



A.D. 1308  
**unipg**  
DIPARTIMENTO  
DI INGEGNERIA

Tesina Finale di  
**Data Security and Blockchain**

Corso di Laurea Magistrale in Ingegneria Informatica e Robotica A.A. 2024-2025  
DIPARTIMENTO DI INGEGNERIA

docente  
Prof. Luca GRILLI

## **AgriCircularToken**

Web DApp eseguita sulla Testnet Sepolia Ethereum  
Tecnologie: HTML/CSS/JavaScript/Web3.js



Studente

374101 **Valentina Milighetti** valentina.milighetti@studenti.unipg.it

Data ultimo aggiornamento: 16 giugno 2025

# Sommario

1. Descrizione del Problema .....	3
1.1 La circolarità nei sistemi agroalimentari.....	3
1.2 Lo standard ERC-20.....	4
2. Specifica dei Requisiti .....	6
3. La Web Dapp AgriCircularToken .....	8
3.1 Architettura del sistema Software.....	8
3.2 Smart Contract .....	9
3.5 Logica di acquisto dei token ACT .....	12
3.6 Verificabilità.....	12
4. Realizzazione .....	13
5. Conclusioni e Sviluppi Futuri .....	19
6. Bibliografia .....	20

# 1. Descrizione del Problema

L'obiettivo di questo lavoro è lo sviluppo di una Web DApp, denominata AgriCircularToken, finalizzata alla creazione di token fungibili aderenti allo standard ERC- 20 [1] della blockchain Ethereum. La caratteristica principale di AgriCircularToken è quella di permettere agli utenti di acquistare token per aderire ad attività nell'ambito della circolarità dei sistemi agroalimentari, che siano questi dei badge, dei corsi o dei progetti.

Gli utenti possono dichiarare il proprio impegno nella circolarità, specificando la tipologia e l'entità dell'attività effettuata e acquistare i token corrispondenti. I token sono acquistabili spendendo criptovaluta ETH.

La Web DApp AgriCircularToken sarà eseguibile interamente nel browser, senza l'impiego di un server backend, fatto eccezione l'interfacciamento con un endpoint della blockchain. L'applicazione sarà implementata facendo uso della tecnologia HTML/CSS/JavaScript, mentre sarà impiegata la libreria Web3.js [4] per gestire l'interazione con la blockchain. Lo smart contract della DApp sarà rilasciato nella rete di test pubblica Sepolia Ethereum.

## 1.1 La circolarità nei sistemi agroalimentari

L'economia circolare nel settore agroalimentare rappresenta un approccio innovativo per affrontare le sfide ambientali legate alla produzione e al consumo di cibo. Questo modello si basa sui principi delle "3R": riduzione, riutilizzo e riciclo, con l'obiettivo di mantenere il massimo valore dei prodotti, componenti e materiali, trasformando gli scarti in risorse preziose.

Nel 2021 il sistema alimentare europeo ha generato oltre 58 milioni di tonnellate di rifiuti alimentari all'anno, con il 54% proveniente dalle famiglie. Questo spreco è responsabile dell'8-10% delle emissioni globali di gas serra, paragonabili a quelle del trasporto su strada. Per contrastare questo fenomeno, l'Unione Europea ha adottato l'Obiettivo di Sviluppo Sostenibile 12.3, che mira a dimezzare lo spreco alimentare pro capite entro il 2030.

La circolarità può essere applicata in tutte le fasi della filiera agroalimentare: dalla produzione alla trasformazione, fino al consumo e alla gestione degli scarti. Ad esempio, gli scarti alimentari possono essere utilizzati per produrre bioplastiche, fertilizzanti o energia rinnovabile. L'adozione di attività agricole sostenibili e l'ottimizzazione del packaging contribuiscono a ridurre l'impatto ambientale del settore.

Per promuovere la circolarità, è fondamentale anche la diffusione di buone pratiche e la collaborazione tra diversi attori della filiera.

In sintesi, l'economia circolare offre un'opportunità concreta per rendere la filiera agroalimentare più sostenibile, contribuendo alla riduzione degli sprechi, alla valorizzazione delle risorse e alla riduzione degli impatti ambientali.

Nel contesto della transizione verso sistemi agroalimentari più circolari, la blockchain può giocare un ruolo chiave nel valorizzare e incentivare i comportamenti virtuosi. Infatti, attraverso un sistema distribuito e verificabile, è possibile:

- tracciare l'impegno concreto nella circolarità, come il recupero degli scarti, l'adozione di packaging sostenibili o la redistribuzione delle eccedenze.
- certificare pubblicamente queste azioni, offrendo visibilità e credibilità a chi adotta pratiche sostenibili.
- incentivare le buone pratiche mediante ricompense, crediti o riconoscimenti digitali assegnati a chi dimostra comportamenti coerenti con i principi dell'economia circolare.
- coinvolgere la comunità, che può accedere e verificare i dati in tempo reale, rafforzando il meccanismo di fiducia e controllo diffuso.

Questa combinazione di trasparenza, tracciabilità e incentivazione può accelerare l'adozione della circolarità nella filiera agroalimentare, rendendo misurabili gli sforzi dei singoli attori e premiando chi contribuisce attivamente alla sostenibilità del sistema.

## 1.2 Lo standard ERC-20

Lo standard ERC-20 (Ethereum Request for Comments) è utilizzato per i token fungibili sulla blockchain Ethereum e definisce un insieme di specifiche ed eventi che un token deve seguire per essere compatibile con ERC-20.

L'utilizzo dello standard ERC-20 garantisce l'uniformità sulla creazione e il trasferimento dei token su Ethereum, rendendoli più sicuri, trasparenti e interoperabili con wallet e altri sistemi, senza bisogno di adattamenti specifici.

Le funzioni principali definite dallo standard ERC-20 sono:

- `totalSupply()`: restituisce il numero totale di token esistenti.
- `balanceOf(address account)`: restituisce il saldo di token dell'indirizzo *account*.
- `transfer(address to, uint256 value)`: trasferisce una quantità *value* di token dal conto del chiamante a quello del destinatario *to*. Restituisce un valore booleano che indica se l'operazione è riuscita.
- `allowance(address owner, address spender)`: restituisce il numero rimanente di token che *spender* sarà autorizzato a spendere per conto di *owner* tramite `transferFrom()`.
- `approve(address spender, uint256 amount)`: autorizza *spender* ad utilizzare una quantità *value* di token dal saldo di chi effettua la transazione.
- `transferFrom(address sender, address recipient, uint256 amount)`: trasferisce token da un indirizzo a un altro, utilizzando il meccanismo di approvazione.

Gli smart contract emettono eventi sulla blockchain che permettono di notificare azioni o cambiamenti di stato.

Lo standard ERC-20 individua due eventi principali che servono a tracciare i trasferimenti e le approvazioni sui token:

- `Transfer(address indexed from, address indexed to, uint256 value)`: emesso quando i token vengono trasferiti da un account all'altro
- `Approval(address indexed owner, address indexed spender, uint256 value)`: emesso quando un account *owner* autorizza un altro account *spender* a spendere un certo valore di token per suo conto

## 2. Specifica dei Requisiti

L'applicazione AgriCircularToken dovrà soddisfare i seguenti requisiti:

1. Connessione al wallet Metamask
2. Deploy dello Smart Contract sulla rete Sepolia
  - Il contratto definisce i token ERC-20 "ACT" specifici dell'applicazione
3. Registrazione di attività circolari attraverso un apposito form
4. Calcolo dei token ERC-20 equivalenti alle attività registrate
5. Acquisto di token ERC-20:
  - mint dei token equivalenti a un'attività registrata
6. Acquisto di badge che certificano l'impegno nella circolarità:
  - sono suddivisi per livello crescente
  - un utente può acquistare un badge di livello  $i$  utilizzando token ERC-20 solo se possiede un badge di livello  $i - 1$
  - i badge possono essere creati solo dal proprietario dello smart contract, che specifica il nome del badge, la descrizione, il livello e il numero di token necessari per riscattarlo
  - i badge disponibili devono essere visualizzabili all'interno di un'area dedicata
7. Acquisto di corsi informativi sulla circolarità nei sistemi agroalimentari:
  - un utente può acquistare un corso solo se non l'ha già acquistato in precedenza
  - i token corrispondenti a un corso vengono trasferiti dall'utente al proprietario del corso
  - i corsi acquistabili devono essere visualizzabili all'interno di un'area dedicata
8. Creazione di corsi informativi, specificando:
  - Nome
  - Descrizione
  - Promotore (società o ente)
  - Costo del corso in token
9. Contribuzione a progetti ecologici:
  - l'utente specifica il numero di token che vuole trasferire al proprietario del progetto
  - i progetti attivi devono essere visualizzabili all'interno di un'area dedicata

- i token vengono trasferiti al proprietario del progetto quando il numero target dei token viene raggiunto

#### 10. Creazione di progetti, specificando:

- nome
- descrizione
- posizione geografica
- numero di token da raccogliere

#### 11. Area personale in cui poter visualizzare:

- saldo dei token ERC-20 posseduti dall'utente
- totale dei token acquistati
- badge posseduti
- corsi acquistati
- progetti contribuiti
- lista delle transazioni con link alla piattaforma etherscan.io

#### 12. Architettura dell'applicazione frontend-only

#### 13. Interfaccia utente semplice e intuitiva

- Deve mostrare dei messaggi di notifica quando si effettuano le operazioni principali, oltre ad avvisi di eventuali errori

## 3. La Web Dapp AgriCircularToken

AgriCircularToken è una Web Dapp incentrata sull'acquisto di token ERC-20 con simbolo ACT, con la possibilità di scambiarli per ottenere dei badge, acquistare dei corsi informativi o contribuire a dei progetti inerenti alla circolarità nella filiera agroalimentare. L'obiettivo principale è quello di incentivare gli utenti ad adottare delle pratiche sostenibili, in quanto la piattaforma può rappresentare una vetrina che contribuisce alla loro reputazione, mostrando il loro impegno in maniera verificabile e trasparente.

La realizzazione della Dapp prevede il deploy dello smart contract sulla testnet Sepolia e l'implementazione dell'interfaccia web che comunica con la blockchain.

Le operazioni ammesse dall'applicazione sono semplici da realizzare da parte dell'utente, in quanto sono previsti dei semplici form html o dei bottoni per inviare le transazioni. Inoltre, la comunicazione tra l'interfaccia web e lo smart contract è svolta con l'impiego dell'estensione MetaMask.

Di seguito è descritta la struttura dell'applicazione, dall'architettura del software allo smart contract.

### 3.1 Architettura del sistema Software

L'architettura del sistema è progettata per essere frontend-only, per cui tutte le operazioni vengono eseguite direttamente nel browser web dell'utente, tranne l'interazione con l'endpoint della blockchain. L'interfaccia utente comunica con la testnet Sepolia tramite l'endpoint dedicato, utilizzando la libreria Web3.js che offre un'interazione semplice ed efficiente con la blockchain.

L'applicazione è accessibile dalla pagina html *index.html*, che contiene tutte le componenti per effettuare le operazioni previste, lo script per il caricamento dello smart contract e il collegamento del wallet. Il file *style.css* gestisce tutta la formattazione delle componenti html. Le funzionalità del software sono gestite dai seguenti file JavaScript:

- ***dashboard.js***: gestisce il caricamento dell'area personale dell'utente. Viene chiamata la funzione `updateDashboard()` a ogni aggiornamento delle operazioni completate e carica le informazioni personali, tra cui il saldo dei token e tutte le componenti acquistate (mostrando nomi di badge, corsi e progetti). Carica anche un'altra sezione contenente le transazioni principali che l'utente ha effettuato, specificandone la tipologia.
- ***buyTokens.js***: permette di comprare token ACT registrando delle attività. L'utente inserisce da un menu l'attività da riscattare e la quantità. A ogni modifica è mostrato a schermo il numero di token ACT equivalenti all'attività registrata, che si potrà riscattare cliccando su un apposito bottone.
- ***badges.js***: se il wallet è collegato, carica tutti i badge disponibili per l'utente e gestisce l'acquisto degli stessi.

Se il wallet collegato è quello del proprietario dello smart contract, permette la creazione di nuovi badge attraverso la compilazione di un modulo.



La funzione `displayBadges()` carica solamente i badge che ancora l'utente non possiede, mostrando a schermo tutte le informazioni.

- ***courses.js***: se il wallet è collegato, carica tutti i corsi disponibili per l'utente e gestisce l'acquisto degli stessi. Permette di creare nuovi corsi
- ***projects.js***: se il wallet è collegato, carica tutti i progetti disponibili per l'utente e gestisce la contribuzione degli stessi, prendendo in input la cifra che l'utente intende devolvere. Permette di creare nuovi progetti

## 3.2 Smart Contract

Lo smart contract *AgriCircularToken.sol* definisce la struttura del token ACT e le operazioni permesse.

Il deploy del contratto è stato realizzato attraverso l'IDE web Remix e il compilatore Sepolia nella sua versione 0.8.20.

Il contratto eredita il comportamento dello standard ERC-20 e di Ownable: quest'ultimo è un modulo contrattuale che fornisce un meccanismo di controllo degli accessi di base, in cui esiste un account (un proprietario) a cui può essere concesso l'accesso esclusivo a funzioni specifiche.

Il proprietario iniziale è rappresentato dall'indirizzo fornito al momento del deploy del contratto, ma può essere successivamente modificato chiamando la funzione `transferOwnership(newOwner)`.

Ownable mette a disposizione il modificatore `onlyOwner`, che può essere applicato alle funzioni per limitarne l'uso al proprietario. Infatti, come riportato in seguito, la funzione che permette di creare i badge è di questo tipo in quanto la creazione di nuovi badge è riservata solo al proprietario del contratto.

*AgriCircularToken.sol* definisce tre strutture di dati, rispettivamente relative a corsi, progetti e badge:

```
struct Course {  
    string name;           // Nome del corso  
    string description;    // Descrizione del corso  
    string promoter;       // Promotore del corso  
    uint256 requiredTokens; // Costo del corso  
    address courseOwner;   // Indirizzo del creatore del corso  
}
```

```

struct EcologicProject {
    string name;                // Nome del progetto
    string description;         // Descrizione del progetto
    string location;            // Posizione del progetto
    uint256 requiredTokens;     // token richiesti
    bool active;                // Stato del progetto
    uint256 totalContributed;   // Totale dei token già contribuiti
    address projectOwner;       // Indirizzo del creatore del progetto
}

```

```

struct CircularityBadge {
    string name;                // Nome del badge
    string description;         // Descrizione del badge
    uint256 level;              // Livello crescente di importanza
    uint256 requiredTokens;     // Costo del badge
}

```

Gli eventi emessi dal contratto sono i seguenti:

- **TokensMinted(address indexed recipient, uint256 amount):** emesso quando un utente registra un'attività e acquista token ACT
- **ActivityRecorded(address indexed user, string activity, uint256 quantity):** emesso quando un utente registra un'attività e acquista token ACT
- **ProjectCreated(uint256 projectId, string name):** emesso dopo la creazione di un progetto
- **ProjectCompensated(uint256 projectId, address indexed user, uint256 tokens):** emesso quando un utente devolve token ACT a un progetto
- **ProjectCompleted(uint256 projectId, address indexed projectOwner, uint256 tokens):** emesso quando un progetto viene completato
- **BadgeCreated(uint256 badgeId, string name):** emesso alla creazione di un badge
- **badgeRedeemed(uint256 badgeId, address indexed user):** emesso quando un utente acquista un badge
- **CourseCreated(uint256 courseId, string name):** emesso alla creazione di un corso
- **courseCompleted(uint256 courseId, address indexed projectOwner, address indexed user):** emesso quando un utente acquista un corso

Seguono le funzioni pubbliche che regolano le operazioni principali.

- **buyTokens(uint256 tokenAmount, string memory activity, uint256 quantity) public payable:** controlla che ci siano ETH necessari per effettuare l'operazione e procede con il mint dei token ACT richiesti. Incrementa la variabile globale che tiene traccia del totale degli ACT creati ed emette gli eventi correlati alla registrazione di attività e il mint dei token.
- **createBadge(string memory name, string memory description, uint256 level, uint256 requiredTokens) public onlyOwner:** la funzione è invocabile solo dall'owner del contratto e crea un nuovo badge, specificando nome, descrizione, livello e token richiesti. Inoltre, incrementa il contatore degli id dei badge e associa un oggetto CircularityBadge all'array dei badge del contratto. Emette l'evento BadgeCreated().
- Le funzioni createProject() e createCourse() sono analoghe a createBadge(), a meno del fatto che possono essere chiamate specificando l'indirizzo di qualsiasi utente.
- **compensateProject(uint256 projectId, uint256 tokenAmount) public:** controlla che l'id del progetto sia valido e che il progetto sia attivo, oltre a verificare che l'indirizzo che chiama la funzione possieda i token che intende spendere. TokenAmount deve anche essere maggiore di zero e non eccedere i token rimanenti del progetto specificato. Successivamente, trasferisce gli ACT dall'utente al proprietario del progetto e aggiorna totalContributed. Se a questo punto totalContributed è uguale a requiredTokens, imposta il progetto come inattivo, trasferisce gli ACT al proprietario del progetto ed emette l'evento ProjectCompleted, altrimenti emette soltanto ProjectCompensated.
- **buyBadge(uint256 badgeId) public:** controlla che l'id passato sia valido, che l'utente possieda abbastanza ACT per riscattare il badge e che non lo abbia già riscattato in precedenza. Verifica poi che l'utente possieda altri badge di livello immediatamente inferiore (se il livello del badge è maggiore di 1), per poi trasferire i token dall'utente al wallet del proprietario del progetto ed emettere l'evento badgeRedeemed.
- **buyCourse(uint256 courseId) public:** controlla che l'id passato sia valido, che l'utente possieda abbastanza ACT per riscattare il corso e che non lo abbia già acquistato in precedenza. Inoltre, verifica che l'indirizzo di chi acquista il corso non sia lo stesso del proprietario del corso, prima di procedere con il trasferimento degli ACT dall'utente al proprietario del progetto e all'emissione dell'evento courseCompleted.

### 3.5 Logica di acquisto dei token ACT

Sono state definite delle attività specifiche che l'utente può registrare per ottenere ACT. La logica di equivalenza tra attività-quantità e token corrispondenti è gestita lato software e non lato Smart Contract, per permettere più flessibilità e prevede le seguenti regole, per ciascuna attività:

- **compostaggio:** 2 ACT per kg
- **utilizzo di energia rinnovabile:** 10 ACT per kWh
- **risparmio idrico:** 3 ACT per L
- **risparmio di plastica nel packaging:** 1.5 ACT per kg

### 3.6 Verificabilità

In linea con il requisito 11 e con la trasparenza della tecnologia blockchain, la Dapp consente di verificare in modo semplice e diretto tutte le transazioni effettuate dall'utente. Queste sono pubblicamente consultabili tramite la piattaforma Etherscan [5], che permette l'esplorazione dei blocchi, delle transazioni e degli smart contract sulla rete Ethereum.

Per ogni interazione dell'utente con lo smart contract, viene generato un hash di transazione univoco. A partire da questo hash, è possibile ottenere un link diretto alla pagina dedicata su Etherscan, dove vengono visualizzati in dettaglio i parametri della transazione, il mittente, il destinatario, l'importo e i dati associati.

Anche lo smart contract AgriCircularToken è completamente tracciabile su Etherscan: dalla sua pagina dedicata è possibile consultare il codice sorgente verificato, gli eventi emessi e le funzioni esposte, come mostrato nella figura seguente.

The screenshot shows the Etherscan interface for the AgriCircularToken (ACT) contract. The top navigation bar includes links for Home, Blockchain, Tokens, NFTs, and More. The main header identifies the token as AgriCircularToken (ACT) and its standard as ERC-20. The overview section provides key statistics: MAX TOTAL SUPPLY (0.000000000000000042 ACT), HOLDERS (2), and TOTAL TRANSFERS (7). The market section shows ONCHAIN MARKET CAP and CIRCULATING SUPPLY MARKET CAP, both currently at zero. The other info section displays the token contract address (0x8b0ada39fc5484c8d186b6d0c69978a5ccfd8ad5). Below these sections, there are tabs for Transfers, Holders, and Contract. The Transfers tab is active, showing a list of transactions. The table includes columns for Transaction Hash, Method, Block, Age, From, To, and Amount. Three transactions are listed, all involving the transfer of ACT tokens between specific addresses.

Transaction Hash	Method	Block	Age	From	To	Amount
0xf94849d476b...	0x2e59a03	8540682	20 mins ago	0x8c27CBc1...9E1761a45	0x8B0AdA39...5ccfD8aD5	0.000000000000000001
0x4488ffde6dd...	0x76e9671d	8540680	20 mins ago	0x00000000...000000000	0x8c27CBc1...9E1761a45	0.0000000000000000022
0x4058cc3367...	0x76e9671d	8540626	31 mins ago	0x00000000...000000000	0x8c27CBc1...9E1761a45	0.0000000000000000002

Figura 1: pagina del token "ACT" all'interno del sito web etherscan.io

## 4. Realizzazione

Per mostrare l'esperienza utente e le logiche implementate, verranno presentati in questo capitolo una serie di immagini che mostrano le principali schermate dell'applicazione ed esempi di utilizzo.

### Pagina iniziale

La pagina iniziale si presenta come segue:



Figura 2: pagina iniziale della web Dapp AgriCircularToken

La navigazione tra le diverse sezioni dell'applicazione è gestita tramite una barra orizzontale, che contiene una serie di pulsanti dedicati alle principali funzionalità offerte dalla DApp. Tra questi è presente un indicatore che mostra se il wallet MetaMask è collegato correttamente oppure no.

### Registrazione di un'attività

Una volta collegato il wallet, è possibile registrare una pratica per ottenere ACT, nella sezione "Record activity", in cui è presente un form che aggiorna dinamicamente il numero di token equivalenti ai dati inseriti. L'utente inserisce la tipologia di attività e ne specifica la quantità, per poi cliccare sul bottone "Record Activity" e autorizzare il trasferimento di ETH dall'estensione MetaMask. Una volta completata l'operazione comparirà un messaggio di notifica o di errore.

The screenshot shows the 'Record Activity' interface. At the top, a green navigation bar contains links for 'Connected', 'Dashboard', 'Record Activity', 'Badge', 'Informative courses', and 'Projects'. The main form area has a title 'Record Activity' with a menu icon. Below it, the 'Activity type:' dropdown is set to 'Water saving'. The 'Quantity(L):' input field contains the number '10'. A green box displays 'Estimated tokens: 30 ACT'. At the bottom, a green button labeled 'Record activity' is visible.

Figura 3: registrazione di un'attività per riscattare dei token ACT

## Badge

Nella sezione “Badge” sono mostrati i badge acquistabili dall’utente e le loro informazioni:

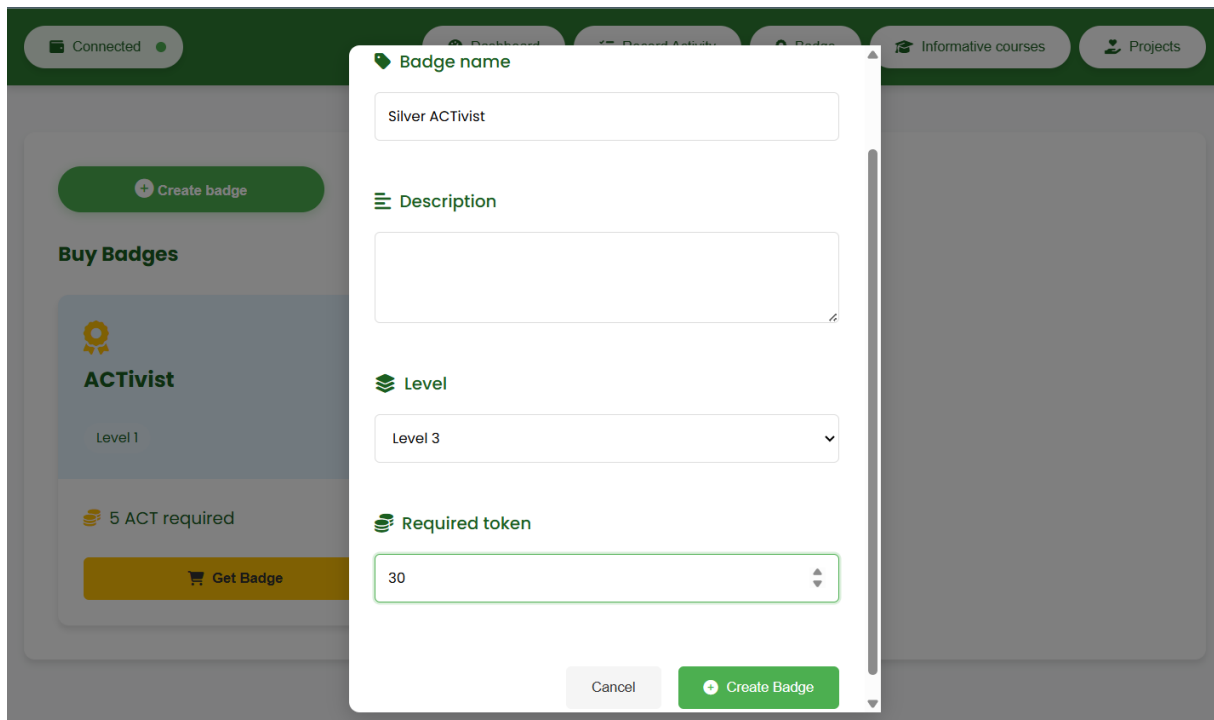
The screenshot shows the 'Buy Badges' section. At the top, a green navigation bar contains links for 'Connected', 'Dashboard', 'Record Activity', 'Badge', 'Informative courses', and 'Projects'. Below the navigation bar, a green button labeled 'Create badge' is visible. The main content area is titled 'Buy Badges' and displays four badge cards. Each card shows a badge icon, the badge name, the level, and the required ACT tokens. A yellow 'Get Badge' button is at the bottom of each card.

Badge Name	Level	ACT Required
ACTivist	Level 1	5 ACT
Bronze ACTivist	Level 2	10 ACT
Silver ACTivist	Level 3	30 ACT
Gold ACTivist	Level 4	60 ACT

Figura 4: negozio dei badge

Un corso è acquistabile, dopo apposite verifiche, tramite il bottone “Get Badge”.

Se l'account collegato è quello del proprietario del contratto, è possibile cliccare sul bottone per creare un nuovo badge, compilando il seguente modulo:



The screenshot shows a modal window titled "Badge name" for creating a new badge. The form contains the following fields and options:

- Badge name:** A text input field containing "Silver ACTivist".
- Description:** A large text area for the badge description.
- Level:** A dropdown menu currently set to "Level 3".
- Required token:** A numeric input field set to "30".
- Buttons:** "Cancel" and "Create Badge" (with a plus icon).

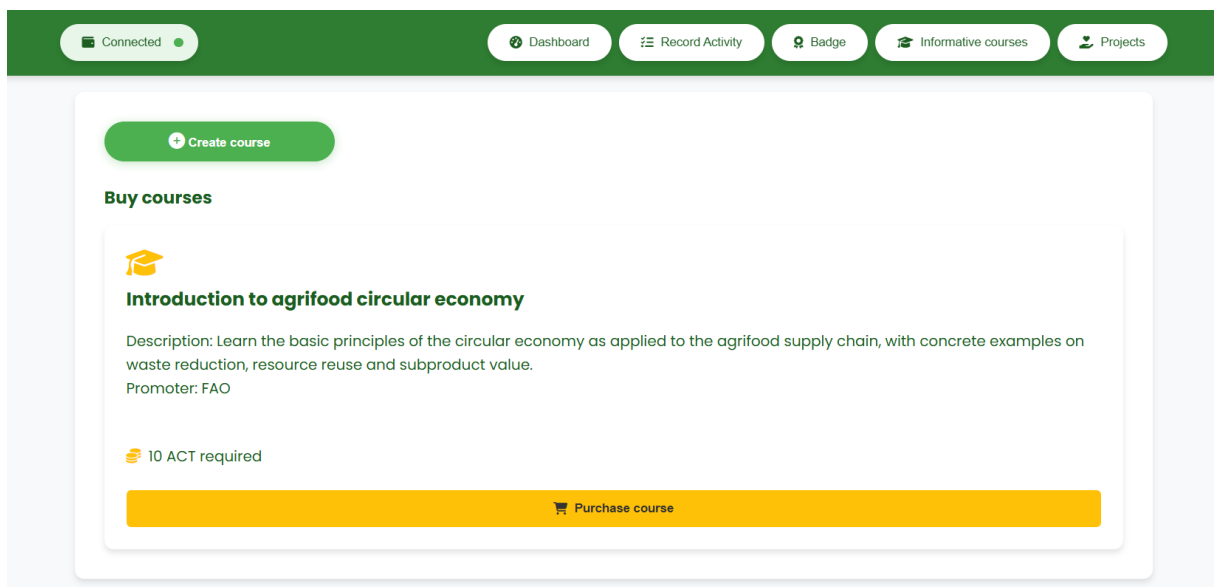
In the background, the "Buy Badges" section is visible, showing an "ACTivist" badge at "Level 1" which requires "5 ACT" and has a "Get Badge" button.

Figura 5: creazione di un nuovo badge

## Corsi informativi

Nella sezione “Informative courses” tutti gli utenti che hanno collegato il wallet possono creare nuovi corsi, con una procedura analoga a quella mostrata in Figura 5.

L'area dei corsi disponibili si popola come segue:



The screenshot displays the "Informative courses" section of the application. At the top, there is a green navigation bar with buttons for "Connected", "Dashboard", "Record Activity", "Badge", "Informative courses" (which is active), and "Projects".

Below the navigation bar, the main content area features a "Create course" button. Under the heading "Buy courses", a course card is shown for "Introduction to agrifood circular economy". The card includes:

- A graduation cap icon.
- The course title: "Introduction to agrifood circular economy".
- Description: "Learn the basic principles of the circular economy as applied to the agrifood supply chain, with concrete examples on waste reduction, resource reuse and subproduct value."
- Promoter: "FAO".
- Requirement: "10 ACT required".
- A yellow "Purchase course" button at the bottom.

Figura 6: sezione dei corsi disponibili per l'acquisto

Un corso è acquistabile, dopo apposite verifiche, tramite il bottone “Purchase course”.

## Progetti

Nella sezione “Projects” tutti gli utenti che hanno collegato il wallet possono creare nuovi progetti, con una procedura analoga a quella in Figura 5.

L’area dei progetti disponibili si popola come segue:

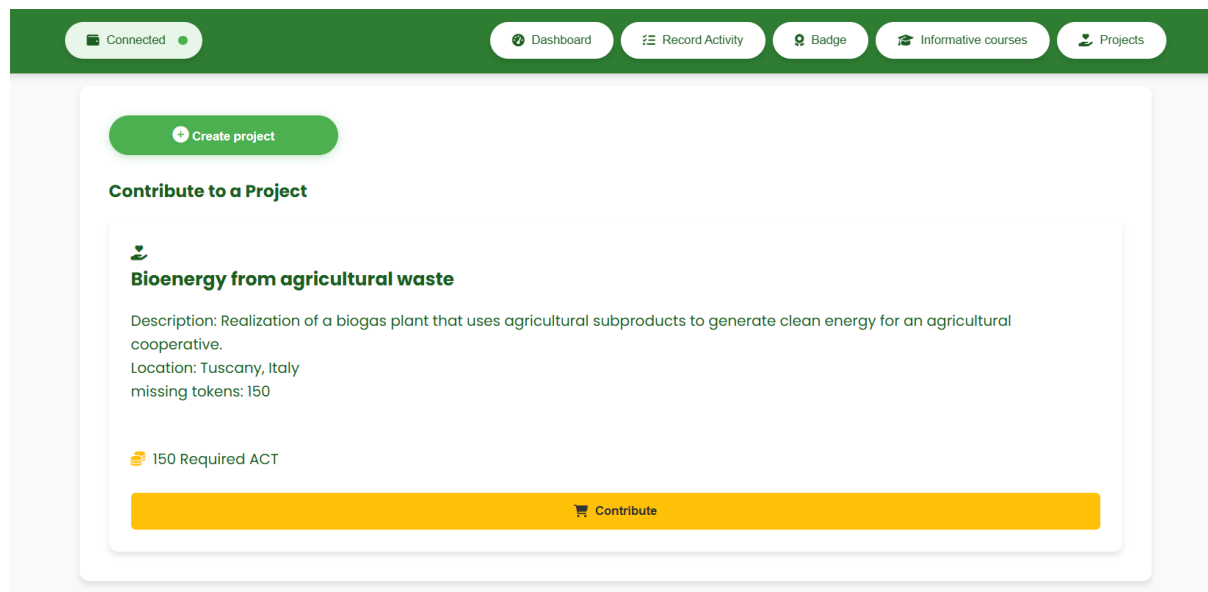


Figura 7: sezione dei progetti attivi a cui un utente può contribuire

Nella scheda di un progetto, oltre che al nome, la descrizione, il luogo e il numero di token da raccogliere, è presente anche il numero di token ACT mancanti al completamento della raccolta.

Una volta cliccato sul bottone “Contribute” è possibile inserire il numero di token da inviare, specificando tra un range che va da 1 ACT al massimo numero possibile, che è il minimo tra il numero di ACT rimanenti e il saldo dell’utente:

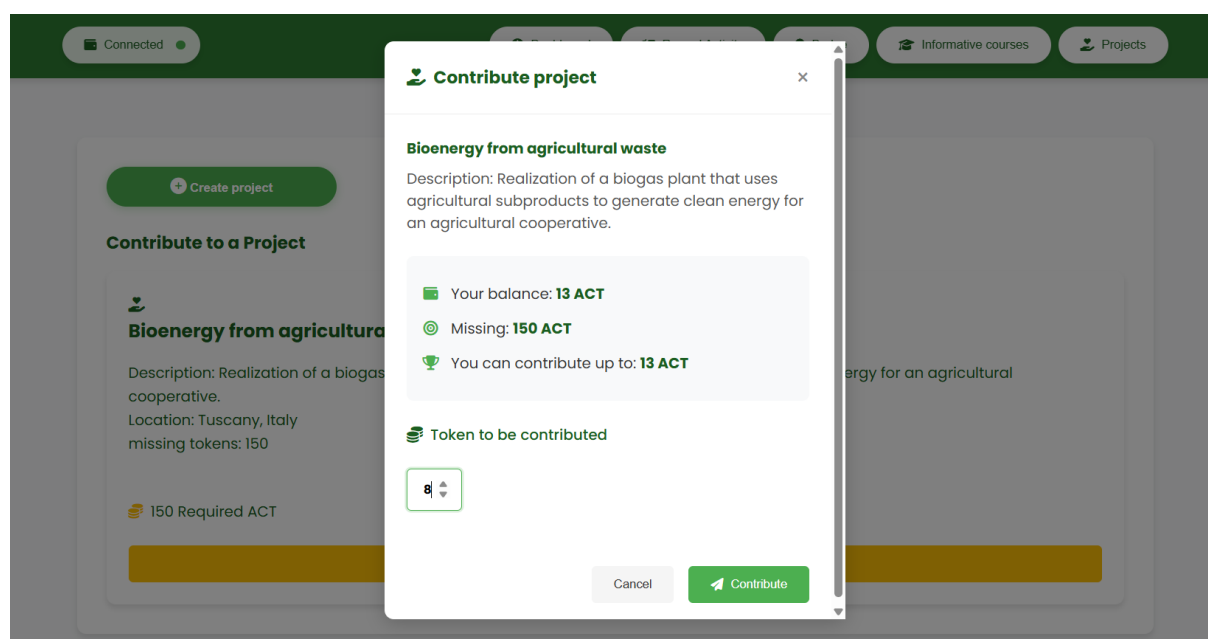


Figura 8: finestra che permette la contribuzione a un progetto



## Dashboard

La sezione “Dashboard”, per un utente che ha collegato il wallet, mostra le sue informazioni:

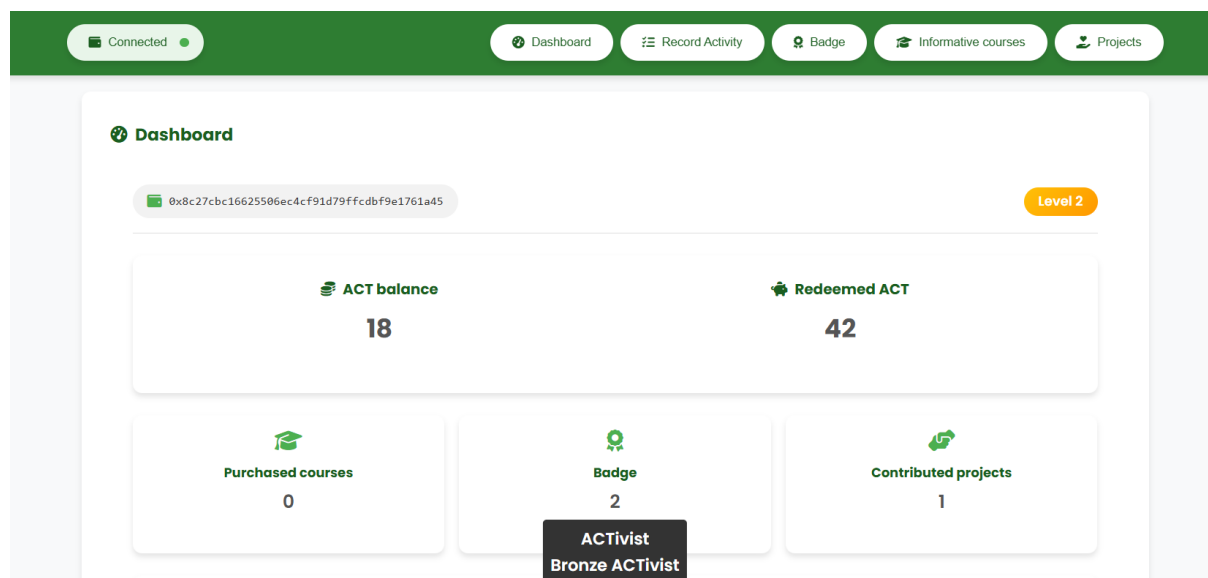


Figura 9: dashboard

Nella precedente figura si possono notare i nomi dei badge in possesso: infatti, passando con il mouse al numero di corsi/badge/progetti interagiti compaiono i nomi coinvolti all'interno di una tooltip.

Inoltre, in alto a destra si mostra il livello dell'utente, che corrisponde al livello massimo dei badge che possiede.

Segue un'altra sezione che mostra le transazioni effettuate:

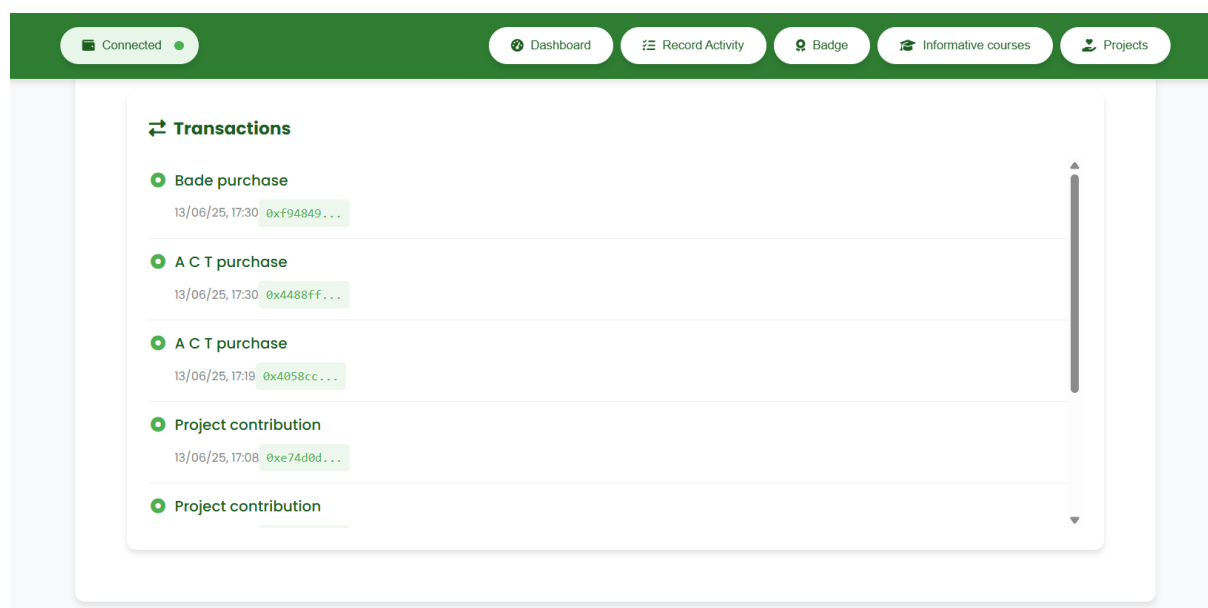


Figura 10: lista delle transazioni effettuate dall'utente. Per ciascuna transazione è specificato il tipo, la data e il link ai dettagli della transazione sul sito etherscan.io

## Gestione degli eventi

L'interfaccia progettata mostra a schermo tutti i messaggi che notificano le avvenute transazioni o gli eventuali errori, come ad esempio quando si cerca di acquistare un articolo senza possedere abbastanza token.

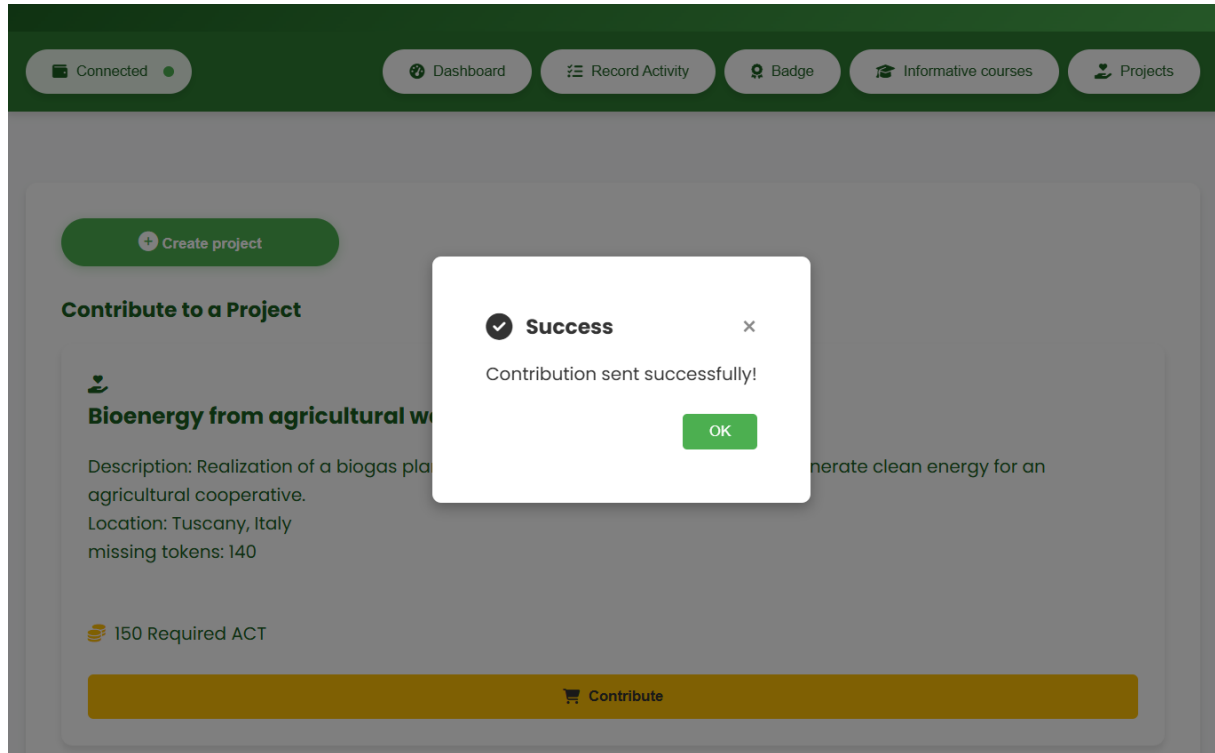


Figura 11: esempio di notifica all'utente, che informa che il contributo a un progetto è stato inviato con successo

## 5. Conclusioni e Sviluppi Futuri

Il progetto AgriCircularToken ha permesso di esplorare le potenzialità offerte dalla tecnologia blockchain nel contesto dell'economia circolare. Attraverso la creazione del token ERC-20 "ACT" con la distribuzione dello smart contract su testnet Sepolia, è stato sviluppato un sistema in cui gli utenti possono essere premiati per comportamenti virtuosi, tracciabili e verificabili, data la presenza di un'area personale e dei collegamenti alle transazioni sul sito etherscan.io. L'applicazione comunica con la blockchain attraverso la libreria Web3.js e il collegamento all'estensione MetaMask per la gestione del wallet e la firma delle transazioni.

L'applicazione creata consente agli utenti di:

- guadagnare token ACT registrando attività sostenibili
- spenderli per ottenere badge
- acquistare corsi informativi
- contribuire direttamente a progetti ecologici con meccanismi trasparenti

Lo smart contract include controlli di proprietà, prevenzione di abusi (ad esempio per gli acquisti ripetuti) e un sistema di tracciabilità attraverso eventi che consentono la costruzione di una dashboard dinamica lato frontend.

Gli sviluppi futuri possibili possono includere l'introduzione di NFT, ovvero token aderenti allo standard ERC-721, per rappresentare i badge in modo visivo e collezionabile.

Inoltre, potrebbe essere utile l'impiego di tecnologie off-chain di archiviazione dati per migliorare la scalabilità della dashboard e l'analisi dei dati storici.

Un ulteriore possibile sviluppo riguarda il miglioramento della logica di assegnazione dei token ACT, in modo da renderla più aderente al valore reale delle attività dichiarate. Sarebbe opportuno introdurre una correlazione più precisa tra tipologia di attività, impatto ambientale e quantità di token generati, eventualmente supportata da un modello di calcolo più complesso o da fonti di verifica esterne.

In definitiva, AgriCircularToken rappresenta un prototipo funzionante che dimostra come la tokenizzazione possa essere impiegata non solo come incentivo, ma anche come strumento di certificazione, trasparenza e partecipazione attiva alla transizione ecologica.

## 6. Bibliografia

- [1] F. Vogelsteller and V. Buterin. ERC-20: Token Standard | eips.ethereum.org, 2015. [Online <https://eips.ethereum.org/EIPS/eip-20>; Consultato il 21-maggio-2025]
- [2] Ethereum Foundation. Remix-ethereum ide, 2025. [Online <https://remix.ethereum.org/>; Consultato il: 5-maggio-2025]
- [3] L. v. Wassenaer, C. Verdouw, A. Kassahun, M. v. Hilten, K. v. d. Meij, B. Tekinerdogan. Tokenizing circularity in agri-food systems: A conceptual framework and exploratory study, 2023. [Online <https://www.sciencedirect.com/science/article/pii/S0959652623016852>; Consultato il 27-aprile-2025]
- [4] ChainSafe Systems. Web3.js — Javascript Ethereum API, 2025. [Online <https://web3js.org/>; Consultato il 3-giugno-2024]
- [5] sepolia.etherscan.io. Testnet Sepolia (ETH) Blockchain Explorer, 2025. [Online <https://sepolia.etherscan.io/>; Consultato il 5-maggio-2025]
- [6] Ente Italiano di Normazione. Circolarità nella filiera agro-alimentare, 2023. [Online <https://www.uni.com/circolarita-nella-filiera-agro-alimentare/>; Consultato il 4-giugno-2025]
- [7] B. Anzà. L'economia circolare nel contesto dei sistemi alimentari, 2024. [Online <https://www.italiaclima.org/leconomia-circolare-nel-contesto-dei-sistemi-alimentari/>; Consultato il 4-giugno-2025]
- [8] PinkUnicorn80. The Power of Giving — The Food Token Project, 2024. [Online <https://medium.com/@PinkUnicorn80/the-power-of-giving-the-food-token-project-26983b34fd87>; Consultato il 4-giugno-2025]