

# XyloNet - Complete Project Documentation

## Table of Contents

1. [Executive Summary](#)
2. [Project Vision & Mission](#)
3. [Technical Architecture](#)
4. [Smart Contracts Deep Dive](#)
5. [Frontend Application](#)
6. [Feature Breakdown](#)
7. [Deployed Infrastructure](#)
8. [Integration Guide](#)
9. [User Flows](#)
10. [Security Model](#)
11. [Grant Alignment](#)
12. [Future Roadmap](#)
13. [Technical Specifications](#)

## Executive Summary

**XyloNet** is a comprehensive DeFi protocol built exclusively for Arc Network, combining three core primitives into a unified stablecoin SuperExchange:

Component	Description	Status
<b>XyloSwap</b>	StableSwap AMM for stablecoin trading	<input checked="" type="checkbox"/> Live
<b>XyloBridge</b>	Circle CCTP V2 cross-chain bridge	<input checked="" type="checkbox"/> Live
<b>XyloVault</b>	ERC-4626 yield aggregation vault	<input checked="" type="checkbox"/> Live

## Key Metrics

- **Network:** Arc Testnet (Chain ID: 5042002)
- **Contracts Deployed:** 6 verified contracts
- **Supported Tokens:** USDC, EURC, USYC
- **Bridge Chains:** 7 networks (Ethereum, Arbitrum, Base, Optimism, Polygon, Avalanche, Arc)
- **Transaction Cost:** ~\$0.01 per operation
- **Finality:** <350ms deterministic

## Project Vision & Mission

### Vision

To become the premier stablecoin exchange infrastructure on Arc Network, enabling seamless trading, bridging, and yield generation for the internet economy.

### Mission

1. **Democratize Stablecoin Access:** Enable anyone to swap, bridge, and earn yield on stablecoins with minimal friction

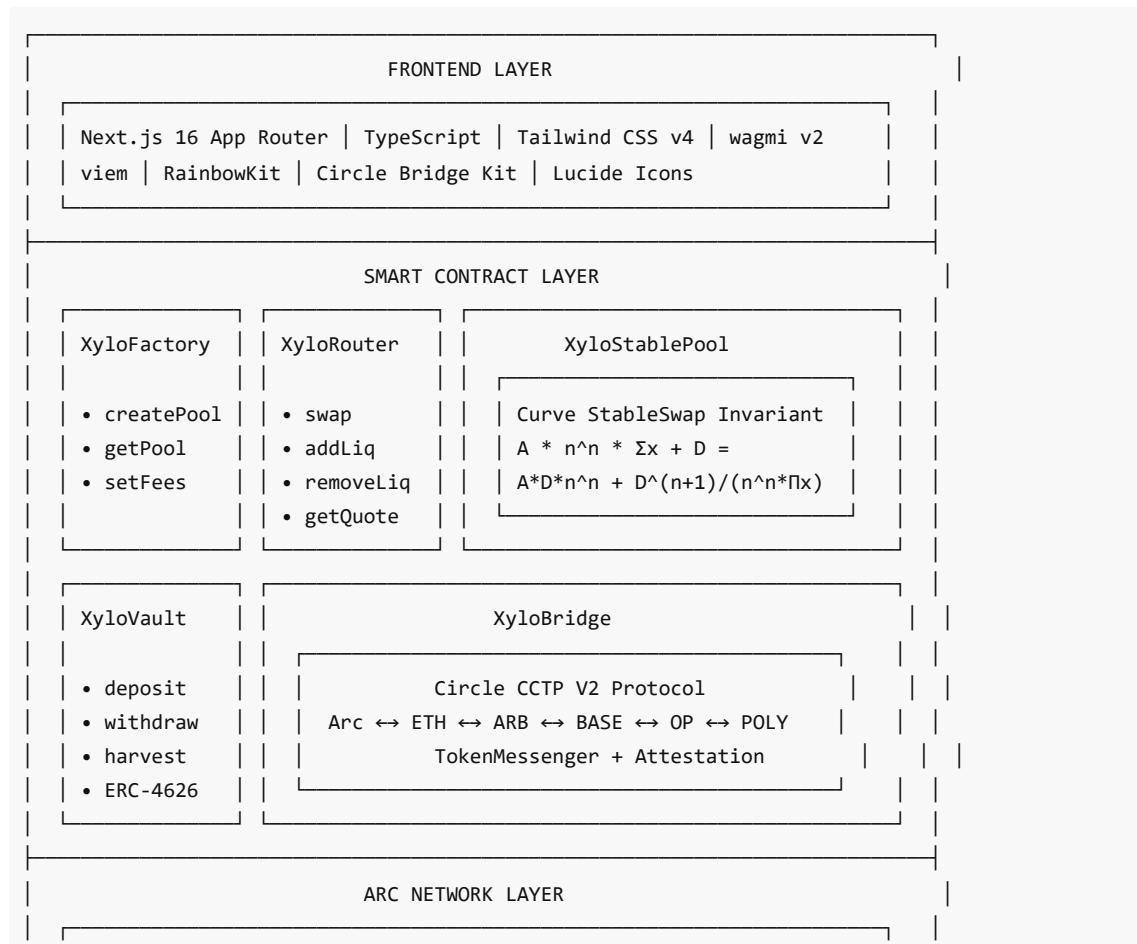
2. **Maximize Capital Efficiency:** Use StableSwap curves for minimal slippage on large trades
3. **Enable Global Payments:** Leverage CCTP V2 for instant cross-border USDC transfers
4. **Generate Sustainable Yield:** Provide safe, auto-compounding yield strategies

## Core Value Propositions

Value	How XyloNet Delivers
<b>Instant Settlement</b>	<350ms deterministic finality on Arc Network
<b>Predictable Costs</b>	~\$0.01 per transaction, denominated in USDC (not volatile ETH)
<b>Native Cross-Chain</b>	Circle CCTP V2 - no wrapped tokens, native USDC everywhere
<b>Real Yield</b>	USYC integration for Treasury-backed yields
<b>Enterprise Ready</b>	Compliance-friendly with opt-in privacy features
<b>Mobile First</b>	Fully responsive UI for mobile traders
<b>Immersive UX</b>	Revolutionary animations, 3D effects, and glassmorphism

## Technical Architecture

### System Overview



Native USDC | Sub-second Finality | USDC Gas | Circle CCTP V2  
Permit2 | Multicall3 | Gateway Wallet | FxEscrow (StableFX)

## Directory Structure

```
XyloNet/
├── contracts/                      # Solidity smart contracts
│   ├── src/
│   │   ├── core/                  # Core DEX infrastructure
│   │   │   └── XyloFactory.sol    # Pool deployment factory
│   │   │   └── XyloRouter.sol     # Swap routing & execution
│   │   │   └── XyloStablePool.sol # StableSwap AMM
│   │   └── XyloERC20.sol         # LP token standard
│   ├── bridge/
│   │   └── XyloBridge.sol        # CCTP V2 bridge wrapper
│   ├── vault/
│   │   └── XyloVault.sol         # ERC-4626 yield vault
│   └── interfaces/                 # Contract interfaces
├── scripts/
│   ├── deploy.js                   # Deployment automation
│   └── add-liquidity.js           # Liquidity bootstrap
├── artifacts/                     # Compiled ABIs
├── hardhat.config.js
└── package.json

├── frontend/                      # Next.js application
│   ├── src/
│   │   ├── app/                  # App Router pages
│   │   │   └── page.tsx          # Home / Swap interface
│   │   │   └── bridge/            # Bridge interface
│   │   │   └── pools/              # Liquidity pools
│   │   │   └── vault/              # Yield vault
│   │   │   └── history/            # Transaction history
│   │   └── layout.tsx            # Root layout
│   └── globals.css                 # Global styles + mobile CSS
├── components/
│   ├── swap/
│   │   └── SwapWidget.tsx       # Token swap component
│   ├── bridge/
│   │   └── BridgeWidget.tsx     # Bridge component
│   ├── ui/
│   │   └── Toast.tsx            # Transaction notifications
│   │   └── TokenLogos.tsx        # Token & chain logos
│   │   └── Skeleton.tsx          # Loading states
│   │   └── Tooltip.tsx           # Info tooltips
│   │   └── Confetti.tsx          # Success celebration
│   └── EmptyState.tsx             # Empty state components
└── ├── Header.tsx                 # Navigation header
    └── Footer.tsx                # Site footer
```

```

    |   |   |   └── Providers.tsx      # Web3 providers
    |   |   ├── config/
    |   |   |   └── wagmi.ts        # Wallet configuration
    |   |   |   └── constants.ts    # Contract addresses
    |   |   └── abis/              # Contract ABIs
    |   └── lib/
    |       └── utils.ts          # Utility functions
    └── transactions.ts        # Transaction history
    └── package.json

    └── README.md                # Quick start guide
    └── XYLONET_PROJECT.md       # This detailed documentation
    └── PROJECT_OVERVIEW.md      # Project summary
    └── IMPLEMENTATION_PLAN.md   # Development roadmap
    └── SECURITY.md              # Security considerations
    └── CONTRIBUTING.md         # Contribution guidelines
    └── LICENSE                  # MIT License

```

## Smart Contracts Deep Dive

### 1. XyloFactory

**Purpose:** Manages pool deployment and protocol configuration.

**Address:** 0x60EDeFB094B84BBC6430cc130B358A43Ba1979e2

**Key Functions:**

Function	Parameters	Description
createPool	tokenA, tokenB, A	Deploy new StableSwap pool
getPool	tokenA, tokenB	Get pool address
setSwapFee	fee	Update global swap fee (owner)
setFeeRecipient	recipient	Set protocol fee recipient
setPaused	status	Emergency pause (owner)

**Protocol Parameters:**

Parameter	Value	Description
Swap Fee	0.04% (4 bps)	Fee charged on each swap
Protocol Fee	50%	Portion of swap fee to protocol
Default A	200	Amplification parameter

### 2. XyloRouter

**Purpose:** User-friendly interface for swaps and liquidity operations.

**Address:** 0x73742278c31a76dBb0D2587d03ef92E6E2141023

**Swap Interface:**

```
struct SwapParams {  
    address tokenIn;  
    address tokenOut;  
    uint256 amountIn;  
    uint256 minAmountOut;  
    address to;  
    uint256 deadline;  
}  
  
function swap(SwapParams calldata params) external returns (uint256 amountOut);
```

**Liquidity Interface:**

```
function addLiquidity(AddLiquidityParams calldata params) external returns (uint256 lpTokens);  
function removeLiquidity(RemoveLiquidityParams calldata params) external returns (uint256, uint256);
```

**Quote Functions:**

```
function getAmountOut(address tokenIn, address tokenOut, uint256 amountIn)  
    external view returns (uint256);
```

### 3. XyloStablePool

**Purpose:** Implements Curve's StableSwap invariant for minimal slippage.

**Deployed Pools:**

Pool	Address	Tokens
USDC-EURC	0x3DF3966F5138143dce7a9cFDdC2c0310ce083BB1	USDC ↔ EURC
USDC-USYC	0x8296cC7477A9CD12cF632042fDDc2aB89151bb61	USDC ↔ USYC

**StableSwap Curve Formula:**

$$A * n^n * \sum x_i + D = A * D * n^n + D^{n+1} / (n^n * \prod x_i)$$

Where:

- **A** = Amplification parameter (200) - controls curve tightness
- **n** = Number of tokens (2)
- **D** = Total liquidity invariant
- **x<sub>i</sub>** = Reserve of token i

**Why StableSwap?**

- Near-zero slippage for stablecoin pairs at normal volumes
  - 10-100x better rates than constant product AMM for pegged assets
  - Ideal for USDC ↔ EURC (both pegged to fiat)
- 

## 4. XyloBridge

**Purpose:** Cross-chain USDC transfers via Circle's CCTP V2.

**Address:** 0xf7Df65Ce418E938ee8d9a0A0d227A43441fe4641

**CCTP V2 Integration:**

```
function bridgeToChain(
    uint256 amount,
    string calldata chainName,
    address recipient
) external returns (bytes32 nonce);
```

**Supported Chains (CCTP Domain IDs):**

Chain	Domain ID	Network	Status
Ethereum	0	Sepolia	<span style="color: green;">✓</span> Active
Avalanche	1	Fuji	<span style="color: green;">✓</span> Active
Optimism	2	Sepolia	<span style="color: green;">✓</span> Active
Arbitrum	3	Sepolia	<span style="color: green;">✓</span> Active
Base	6	Sepolia	<span style="color: green;">✓</span> Active
Polygon	7	Amoy	<span style="color: green;">✓</span> Active
Arc Network	26	Testnet	<span style="color: green;">✓</span> Active

**Bridge Flow:**

1. User initiates bridge → Approve USDC
  2. XyloBridge calls `depositForBurn` on TokenMessengerV2
  3. Circle attests the burn
  4. Bridge Kit automatically mints on destination chain
  5. User receives native USDC (~30 seconds for fast mode)
- 

## 5. XyloVault

**Purpose:** ERC-4626 compliant yield vault for USDC deposits.

**Address:** 0x240Eb85458CD41361bd8C3773253a1D78054f747

**ERC-4626 Interface:**

```
// Deposit assets, receive shares
function deposit(uint256 assets, address receiver) external returns (uint256 shares);
```

```

// Withdraw assets by burning shares
function withdraw(uint256 assets, address receiver, address owner) external returns (uint256
shares);

// Preview functions
function previewDeposit(uint256 assets) external view returns (uint256);
function previewWithdraw(uint256 shares) external view returns (uint256);

```

#### **Fee Structure:**

Fee Type	Rate	Description
Deposit Fee	0%	No entry fee
Withdraw Fee	0.1%	Small exit fee
Performance Fee	10%	On profit only

## Frontend Application

#### **Technology Stack**

Technology	Version	Purpose
Next.js	16	React framework with App Router
TypeScript	5.x	Type safety
Tailwind CSS	4.x	Utility-first styling
wagmi	2.x	React Hooks for Ethereum
viem	2.x	TypeScript Ethereum client
RainbowKit	2.x	Wallet connection
Circle Bridge Kit	Latest	Cross-chain UI
Lucide React	Latest	Icon library

#### **Pages**

Route	Description	Features
/	Swap	Token swap with slippage control, price impact display
/bridge	Bridge	Cross-chain USDC transfer with chain selector
/pools	Pools	Liquidity management, TVL, APY display
/vault	Vault	USDC yield vault with APY tracking
/history	History	Transaction history with filtering

## UI Components

| Component | Location | Purpose | -----|-----|-----|| SwapWidget | components/swap/ | Token swap interface || BridgeWidget | components/bridge/ | Cross-chain bridge UI || Header | components/ | Navigation, wallet connect || Footer | components/ | Social links, credits || Toast | components/ui/ | Transaction notifications || TokenLogo | components/ui/ | Official token/chain logos || Skeleton | components/ui/ | Loading states || Tooltip | components/ui/ | Info tooltips for DeFi terms || Confetti | components/ui/ | Success celebration || **AnimatedBackground** | components/ui/ | Canvas particle system with aurora effect || **TiltCard** | components/ui/ | 3D perspective cards with glare || **AnimatedElements** | components/ui/ | Magnetic buttons, animated numbers || **CommandPalette** | components/ui/ | Keyboard navigation (Ctrl+K) |

## Mobile Responsiveness

The entire application is fully mobile-responsive with:

- **Touch-friendly targets:** Minimum 44-48px height for all interactive elements
- **Bottom sheet modals:** Slide-up modals on mobile for better UX
- **Safe area insets:** Support for notched devices (iPhone X+)
- **Active state feedback:** Visual feedback on touch (scale animations)
- **Responsive typography:** Scaled text sizes for different screens
- **Horizontal scroll:** Filter pills with horizontal scroll on mobile

## Revolutionary UI/UX Features

XyloNet implements cutting-edge visual effects that create an immersive, premium experience:

### AnimatedBackground

- **Canvas-based particle system** with 120+ particles
- **Aurora borealis effect** using radial gradients
- **Connection lines** between nearby particles
- **Mouse interaction** - particles attracted to cursor
- **GPU-accelerated** with requestAnimationFrame

### TiltCard

- **3D perspective transforms** with customizable tilt (0-20°)
- **Dynamic glare effect** that follows mouse movement
- **Holographic shine** on hover
- **Floating animation** for emphasis

### AnimatedElements

Component	Effect
AnimatedNumber	Counting animations with easing
MagneticButton	Buttons attracted to cursor
GlowButton	Pulse glow effects on hover
RippleButton	Material Design ripple effects
FloatingElement	Continuous floating animation
ParallaxLayer	Mouse-following parallax

### **CommandPalette (Ctrl+K)**

- **Keyboard-accessible** command palette
- **Fuzzy search** across all pages
- **Actions:** Navigate, Connect Wallet, Toggle Theme
- **Arrow key navigation** with instant execution

### **Advanced CSS Effects**

Effect	Description
Glassmorphism 2.0	Multi-layer blur with saturation boost
Holographic Shine	Animated gradient overlay
Depth Shadow	Layered shadows for 3D appearance
Stagger Fade	Sequential fade-in for list items
Gradient Text	Animated color cycling
Reduced Motion	Accessibility support for users who prefer minimal motion

---

## **Feature Breakdown**

### **1. XyloSwap (Token Exchange)**

#### **User Journey:**

1. Connect wallet → Select input token (USDC/EURC/USYC)
2. Enter amount → See real-time quote
3. Adjust slippage tolerance (0.1%, 0.5%, 1.0%)
4. Click "Swap" → Approve token (if needed) → Execute swap
5. See confetti celebration on success

#### **Features:**

- Real-time price quotes via Router contract
- USD value display for amounts
- MAX button for full balance swaps
- Price impact indicator (green/yellow/red)
- Network fee display
- Slippage tolerance settings
- Token switcher with flip animation

---

### **2. XyloBridge (Cross-Chain)**

#### **User Journey:**

1. Connect wallet → Select destination chain
2. Enter USDC amount → See estimated output
3. Click "Bridge" → Approve USDC → Execute bridge
4. Wait for Circle attestation (~30 seconds)
5. USDC automatically arrives on destination chain

#### **Features:**

- Chain selector with official logos
  - Fast transfer mode (~30 seconds)
  - Step-by-step progress indicator
  - Real-time status updates
  - Auto-delivery via Bridge Kit
  - Estimated fees and time display
- 

### **3. XyloPools (Liquidity Provision)**

#### **User Journey:**

1. View available pools → See TVL, APY, your position
2. Click "Add" → Enter token amounts
3. Approve tokens (if needed) → Add liquidity
4. Receive LP tokens representing pool share
5. Earn 0.04% of all trades

#### **Features:**

- Pool analytics (TVL, 24h volume, fees)
  - Your LP position and share percentage
  - Add/remove liquidity modals
  - Real-time reserve data
  - Volume sparklines
- 

### **4. XyloVault (Yield Generation)**

#### **User Journey:**

1. View vault stats → See TVL, APY, your position
2. Click "Deposit" → Enter USDC amount
3. Approve USDC → Execute deposit
4. Receive xyUSDC vault shares
5. Earn auto-compounding yield

#### **Features:**

- Real-time APY display (calculated dynamically)
  - Your position value in USDC
  - Deposit/withdraw modals
  - Strategy description
  - APY history sparkline
- 

### **5. Transaction History**

#### **User Journey:**

1. View all past transactions
2. Filter by type (swap, liquidity, vault, bridge)
3. See status (success, pending, failed)
4. Click to view on block explorer

#### **Features:**

- Type badges with colors

- Status indicators
  - Time formatting (relative)
  - Explorer links
  - Local + API data merge
  - Clear history option

# Deployed Infrastructure

## XyloNet Contracts

Contract	Address	Explorer Link
XyloFactory	0x60EDeFB094B84BBC6430cc130B358A43Ba1979e2	<a href="#">View</a>
XyloRouter	0x73742278c31a76dBb0D2587d03ef92E6E2141023	<a href="#">View</a>
XyloBridge	0xf7Df65Ce418E938ee8d9a0A0d227A43441fe4641	<a href="#">View</a>
XyloVault	0x240Eb85458CD41361bd8C3773253a1D78054f747	<a href="#">View</a>
USDC-EURC Pool	0x3DF3966F5138143dce7a9cFDdC2c0310ce083BB1	<a href="#">View</a>
USDC-USYC Pool	0x8296cC7477A9CD12cF632042fDDc2aB89151bb61	<a href="#">View</a>

## Arc Network Native Contracts

Contract	Address	Purpose
USDC (Native)	0x3600	Native stablecoin
EURC	0x89B50855Aa3bE2F677cD6303Cec089B5F319D72a	Euro stablecoin
USYC	0xe9185F0c5F296Ed1797AaE4238D26CCaBEadb86C	Yield-bearing stablecoin
TokenMessengerV2	0x8FE6B999Dc680CcFDD5BF7EB0974218be2542DAA	CCTP V2 messenger
MessageTransmitterV2	0xE737e5cEBEEBa77EFE34D4aa090756590b1CE275	CCTP V2 transmitter
Gateway Wallet	0x0077777d7EBA4688BDeF3E311b846F25870A19B9	Fiat on/off ramp
FxEscrow	0x1f91886C7028986aD885ffCee0e40b75C9cd5aC1	StableFX escrow
Permit2	0x000000000022D473030F116dDEE9F6B43aC78BA3	Gasless approvals
Multicall3	0xA11bde05977b3631167028862bE2a173976CA11	Batch calls

## Network Configuration

Parameter	Value
Network Name	Arc Testnet
Chain ID	5042002
RPC URL	<a href="https://rpc.testnet.arc.network">https://rpc.testnet.arc.network</a>

WebSocket	wss://rpc.testnet.arc.network
Block Explorer	<a href="https://testnet.arcscan.app">https://testnet.arcscan.app</a>
Currency	USDC
Block Time	<350ms

## Integration Guide

### Quick Start

```
# Clone repository
git clone https://github.com/Panchu11/XyloNet.git
cd XyloNet

# Install frontend dependencies
cd frontend
npm install

# Start development server
npm run dev
```

### Contract Integration (JavaScript/TypeScript)

```
import { createPublicClient, http } from 'viem';
import { CONTRACTS, ARC_NETWORK } from './config/constants';
import { XYLO_ROUTER_ABI } from './config/abis';

// Create client
const client = createPublicClient({
  chain: {
    id: 5042002,
    name: 'Arc Testnet',
    rpcUrls: { default: { http: ['https://rpc.testnet.arc.network'] } }
  },
  transport: http()
});

// Get swap quote
const quote = await client.readContract({
  address: CONTRACTS.ROUTER,
  abi: XYLO_ROUTER_ABI,
  functionName: 'getAmountOut',
  args: [CONTRACTS.USDC, CONTRACTS.EURC, parseUnits('100', 6)]
});

// Execute swap
const hash = await walletClient.writeContract({
```

```

address: CONTRACTS.ROUTER,
abi: XYLO_ROUTER_ABI,
functionName: 'swap',
args: [{
  tokenIn: CONTRACTS.USDC,
  tokenOut: CONTRACTS.EURC,
  amountIn: parseUnits('100', 6),
  minAmountOut: quote * 99n / 100n,
  to: userAddress,
  deadline: BigInt(Math.floor(Date.now() / 1000) + 3600)
}]
});

```

## React Integration (wagmi)

```

import { useReadContract, useWriteContract } from 'wagmi';
import { parseUnits } from 'viem';
import { CONTRACTS } from '@/config/constants';
import { XYLO_ROUTER_ABI } from '@/config/abis';

function SwapComponent() {
  // Read quote
  const { data: quote } = useReadContract({
    address: CONTRACTS.ROUTER,
    abi: XYLO_ROUTER_ABI,
    functionName: 'getAmountOut',
    args: [CONTRACTS.USDC, CONTRACTS.EURC, parseUnits('100', 6)]
  });

  // Write swap
  const { writeContract } = useWriteContract();

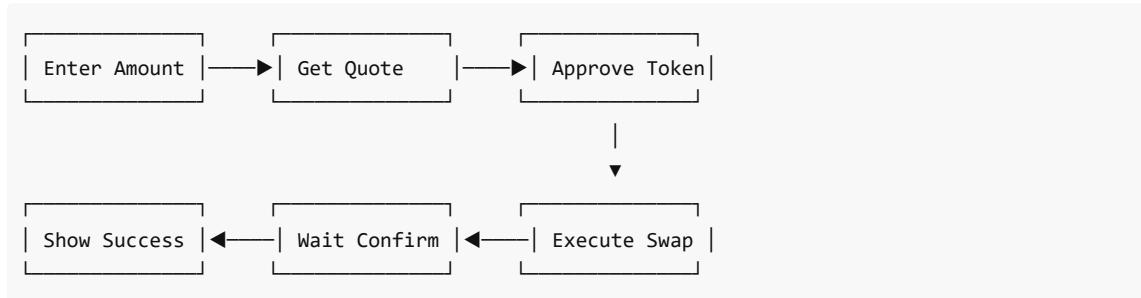
  const handleSwap = () => {
    writeContract({
      address: CONTRACTS.ROUTER,
      abi: XYLO_ROUTER_ABI,
      functionName: 'swap',
      args: [
        {
          tokenIn: CONTRACTS.USDC,
          tokenOut: CONTRACTS.EURC,
          amountIn: parseUnits('100', 6),
          minAmountOut: quote * 99n / 100n,
          to: address,
          deadline: BigInt(Math.floor(Date.now() / 1000) + 3600)
        }
      ]
    });
  };

  return <button onClick={handleSwap}>Swap</button>;
}

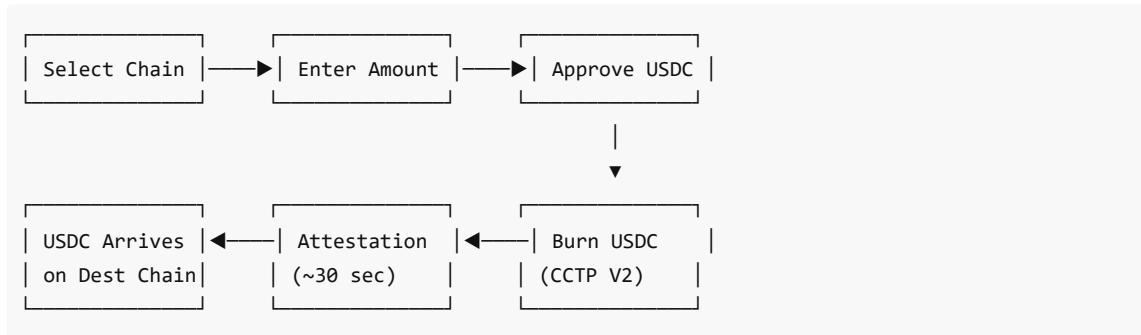
```

## User Flows

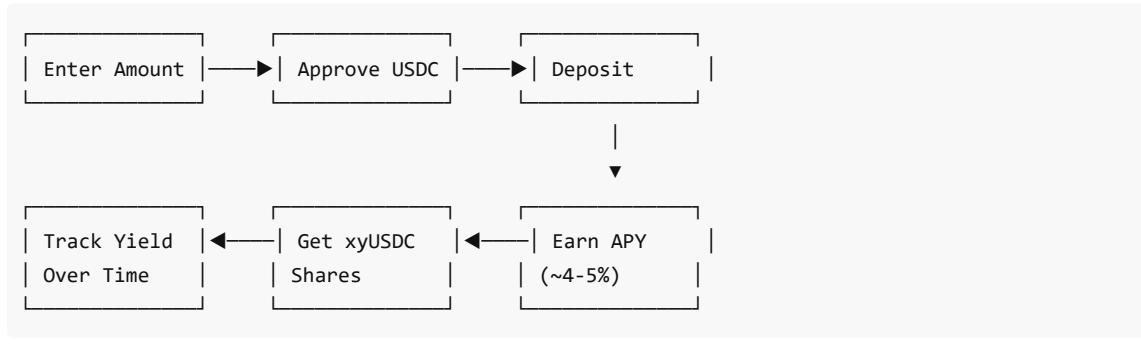
### Swap Flow



### Bridge Flow



### Vault Flow



## Security Model

### Smart Contract Security

Safeguard	Implementation
Reentrancy Protection	lock modifier on all state-changing functions
CEI Pattern	Checks-Effects-Interactions ordering
Access Control	Owner-only for sensitive operations

Deadline Validation	All trades require valid deadline
Minimum Output	All swaps enforce minAmountOut
Zero Address Checks	Validate all address parameters
Safe Math	Solidity 0.8+ built-in overflow checks
Decimal Handling	Proper scaling for 6/18 decimal tokens

## Known Limitations

1. No emergency pause (planned for v2)
2. StableSwap assumes tokens maintain peg
3. Pool imbalance can increase slippage
4. Single owner for protocol contracts

## Audit Status

- Internal review completed
- External audit planned for mainnet launch

## Grant Alignment

XyloNet directly addresses the [Arc Builders Fund](#) priority verticals:

Arc Builders Fund Vertical	XyloNet Implementation
<b>Always-on Markets &amp; Capital Formation</b>	24/7 DEX with instant settlement, concentrated liquidity
<b>Onchain FX</b>	Multi-stablecoin swaps (USDC ↔ EURC ↔ USYC)
<b>Offchain Assets &amp; Credit</b>	USYC yield integration for Treasury-backed returns
<b>Cross-border Payments</b>	CCTP V2 bridge for instant global USDC transfers
<b>Account Abstraction</b>	Planned: Gasless transactions via Pimlico
<b>AI Agents</b>	Planned: MCP tool definitions for automated trading

## Future Roadmap

### Phase 4: Advanced Features (Planned)

Feature	Description	Status
StableFX RFQ	Integration with FxEscrow for large trades	Planned
Account Abstraction	Gasless transactions via Pimlico/ZeroDev	Planned
AI Agent Interface	MCP tools for automated trading	Planned
Limit Orders	Time-weighted average price orders	Planned

### Phase 5: Governance & Token (Planned)

Feature	Description	Status
XYLO Token	Protocol governance token	Planned
Staking	Lock XYLO for fee sharing	Planned
DAO Governance	On-chain proposal system	Planned
Fee Distribution	Protocol fees to stakers	Planned

### Phase 6: Mainnet Launch (Planned)

Milestone	Description	Status
Security Audit	External audit by top firm	Planned
Mainnet Deployment	Deploy to Arc mainnet	Planned
Liquidity Mining	Bootstrap liquidity program	Planned
Exchange Listings	CEX listings for XYLO	Planned

## Technical Specifications

### Gas Estimates

Operation	Gas Units	Cost (USDC)
Swap	~120,000	~\$0.0012
Add Liquidity	~180,000	~\$0.0018
Remove Liquidity	~150,000	~\$0.0015
Bridge Out	~100,000	~\$0.0010
Vault Deposit	~80,000	~\$0.0008
Vault Withdraw	~90,000	~\$0.0009
Token Approval	~45,000	~\$0.00045

### Performance Metrics

Metric	Value
Block Time	<350ms
Transaction Finality	Deterministic
RPC Response Time	<100ms
Frontend Load Time	<2s

Mobile Compatibility	100%
----------------------	------

## Browser Support

Browser	Version	Status
Chrome	90+	<input checked="" type="checkbox"/> Supported
Firefox	90+	<input checked="" type="checkbox"/> Supported
Safari	14+	<input checked="" type="checkbox"/> Supported
Edge	90+	<input checked="" type="checkbox"/> Supported
Mobile Safari	iOS 14+	<input checked="" type="checkbox"/> Supported
Mobile Chrome	Android 10+	<input checked="" type="checkbox"/> Supported

---

## Resources

Resource	Link
GitHub Repository	<a href="https://github.com/Panchu11/XyloNet">https://github.com/Panchu11/XyloNet</a>
Arc Network Docs	<a href="https://docs.arc.network">https://docs.arc.network</a>
Arc Testnet Explorer	<a href="https://testnet.arctscan.app">https://testnet.arctscan.app</a>
Circle Faucet	<a href="https://faucet.circle.com">https://faucet.circle.com</a>
Circle CCTP Docs	<a href="https://developers.circle.com/stablecoins/cctp-getting-started">https://developers.circle.com/stablecoins/cctp-getting-started</a>
Arc Builders Fund	<a href="https://www.circle.com/blog/introducing-the-arc-builders-fund">https://www.circle.com/blog/introducing-the-arc-builders-fund</a>
X (Twitter)	<a href="https://x.com/Xylonet">https://x.com/Xylonet</a>

---

## License

MIT License - see [LICENSE](#) file for details.

---

Built with ❤ by ForgeLabs for the Arc Network ecosystem