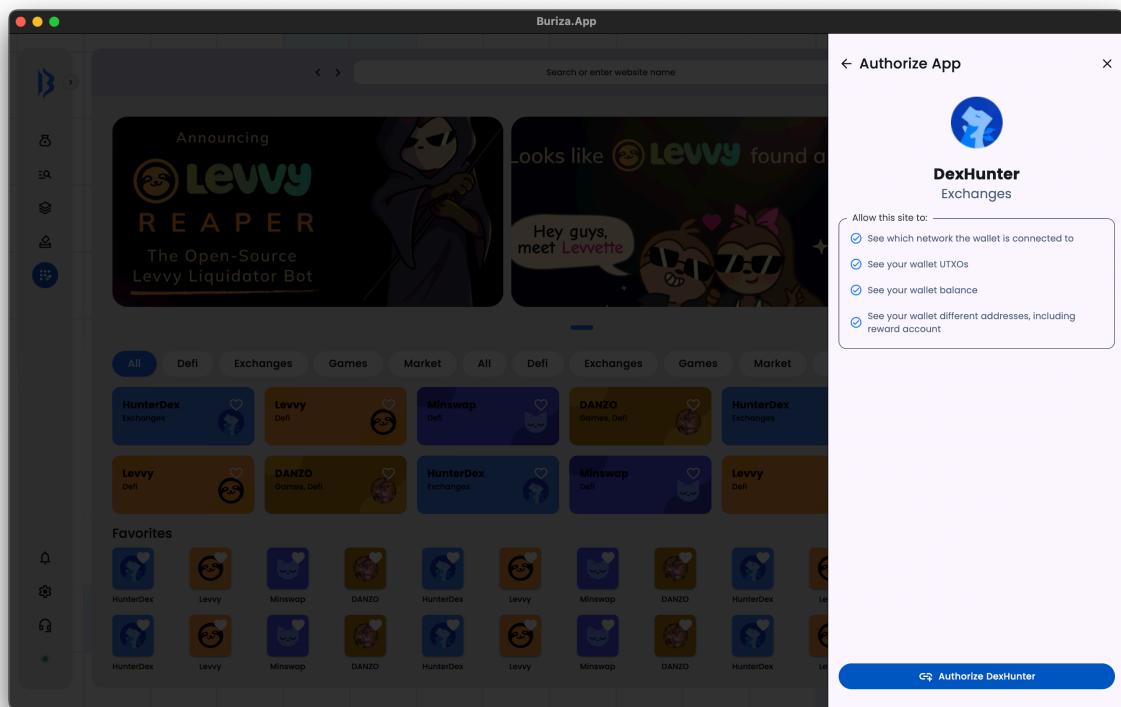


Figure 124 – Buriza UI - Desktop: dApp authorization request



iv.3.7 Wallet Creation and Account Management

Figure 125 – Buriza UI - Desktop: Manage wallet

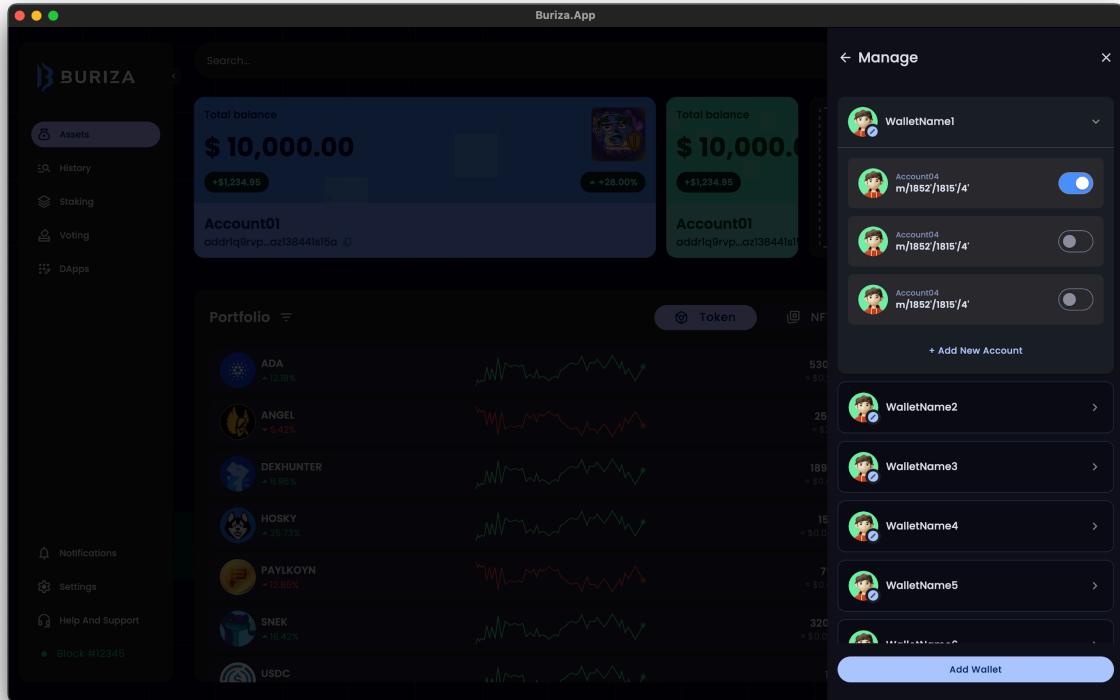


Figure 126 – Buriza UI - Desktop: Manage wallet

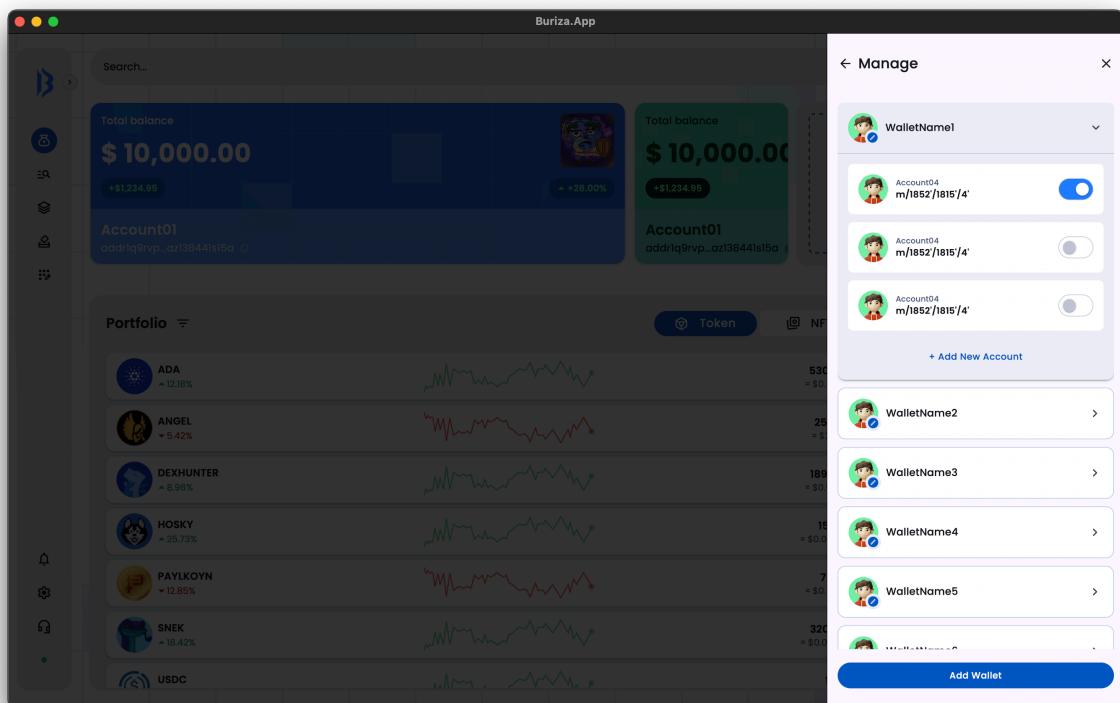


Figure 127 – Buriza UI - Desktop: Manage wallet - add new account

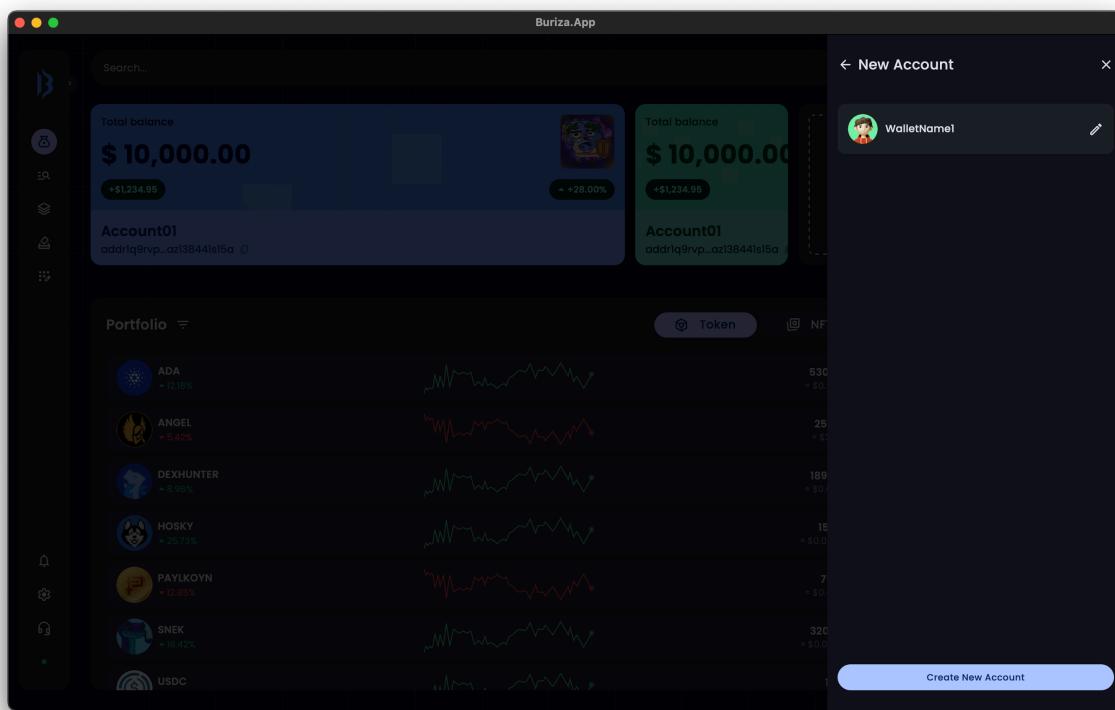


Figure 128 – Buriza UI - Desktop: Manage wallet - add new account

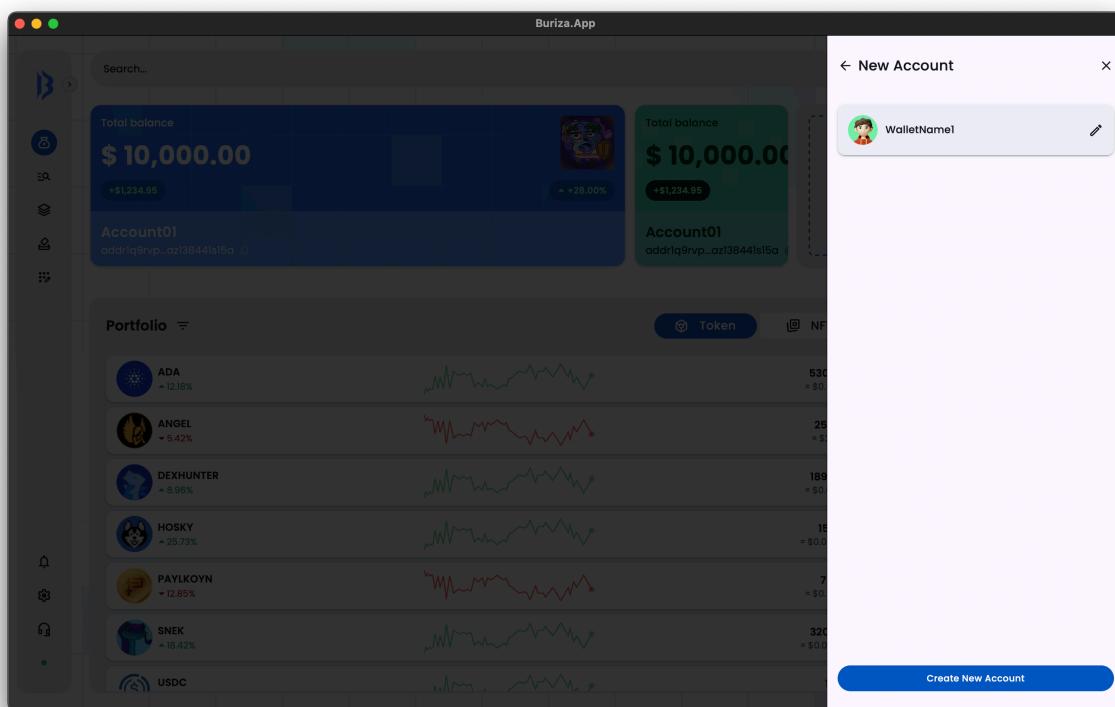


Figure 129 – Buriza UI - Desktop: Manage wallet - edit account

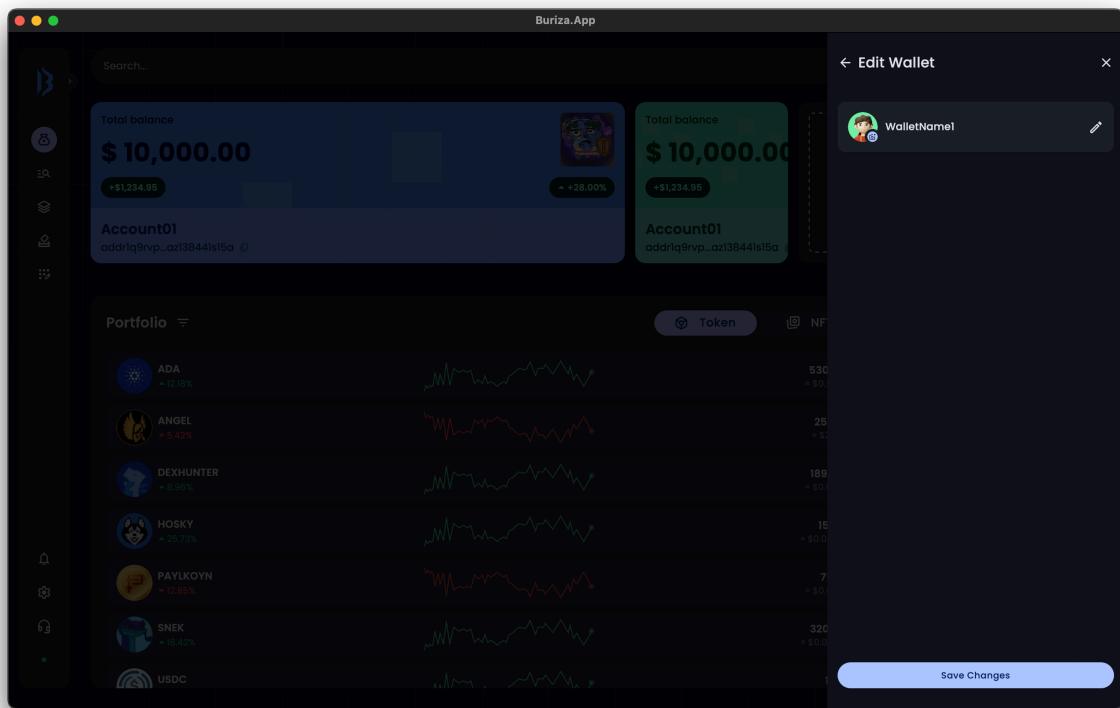


Figure 130 – Buriza UI - Desktop: Manage wallet - edit account

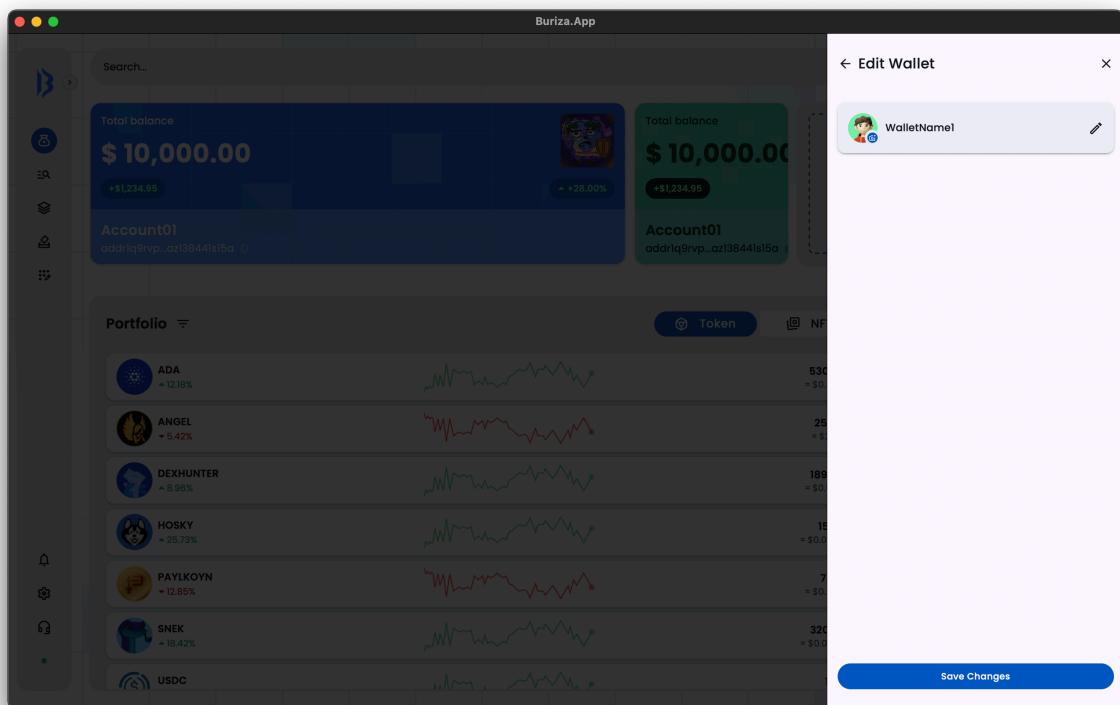


Figure 131 – Buriza UI - Desktop: Manage wallet - add new wallet

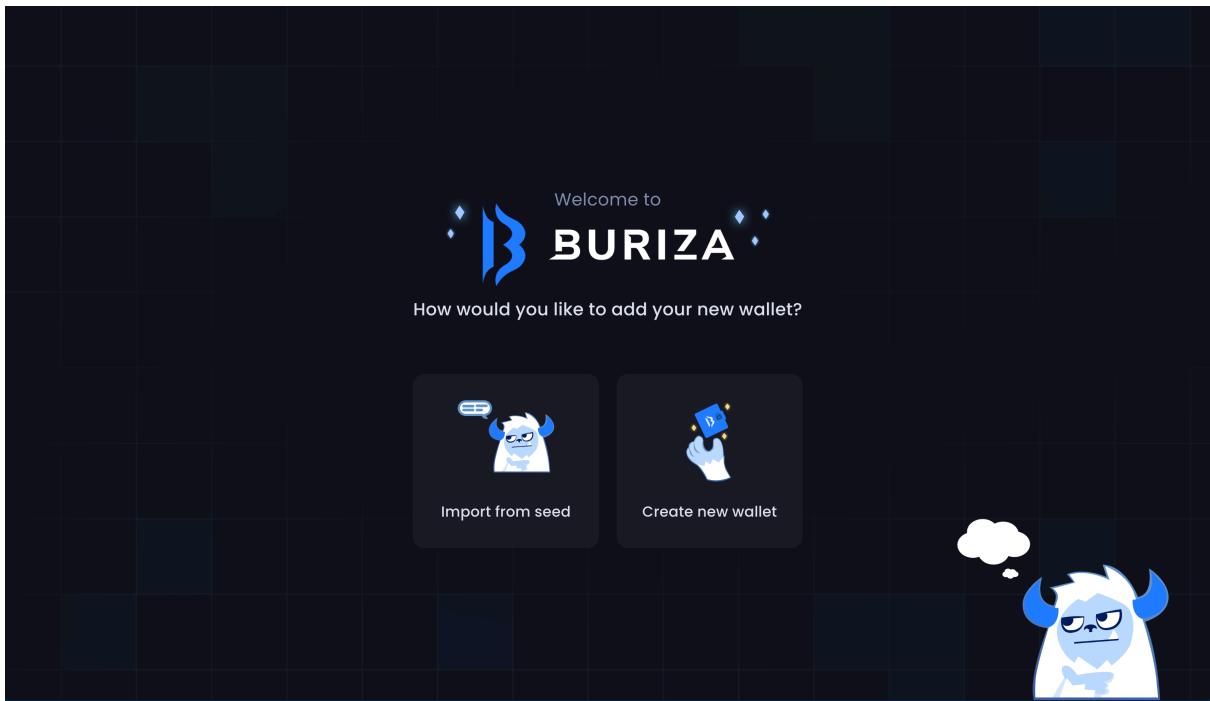


Figure 132 – Buriza UI - Desktop: Manage wallet - add new wallet

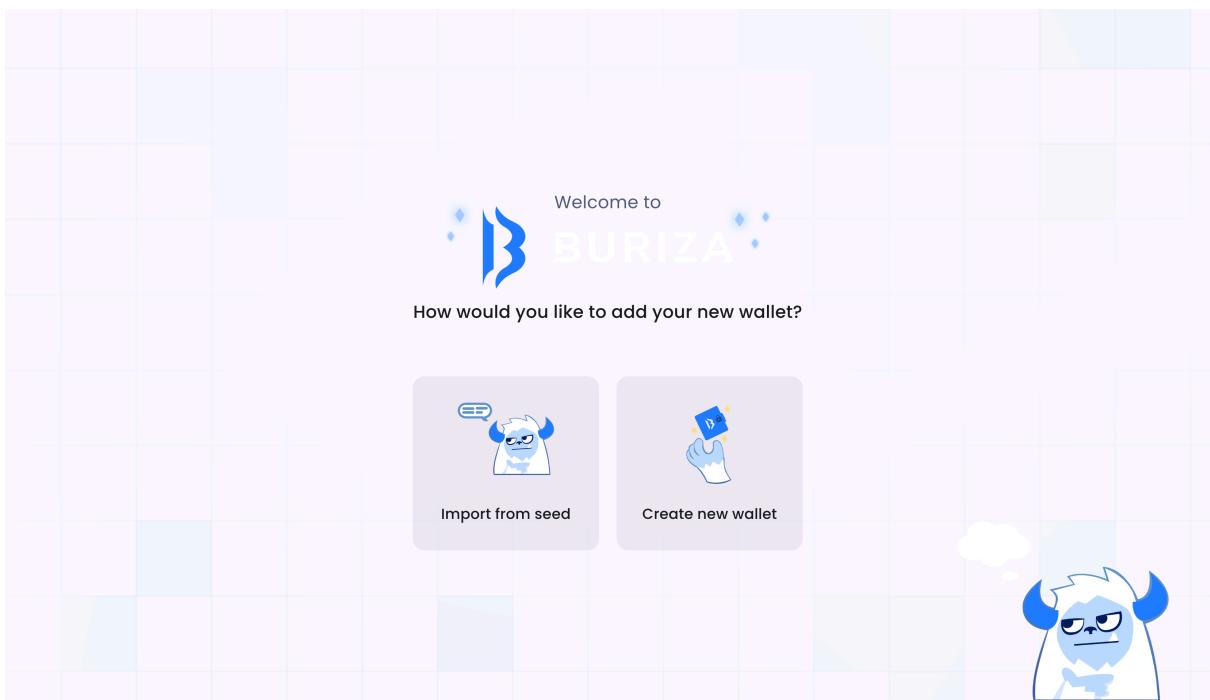


Figure 133 – Buriza UI - Desktop: Create new wallet - save recovery phrase

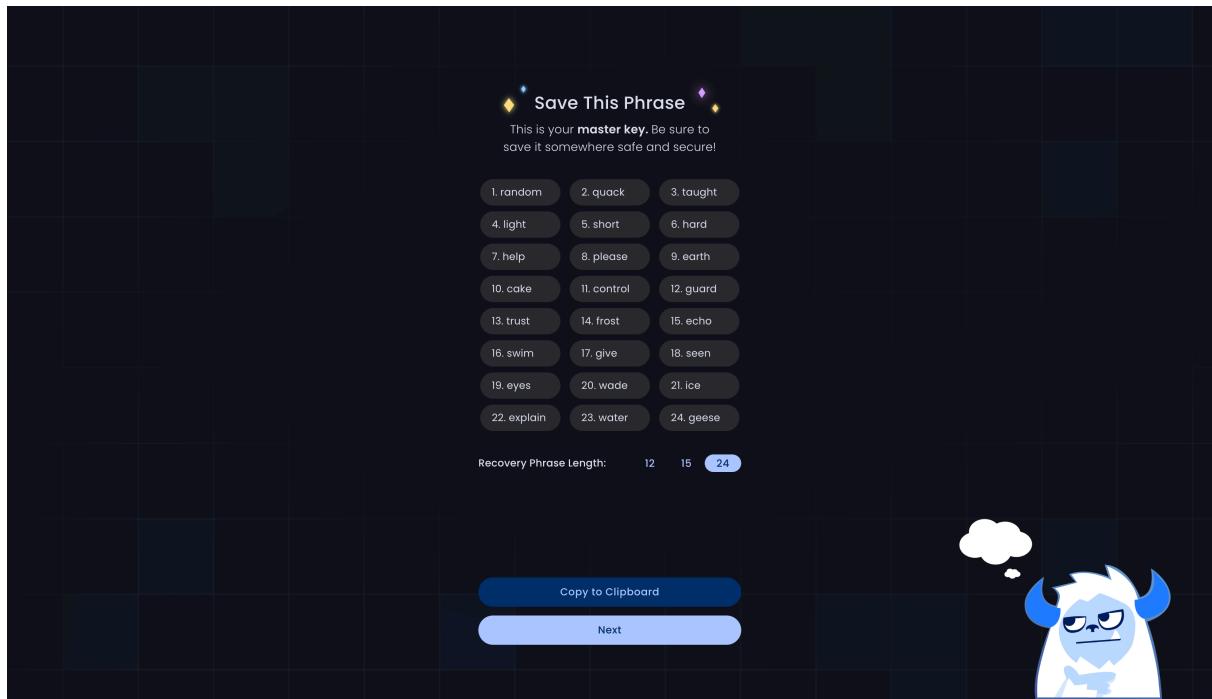


Figure 134 – Buriza UI - Desktop: Create new wallet - save recovery phrase

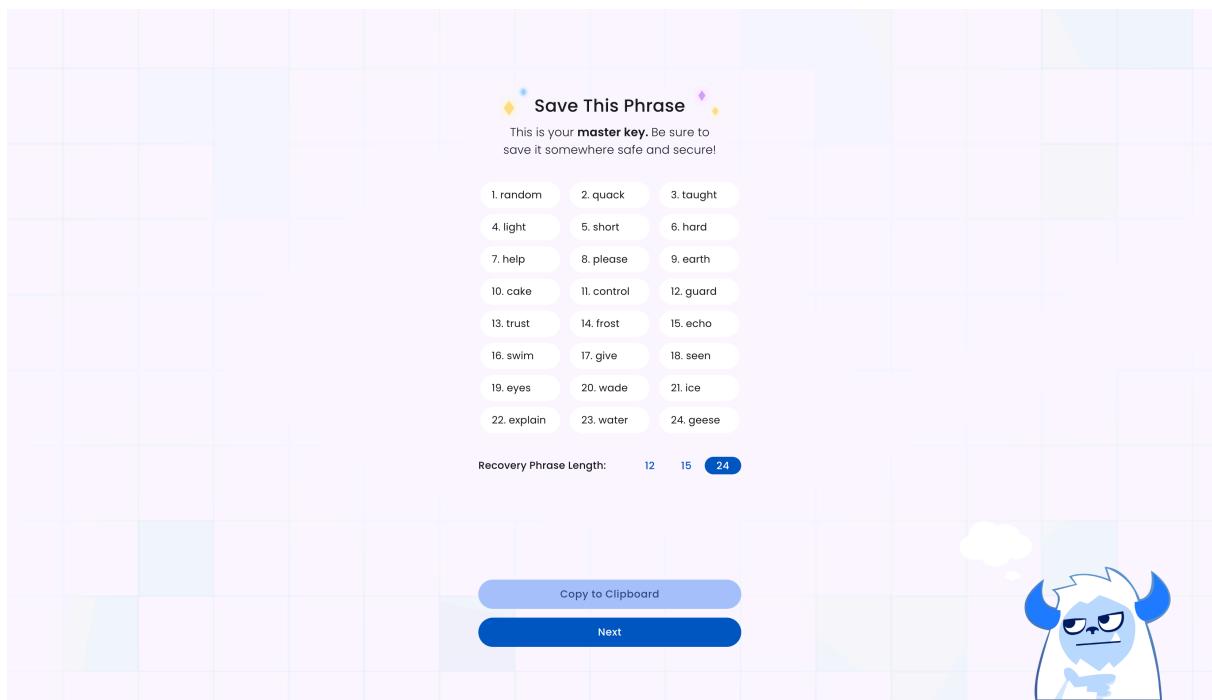


Figure 135 – Buriza UI - Desktop: Create new wallet - verify recovery phrase

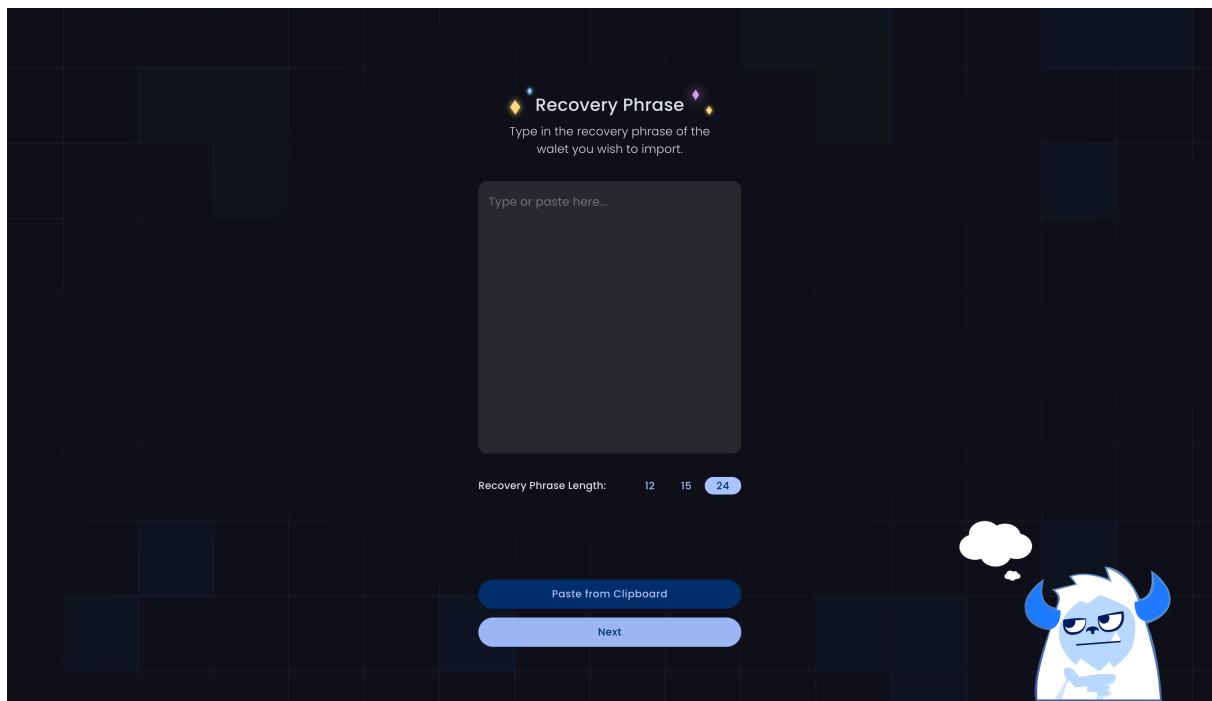


Figure 136 – Buriza UI - Desktop: Create new wallet - verify recovery phrase

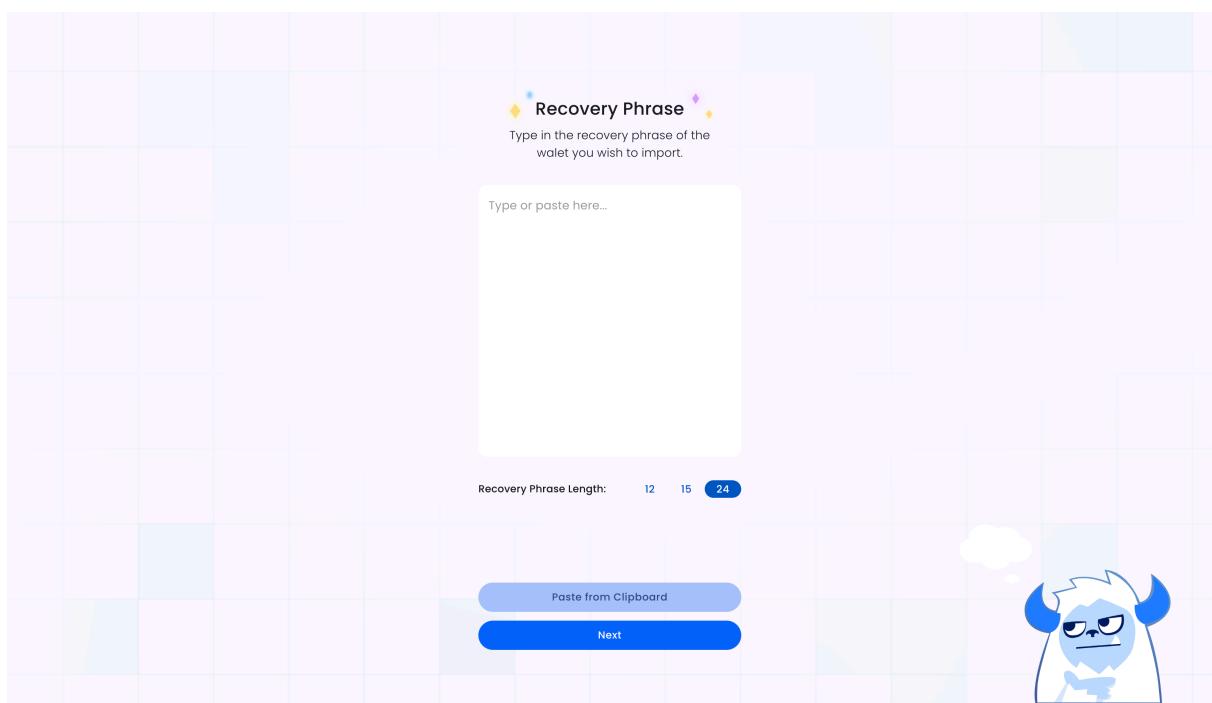


Figure 137 – Buriza UI - Desktop: Create new wallet - setup

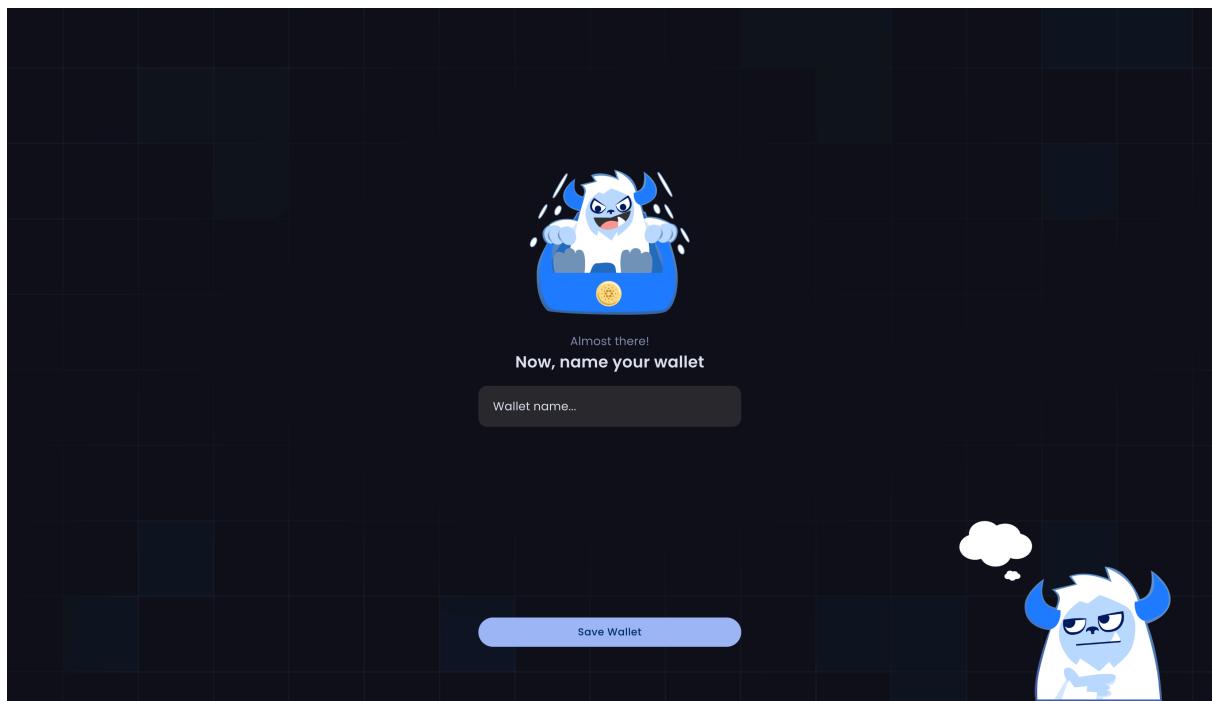


Figure 138 – Buriza UI - Desktop: Create new wallet - setup

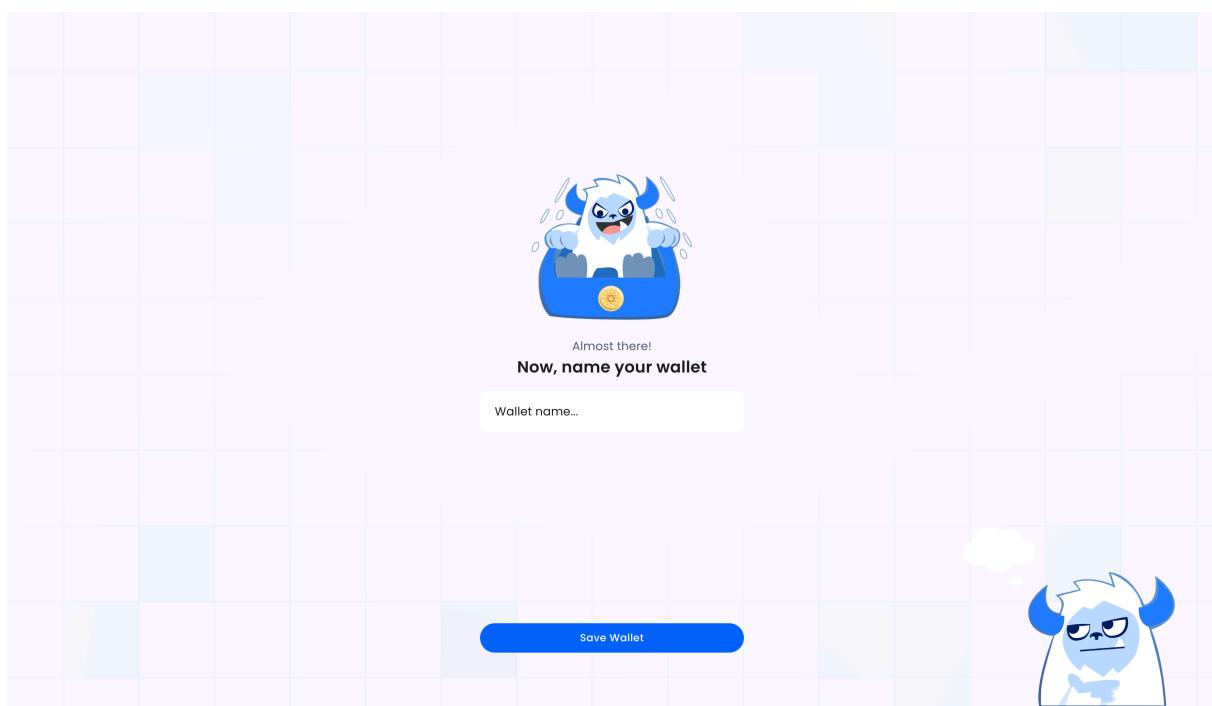


Figure 139 – Buriza UI - Desktop: Create new wallet - authentication

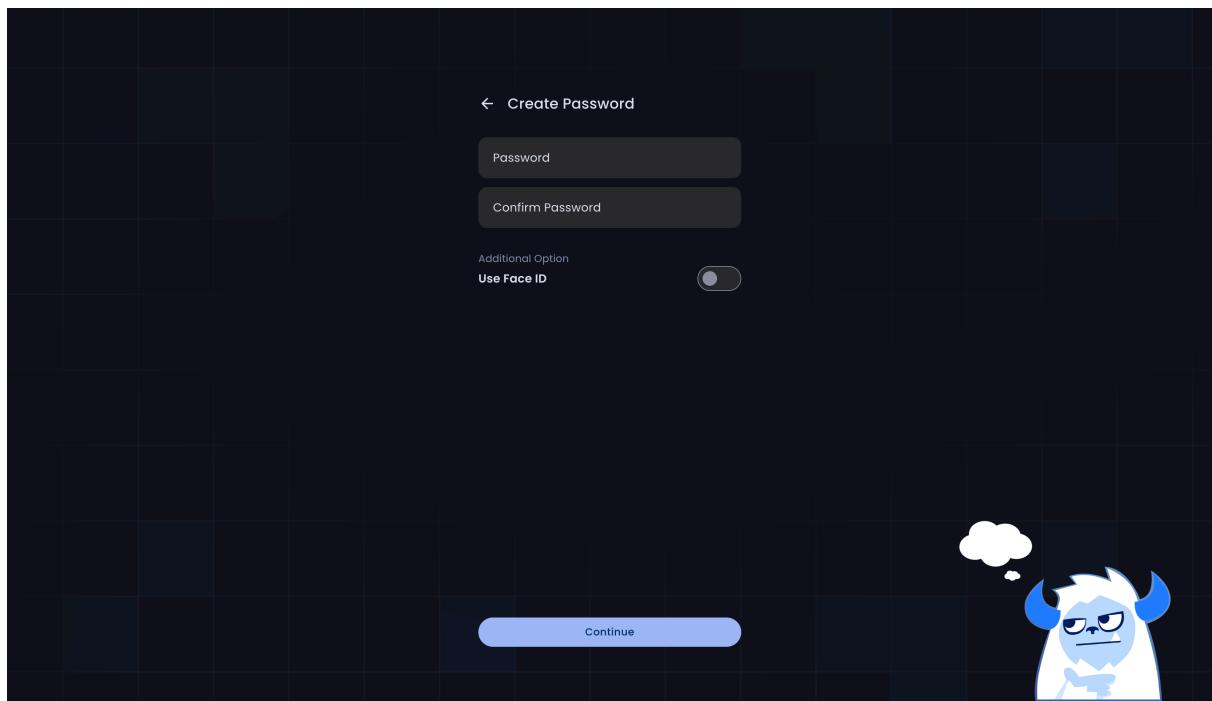


Figure 140 – Buriza UI - Desktop: Create new wallet - authentication

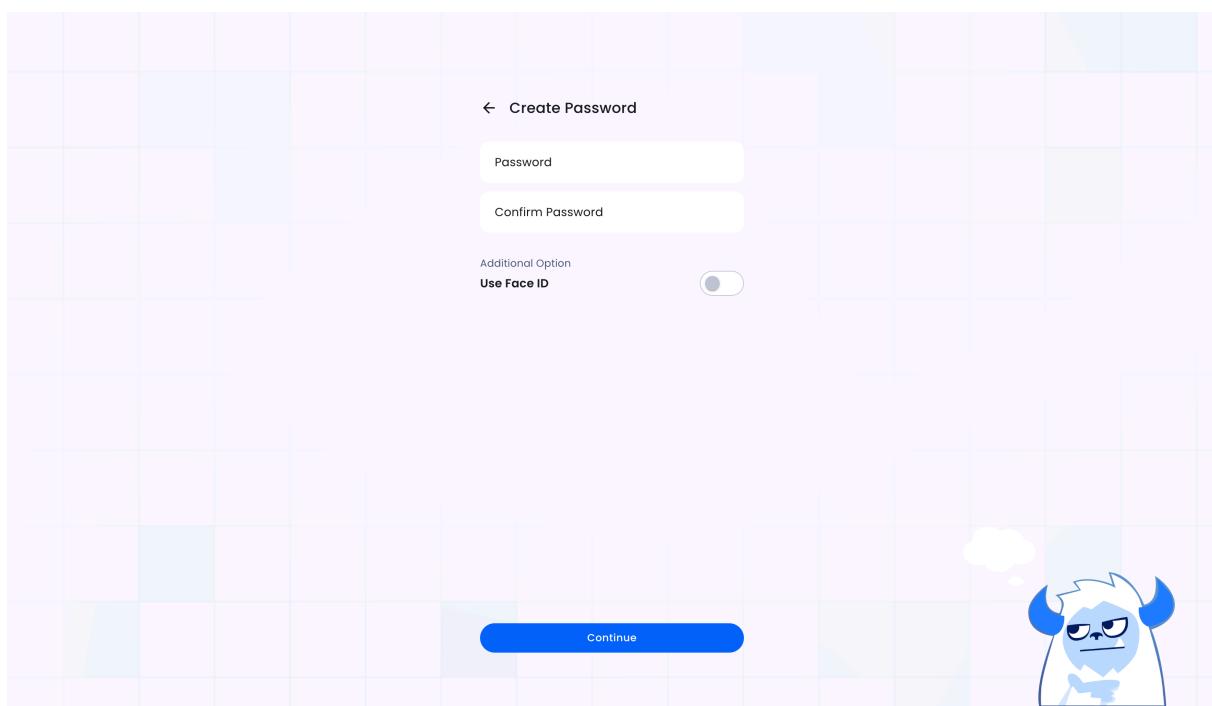


Figure 141 – Buriza UI - Desktop: Create new wallet - success screen

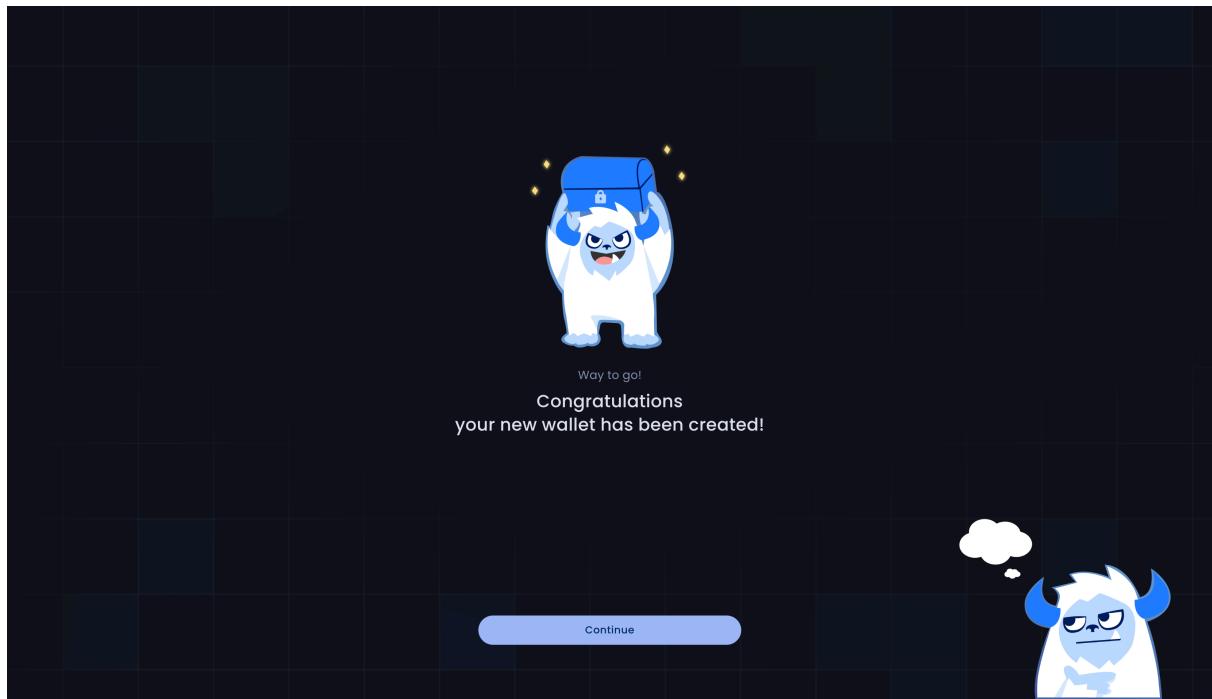
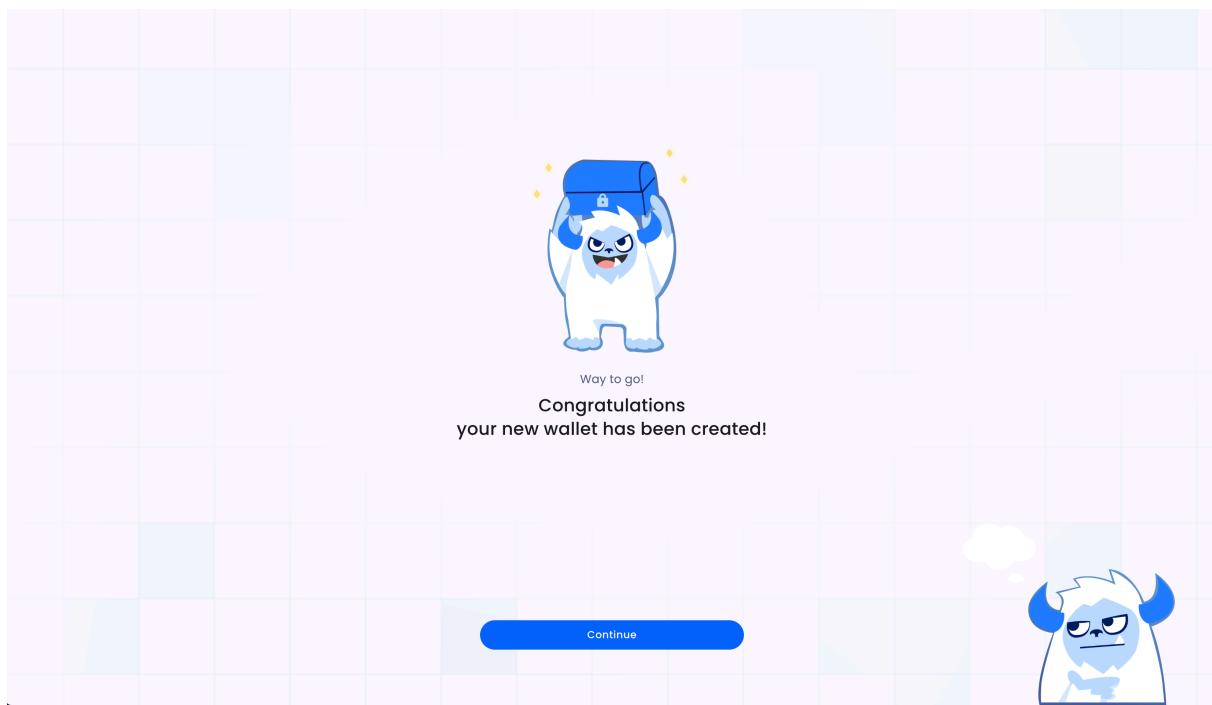


Figure 142 – Buriza UI - Desktop: Create new wallet - success screen



V. Platform Development Setup

This section provides step-by-step instructions for building and running Buriza from source. As an open-source Cardano wallet suite, Buriza supports multiple deployment targets including desktop applications, mobile platforms, browser extensions, and progressive web apps.

v.1 Prerequisites

System Requirements:

- [.NET 9 SDK](#) (v9.0.0+)
- [Bun](#) (v1.2.0+)
- [Git](#) (v2.39.0+)

Platform-Specific Requirements:

- **iOS Development:** macOS, Xcode, Apple Developer account
- **Android Development:** Android SDK, Android Studio
- **Browser Extension:** Chrome/Chromium or Firefox developer mode

v.2 Installation

Clone the repository and install dependencies:

```
git clone git@github.com:SAIB-Inc/Buriza.git
cd Buriza

# Restore workloads for MAUI development
dotnet workload restore

# Install CSS dependencies
cd src/Buriza.UI/wwwroot && bun install

# Build the entire solution
dotnet build
```

v.3 Platform Development Instructions

v.3.1 Desktop Application (macOS)

```
# Navigate to app directory
cd src/Buriza.App

# Build and run macOS application
dotnet build -f net9.0-maccatalyst && dotnet run -f net9.0-maccatalyst
```

v.3.2 Desktop Application (Windows)

```
# Build Windows application
cd src/Buriza.App && dotnet dotnet build -f net9.0-windows10.0.19041.0 &&
dotnet build -f net9.0-windows10.0.19041.0
```

v.3.3 iOS Simulator

```
# Install required workloads
dotnet workload restore

# Build for iOS simulator
cd src/Buriza.App && dotnet build . -f net9.0-ios

# Install and launch on simulator
xcrun simctl boot "iPhone 16 Pro" # or any available device
xcrun simctl install booted bin/Debug/net9.0-ios/iossimulator-arm64/
Buriza.App.app
```

```
xcrun simctl launch booted com.saibinc.buriza  
# If the simulator does not appear  
open -a Simulator
```

v.3.4 Physical iPhone

Prerequisites:

- Connect your iPhone via USB
- Click on “Trust This Computer” on your iPhone when prompted
- Change bundle ID to com.yourname.buriza in Buriza.App.csproj
- Create an Xcode project with same bundle ID to generate provisioning profile

```
# Build for physical device
cd src/Buriza.App && dotnet build . -f net9.0-ios -p:RuntimeIdentifier=ios-arm64

# Get device ID
xcrun devicectl list devices

# Install to connected device
xcrun devicectl device install app --device [device-id] bin/Debug/net9.0-ios/ios-arm64/Buriza.App.app
```

Note:

- You may need to go to Settings > Privacy & Security > Developer Mode on your iPhone
- You may need to go to Settings > General > VPN & Device Management on your iPhone to trust the developer certificate

v.3.5 Android Development

Prerequisites:

1. Install Android Studio or Android SDK
2. Install OpenJDK 11+ and add to system PATH ([Download](#))

Building and Running:

```
# Start Android emulator
emulator -avd <emulator-name> &

# Wait for device to boot
adb wait-for-device

# Build and deploy to emulator/device
cd src/Buriza.App && dotnet build -t:Run -f net9.0-android

# Build for specific device architecture
dotnet build -t:Run -f net9.0-android -p:RuntimeIdentifier=android-arm64
```

Useful Commands:

```
# List available emulators
emulator -list-avds

# Check connected devices
adb devices

# Uninstall app
adb uninstall com.saibinc.buriza
```

```
# View app logs  
adb logcat | grep -i buriza
```

Note: For physical Android devices, you need to enable Developer Options and USB Debugging in Settings > Developer Options

System Requirements:

- Android 6.0+ / API 23+
- Minimum 2GB free disk space for emulator

v.3.6 Browser Extension

```
# Build browser extension  
cd src/Buriza.Extension  
dotnet build -c Release
```

```
# Load unpacked extension in browser:
```

1. Navigate to browser's extension management page
2. Enable "Developer mode"
3. Click "Load unpacked" and select src/Buriza.Extension/bin/Release/net9.0/browserextension/

v.3.7 Progressive Web App

```
# Run web application  
cd src/Buriza.Web  
dotnet run
```

```
# Run CSS watch (in separate terminal)  
cd src/Buriza.UI/wwwroot && bun watch
```

VI. Conclusion

This document demonstrates the successful completion of Milestone 2 for the Buriza wallet suite, showcasing significant progress in UI/UX design and front-end development. Through the adoption of Material Design 3 principles and the introduction of Buriza Frostling as the new mascot, the project has evolved into a more refined, accessible, and visually cohesive application that embodies the core values of security, trust, and power.

The technical implementation leveraging .NET 9, Blazor, and a shared component architecture has proven highly effective, enabling a single codebase to serve multiple deployment targets across mobile, desktop, and browser platforms. This approach has dramatically reduced development complexity while ensuring consistent user experiences across all platforms.

However, achieving Milestone 2 represents just the beginning of Buriza's journey. The development team remains committed to continuous improvement and feature expansion. Future developments will focus on enhancing wallet functionality, improving performance, and incorporating community feedback to create an even more robust and user-centered Cardano wallet experience.

SAIB Inc. is dedicated to maintaining and evolving Buriza as a leading open-source, decentralized Cardano wallet solution. The project will continue to uphold its principles of complete decentralization, self-verifiability, and user sovereignty while adapting to the evolving needs of the Cardano ecosystem and its community.

VII. Links

This section contains links relevant to the Buriza project:

- [Buriza GitHub repository](#)
- [Screenshots and Videos](#)
- [Buriza Design Report](#)
- [Buriza Material Design 3](#)

Project Catalyst:

- [Project Catalyst: Buriza.Browser](#)
- [Project Catalyst: Buriza/Desktop](#)
- [Project Catalyst: Buriza.Mobile](#)

VIII. References

- Google. (2025). *Material Design 3*. Material Design. <https://m3.material.io/>
- Google. (2023). *Manifest V3 - Chrome Extensions*. Chrome Developers. <https://developer.chrome.com/docs/extensions/mv3/>
- Material Design. (2025). *Material Design 3*. Google. <https://m3.material.io/>
- Microsoft. (2024). *ASP.NET Core Blazor*. Microsoft Learn. <https://learn.microsoft.com/en-us/aspnet/core/blazor/>
- Microsoft. (2024). *.NET Multi-platform App UI (.NET MAUI)*. Microsoft Learn. <https://learn.microsoft.com/en-us/dotnet/maui/>
- Microsoft. (2024). *BlazorWebView for .NET MAUI*. Microsoft Learn. <https://learn.microsoft.com/en-us/dotnet/maui/user-interface/controls/blazorwebview>
- Mozilla. (2023). *Progressive web apps (PWAs)*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps
- MudBlazor Team. (2024). *MudBlazor - Blazor component library*. MudBlazor. <https://mudblazor.com/>
- Tailwind Labs. (2024). *Tailwind CSS - A utility-first CSS framework*. Tailwind CSS. <https://tailwindcss.com/>
- WebAssembly Community Group. (2023). *WebAssembly*. WebAssembly. <https://webassembly.org/>