The Token That Protects Authenticity.

AFTK is a utility token designed to verify digital ownership, prevent forgery, and reward authenticity — all secured on the Polygon blockchain.

AFTK — AntiFakeToken

Whitepaper v2.0

Authenticity is the new currency.
A blockchain token powering real-world trust.
October 2025

AFTK Whitepaper Contents

Executive Summary 3

1. Background: Why Authenticity Matters 3

2. Problem Statement 4

3. AntiFake App Overview 4

4. QR Proof Protocol 5

5. Smart Contract Design 6

6. Backend & API Architecture 8

7. Tokenomics and Economic Model 9

8. Deflationary Mechanics 10

9. Revenue & Sustainability 10

10. Governance, Control and Upgradability 11

11. Risks and Limitations 12

12. Legal Disclaimer 12

- 12.1 Nature of the Token (Utility Only) 12
- 12.2 No Financial, Investment, or Security Characteristics 13
- 12.3 No Guarantees or Forward-Looking Commitments 13
- 12.4 Regulatory and Jurisdictional Considerations 13
- 12.5 User Responsibility & Risk Acknowledgement 13

12. Legal Disclaimer 14

- 12.1 Nature of the Token (Utility Only) 14
- 12.2 No Financial, Investment, or Security Characteristics 14
- 12.3 No Guarantees or Forward-Looking Commitments 14
- 12.4 Regulatory and Jurisdictional Considerations 15
- 12.5 User Responsibility & Risk Acknowledgement 15
- 12. Legal Disclaimer 15
  - 12.6 Legal Framework Overview 15
  - 12.6.1 Utility Token Classification 15
  - 12.6.2 No Financial Incentives or Return Expectations 16
  - 12.6.3 Custodial vs. Self-Custodial Mode 16
  - 12.6.4 No Token Sale, No Fundraising, No Investment Rounds 17
  - 12.6.5 FIAT Payments and Virtual Credits 17
  - 12.6.6 Compliance with App Store & Google Play Rules 17

12.6.7 Evolving Regulatory Landscape 18
12.7 MiCA Compliance Statement (EU Regulation 2023/1114) 18
12.7.1 Token Classification Under MiCA 18
12.7.2 No Public Offering Under MiCA Definitions 19
12.7.3 Whitepaper Requirements Applicability 19
12.7.4 Custodial Services & MiCA Compliance 19
12.7.5 Exchange Listings & MiCA 19
12.7.6 Ongoing Monitoring 19
13. Conclusion 20
14. System Architecture Overview 20
15. Backend Services (Human-Friendly View) 20
16. End-to-End Verification Flow (From User to Blockchain) 21
17. Public Verification Page 22
18. Custodial Credits, Deposits and Wallet Linking 23
19. User Experience and Onboarding 23
20. Strategic Positioning 24
21. Advanced Users: Wallet Linking & Self-Custody 24
22. Internal Credits & Signup Rewards 25
23. On-Chain Deposits & Scanner 26
24. Creator Monetisation & Tips with AFTK 26
Appendix A — Diagrams & Flows 27

Executive Summary

AFTK (AntiFakeToken) is a blockchain-powered ecosystem designed to prove digital authenticity in a fast, accessible, and user-friendly way.

Unlike speculative tokens launched on promises, AFTK is directly integrated with a live, production-ready application — the AntiFake mobile app for Android and iOS. Users can capture or upload content, verify originality, and generate on-chain QR Proof certificates in seconds, without manually interacting with wallets, private keys, seed phrases, or exchanges.

AFTK introduces a practical authenticity layer for the modern digital world, where AI-generated images, deepfakes and large-scale manipulation make it increasingly difficult to determine what is real. By combining blockchain immutability with intuitive mobile UX, the AntiFake platform enables anyone — creators, brands, consumers, and social users — to verify content or claim ownership with a single tap.

This whitepaper outlines the motivation behind AFTK, the architecture of the AntiFake platform, the design of the AFTK token and its smart contracts, the backend infrastructure, the economic and deflationary model, and the long-term roadmap towards global adoption. The emphasis is on real-world utility rather than speculation: verifiable ownership and originality of digital content, backed by transparent on-chain records, a fixed-supply utility token, and a sustainable FIAT-driven business model.

AFTK is not a financial instrument, investment product, or security. It is a utility token used exclusively within the AntiFake ecosystem to power authenticity verification, reward participation, and support on-chain operations. Its purpose is functional: to enable users to interact with the authenticity protocol. Nothing in this document should be interpreted as investment advice or a promise of financial returns.

AFTK's mission is simple:

make digital authenticity accessible to everyone — and verifiable forever.

1. Background: Why Authenticity Matters

Digital content used to be scarce and relatively easy to attribute. Today, AI-generated images, deepfakes and large-scale manipulation are common. Photos can be fabricated, identities can be spoofed, and original works can be copied or resold without consent. Creators, brands and everyday users face three core problems:

• It is difficult to prove that you created a specific piece of content first.
• It is difficult to verify whether a given image or media file is genuine.
• There is no simple, universal way for non-technical users to check authenticity.

Traditional watermarking and centralized databases do not solve this at scale. They depend on trusted intermediaries and can be modified or censored. Blockchain, on the other hand, offers an immutable public ledger that can store proofs of existence and ownership in a verifiable and permanent way.

AFTK leverages these properties to build a simple, user-friendly authenticity layer that anyone can use, even if they have never interacted with crypto before.

2. Problem Statement

The AntiFake project is built around the following observations:

• Content forgery is cheap, fast, and improving with AI.
• Verification tools are fragmented, complex or inaccessible to everyday users.
• Creators lack a portable, standardized credential that travels with their work.
• Platforms have no neutral, open infrastructure for authenticity that they can integrate.

Without a robust authenticity infrastructure, misinformation spreads, creators lose revenue and trust, and platforms face reputational risk. There is a need for a neutral verification layer that is:

• Cryptographically secure and publicly auditable.
• Independent of any single platform or company.
• Accessible via simple user experiences on mobile and web.
• Economically sustainable without relying on speculation.

AFTK is designed as this missing layer: a neutral authenticity protocol backed by a utility token and a live, production app.

3. AntiFake App Overview

The AntiFake mobile application is the primary user interface for the AFTK ecosystem. It is available for Android and iOS and is built for non-technical users. The app hides blockchain complexity behind familiar flows and clear language.

Core capabilities:

1. Capture or upload content
   Users can take a photo directly in the app or upload an existing image or media file. The file never needs to leave the user-facing environment in raw form; instead a cryptographic hash is computed client-side or server-side.

2. On-chain authenticity registration
The content hash is sent to the AntiFake backend, which communicates with the AFCertRegistry smart contract. The hash is registered along with an owner address and a timestamp, creating a permanent authenticity record on the Polygon blockchain.

3. QR Proof generation
For each registered hash, the system generates a signed QR payload. The QR encodes either
a verifier URL or a compact payload that can be independently validated. Users can save, share or embed this QR Proof anywhere: in social posts, product pages, marketplaces or printed materials.

4. Public verification
Anyone scanning a QR Proof is redirected to a verification page that displays: whether the hash exists on-chain, who owns it, when it was registered, and a preview of the associated content (if available). This enables frictionless authenticity checks without requiring wallets or browser extensions.

To encourage adoption, every new user receives 25 free verification credits backed by AFTK.
This allows them to experience the full authenticity flow before deciding whether to purchase additional credits or interact with the token directly.

4. QR Proof Protocol

The QR Proof protocol transforms an on-chain authenticity record into a portable, verifiable certificate.

4.1 Content hashing
For each uploaded file, the AntiFake backend computes a deterministic SHA-256 hash. The hash is normalized into a 32-byte value represented as a 0x-prefixed 64-character hex string. The original file does not need to be stored on-chain; only the hash is anchored, preserving user privacy while enabling strong verification.

4.2 Registration on-chain
The hash is registered in the AFCertRegistry contract, which stores two fields per hash: the owner address and a 64-bit timestamp (seconds since Unix epoch). Once stored, the record cannot be deleted or overwritten. Attempts to register the same hash twice are rejected.

4.3 QR payload
To generate a QR Proof, the backend creates a JSON payload containing at least:

- v – protocol version
- h – the normalized hash (0x + 64 hex)
- iss – issuer identifier (the AntiFake signing key)

• alg – signature algorithm
• ts – payload creation timestamp

In "verbose" mode the payload may also include the chain name, contract address, and transaction hash of the on-chain registration. The payload is serialized as JSON, base64url-encoded, and then signed using an EIP-191 compatible Ethereum signing key.

4.4 Verification
Verifiers receive two inputs: the payload and the signature. Using the same EIP-191 standard, they recover the signing address and check that it matches the publicly known AntiFake QR signer address. They then decode the payload, validate the format, and use the embedded hash to query on-chain and database records. Optionally, a user may also supply a file or hash to prove that their content matches the registered hash.

4.5 QR images
The QR image itself is generated from either a verifier URL or a compact scheme URL (afcert:<payload>.<sig>). Thumbnails and QR images can be stored and served by the AntiFake backend, allowing rich previews and galleries inside the app and web interface.

5. Smart Contract Design

The AFTK ecosystem currently uses two primary smart contracts on Polygon:

• AFCoin (AFC) — the ERC-20 utility token.
• AFCertRegistry — the authenticity registry for content hashes.

Both contracts are implemented in Solidity using audited OpenZeppelin libraries, and are deployed with a fixed configuration that prevents arbitrary minting or hidden supply changes.

5.1 AFCoin (AFC)
AFCoin is an ERC-20 token with the following properties:

• Name: AFCoin
• Symbol: AFC
• Decimals: 18
• Initial supply: 1,000,000,000 AFC minted once in the constructor
• No further minting: there is no public or internal mint function beyond the initial supply creation.

Ownership and control:
• The contract inherits Ownable. The owner is a hardware-secured address (Ledger) used for managing a small set of configuration parameters.
• The owner can change the treasury address and adjust the burn ratio within predefined bounds, or pause transfers in an emergency via the Pausable extension.

Treasury and fees:
- The contract maintains a treasury address which receives the non-burned portion of any explicit fee payments.
- The payFee(uint256 totalFee) function allows a user (or backend on their behalf) to pay a fee in AFC. The function debits the caller, burns a configurable percentage (burnBps) and sends the remainder to the treasury.
- burnBps is expressed in basis points (1/10,000) and capped at 10,000 (100%).

Pausability:
- By calling pause() or unpause(), the owner can temporarily stop all token transfers and fee payments in case of a security incident, misconfiguration or migration event. This gives the project an additional protection layer at early stages, while still preserving a fixed total supply.

5.2 AFCertRegistry

AFCertRegistry is a minimal and purpose-built registry for authenticity records.

State:
- For each content hash (bytes32), the contract stores a Record struct containing:
  – owner: address that currently controls the record
  – timestamp: uint64 timestamp of registration

Functions:
- register(bytes32 contentHash)
  Allows any user to register a new hash where msg.sender becomes the owner. The function
  rejects a zero hash and any hash that has already been registered.

- registerFor(bytes32 contentHash, address owner)
  Allows the contract owner (the platform) to register a hash in favour of a specified owner address. This enables custody and migration flows where the platform temporarily holds ownership before handing it over.

- transferOwnershipOf(bytes32 contentHash, address newOwner)
  Allows the owner account to reassign ownership of an existing record to a new address. This is useful when users migrate from a custodial to a self-custodial wallet, or when platforms perform account recovery flows.

- ownerOf(bytes32 contentHash), timestampOf(bytes32 contentHash), exists(bytes32 contentHash)
  Read-only view functions that provide efficient access to registry data.

Design goals:
- No loops, no complex logic — gas efficient and easy to audit.
- No deletion — the history of registrations remains intact.

• The owner account holds limited, well-defined powers; it cannot alter past timestamps or erase records.

6. Backend & API Architecture

While the blockchain provides the trust anchor, most user interactions happen through the AntiFake backend and API layer. The backend is implemented using ASP.NET Core and uses Entity Framework for persistence.

Key components:

• HashService — computes SHA-256 hashes of uploaded files and normalizes them to the 0x-prefixed format used on-chain.
• BlockchainService — wraps web3 calls to AFCoin and AFCertRegistry, providing methods such as RegisterForAsync, StatusAsync, PayFeeWithCostAsync and ExistsOwnerAsync.
• WalletService — manages internal user balances and accounts for in-app credits.
• QrSigningService — generates and verifies signed QR payloads using an Ethereum signing key that is never exposed directly.
• AssetPreviewService — generates thumbnails and QR images for better UX.

6.1 Register-and-Charge flow

The RegisterAndCharge endpoints implement a complete flow for authenticity registration and payment in a single call:

1. A user sends a file or precomputed hash via a multipart/form-data request.
2. The backend computes or validates the hash.
3. It selects the correct owner address: either the user's primary wallet (Self mode) or the platform treasury (Custody mode).
4. The backend debits an internal AFTK balance from the user, ensuring that they have sufficient credits.
5. Using BlockchainService, it calls registerFor(hash, owner) on AFCertRegistry.
6. All relevant data — hash, owner, transaction hash, chain, contract address, file metadata, gas usage and fees — is stored in AssetRegistrations.
7. Optionally, a thumbnail is created and stored through AssetPreviewService.
8. The response includes the transaction hash, hash, owner and the user's new balance.

This design allows AntiFake to cover gas costs centrally while charging users in AFTK or FIAT, enabling a smooth onboarding path for non-crypto users.

6.2 QR payload endpoints

The qr-payload endpoint creates signed QR payloads for existing registrations. It:

• Validates that the hash exists in the database.
• Builds a compact or verbose JSON payload.
• Signs it using the QR signing key (EIP-191).
• Base64url-encodes the payload and constructs a verifier URL if configured.

• Logs the payload and signature in QrPayloads for audit.
• Optionally creates and stores a QR image through AssetPreviewService.

6.3 Verification endpoints
The verify endpoints accept a payload + signature pair, and optionally a file or hash:

• Decode and parse the JSON payload.
• Validate algorithm and issuer.
• Recover the signing address from the signature and compare it with the expected signer.
• Query on-chain status using BlockchainService.
• Optionally check that a provided file or hash matches the payload hash.
• Return a detailed VerifyResponse that powers the public verification page.

6.4 Asset history endpoints
The my-assets and my-asset-urls endpoints provide a paginated view of a user's registered content, including thumbnails, QR images, timestamps and transaction hashes. This turns the AntiFake app into a personal authenticity portfolio for each user.

7. Tokenomics and Economic Model

Total Supply: 1,000,000,000 AFTK (fixed — no minting).

The token distribution is designed to balance user incentives, project sustainability and market liquidity:

• 40% Ecosystem, Rewards & Liquidity
  – 25% for user rewards, onboarding bonuses and authenticity incentives.
  – 10% for initial and early-stage liquidity pools on decentralized exchanges.
  – 5% for community campaigns, partnerships and future liquidity expansion.

• 30% Team & Development (locked)
  Allocated to core contributors, engineers, designers and operational staff. Tokens are subject to a multi-year vesting schedule with an initial cliff, ensuring alignment with long-term success.

• 20% Strategic Partners / Early Supporters
  Reserved for future partnerships, integrations, infrastructure collaborators and, where applicable, early supporters. These allocations may be subject to their own vesting agreements.

• 10% Reserve Fund
  Held as a strategic buffer for unforeseen costs, strategic opportunities and protocol safety. Not intended for short-term trading.

Launch parameters (illustrative):
• Launch price: €0.005 per AFTK.

• Initial circulating market cap: ~€5M.
• Initial DEX liquidity: ~€2,000 pool (e.g., €1,000 of paired asset + 200,000 AFTK).

The primary revenue source for the project is FIAT-based in-app purchases of verification credits. This allows the platform to operate profitably without selling large amounts of AFTK on the market, reducing sell pressure and supporting long-term price stability.

8. Deflationary Mechanics

AFTK implements a simple, transparent deflationary model tied directly to real usage of the AntiFake app.

Every verification action conceptually consumes 5 AFTK:

• 1 AFTK (20%) — permanently burned, reducing total circulating supply.
• 4 AFTK (80%) — recycled back into the ecosystem: app rewards, maintenance, infrastructure
costs and future growth.

Because the total supply is fixed and there is no minting, each burn is irreversible.
As the number of verified assets grows, the circulating supply of AFTK shrinks while demand for authenticity services increases.

Example scenario (illustrative):

At 20,000 active users performing one verification per month:
• 20,000 verifications × 12 months × 1 AFTK burned = 240,000 AFTK burned annually.

At 500,000 active users with similar behaviour:
• 500,000 verifications × 12 months × 1 AFTK burned = 6,000,000 AFTK burned annually.

Over several years, this creates a meaningful reduction in circulating supply while the ecosystem continues to grow, aligning long-term value with real-world usage instead of pure market speculation.

9. Revenue & Sustainability

The AntiFake platform is designed to reach operational profitability without relying on continuous token sales. The primary revenue stream consists of FIAT in-app purchases where
users buy verification credits using traditional payment methods.

Simplified revenue table (illustrative, based on earlier modelling):

• 20,000 users
– Gross monthly revenue: ~€2,000
– After payment processor and platform fees: ~€1,772

– Estimated gas costs: ~€400
– Net: ~€1,372

• 100,000 users
– Gross monthly revenue: ~€10,000
– After fees: ~€8,860
– Gas costs: ~€2,000
– Net: ~€6,860

• 500,000 users
– Gross monthly revenue: ~€50,000
– After fees: ~€44,300
– Gas costs: ~€10,000
– Net: ~€34,300

Even at modest adoption levels (around 20,000 users), the platform can operate profitably while continuously registering authenticity proofs on-chain and burning AFTK. This enables sustainable long-term development without aggressive token dumping or unsustainable reward schemes.

10. Governance, Control and Upgradability

The current version of the AFTK ecosystem uses a pragmatic governance model suitable for an early-stage utility protocol.

Smart contract control:
• AFCoin is owned by a hardware-secured address controlled by the core team. Owner powers
are intentionally limited to: changing the treasury address, adjusting the burn ratio within safe bounds, and pausing/unpausing transfers. There is no mint function.
• AFCertRegistry is owned by a platform-controlled address which is used purely to perform
registerFor and transferOwnershipOf actions, enabling custody flows and migrations. The owner cannot erase history or modify timestamps.

Road to decentralization:
• Over time, as the protocol matures and risk decreases, ownership rights on AFCoin may be renounced or migrated to a multi-signature or community-governed structure.
• AFCertRegistry may be wrapped or extended with additional interfaces, but the base registry is intentionally minimal and immutable to ensure long-term reliability.

Off-chain governance:
• Protocol parameters such as default verification fees, reward schedules, and app UX changes are currently governed by the core team. In later phases, the project may

introduce advisory councils, community input channels or formal governance mechanisms where appropriate.

11. Risks and Limitations

Like any emerging technology project, AFTK carries technical, market and regulatory risks.

Technical risks:
- Smart contract bugs, while mitigated by using standard libraries and careful review, may still exist.
- Blockchain networks can experience congestion, fee spikes or outages that temporarily affect registration or verification speed.
- Private keys controlling treasury and ownership accounts must be securely managed to prevent compromise.

Market risks:
- Demand for authenticity services may grow slower than expected.
- Token price may be volatile and influenced by broader crypto market cycles beyond the control of the project.
- Competing solutions may emerge in parallel.

Regulatory risks:
- Legal treatment of utility tokens varies by jurisdiction and may change over time.
- The project will continue to monitor local and international regulations and adapt its communications and integrations accordingly.

Limitations:
- The system proves that a given file was registered at a certain time and associated with a given address, but it cannot by itself determine the real-world identity of that address. Off-chain KYC or identity solutions may be layered on top where required.
- If users lose access to their wallets, ownership of registry entries may need to be migrated via platform-assisted flows (for example, from custodial to self-custodial wallets).

12. Legal Disclaimer

12.1 Nature of the Token (Utility Only)

AFTK is a **utility token** intended exclusively for use within the AntiFake ecosystem. Its functions include accessing authenticity verification services, generating QR Proof certificates, participating in app features, and supporting on-chain operational workflows. AFTK **is not designed for investment**, speculation, or financial gain.

12.2 No Financial, Investment, or Security Characteristics

AFTK **does not** represent or provide:

- equity or ownership in AntiFake or any related entity,
- voting or governance rights,
- dividends, revenue sharing, profit participation,
- expectations of appreciation driven by third-party efforts,
- any characteristics of securities or regulated financial instruments.

Nothing in this document constitutes investment advice, financial promotion, or an offer to buy or sell securities or financial products.

12.3 No Guarantees or Forward-Looking Commitments

The contents of this whitepaper reflect the project status and roadmap as of **October 2025**.
All features, timelines, and implementations may evolve as part of ongoing development, technical improvements, platform scaling, security audits, and regulatory considerations.
The AntiFake team provides **no guarantee** of:

- future performance,
- token price behaviour,
- long-term market conditions,
- delivery of features exactly as described.

12.4 Regulatory and Jurisdictional Considerations

The regulatory status of utility tokens varies across jurisdictions and may evolve over time.
Users and partners are responsible for ensuring compliance with:

- local digital asset regulations,
- tax obligations,
- consumer protection laws,
- and any additional industry-specific requirements.

AntiFake does not provide legal, financial, or tax advice. Individuals should consult qualified professionals if uncertain about their obligations.

12.5 User Responsibility & Risk Acknowledgement

By interacting with the AntiFake platform or the AFTK token, users acknowledge that blockchain systems inherently involve certain risks, including but not limited to:

- smart contract vulnerabilities,
- network congestion or downtime,
- cybersecurity threats,
- regulatory changes,
- wallet mismanagement or loss of private keys,
- volatility and technical failures of public blockchain networks.

Users participate voluntarily and at their own discretion. They remain fully responsible for securing their devices, wallets, and access credentials.

12. Legal Disclaimer

12.1 Nature of the Token (Utility Only)

AFTK is a **utility token** intended exclusively for use within the AntiFake ecosystem. Its functions include accessing authenticity verification services, generating QR Proof certificates, participating in app features, and supporting on-chain operational workflows. AFTK **is not designed for investment**, speculation, or financial gain.

12.2 No Financial, Investment, or Security Characteristics

AFTK **does not** represent or provide:

- equity or ownership in AntiFake or any related entity,
- voting or governance rights,
- dividends, revenue sharing, profit participation,
- expectations of appreciation driven by third-party efforts,
- any characteristics of securities or regulated financial instruments.

Nothing in this document constitutes investment advice, financial promotion, or an offer to buy or sell securities or financial products.

12.3 No Guarantees or Forward-Looking Commitments

The contents of this whitepaper reflect the project status and roadmap as of **October 2025**.
All features, timelines, and implementations may evolve as part of ongoing development, technical improvements, platform scaling, security audits, and regulatory considerations. The AntiFake team provides **no guarantee** of:

- future performance,
- token price behaviour,
- long-term market conditions,
- delivery of features exactly as described.

12.4 Regulatory and Jurisdictional Considerations

The regulatory status of utility tokens varies across jurisdictions and may evolve over time.
Users and partners are responsible for ensuring compliance with:

- local digital asset regulations,
- tax obligations,
- consumer protection laws,
- and any additional industry-specific requirements.

AntiFake does not provide legal, financial, or tax advice. Individuals should consult qualified professionals if uncertain about their obligations.

12.5 User Responsibility & Risk Acknowledgement

By interacting with the AntiFake platform or the AFTK token, users acknowledge that blockchain systems inherently involve certain risks, including but not limited to:

- smart contract vulnerabilities,
- network congestion or downtime,
- cybersecurity threats,
- regulatory changes,
- wallet mismanagement or loss of private keys,
- volatility and technical failures of public blockchain networks.

Users participate voluntarily and at their own discretion. They remain fully responsible for securing their devices, wallets, and access credentials.

12. Legal Disclaimer

12.6 Legal Framework Overview

The AntiFake platform and the AFTK utility token are designed in accordance with widely accepted global regulatory principles for digital utility tokens. The purpose of this framework is to clearly define the boundaries, assumptions, and compliance posture under which the ecosystem operates. The platform maintains a strict separation between **utility access rights** and **financial or investment characteristics**, ensuring that AFTK remains outside classifications applicable to securities, e-money, or investment-oriented digital assets.

12.6.1 Utility Token Classification

AFTK is structured to qualify as a *pure utility token* under major regulatory frameworks where applicable. This includes, but is not limited to:

- **EU MiCA (Markets in Crypto-Assets Regulation)** — defining utility tokens as crypto-assets granting digital access to a service without financial rights.
- **FINMA Token Classification (Switzerland)** — where utility tokens provide access to a digital application but are not intended as investments.
- **U.S. SEC "Investment Contract" Framework** — AFTK is deliberately engineered to avoid characteristics associated with securities under the Howey Test (e.g., expectation of profit, reliance on managerial efforts, speculative intent).

AFTK's sole purpose is to enable access to authenticity-verification functionality within the AntiFake ecosystem. It is not engineered, marketed, or structured to function as a speculative or financial instrument.

12.6.2 No Financial Incentives or Return Expectations

The AntiFake ecosystem is explicitly not designed to produce financial gains for users or token holders. AFTK does not entitle holders to:

- equity, shares, or ownership in AntiFake or affiliated entities,
- governance rights or participation in corporate decisions,
- dividends, profit-sharing, yield, or other income,
- any expectation of token appreciation driven by project performance or third-party efforts.

Token burning mechanisms serve strictly **operational** and **supply-management** purposes tied to authenticity-verification actions. They do **not** create any form of passive return or financial entitlement for users.
These principles ensure that AFTK avoids classification as a security or investment product across global regulatory regimes.

12.6.3 Custodial vs. Self-Custodial Mode

To ensure compliance and accessibility, the platform supports two usage modes:

- **Custodial Credits:**
  Internal in-app credits behave as platform service credits. They are not crypto-assets, cannot be transferred, and hold no redeemable monetary value.
- **Self-Custodial Mode:**
  Advanced users may optionally connect their personal wallet to hold actual AFTK directly. AntiFake, however, does not provide brokerage, exchange, trading, or custodial services for user-owned crypto assets.

This hybrid approach minimizes regulatory exposure while ensuring broad user accessibility without requiring prior knowledge of crypto.

12.6.4 No Token Sale, No Fundraising, No Investment Rounds

The AntiFake project does not engage in any form of token-based fundraising. Specifically, the project does not conduct:

- ICOs, IDOs, TGE-based fundraising,
- public or private token sales,
- investment rounds involving token swaps,
- STOs,
- equity-token crowdfunding schemes.

AFTK tokens are allocated strictly for ecosystem functionality, reward programs, operational flows, and platform usage — not for capital raising.

12.6.5 FIAT Payments and Virtual Credits

To ensure regulatory compliance and ease of onboarding:

- All purchases inside the AntiFake app are processed in **FIAT** via licensed and regulated payment providers.
- Purchased credits represent **virtual service credits** solely for accessing verification features.
- They are not cryptocurrency, not transferable, not redeemable, and cannot be withdrawn.

This FIAT-first structure ensures compliance with consumer protection rules and app-store financial guidelines.

12.6.6 Compliance with App Store & Google Play Rules

The AntiFake app is fully aligned with platform requirements for crypto-related features:

Apple App Store (Guideline 3.1.5)

- No trading, swapping, or exchanging of cryptocurrencies inside the app.
- No unlocking app features by sending tokens externally.
- All purchases made through **Apple In-App Purchases (IAP)** exclusively.
- AFTK is clearly presented as a *utility mechanism* rather than a financial asset.

Google Play Store Cryptocurrency Policies

- No mining, no trading, and no unregulated financial operations.
- Crypto-related interactions are entirely *off-chain* and user-initiated.
- Wallet linking is optional and used only for **ownership attribution**, not payments.

Full adherence ensures smooth app approval and long-term compliance.

12.6.7 Evolving Regulatory Landscape

As global crypto regulation continues to develop, AntiFake maintains a proactive compliance strategy. The project is committed to:

- monitoring relevant legal frameworks across the EU, US, UK, and international jurisdictions,
- integrating ESMA and EBA technical standards under MiCA (RTS/ITS) as they become applicable,
- conducting periodic legal reviews with certified compliance advisors,
- updating platform features or token mechanics when required by law,
- maintaining transparent documentation for exchanges, partners, regulators, and users.

This ensures long-term regulatory stability while preserving a user-friendly ecosystem.

12.7 MiCA Compliance Statement (EU Regulation 2023/1114)

The AFTK utility token and AntiFake platform are designed with explicit reference to the Markets in Crypto-Assets Regulation (MiCA), fully applicable across the EU between 2024–2025. The intention is **alignment**, not classification as a regulated financial product.

12.7.1 Token Classification Under MiCA

MiCA identifies three regulated crypto-asset categories:

1. Asset-Referenced Tokens (ARTs)
2. E-Money Tokens (EMTs)
3. Utility Tokens

AFTK falls strictly into the **Utility Token** category. It:

- provides digital access to a service (authenticity verification),
- does not reference external assets (no backing, no pegs),
- does not function as money or payment method,
- does not store value or provide financial rights.

Thus AFTK is **not** an ART or EMT, and therefore avoids associated issuer obligations.

12.7.2 No Public Offering Under MiCA Definitions

Since AFTK is not sold to the public as a form of financing or investment, MiCA's requirements for public offerings do not apply. The token is exclusively distributed through ecosystem-driven usage, rewards, and technical utilities.

12.7.3 Whitepaper Requirements Applicability

MiCA mandates a regulated whitepaper only for publicly offered utility tokens. Because AFTK is **not sold**, this requirement is not triggered. Nonetheless, the document voluntarily follows MiCA's transparency guidelines.

12.7.4 Custodial Services & MiCA Compliance

The platform's internal credits:

- are not crypto-assets,
- are not subject to MiCA custodial rules,
- do not grant withdrawal or transfer rights.

Where users connect self-custodial wallets, AntiFake does not engage in activities defined under CASP (Crypto Asset Service Provider) obligations, such as exchange, execution, or custody.

12.7.5 Exchange Listings & MiCA

If AFTK ever appears on a regulated EU exchange, the exchange (not the issuer) will handle admission requirements. AntiFake will cooperate transparently but will not engage in exchange-like services.

12.7.6 Ongoing Monitoring

Because regulatory frameworks evolve, AntiFake ensures continuous compliance through:

- active monitoring of MiCA implementation guidance and ESMA RTS/ITS standards,
- periodic legal assessments if token utility expands,
- maintenance of internal compliance and documentation processes,
- transparent communication with partners, auditors, and regulators.

The goal is long-term compliance without compromising usability or accessibility for non-crypto users.

13. Conclusion

AntiFake and AFTK together form a practical, live and expanding solution to one of the biggest challenges of the digital era: knowing what is real.

By combining a user-friendly mobile experience, a robust QR Proof protocol, transparent smart contracts, a deflationary token and a sustainable business model, AFTK aims to establish itself as a global standard for digital authenticity.

Authenticity is the new currency.
AFTK proves it.

14. System Architecture Overview

The AntiFake platform is designed with a simple principle in mind: mainstream users should experience a clean, intuitive app, while all of the complexity – blockchain, gas fees, signatures, storage and indexing – runs in the background.

At a high level, the architecture consists of four main layers:

- Mobile apps (Android and iOS)
  The primary interface for creators and everyday users. They handle capture/upload, account management, previewing assets and QR Proofs.

- Backend API
  A secure REST API built on ASP.NET Core, responsible for hashing files, talking to smart contracts, managing user balances, storing metadata, and generating QR proofs.

- Blockchain layer (Polygon)
  The trust anchor of the system. It holds the AFTK token (AFCoin) and the authenticity registry (AFCertRegistry). Only compact hashes and ownership/timestamp data are stored on-chain, keeping the system efficient and privacy-preserving.

- Storage & previews
  Asset thumbnails and QR images are stored in blob storage with efficient caching, while the original files remain under the user's control or in application-specific storage locations.

This separation means that AntiFake can evolve its app and backend quickly, while the core authenticity guarantees remain anchored to an immutable public chain.

15. Backend Services (Human-Friendly View)

Behind the app, several specialized backend services work together to provide a smooth experience without exposing technical details to end users.

Hashing service
Before anything touches the blockchain, the platform computes a cryptographic fingerprint of the content. The HashService takes a file stream, optionally prepends a secret salt value (configured on the server), and then computes a SHA-256 hash. The result is a 32-byte hash returned as a 0x-prefixed hex string. This approach ensures that:

- The same file always produces the same hash.
- The original content never needs to be stored on-chain.
- A small change to the file produces a completely different hash.

Blockchain service
The BlockchainService is a dedicated component that knows how to talk to Polygon nodes safely and efficiently. It is responsible for:

- Registering authenticity records (calling registerFor on AFCertRegistry).
- Checking whether a hash already exists and who owns it.
- Paying authenticity fees in AFTK via the payFee function.
- Sending AFTK from the project's treasury wallet when rewarding users.
- Estimating gas and constructing modern EIP-1559 transactions with capped fees.

All this is done using a private key stored on a secure hardware wallet or in an encrypted vault. Users never see these details – they simply tap "Verify" and the backend takes care of the rest.

QR signing service
The QrSigningService is responsible for turning on-chain facts into portable QR Proofs. It uses a dedicated Ethereum private key stored in a secure secret vault to sign JSON payloads that describe the registered hash. This signature proves that the QR came from the official AntiFake infrastructure and not from a third party.

Preview service
For a smooth visual experience, the AssetPreviewService generates optimized thumbnails and QR images. It resizes and converts images to modern formats such as WebP, stores them in blob storage, and records metadata in the database. This allows the app and web verification pages to load quickly while minimizing bandwidth costs.

16. End-to-End Verification Flow (From User to Blockchain)

From a user's perspective, the AntiFake experience is simple: they open the app, capture or upload a photo, and receive a QR Proof. Under the hood, the following steps happen:

1. Capture or upload
   The user takes a picture or chooses an existing image. The app sends the file (or precomputed hash) to the backend over an encrypted connection.

2. Hashing and validation
The backend computes a SHA-256 hash (with optional salt), validates the format, and checks whether the hash already exists in the authenticity registry. If the hash is already registered to the same owner, the system can return an "already registered" response instead of duplicating the entry.

3. Fee and credits
Instead of asking the user to send tokens manually, AntiFake maintains an internal credit balance for each account. When a user verifies content, the platform debits a small amount of AFTK-equivalent credits from their balance. This internal ledger is kept in sync with on-chain payments made by the platform's treasury wallet via the payFee function on the AFCoin contract.

4. On-chain registration
Once the user has sufficient credits, the backend calls registerFor on AFCertRegistry. It sends the content hash and the designated owner address (either the user's wallet in "self" mode, or a platform custody address that can later transfer ownership). The transaction is built with carefully chosen gas limits and EIP-1559 fee settings to avoid overpaying while ensuring timely confirmation.

5. Database indexing and previews
After the transaction succeeds, the backend records the hash, owner, transaction hash, chain name, gas usage and other details in its database. If a file was included, it may also generate a thumbnail and store it as a preview. This turns the raw blockchain entry into a rich, user-friendly asset card inside the app.

6. QR Proof generation
Finally, the system builds a signed JSON payload describing the hash, issuer and optional chain/transaction details. The payload is base64url-encoded and signed with the QR signing key according to the EIP-191 standard. A QR image is generated from either a verification URL or a compact payload string, and stored as another preview.

In a matter of seconds, the user receives a visual QR Proof that is backed by a fully auditable on-chain record.

17. Public Verification Page

To make verification as accessible as possible, AntiFake provides a public verification page that does not require installing the app or using a crypto wallet.

When a QR Proof is scanned, the user is taken to a dedicated verification URL containing the encoded payload and signature. The verification backend then:

- Decodes and validates the payload structure.
- Verifies the EIP-191 signature against the official QR signing address.
- Normalizes the hash and checks whether it exists on-chain and in the database.

• Attempts to locate any public thumbnail or QR previews for a better visual experience.
• Looks up the associated user profile for social context where available.

The web page displays the result in a clear, non-technical way: whether the signature is valid, whether the hash exists on-chain, who the current owner is, when it was registered, and – where permitted – a preview of the content.

This makes AFTK proofs easily understandable for journalists, buyers, collectors, platforms and ordinary users, without requiring them to know how to read block explorers or interpret raw transaction data.

18. Custodial Credits, Deposits and Wallet Linking

Because AFTK aims to be usable by people who have never held crypto before, the platform supports a hybrid model of custodial credits and self-custody.

Custodial credits
Users can top up verification credits inside the app using traditional payment methods. These credits are tracked in an internal ledger stored in the platform database. When a user verifies content, the system debits their internal balance and uses the project's treasury wallet to pay the corresponding on-chain fee in AFTK. This shields users from gas complexity and removes the need to handle private keys for simple use cases.

Wallet linking
Advanced users who already use wallets such as MetaMask can link their own Polygon address. Once a wallet is linked and verified, the platform can register authenticity records directly in that address' name, or, in the future, enable flows where users pay fees directly in AFTK from their own wallets.

Deposits and scanners
For scenarios where users send AFTK or other supported tokens directly on-chain, the platform includes a deposit scanner component. This worker periodically monitors the blockchain for transfers into known deposit addresses, records them in the OnChainTransfers
table, and credits users' internal balances after sufficient confirmations. There are manual and automated "one scan" and "backfill" operations that allow the system to recover history or rescan a specific address if needed.

This hybrid strategy means users can start with simple FIAT payments and later transition into full Web3 self-custody without losing their authenticity records or history.

19. User Experience and Onboarding

The ultimate goal of AFTK and the AntiFake app is to bridge two worlds: the simplicity of consumer apps and the trust guarantees of blockchain.

Onboarding new users
When a new user installs the app, they can create an account with familiar patterns: email, password or OAuth, depending on local regulations and platform policies. Without ever mentioning wallets or seed phrases, the app provides them with 25 free authenticity credits so they can immediately try out the full verification workflow.

Clear language
Technical concepts are translated into everyday language:

- "Authenticity record" instead of "on-chain hash"
- "Verification credits" instead of "tokens and gas fees"
- "Proof of originality" instead of "immutable ledger entry"

At the same time, advanced users can still see the raw details – contract address, transaction hash, block explorer links – if they wish.

Growth into Web3
As users become more comfortable, the platform can gently introduce optional features: linking an external wallet, receiving AFTK tips, claiming rewards directly on-chain, or participating in governance once such mechanisms are introduced. This progression lowers the barrier to entry while still aligning long-term behaviour with Web3 principles of ownership, transparency and decentralization.

20. Strategic Positioning

AFTK is not competing to be just another speculative token. Its value comes from:

- A live, working app that solves a real problem: "Is this content real?"
- A clear, focused use case: digital authenticity and provenance.
- A hybrid business model that can be profitable in FIAT while still leveraging Web3.
- A deflationary tokenomics design linked directly to genuine usage.
- A robust, extensible architecture ready for integrations with social media, creative tools, marketplaces and platforms that care about authenticity.

As the cost of misinformation and digital forgery continues to rise, the need for a neutral, interoperable authenticity protocol will only grow. AFTK aims to become that neutral layer: simple for users, powerful for developers, and trustworthy for partners.

21. Advanced Users: Wallet Linking & Self-Custody

While AFTK is designed to work smoothly for non-crypto users through custodial credits, it also fully supports advanced Web3 users who already have their own wallets.

Wallet linking flow
Advanced users can link a Polygon-compatible wallet (e.g. MetaMask, hardware wallet) to their AntiFake account using a secure message-signing process:

1. The user requests a “link wallet” challenge from the app.
2. The backend generates a unique random nonce and stores it temporarily.
3. The user signs a human-readable message containing that nonce with their wallet.
4. The backend verifies the signature and confirms that it was produced by the claimed address.
5. If valid and not already linked to another user, the address is attached to the user’s AntiFake account as a linked wallet. One wallet can be marked as “primary”.

This approach ensures that only the true owner of a wallet can link it to an account. No private keys are ever exposed to the server; only signatures are transmitted.

Benefits of linking a wallet

- Authenticity records can be registered directly in the user’s own address, reinforcing self-custody.
- Future features such as direct on-chain tipping or rewards withdrawals can be enabled without changing accounts.
- The platform can distinguish between fully self-custodial users and those using the default custodial model, and adapt the UX accordingly.

22. Internal Credits & Signup Rewards

To keep onboarding simple, AFTK uses an internal credits ledger for each user account.

Internal balance

Behind the scenes, every user has an internal “AFC credits” balance stored in the platform database. This balance is:

- Credited when a user receives a signup bonus, a promotional reward or a deposit.
- Debited when a user performs authenticity verifications or other paid actions.
- Tracked in an append-only ledger for full transparency and auditability.

Signup reward

New users may receive a one-time signup grant in the form of internal credits. These credits allow them to verify their first assets without needing tokens or a wallet.
The grant is controlled by a budget system in the database so that the campaign can be limited in total volume and monitored over time.

Why this matters

- Non-technical users can start using the app immediately.
- The project can run controlled acquisition campaigns without changing the token supply.
- The internal ledger is transactional and consistent, making it suitable for financial reporting and analytics.

23. On-Chain Deposits & Scanner

For users who prefer to fund their account using tokens directly from their own wallet, AntiFake supports on-chain deposits.

Deposit instructions
The app can generate clear deposit instructions that include:

- The chain and token contract address.
- A vault (treasury) address belonging to the platform.
- A reminder that deposits should be made from the user's linked wallet address, not from a centralized exchange.
- An optional EIP-681 payment URI and a human-readable QR code with instructions.

On-chain scanner
A background worker (deposit scanner) periodically:

1. Monitors the blockchain for incoming transfers to the vault address.
2. Matches those transfers to known user wallets based on the sending address.
3. Records the transfer in an OnChainTransfers table.
4. Credits the user's internal balance once sufficient confirmations are observed.

The scanner includes safeguards such as rate limiting, retry logic and backfill tools to rescan specific wallets or time windows if needed. This ensures deposits are processed reliably even in case of temporary node issues or network congestion.

24. Creator Monetisation & Tips with AFTK

A core part of AFTK's vision is that authenticity should not only protect creators, but also help them earn from their work.

Tipping model
When a visitor scans a QR Proof or views a verified asset, they can choose to send a voluntary tip to the creator in AFTK. Over time, this enables creators to build a reputation and a revenue stream based on genuinely verified content.

There are two main ways tips can be handled:

- Custodial tips
  Tips are directed to the platform's treasury or a managed wallet and recorded as credits in the creator's internal balance. The creator can then use these credits for further verifications or, in the future, withdraw them to a self-custodial wallet once a suitable compliance framework is in place.

- Direct on-chain tips (for linked wallets)
  For advanced users who have linked their own wallet, tips can be sent directly on-chain

to their address. In this case, the platform may still track the tip event for analytics and social reputation, but does not control the funds.

Benefits for creators
• New monetisation channel based on authenticity rather than pure reach.
• Stronger relationship with audiences who value verified, original content.
• Incentives to continue using the AntiFake platform and AFTK ecosystem.

Appendix A — Diagrams & Flows

The following diagrams are conceptual and can be visualised in slide decks, on the website or in investor presentations.

Diagram 1 — High-Level Architecture
[AntiFake Mobile App] → [AntiFake API] → [Polygon Blockchain]
• The app captures content and sends it to the API.
• The API hashes and registers content, and generates QR proofs.
• The blockchain stores authenticity records and token balances.

Diagram 2 — Verification Flow
User takes photo → App uploads → API hashes file →
API checks internal credits → BlockchainService calls registerFor →
Transaction confirmed on Polygon → Database saves record & previews →
QR payload signed → QR image generated → User receives QR Proof.

Diagram 3 — Wallet Linking & Self-Custody
User opens “Link Wallet” → App requests challenge → Backend generates nonce →
User signs message in wallet → Backend verifies signature & address →
Wallet stored as linked and optionally primary → Future actions (deposits, tips, ownership) can target this address.

Diagram 4 — Tipping & Creator Rewards
Viewer scans QR → Verification page shows creator profile and asset →
Viewer clicks “Tip Creator” → Chooses amount →
Either: AFTK transfer is sent on-chain to creator’s linked wallet,
or: credits are added to creator’s internal balance via the treasury and logged for analytics and future withdrawals.

These flows demonstrate how AFTK combines a simple user experience with a robust technical
foundation, enabling both casual users and advanced Web3 participants to benefit from the same authenticity infrastructure.