S3MT Smart Contract Stack Selection — Technical Specification

This document outlines the technical architecture, tooling decisions, module designs, and implementation phases for the S3MT smart contract ecosystem on Solana. It defines how key functionality—including token minting, burning, presale operations, DAO governance, and token/NFT metadata—will be structured using modern Solana-native tooling.

---

1. Overview

S3MT is a real-world asset (RWA) backed token deployed on Solana. The system is structured around:

- A fungible token (S3MT) with dynamic minting and burning
- A presale module supporting phased access and payment handling
- Treasury management for splitting funds across operations, development, and discretionary allocations
- Founders NFTs to unlock reward rights and provide verifiable access for founder distributions
- DAO governance through token staking and proposal voting

This architecture will be built in **Rust using the Anchor framework**, deployed to Solana devnet/testnet/mainnet environments, and accessible via a web frontend with wallet integration.

---

2. Core Components

2.1 Anchor Framework

Role: Core development environment for all Solana smart contracts.

Functionality:

- Modular program structure using Anchor macros
- Type-safe instruction handling
- PDA-based account management for deterministic contract addresses
- Built-in support for upgradable programs and access control

Modules:

- `mint.rs`: Handles token minting post-purchase
- `burn.rs`: Handles token burn and calculates treasury value-based returns
- `presale.rs`: Presale lifecycle logic, including tiering and allocation
- `founders.rs`: Founder NFT gating logic for special mint privileges
- `dao.rs`: DAO staking, voting, and proposal management

2.2 SPL Token Program

Role: Token interface and base S3MT token standard

Token Specs:

- Decimals: 6
- Initial supply: Minted on demand via presale logic
- Burnable: Enabled
- Metadata: Enhanced via Metaplex if needed

Tooling: SPL CLI and Anchor CPI interfaces

2.3 Metaplex Token Metadata

Role: Used for:

- Founders NFTs
- Optional branding metadata on S3MT token (e.g. icon, website)

Features:

- Immutable metadata attached to NFTs
- Optional updates via authority control
- On-chain verification of NFT ownership + metadata logic

2.4 Solana Pay

Role: UX layer for presale purchase flows

Functionality:

- Buyers initiate USDC or SOL transfers from wallet
- Transaction includes metadata to identify tier and wallet
- Backend mints S3MT tokens and sends to buyer's wallet

Presale Tiers (Example):

- Founders: 100% token allocation
- Pre-Family: 95%
- Pre-Launch: 90%
- Launch: 85%
- Post-Launch: 80%

2.5 Helius SDK

Role: Event tracking, transaction logging, and webhook-based notifications

Tracked Events:

- Mint and burn events
- Token transfer logs
- Founders NFT minting
- DAO proposals and votes (future phase)

Integration Use Cases:

- Admin dashboards
- Slack/Discord alerts
- Analytics dashboards

---

3. System Architecture

Component	Framework/Tool	Functionality
Token Mint/Burn	Anchor + SPL Token	Secure, validated token issuance and redemption
Presale Contract	Anchor + Solana Pay	Phased token sale with USDC/SOL payments
Founders NFTs	Metaplex Metadata	NFT-based access gating + metadata

Treasury Logic	Anchor	Allocation to dev, ops, marketing, founders, buyers
Event Tracking	Helius Webhooks	Real-time insights into transactions
DAO Governance	Anchor Modules + Realm	Governance proposals, voting, and token staking (Phase 2+)

---

4. Deployment Strategy

Environments

- **Local -** local development of everything, using the “validator” container, located at http://validator:8899
- **Devnet**: Initial testing of mint/burn + presale contract
- **Testnet**: Integrated testing with frontend + Solana Pay
- **Mainnet**: Final audited deployment with phased presale activation

Deployment Workflow

1. Compile and deploy smart contracts via Anchor CLI
2. Set token authorities and PDAs
3. Integrate frontend using wallet adapter + Solana Pay
4. Register webhooks via Helius API
5. Launch Founders NFT mint
6. Monitor using custom admin dashboard (optionally powered by Helius + Supabase)

---

5. Technical Requirements

- Rust 1.85+
- Anchor 0.30+
- Solana CLI 2.1.16
- SPL CLI tools
- Metaplex CLI (for NFT minting)
- Helius API key
- Phantom, Backpack, or Solflare wallet support on frontend

---

6. Roadmap Alignment

Phase	Deliverables
MVP	Token mint/burn, presale logic, Founders NFT support
Presale Launch	Wallet-based payments, live token delivery, early telemetry
Governance Prep	Treasury tracking, NFT gating, DAO prototype
DAO Expansion	Full staking + voting, proposal creation, community decision flow

---

This specification provides the technical foundation to launch, evolve, and govern the S3MT token ecosystem. It is designed to ensure security, modularity, and transparency while enabling composability with Solana-native tools and future integrations.