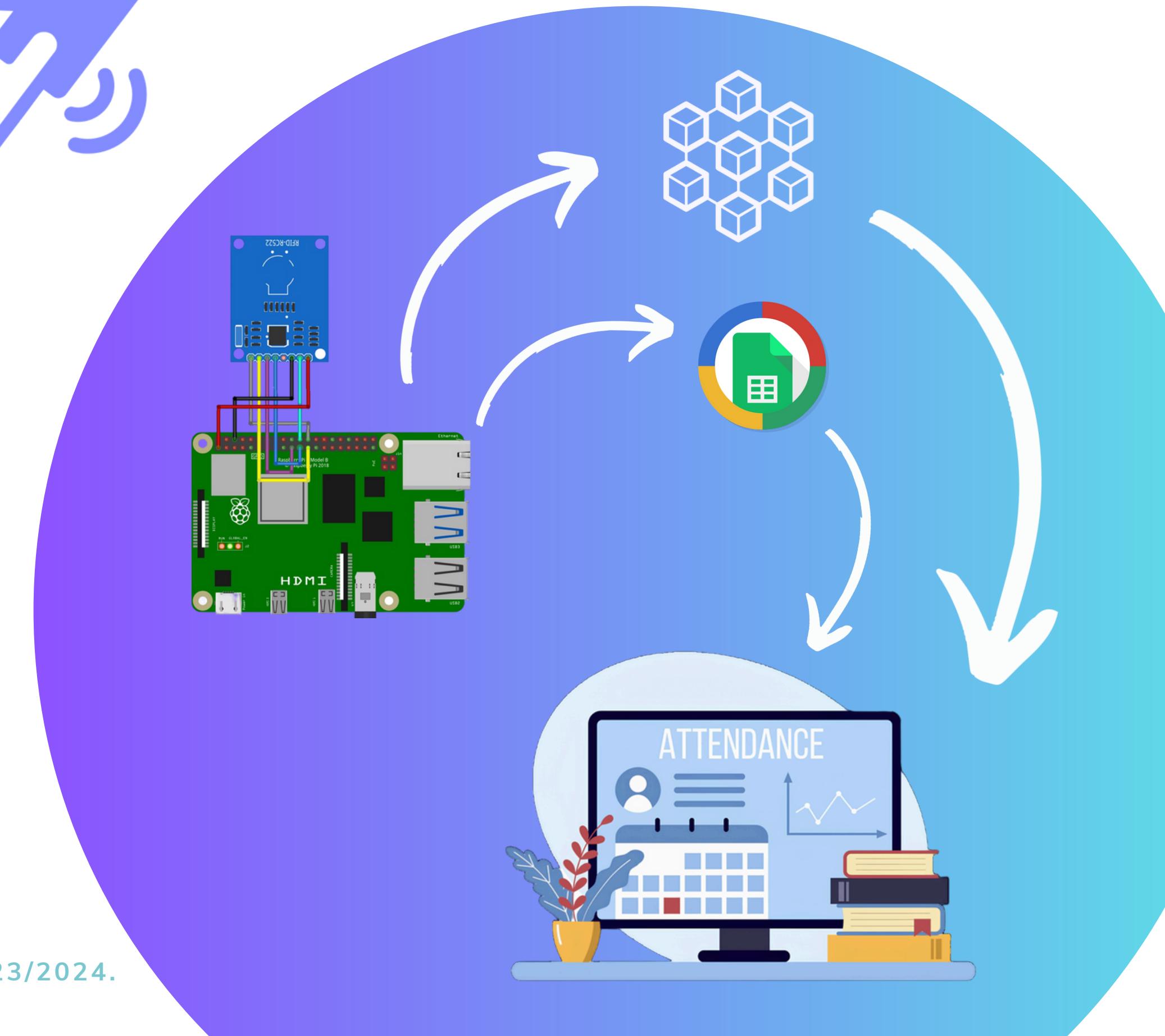


Smart-Tag Attendance System

Sistema automatizzato per la registrazione delle presenze degli studenti a lezioni e durante gli esami, basato su tecnologia RFID e Blockchain.

PROGETTO COMBINATO PER I CORSI DI
SICUREZZA DEI DATI & IOT SECURITY

ANNO ACCADEMICO 2023/2024.





Iniziamo con uno scenario familiare...

Iniziamo con uno scenario familiare...



Iniziamo con uno scenario familiare...



Iniziamo con uno scenario familiare...



Iniziamo con uno scenario familiare...



Iniziamo con uno scenario familiare...

SVANTAGGI DEL SISTEMA PRESENTATO

- Problema di Archiviazione
- Difficoltà di Condivisione
- Fatica di Raccolta e Recupero Dati
- Dispendio in Termini di Tempo
- Inefficienza
- Limitata Tracciabilità
- Mancanza di Backup
- Prone ad Errore Umano
- Rischio di Frode
- Sicurezza Limitata
- ...

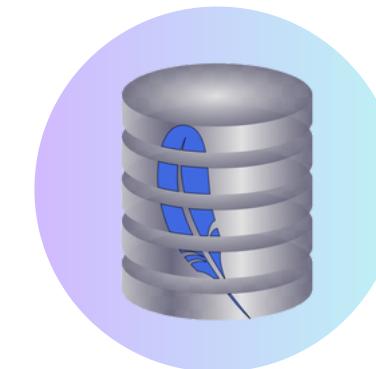
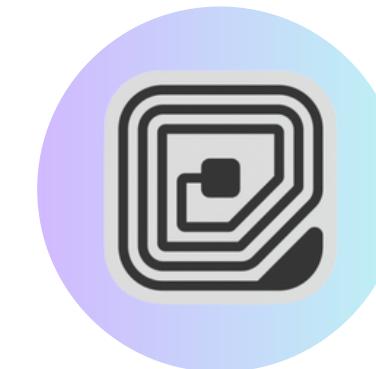
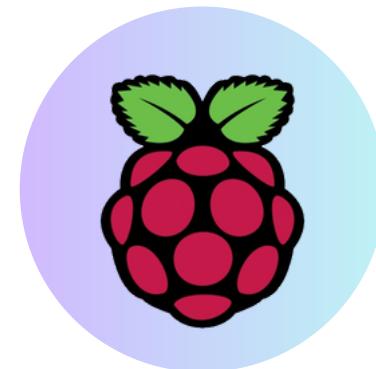
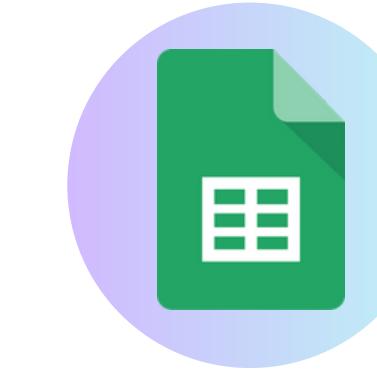
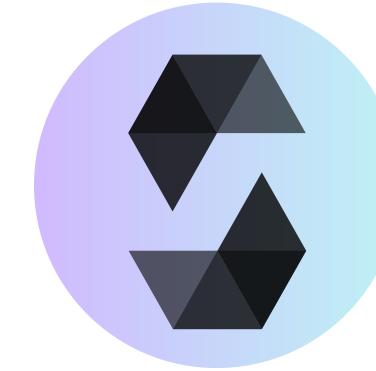
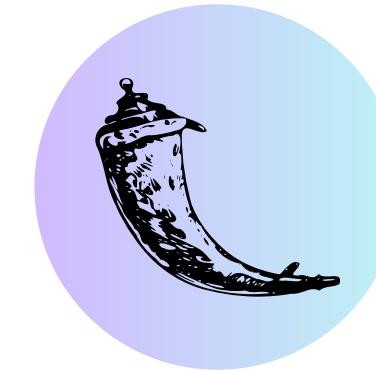
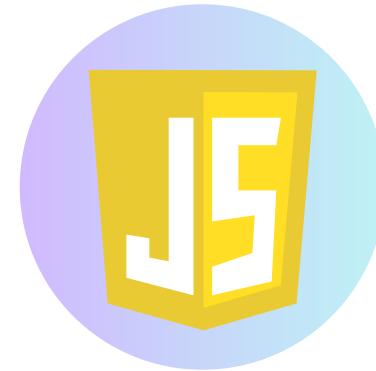




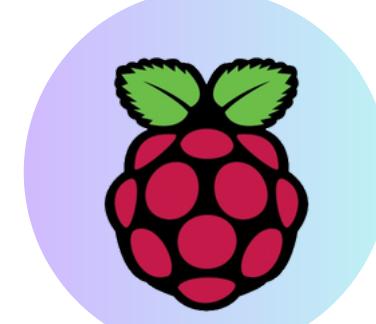
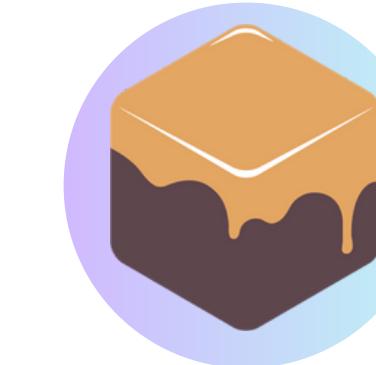
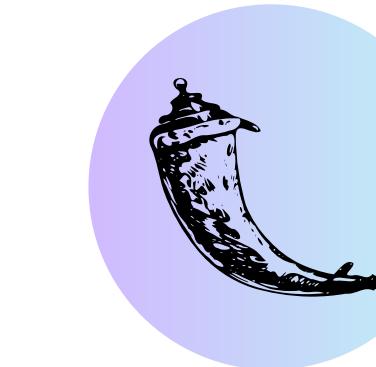
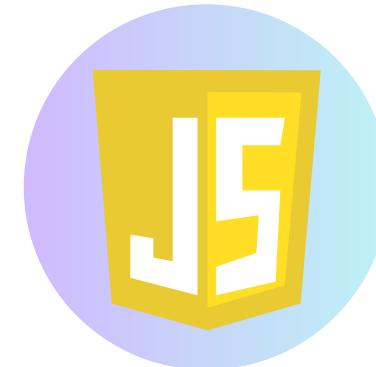
Obiettivo... Superare l'Obsolescenza.

MIGLIORARE L'ACCURATEZZA E L'EFFICIENZA DELLA REGISTRAZIONE DELLE PRESENZE,
SUPERANDO IL METODO CARTACEO ANTICO, GARANTENDO ALLO STESSO TEMPO
LA SICUREZZA DEI DATI E FACILITANDO L'ACCESSO E LA GESTIONE DELLE INFORMAZIONI.

Componenti e Tecnologie Utilizzate



Componenti e Tecnologie Utilizzate



RASPBERRY-PI



RFID



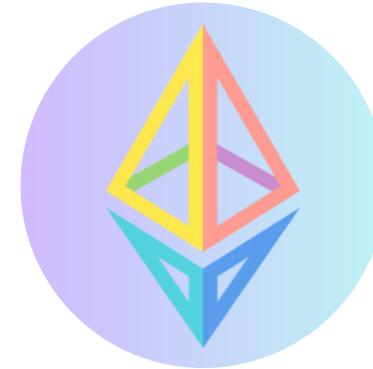
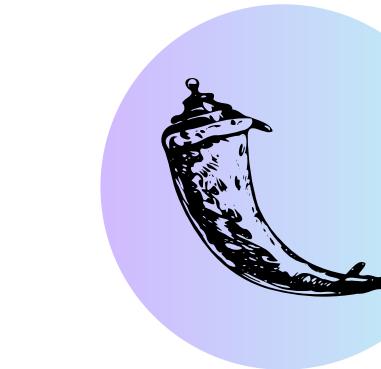
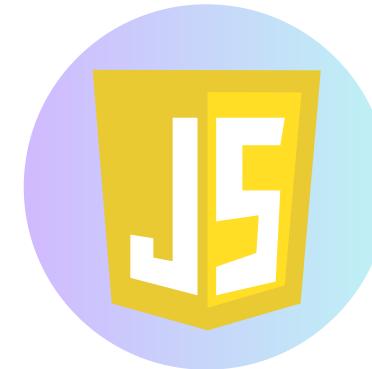
PYTHON



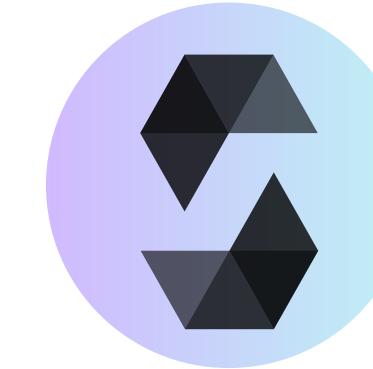
SQLite

Base

Componenti e Tecnologie Utilizzate



ETHEREUM



SOLIDITY



GANACHE



G. DRIVE

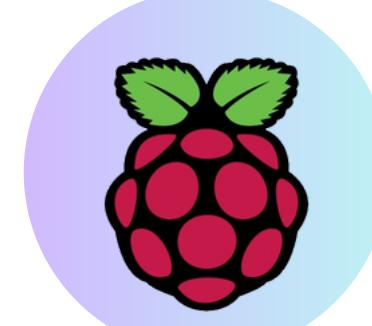


G. CLOUD



G. SHEET

**Gestione
Dati**



RASPBERRY-PI



RFID

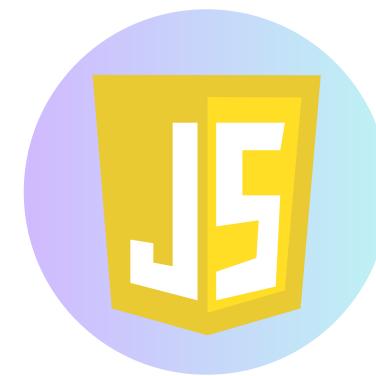


PYTHON



SQLite

Componenti e Tecnologie Utilizzate



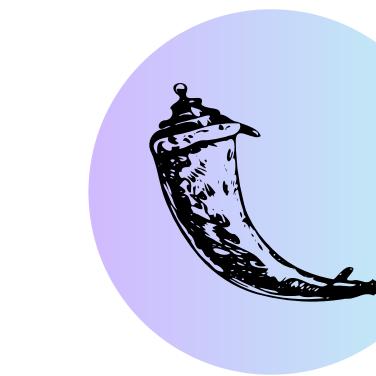
JAVASCRIPT



HTML



CSS

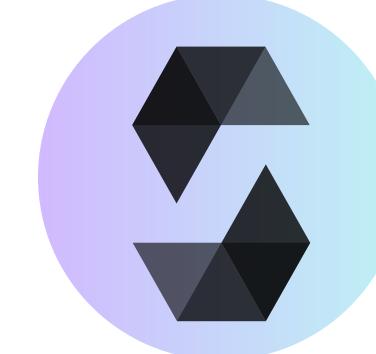


FLASK

Interfaccia di
Monitoraggio



ETHEREUM



SOLIDITY



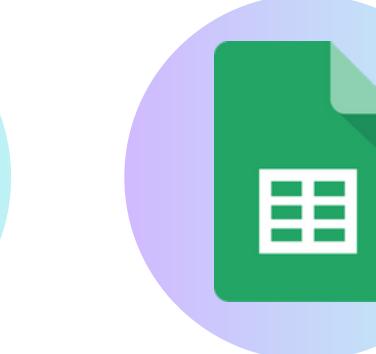
GANACHE



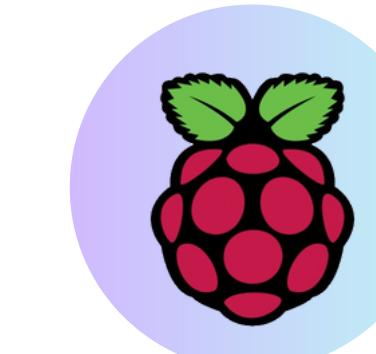
G. DRIVE



G. CLOUD



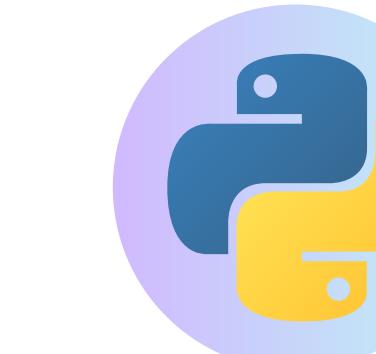
G. SHEET



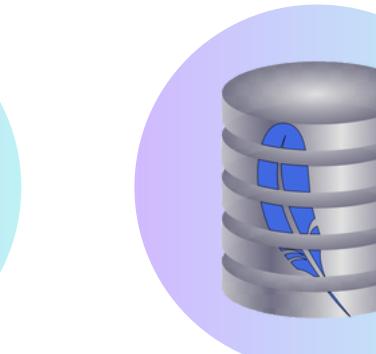
RASPBERRY-PI



RFID

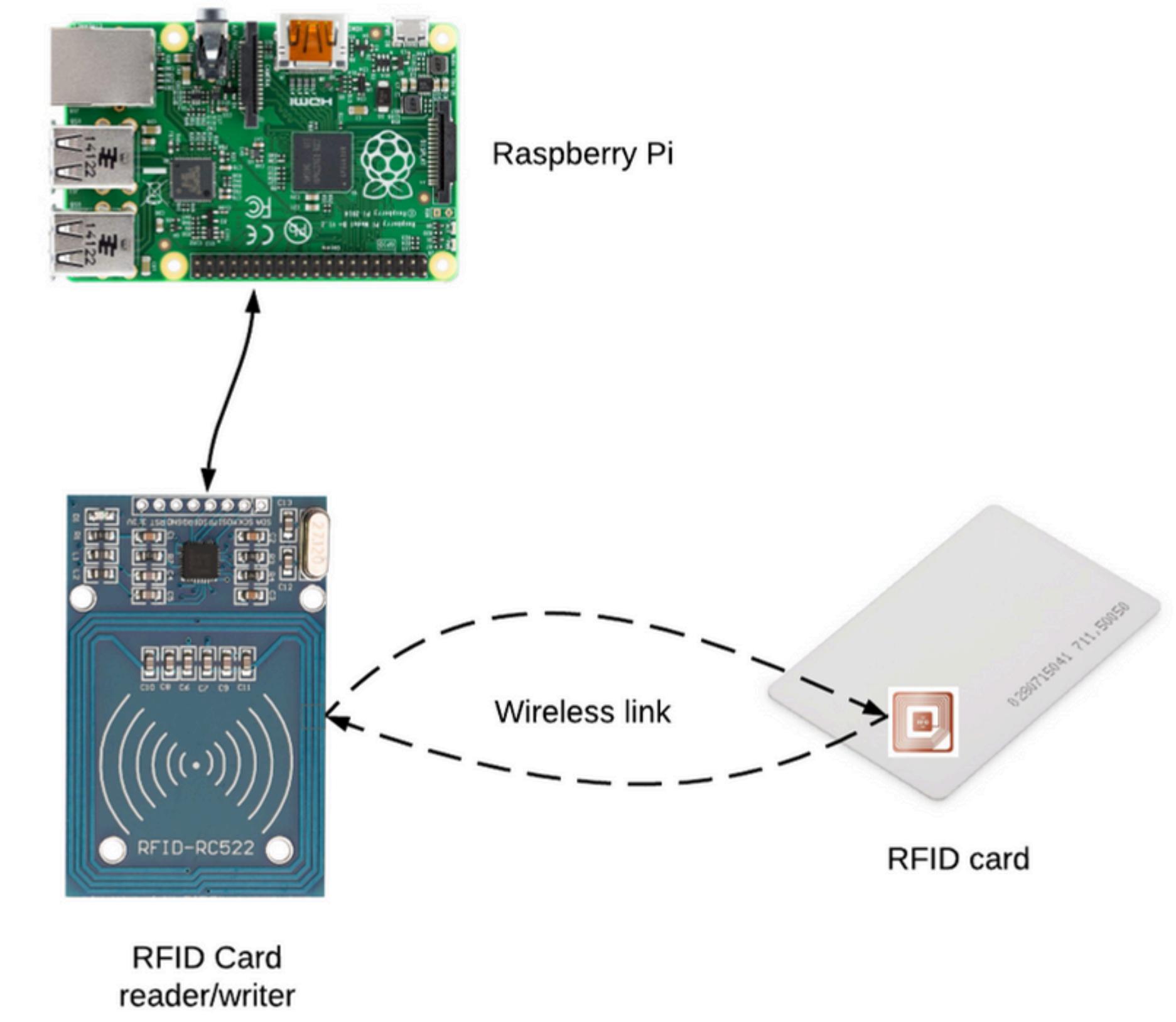


PYTHON



SQLite

Il Cuore del Sistema: Raspberry Pi



Il Cuore del Sistema: Raspberry Pi



Il Cuore del Sistema: Raspberry Pi



Il Cuore del Sistema: Raspberry Pi



1. ISCRIZIONE dello studente al CORSO.

Il Cuore del Sistema: Raspberry Pi



1. ISCRIZIONE dello studente al CORSO.
2. REGISTRAZIONE della presenza a LEZIONE.

Il Cuore del Sistema: Raspberry Pi



1. ISCRIZIONE dello studente al CORSO.
2. REGISTRAZIONE della presenza a LEZIONE.
3. REGISTRAZIONE della presenza all'ESAME.

Il Cuore del Sistema: Raspberry Pi



1. ISCRIZIONE dello studente al CORSO.
2. REGISTRAZIONE della presenza a LEZIONE.
3. REGISTRAZIONE della presenza all'ESAME.

Il Cuore del Sistema: Raspberry Pi



1. ISCRIZIONE dello studente al CORSO.
2. REGISTRAZIONE della presenza a LEZIONE.
3. REGISTRAZIONE della presenza all'ESAME.

Cifratura dei Dati

Cifratura dei Dati

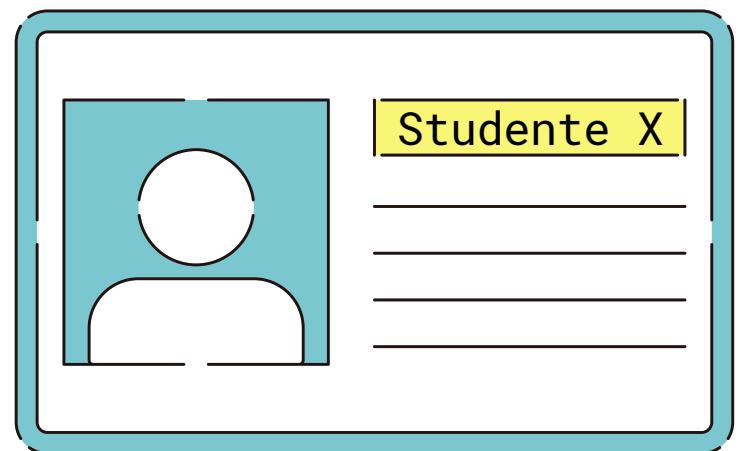


Cifratura dei Dati



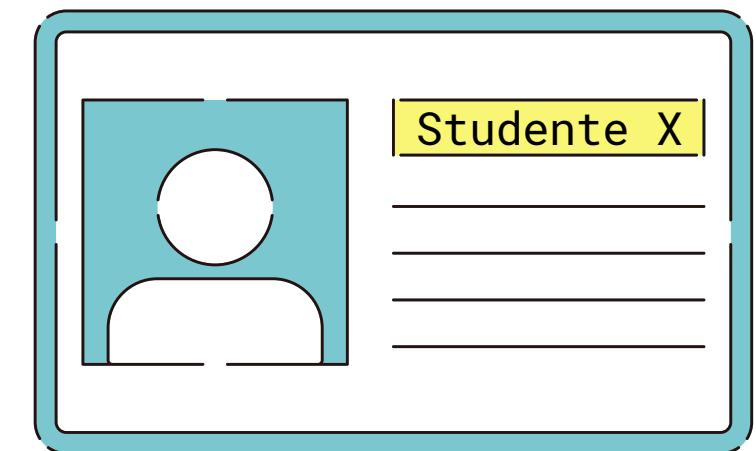
Codice **UNIVOCO** che
permette di risalire al
Profilo Digitale dello
Studente salvato nel
Database.

Cifratura dei Dati

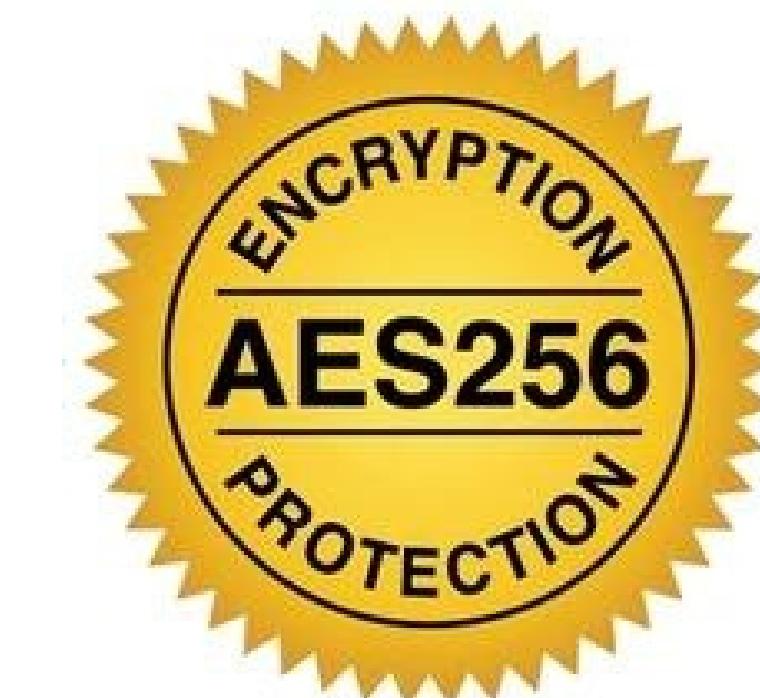
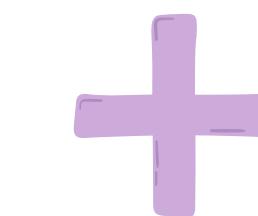


Codice **UNIVOCO** che
permette di risalire al
Profilo Digitale dello
Studente salvato nel
Database.

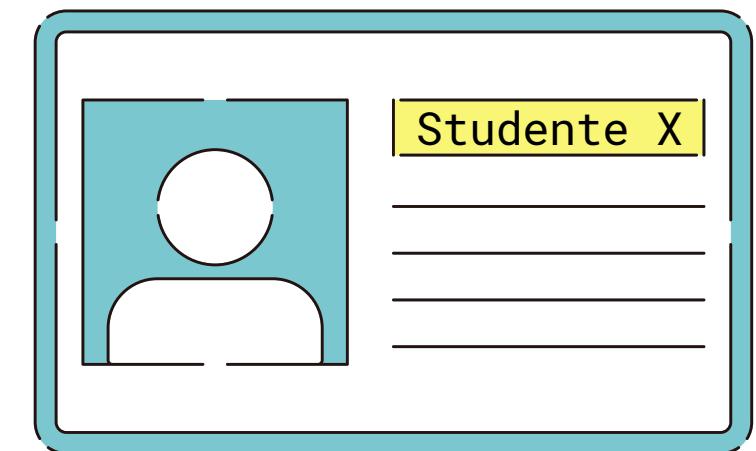
Cifratura dei Dati



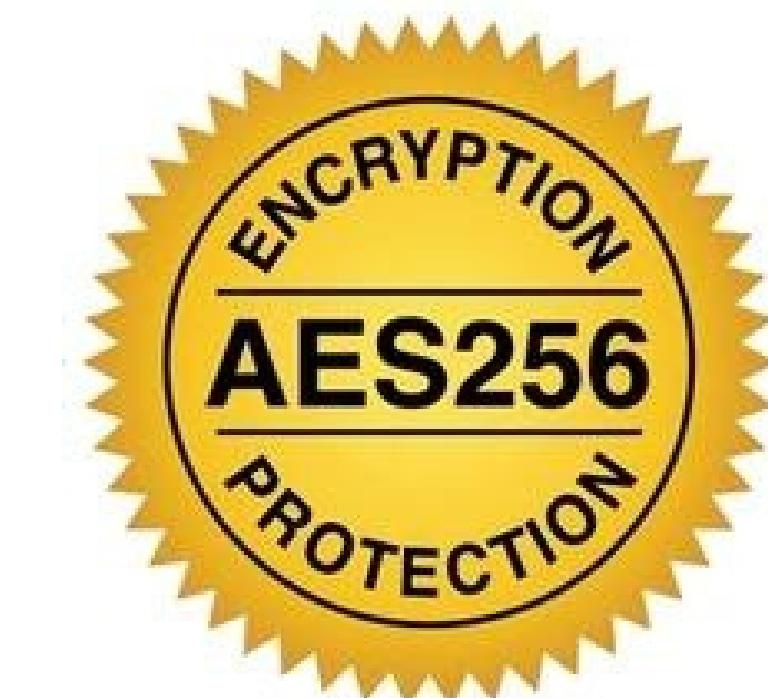
Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



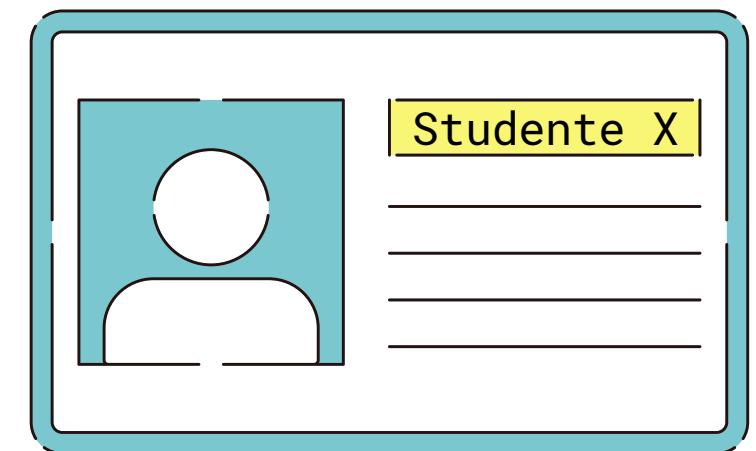
Cifratura dei Dati



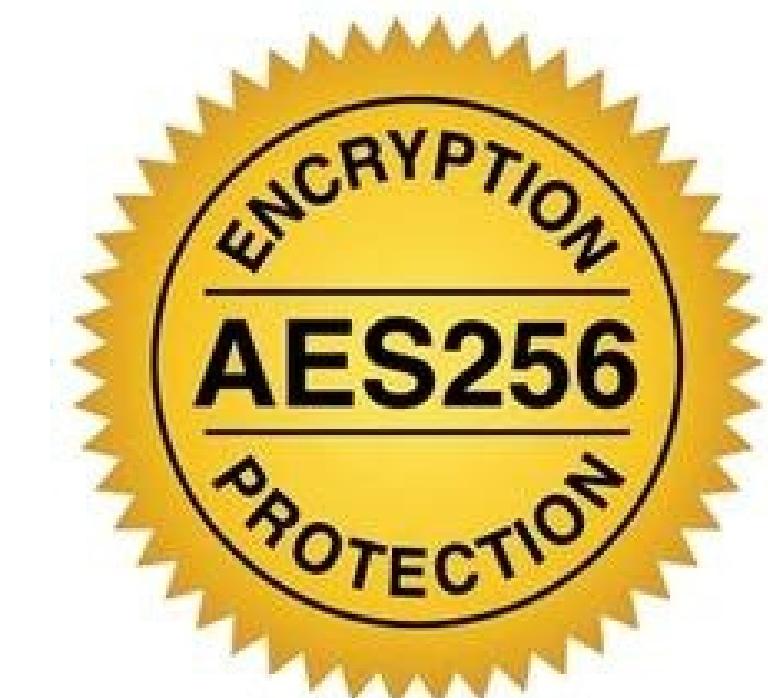
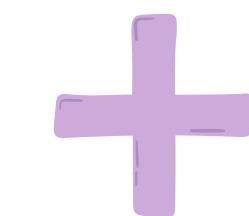
Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



Cifratura dei Dati



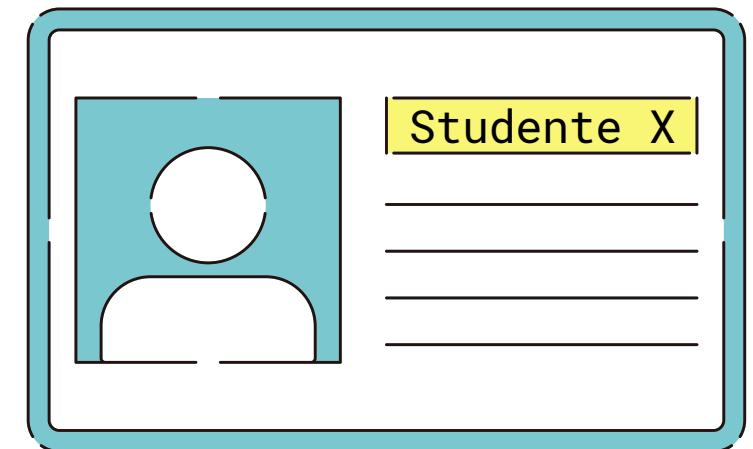
Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



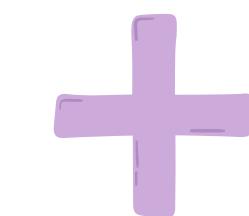
e IV



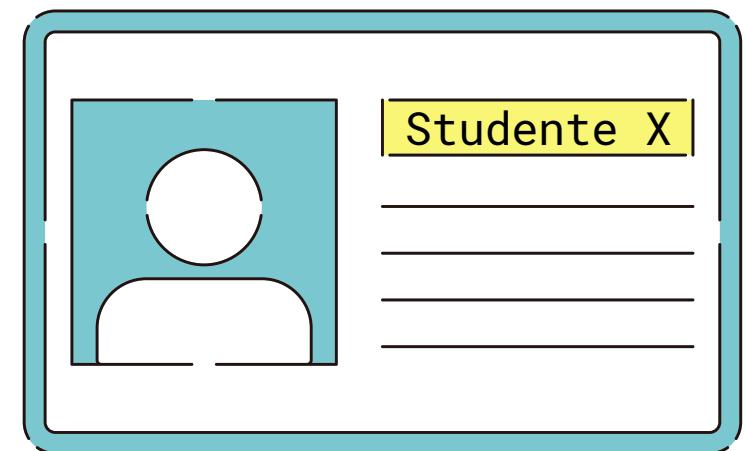
Cifratura dei Dati



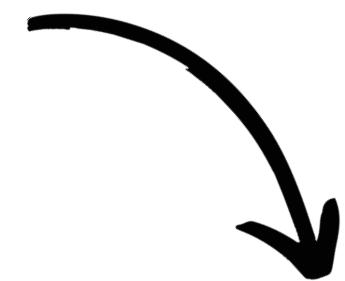
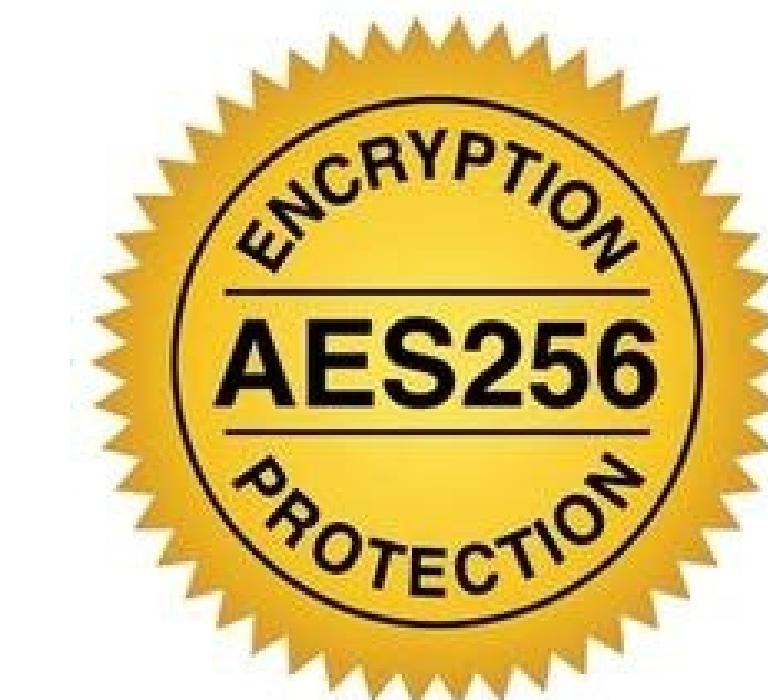
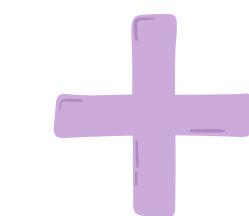
Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



Cifratura dei Dati



Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



Gestione e IV

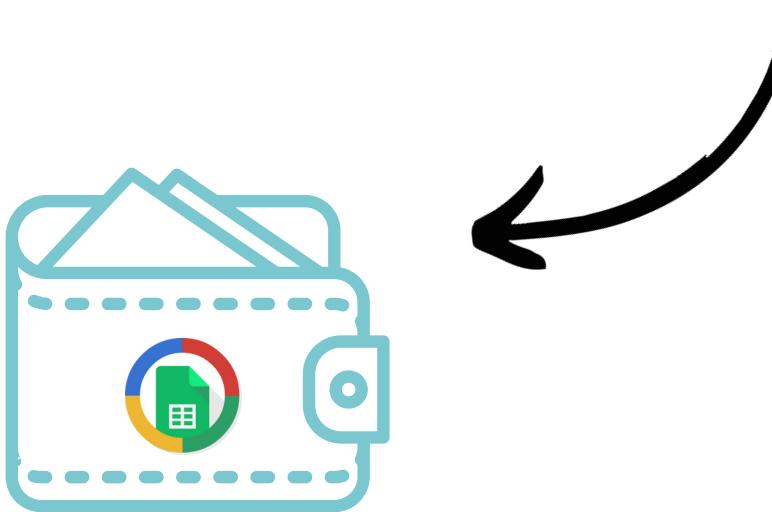


Gestione e IV

\forall Operazione $\in \{\text{Iscrizione, Lezione, Esame}\}$
 $\exists!$ (k,iv) \in Wallet | Operazione \leftrightarrow (k,iv)

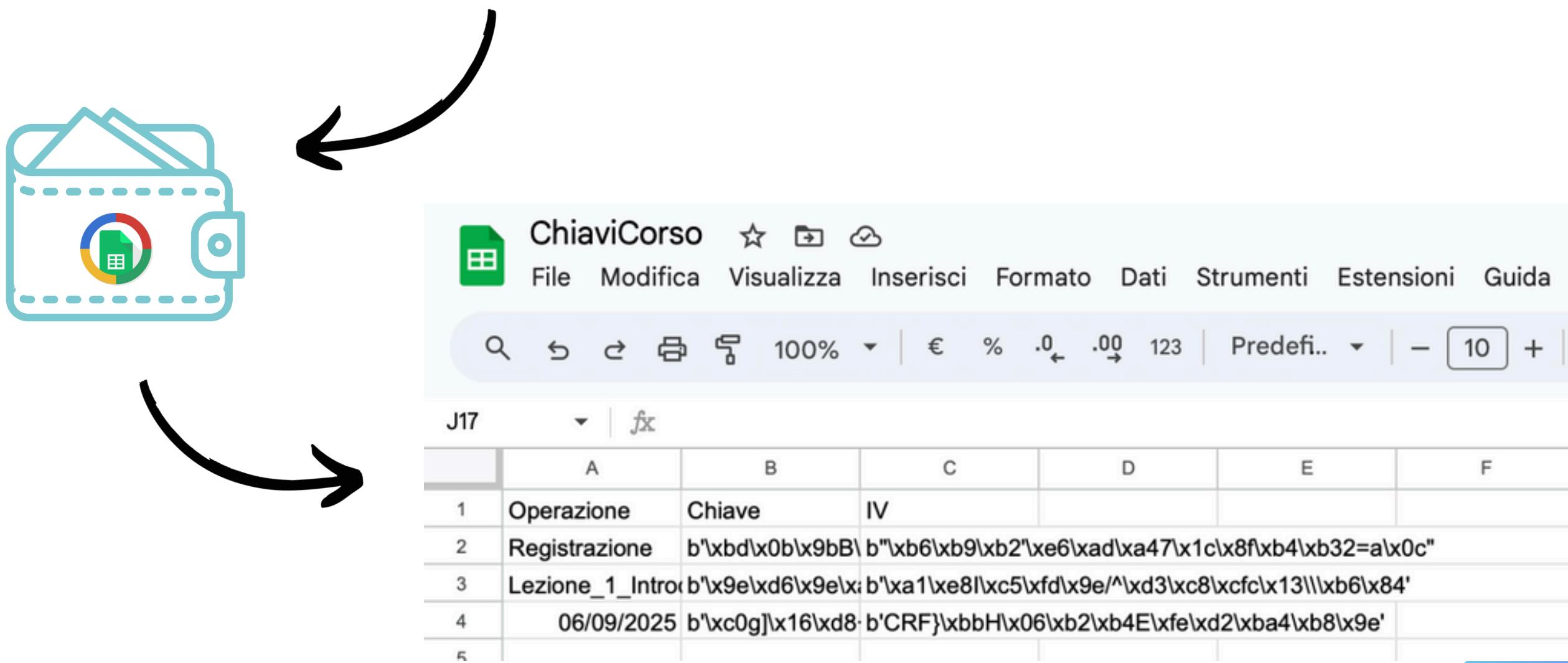
Gestione e IV

$\forall \text{Operazione} \in \{\text{Iscrizione, Lezione, Esame}\}$
 $\exists! (k, iv) \in \text{Wallet} \mid \text{Operazione} \leftrightarrow (k, iv)$



Gestione e IV

$\forall \text{Operazione} \in \{\text{Iscrizione, Lezione, Esame}\}$
 $\exists! (k, iv) \in \text{Wallet} \mid \text{Operazione} \leftrightarrow (k, iv)$



Gestione e IV

\forall Operazione $\in \{\text{Iscrizione, Lezione, Esame}\}$
 $\exists!$ (k,iv) \in Wallet | Operazione \leftrightarrow (k,iv)

Gestione e IV

$\forall \text{Operazione} \in \{\text{Iscrizione, Lezione, Esame}\}$
 $\exists! (k, iv) \in \text{Wallet} \mid \text{Operazione} \leftrightarrow (k, iv)$



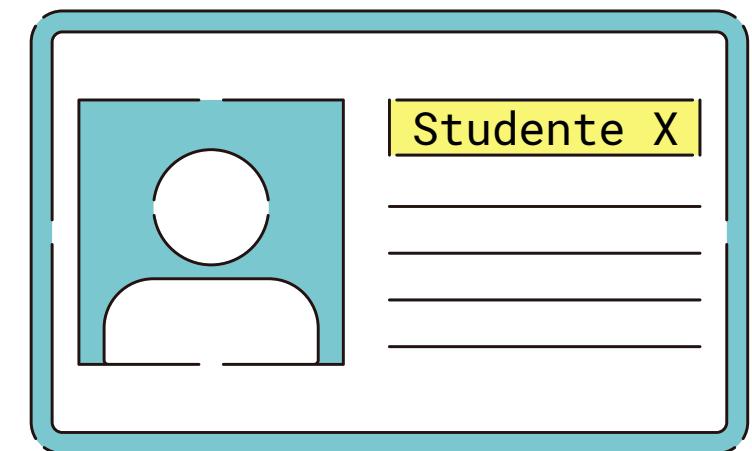
\neq



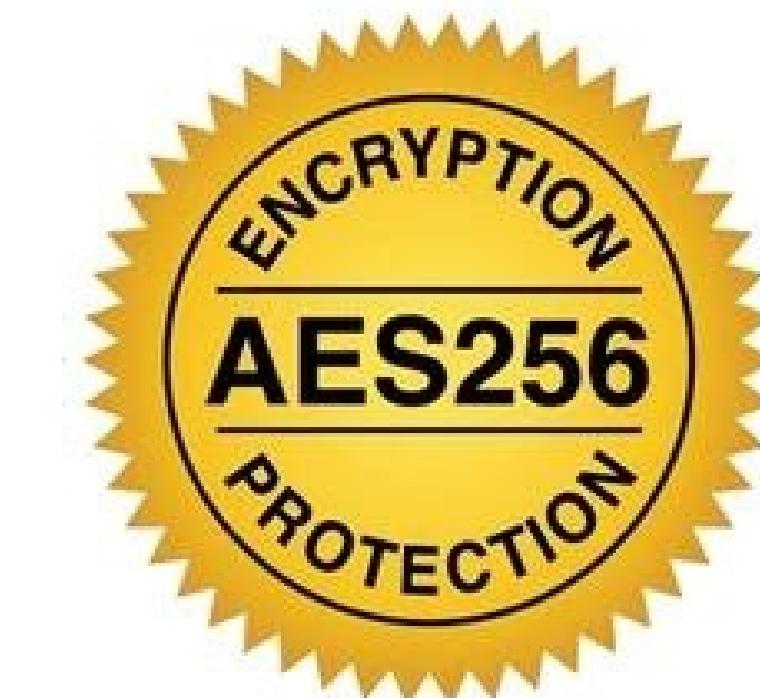
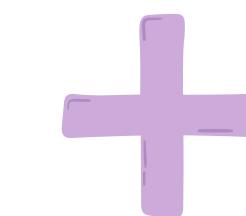
Codice UNIVOCO di
Studente X CIFRATO con k
e iv di Lezione_1

Codice UNIVOCO di
Studente X CIFRATO con k
e iv di Lezione_2

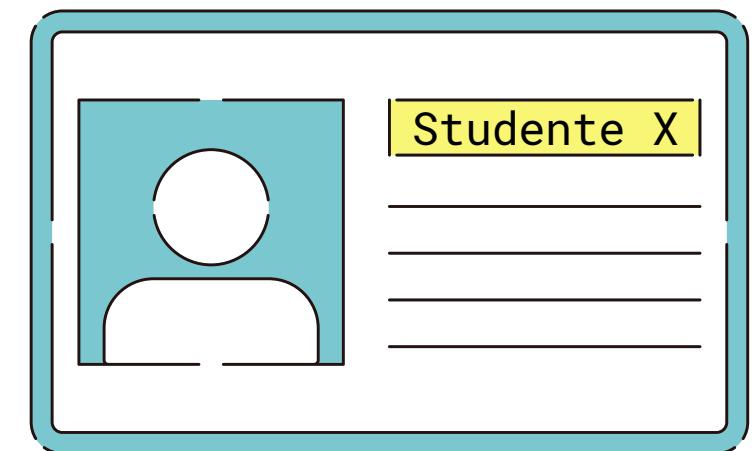
Cifratura dei Dati



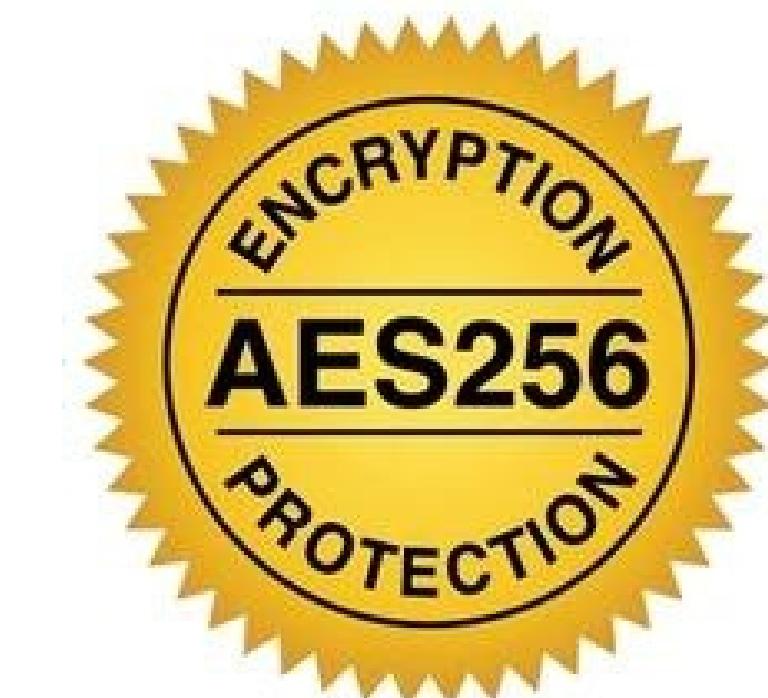
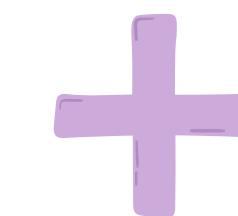
Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



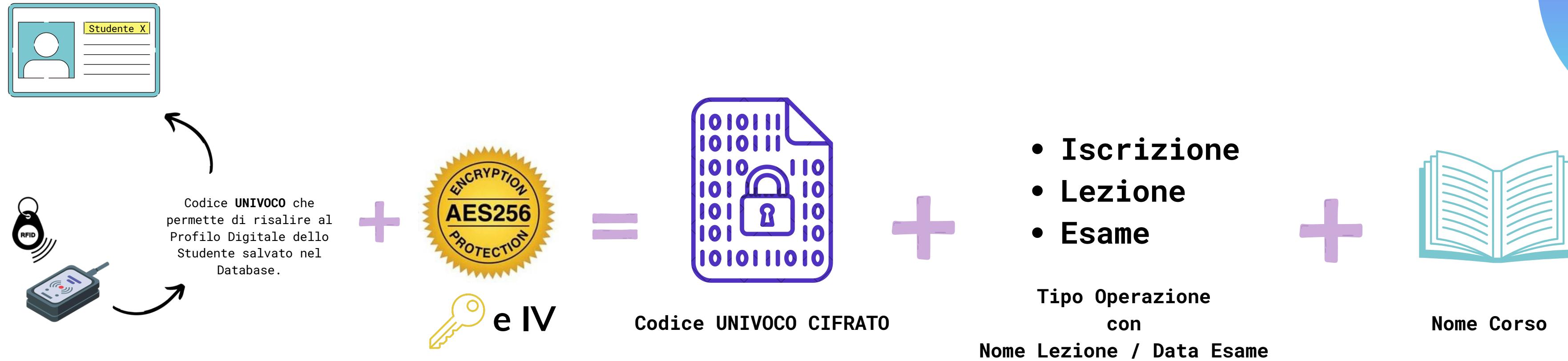
Cifratura dei Dati



Codice **UNIVOCO** che permette di risalire al Profilo Digitale dello Studente salvato nel Database.



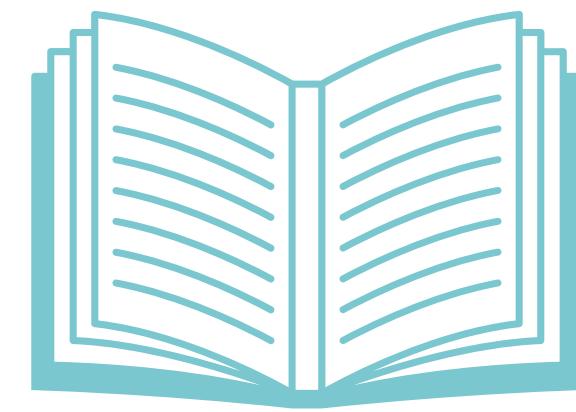
Cifratura dei Dati



Dati Ottenuuti



- **Iscrizione**
- **Lezione**
- **Esame**



Codice UNIVOCO CIFRATO

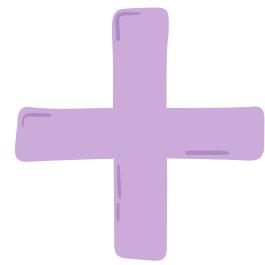
Tipo Operazione
con
Nome Lezione / Data Esame

Nome Corso

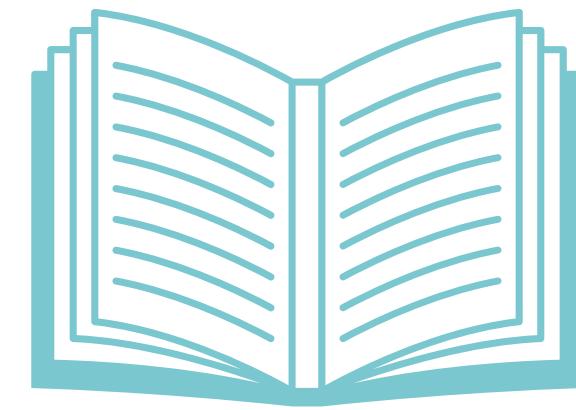
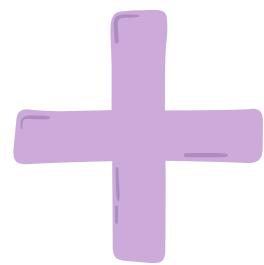
Dati Ottenuuti



Codice UNIVOCO CIFRATO



- Iscrizione
- Lezione
- Esame



Nome Corso

Tipo Operazione
con
Nome Lezione / Data Esame

Dati per la Transazione

BlockChain



```

1 // SPDX-License-Identifier: MIT
2 pragma solidity ^0.8.0;
3
4 contract AttendanceTracker {
5     struct Record {
6         string operationType; // "Registrazione", "Lezione", "Esame"
7         string courseName; // Nome del corso
8         string additionalInfo; // Può essere "nomelezione" o "dataEsame", se l'operazione è "Registrazione" il campo è vuoto
9         string encryptedId; // ID dello studente cifrato
10    }
11
12    // Array di tutti i record
13    Record[] private records;
14
15    // Evento emesso ogni volta che un record viene creato
16    event RecordCreated(string operationType, string courseName, string additionalInfo, string encryptedId);
17
18    // Funzione per aggiungere un record
19    function addRecord(string memory operationType, string memory courseName, string memory additionalInfo, string memory encryptedId) public {
20        records.push(Record(operationType, courseName, additionalInfo, encryptedId));
21        emit RecordCreated(operationType, courseName, additionalInfo, encryptedId);
22    }
23
24    // Funzione per contare le registrazioni di un dato corso
25    function countRegistrations(string memory courseName) public view returns (uint) {
26        uint count = 0;
27        for(uint i = 0; i < records.length; i++) {
28            if(keccak256(bytes(records[i].operationType)) == keccak256(bytes("Registrazione")) && keccak256(bytes(records[i].courseName)) == keccak256(bytes(courseName))) {
29                count++;
30            }
31        }
32        return count;
33    }
34
35    // Funzione per contare le presenze per una data lezione di un corso specifico
36    function countLessonAttendances(string memory courseName, string memory lessonName) public view returns (uint) {
37        uint count = 0;
38        for(uint i = 0; i < records.length; i++) {
39            if(keccak256(bytes(records[i].operationType)) == keccak256(bytes("Lezione")) &&
40                keccak256(bytes(records[i].additionalInfo)) == keccak256(bytes(lessonName)) &&
41                keccak256(bytes(records[i].courseName)) == keccak256(bytes(courseName))) {
42                count++;
43            }
44        }
45        return count;
46    }

```

BlockChain

SMARTCONTRACT



BlockChain

SMARTCONTRACT

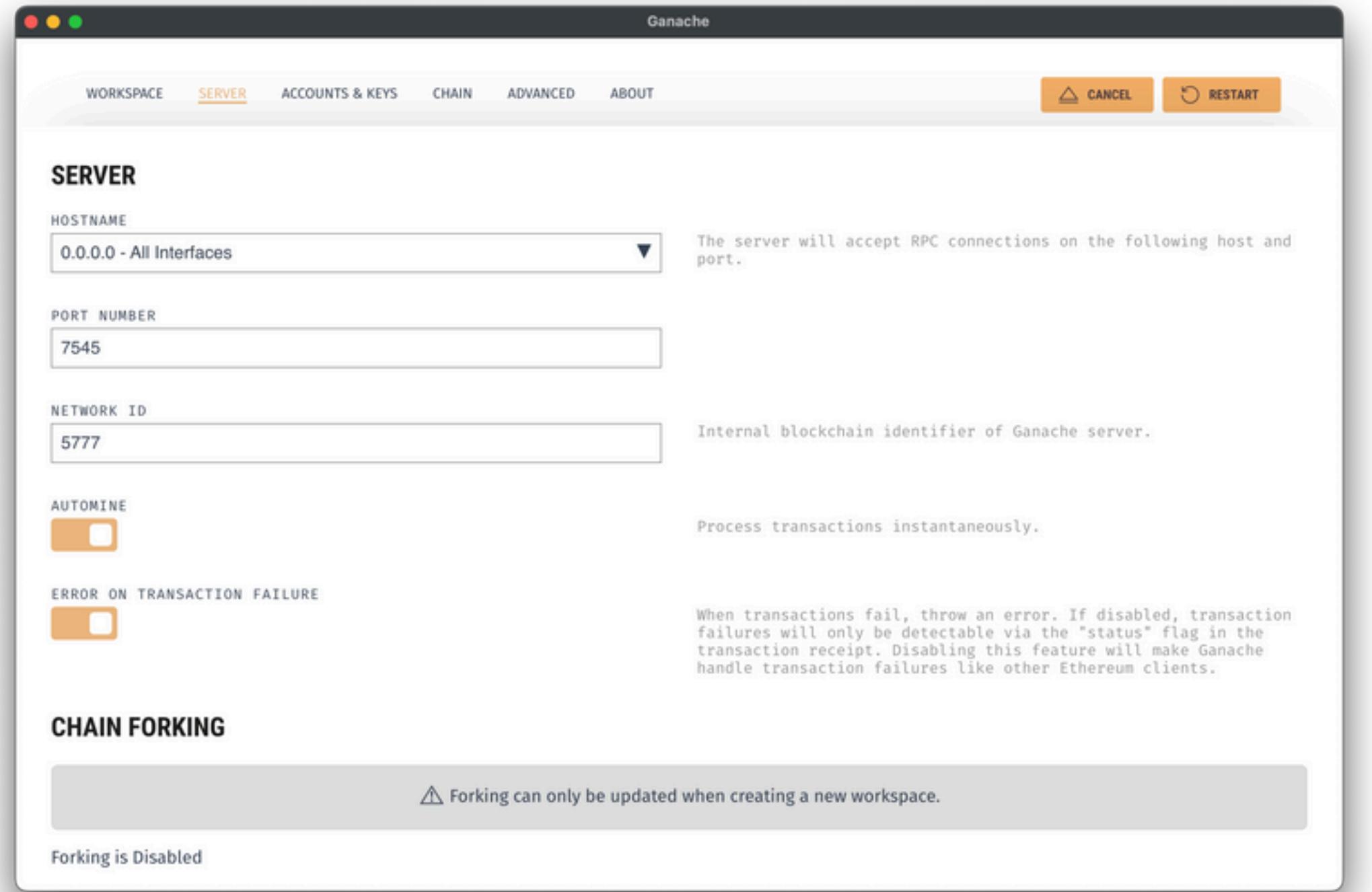


```
47
48 // Funzione per contare quanti studenti hanno partecipato ad un dato esame di un determinato corso
49 function countExamParticipations(string memory courseName, string memory examDate) public view returns (uint) {
50     uint count = 0;
51     for(uint i = 0; i < records.length; i++) {
52         if(keccak256(bytes(records[i].operationType)) == keccak256(bytes("Esame")) &&
53             keccak256(bytes(records[i].additionalInfo)) == keccak256(bytes(examDate)) &&
54             keccak256(bytes(records[i].courseName)) == keccak256(bytes(courseName))) {
55             count++;
56         }
57     }
58     return count;
59 }
60
61 // Funzione per ottenere i record in base al tipo di operazione e dettagli specificati
62 function getRecordsByOperation(string memory operationType, string memory courseName, string memory additionalInfo) public view returns (Record[] memory) {
63     Record[] memory tempRecords = new Record[](records.length);
64     uint count = 0;
65
66     for(uint i = 0; i < records.length; i++) {
67         bool matchOperationType = keccak256(bytes(records[i].operationType)) == keccak256(bytes(operationType));
68         bool matchCourseName = keccak256(bytes(records[i].courseName)) == keccak256(bytes(courseName));
69         bool matchAdditionalInfo = keccak256(bytes(records[i].additionalInfo)) == keccak256(bytes(additionalInfo)) || keccak256(bytes(additionalInfo)) == keccak256("");
70
71         if(matchOperationType && matchCourseName && (matchAdditionalInfo || keccak256(bytes(operationType)) == keccak256(bytes("Registrazione")))) {
72             tempRecords[count] = records[i];
73             count++;
74         }
75     }
76
77     Record[] memory filteredRecords = new Record[](count);
78     for(uint j = 0; j < count; j++) {
79         filteredRecords[j] = tempRecords[j];
80     }
81
82     return filteredRecords;
83 }
84 }
```

BlockChain

GANACHE

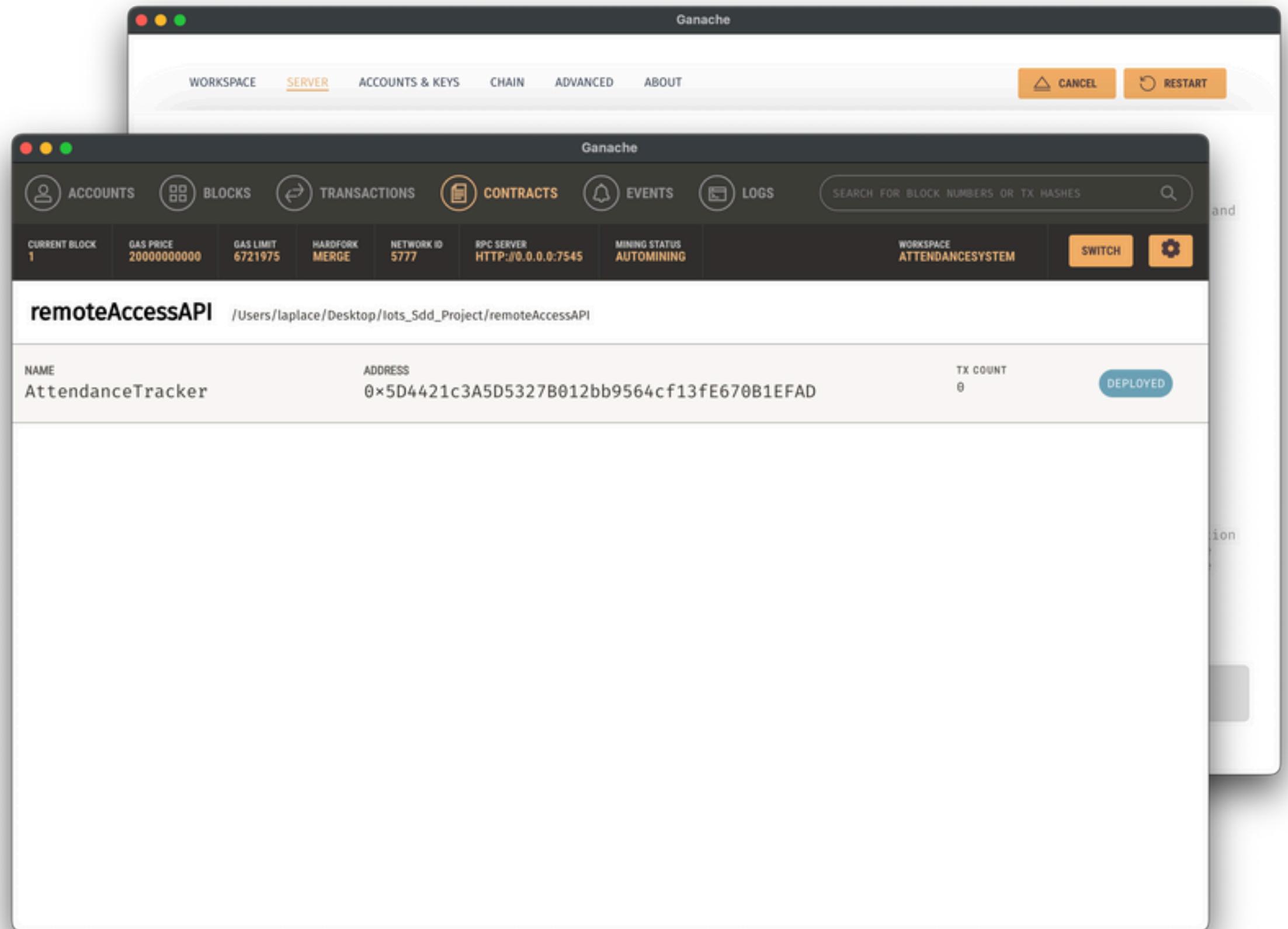




BlockChain

GANACHE





BlockChain

GANACHE



BlockChain

GANACHE



Ganache

WORKSPACE SERVER ACCOUNTS & KEYS CHAIN ADVANCED ABOUT CANCEL RESTART

Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK 1 GAS PRICE 20000000000 GAS LIMIT 6721975 HARDFORK MERGE NETWORK ID 5777 RPC SERVER HTTP://0.0.0.0:7545 MINING STATUS AUTOMINING WORKSPACE ATTENDANCESYSTEM SWITCH

Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK 1 GAS PRICE 20000000000 GAS LIMIT 6721975 HARDFORK MERGE NETWORK ID 5777 RPC SERVER HTTP://0.0.0.0:7545 MINING STATUS AUTOMINING WORKSPACE ATTENDANCESYSTEM SWITCH DEPLOYED

TX HASH 0x9744effea04180783ecc035cd8afb8ce6f05c8708da8e9990daed34179cd556a CONTRACT CREATION

FROM ADDRESS 0xd27D7405775F3C7F0A8b00C14297954F3F11B656 CREATED CONTRACT ADDRESS 0x5D4421c3A5D5327B012bb9564cf13fE670B1EFAD GAS USED 1274156 VALUE 0

BlockChain

GANACHE



Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK: 4 GAS PRICE: 20000000000 GAS LIMIT: 6721975 HARDFORK: MERGE NETWORK ID: 5777 RPC SERVER: HTTP://0.0.0.0:7545 MINING STATUS: AUTOMINING WORKSPACE: ATTENDANCESYSTEM SWITCH

EVENT NAME	TX HASH	LOG INDEX	BLOCK TIME
RecordCreated	0x9156bea86a8f96930e04b6b157b342f7407c1f5db0aa790 79b57a79d492d9511	0	2024-09-03 20:22:42
RecordCreated	0x292b47b0dal1c1554613cf08174954a5870788623819c6b9 d90e56ff4658790b8	0	2024-09-03 20:14:40
RecordCreated	0x22e1f9d4de1dae4fdf70f48d15b00387d130753d592cc79 139340465d36655b5	0	2024-09-03 20:10:51

BlockChain

GANACHE



Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK: 4 GAS PRICE: 20000000000 GAS LIMIT: 6721975 HARDFORK: MERGE NETWORK ID: 5777 RPC SERVER: HTTP://0.0.0.0:7545 MINING STATUS: AUTOMINING WORKSPACE: ATTENDANCESYSTEM SWITCH

-- BACK 0x9156bea86a8f96930e04b6b157b342f7407c1f5db0aa79079b57a79d492d9511 (0)

CONTRACT NAME: AttendanceTracker CONTRACT ADDRESS: 0x5D4421c3A5D5327B012bb9564cf13fE670B1EFAD

SIGNATURE (DECODED): RecordCreated(operationType: string, courseName: string, additionalInfo: string, encryptedId: string)

TX HASH: 0x9156bea86a8f96930e04b6b157b342f7407c1f5db0aa79079b57a79d492d9511 LOG INDEX: 0 BLOCK TIME: 2024-09-03 20:22:42

RETURN VALUES

OPERATIONTYPE: Registrazione

COURSENAME: SDD_2025

ADDITIONALINFO:

ENCRYPTEDID: b'yi\xbe\xcc\x11`\x82{qn\x19\x80\xba\x8b\x0c\t'

BlockChain

GANACHE



Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK: 10 GAS PRICE: 20000000000 GAS LIMIT: 6721975 HARDFORK: MERGE NETWORK ID: 5777 RPC SERVER: HTTP://0.0.0.0:7545 MINING STATUS: AUTOMINING WORKSPACE: ATTENDANCESYSTEM SWITCH

← BACK 0xd178b22fc865d45134c15e05b8fa0104dc4cecf851804450f42862a945487918 (0)

CONTRACT NAME: AttendanceTracker CONTRACT ADDRESS: 0x5D4421c3A5D5327B012bb9564cf13fE670B1EFAD

SIGNATURE (DECODED): RecordCreated(operationType: string, courseName: string, additionalInfo: string, encryptedId: string)

TX HASH: 0xd178b22fc865d45134c15e05b8fa0104dc4cecf851804450f42862a945487918 LOG INDEX: 0 BLOCK TIME: 2024-09-03 20:31:26

RETURN VALUES

OPERATIONTYPE: Lezione

COURSENAME: SDD_2025

ADDITIONALINFO: Lezione_1_Introduzione

ENCRYPTEDID: b'\x9eeq\x2r\xdcx\xd3\x04h\xff\xd1\xf3z\xcc'

BlockChain

GANACHE



Ganache

ACCOUNTS BLOCKS TRANSACTIONS CONTRACTS EVENTS LOGS SEARCH FOR BLOCK NUMBERS OR TX HASHES

CURRENT BLOCK: 10 GAS PRICE: 20000000000 GAS LIMIT: 6721975 HARDFORK: MERGE NETWORK ID: 5777 RPC SERVER: HTTP://0.0.0.0:7545 MINING STATUS: AUTOMINING WORKSPACE: ATTENDANCESYSTEM SWITCH

← BACK 0x6c9dd0f6066acca4d94f92adf8027457f0ba3b8587ae850d33dfe9d90a39cd59 (0)

CONTRACT NAME: AttendanceTracker CONTRACT ADDRESS: 0x5D4421c3A5D5327B012bb9564cf13fE670B1EFAD

SIGNATURE (DECODED): RecordCreated(operationType: string, courseName: string, additionalInfo: string, encryptedId: string)

TX HASH: 0x6c9dd0f6066acca4d94f92adf8027457f0ba3b8587ae850d33dfe9d90a39cd59 LOG INDEX: 0 BLOCK TIME: 2024-09-03 20:36:59

RETURN VALUES

OPERATIONTYPE: Esame

COURSENAME: SDD_2025

ADDITIONALINFO: 06/09/2025

ENCRYPTEDID: b'\xf6Au\x06\x97Z\x04\xd9.\xed\x8e\x0b5\x87!'

Servizi Google



Servizi Google



- Google Drive: Archiviazione e gestione dei file;

Servizi Google



- Google Drive: Archiviazione e gestione dei file;
- Google Sheets: Creazione e aggiornamento di fogli di calcolo per la registrazione dei dati;

Servizi Google



- Google Drive: Archiviazione e gestione dei file;
- Google Sheets: Creazione e aggiornamento di fogli di calcolo per la registrazione dei dati;
- Google Cloud: Autenticazione e gestione delle credenziali per l'accesso ai servizi.

Servizi Google



createGsheet.py

- 1. Avvia il flusso di autenticazione dell'utente ai servizi Google.**
- 2. Crea una cartella per il Corso.**

```
 1 import os
 2 import json
 3 import google.auth
 4 from google.oauth2.credentials import Credentials
 5 from googleapiclient.discovery import build
 6 from googleapiclient.errors import HttpError
 7 from googleapiclient.http import MediaFileUpload
 8
 9 SCOPES = ['https://www.googleapis.com/auth/drive']
10 CREDENTIALS_FILE = 'credentials.json'
11
12 def auth():
13     """Authenticates to Google Drive API.
14     Returns:
15         An authorized Google Drive API service object.
16     """
17     credentials = None
18     if os.path.exists(CREDENTIALS_FILE):
19         with open(CREDENTIALS_FILE, 'r') as token:
20             credentials = Credentials.from_authorized_user_file(token)
21     # If not found or not valid, raise exception
22     if not credentials or not credentials.valid:
23         if credentials and credentials.expired and credentials.refresh_token:
24             credentials.refresh(Request())
25         else:
26             flow = InstalledAppFlow.from_client_secrets_file(
27                 CREDENTIALS_FILE, SCOPES)
28             credentials = flow.run_local_server()
29     return build('drive', 'v3', credentials=credentials)
30
31 def create_folder(service, folder_name, parent_id):
32     body = {
33         "name": folder_name,
34         "mimeType": "application/vnd.google-apps.folder",
35         "parents": [parent_id]
36     }
37     response = service.files().create(body=body).execute()
38     folder = response.get("id")
39
40     return folder
41
42 def get_file(service, file_id):
43     file = service.files().get(fileId=file_id).execute()
44     return file
45
46 def update_file(service, file_id, new_name):
47     file = service.files().update(fileId=file_id, name=new_name).execute()
48
49     return file
50
51 def main():
52     """Shows basic usage of the Google Drive API.
53     Prints the names and ids of the first 10 files and folders.
54     """
55     # The file ID of a sample document to print.
56     FILE_ID = '1JGgXWzqfjyDwzCQHcOOGPQYUWzZBzI'
57
58     # If modifying these scopes, delete the file token.pickle.
59     SCOPES = ['https://www.googleapis.com/auth/drive']
60
61     # The path to a file containing the OAuth 2.0 information for this
62     # application, including its client ID, client secret, and
63     # OAuth 2.0 token.
64     CREDENTIALS_FILE = 'credentials.json'
65
66     # Create a Google Drive API service object.
67     service = auth()
68
69     # Print the names and IDs of up to 10 files and folders.
70     results = service.files().list(pageSize=10).execute()
71     items = results.get('items', [])
72
73     if not items:
74         print('No files found.')
75     else:
76         print(f'{len(items)} files found.')
77         for item in items:
78             print(f'{item["name"]} ({item["id"]})')
79
80
81 if __name__ == '__main__':
82     main()
```

Servizi Google



createGsheet.py

3. Crea il foglio di calcolo, con intestazioni diverse in base al tipo di operazione, pronto per contenere i dati degli studenti:

- **iscritti al corso;**
 - **partecipanti alla lezione;**
 - **iscritti all'esame.**

Servizi Google



createGsheet.py

A screenshot of a Google Sheets document titled "Registrazione". The sheet has four columns labeled A, B, C, and D. Row 1 contains the headers: Tag_ID, Nome, Cognome, and Matricola. Rows 2 through 5 are empty.

	A	B	C	D	
1	Tag_ID	Nome	Cognome	Matricola	
2					
3					
4					
5					

Servizi Google



updateGsheet.py

1. Avvia il flusso di autenticazione dell'utente ai servizi Google.
 2. Cerca la cartella del Corso.

Servizi Google



updateGsheet.py

3. Inserisce nel foglio di calcolo i dati degli studenti:

Per i nuovi iscritti al corso:

- ID, Nome, Cognome, Matricola

Per i partecipanti alla lezione:

- ID, Nome, Cognome, Matricola
Orario di Arrivo

Per gli iscritti all'esame:

- **ID, Nome, Cognome, Matricola**

```
    if not Python:
        if len(error_lines) > 0:
            return Python, error_lines[0], error_lines[1]
    else:
        return Python[0], getPython()

def report_error_to_distro(distro, version, distro_id, distro_name, message):
    if distro_id == "UbuntuServer":
        current_error_message = "Ubuntu server"
    elif distro_id == "UbuntuGuest":
        current_error_message = "Ubuntu guest"
    else:
        current_error_message = "Python %s" % distro[0]

    request = requests.get("https://api.ubuntu.com/v1/errors/%s" % current_error_message)
    operation_id = str(uuid.uuid4())
    configuration_error_message = "Ubuntu %s error: %s" % (distro_name, message)
    body = {"error": configuration_error_message}
    response = request.put(operation_id, json=body)

    print("Python reports an error of %s. (%s) %s" % (current_error_message, distro_id, message))

def report_error_to_distro(distro, version, distro_id, distro_name, message):
    if not Python:
        if len(error_lines) > 0:
            return Python, error_lines[0], error_lines[1]
    else:
        return Python[0], getPython()

    if distro_id == "UbuntuServer":
        current_error_message = "Ubuntu server"
    elif distro_id == "UbuntuGuest":
        current_error_message = "Ubuntu guest"
    else:
        current_error_message = "Python %s" % distro[0]

    request = requests.get("https://api.ubuntu.com/v1/errors/%s" % current_error_message)
    operation_id = str(uuid.uuid4())
    configuration_error_message = "Ubuntu %s error: %s" % (distro_name, message)
    body = {"error": configuration_error_message}
    response = request.put(operation_id, json=body)

    print("Python reports an error of %s. (%s) %s" % (current_error_message, distro_id, message))

def report_error_to_distro(distro, version, distro_id, distro_name, message):
    if not Python:
        if len(error_lines) > 0:
            return Python, error_lines[0], error_lines[1]
    else:
        return Python[0], getPython()
```

Servizi Google



updateGsheet.py

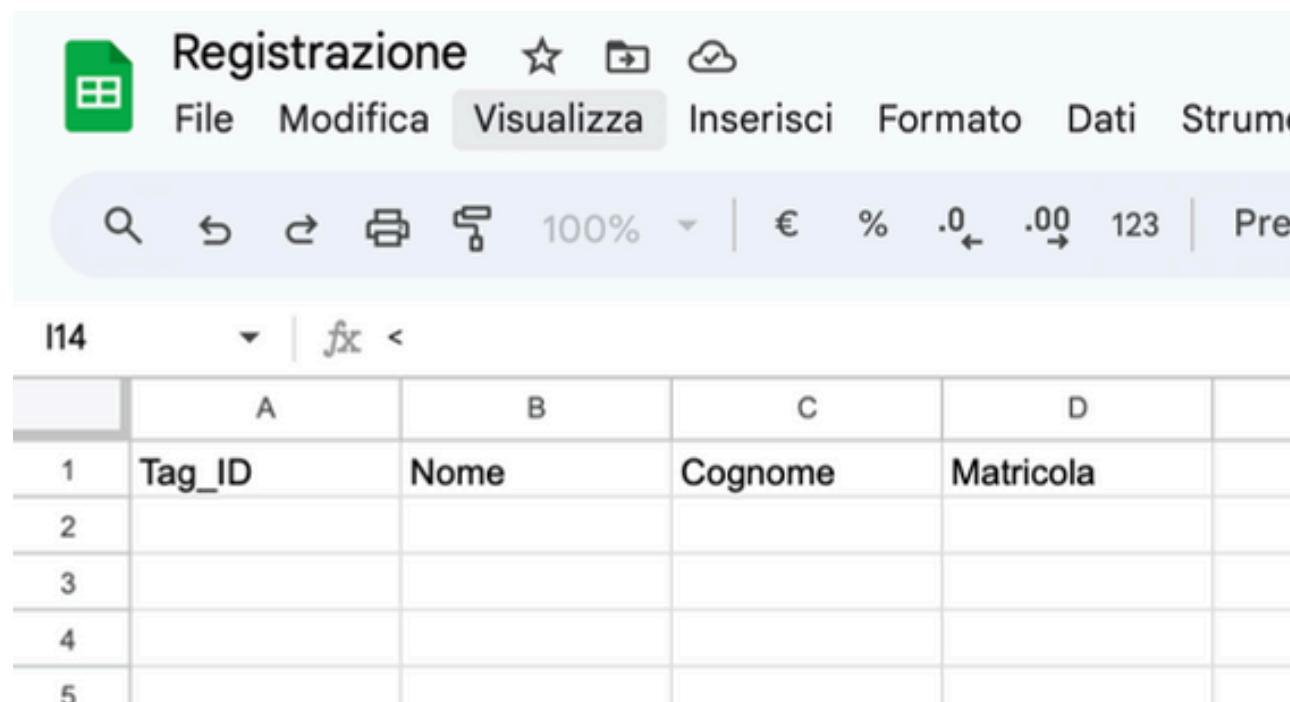
A screenshot of a Google Sheets document titled "Registrazione". The interface includes a menu bar with File, Modifica, Visualizza, Inserisci, Formato, Dati, and Strumenti. Below the menu is a toolbar with search, filter, print, and other tools. The sheet has columns A, B, C, and D. Row 1 contains headers: Tag_ID, Nome, Cognome, and Matricola. Rows 2 through 5 are empty. The formula bar shows "I14" and "fx <".

	A	B	C	D	
1	Tag_ID	Nome	Cognome	Matricola	
2					
3					
4					
5					

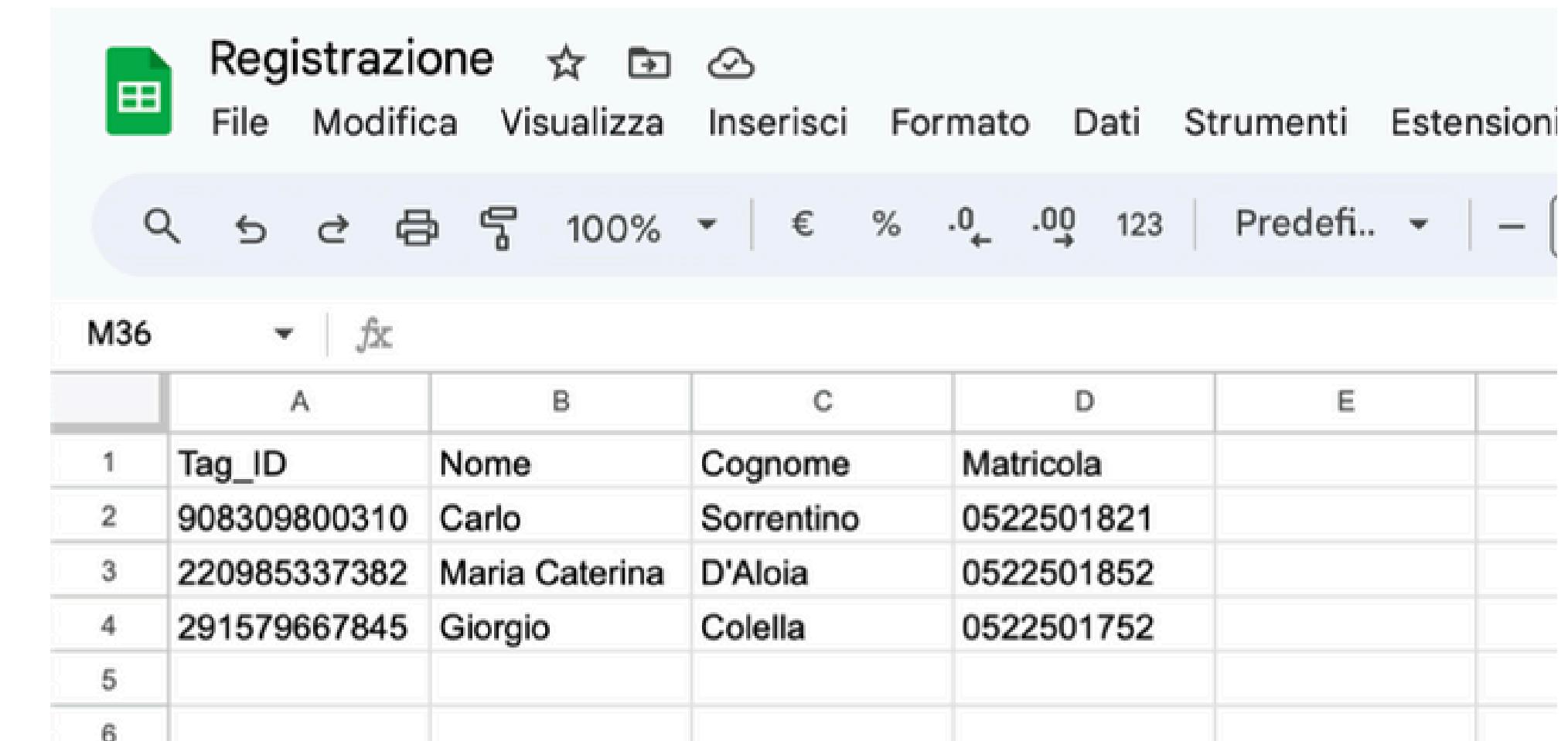
Servizi Google



updateGsheet.py



	A	B	C	D	
1	Tag_ID	Nome	Cognome	Matricola	
2					
3					
4					
5					



	A	B	C	D	E
1	Tag_ID	Nome	Cognome	Matricola	
2	908309800310	Carlo	Sorrentino	0522501821	
3	220985337382	Maria Caterina	D'Aloia	0522501852	
4	291579667845	Giorgio	Colella	0522501752	
5					
6					

Servizi Google



keyChainGsheet.py

1. Avvia il flusso di autenticazione dell'utente ai servizi Google.
2. Cerca la cartella del Corso.

```
#!/usr/bin/python3
# -*- coding: utf-8 -*-
# Author: [REDACTED]
# Date: [REDACTED]
# Description: [REDACTED]

# Import required libraries
import os
import json
from google.oauth2 import service_account
from googleapiclient.discovery import build
from googleapiclient.errors import HttpError

# Set up the Google Sheets API client
SCOPES = ['https://www.googleapis.com/auth/spreadsheets']
KEY_FILE = 'credentials.json'
SERVICE_ACCOUNT_FILE = 'service-account-key.json'

def get_credentials():
    """Get credentials from a service account file"""
    credentials = service_account.Credentials.from_service_account_file(
        SERVICE_ACCOUNT_FILE, scopes=SCOPES)
    return credentials

def main():
    """Main function to interact with Google Sheets API"""
    # Create a Sheets API client
    credentials = get_credentials()
    client = build('sheets', 'v4', credentials=credentials)

    # Define the range to read
    range_name = 'A1:D2'

    # Call the Sheets API
    try:
        result = client.spreadsheets().values().get(
            spreadsheetId='[REDACTED]', range=range_name).execute()
        values = result.get('values', [])

        if not values:
            print('No data found.')
        else:
            print(f'{len(values)} rows retrieved.')
            for row in values:
                print(row)
    except HttpError as error:
        print(f'An error occurred: {error}')

if __name__ == '__main__':
    main()
```

Servizi Google



keyChainGsheets.py

3. Cosa restituisce?

- Se il Wallet non esiste crea il foglio di calcolo con intestazioni: Operazione, Chiave ed IV ed inserisce nei rispettivi campi i dati appena creati e li restituisce;
 - Se il Wallet esiste ed esiste anche l'operazione passata restituisce direttamente la Chiave e l'IV dell'Operazione del Corso richiesto;
 - Se il Wallet esiste ma non esiste l'operazione passata inserisce nel Wallet una nuova Chiave ed IV che poi restituisce.

Servizi Google



keyChainGsheet.py

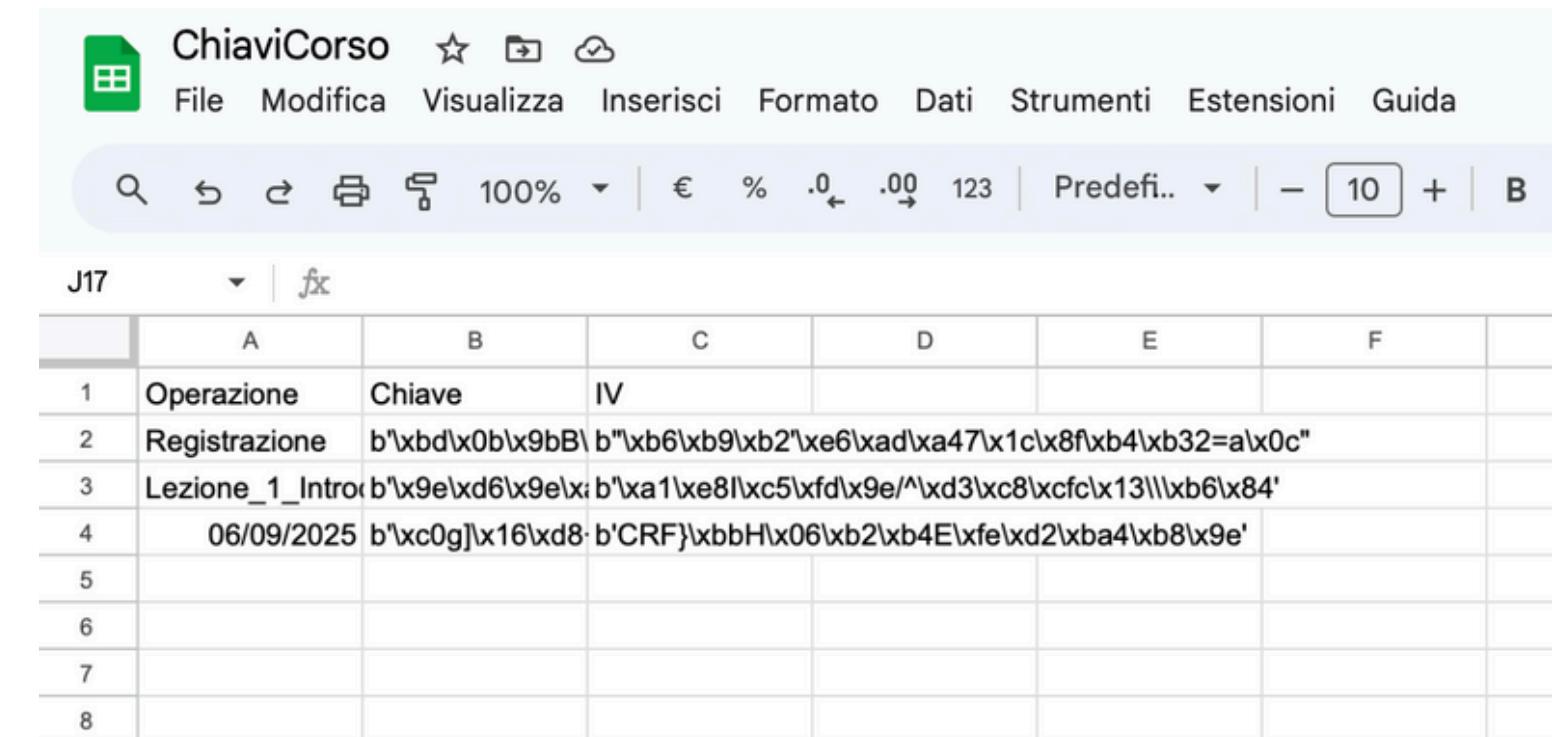
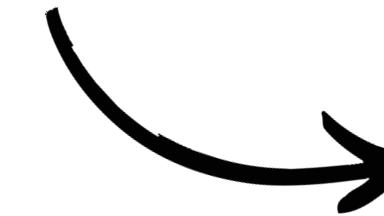
	A	B	C	D	E	F
1	Operazione	Chiave	IV			
2	Registrazione	b'\xbd\b\0\b\x9bB\b"\xb6\xb9\xb2\xe6\xad\x47\x1c\x8f\xb4\xb32=a\x0c"				
3	Lezione_1_Intro	b'\x9e\xd6\x9e\x:b'\xa1\xe8\xc5\xfd\x9e/^\\xd3\xc8\xcf\x13\\ \xb6\x84'				
4	06/09/2025	b'\xc0g]\x16\xd8 b'CRF}\xbbbH\x06\xb2\xb4E\xfe\xd2\xba4\xb8\x9e'				
5						
6						
7						
8						

Servizi Google



keyChainGsheet.py

Insert Key&IV



The screenshot shows a Google Sheets document titled "ChiaviCorso". The spreadsheet has columns A through F and rows 1 through 8. The data is as follows:

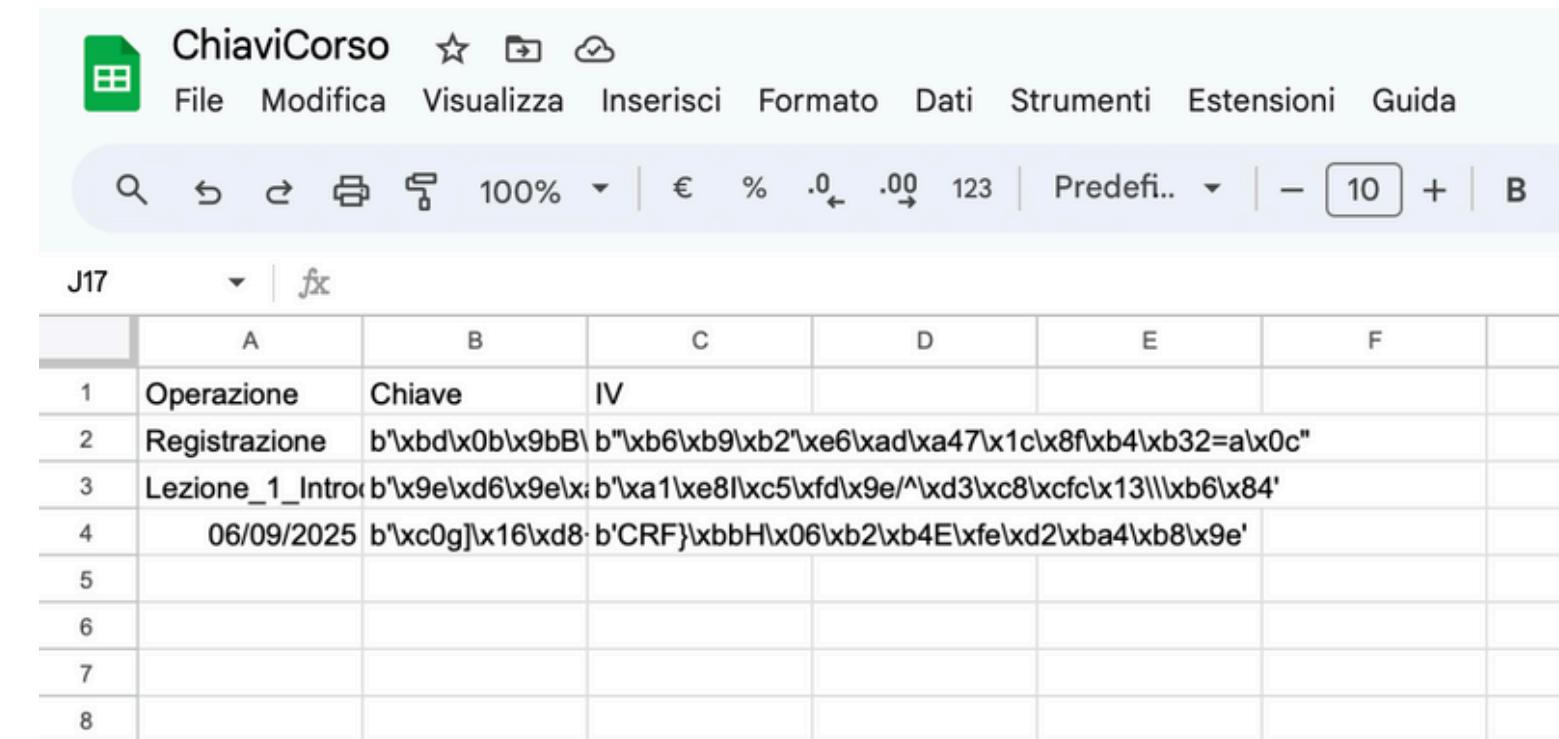
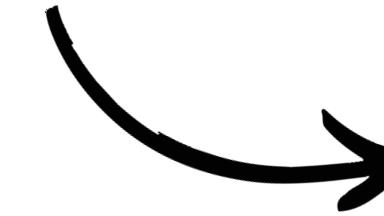
	A	B	C	D	E	F
1	Operazione	Chiave	IV			
2	Registrazione	b'\xbd\b\0b\b\x9bB\b\b"\xb6\xb9\xb2\xe6\xad\x47\x1c\x8f\xb4\xb32=a\x0c"				
3	Lezione_1_Intro	b'\x9e\xd6\x9e\x:b'\xa1\xe8\xc5\xfd\x9e/^\\xd3\xc8\xcf\x13\\ \xb6\x84'				
4	06/09/2025	b'\xc0g]\x16\xd8 b'CRF}\xbbbH\x06\xb2\xb4E\xfe\xd2\xba4\xb8\x9e'				
5						
6						
7						
8						

Servizi Google



keyChainGsheet.py

Insert Key&IV



A screenshot of a Google Sheets document titled "ChiaviCorso". The sheet has columns A through F and rows 1 through 8. Row 1 contains the headers "Operazione", "Chiave", and "IV". Rows 2, 3, and 4 contain data entries. Row 2: "Registrazione" followed by two long hex strings. Row 3: "Lezione_1_Intro" followed by a long hex string. Row 4: "06/09/2025" followed by a long hex string. The rest of the rows are empty.

	A	B	C	D	E	F
1	Operazione	Chiave	IV			
2	Registrazione	b'\xbdb\x0b\x9bB\b"\xb6\xb9\xb2\xe6\xad\x47\x1c\x8f\xb4\xb32=a\x0c"				
3	Lezione_1_Intro	b'\x9e\xd6\x9e\x:b'\xa1\xe8\xc5\xfd\x9e/^\\xd3\xc8\xcf\x13\\\'\xb6\x84'				
4	06/09/2025	b'\xc0g]\x16\xd8 b'CRF}\xbbbH\x06\xb2\xb4E\xfe\xd2\xba4\xb8\x9e'				
5						
6						
7						
8						

Return Key&IV



Servizi Google



Risultato Finale

Il mio Drive > SDD_2025 ▾

Tipo ▾ Persone ▾ Data modifica ▾

Nome ↑



Esami



ChiaviCorso



Lezione_1_Introduzione

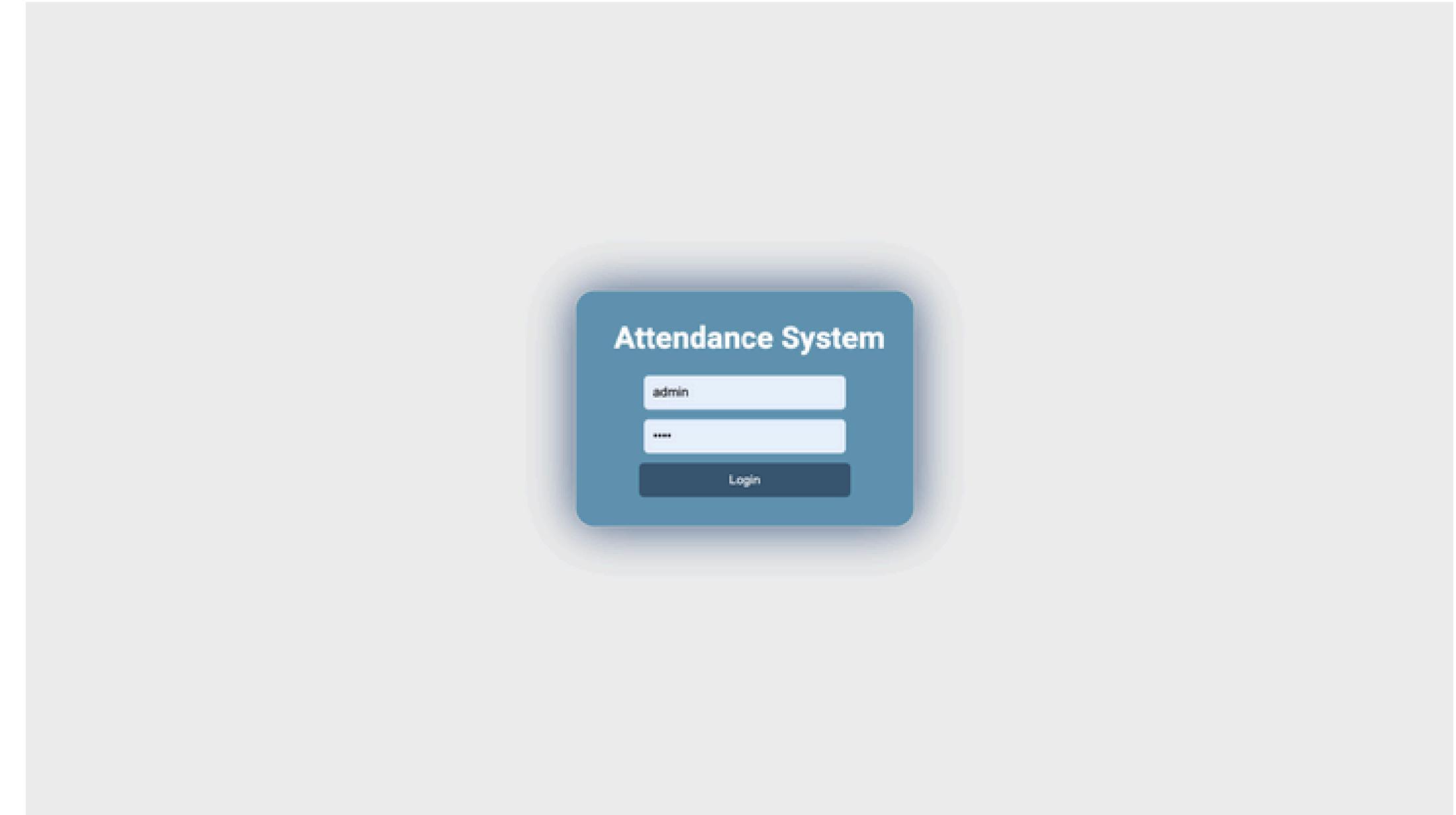


Registrazione

Interfaccia di Monitoraggio



Interfaccia di Monitoraggio



Interfaccia di Monitoraggio



Welcome to the Attendance System's API Interface [Manage Users](#) [Logout](#)

Available Operations:

Count Registrations

Course Name

Count Lesson Attendances

Course Name
 Lesson Name

Count Exam Participations

Course Name
 Exam Date

Get Transactions Record

Operation Type
 Course Name
 Additional Info (Blank If Not Needed)

Interfaccia di Monitoraggio



- 01 Count Registration
- 02 Count Lesson Attendances
- 03 Count Exam Participations
- 04 Count Lesson Attendances
- 05 Manage Users

Interfaccia di Monitoraggio



01

Count Registration

Welcome to the Attendance System's API Interface Manage Users
Logout

Available Operations:

Count Registrations

SOD_2025

Get Registrations

Count Lesson Attendances

Course Name

Lesson Name

Get Attendances

Count Exam Participations

Course Name

Exam Date

Get Participations

Get Transactions Record

Operation Type

Course Name

Additional Info (Blank If Not Needed)

Get Records

Course Name	Registrations
SOD_2025	3

Interfaccia di Monitoraggio



02

Count Lesson Attendances

Welcome to the Attendance System's API Interface Manage Users
Logout

Available Operations:

Count Registrations

Count Lesson Attendances

Count Exam Participations

Get Transactions Record

Course Name	Lesson Name	Attendances
SDD_2025	Lezione_1_Introduzione	3

Interfaccia di Monitoraggio



03

Count Exam Participations

Welcome to the Attendance System's API Interface [Manage Users](#) [Logout](#)

Available Operations:

Count Registrations

Course Name:

Get Registrations

Count Lesson Attendances

Course Name:

Lesson Name:

Get Attendances

Count Exam Participations

SDD_2025:

06/09/25:

Get Participations

Get Transactions Record

Operation Type:

Course Name:

Additional Info (Blank if Not Needed):

Get Records

Course Name	Exam Date	Participations
SDD_2025	06/09/25	3

Interfaccia di Monitoraggio



04

Get Transactions Record Registrazione Corso

Welcome to the Attendance System's API Interface Manage Users
Logout

Available Operations:

Count Registrations

Get Registrations

Count Lesson Attendances

Get Attendances

Count Exam Participations

Get Participations

Get Transactions Record

Get Records

Operation_Type	Course_Name	Additional_Info	Student_ID
Registrazione	SDD_2025		908309800310
Registrazione	SDD_2025		220985337382
Registrazione	SDD_2025		291579667845

Interfaccia di Monitoraggio



04

Get Transactions Record Lezione

Welcome to the Attendance System's API Interface Manage Users
Logout

Available Operations:

Count Registrations

Count Lesson Attendances

Count Exam Participations

Get Transactions Record

Operation_Type	Course_Name	Additional_Info	Student_ID
Lezione	SDD_2025	Lezione_1_Introduzione	908309800310
Lezione	SDD_2025	Lezione_1_Introduzione	220985337382
Lezione	SDD_2025	Lezione_1_Introduzione	291579667845

Interfaccia di Monitoraggio



04

Get Transactions Record Esame

Welcome to the Attendance System's API Interface Manage Users
Logout

Available Operations:

Count Registrations

Count Lesson Attendances

Count Exam Participations

Get Transactions Record

Operation_Type	Course_Name	Additional_Info	Student_ID
Esame	SDD_2025	06/09/2025	220985337382
Esame	SDD_2025	06/09/2025	291579667845
Esame	SDD_2025	06/09/2025	908309800310

Interfaccia di Monitoraggio



05

Manage Users

Manage Users

Back

Register New User

Username:

Password:

Register

Current Users

admin	<button>Remove</button>
-------	-------------------------

Interfaccia di Monitoraggio



05

Manage Users

Manage Users

Back

Register New User

CarloSorrentino

password

Register

Current Users

admin	Remove
-------	--------

Interfaccia di Monitoraggio



05

Manage Users

Manage Users

Back

Register New User

Username:

Password:

Register

Current Users

admin	<button>Remove</button>
CarloSorrentino	<button>Remove</button>

Interfaccia di Monitoraggio



Misure di Sicurezza Adottate

- **HTTPS per garantire la trasmissione sicura dei dati; di fatti i cookie di sessione siano cifrati e contrassegnati con attributi HttpOnly e Secure.**
- **JWT (JSON Web Token) sono utilizzati per autenticare gli utenti senza mantenere sessioni lato server. La protezione CSRF con JWT sui cookie garantisce che solo richieste legittime provenienti dal client autentico possano accedere alle risorse del server.**
- **CSRF (Cross-Site Request Forgery) impedisce quindi che utenti malintenzionati eseguano azioni indesiderate a nome di un utente autenticato poiché ogni richiesta che modifica lo stato del server (come un'operazione di registrazione o modifica dati) deve includere un token univoco e non prevedibile generato dal server.**
- **Hashing delle password degli Users.**
- **Le richieste AJAX nei file HTML gestiscono efficacemente gli errori, catturando le richieste fallite e mostrando messaggi di errore all'utente nel file app.log degli eventi critici tramite Loggers.**

Interfaccia di Monitoraggio



Altre Misure di Sicurezza Fondamentali

- Sanitizzazione degli input con la libreria bleach che viene utilizzata per eliminare codice potenzialmente dannoso dagli input degli utenti, prevenendo attacchi di tipo XSS (Cross-Site Scripting). Sanitizzando l'input, si evita che l'utente possa iniettare codice malevolo, come script che potrebbero essere eseguiti nel contesto della pagina web di altri utenti.
- Validazione degli input con WTForms con check dei dati lato server il quale assicura che i campi del form contengano dati validi e che rispettino regole specifiche (come lunghezza minima e massima). Questo riduce il rischio che input malevoli o formati errati vengano inviati al server.

Flusso Operativo Completo

01

Lo studente scansiona il tag attraverso
il lettore collegato al Raspberry Pi.

Flusso Operativo Completo

01

Lo studente scansiona il tag attraverso
il lettore collegato al Raspberry Pi.

02

Vengono recuperati tutti i suoi dati
personalni.

Flusso Operativo Completo

01

Lo studente scansiona il tag attraverso il lettore collegato al Raspberry Pi.

02

Vengono recuperati tutti i suoi dati personali.

03

Il Raspberry Pi cifra l'ID univoco letto dal tag.

Flusso Operativo Completo

01

Lo studente scansiona il tag attraverso il lettore collegato al Raspberry Pi.

02

Vengono recuperati tutti i suoi dati personali.

03

Il Raspberry Pi cifra l'ID univoco letto dal tag.

04

Vengono salvate le Chiavi e gli IV usati per poterli riutilizzare in futuro.

Flusso Operativo Completo

Flusso Operativo Completo

01

Lo studente scansiona il tag attraverso il lettore collegato al Raspberry Pi.

02

Vengono recuperati tutti i suoi dati personali.

03

Il Raspberry Pi cifra l'ID univoco letto dal tag.

04

Vengono salvate le Chiavi e gli IV usati per poterli riutilizzare in futuro.

05

Il dato cifrato insieme al Nome del Corso, al tipo di Operazione ed info aggiuntive vengono memorizzati sia nella BlockChain che dall'API in Google Sheet.

Flusso Operativo Completo

01

Lo studente scansiona il tag attraverso il lettore collegato al Raspberry Pi.

02

Vengono recuperati tutti i suoi dati personali.

03

Il Raspberry Pi cifra l'ID univoco letto dal tag.

04

Vengono salvate le Chiavi e gli IV usati per poterli riutilizzare in futuro.

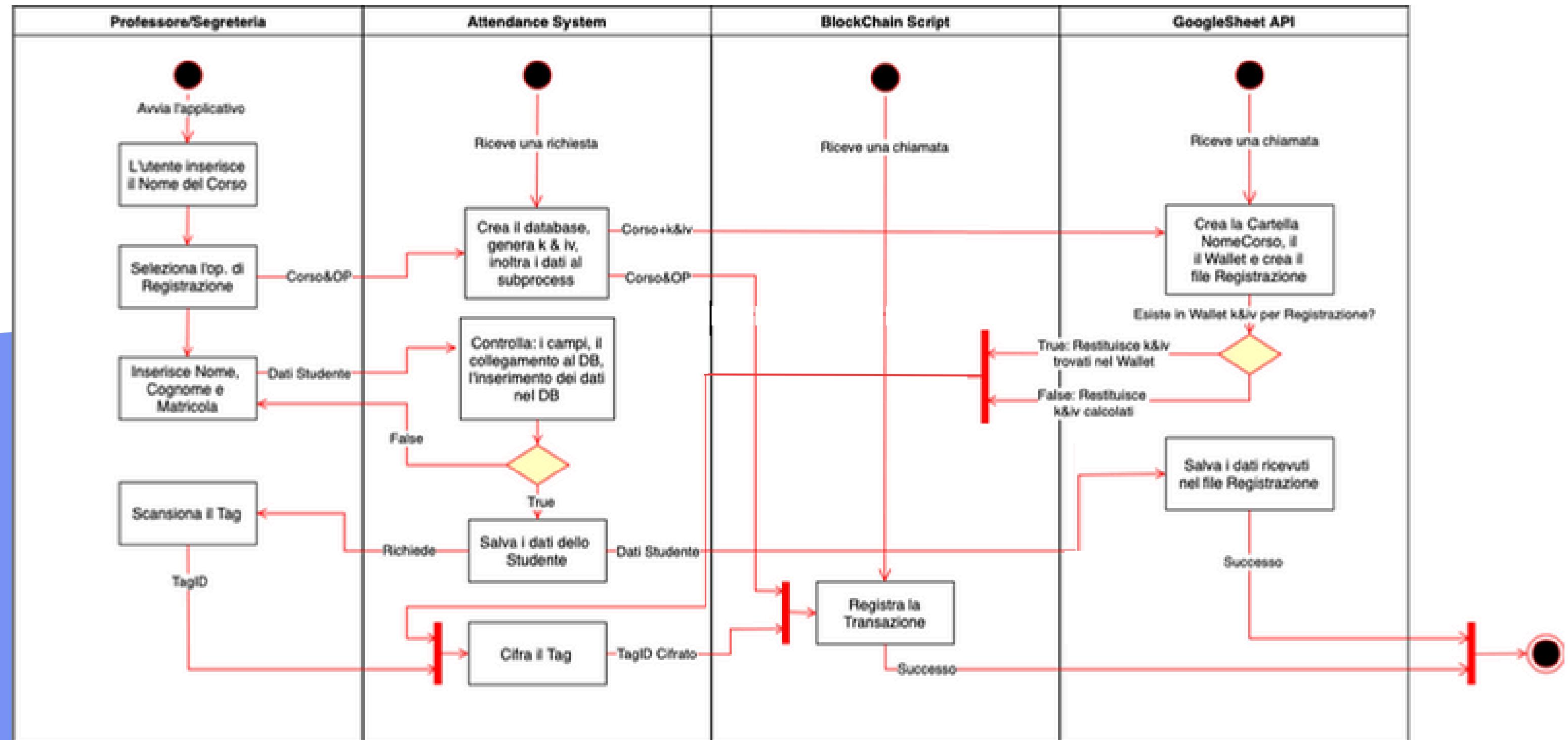
06

Questi dati in qualsiasi momento possono essere consultati sia sulla BlockChain che sul proprio Drive Google oppure anche dall'applicazione di monitoraggio in Flask.

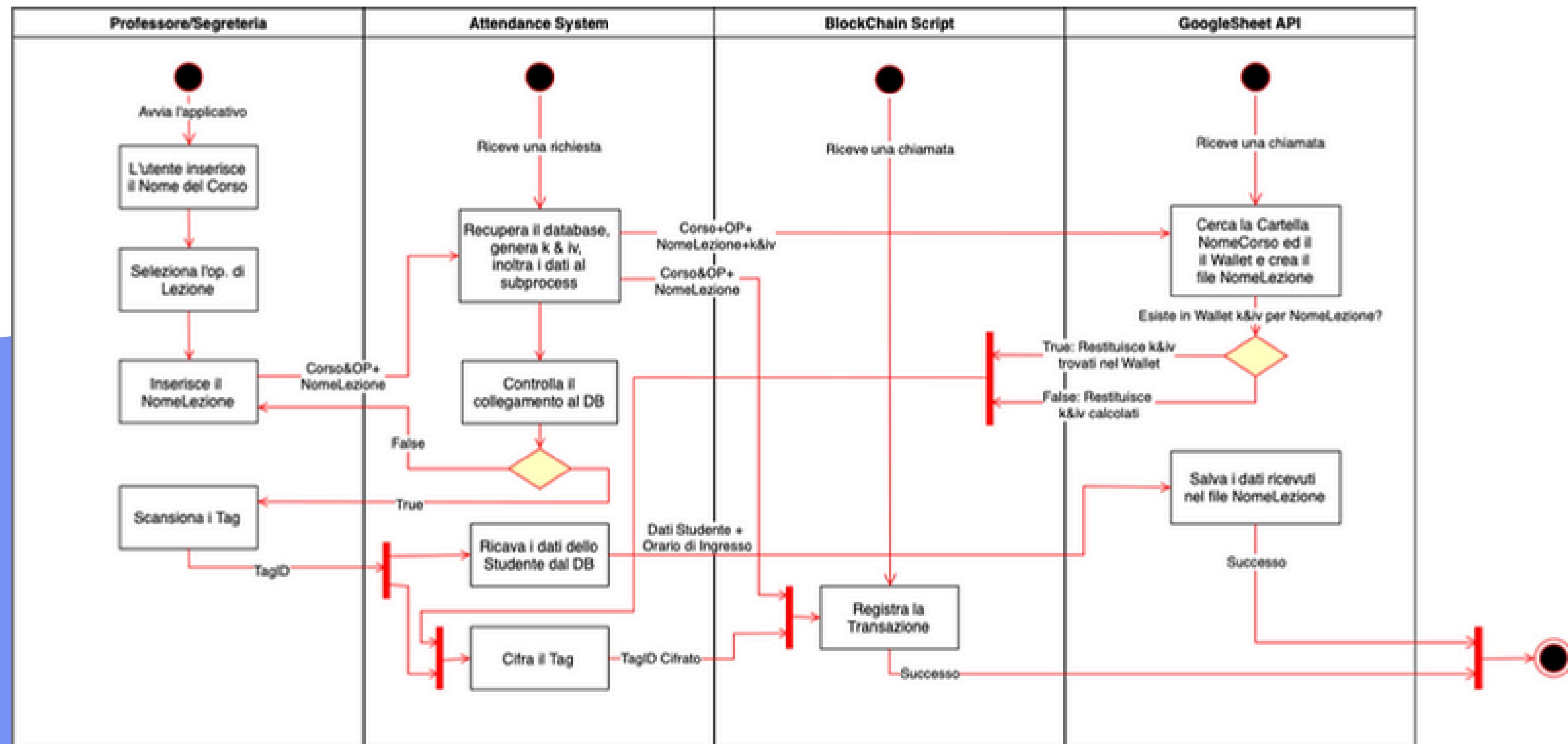
05

Il dato cifrato insieme al Nome del Corso, al tipo di Operazione ed info aggiuntive vengono memorizzati sia nella BlockChain che dall'API in Google Sheet.

Activity Diagram (Registrazione)



Activity Diagram (Lezione)



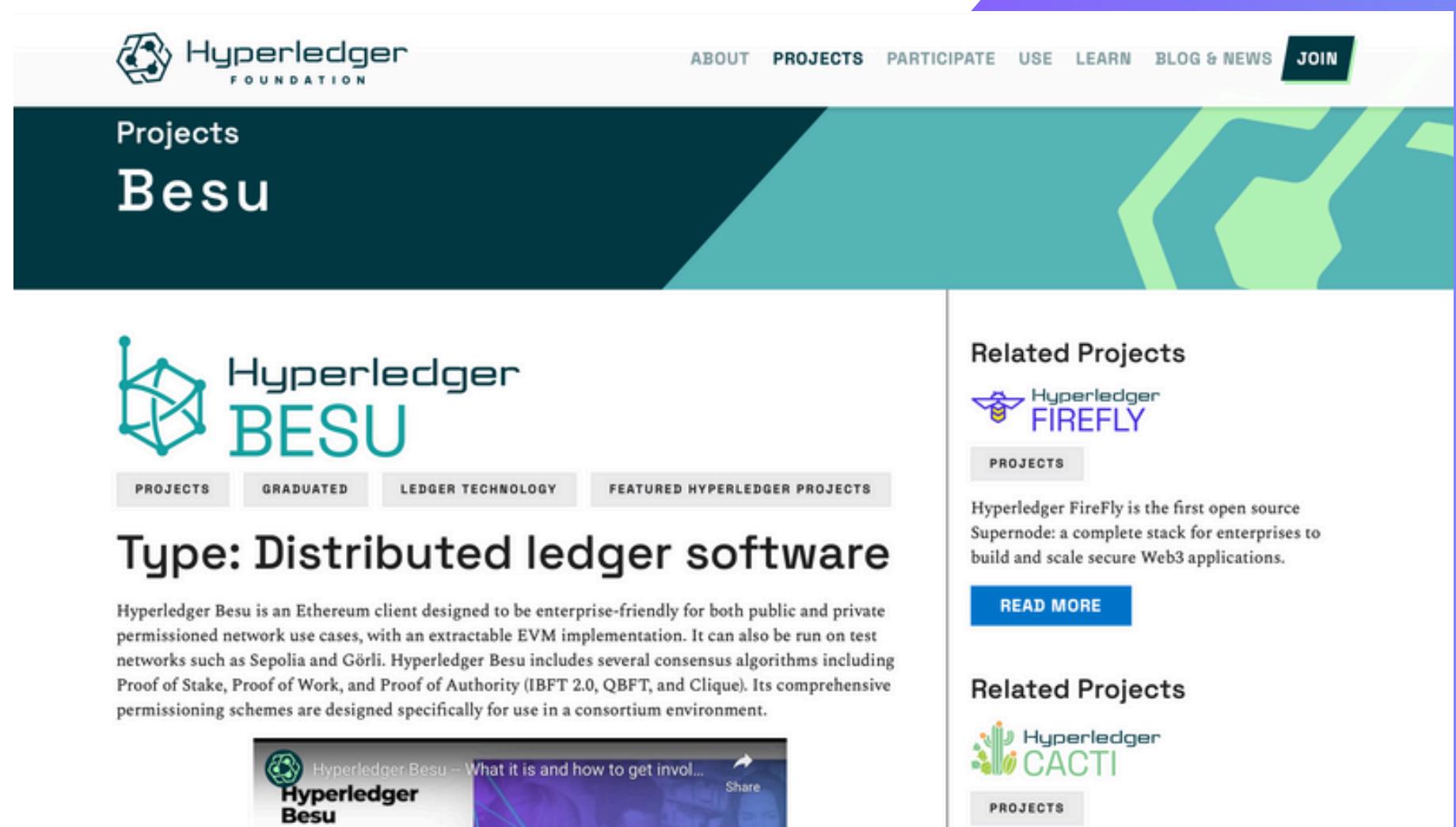
BrainStorming

Iniziale &

Conclusioni



BrainStorming Iniziale & Conclusioni



The screenshot shows the Hyperledger Besu project page. At the top, there's a navigation bar with links for ABOUT, PROJECTS, PARTICIPATE, USE, LEARN, BLOG & NEWS, and JOIN. Below the navigation is a dark teal header with the word "Projects" and the "Besu" logo. The main content area features the "Hyperledger BESU" logo with sub-sections for PROJECTS, GRADUATED, LEDGER TECHNOLOGY, and FEATURED HYPERLEDGER PROJECTS. A large heading says "Type: Distributed ledger software". Below this, a paragraph describes Hyperledger Besu as an Ethereum client designed for enterprise use, mentioning various consensus algorithms and permissioning schemes. At the bottom, there's a snippet of a blog post titled "Hyperledger Besu – What it is and how to get invol..." with a "Share" button.

Hyperledger Foundation

ABOUT PROJECTS PARTICIPATE USE LEARN BLOG & NEWS JOIN

Projects

Besu

Hyperledger BESU

PROJECTS GRADUATED LEDGER TECHNOLOGY FEATURED HYPERLEDGER PROJECTS

Type: Distributed ledger software

Hyperledger Besu is an Ethereum client designed to be enterprise-friendly for both public and private permissioned network use cases, with an extractable EVM implementation. It can also be run on test networks such as Sepolia and Görli. Hyperledger Besu includes several consensus algorithms including Proof of Stake, Proof of Work, and Proof of Authority (IBFT 2.0, QBFT, and Clique). Its comprehensive permissioning schemes are designed specifically for use in a consortium environment.

Hyperledger Besu – What it is and how to get invol... Share

Hyperledger Besu

Related Projects

Hyperledger FIREFLY

PROJECTS

Hyperledger FireFly is the first open source Supernode: a complete stack for enterprises to build and scale secure Web3 applications.

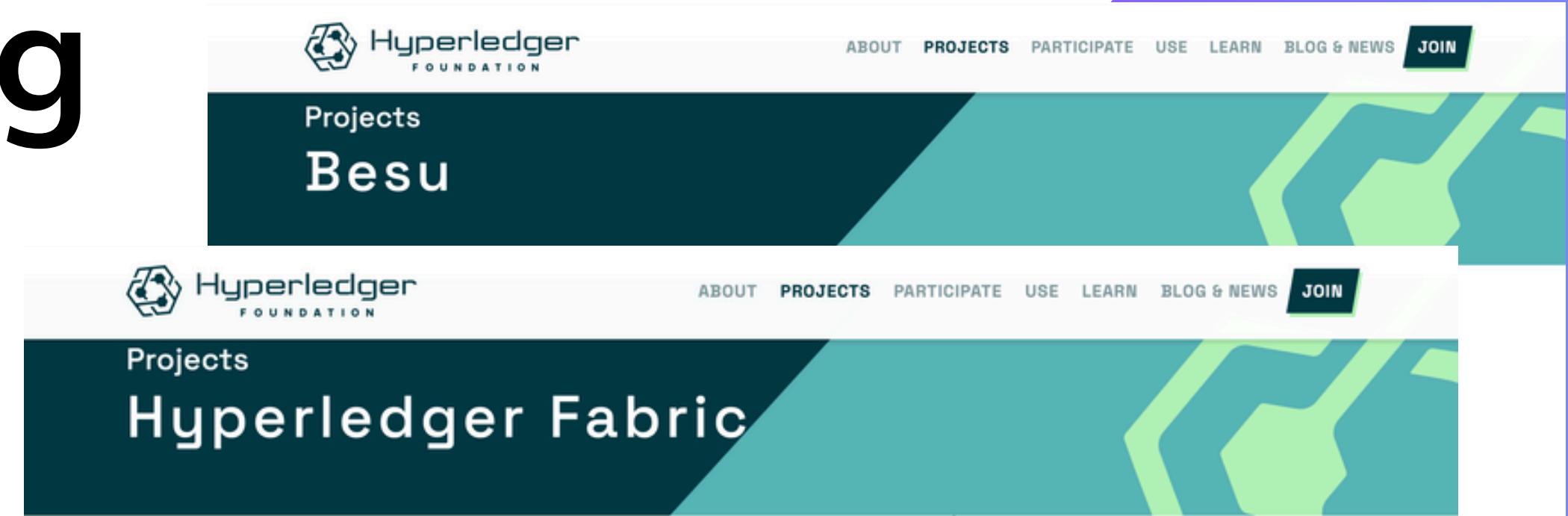
READ MORE

Related Projects

Hyperledger CACTI

PROJECTS

BrainStorming Iniziale & Conclusioni



The image displays two screenshots of the Hyperledger Foundation website. The top screenshot shows the 'Besu' project page, featuring the Hyperledger logo and navigation links for About, Projects, Participate, Use, Learn, Blog & News, and Join. The bottom screenshot shows the 'Hyperledger Fabric' project page, also with the Hyperledger logo and similar navigation links.



Hyperledger
FABRIC

PROJECTS GRADUATED LEDGER TECHNOLOGY FEATURED HYPERLEDGER PROJECTS

Type: Distributed ledger software

Hyperledger Fabric serves as the foundation for developing applications or solutions with a modular architecture. It allows interchangeable components, including consensus and membership services, enabling a plug-and-play environment.

Hyperledger Fabric is designed to meet diverse industry needs. Additionally, it offers a unique approach to consensus that facilitates scalable performance while maintaining privacy.

There are a number of other projects and labs that are related to Fabric and can support you with deploying solutions. [Check out the Fabric Ecosystem map to learn more](#)

[> Read the Datasheet](#)

Related Projects



Hyperledger
CACTI

PROJECTS

Cacti is a blockchain integration tool designed to allow users to securely integrate different blockchains.

[READ MORE](#)

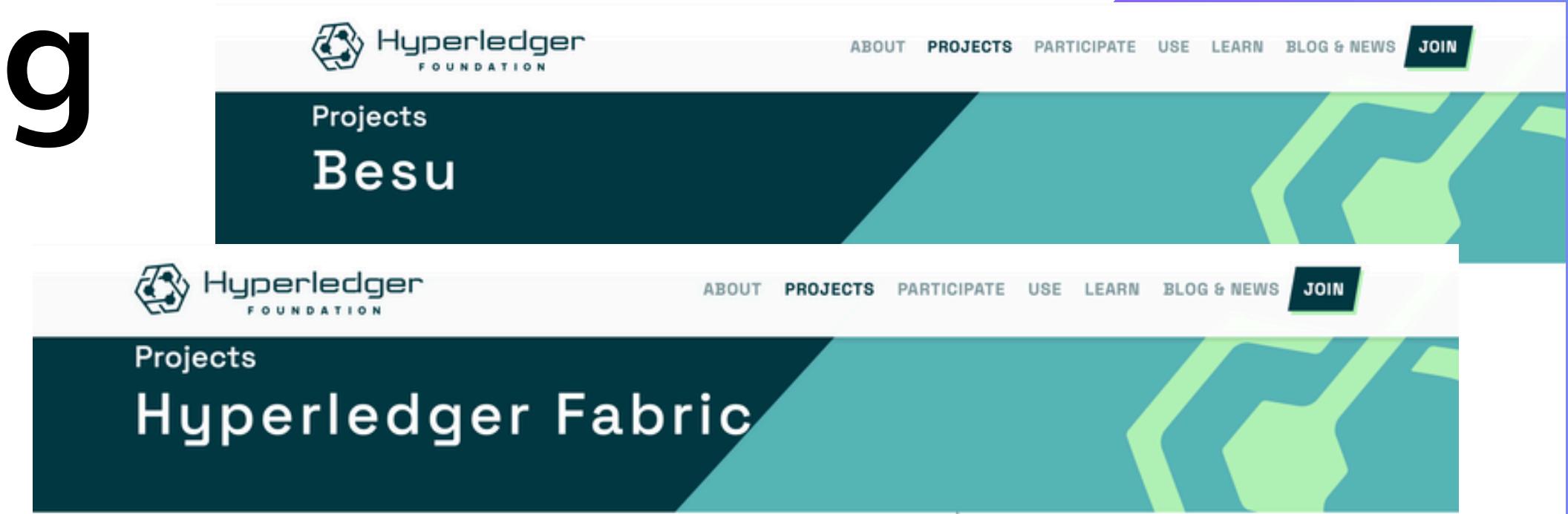
Related Projects



Hyperledger
BEVEL

PROJECTS

BrainStorming Iniziale & Conclusioni



INFURA Products Solutions Developers Resources Pricing Contact Us > Sign In > EN

Unleash the Full Potential of Web3

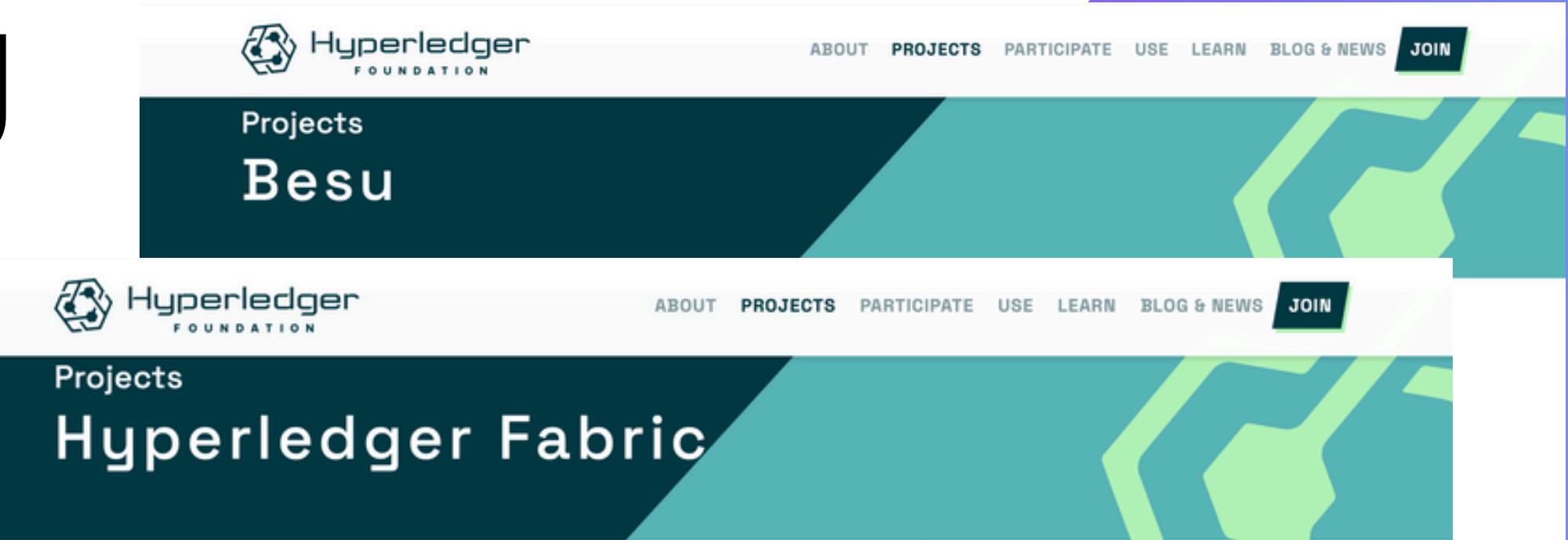
Powerful APIs, Robust Tools, and Unmatched Reliability for Seamless Web3 Development

Get Started > Access Docs →

METAMASK UNISWAP coinbase reddit

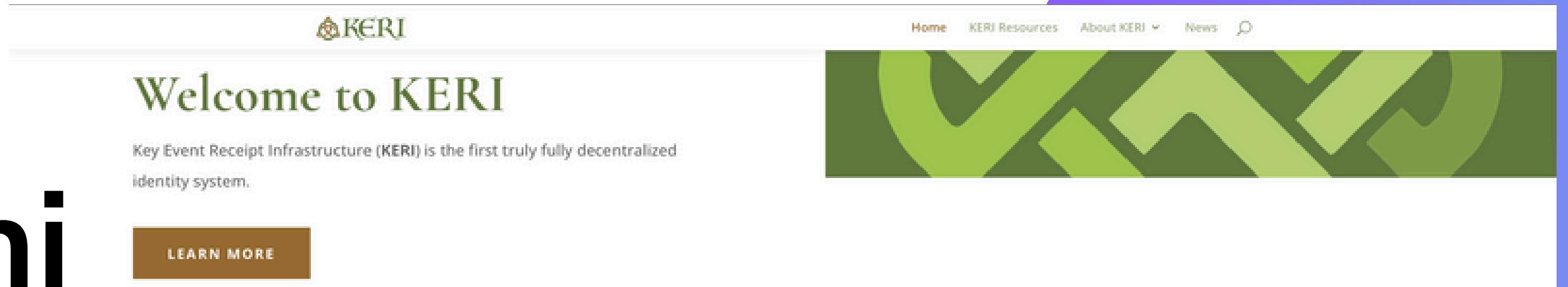
OpenZeppelin brave Compound polygon

BrainStorming Iniziale & Conclusioni

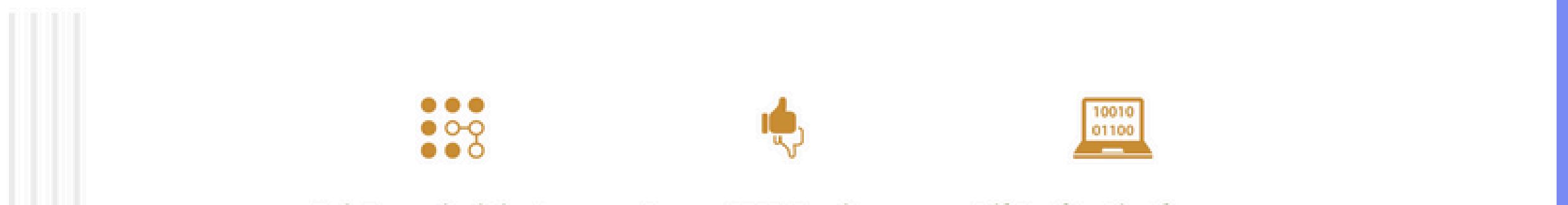


The image displays two blockchain platform websites. The top part shows the Infura website, which has a dark background with a red and orange gradient. It features the Infura logo, a navigation bar with 'Products', 'Solutions', 'Developers', 'Resources', and 'Pricing', and links for 'Contact Us', 'Sign In', and 'EN'. The bottom part shows the Kaleido website, which has a blue background. It features the Kaleido logo, a navigation bar with 'Products', 'Solutions', 'Pricing', 'Developers', 'Resources', and 'Company', and links for 'Log In' and 'Try It Free'. The main content area for Kaleido includes the text 'Blockchain, Digital Assets & Tokenization Radically Simple. Enterprise-Grade.' and 'Build transformative solutions for ESG & Climate CBDCs'. It also features a 'Try It Free' button and a 'Talk to an Expert' button. A photograph of wind turbines is visible in the background. At the bottom of the Kaleido page, there is a row of logos for various partners and clients, including T-Mobile, Nationwide, Swift, McDonald's, pepsi, BNP PARIBAS, SONY, SOCIETE GENERALE, and others. A small circular icon with a person symbol is located in the bottom right corner.

BrainStorming Iniziale & Conclusioni



The screenshot shows the KERI website's main page. At the top right, there is a navigation bar with links for Home, KERI Resources, About KERI, News, and a search icon. The main content area features the KERI logo and the heading "Welcome to KERI". Below the heading, a text block states: "Key Event Receipt Infrastructure (KERI) is the first truly fully decentralized identity system." A brown rectangular button labeled "LEARN MORE" is positioned below the text.

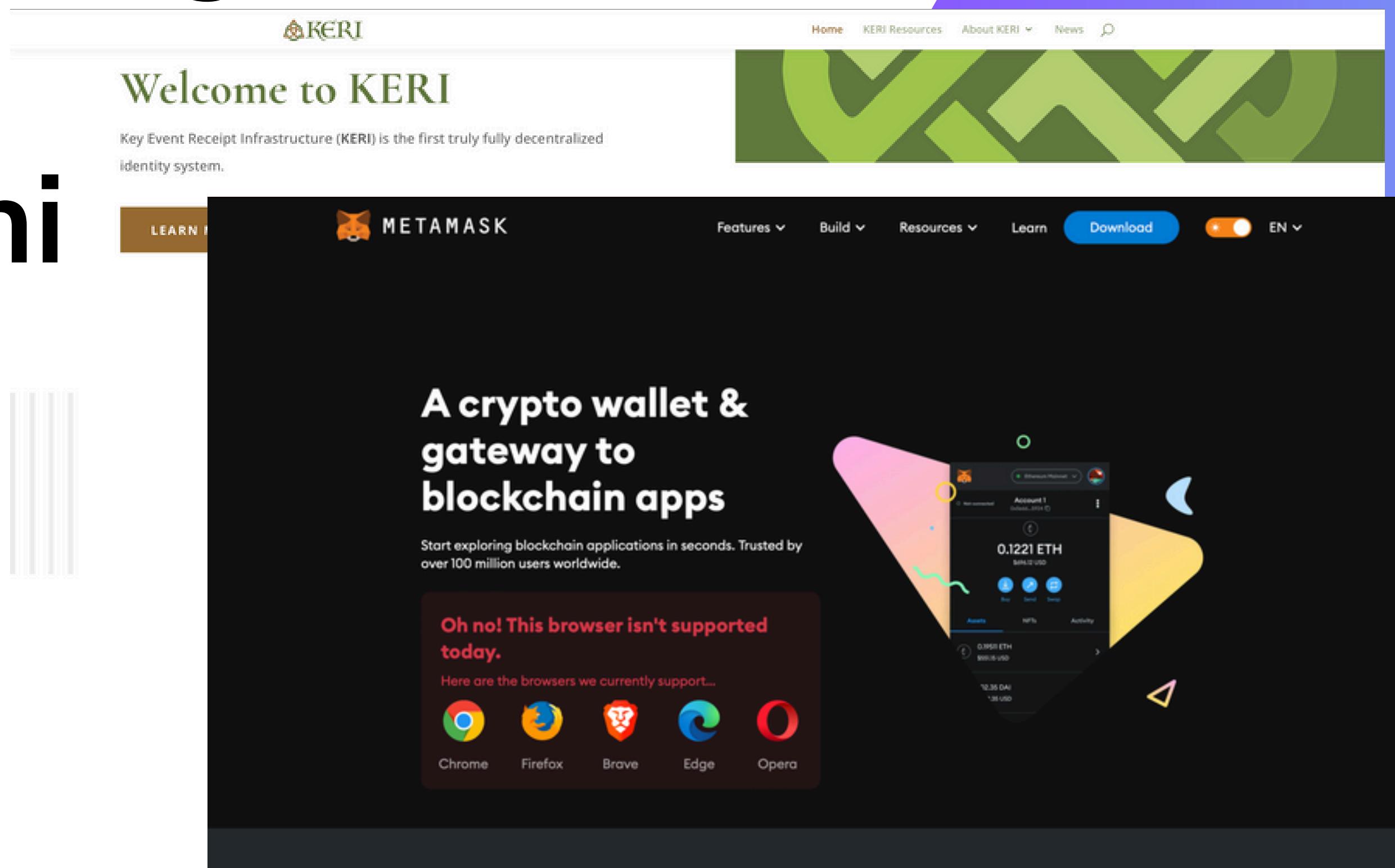


The page contains four distinct sections, each with an orange icon and descriptive text:

- Truly Decentralized Identity**: An orange icon consisting of a 3x3 grid of circles. The text explains that KERI is a ledger-less system that is non-intertwined and portable.
- Supports GDPR Compliance**: An orange icon of a hand giving a thumbs up. The text states that KERI supports GDPR compliance by being inherently supportive of global data protection rights.
- Self-Certifying Identifiers**: An orange icon showing a binary code sequence (10010 01100) inside a small computer monitor-like screen. The text describes KERI's decentralized secure root-of-trust based on cryptographic self-certifying identifiers.

BrainStorming

Iniziale & Conclusioni



The image shows the homepage of the Metamask wallet. At the top, there's a navigation bar with links for Home, KERI Resources, About KERI, News, and a search icon. Below the navigation is a large green decorative banner with abstract geometric shapes. The main content area features the KERI logo and the text "Welcome to KERI". It explains that KERI is the first truly decentralized identity system. There are buttons for "LEARN", "METAMASK", and "Download". A dark sidebar on the right contains links for Features, Build, Resources, Learn, and a language switcher set to EN. The central part of the page has a dark background with a colorful gradient overlay. It features the text "A crypto wallet & gateway to blockchain apps" and a message about browser support. It lists supported browsers: Chrome, Firefox, Brave, Edge, and Opera. To the right, there's a screenshot of the Metamask interface showing a wallet balance of 0.1221 ETH and 0.0958 DAI.

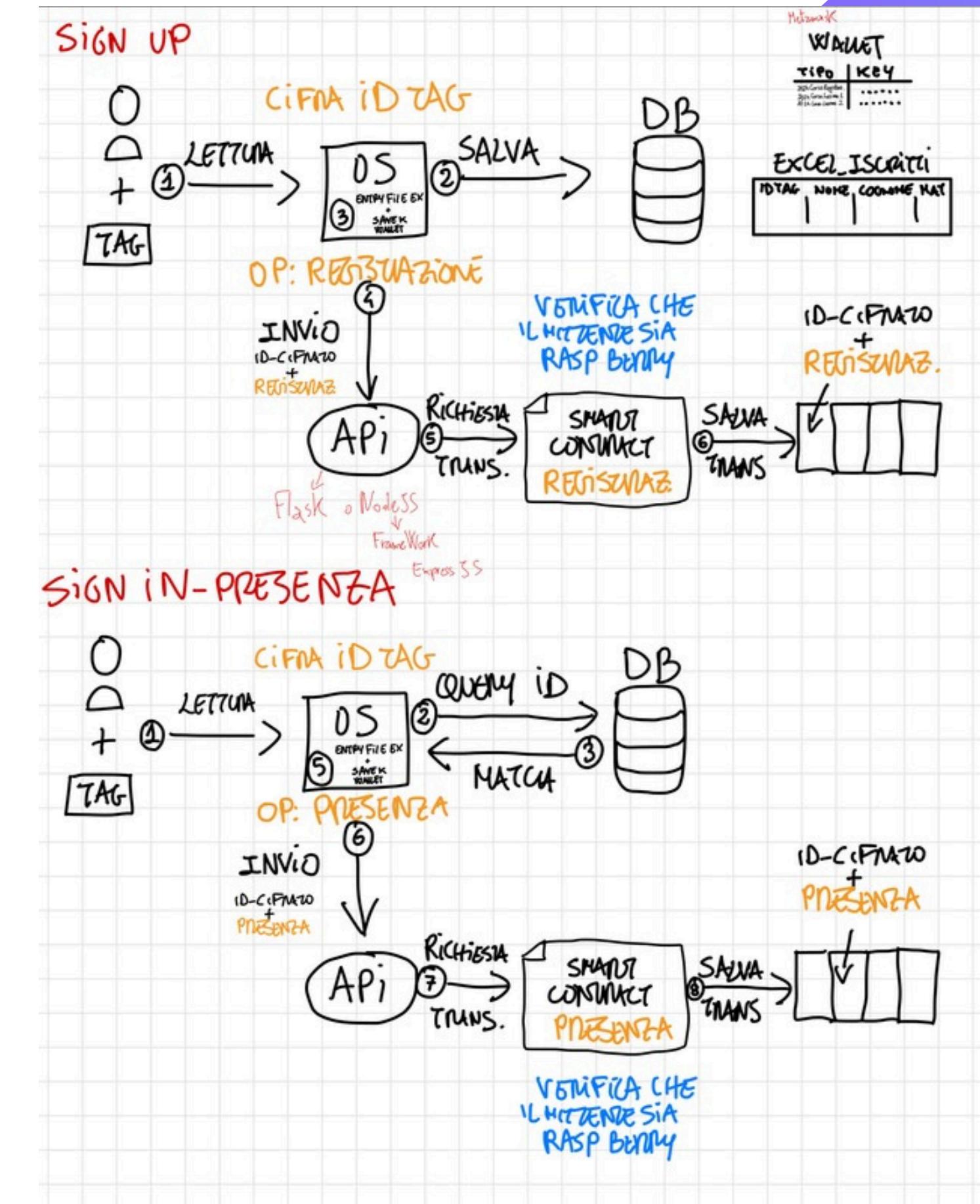
BrainStorming

Iniziale & Conclusioni

The image displays three screenshots from blockchain-related websites:

- KERI:** The homepage features the title "Welcome to KERI" and a subtext: "Key Event Receipt Infrastructure (KERI) is the first truly fully decentralized identity system." It includes a "LEARN" button, the Metamask logo, and a navigation bar with "Home", "KERI Resources", "About KERI", "News", and a search icon.
- MetaMask:** The interface shows the Metamask logo and a navigation bar with "Features", "Build", "Resources", "Learn", "Download", and a language switcher.
- Keplr:** The homepage features the title "Your Interchain Gateway" and a subtext: "Introducing Keplr, the fast, simple, secure wallet that plugs you into any blockchains and apps wherever you go. Pioneering its ways in the multichain future from day one." It includes a "Get Keplr" button, a user count of "1M+ Users", and a social media icon.

BrainStorming Iniziale & Conclusioni





Sviluppi Futuri



Sviluppi Futuri



Sviluppi Futuri



Eventuali Ottimizzazioni & Integrazioni



Chi è il mittente?

**Eventuali Ottimizzazioni
& Integrazioni**



Chi è il mittente? E come lo identifico?

**Eventuali Ottimizzazioni
& Integrazioni**

Aggiungere la firma del dispositivo alla transazione

Chi è il mittente? E come lo identifico?



**Eventuali Ottimizzazioni
& Integrazioni**

BlackList / WhiteList / Access Control List

Aggiungere la firma del dispositivo alla transazione

Chi è il mittente? E come lo identifico?



**Eventuali Ottimizzazioni
& Integrazioni**

BlackList / WhiteList / Access Control List

Verifica tramite IP o MAC Address

Aggiungere la firma del dispositivo alla transazione

Chi è il mittente? E come lo identifico?



**Eventuali Ottimizzazioni
& Integrazioni**

BlackList / WhiteList / Access Control List

Verifica tramite IP o MAC Address

Verifica tramite TLS/SSL con Certificato Client

Chi è il mittente? E come lo identifico?



Aggiungere la firma del dispositivo alla transazione

**Eventuali Ottimizzazioni
& Integrazioni**



BlackList / WhiteList / Access Control List

Verifica tramite IP o MAC Address

Aggiungere la firma del dispositivo alla transazione

Verifica tramite TLS/SSL con Certificato Client

Chi è il mittente? E come lo identifico?

Two-Factor Authentication (2FA) o Token OTP

**Eventuali Ottimizzazioni
& Integrazioni**



Obiettivo... Superare l'Obsolescenza.

MIGLIORARE L'ACCURATEZZA E L'EFFICIENZA DELLA REGISTRAZIONE DELLE PRESENZE,
SUPERANDO IL METODO CARTACEO ANTICO, GARANTENDO ALLO STESSO TEMPO
LA SICUREZZA DEI DATI E FACILITANDO L'ACCESSO E LA GESTIONE DELLE INFORMAZIONI.

Obiettivo... Superare l'obsolescenza.

MIGLIORARE L'ACCURATEZZA E LA RELIABILITÀ DELLA REGISTRAZIONE DELLE PRESENZE,
SUPERANDO IL METODO DI REGISTRAZIONE ANTICO, GARANTENDO ALLO STESSO TEMPO
LA SICUREZZA DEI DATI E FAR SI CHE IL SISTEMA GARANTISCA L'ACCESSO E LA GESTIONE DELLE INFORMAZIONI.

Smart-Tag Attendance System

Grazie per la vostra
attenzione!

PROGETTO COMBINATO PER I CORSI DI
SICUREZZA DEI DATI & IOT SECURITY

ANNO ACCADEMICO 2023/2024.

