



EX<sup>change</sup>  
tensions

COINsolidation.

Tallinn, Estonia. (E-residenza)

# White Paper - Libro bianco.

versione 1.0.0

Dicembre 2020.

COINsolidation.org è un marchio registrato di COINsolidation International, sotto licenza d'uso gratuito e commerciale. Termini e condizioni d'uso su: [www.Coinsolidation.org](http://www.Coinsolidation.org)

COINsolidation International si è fusa con [www.OpenQbit.com](http://www.OpenQbit.com) per la cooperazione tecnologica basata sulla meccanica quantistica (Quantum Security & Quantum Computing). Questa fusione permette l'uso, la condivisione e la reingegnerizzazione della tecnologia sviluppata da OpenQbit Inc. (Estonia, E-residenza).

## Contenuto

1. Introduzione.....	3
2. Sicurezza Quantum Computing.....	6
3. Creazione di un dispositivo "Hardware" di un QRNG (Quantum Random Number Generator). 11	
4. Che cos'è la Prova del Quantum (PQu)? .....	17
5. Algoritmo per la creazione di un indirizzo universale consolidato (CUA) .....	20
6. Algoritmo per il doppio indirizzo consolidato (DAC) e (HAC).....	21
Progetto e soluzione di COINsolidation. ....	23
7. Creazione di App CUA (Consolidated Universal Address) in 15 minuti.....	24
8. Crea il tuo cambio valuta cripto Ethereum su Android in soli 15 minuti. ....	28
9. Roadmap COINsolidamento. ....	31
10. Gettone di consolidamento COIN (CUAG) - PIANO DI DISTRIBUZIONE ICO.....	32
11. Caratteristiche generali del gettone di consolidamento COIN: .....	33
12. Concetti di base applicati nelle piattaforme Blockchain.....	34
13. Cos'è la programmazione Blockly?.....	37
14. Allegato "Codice per l'algoritmo CUA".....	37
15. Termini e condizioni. ....	37

## 1. Introduzione.

Attualmente, le fusioni sono aggiornate, sia che si tratti di un bene economico, tecnologico o di mercato.

Vi presentiamo il primo modello di crypto fusion o crypto-tokens che offre un backup tra due crittomonadi, token o una miscela di questi, basato su un algoritmo per creare un indirizzo consolidato che viene utilizzato e generato nell'ambiente di COINsolidation.

Abbiamo creato tre tipi di indirizzi consolidati.

Il **CUA** (Consolidated Universal Address) viene utilizzato per consolidare e creare un nuovo token (attivo) da utilizzare da parte dell'utente. La combinazione può essere di tre tipi: Cryptocurrency-Cryptocurrency, Cryptocurrency-Token o Token-Token. Nel caso della CUA è formata da un rapporto Token-Token.

L'**HAC** (Hibric Address Consolidated) viene utilizzato quando abbiamo bisogno di consolidare un indirizzo relativo a una valuta criptata e/o un token e un indirizzo normale per il trasferimento di attività.

Il **DAC** (Dual Address Consolidated) è utilizzato per gestire e consolidare due indirizzi normali dalla stessa catena di blocco o da due tecnologie diverse.

Cominciamo a considerare i vantaggi della CUA.

Un indirizzo CUA è costituito dall'indirizzo del token di consolidamento COIN (indirizzo statico) e da un ulteriore token denominato "Moneta colorata" (indirizzo variabile). In questo caso possiamo vedere che gli indirizzi CUA saranno sempre formati da indirizzi di qualche tipo di combinazione di asset (Cryptosolids o tokens).

Nel nostro caso, quando consolideremo il token di consolidamento COIN e un token OAP lo conosceremo come "CUA genesis" o **CUAG (Consolidated Universal Address Genesis)**.

El token COINsolidation esta creado en el *blockchain Ethereum* y usa el standard ERC20 (Ethereum Request for Comments 20).

Il gettone "Moneta colorata" è basato e creato dalla *catena a blocchi Bitcoin* e utilizza lo standard **OAP** (Open Assest Protocol).

Cominciamo a esaminare quali sono le potenzialità e i vantaggi del consolidamento degli indirizzi.

- I. Per gli utenti che creano un CUA sarà possibile creare un token (**OAP**) che potrà essere personalizzato dall'utente che ha creato il CUA, l'utente avrà la possibilità di avere il proprio token o crypto attivo in modo da poterlo utilizzare per la creazione, il

supporto o l'espansione del proprio business(i), in modo semplice e facile avrà un asset nel mondo dei crypto-tokens.

- II. Per le aziende che creano una CUA possono avere un token **(OAP)** che possono utilizzare per creare valore nella loro catena di fornitura o utilizzare l'attività in operazioni di liquidità sulla base del supporto economico delle loro attività e passività aziendali.
- III. Per i crittomoniadi e i token esistenti, creando un CUA, potranno utilizzare il vostro indirizzo che identifica il vostro asset e consolidandolo con il token **(OAP)** potranno far crescere la loro domanda offrendo ai loro investitori attuali e futuri il proprio token per i loro utenti.

Esempio di **CUAG**, abbiamo i rispettivi indirizzi di due differenti Blockchain:

Indirizzo Bitcoin- Token - (OAP).

**akXma4vqxvmEqnVAKSM953wYsnjNBhN3GM7**

Indirizzo Ethereum - Token COINsolidation - (ERC20).

**0x8390f8abb8fd8ad3bf8457db59f2ed75e015d303**

Applicando un algoritmo per consolidare gli indirizzi precedenti si ottiene l'indirizzo CUA.

**cua50d0615d303k8X3m9a04fv8qaxbvb8EfQdn8VaAdK3SbMf985435w7Ydsbn5j9NfB2heNd37G5Me70**

\* Per maggiori dettagli sull'algoritmo si rimanda alla sezione 7.- "Algoritmo per la creazione di un indirizzo universale consolidato".

Abbiamo di conseguenza un'unica direzione che rappresenta due diverse tecnologie da due diverse direzioni consolidate in un'unica direzione.

Riflettiamo questo nel campo della redditività e dell'espansione finanziaria in modo semplice e diretto investendo in uno dei token che integra il nostro CUA immediatamente si otterrà un token basato sulla catena di blocco Bitcoin (OAP).

Vediamo ora due linee guida che proponiamo anche in COINsolidation per il mondo dei criptomontaggi e/o tokens.

COINsolidation token è il progetto per consolidare gli indirizzi e avere un supporto immediato per ottenere un token personalizzato da utilizzare nella crescita di ogni utente nel mondo dei crittomontaggi. Il progetto è nato nel 2018 con un gruppo di ingegneri e finanziatori interessati a fondere i settori finanziario e tecnologico, per sfruttare i fondi di investimento e occuparsi di tecnologie innovative come il calcolo quantistico per dare sicurezza ai beni, nonché lo scopo di utilizzare strumenti che possano essere a disposizione di tutti.

Dopo aver valutato diverse possibilità di sviluppo abbiamo scelto l'opzione della metodologia di programmazione visiva Blockly, che si basa sull'utilizzo di estensioni o moduli (programmi in linguaggio di programmazione java) con funzionalità semplici ma potenti per espandere il business in asset crypto per qualsiasi persona, per raggiungere questo obiettivo dobbiamo coprire i seguenti punti:

- ✓ a.- ROI finanziario immediato per gli utenti, gli investitori e i beni, grazie alla possibilità di creare un bene non tangibile (gettoni personali) ad uso esclusivo del creatore e dell'utente del (CUA)
- ✓ b.- Utilizziamo i vantaggi dell'unione di due catene di blocco a scelta dell'utente per far crescere gli investimenti attuali e futuri nel mercato crypto attivo utilizzando il (CUA).
- ✓ c.- Facilitare la gestione di indirizzi separati consolidandoli in (DAC).
- ✓ d.- Creare e utilizzare la sicurezza basata sul Quantum Computing.

La sfida è iniziata nella creazione della tecnologia delle estensioni sufficientemente modulari nelle funzionalità e nelle "dimensioni" quest'ultima è stata la sfida del team di sviluppo di COINsolidation poiché le estensioni che vengono utilizzate nella metodologia Blockly e sistemi di questo tipo (AppInventor, AppyBuilder, Thunkable, Kondular, etc) sono solitamente estensioni (programmi) create che non superano i 100k - 300k byte, con le restrizioni che hanno nella loro dimensione il compito di creare estensioni per l'uso nella Blockchain corrente erano virtualmente impossibili a causa delle librerie che vengono utilizzate nella loro creazione superano tra i 10MB e i 35MB queste dimensioni per gli attuali strumenti I sistemi Blockly non sono funzionali all'uso.

Il team ha dovuto creare, adattare e minimizzare la metodologia di programmazione e le librerie per ottenere le estensioni con le funzionalità, la sicurezza e le dimensioni ottimali.

Dopo quasi due anni di sviluppo e test abbiamo finito la prima catena di blocchi "beta" utilizzando estensioni per Blockly, incluso l'algoritmo di consenso "Proof of Quantum" che utilizza la sicurezza quantistica per lo scambio di critomonia.

Attualmente abbiamo una blockchain proprietaria che è stata rilasciata per il test "Beta" ed entro la fine del 2021 rilasceremo la versione di produzione per la distribuzione delle informazioni. Attualmente il nostro COINsolidation Token si basa sulle catene a blocchi Ethereum e Bitcoin, quest'ultimo per la creazione di gettoni personalizzati per gli utenti.

## 2. Sicurezza Quantum Computing.

### Come funziona il calcolo quantistico? <sup>(2)</sup>

La trasformazione digitale sta portando il mondo a un cambiamento più rapido che mai: ci credereste che l'era digitale sta per finire? **L'alfabetizzazione digitale** è già stata identificata come un'area in cui la conoscenza aperta e le opportunità accessibili di apprendere la tecnologia sono urgenti per affrontare le lacune nello sviluppo sociale ed economico. Imparare dai concetti chiave dell'era digitale diventerà ancora più critico con l'imminente arrivo di un'altra nuova ondata tecnologica in grado di trasformare i modelli esistenti con velocità e potenza sorprendenti: **le tecnologie quantistiche**.

In questo articolo confrontiamo i concetti di base del calcolo tradizionale e del calcolo quantistico; e iniziamo anche ad esplorare la loro applicazione in altre aree correlate.

Cosa sono le tecnologie quantistiche?

Nel corso della storia, gli esseri umani hanno sviluppato la tecnologia così come hanno capito come funziona la natura attraverso la scienza. Tra il 1900 e il 1930, lo studio di alcuni fenomeni fisici non ancora ben compresi ha dato origine ad una nuova teoria fisica, la **Meccanica Quantistica**. Questa teoria descrive e spiega il funzionamento del mondo microscopico, l'habitat naturale di molecole, atomi o elettroni. Grazie a questa teoria, non solo è stato possibile spiegare questi fenomeni, ma è stato anche possibile comprendere che la realtà subatomica funziona in modo completamente contro-intuitivo, quasi magico, e che nel mondo microscopico avvengono eventi che non si verificano nel mondo macroscopico.

Queste **proprietà quantistiche** comprendono la sovrapposizione quantistica, l'entanglement quantistico e il teletrasporto quantistico.


- La **sovrapposizione quantistica** descrive come una particella può trovarsi in diversi stati contemporaneamente.
- L'**entanglement quantistico** descrive come due particelle così distanti tra loro possano essere correlate in modo tale che, interagendo con una, l'altra ne sia consapevole.
- Il **teletrasporto quantistico** usa l'entanglement quantistico per inviare informazioni da un luogo ad un altro nello spazio senza doverlo attraversare.

Le tecnologie quantistiche si basano su queste proprietà quantistiche di natura subatomica.

In questo caso, oggi la comprensione del mondo microscopico attraverso la Meccanica Quantistica ci permette di inventare e progettare tecnologie in grado di migliorare la vita delle persone. Ci sono molte e molto diverse tecnologie che utilizzano i fenomeni quantistici e alcune di esse, come i laser o la risonanza magnetica (MRI), sono con noi da più di mezzo secolo. Tuttavia, stiamo attualmente assistendo ad una rivoluzione tecnologica in settori

quali il calcolo quantistico, l'informazione quantistica, la simulazione quantistica, l'ottica quantistica, la metrologia quantistica, gli orologi quantistici o i sensori quantistici.

Cos'è il calcolo quantistico? Per prima cosa, bisogna capire l'informatica classica.



**FIGURA 1.**  
Ejemplos de  
caracteres  
en lenguaje  
binario.

Caracter	Bits
7	111
A	01000001
\$	00100100
:)	0011101000101001

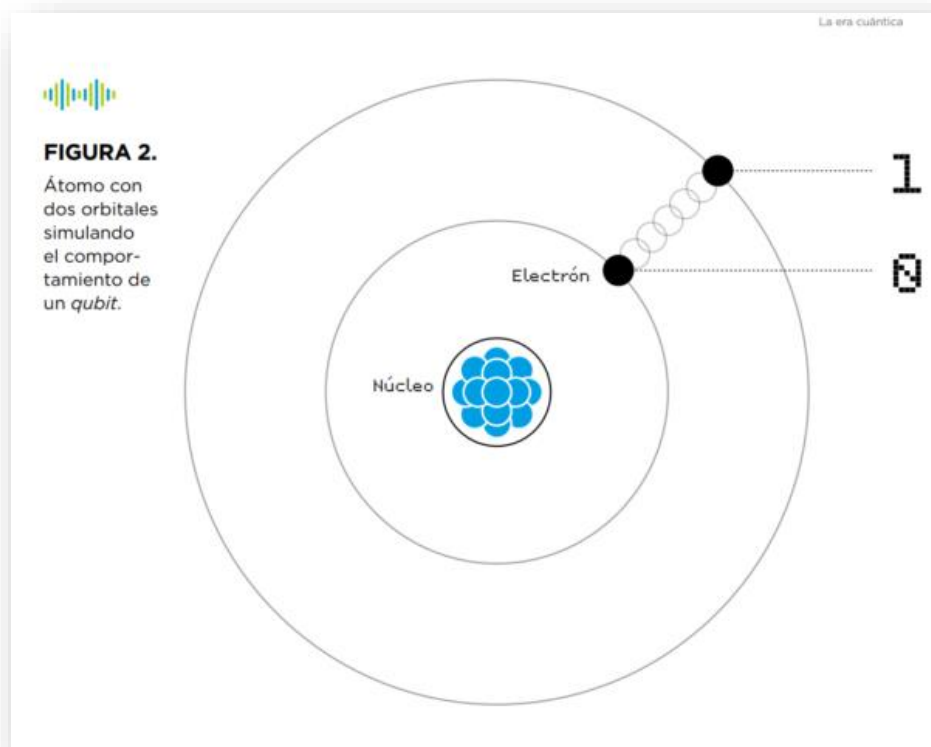
Per capire come funzionano i computer quantistici, è conveniente spiegare prima di tutto come funzionano i computer che usiamo ogni giorno, che in questo documento chiameremo computer digitali o classici. Questi, come il resto dei dispositivi elettronici come i tablet o i telefoni cellulari, utilizzano i bit come unità fondamentali della memoria. Ciò significa che i programmi e le applicazioni sono codificati in bit, cioè in linguaggio binario di zeri e uno. Ogni volta che interagiamo con uno di questi dispositivi, ad esempio, premendo un tasto sulla tastiera, si creano, distruggono e/o modificano stringhe di zeri e uno all'interno del computer.

La domanda interessante è: quali sono questi zeri e quali sono fisicamente all'interno del computer? Gli stati zero e uno corrispondono alla corrente elettrica che circola, o meno, attraverso parti microscopiche chiamate transistor, che fungono da interruttori. Quando non scorre corrente, il transistor è "spento" e corrisponde al bit 0, e quando scorre è "acceso" e corrisponde al bit 1.

Più semplicemente, è come se i bit 0 e 1 corrispondessero a fori, in modo che un foro vuoto sia un bit 0 e un foro occupato da un elettrone sia un bit 1. Per questo motivo questi apparecchi sono chiamati elettronici. Come esempio, la figura 1 mostra la scrittura binaria di alcuni caratteri. Ora che abbiamo un'idea di come funzionano i computer di oggi, cerchiamo di capire come funzionano i quantum.

## Da bit a qubits

L'unità fondamentale dell'informazione nel calcolo quantistico è il bit quantistico o qubit. I Qubit sono, per definizione, sistemi quantistici a due livelli - vedremo qui degli esempi - che, come i bit, possono essere a basso livello, che corrisponde ad uno stato di bassa eccitazione o di energia definito come 0, o ad alto livello, che corrisponde ad uno stato di eccitazione superiore o definito come 1. Tuttavia, e qui sta la differenza fondamentale con il calcolo classico, i qubit possono anche trovarsi in uno qualsiasi degli infiniti stati intermedi tra 0 e 1, come ad esempio uno stato che è metà 0 e metà 1, o tre quarti di 0 e un quarto di 1.



Algoritmi quantistici, calcolo esponenzialmente più potente ed efficiente

Lo scopo dei computer quantistici è quello di sfruttare queste proprietà quantistiche dei *qubit*, come sistemi quantistici che sono, al fine di eseguire algoritmi quantistici che utilizzano la sovrapposizione e l'interleaving per fornire una potenza di elaborazione molto maggiore rispetto ai classici. È importante sottolineare che il vero cambiamento di paradigma non consiste nel fare la stessa cosa che fanno i computer digitali o classici -quelli attuali- ma più velocemente, come si può leggere in molti articoli, ma che gli algoritmi quantistici permettono di eseguire certe operazioni in un modo totalmente diverso che in molti casi si rivela più efficiente - cioè in molto meno tempo o utilizzando molto meno risorse computazionali.



Vediamo un esempio concreto di ciò che questo comporta. Immaginiamo di essere a Bogotá e vogliamo sapere qual è il percorso migliore per arrivare a Lima tra un milione di opzioni per arrivarci ( $N=1.000.000$ ). Per utilizzare i computer per trovare il percorso ottimale dobbiamo digitalizzare 1.000.000 di opzioni, il che implica la loro traduzione in linguaggio bit per il computer classico e in *qubit per il computer* quantistico. Mentre un computer classico dovrebbe andare uno ad uno analizzando tutti i percorsi fino a trovare quello desiderato, un computer quantistico sfrutta il processo noto come parallelismo quantistico che gli permette di considerare tutti i percorsi contemporaneamente. Ciò implica che, mentre il computer classico ha bisogno dell'ordine di  $N/2$  passi o iterazioni, cioè 500.000 tentativi, il computer quantistico troverà il percorso ottimale dopo solo  $\sqrt{N}$  operazioni sul registro, cioè 1.000 tentativi.

Nel caso precedente il vantaggio è quadratico, ma in altri casi è addirittura esponenziale, il che significa che con  $n$  *qubit* si può ottenere una capacità di calcolo equivalente a  $2^n$  bit. Per esemplificare questo, è comune contare che con circa 270 qubit potremmo avere più stati di base in un computer quantistico - più stringhe di caratteri diversi e simultanei - rispetto al numero di atomi nell'universo, che è stimato intorno ai  $10^{80}$ . Un altro esempio è che si stima che con un computer quantistico tra 2000 e 2500 *qubit* potremmo rompere praticamente tutta la crittografia utilizzata oggi (la cosiddetta crittografia a chiave pubblica).

Perché è importante conoscere la tecnologia quantistica?

Siamo in un momento di trasformazione digitale in cui diverse tecnologie emergenti come blockchain, intelligenza artificiale, droni, Internet delle cose, realtà virtuale, 5G, stampanti 3D, robot o veicoli autonomi sono sempre più presenti in molteplici campi e settori. Queste tecnologie, chiamate a migliorare la qualità della vita dell'essere umano accelerando lo sviluppo e generando un impatto sociale, avanzano oggi in modo parallelo. Solo raramente vediamo aziende che sviluppano prodotti che sfruttano combinazioni di due o più di queste tecnologie, come blockchain e IoT o droni e intelligenza artificiale. Sebbene siano destinati a convergere, generando così un impatto esponenzialmente maggiore, la fase iniziale di sviluppo in cui si trovano e la scarsità di sviluppatori e di persone con profili tecnici fanno sì che la convergenza sia ancora un compito in sospeso.

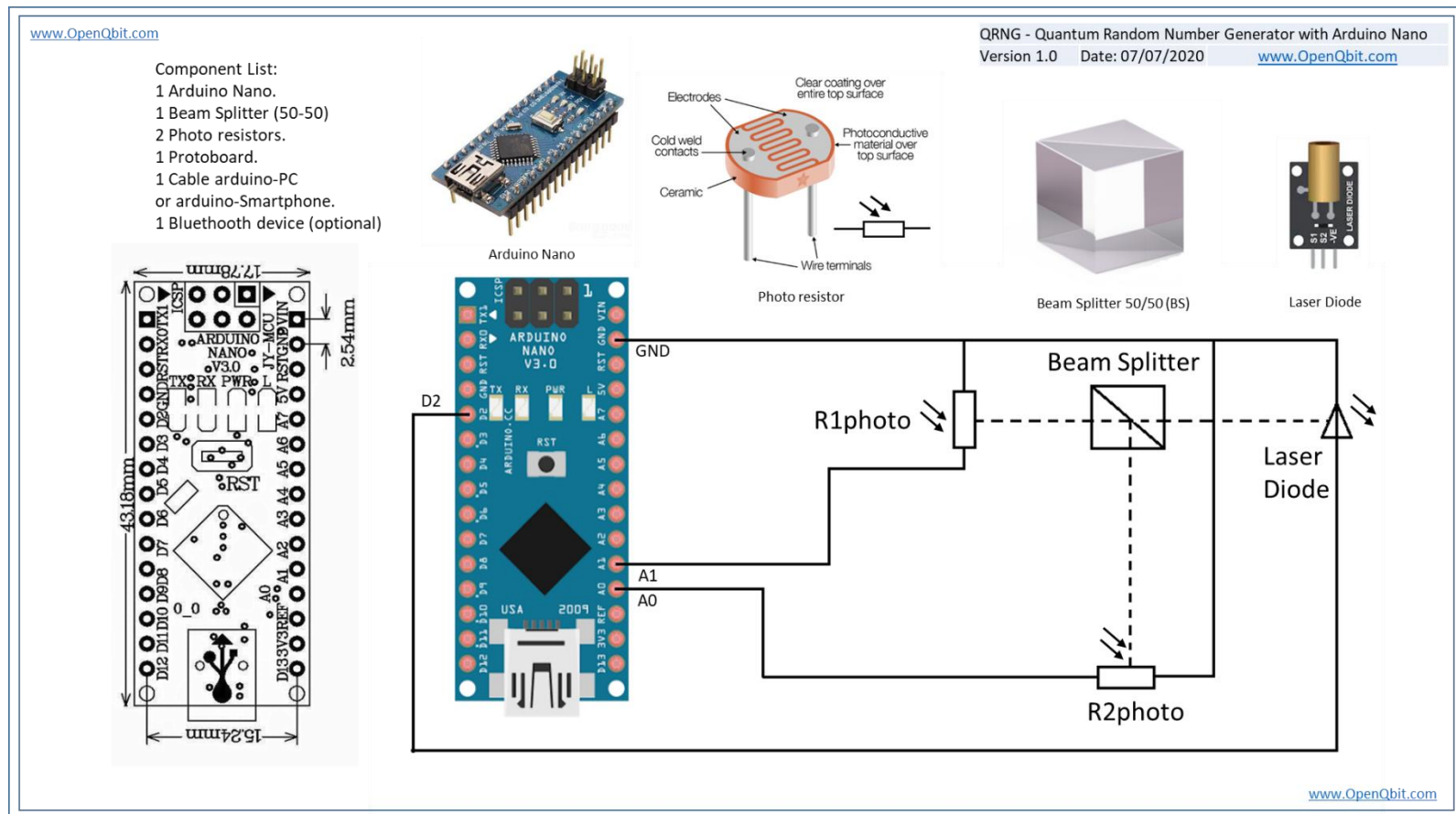
A causa del loro potenziale dirompente, ci si aspetta che le tecnologie quantistiche non solo convergano con tutte queste nuove tecnologie, ma che abbiano un'influenza trasversale su praticamente tutte. Il calcolo quantistico minaccerà l'autenticazione, lo scambio e l'archiviazione sicura dei dati, con un impatto importante sulle tecnologie in cui la crittografia ha un ruolo più rilevante, come la sicurezza informatica o la blockchain, e un impatto negativo minore, ma anche da considerare in tecnologie come il 5G, l'IoT o i droni.

### **Volete praticare il calcolo quantistico?**

Decine di simulatori di computer quantistici sono già disponibili in rete con diversi linguaggi di programmazione già in uso come C, C++, Java, Matlab, Maxima, Python o Octave. Inoltre, nuovi linguaggi come il Q#, lanciato da Microsoft. È possibile esplorare e giocare con una macchina quantistica virtuale attraverso piattaforme come IBM e Rigetti.

### 3. Creazione di un dispositivo "Hardware" di un QRNG (Quantum Random Number Generator).

Ora creeremo un dispositivo fisico "Hardware" per generare numeri casuali quantistici (QRNG) con componenti economici che possono essere facilmente assemblati a casa e costano circa 35 dollari.



## QRNGv1.0.ino

Software  
Program to arduino nano.

```
/* OpenQbitQRNG Firmware V1.0
 *Author: Guillermo Vidal
 *Copyright © 2020 OpenQbit, Inc.
 *License: MIT
 */
```

```
int triggerQ = 2; // This pin will pulse our quantum circuit
int QuA0Pin = A0; // This pin measures the horizontal polarized photons
int QuA1Pin = A1; // This pin measures the vertically polarized photons
float Qu0 = 0;
float Qu1 = 0;
```

```
void setup() {
  // Just setting up triggerPin and serial connection
  pinMode(triggerQ, OUTPUT); // sets the digital pin 2 as output
  Serial.begin(9600);
}
```

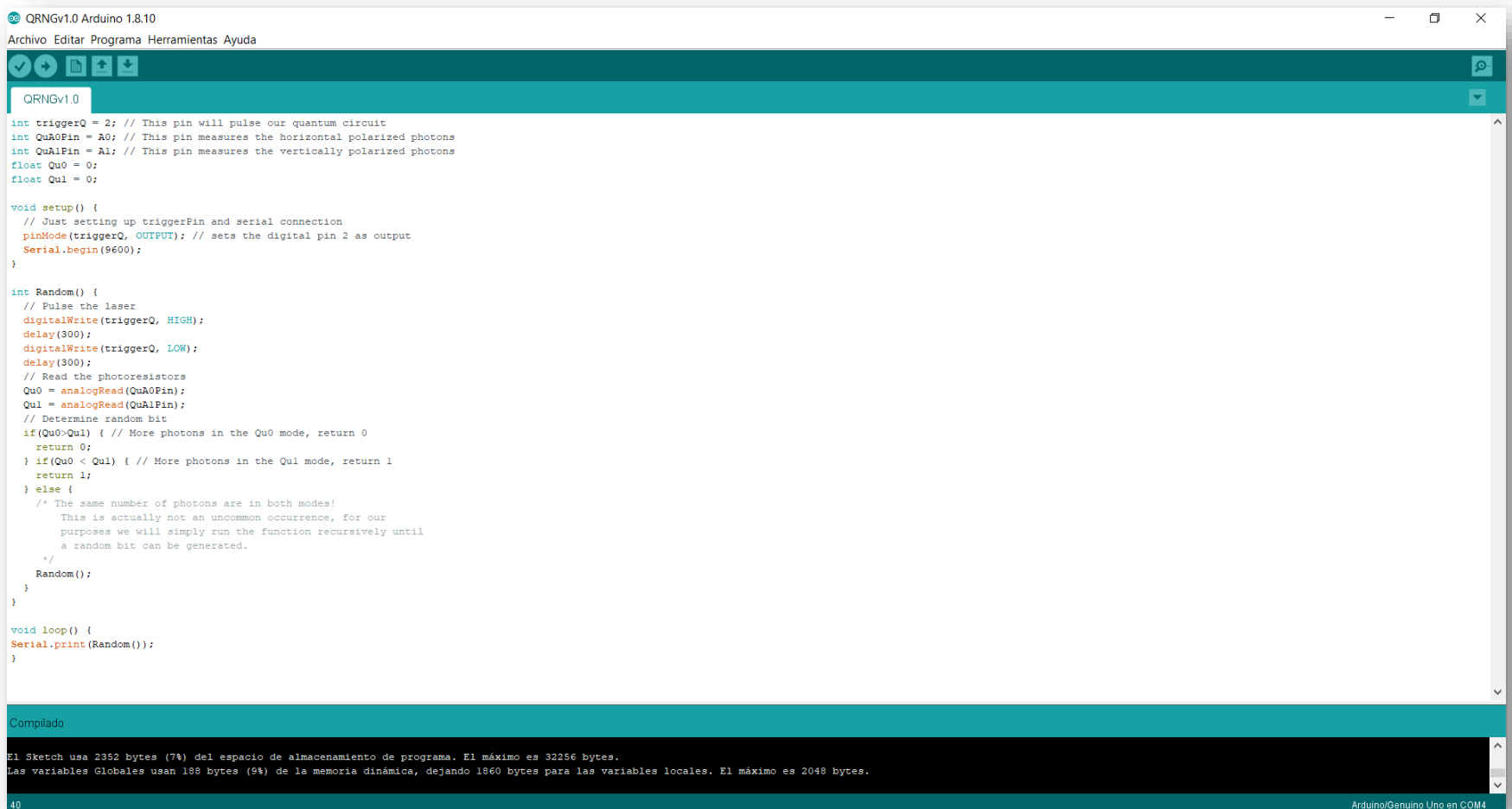
```
int Random() {
  // Pulse the laser
  digitalWrite(triggerQ, HIGH);
  delay(300);
  digitalWrite(triggerQ, LOW);
  delay(300);
  // Read the photoresistors
  Qu0 = analogRead(QuA0Pin);
  Qu1 = analogRead(QuA1Pin);
  // Determine random bit
  if(Qu0>Qu1) { // More photons in the Qu0 mode, return 0
    return 0;
  } if(Qu0 < Qu1) { // More photons in the Qu1 mode, return 1
    return 1;
  } else {
    /* The same number of photons are in both modes!
     This is actually not an uncommon occurrence, for our
     purposes we will simply run the function recursively until
     a random bit can be generated.
    */
    Random();
  }
}
```

```
void loop() {
  Serial.print(Random());
}
```

## Output console

0010110101011110101011010.....

Compilazione del programma QRNGv10.ino e caricamento su arduino nano....



The screenshot shows the Arduino IDE interface with the file 'QRNGv1.0' open. The code is written in C++ and implements a quantum random number generator using two photoresistors. The IDE's status bar at the bottom indicates the target board is 'Arduino/Genuino Uno en COM4'.

```
QRNGv1.0 Arduino 1.8.10
Archivo Editor Programa Herramientas Ayuda

QRNGv1.0

int triggerQ = 2; // This pin will pulse our quantum circuit
int QuA0Pin = A0; // This pin measures the horizontal polarized photons
int QuA1Pin = A1; // This pin measures the vertically polarized photons
float Qu0 = 0;
float Qu1 = 0;

void setup() {
  // Just setting up triggerPin and serial connection
  pinMode(triggerQ, OUTPUT); // sets the digital pin 2 as output
  Serial.begin(9600);
}

int Random() {
  // Pulse the laser
  digitalWrite(triggerQ, HIGH);
  delay(300);
  digitalWrite(triggerQ, LOW);
  delay(300);
  // Read the photoresistors
  Qu0 = analogRead(QuA0Pin);
  Qu1 = analogRead(QuA1Pin);
  // Determine random bit
  if(Qu0>Qu1) { // More photons in the Qu0 mode, return 0
    return 0;
  } if(Qu0 < Qu1) { // More photons in the Qu1 mode, return 1
    return 1;
  } else {
    /* The same number of photons are in both modes!
       This is actually not an uncommon occurrence, for our
       purposes we will simply run the function recursively until
       a random bit can be generated.
    */
    Random();
  }
}

void loop() {
  Serial.print(Random());
}
```

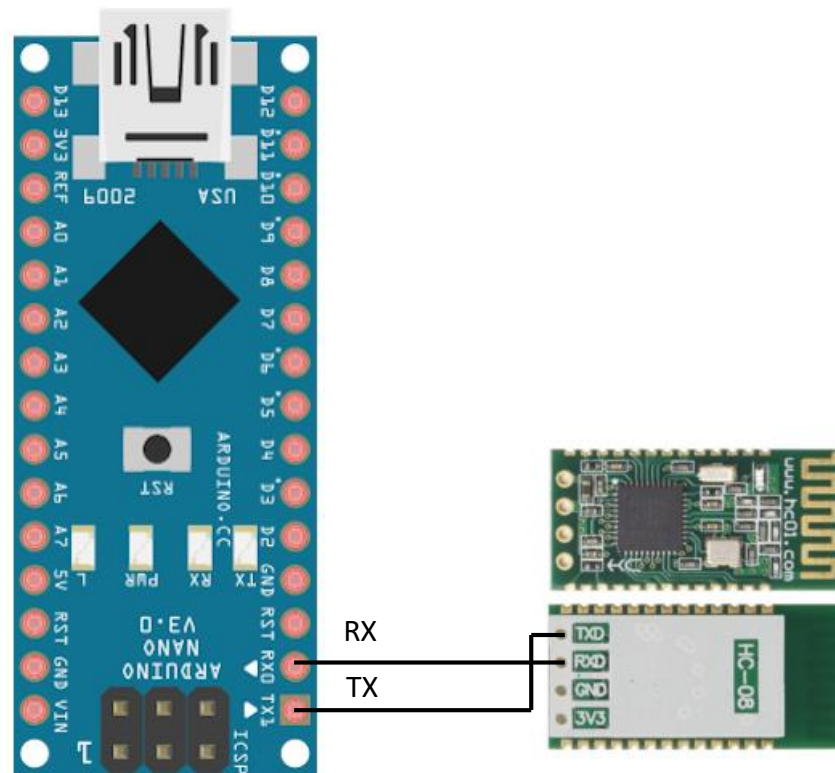
Compilado

El Sketch usa 2352 bytes (7%) del espacio de almacenamiento de programa. El máximo es 32256 bytes.  
Las variables Globales usan 188 bytes (9%) de la memoria dinámica, dejando 1860 bytes para las variables locales. El máximo es 2048 bytes.

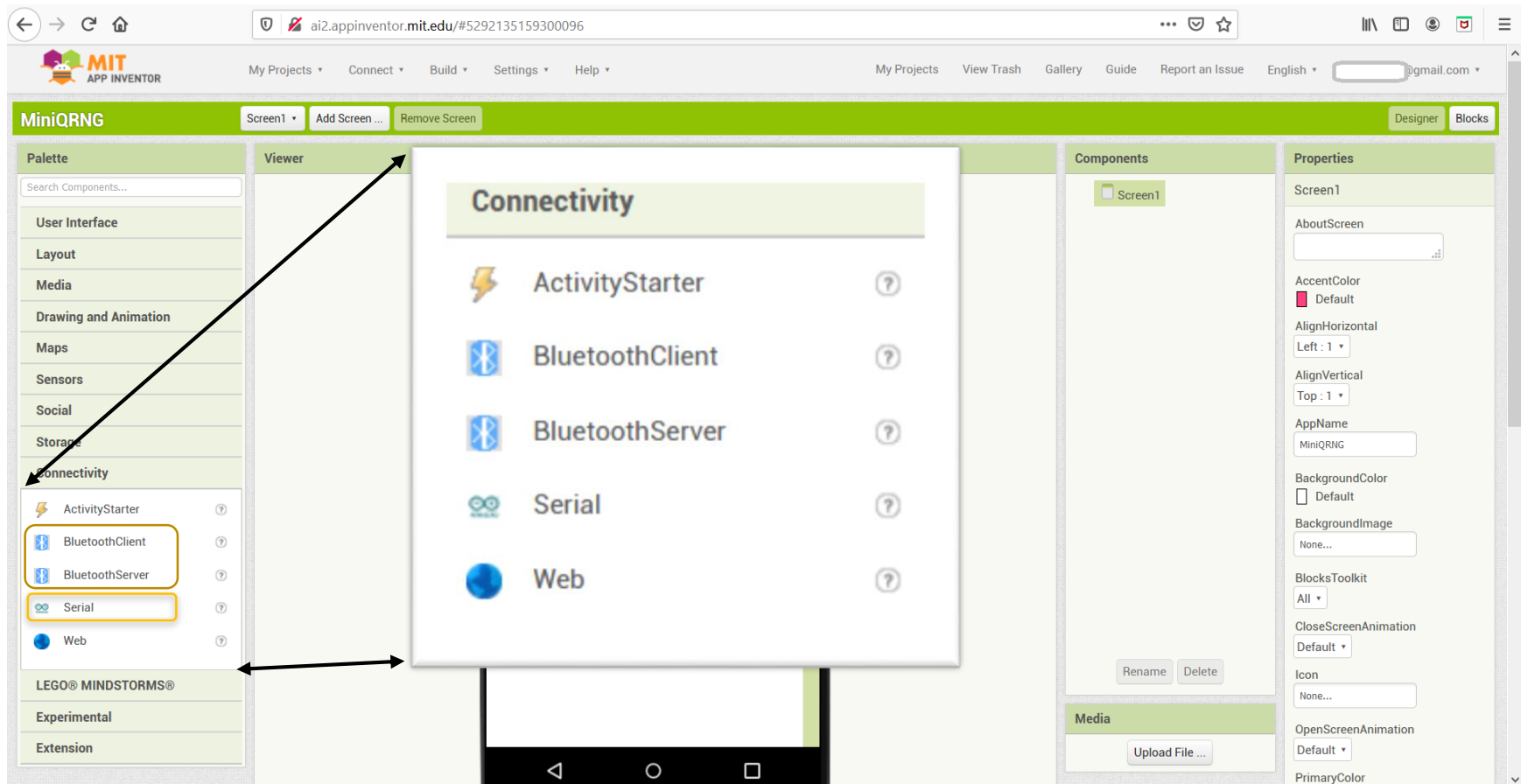
40 Arduino/Genuino Uno en COM4

Ci sono due modi per comunicare con l'arduo nano, uno è tramite la porta seriale e l'altro è tramite una connessione Bluetooth.

Per la connessione bluetooth è molto semplice basta acquistare il modulo HC-08 o uno simile e collegarlo come segue:



I seguenti componenti seriali o Bluetooth possono essere utilizzati per collegare App Inventor ad Arduino:





Ora compilato e caricato il programma QRNGv10.ino manca solo la comunicazione con l'arduo nano per salvare i dati (numeri casuali quantistici) questi saranno in formato binario, tuttavia, i dati ottenuti possono essere facilmente passati ad un altro formato come esadecimale o decimale a seconda del requisito finale.

Infine, per vedere un esempio di come funziona la connessione seriale o Bluetooth, ecco alcuni link di riferimento.

Ricordate che tutto è attraverso la programmazione Blockly da testare con App Inventor questo ha già blocchi per la comunicazione con arduino seriale o altro sistema di tipo blockly può essere attraverso il bluetooth simile online.

[http://kio4.com/appinventor/9A0\\_bluetooth\\_RXTX.htm](http://kio4.com/appinventor/9A0_bluetooth_RXTX.htm)

<http://kio4.com/appinventor/index.htm#bluetooth>

<https://community.appinventor.mit.edu/>

Rivedere l'intero progetto di progettazione e di utilizzo delle estensioni QRNG (Quantum Random Number Generator). Consultare il manuale d'uso all'indirizzo

<https://github.com/COINsolidation/UserGuide>





#### 4. Che cos'è la Prova del Quantum (PQu)?

PoQu. - "Proof of Quantum" è un algoritmo di consenso sviluppato per Mini BlocklyChain e COINsolidation, questo test è una variante della Proof of Work (PoW) che funziona come segue.

Il Test of Quantum (PoQu) all'avvio viene eseguito con lo stesso algoritmo del "Test of Work" (PoW) si basa sul mettere in funzione il processore del dispositivo (PC, Server, Tablet o Cellulare) per ottenere una stringa di caratteri che è un puzzle matematico chiamato "hash".

Ricordate che un "hash" è un algoritmo o processo matematico che quando si introduce una frase o un qualche tipo di informazione digitale come file di testo, programma, immagine, video, suono o altri diversi tipi di informazione digitale ci dà come risultato un carattere alfanumerico che rappresenta la firma digitale che la rappresenta in modo unico e non ripetibile dei dati, l'algoritmo di hash è unidirezionale, questo significa che quando si inserisce un dato per ottenere la sua firma "hash" il suo processo inverso non può essere eseguito, avendo una firma "hash" non possiamo sapere quali informazioni è stato ottenuto questa proprietà ci dà un vantaggio di sicurezza per elaborare le informazioni che inviamo su Internet. Come funziona? Immaginate di inviare qualsiasi tipo di informazione attraverso canali non sicuri e di accompagnarla con il suo rispettivo "source hash", il ricevitore quando riceve l'informazione può ottenere l'"hash" dell'informazione ricevuta lo chiameremo "destination hash" e verificheremo con l'"hash della fonte" se entrambi gli "hash" sono gli stessi possiamo confermare che l'informazione non è stata alterata nel canale che è stato inviato, è solo un esempio in cui questo tipo di processo di sicurezza dell'informazione è attualmente utilizzato.

Attualmente esistono diversi tipi di algoritmi o processi di hashish che si differenziano per il livello di sicurezza. I più usati o conosciuti sono: MD5, SHA256 e SHA512.

Esempio di SHA256:

Abbiamo una catena o una frase come segue: "Mini BlocklyChain è modulare.

Se applichiamo un hash SHA256 alla stringa precedente ci darà l'hash successivo.

**f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db8**



La stringa alfanumerica di cui sopra è la firma che rappresenta la frase nell'esempio precedente

Per ulteriori esempi possiamo utilizzare il sito su internet:

<https://emn178.github.io/online-tools/sha256.html>

Nel caso dell'algoritmo "Test Work" (PoW), funziona utilizzando la potenza di calcolo per ottenere un hash predefinito.

Immaginiamo di avere il precedente "hashish" che abbiamo preso dalla catena "Mini BlocklyChain è modulare".

**f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db8**

A questo "hash" all'inizio mettiamo il parametro di difficoltà che è semplicemente mettere degli zeri "0" all'inizio, cioè se diciamo che la difficoltà è di 4 avrà "0000" + "hash" a questo lo chiameremo "seed hash".

**0000 f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db8**

Ora tenendo conto che conosciamo l'informazione di input che è la stringa: "Mini BlocklyChain è modulare" aggiungiamo alla fine della stringa un numero che parte da zero "0" e togliamo il suo hash a questo lo chiameremo "hash nonce":

**f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db80**

Abbiamo l'hashish nonce:

**7529f3ad273fc8a9eff12183f8d6f886821900750bb6b59c1504924dfd85a7c8**

Poi effettuiamo un confronto del nuovo "nonce hash" con il "seme di hash" se sono uguali il nodo che per primo trova l'uguaglianza vincerà l'esecuzione dell'elaborazione della transazione corrente. Come possiamo vedere questo processo si basa sulla probabilità e sulla forza di calcolo del dispositivo che dà al test "Proof of Work" un'equità di consenso per tutti i nodi.

Se l'"hashish di seme" non coincide con l'"hash nonce", la difficoltà viene aumentata di uno e l'"hash nonce" viene eliminato di nuovo, il numero che viene aumentato viene chiamato numero "nonce", viene confrontato con l'"hashish di seme" finché non coincidono o sono uguali.

Come possiamo vedere il numero "nonce" o aumento è quello che aiuterà ad ottenere l'"hash" dell'uguaglianza.



Basato sull'algoritmo "Test of Work" (PoW), l'algoritmo Quantum Test (PoQu) si basa sull'ottenimento del numero "nonce" come fa PoW e sull'utilizzo di un livello minimo di difficoltà che va da 1 a 5, questo serve solo a guadagnare il diritto del dispositivo mobile di essere un candidato per ottenere il consenso.

Il Quantum Test (PoQu), si attiva quando il telefono cellulare ha terminato il PoW minimo e vince il passaggio per ottenere un numero di probabilità nel sistema QRNG.

Il QRNG (Quantum Random Number Generator) è un Generatore di Numeri Casuali Quantici, questo sistema si basa sulla generazione di veri numeri casuali basati sulla meccanica quantistica è il sistema più sicuro oggi per generare tali numeri. Per maggiori dettagli vedere "Sicurezza del calcolo quantistico" nell'indice 3.

COINsolidation può implementare sia i tipi di concessione minima PoW che PoQu.

Il test PoQu si basa sull'ottenimento del numero "nonce". Questo numero nel test PoQu è conosciuto come "Magic Number" e con questo il sistema Peer to Peer confermerà se il numero è corretto e poi un numero casuale sarà ottenuto con il pool di server COINsolidation QRNG. Questo numero casuale verrà registrato in tutti i nodi, verrà creata una lista contenente **((Node Addition) /2) +1** e da questa lista verrà scelto quello con la più alta percentuale di probabilità di essere il candidato vincitore del consenso (PoQu) e verrà eseguita la coda di transazione corrente.

L'algoritmo PoQu utilizza anche i test del **NIST** (National Institute of Standards and Technology) per assicurarci che i numeri casuali nel QRNG siano veramente casuali.

<https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-22r1a.pdf>

In COINsolidation abbiamo implementato un blocco per PoW e un blocco per PoQu. Questi blocchi utilizzano un tipo di hash: SHA256 per uso gratuito, per uso commerciale abbiamo un SHA512 e altri hash come richiesto.

Per maggiori dettagli sul concetto di HASH vedi:

[https://es.wikipedia.org/wiki/Funcion\\_hash](https://es.wikipedia.org/wiki/Funcion_hash)

NOTA: Il Test of Work (PoW) utilizzato nei telefoni cellulari può utilizzare solo una difficoltà massima di 5 poiché l'elaborazione matematica di questi dispositivi non è dedicata come server o PC. Utilizziamo l'algoritmo PoW solo per ottenere la possibilità di ottenere il pass o il permesso di entrare nel sistema Quantum Random Number Generator (QRNG) e con esso eseguire l'algoritmo Quantum Random Number Generator (PoQu). Vedi uso di (PoQu) in Mini BlocklyChain:

<https://github.com/openqbit-diy/MiniBlocklyChain>



## 5. Algoritmo per la creazione di un indirizzo universale consolidato (CUA)

I nostri indirizzi CUA (Consolidated Universal Address) sono creati utilizzando il seguente algoritmo:

Passo 1.- Gli identificatori vengono rimossi dai rispettivi indirizzi, sono i caratteri alfanumerici che identificano l'indirizzo dal quale è stata creata la catena di blocco.

Indirizzo Bitcoin- Token - (OAP).

**akXma4vqxvmEqnVAKSM953wYsnjNBhN3GM7**

Indirizzo Ethereum - Token COINsolidation - (ERC20).

**0x9d08c0ac0f2fdf078c883db6fa617b15776e4b41**

Passo 2.- L'SHA512(indirizzo Stringa-Testo) di ogni indirizzo senza il suo identificatore iniziale si ottiene togliendo la "a" da A1 e lo "0x" da A2 e prendendo gli ultimi due caratteri di ogni operazione di hash simboleggiata con "U". Numeri di verifica.

$U(\text{SHA512}(\text{akXma4vqxvmEqnVAKSM953wYsnjNBhN3GM7})) = \text{50}$

$U(\text{SHA512}(\text{9d08c0ac0f2fdf078c883db6fa617b15776e4b41})) = \text{fb}$

Fase 3.- I caratteri di ogni indirizzo sono concatenati uno per uno a partire dall'indirizzo che ha meno caratteri che lo compongono, in caso di avere la stessa quantità di caratteri la concatenazione può partire da qualsiasi indirizzo.

Indirizzo 1 = A10 [0], A11 [1], A12 [2], A13 [3], A14 [4] ..... A1N[n], A1N+1[n+1].

Indirizzo 2 = A20 [0], A21 [1], A22 [2], A23 [3], A24 [4] ..... A2N[n], A2N+1[n+1].

Concatenazione di indirizzi:

A10 [0] + A20 [0] + A10 [1] + A20 [1] + A20 [1] + A11 [2] + A22 [2] + .... A1N+1 [n+1] + A2N+2 [n+1].

**\*\*Gli ultimi caratteri che non possono essere concatenati vengono messi all'inizio della stringa.**

6e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777

Passo 4.- Il numero di caratteri che potrebbero essere concatenati nel passo 3 viene aggiunto all'inizio della stringa risultante dal passo 3.

66e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777



Passo 5.- Le due coppie di verificatori del passo 2 di ogni direzione sono concatenate all'inizio della catena risultante dal passo 3 nello stesso ordine A1 + A2.

50fb66e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777

Fase 6.- L'identificazione **CUAG** (Consolidated Universal Address Genesis) è integrata all'inizio dell'indirizzo creato nella fase 5.

**cua**50fb66e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777

**\*\***Nel caso del Bitcoin e dell'Ethereum, l'indirizzo sarà composto da **82** caratteri esadecimali.

## 6. Algoritmo per il doppio indirizzo consolidato (DAC) e (HAC).

La creazione di un DAC è la stessa del CUA, la differenza è che nei DAC sono utilizzati per consolidare gli indirizzi normali per ricevere le transazioni, questi indirizzi non rappresentano alcun cryptomonedà o token.

Passo 1.- Gli identificatori vengono rimossi dai rispettivi indirizzi, sono i caratteri alfanumerici che identificano l'indirizzo dal quale è stata creata la catena di blocco.

18gYNA9c2G9G9X8HZ8HZ8QxWLpLXZauAxFnsJbe (Indirizzo Bitcoin)

0x5d2Acdb34c279Aa6d1e94a77F7b18aB938BFb2bB (Dirección Ethereum)

Passo 2.- L'SHA512(indirizzo Stringa-Testo) di ogni indirizzo senza il suo identificatore iniziale si ottiene togliendo il "**1**" da A1 e lo "**0x**" da A2 e prendendo gli ultimi due caratteri di ogni operazione di hash simboleggiata con "U". Numeri di verifica.

U(SHA512(8gYNA9c2G9G9X8HZ8HZ8QxWLpLXZauAxFnsJbe)) = **48**

U(SHA512(5d2Acdb34c279Aa6d1e94a77F7b18aB938BFb2bB)) = **f3**

Fase 3.- I caratteri di ogni indirizzo sono concatenati uno per uno a partire dall'indirizzo che ha meno caratteri che lo compongono, in caso di avere la stessa quantità di caratteri la concatenazione può partire da qualsiasi indirizzo.

Indirizzo 1 = A10 [0], A11 [1], A12 [2], A13 [3], A14 [4] ..... A1N[n], A1N+1[n+1].

Indirizzo 2 = A20 [0], A21 [1], A22 [2], A23 [3], A24 [4] ..... A2N[n], A2N+1[n+1].



Concatenazione di indirizzi:

$A10 [0] + A20 [0] + A10 [1] + A20 [1] + A20 [1] + A11 [2] + A22 [2] + \dots A1N+1 [n+1] + A2N+2 [n+1]$ .

**\*\***Gli ultimi caratteri che non possono essere concatenati vengono messi all'inizio della stringa.

**8BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3**

Passo 4.- Il numero di caratteri che potrebbero essere concatenati nel passo 3 viene aggiunto all'inizio della stringa risultante dal passo 3.

**78BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3**

Passo 5.- Le due coppie di verificatori del passo 2 di ogni direzione sono concatenate all'inizio della catena risultante dal passo 3 nello stesso ordine  $A1 + A2$ .

**48f378BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3**

Fase 6.- L'identificazione **DAC** (Dual Address Consolidated) è integrata all'inizio dell'indirizzo creato nella fase 5.

**dac48f378BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3**

**\*\***Nel caso del Bitcoin e dell'Ethereum, il consolidamento degli indirizzi darà un indirizzo composto da **81** caratteri esadecimali.

Nel caso di HAC (Hibric Address Consolidated) si applica nel precedente ciò che varia sono gli indirizzi che vengono utilizzati, in questo caso si utilizzerà un indirizzo che rappresenta un asset (Cryptomonedà o token) e un normale indirizzo standard di trasferimento di asset di qualche tipo di blockchain.

**NOTA:** Le dimensioni degli indirizzi CUA, HAC e DAC possono variare in ogni caso a seconda degli indirizzi che li compongono.



## Progetto e soluzione di COINsolidation.

Attualmente esistono diversi tipi di Blockchain orientati ad asset con caratteristiche diverse, questo porta ad avere un numero infinito di tipi di indirizzi di uso quotidiano devono mantenere uno stretto controllo per evitare di commettere errori di trasferimento.

D'altra parte, il mondo dei crittomoni e dei gettoni è limitato agli esperti di finanza o, nel loro caso, agli esperti di tecnologia a catena, per cui è difficile per la persona media avventurarsi nella creazione di un proprio crittomone o gettone.

Abbiamo risolto i due problemi precedenti in COINsolidamento facendo i seguenti punti e/o strumenti che abbiamo creato.

Per il punto di controllo degli indirizzi di diverse catene a blocchi, abbiamo creato un algoritmo in cui si consolidano (unisce) due o più indirizzi nelle loro diverse combinazioni dando come risultato un unico indirizzo di tipo CUA, HAC e/o DAC.

Con questa soluzione, invece di inviare due indirizzi dalla stessa o da diverse catene di blocco, verrà utilizzato un solo indirizzo consolidato.

Per il secondo problema abbiamo utilizzato la metodologia di programmazione chiamata Blockly, si tratta di uno strumento visivo in cui non è necessaria una grande conoscenza della programmazione e qualsiasi persona o azienda media sarà in grado di creare le proprie applicazioni senza dover investire costosi team di sviluppo, tempo e denaro.

Abbiamo creato le estensioni (moduli) per installarle e utilizzarle semplicemente per creare applicazioni mobili, in 15 minuti. Esempio il proprio cambio di valuta criptata o sviluppare la propria valuta (token) in pochi minuti. Tutto ciò utilizzando lo stato dell'arte della sicurezza dei dati chiamato PQC (Post-Quantum Cryptography).

Basta installare le estensioni su qualsiasi strumento gratuito come Appinventor, AppyBuider, Thunkable, Kondular o altri e in pochi minuti si può entrare nel mondo delle crittomie e della creazione di gettoni tutto nel palmo della mano.

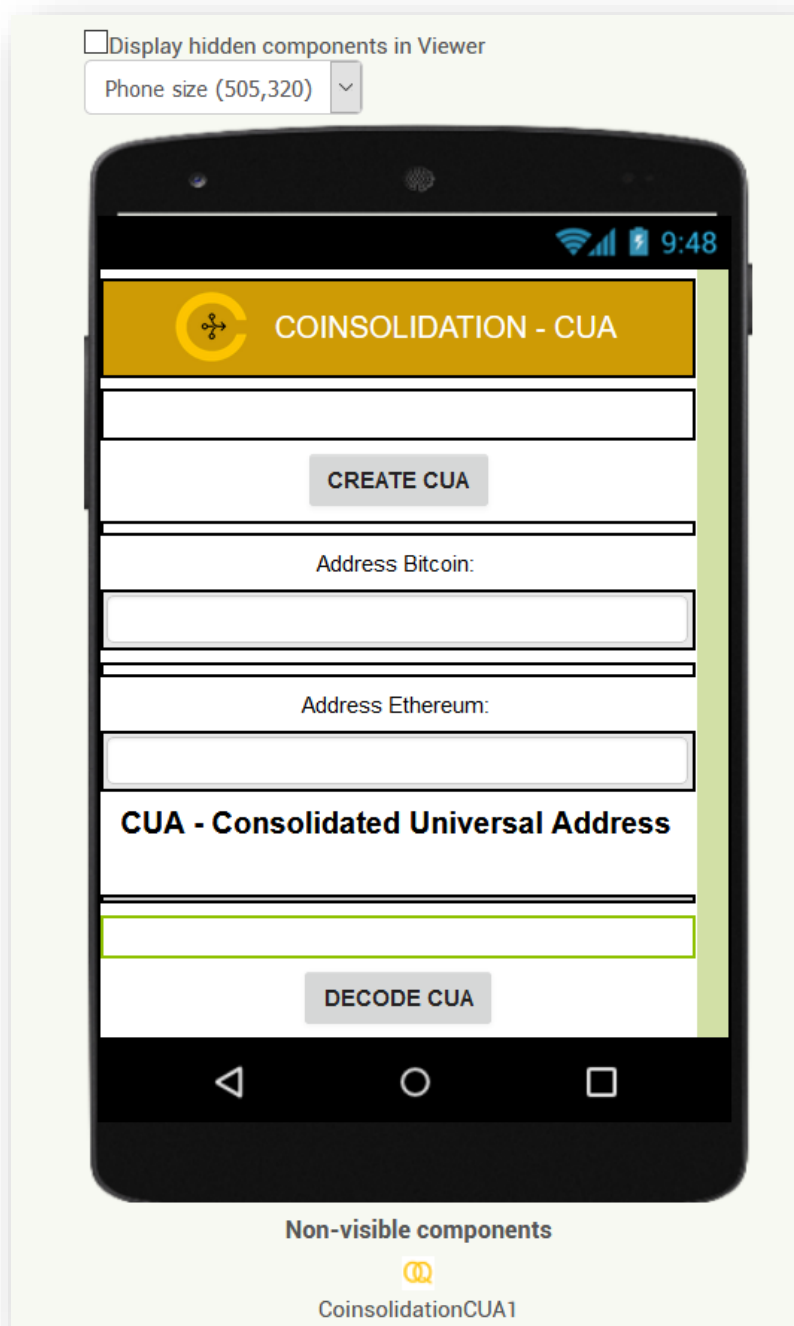
Infine, COINsolidation sta creando l'uso di sicurezza quantistica a basso costo (software e hardware) che può essere utilizzata per proteggere i dati del computer a casa. Attualmente le tecnologie basate sul calcolo quantistico e sulla sicurezza hanno un costo elevato che solo le aziende con un elevato livello finanziario possono creare e utilizzare. Tuttavia, in COINsolidation crediamo che le nuove tecnologie debbano essere disponibili a tutti, che l'equità di utilizzo della Blockchain e del Quantum Computing debba essere per tutti, che si crei software libero (crittomonia) e hardware a basso costo (sicurezza quantistica).



## 7. Creazione di App CUA (Consolidated Universal Address) in 15 minuti.

\*App per le monete Bitcoin e Ethereum (BTC-ETH)

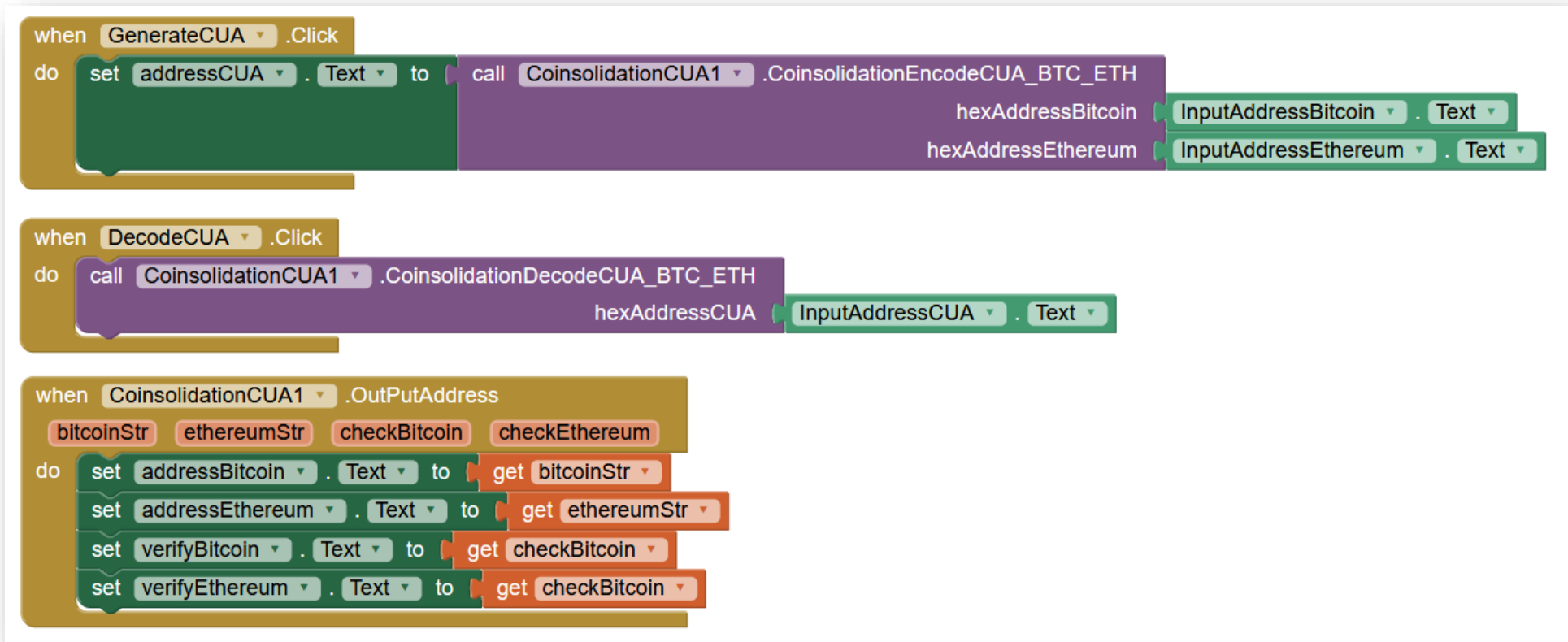
Schermata di progettazione 5 minuti in <https://appinventor.mit.edu/>







Uso della prolunga CoinsolidatioCUA.AIX (5 minuti).





Creiamo l'applicazione in **Menu > Build > App** (fornire il codice QR per .apk) - (5 minuti).

when **GenerateCUA** .Click

do

set **addressCUA** . Text to

call **CoinsolidationCUA1** .CoinsolidationEncodeCUA\_BTC\_ETH

hexAddressBitcoin **InputAddressBitcoin** . Text

hexAddressEthereum **InputAddressEthereum** . Text

when **DecodeCUA** .Click

do

call **CoinsolidationCUA1** .CoinsolidationDecodeCUA\_BTC\_ETH

hexAddressCUA **InputAddressCUA** . Text

when **CoinsolidationCUA1** .OutPutAddress

bitcoinStr **ethereumStr** **checkBitcoin** **checkEthereum**

do

set **addressBitcoin** . Text to **get bitcoinStr**

set **addressEthereum** . Text to **get ethereumStr**

set **verifyBitcoin** . Text to **get checkBitcoin**

set **verifyEthereum** . Text to **get checkBitcoin**

0

0

Show Warnings

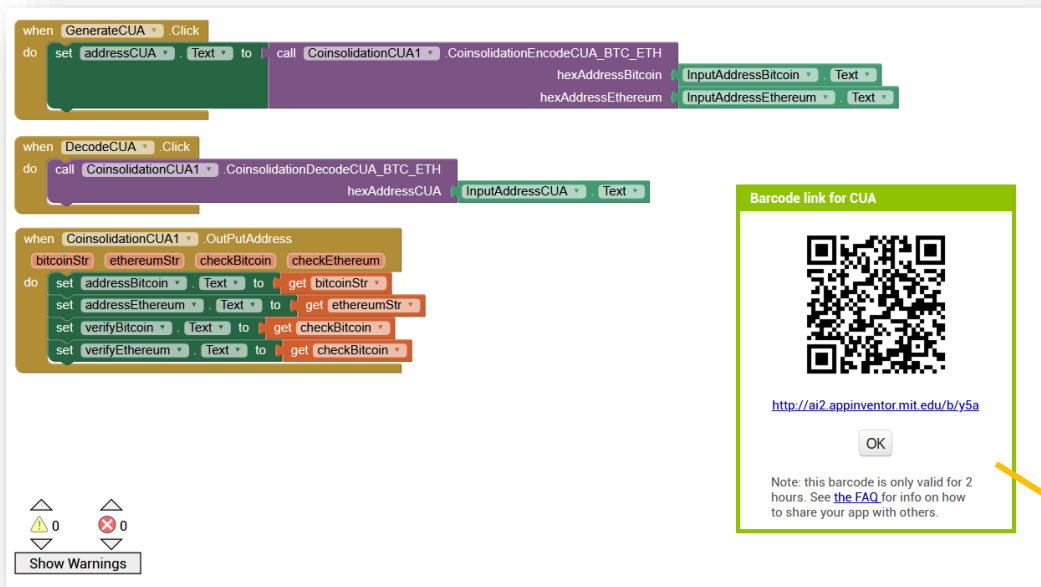
CUA Progress Bar

35%

Compiling part 2 (please wait)



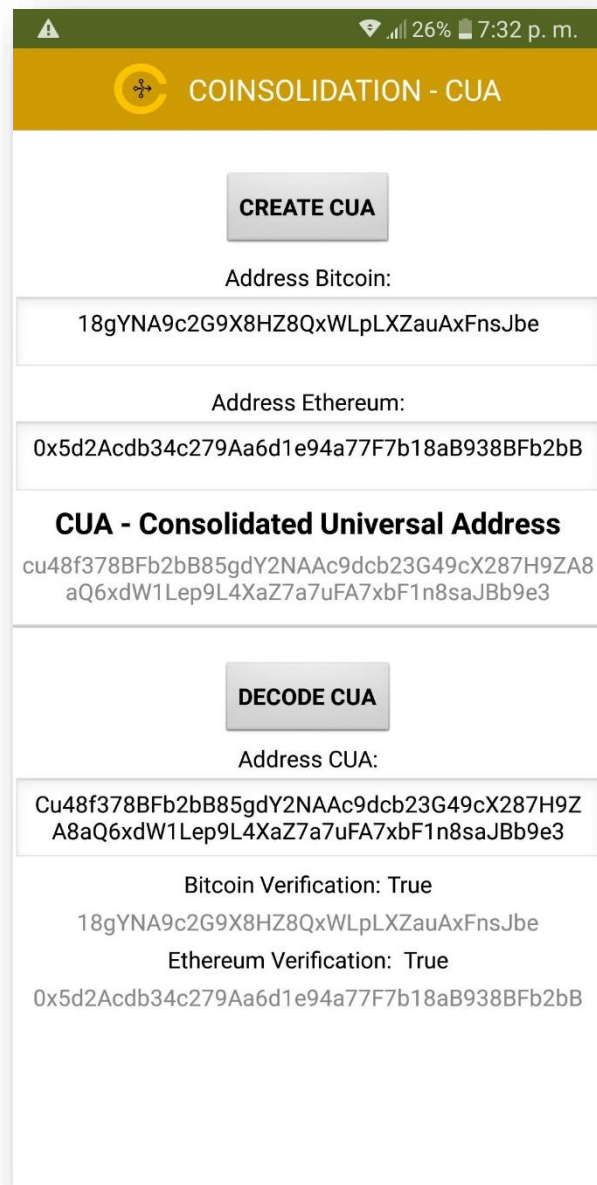
Abbiamo installato l'applicazione sul cellulare dal QR utilizzando l'applicazione Android di AppInventor (MIT AI2 Companion) - <https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>



**NOTA:** Il file APK dell'applicazione pronto per essere installato si trova nel seguente repository:

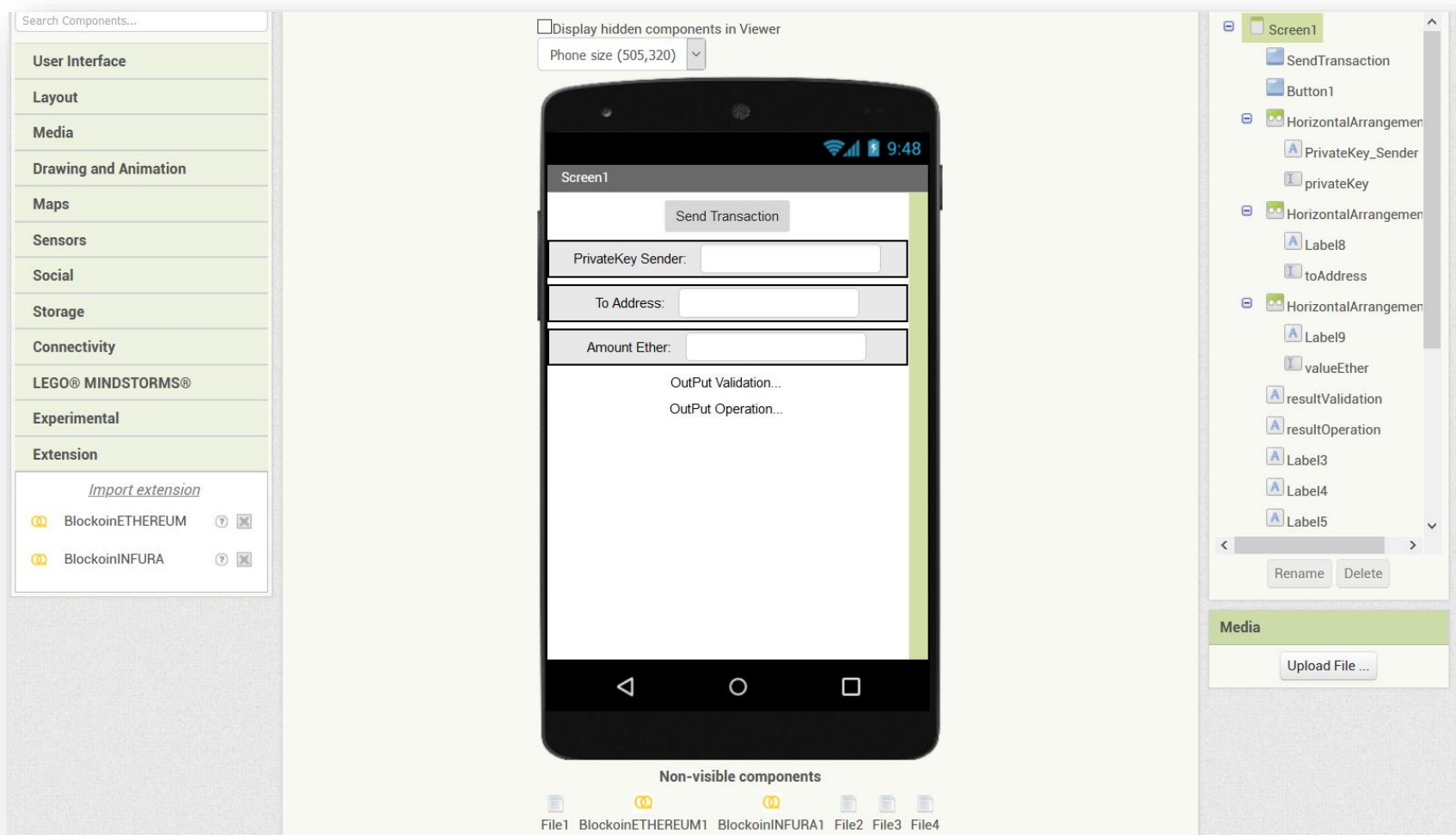
<https://github.com/COINsolidation/App>

Per rivedere il Java Code for CUA extension generation e per implementare un algoritmo di generazione di indirizzi universale consolidato, rivedere l'allegato "Code for CUA algorithm" o consultare il link del codice: <https://github.com/COINsolidation/source>



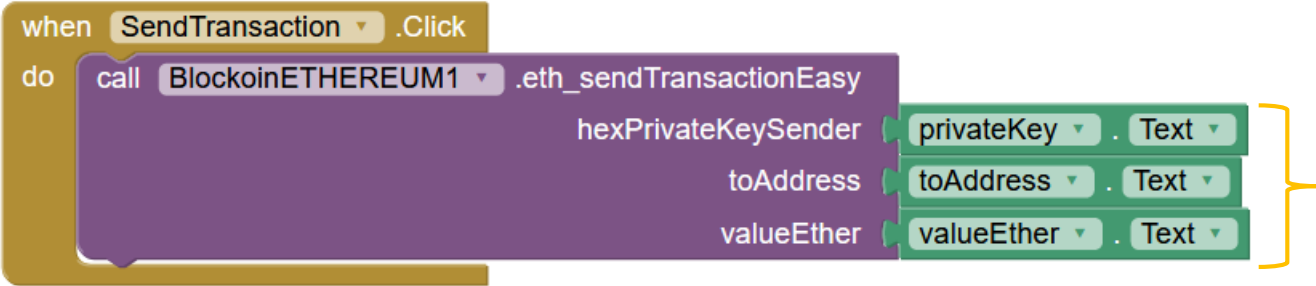


8. Crea il tuo cambio valuta crypto Ethereum su Android in soli 15 minuti.  
Design in App Inventor (schermo). - 5 minuti.





Blocchi funzione (eth\_SendTransactionEasy) ed evento (OutPutSendTransactionEasy) - 5 minuti

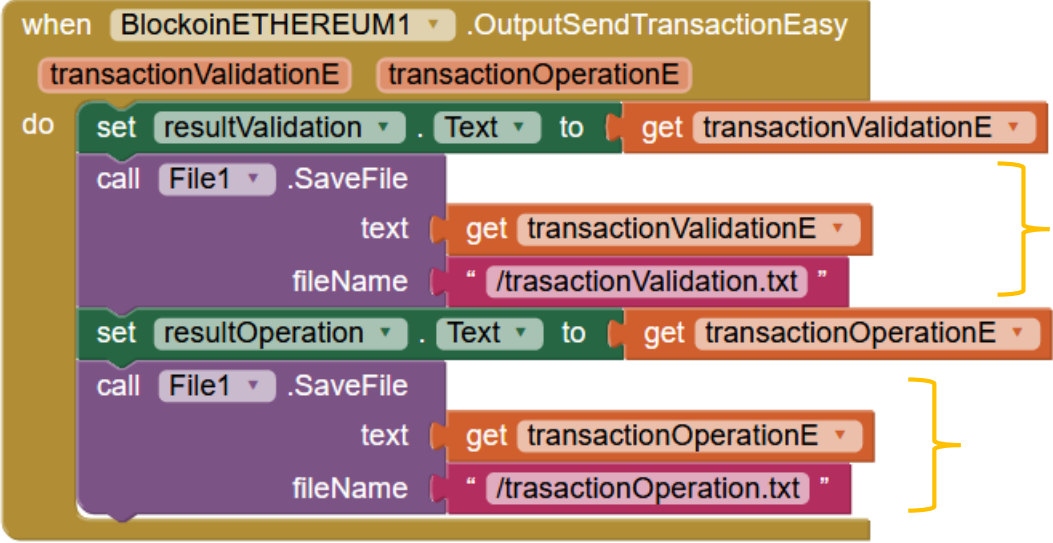


Dati di ingresso:

**PrivateKey:** chiave primaria per l'indirizzo del mittente.

**toAddress:** indirizzo esadecimale del destinatario.

**valueEther:** indicare la quantità di Etere che verrà inviata.



Salvare i risultati in file di testo:

Funzione File1: File **trasactionValidation.txt**

Salvare i risultati in file di testo:

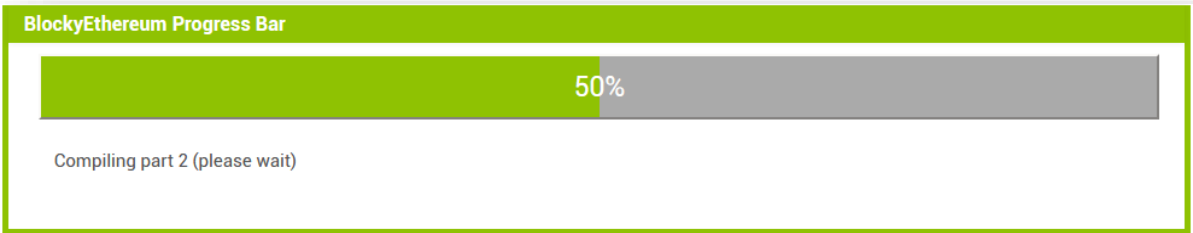
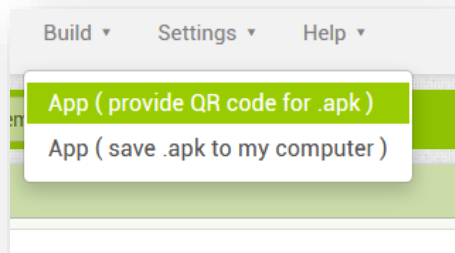
Funzione File2: **Traslazione dei fileValidazione.txt**

\*\*Per maggiori dettagli vedere il manuale d'uso dell'estensione Ethereum Exchange (EEE) nel repository: <https://github.com/COINsolidation/userguide>

\*\*Repository di estensioni COINsolidation: <https://github.com/coinsolidation/Extesions-Cryptocurrencies> o OpenQbit (Blockchain & Quantum Computing) <https://github.com/openqbit-diy>



Compiliamo, generiamo un file APK per installarlo sul dispositivo Android. - 5 minuti



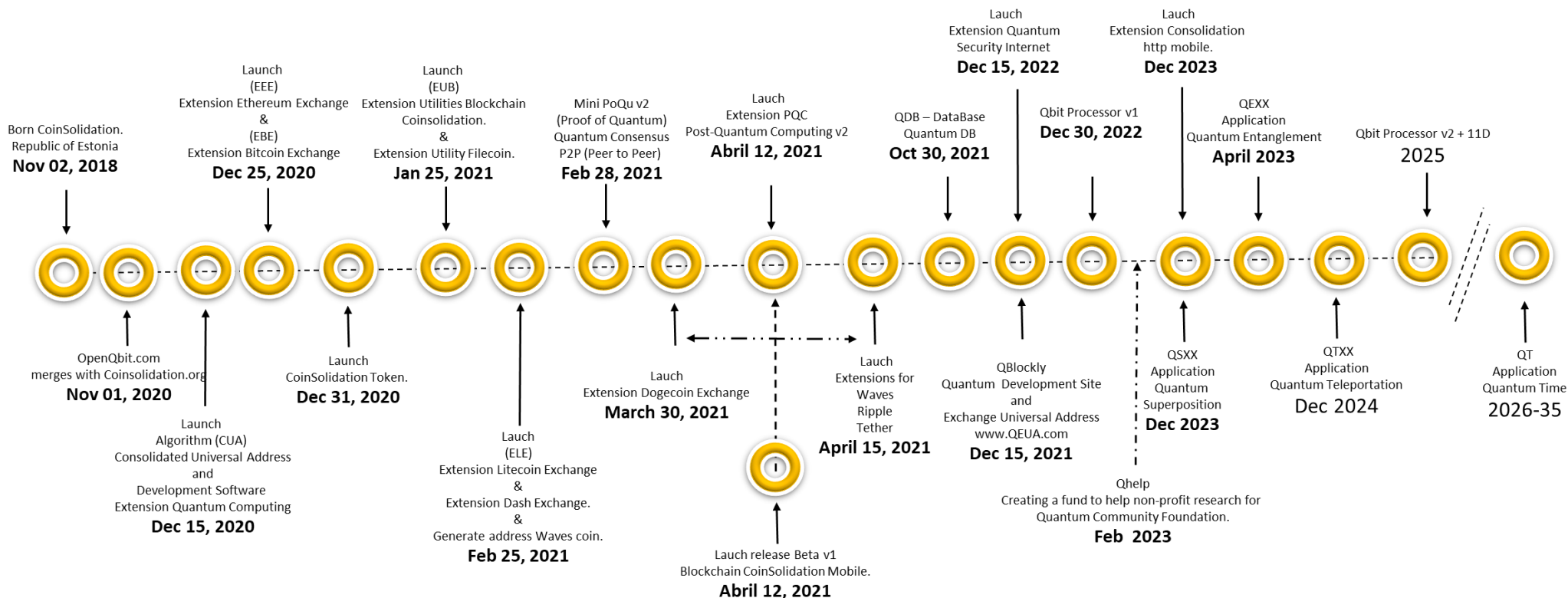
NOTA: Quando la transazione viene eseguita, ci vorranno circa 6-8 secondi per rilasciare il pulsante "Invia transazione". A causa del tempo di connessione con la rete dell'Ethereum.

**Per maggiori dettagli sull'estensione del SEE - (estensione dello scambio di etereum). Vedere il manuale d'uso dell'AEE nel link:**

<https://github.com/COINsolidation/UserGuide>



## ROADMAP



\*Il processore quantistico versione 1 utilizzerà i cancelli logici quantistici di base per uso domestico.



## 10. Gettone di consolidamento COIN (CUAG) - PIANO DI DISTRIBUZIONE IC



EXchange  
tensions

L'ICO è diviso in tre fasi:

The private sale	\$ 0.01 USD	(30/Dec 2020 - 30/Jan 2021)	HARD CAPITAL: \$ 280,000,000.00 USD
ICO FIRST PHASE	\$ 0.01 USD	(31/Jan 2021 - 28/Feb 2021)	SOFT CAPITAL: \$ 280,000 USD
ICO SECOND PHASE	\$ 0.05 USD	(1/Mar 2021 - 31/Mar 2021)	

CoinSolidation TOKEN DISTRIBUTION		
	%	TOKENS
TOKEN SALE	70	28,000,000,000.00
TEAM AND DEVELOPMENT	10	4,000,000,000.00
ADVISORS	5	2,000,000,000.00
PARTNERS	5	2,000,000,000.00
EXCHANGES MARKET	1.5	600,000,000.00
MARKETING	5	2,000,000,000.00
COINsolidation FOUNDATION	0.5	200,000,000.00
BLOCKLY DEVELOPER COMMUNITIES	0.5	200,000,000.00
DEVELOPMENT AND RESEARCH OF QUANTUM COMPUTING	1.5	600,000,000.00
TOTAL SUPPLY 100%		40,000,000,000.00

0x4B7355FD05be6dAC458B004f54e12d6527A54A58	INDIRIZZO ICO
0x9d08c0ac0f2fdf078c883db6fa617b15776e4b41	INDIRIZZO DELL'OPERAZIONE DI CO-SOLIDAMENTO





## 11. Caratteristiche generali del gettone di consolidamento COIN:

Creato da: Lugu Samaya.

Nome: COINsolidamento

Simbolo: CUAG - (Consolidated Universal Address Genesis).

Tipo: NFT

Totale gettoni creati: 40.000.000.000.000,00

Numero di decimali: 18

Paese di lancio: Estonia

Sito ufficiale: [www.COINsolidation.org](http://www.COINsolidation.org)

Azienda: COINsolidation International.

Data di lancio: 30 dicembre 2020

Algoritmo di consenso: PQu (Proof of Quantum)

Algoritmo di indirizzo: Indirizzo universale consolidato (CUA).

Sicurezza utilizzata: PQC (Post-Quantum Cryptography) basata sul calcolo quantistico.

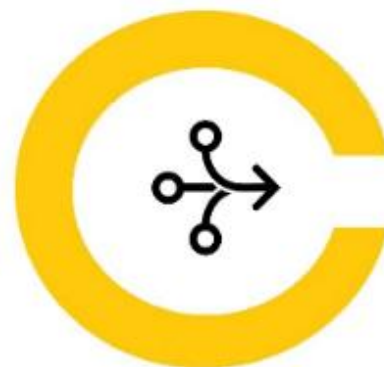
Proposta tecnologica: Estensioni per i sistemi Blockly per l'utilizzo di crittomonadi e implementazione della sicurezza quantistica.

Partnership o accordi tecnologici (fusione):

Azienda: OpenQbit Inc.

Industria: calcolo quantistico e PQC (Crittografia Post-Quantistica).

Sito web ufficiale: [www.OpenQbit.com](http://www.OpenQbit.com)





## 12. Concetti di base applicati nelle piattaforme Blockchain.

### Cos'è una catena di blocco?

La blockchain è generalmente associata a Bitcoin e ad altre valute crittografiche, ma queste sono solo la punta dell'iceberg, poiché non viene utilizzata solo per la moneta digitale, ma può essere utilizzata per qualsiasi informazione che possa avere un valore per gli utenti e/o le aziende. Questa tecnologia, che ha le sue origini nel 1991, quando Stuart Haber e W. Scott Stornetta hanno descritto il primo lavoro su una catena di blocchi crittografati, è stata notata solo nel 2008, quando è diventata popolare con l'arrivo del bitcoin. Ma attualmente il suo utilizzo è richiesto in altre applicazioni commerciali e si prevede che crescerà nel medio futuro in diversi mercati, come le istituzioni finanziarie o l'Internet degli oggetti tra gli altri settori.

La blockchain, meglio conosciuta con il termine blockchain, è un singolo record concordato distribuito su più nodi (dispositivi elettronici come PC, smartpohones, tablet, ecc.) in una rete. Nel caso delle valute criptate, possiamo pensarlo come il libro contabile in cui viene registrata ciascuna delle operazioni.

Il suo funzionamento può essere complesso da capire se entriamo nei dettagli interni della sua realizzazione, ma l'idea di base è semplice da seguire.

Viene memorizzato in ogni blocco:

- 1.- una serie di registrazioni o transazioni valide,
- 2.- informazioni relative a quel blocco,
- 3.- il suo collegamento con il blocco precedente e il blocco successivo attraverso l'hash di ogni blocco –un codice unico che sarebbe come l'impronta digitale del blocco.

Pertanto, **ogni blocco** ha un **posto specifico e non mobile all'interno della catena**, in quanto ogni blocco contiene informazioni provenienti dall'hash del blocco precedente. L'intera catena è memorizzata su ogni nodo di rete che compone la blockchain, quindi **una copia esatta della catena è memorizzata su tutti i partecipanti alla rete**.

### Che cos'è un indirizzo o un account all'interno della piattaforma Blockchain Ethereum?

Si tratta di una stringa di 42 caratteri nella piattaforma dell'Ethereum che rappresenta un numero in base esadecimale, dove i beni definiti nell'Ethereum saranno depositati o inviati. In altre piattaforme a catena di blocco, ad esempio, il numero di caratteri del conto o dell'indirizzo può essere diverso:



0x5d2Acdb34c279Aa6d1e94a77F7b18aB938BFb2bB

### **Cos'è un kryptomoney?**

Si tratta di una moneta digitale o virtuale progettata per funzionare come mezzo di scambio. Utilizza la crittografia (sicurezza digitale) per mettere in sicurezza e verificare le transazioni, oltre che per controllare la creazione di nuove unità di una particolare crittografia.

### **Cos'è un gettone?**

I gettoni sono risorse digitali che possono essere utilizzate all'interno di un determinato ecosistema di progetto.

La principale distinzione tra gettoni e monete criptate è che le prime richiedono un'altra piattaforma a catena di blocco (non la propria) per funzionare. L'Ethereum è la piattaforma più comune per la creazione di gettoni, soprattutto grazie alla sua funzione di contratto intelligente. I gettoni creati sulla catena di blocco dell'Ethereum sono generalmente noti come gettoni ERC-20, anche se esistono altri tipi di gettoni più specializzati, come il gettone ERC-721, utilizzato principalmente per beni da collezione (carte, uso nei videogiochi, opere d'arte, ecc.).

### **Cos'è uno scambio?**

Un cambio di valuta criptata è il punto d'incontro dove avviene lo scambio di valute criptate in cambio di denaro fiat o di altre valute criptate. In queste borse online si genera il prezzo di mercato che segna il valore delle crittomie in base alla domanda e all'offerta.

### **Che cosa sono i tassi di cambio?**

Questi sono i tassi del valore di un Etere o di un'altra valuta criptata nella valuta circolante di ogni paese. Ad esempio, il giorno della creazione di questo manuale, un Etere ha un valore in dollari USA di \$430,94

### **Cos'è una transazione?**

Si tratta dell'esecuzione o del trasferimento di un qualche tipo di bene non tangibile che può essere dato un valore prestabilito all'interno del sistema Ethereum e che può essere successivamente modificato in un valore tangibile per una società o una persona.

### **Che cos'è txHash?**

Si tratta di un numero esadecimale che aiuta a tracciare il risultato nel dettaglio di ogni transazione.



### Quali tipi di transazioni esistono?

Si hanno due tipi, uno è la transazione "offline" che si crea senza la necessità di avere la connessione alla rete principale di Ethereum può essere memorizzata fino a quando non si sceglie di connettersi alla rete di Ethereum e rilasciare la transazione, hanno il vantaggio della sicurezza perché l'intera transazione viene elaborata offline che evita qualsiasi anomalia che potrebbe essere nella connessione di rete. L'altra transazione è quella "online", che deve sempre essere collegata a Internet con i vantaggi e gli svantaggi che ne derivano in termini di sicurezza.

### Cos'è un indirizzo Blockchain?

Un indirizzo o un conto è composto da tre parti, l'indirizzo, la chiave pubblica e la chiave privata, queste due chiavi sono una stringa di numeri e caratteri in formato esadecimale che vengono utilizzati per inviare e ricevere (attivo) o etere (valuta digitale).

La chiave primaria non deve mai essere condivisa con nessuno in quanto è quella che autorizza il rilascio del saldo (firma le transazioni) detenuto sul conto.

La chiave pubblica è nota a tutto il pubblico ed è condivisa con chiunque in quanto è il riferimento per confermare che la transazione è corretta sia in termini di valore che a chi la invia.

Esempi di componenti di gestione della rete Ethereum:

```
{  
  "private": "429a043ea6393b358d3542ff2aab9338b9c0ed928e35ec0aed630b93adb14a1c",  
  "public":  
    "049b4b7e72701a09d3ee09165bba460f2549494a9d9fd7a95aaac57c2827eac162fd9e105b  
    2461cd6594ca8ca6a8daf10fe982f918be1b0060c87db9cfbcd289a8",  
  "address": "88ab6dcecc3603c7042f4334fc06db8e8d7062d5"  
}
```



### 13. Cos'è la programmazione Blockly?

**Blockly** è una **metodologia di programmazione visiva** composta da un semplice insieme di comandi che possiamo combinare come se fossero i pezzi di un puzzle. È uno strumento molto utile per chi vuole **imparare a programmare** in modo intuitivo e semplice o per chi sa già programmare e vuole vedere le potenzialità di questo tipo di programmazione. Si basa sul linguaggio JavaScript ed è stato sviluppato dalla società Google e dal MIT.

Blockly è una forma di programmazione in cui non è necessario alcun background in nessun tipo di linguaggio informatico, questo perché si tratta solo di unire blocchi grafici come se stessi giocando a lego o a un puzzle, basta avere un po' di logica ed è tutto!

Chiunque può creare programmi per cellulari (smartphone) senza pasticciare con quei linguaggi di programmazione difficili da capire, basta mettere insieme i blocchi in modo grafico in modo semplice, facile e veloce da creare.

### 14. Allegato "Codice per l'algoritmo CUA".

Riferimento a Github: <https://github.com/coinsolidation/source>

### 15. Termini e condizioni.

Termini e condizioni d'uso vedi sul sito [www.coinsolidation.org](http://www.coinsolidation.org) o <https://github.com/coinsolidation/Terms>

Supporto con uso commerciale.

[support@coinsolidation.org](mailto:support@coinsolidation.org)

Vendita blockchain uso commerciale.

[sales@coinsolidation.org](mailto:sales@coinsolidation.org)

Informazioni legali e domande o dubbi sulla licenza

[legal@coinsolidation.org](mailto:legal@coinsolidation.org)

I social network:

Twitter: <https://twitter.com/ecoinsolidation>

Facebook: <https://www.facebook.com/coinsolidation>