



EXchange
tensions

COINsolidação.

Tallinn, Estónia. (E-residência)

White Paper - Livro Branco.

versão 1.0.0

Dezembro de 2020.

COINsolidation.org é uma marca registada de COINsolidation International, sob licença de uso livre e comercial. Termos e condições de utilização em: www.Coinsolidation.org

COINsolidation International fundiu-se com www.OpenQbit.com para cooperação tecnológica baseada na mecânica quântica (Quantum Security & Quantum Computing). Esta fusão permite a utilização, partilha e reengenharia da tecnologia desenvolvida pela OpenQbit Inc. (Estónia, E-residência)

Conteúdo

1. Introdução.....	3
2. Segurança Quantum Computing.....	6
3. Criação de um dispositivo "Hardware" de um QRNG (Quantum Random Number Generator). 11	
4. O que é a Prova de Quantum (PQu)?.....	17
5. Algoritmo para a criação de um endereço universal consolidado (CUA)	20
6. Algoritmo para endereço duplo consolidado (DAC) e (HAC).	21
Projecto e solução por COINsolidation.	23
7. Criação da App CUA (Consolidated Universal Address) em 15 minutos.....	24
8. Crie o seu Ethereum crypto de troca de moeda no Android em apenas 15 minutos.	28
9. Consolidação de COINsolidação do roteiro.....	31
10. Ficha de Consolidação de COINsolidação (CUAG) - PLANO DE DISTRIBUIÇÃO DE ICO.	32
11. Características gerais do símbolo de COINsolidação:	33
12. Conceitos básicos aplicados em plataformas de Blockchain.	35
13. O que é a programação Blockly?.....	38
14. Anexo "Código para o algoritmo CUA".	38
15. Termos.....	38

1. Introdução.

Actualmente, as fusões estão actualizadas, quer se trate de um bem económico, tecnológico ou de mercado.

Apresentamos o primeiro modelo de fusão criptográfica ou crypto-tokens que oferece um backup entre duas criptomonas, fichas ou uma mistura destas, com base num algoritmo para criar um endereço consolidado que é utilizado e gerado no ambiente de COINsolidação.

Criámos três tipos de endereços consolidados.

O **CUA** (Consolidated Universal Address) é utilizado para consolidar e criar uma nova ficha (activa) para ser utilizada pelo utilizador. A combinação pode ser de três tipos: Cryptocurrency-Cryptocurrency, Cryptocurrency-Token ou Token-Token. No caso da CUA, é formada por uma relação Token-Token.

O **HAC** (Hibric Address Consolidated) é utilizado quando é necessário consolidar um endereço relativo a uma moeda criptográfica e/ou ficha e um endereço normal para a transferência de bens.

O **DAC** (Dual Address Consolidated) é utilizado para gerir e consolidar dois endereços normais a partir da mesma cadeia de blocos ou de duas tecnologias diferentes.

Começemos por olhar para as vantagens da CUA.

Um endereço CUA consiste no endereço simbólico de COINsolidation (endereço estático) e um código adicional conhecido como "Moeda Colorida" (endereço variável). Neste caso, podemos ver que os endereços CUA serão sempre formados por endereços de algum tipo de combinação de bens (Cryptosolids ou tokens).

No nosso caso, quando consolidarmos a ficha de COINsolidação e uma ficha OAP, conheceremos como a "Génese CUA" ou **CUAG (Consolidated Universal Address Genesis)**.

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

A ficha "Moeda colorida" é baseada e criada pela *cadeia de blocos Bitcoin* e utiliza o padrão Open Assest Protocol (**OAP**).

Vamos começar a rever qual é o potencial e o benefício de consolidar endereços.

- I. Para os utilizadores que criarem uma CUA será possível criar um token (**OAP**) que pode ser personalizado pelo utilizador que criou a CUA, o utilizador terá a possibilidade de ter o seu próprio token ou crypto activo para que o possa utilizar na

criação, suporte ou expansão do(s) seu(s) negócio(s), de uma forma simples e fácil terá uma vantagem no mundo dos crypto-tokens.

- II. Para as empresas que criam uma CUA podem ter um token **(OAP)** que podem utilizar para criar valor na sua cadeia de fornecimento ou utilizar o activo em transacções de liquidez com base no apoio económico dos activos e passivos da sua empresa.
- III. Para as criptomonedas e fichas existentes através da criação de uma CUA, poderão utilizar o seu endereço que identifica o seu bem e ao consolidá-lo com a ficha **(OAP)** poderão aumentar a sua procura oferecendo aos seus actuais e futuros investidores a sua própria ficha aos seus utilizadores.

Exemplo de **CUAG**, temos os endereços respectivos de dois Blockchain diferentes:

Endereço Bitcoin- Token - (OAP).

akXma4vqxvmEqnVAKSM953wYsnjNBhN3GM7

Endereço Ethereum - Token COINsolidation - (ERC20).

0x8390f8abb8fd8ad3bf8457db59f2ed75e015d303

Aplicando um algoritmo para consolidar os endereços anteriores, obtemos o endereço CUA.

cua50d0615d303k8X3m9a04fv8qaxbvb8Efqdn8VaAdK3SbMf985435w7Ydsbn5j9NfB2heNd37G5Me70

* Para mais detalhes do algoritmo consulte a secção 7.- "Algoritmo para a criação de um endereço universal consolidado".

Temos como resultado uma única direcção que representa duas tecnologias diferentes de duas direcções diferentes consolidadas numa única direcção.

Reflectimos isto no campo da rentabilidade e da expansão financeira de uma forma simples e directa, investindo numa das fichas que integra a nossa CUA imediatamente obterá uma ficha baseada na cadeia de bloqueio Bitcoin (OAP).

Agora vejamos duas directrizes que também apresentamos em COINsolidation para o mundo das criptomonedas e/ou fichas.

COINsolidation token é o projecto para consolidar endereços e ter um apoio imediato na obtenção de um token personalizado para utilizar no crescimento de cada utilizador no mundo das criptomonedas. O projecto nasceu em 2018 com um grupo de engenheiros e financeiros interessados em fundir os sectores financeiro e tecnológico, para tirar partido dos fundos de investimento e ocupar tecnologia inovadora como a computação quântica para dar segurança aos activos, bem como o objectivo de utilizar ferramentas que pudessem estar disponíveis para todos.

Após uma avaliação de várias possibilidades de desenvolvimento, escolhemos a opção da metodologia de programação Visual Blockly. Esta metodologia baseia-se na utilização de extensões ou módulos (programas em linguagem de programação java) com funcionalidades simples mas poderosas para expandir o negócio em activos criptográficos para qualquer pessoa, para o conseguir temos de cobrir os seguintes pontos:

- ✓ a.- ROI financeiro imediato para os utilizadores, investidores e activos, podendo criar um activo não tangível (fichas pessoais) para uso exclusivo do criador e utilizador do (CUA)
- ✓ b.- Utilizamos as vantagens de juntar duas cadeias de bloqueio na selecção do utilizador para fazer crescer os investimentos actuais e futuros no mercado activo de criptografia utilizando o (CUA).
- ✓ c.- Facilitar a administração de endereços separados, consolidando-os em (DAC).
- ✓ d.- Criar e utilizar segurança com base na Quantum Computing.

O desafio começou na criação da tecnologia de extensões suficientemente modulares em funcionalidade e "tamanho" este último foi o desafio da equipa de desenvolvimento da COINsolidation desde as extensões que são utilizadas na metodologia Blockly e sistemas deste tipo (AppInventor, AppyBuilder, Thunkable, Kondular, etc) são geralmente extensões (programas) criados que não excedem 100k - 300k bytes, com as restrições que têm no seu tamanho a tarefa de criar extensões para utilização na Blockchain actual eram virtualmente impossíveis devido ao facto de as bibliotecas que são utilizadas na sua criação excederem entre 10MB e 35MB estes tamanhos para as ferramentas actuais Os sistemas Blockly não são funcionais para a sua utilização.

A equipa teve de criar, adaptar e minimizar a metodologia de programação e as bibliotecas a fim de obter as extensões com a funcionalidade, segurança e tamanho óptimos.

Após quase dois anos de desenvolvimento e testes, terminámos a primeira cadeia de bloqueio "beta" utilizando extensões para Blockly incluindo o algoritmo de consenso "Proof of Quantum" utilizando segurança quântica para a troca de criptomónias.

Actualmente, temos uma cadeia de bloqueio proprietária que foi lançada para testes "Beta" e no final de 2021 estaremos a lançar a versão de produção para distribuição de informação. Actualmente a nossa Ficha de Consolidação de COINsolidation baseia-se nas cadeias de bloqueio Ethereum e Bitcoin, esta última para a criação de fichas personalizadas para os utilizadores.

2. Segurança Quantum Computing.

Como funciona a computação quântica? ⁽²⁾

A transformação digital está a provocar mudanças no mundo mais rapidamente do que nunca. Acreditaria que a era digital está prestes a terminar? A **literacia digital** já foi identificada como uma área onde o conhecimento aberto e as oportunidades acessíveis de aprender sobre a tecnologia são urgentes para colmatar lacunas no desenvolvimento social e económico. Aprender com os conceitos-chave da era digital tornar-se-á ainda mais crítico com a chegada iminente de outra nova onda tecnológica capaz de transformar os modelos existentes com velocidade e potência espantosas: **as tecnologias quânticas**.

Neste artigo, comparamos os conceitos básicos da computação tradicional e da computação quântica; e também começamos a explorar a sua aplicação em outras áreas relacionadas.

O que são tecnologias quânticas?

Ao longo da história, os seres humanos desenvolveram a tecnologia à medida que compreendiam como a natureza funciona através da ciência. Entre 1900 e 1930, o estudo de alguns fenómenos físicos que ainda não eram bem compreendidos deu origem a uma nova teoria física, a **Mecânica Quântica**. Esta teoria descreve e explica o funcionamento do mundo microscópico, o habitat natural das moléculas, átomos ou electrões. Graças a esta teoria, não só foi possível explicar estes fenómenos, como também foi possível compreender que a realidade subatômica funciona de uma forma completamente contra-intuitiva, quase mágica, e que no mundo microscópico ocorrem eventos que não ocorrem no mundo macroscópico.

Estas **propriedades quânticas** incluem sobreposição quântica, enredamento quântico e teleportação quântica.

- A **sobreposição quântica** descreve como uma partícula pode estar em diferentes estados ao mesmo tempo.
- O **enredamento quântico** descreve como duas partículas tão distantes quanto desejado podem ser correlacionadas de tal forma que, ao interagirem com uma, a outra está ciente disso.
- O **teletransporte quântico** utiliza emaranhamento quântico para enviar informação de um lugar para outro no espaço sem ter de viajar através dele.

As tecnologias quânticas baseiam-se nestas propriedades quânticas da natureza subatômica.

Neste caso, hoje em dia, a compreensão do mundo microscópico através da Mecânica Quântica permite-nos inventar e conceber tecnologias capazes de melhorar a vida das pessoas. Existem muitas e muito diferentes tecnologias que utilizam fenómenos quânticos e

algumas delas, tais como lasers ou ressonância magnética (MRI), estão connosco há mais de meio século. Contudo, estamos actualmente a assistir a uma revolução tecnológica em áreas como a computação quântica, informação quântica, simulação quântica, óptica quântica, metrologia quântica, relógios quânticos ou sensores quânticos.

O que é computação quântica? Primeiro, é preciso compreender a computação clássica.




FIGURA 1.
Ejemplos de
caracteres
en lenguaje
binario.

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

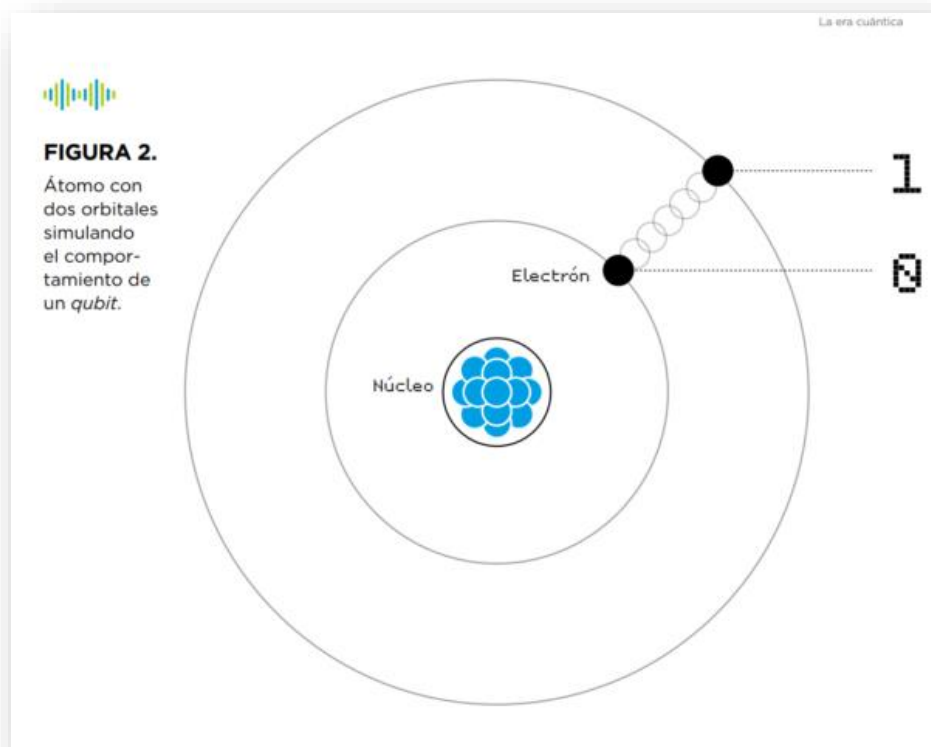
Para compreender como funcionam os computadores quânticos, é conveniente explicar primeiro como funcionam os computadores que utilizamos todos os dias, que iremos referir neste documento como computadores digitais ou clássicos. Estes, tal como os restantes dispositivos electrónicos, tais como comprimidos ou telemóveis, utilizam bits como unidades fundamentais de memória. Isto significa que os programas e aplicações são codificados em bits, ou seja, em linguagem binária de zeros e uns. Sempre que interagimos com qualquer um destes dispositivos, por exemplo, premindo uma tecla no teclado, são criadas, destruídas e/ou modificadas cordas de zeros e uns dentro do computador.

A questão interessante é, o que são estes zeros e uns fisicamente dentro do computador? O zero e um estados correspondem à corrente eléctrica que circula, ou não, através de peças microscópicas chamadas transístores, que actuam como interruptores. Quando não está a fluir corrente, o transistor está "desligado" e corresponde ao bit 0, e quando está a fluir está "ligado" e corresponde ao bit 1.

Mais simplesmente, é como se os bits 0 e 1 correspondessem a buracos, de modo que um buraco vazio é um bit 0 e um buraco ocupado por um electrão é um bit 1. Como exemplo, a figura 1 mostra a escrita binária de alguns caracteres. Agora que temos uma ideia de como funcionam os computadores de hoje, vamos tentar compreender como funcionam os quantum.

De bits a qubits

A unidade fundamental de informação na computação quântica é o bit quântico ou qubit. Os Qubits são, por definição, sistemas quânticos de dois níveis - veremos aqui exemplos - que, tal como os bits, podem estar no nível baixo, que corresponde a um estado de baixa excitação ou energia definida como 0, ou no nível alto, que corresponde a um estado de maior excitação ou definido como 1. No entanto, e aqui reside a diferença fundamental com a computação clássica, as desistências podem também estar em qualquer dos infinitos estados intermédios entre 0 e 1, tais como um estado que é meio 0 e meio 1, ou três quartos de 0 e um quarto de 1.



Algoritmos quânticos, exponencialmente mais potentes e eficientes de computação

O objectivo dos computadores quânticos é tirar partido destas propriedades quânticas dos *qubits*, como sistemas quânticos que são, a fim de executar algoritmos quânticos que utilizam sobreposição e intercalação para fornecer um poder de processamento muito maior do que os clássicos. É importante salientar que a verdadeira mudança de paradigma não consiste em fazer o mesmo que os computadores digitais ou clássicos - os actuais - mas sim mais rápido, como se pode ler em muitos artigos, mas que os algoritmos quânticos permitem realizar certas operações de uma forma totalmente diferente que em muitos casos se revela mais eficiente - ou seja, em muito menos tempo ou utilizando muito menos recursos computacionais -.

Vejamos um exemplo concreto do que isto envolve. Imaginemos que estamos em Bogotá e queremos saber qual é a melhor rota para chegar a Lima de entre um milhão de opções para lá chegar ($N=1,000,000$). A fim de utilizar computadores para encontrar o melhor caminho, precisamos de digitalizar 1.000.000 opções, o que implica traduzi-las em linguagem bit para o computador clássico e em *qubits* para o computador quântico. Enquanto um computador clássico teria de ir um a um analisando todos os caminhos até encontrar o desejado, um computador quântico tira partido do processo conhecido como paralelismo quântico que lhe permite considerar todos os caminhos ao mesmo tempo. Isto implica que, enquanto o computador clássico necessita da ordem de passos $N/2$ ou iterações, ou seja, 500.000 tentativas, o computador quântico encontrará o caminho óptimo depois de apenas \sqrt{N} operações no registo, ou seja, 1.000 tentativas.

No caso anterior a vantagem é quadrática, mas noutros casos é mesmo exponencial, o que significa que com n *qubits* podemos obter uma capacidade computacional equivalente a 2^n bits. Para exemplificar isto, é comum contar que com cerca de 270 qubits poderíamos ter mais estados base num computador quântico - mais cadeias de caracteres diferentes e simultâneas - do que o número de átomos no universo, que é estimado em cerca de 10^{80} . Outro exemplo é que se estima que com um computador quântico entre 2000 e 2500 *qubits* poderíamos quebrar praticamente toda a criptografia utilizada hoje em dia (a chamada criptografia de chave pública).

Porque é importante saber sobre a tecnologia quântica?

Estamos num momento de transformação digital em que diferentes tecnologias emergentes, tais como cadeias de bloqueio, inteligência artificial, drones, Internet das coisas, realidade virtual, impressoras 3D, robôs ou veículos autónomos, têm cada vez mais presença em múltiplos campos e sectores. Estas tecnologias, chamadas a melhorar a qualidade de vida do ser humano acelerando o desenvolvimento e gerando impacto social, avançam hoje em dia de uma forma paralela. Só raramente vemos empresas a desenvolver produtos que exploram combinações de duas ou mais destas tecnologias, tais como a cadeia de bloqueios e a LPC ou os zangões e a inteligência artificial. Embora estejam destinados a convergir, gerando assim um impacto exponencialmente maior, a fase inicial de desenvolvimento em que se encontram e a escassez de promotores e pessoas com perfis técnicos significam que a convergência ainda é uma tarefa pendente.

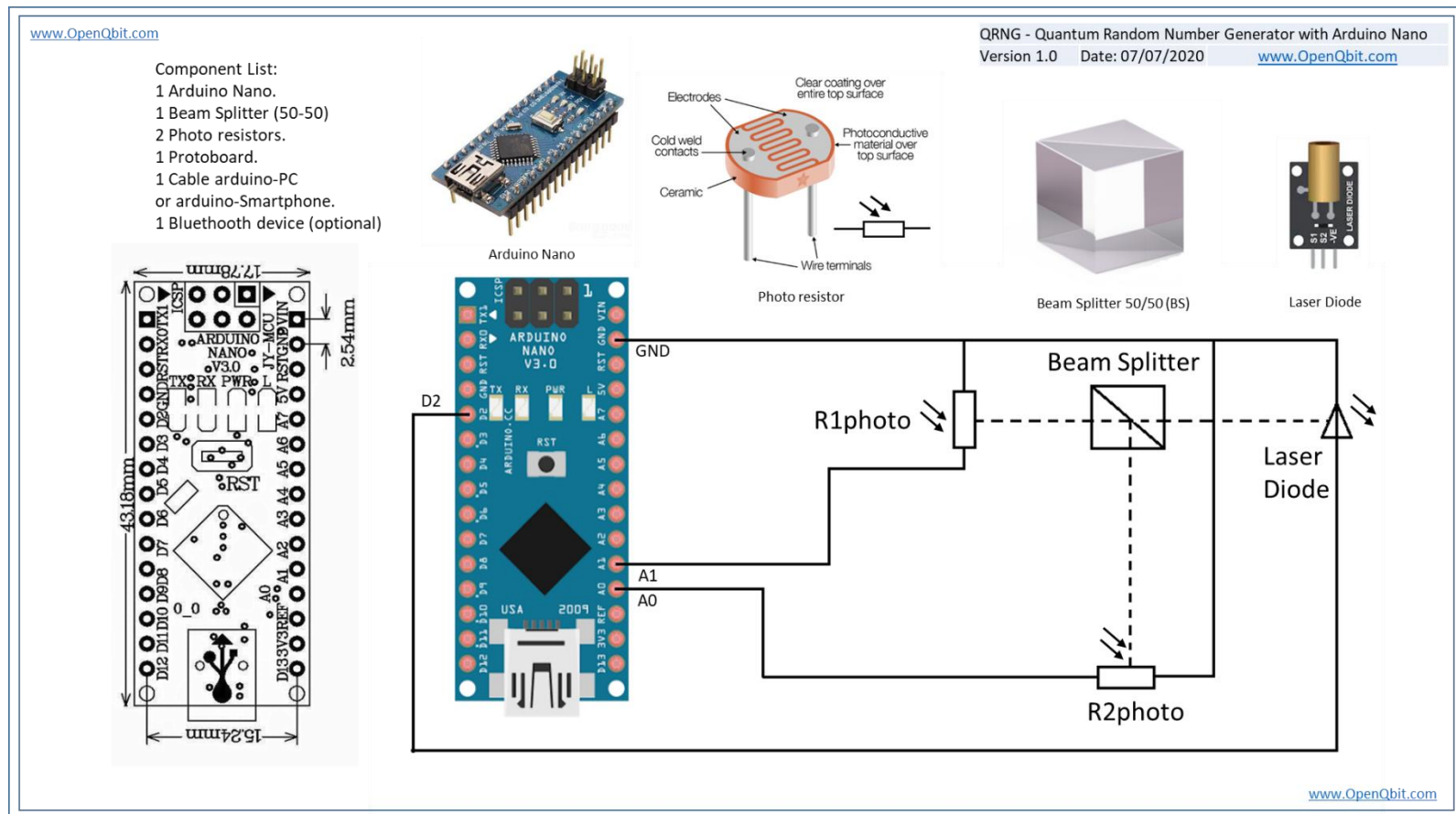
Devido ao seu potencial disruptivo, espera-se que as tecnologias quânticas não só converjam com todas estas novas tecnologias, mas que tenham uma influência transversal em praticamente todas elas. A computação quântica ameaçará a autenticação, troca e armazenamento seguro de dados, tendo um grande impacto nas tecnologias em que a criptografia tem um papel mais relevante, como a segurança cibernética ou a cadeia de bloqueio, e um impacto negativo menor, mas também a ser considerado em tecnologias como a 5G, IoT ou drones.

Quer praticar computação quântica?

Dezenas de simuladores quânticos de computador já estão disponíveis na rede com diferentes linguagens de programação já em uso tais como C, C++, Java, Matlab, Maxima, Python ou Octave. Também, novas línguas como o Q#, lançado pela Microsoft. Pode explorar e jogar com uma máquina quântica virtual através de plataformas como a IBM e a Rigetti.

3. Criação de um dispositivo "Hardware" de um QRNG (Quantum Random Number Generator).

Vamos agora criar um dispositivo físico "Hardware" para gerar Números Quânticos Aleatórios (QRNG) com componentes baratos que podem ser facilmente montados em casa e custam aproximadamente \$35 USD.



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);
}
```

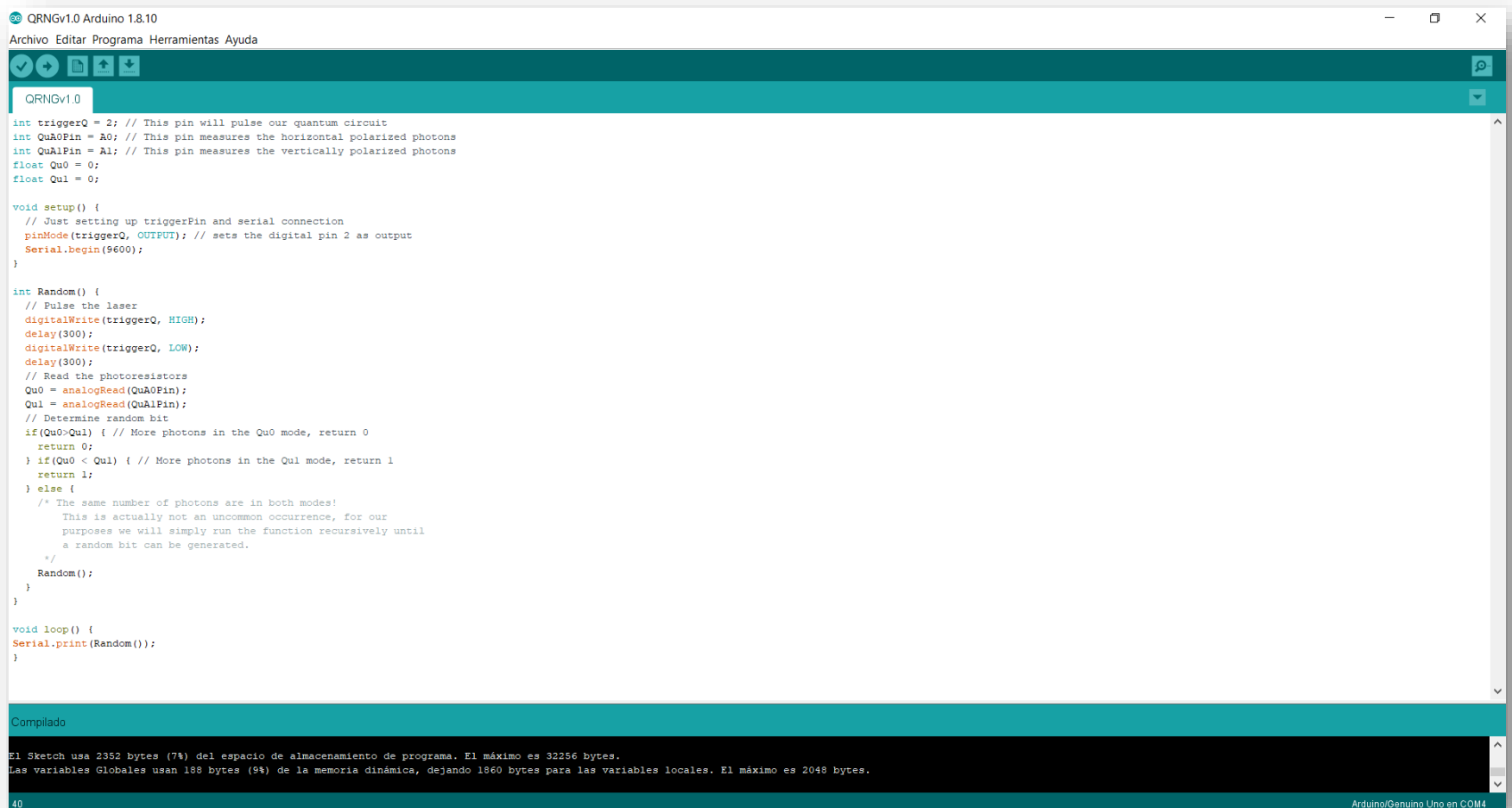
```
int Random() {
  // Pulse the laser
  digitalWrite(triggerQ, HIGH);
  delay(300);
  digitalWrite(triggerQ, LOW);
  delay(300);
  // Read the photoresistors
  Qu0 = analogRead(QuA0Pin);
  Qu1 = analogRead(QuA1Pin);
  // Determine random bit
  if(Qu0>Qu1) { // More photons in the Qu0 mode, return 0
    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.....

Compilando o programa QRNGv10.ino e carregando para arduino nano....



The screenshot shows the Arduino IDE interface with the sketch 'QRNGv1.0' open. The code defines two pins for measuring polarized photons and a function to generate a random bit based on their readings. The compilation output at the bottom indicates the sketch size is 2352 bytes (7% of 32256 bytes) and global variables use 188 bytes (9% of 2048 bytes).

```
QRNGv1.0 Arduino 1.8.10
Archivo Editar 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());
}
```

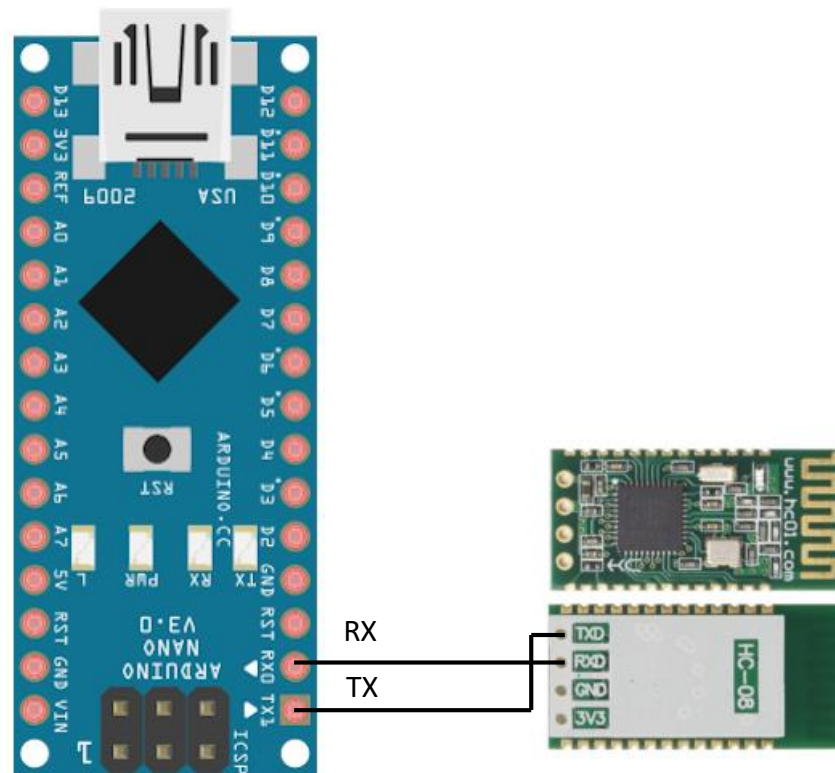
Compilado

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.

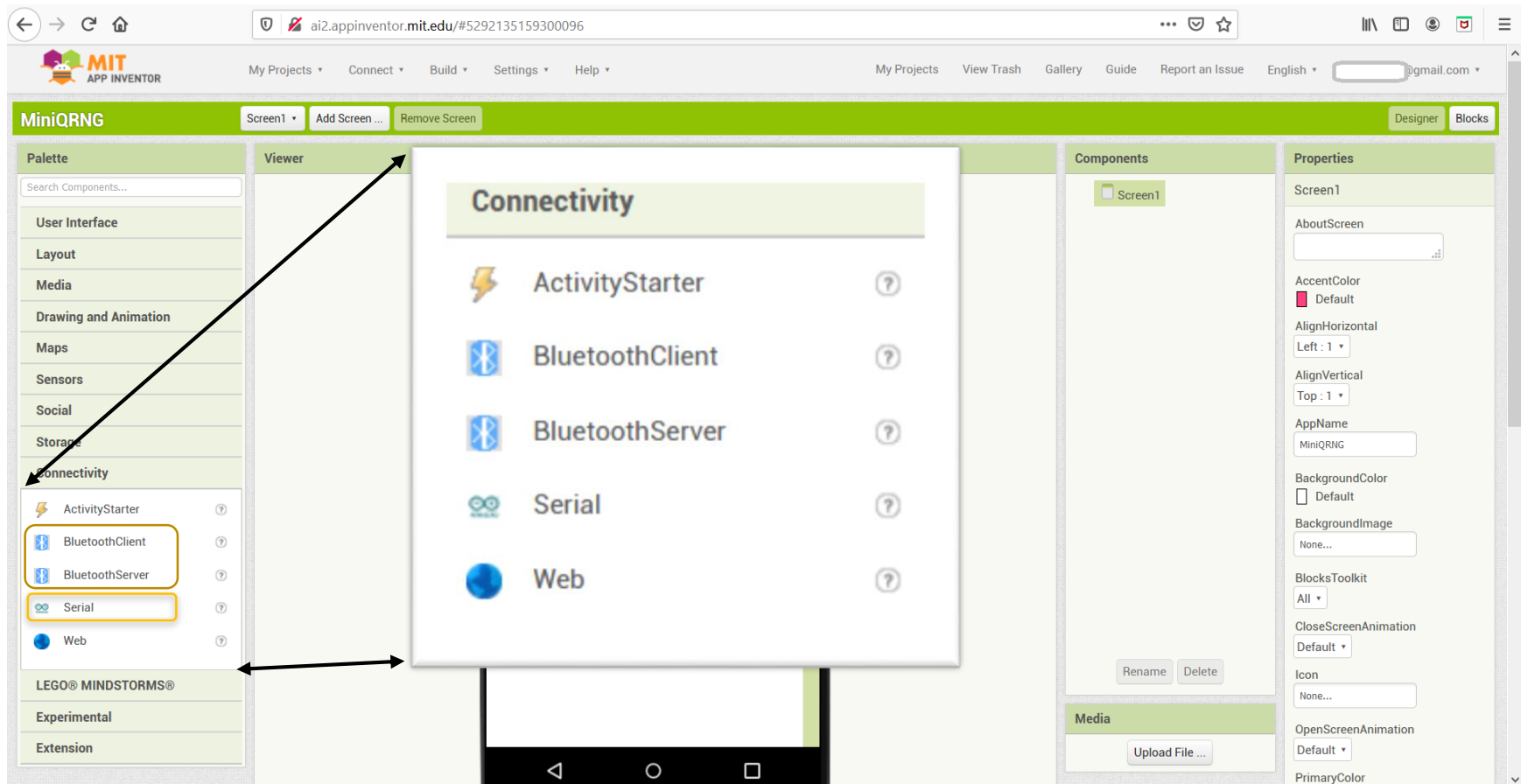
40 Arduino/Genuino Uno en COM4

Há duas formas de comunicar com o nano árduo, uma é através da porta Serial e a outra é através de uma ligação Bluetooth.

Para a ligação bluetooth é muito simples, só temos de comprar o módulo HC-08 ou um semelhante e ligá-lo como se segue:



Os seguintes componentes Serial ou Bluetooth podem ser utilizados para ligar o App Inventor ao Arduino:





Agora compilado e carregado o programa QRNGv10.ino só falta comunicar com o nano árduo para guardar os dados (números quânticos aleatórios) estes serão em formato binário, no entanto, os dados obtidos podem ser facilmente passados para outro formato, tal como hexadecimal ou decimal, dependendo do requisito final.

Finalmente, para ver um exemplo de como funciona a ligação em série ou Bluetooth, aqui estão algumas ligações de referência.

Lembre-se que tudo é através de programação Blockly para ser testado com App Inventor este já tem blocos para comunicação com arduino serial ou outro sistema do tipo blockly pode ser através de bluetooth online similar.

http://kio4.com/appinventor/9A0_bluetooth_RXTX.htm

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

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

Rever todo o projecto de concepção e utilização de extensões QRNG (Quantum Random Number Generator). Rever o manual do utilizador em:

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



4. O que é a Prova de Quantum (PQu)?

PoQu. - "Proof of Quantum" é um algoritmo de consenso desenvolvido para Mini BlocklyChain e COINsolidation, este teste é uma variante do Proof of Work (PoW) que funciona da seguinte forma.

O Teste de Quantum (PoQu) no arranque é executado com o mesmo algoritmo que o "Teste de Trabalho" (PoW) baseia-se em colocar o processador do dispositivo (PC, Servidor, Tablet ou Telemóvel) a trabalhar para obter uma sequência de caracteres que é um puzzle matemático chamado "hash".

Lembre-se que um "hash" é um algoritmo ou processo matemático que ao introduzir uma frase ou algum tipo de informação digital tal como ficheiros de texto, programa, imagem, vídeo, som ou outro tipo de informação digital diversa nos dá como resultado um carácter alfanumérico que representa a assinatura digital que a representa de uma forma única e irrepetível dos dados, o algoritmo hash é unidireccional, isto significa que quando se introduz um dado para obter a sua assinatura "hash" o seu processo inverso não pode ser executado, tendo uma assinatura "hash" não podemos saber que informação foi obtida esta propriedade dá-nos uma vantagem de segurança para processar a informação que enviamos através da Internet. Como funciona? Imagine enviar qualquer tipo de informação através de canais não seguros e acompanhá-la com o seu respectivo "hash de origem", o receptor ao receber a informação pode obter o "hash" da informação recebida chamar-lhe-emos "hash de destino" e verificá-lo com o "hash de origem" se ambos os "hashes" forem os mesmos podemos confirmar que a informação não foi alterada no canal que foi enviado, é apenas um exemplo onde este tipo de processo de segurança da informação é actualmente utilizado.

Actualmente existem diferentes tipos de algoritmos ou processos de haxixe que diferem no nível de segurança. Os mais utilizados ou conhecidos são: MD5, SHA256 e SHA512.

Exemplo de SHA256:

Temos uma cadeia ou frase como se segue: "A Mini BlocklyChain é modular.

Se aplicarmos um hash do tipo SHA256 à corda anterior, ele dar-nos-á o próximo hash.

f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db8



A cadeia alfanumérica acima é a assinatura que representa a frase no exemplo acima

Para mais exemplos, podemos utilizar o sítio na Internet:

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

No caso do algoritmo "Test Work" (PoW), funciona usando a potência de computação para obter um hash pré-definido.

Imaginemos que temos o "hash" anterior que retirámos da cadeia "Mini BlocklyChain é modular".

f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db8

A este "hash" no seu início colocamos o parâmetro de dificuldade que é simplesmente colocar zeros "0" no início, ou seja, se dissermos que a dificuldade é de 4 terá "0000" + "hash" a isto chamar-lhe-emos "hash de semente".

0000 f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db8

Agora tendo em conta que sabemos a informação de entrada que é a cadeia: "Mini BlocklyChain é modular" adicionamos no final da cadeia um número que começa em zero "0" e retiramos o seu hash a este, chamar-lhe-emos "hash nonce":

f41af7e61c3b02fdd5e5c612302b62a2dd52fcb38f9de97cb2afd827e8804db80

Temos hash nonce:

7529f3ad273fc8a9eff12183f8d6f886821900750bb6b59c1504924dfd85a7c8

Depois realizamos uma comparação do novo "hash nonce" com o "hash seed" se forem iguais o nó que primeiro encontrar a igualdade ganhará a execução do processamento da transacção actual. Como podemos ver este processo baseia-se na probabilidade e força computacional do dispositivo que dá ao teste "Prova de Trabalho" uma equidade consensual para todos os nós.

Se o "hash semente" não coincide com o "hash nonce", a dificuldade é aumentada em um e o "hash nonce" é novamente removido, o número que está a ser aumentado é chamado de "nonce", é comparado com o "hash semente" até que coincidam ou sejam o mesmo.

Como podemos ver o número "nonce" ou aumento é o que ajudará a obter o "hash" da igualdade.



Baseado no algoritmo "Test of Work" (PoW), o algoritmo do Teste Quântico (PoQu) baseia-se na obtenção do número "nonce" como o PoW e utilizando um dificuldade de nível mínimo que vai de 1 a 5, isto serve apenas para ganhar o direito do dispositivo móvel a ser um candidato para ganhar consenso.

O Teste Quantum (PoQu), é activado quando o telemóvel termina o PoW mínimo e ganha o passe para obter um número de probabilidade no sistema QRNG.

O QRNG (Quantum Random Number Generator) é um Gerador de Números Quânticos Aleatórios, este sistema baseia-se na geração de números verdadeiramente aleatórios com base na mecânica quântica é o sistema mais seguro hoje em dia para gerar tais números. Para mais detalhes ver "Quantum Computation Security" no índice 3.

A COINsolidação pode implementar ambos os tipos de concessões mínimas PoW e PoQu.

O teste PoQu é baseado na obtenção do número "nonce". Este número no teste PoQu é conhecido como "Número Mágico" e com isto o sistema Peer to Peer confirmará se o número está correcto e então um número aleatório será obtido com o pool de servidores COINsolidation QRNG. Este número aleatório será registado em todos os nós, será criada uma lista contendo **((Soma do nó /2)) +1** e desta lista será escolhida a que tiver a maior percentagem de probabilidade de ser o candidato vencedor do consenso (PoQu) e este executará a fila de transacções em curso.

O algoritmo PoQu também utiliza testes **NIST** (National Institute of Standards and Technology) para nos assegurar que os números aleatórios no QRNG são números verdadeiramente aleatórios.

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

Na COINsolidation implementámos um bloco para PoW e um bloco para PoQu. Estes blocos utilizam um tipo de hash: SHA256 para uso livre, para uso comercial temos um SHA512 e outros hashes conforme necessário.

Para mais detalhes sobre o conceito de HASH, ver:

https://es.wikipedia.org/wiki/Funcion_hash

NOTA: O Teste de Trabalho (PoW) utilizado em telemóveis só pode utilizar uma dificuldade máxima de 5, uma vez que o processamento matemático destes dispositivos não é dedicado como servidores ou PCs. Utilizamos apenas o algoritmo PoW para obter a oportunidade de obter o seu passe ou permissão para entrar no sistema Quantum Random Number Generator (QRNG) e com ele executar o algoritmo Quantum Random Number Generator (PoQu). Ver utilização de (PoQu) em Mini BlockyChain:



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

5. Algoritmo para a criação de um endereço universal consolidado (CUA)

Os nossos endereços CUA (Consolidated Universal Address) são criados utilizando o seguinte algoritmo:

Passo 1.- Os identificadores são removidos dos respectivos endereços, são os caracteres alfanuméricos que identificam o endereço a partir do qual a cadeia de bloqueio foi criada.

Endereço Bitcoin- Token - (OAP).

akXma4vqxvmEqnVAKSM953wYsnjNBhN3GM7

Endereço Ethereum - Token COINsolidation - (ERC20).

0x9d08c0ac0f2fdf078c883db6fa617b15776e4b41

Passo 2.- O SHA512(endereço String-Text) de cada endereço sem o seu identificador inicial é obtido removendo o "a" de A1 e o "0x" de A2 e tirando os dois últimos caracteres de cada operação de hash simbolizada com "U". Números de verificação.

$U(\text{SHA512}(\text{kXma4vqxvmEqnVAKSM953wYsnjNBhN3GM7})) = 50$

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

Passo 3.- Os caracteres de cada endereço são concatenados um a um a partir do endereço que tem menos caracteres que o compõem, no caso de ter a mesma quantidade de caracteres a concatenação pode começar a partir de qualquer endereço.

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

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

Concatenação de endereços:

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

****Os últimos caracteres que não podem ser concatenados são colocados no início da corda.**

6e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777

Passo 4.- O número de caracteres que poderiam ser concatenados no passo 3 é adicionado ao início da cadeia resultante do passo 3.

66e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777



Passo 5.- Os dois pares de verificadores do passo 2 de cada direcção são concatenados no início da cadeia resultante do passo 3 na mesma ordem A1 + A2.

50fb66e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777

Passo 6.- A identificação **CUAG** (Consolidated Universal Address Genesis) é integrada no início do endereço criado no passo 5.

cua50fb66e4b41k9Xdm0a84cv0qaxcv0mfE2qfndVfA0K7S8Mc985833wdYbs6nfjaN6B1h7Nb31G5M777

** No caso da consolidação de endereços Bitcoin e Ethereum dará um endereço composto por **82** caracteres hexadecimais.

6. Algoritmo para endereço duplo consolidado (DAC) e (HAC).

A criação de um DAC é a mesma que a CUA, a diferença é que nos DACs são utilizados para consolidar endereços normais para receber transacções, estes endereços não representam nenhuma criptomoneda ou ficha.

Passo 1.- Os identificadores são removidos dos respectivos endereços, são os caracteres alfanuméricos que identificam o endereço a partir do qual a cadeia de bloqueio foi criada.

18gYNA9c2G9X8HZ8QxWLpLXZauAxFnsJbe (Endereço Bitcoin)

0x5d2Acdb34c279Aa6d1e94a77F7b18aB938BFb2bB (Direcção Ethereum)

Passo 2.- O SHA512(endereço String-Text) de cada endereço sem o seu identificador inicial é obtido removendo o "**1**" de A1 e o "**0x**" de A2 e tirando os dois últimos caracteres de cada operação de hash simbolizados com "U". Números de verificação.

U(SHA512(8gYNA9c2G9X8HZ8QxWLpLXZauAxFnsJbe)) = **48**

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

Passo 3.- Os caracteres de cada endereço são concatenados um a um a partir do endereço que tem menos caracteres que o compõem, no caso de ter a mesma quantidade de caracteres a concatenação pode começar a partir de qualquer endereço.

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

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



Concatenação de endereços:

$A_{10} [0] + A_{20} [0] + A_{10} [1] + A_{20} [1] + A_{11} [2] + A_{22} [2] + \dots A_{1N+1} [n+1] + A_{2N+2} [n+1]$

******Os últimos caracteres que não podem ser concatenados são colocados no início da corda.

8BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3

Passo 4.- O número de caracteres que poderiam ser concatenados no passo 3 é adicionado ao início da cadeia resultante do passo 3.

78BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3

Passo 5.- Os dois pares de verificadores do passo 2 de cada direcção são concatenados no início da cadeia resultante do passo 3 na mesma ordem A1 + A2.

48f378BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3

Passo 6.- A identificação **DAC** (Dual Address Consolidated) é integrada no início do endereço criado no passo 5.

dac48f378BFb2bB85gdY2NAAc9dcb23G49cX287H9ZA8aQ6xdW1Lep9L4XaZ7a7uFA7xbF1n8saJBb9e3

****** No caso da consolidação de endereços Bitcoin e Ethereum dará um endereço composto por **81** caracteres hexadecimais.

No caso de HAC (Hibric Address Consolidated) é aplicado no anterior o que varia são os endereços que são utilizados, neste caso utilizaremos um endereço que representa um bem (Cryptomonedra ou token) e um endereço padrão normal de transferências de bens de algum tipo de cadeia de bloqueio.

NOTA: O tamanho dos endereços CUA, HAC e DAC podem variar em cada caso, dependendo dos endereços que os compõem.



Projecto e solução por COINsolidation.

Actualmente existem diferentes tipos de Blockchain orientados para bens de características diferentes, o que leva a ter um número infinito de tipos de endereços de uso diário têm de manter um controlo apertado para evitar cometer erros na transferência.

Por outro lado, o mundo da criptomoney e das fichas está limitado aos peritos financeiros ou, no seu caso, aos peritos em tecnologia de cadeias de bloqueio, pelo que é difícil para uma pessoa média aventurar-se na criação da sua própria criptomoney ou ficha.

Resolvemos os dois problemas anteriores na consolidação de COINsolidation, fazendo os seguintes pontos e/ou ferramentas que criámos.

Para o ponto de controlo de endereços de diferentes cadeias de bloqueio, criámos um algoritmo onde consolida (junta) dois ou mais endereços nas suas diferentes combinações dando como resultado um único endereço do tipo CUA, HAC e/ou DAC.

Com esta solução, em vez de enviar dois endereços da mesma ou diferente cadeia de bloqueio, será utilizado apenas um endereço consolidado.

Para o segundo problema utilizámos a metodologia de programação chamada Blockly, trata-se de uma ferramenta visual onde não é necessário um grande conhecimento de programação e qualquer pessoa ou empresa média será capaz de criar as suas próprias aplicações sem ter de investir equipas de desenvolvimento dispendiosas, tempo e dinheiro.

Criámos as extensões (módulos) para apenas as instalar e utilizar para criar aplicações móveis, em 15 minutos. Exemplo a sua própria troca de moeda criptográfica ou desenvolver a sua própria moeda (token) em minutos. Tudo isto utilizando a segurança de dados de última geração chamada PQC (Post-Quantum Cryptography).

Basta instalar as extensões em qualquer ferramenta gratuita como Appventor, AppyBuider, Thunkable, Kondular ou outras e em minutos pode entrar no mundo das criptoménias e da criação de fichas, tudo na palma da sua mão.

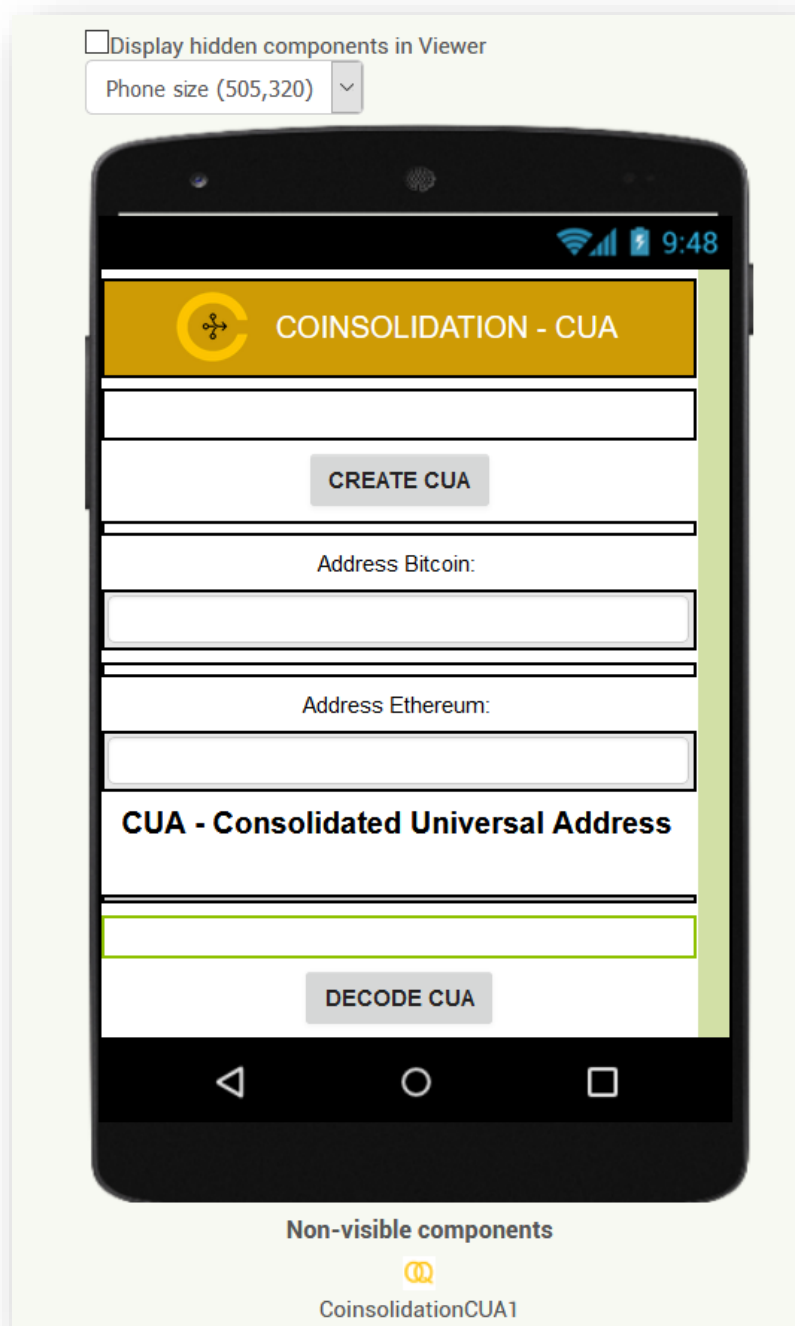
Finalmente, a COINsolidação está a criar a utilização de segurança quântica de baixo custo (software e hardware) que pode ser utilizada para proteger os dados informáticos em casa. Actualmente, as tecnologias baseadas em computação quântica e segurança têm custos elevados que só empresas com um elevado nível financeiro as podem criar e utilizar. No entanto, na COINsolidation acreditamos que as novas tecnologias devem estar disponíveis para todos, a justiça de utilização da Blockchain e da Quantum Computing deve ser para todos, criamos software livre (criptomónica) e hardware de baixo custo (segurança quântica).



7. Criação da App CUA (Consolidated Universal Address) em 15 minutos.

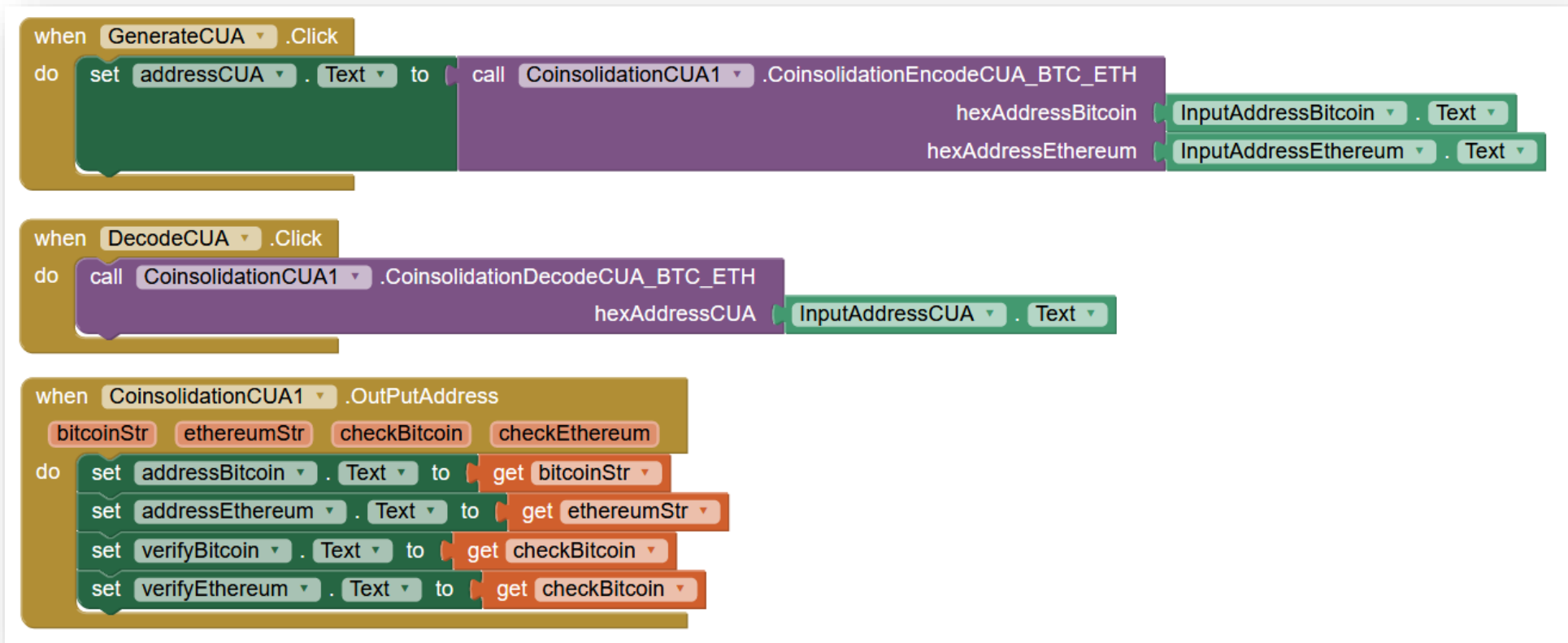
*App para moedas Bitcoin e Ethereum (BTC-ETH)

Ecrã de design 5 minutos em <https://appinventor.mit.edu/>





Utilização da extensão **CoinsolidatioCUA.AIX** (5 minutos).





Criamos a aplicação em **Menu > Construir > Aplicação** (fornecer código QR para .apk) - (5 minutos).

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

do

bitcoinStr

ethereumStr

checkBitcoin

checkEthereum

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%

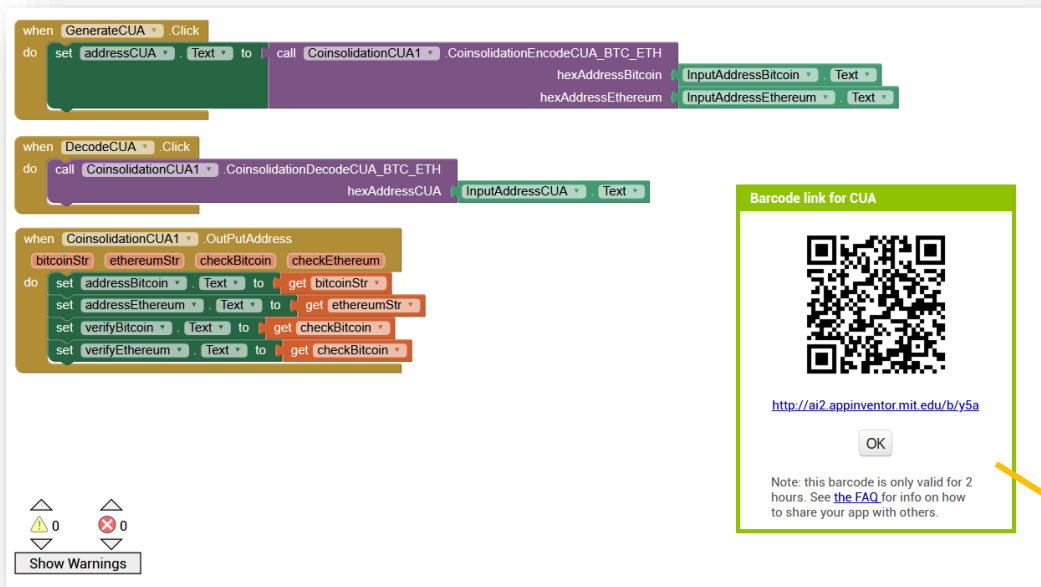
Compiling part 2 (please wait)

+

-



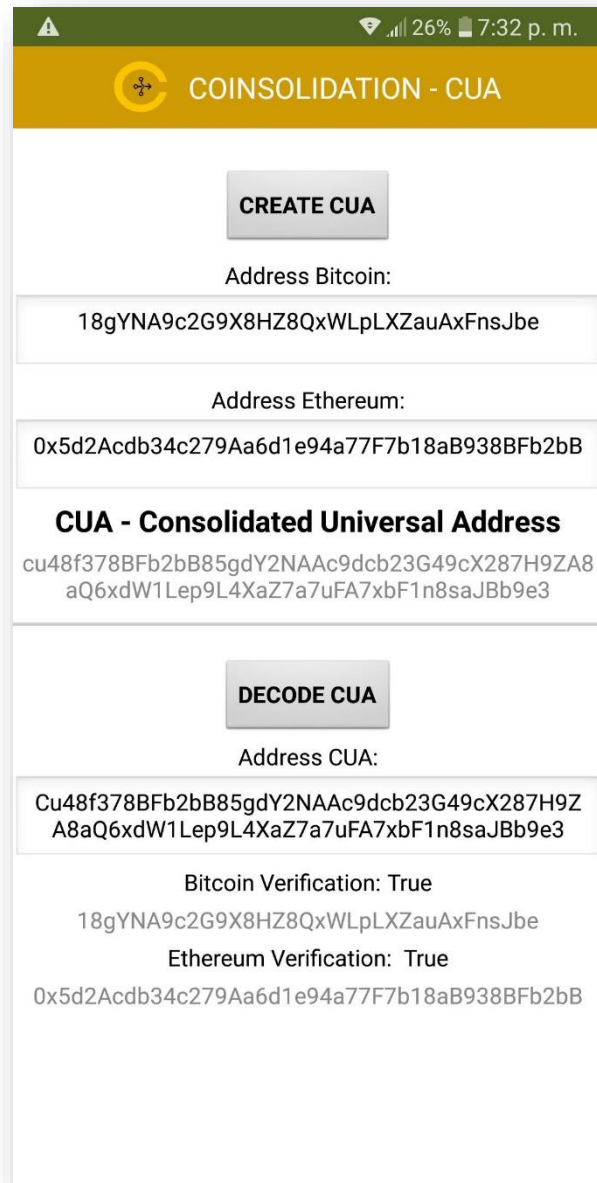
Instalámos a aplicação no telemóvel a partir do QR utilizando a aplicação Android da AppInventor (MIT AI2 Companion) - <https://play.google.com/store/apps/details?id=edu.mit.appinventor.aicompanion3>



NOTA: A aplicação de ficheiro APK pronta a ser instalada está localizada no seguinte repositório:

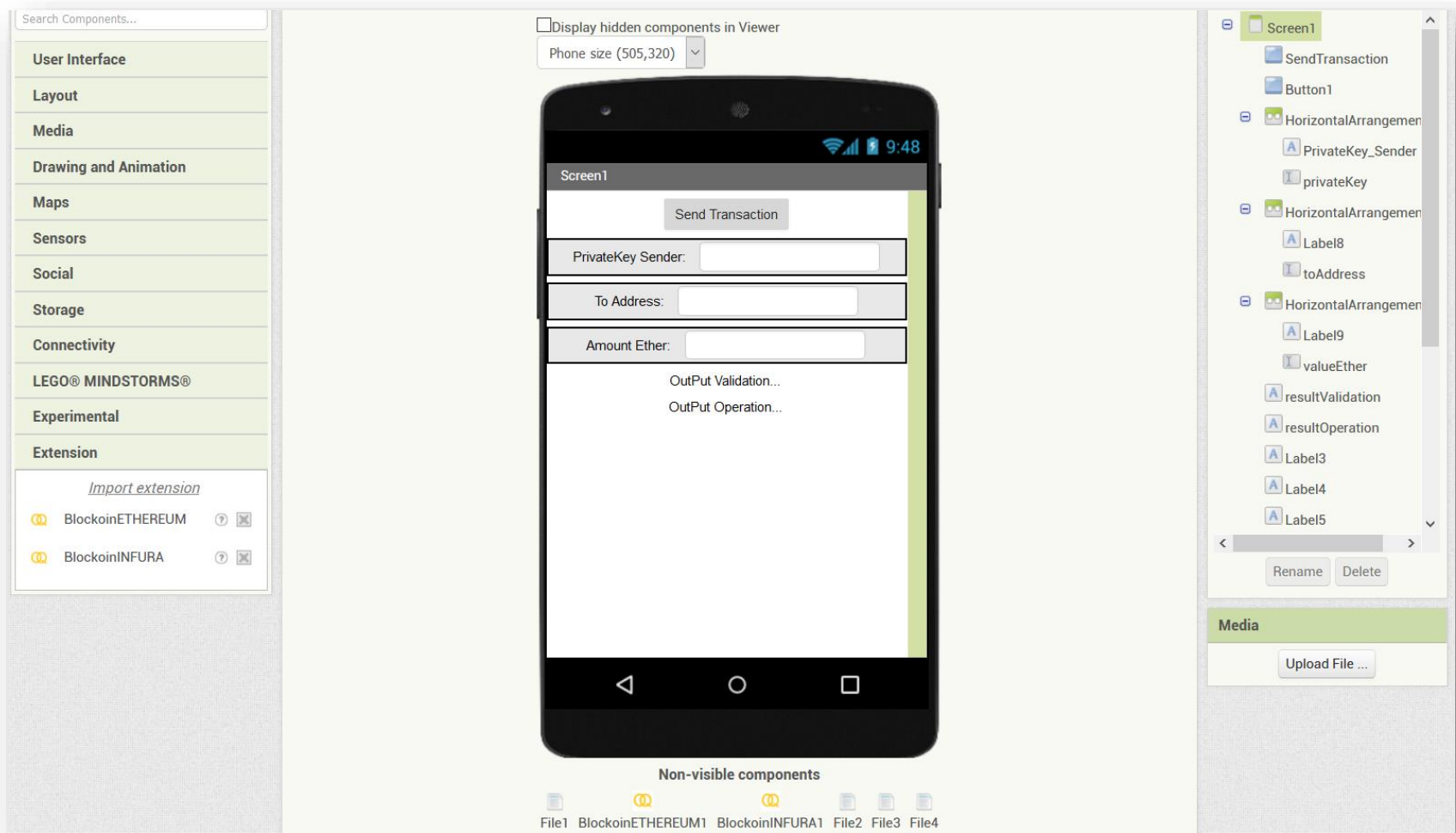
<https://github.com/COINsolidation/App>

Para rever o Código Java para geração de extensão CUA e implementar um algoritmo consolidado de geração de endereços universais, rever o Anexo "Code for CUA algorithm" ou consultar o link do código: <https://github.com/COINsolidation/source>



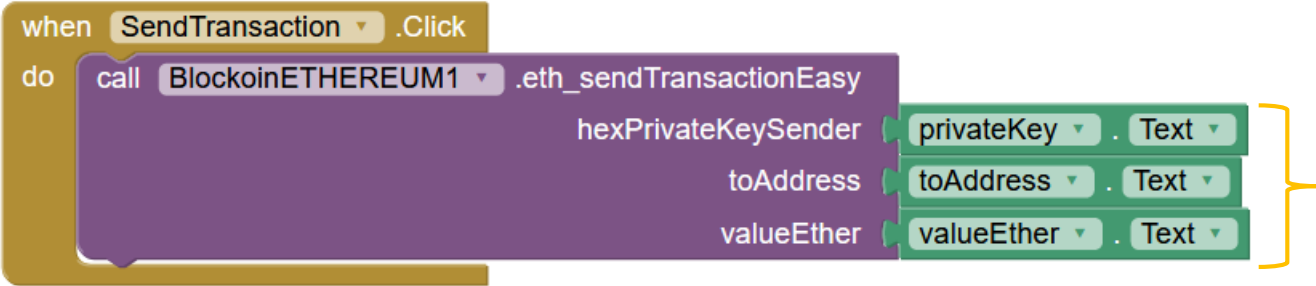


8. Crie o seu Ethereum crypto de troca de moeda no Android em apenas 15 minutos.
Desenho em App Inventor (Screen). - 5 minutos.





Blocos de funções (eth_SendTransactionEasy) e evento (OutPutSendTransactionEasy) - 5 minutos

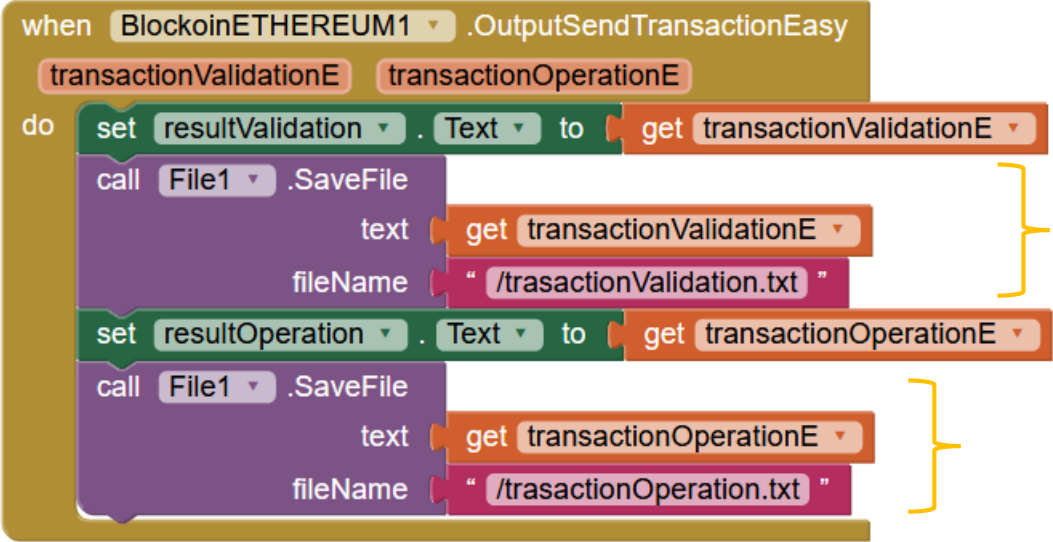


Dados de entrada:

PrivateKey: Chave primária para o endereço do remetente.

Endereço: Endereço hexadecimal do destinatário.

valorEther: Dê a quantidade de Éter que será enviada.



Guardar os resultados em ficheiros de texto:
Função File1: File **trasactionValidation.txt**

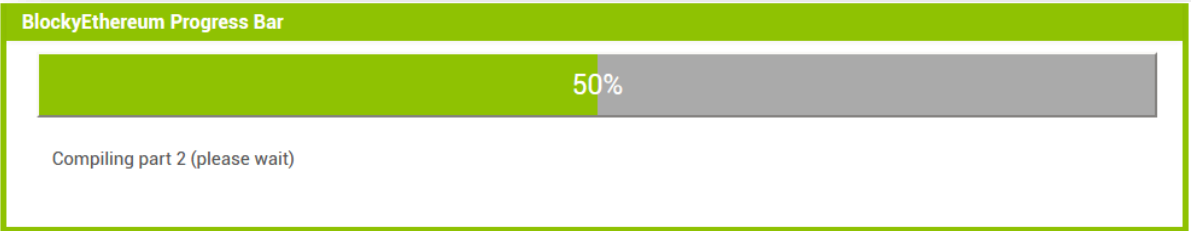
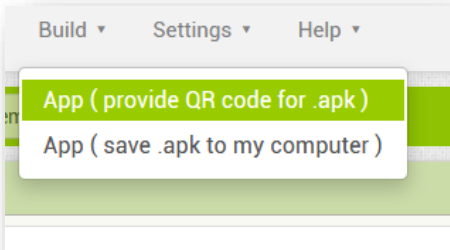
Guardar os resultados em ficheiros de texto:
Função File2: File **trasactionValidation.txt**

**Outros detalhes ver Guia do Utilizador Ethereum Exchange (EEE) Extensão no repositório: <https://github.com/COINsolidation/userguide>

**Repositorio de extensiones COINsolidation: <https://github.com/coinsolidation/Extesions-Cryptocurrencies> o OpenQbit (Blockchain & Quantum Computing) <https://github.com/openqbit-diy>



Compilamos, geramos o ficheiro APK para o instalar no dispositivo Android. - 5 minutos



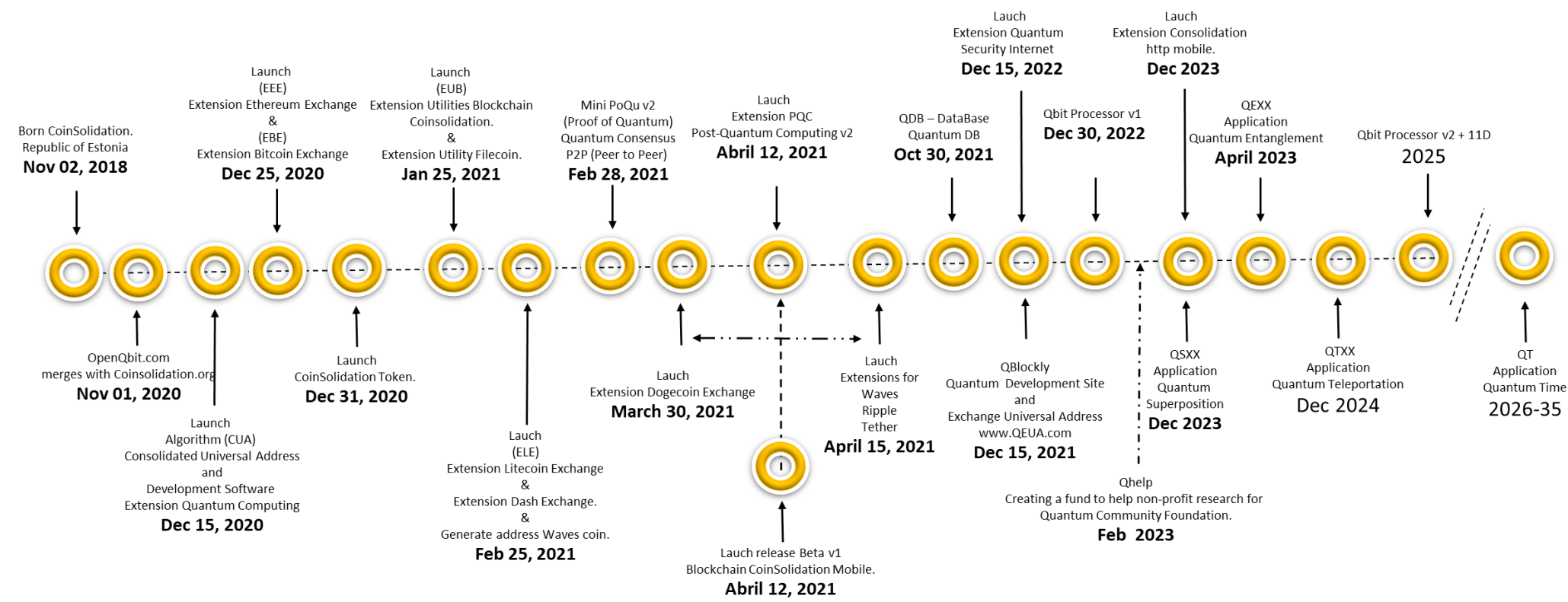
NOTA: Quando a transacção for executada, levará aproximadamente 6 a 8 segundos para libertar o botão "Enviar Transacção". Devido ao tempo de ligação com a rede Ethereum.

Para mais pormenores sobre a extensão do EEE - (Extensão do Ethereum Exchange). Ver o manual do utilizador do EEE no link: <https://github.com/COINsolidation/UserGuide>



9. Consolidação de COINsolidação do roteiro.

ROADMAP



*OpenQbit.com funde-se com COINsolidation.org (Nov 01, 2020) / OpenQbit é especializada em Quantum Computing e Security Quantum.
*O processador quântico versão 1 irá utilizar portões lógicos quânticos básicos para uso doméstico.



10.Ficha de Consolidação de COINsolidação (CUAG) - PLANO DE DISTRIBUIÇÃO DE ICO.

O ICO está dividido em três fases:

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: \$ 10,000,000 USD
ICO SECOND PHASE	\$ 0.15 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	1	400,000,000.00
OPENQBIT DEVELOPMENT AND RESEARCH OF QUANTUM COMPUTING	2	800,000,000.00
TOTAL SUPPLY 100%		40,000,000,000.00

0x9d08c0ac0f2fdf078c883db6fa617b15776e4b41	COINsolidation TOKEN
0xbbF57DE98c59B4C304C9d15BC5FAb01304aeCD97	ENDEREÇO ICO
0xa646c054394f85257E18D56Cf5c6b5E603447470	ENDEREÇO DA OPERAÇÃO COINsolidation



11. Características gerais do símbolo de COINsolidação:

Criado por: Lugu Samaya.

Nome: COINsolidation

Símbolo: CUAG - (Endereço Universal Consolidado Génesis).

Tipo: NFT

Total de fichas criadas: 40.000.000.000.000,00

Número de decimais: 18

País de lançamento: Estónia

Site oficial: www.COINsolidation.org

Empresa: COINsolidation International.

Data de lançamento: 30 de Dezembro de 2020

Algoritmo de consenso: PQu (Prova de Quantum)

Algoritmo de endereço: Endereço Universal Consolidado (CUA).

Segurança utilizada: PQC (Post-Quantum Cryptography) baseada em computação quântica.

Proposta tecnológica: Extensões dos sistemas Blockly para utilização de criptomonas e implementação de segurança quântica.



Parcerias ou acordos tecnológicos (fusão):

Empresa: OpenQbit Inc.

Indústria: Computação Quântica e PQC (Criptografia Pós-Quântica).

Sítio Web oficial: www.OpenQbit.com







12. Conceitos básicos aplicados em plataformas de Blockchain.

O que é uma cadeia de bloqueio?

A cadeia de bloqueio é geralmente associada à moeda Bitcoin e outras moedas criptográficas, mas estas são apenas a ponta do iceberg uma vez que não é apenas utilizada para dinheiro digital, mas pode ser utilizada para qualquer informação que possa ter um valor para os utilizadores e/ou empresas. Esta tecnologia, que tem as suas origens em 1991, quando Stuart Haber e W. Scott Stornetta descreveram o primeiro trabalho sobre uma cadeia de blocos criptografados, só foi notada em 2008, quando se tornou popular com a chegada do bitcoin. Mas actualmente a sua utilização está a ser exigida noutras aplicações comerciais e prevê-se que cresça no futuro médio em vários mercados, tais como instituições financeiras ou a Internet das Coisas da Internet, entre outros sectores.

A cadeia de bloqueio, mais conhecida pelo termo cadeia de bloqueio, é um registo único, acordado e distribuído por vários nós (dispositivos electrónicos tais como PCs, smartpohones, tablets, etc.) numa rede. No caso de moedas criptográficas, podemos pensar nisto como o livro de contabilidade onde cada uma das transacções é registada.

O seu funcionamento pode ser complexo de compreender se entrarmos nos detalhes internos da sua implementação, mas a ideia básica é simples de seguir.

É armazenado em cada bloco:

- 1.- uma série de registos ou transacções válidos,
- 2.- informações relativas a esse bloco,
- 3.- a sua ligação com o bloco anterior e o bloco seguinte através do hash de cada bloco –um código único que seria como a impressão digital do bloco.

Portanto, **cada bloco** tem um **lugar específico e inamovível dentro da cadeia**, uma vez que cada bloco contém informação do hash do bloco anterior. Toda a cadeia é armazenada em cada nó da rede que compõe a cadeia de bloqueio, pelo que **uma cópia exacta da cadeia é armazenada em todos os participantes da rede**.

O que é um endereço ou uma conta dentro da plataforma Ethereum da cadeia de bloqueio?

É uma cadeia de 42 caracteres na plataforma Ethereum, representando um número em base hexadecimal, onde os bens definidos no Ethereum serão depositados ou enviados. Em outras plataformas de cadeia de bloqueio, o número de caracteres da conta ou endereço pode ser diferente, por exemplo:



0x5d2Acdb34c279Aa6d1e94a77F7b18aB938BFb2bB

O que é uma kryptomoney?

É uma moeda digital ou virtual concebida para funcionar como um meio de troca. Utiliza criptografia (segurança digital) para garantir e verificar as transacções, bem como para controlar a criação de novas unidades de uma kryptomoney em particular.

O que é uma ficha?

As fichas são bens digitais que podem ser utilizados dentro de um determinado ecossistema de projecto.

A principal distinção entre fichas e moedas criptográficas é que as primeiras requerem outra plataforma de cadeia de bloqueio (não a sua própria) para funcionar. O Ethereum é a plataforma mais comum para a criação de fichas, principalmente devido à sua função de contrato inteligente. As fichas criadas na cadeia de bloqueio Ethereum são geralmente conhecidas como fichas ERC-20 embora existam outros tipos de fichas mais especializadas, tais como a ficha ERC-721 utilizada principalmente para bens coleccionáveis (cartões, utilização em jogos de vídeo, obras de arte, etc.).

O que é uma troca?

Uma troca de moedas criptográficas é o ponto de encontro onde se realizam trocas de moedas criptográficas em troca de fiat money ou outras moedas criptográficas. Nestas casas de câmbio online é gerado o preço de mercado que marca o valor das criptomónias com base na oferta e procura.

O que são taxas de câmbio?

Estas são as taxas do valor de um Éter ou outra moeda criptográfica na moeda em circulação de cada país. Por exemplo, no dia da criação deste manual, um Éter tem um valor em dólares americanos de \$430,94

O que é uma transacção?

É a execução ou transferência de algum tipo de activo não tangível que pode receber um valor pré-estabelecido dentro do sistema Ethereum e que pode mais tarde ser alterado para um valor tangível para uma empresa ou pessoa.

O que é txHash?

É um número hexadecimal que ajuda a rastrear o resultado em detalhe de cada transacção.



Que tipos de transacções existem?

Tem dois tipos, um é a transacção "offline" que isto cria sem a necessidade de ter ligação à rede principal do Ethereum pode ser armazenada até optar por se ligar à rede do Ethereum e libertar a transacção, tem a vantagem da segurança porque toda a transacção é processada offline o que evita qualquer anomalia que possa estar na ligação à rede. A outra transacção é a "online" que precisa sempre de estar ligada à Internet com as vantagens e desvantagens de segurança que ela traz.

O que é um endereço Blockchain?

Um endereço ou conta é composto por três partes, o endereço, a chave pública e a chave privada, estas duas chaves são uma cadeia de números e caracteres em formato hexadecimal que são utilizados para enviar e receber (activo) ou éter (moeda digital).

A chave primária nunca deve ser partilhada com ninguém, pois é a que autoriza a libertação do saldo (assina as transacções) mantido na conta.

A chave pública é conhecida de todo o público e é partilhada com qualquer pessoa, pois é a referência para confirmar que a transacção é correcta tanto em termos de valor como para quem a mesma é enviada.

Exemplos de componentes de gestão da rede Ethereum:

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



13.O que é a programação Blockly?

Blockly é uma **metodologia de programação visual** composta por um simples conjunto de comandos que podemos combinar como se fossem as peças de um puzzle. É uma ferramenta muito útil para aqueles que querem **aprender a programar de forma** intuitiva e simples ou para aqueles que já sabem programar e querem ver o potencial deste tipo de programação. É baseado na linguagem JavaScript e foi desenvolvido pela empresa Google e pelo MIT.

Blockly é uma forma de programação em que não é necessário qualquer tipo de linguagem informática, isto porque é apenas juntar blocos gráficos como se estivéssemos a jogar lego ou um puzzle, só é preciso ter alguma lógica e mais nada!

Qualquer pessoa pode criar programas para telemóveis (smartphones) sem se meter com as linguagens de programação difíceis de compreender, basta juntar blocos de forma gráfica de uma forma simples, fácil e rápida de criar.

14.Anexo "Código para o algoritmo CUA".

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

15.Termos.

Termos e condições de utilização ver no site www.coinsolidation.org ou <https://github.com/coinsolidation/Terms>

Apoio com uso comercial.

support@coinsolidation.org

Utilização comercial da cadeia de bloqueio de vendas.

sales@coinsolidation.org

Informação legal e questões ou preocupações sobre licenças

legal@coinsolidation.org

Redes sociais:

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

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