



# DELIVERY MANAGEMENT OPEN THE BOX

DEPARTAMENTO DE CIÊNCIAS DOS COMPUTADORES  
FACULDADE DE CIÊNCIAS DA UNIVERSIDADE DO PORTO  
27 DE MAIO DE 2018

200901635 José Querubim Rocha Reisinho

201305059 Simon Afonso

201404877 André Couto Meira

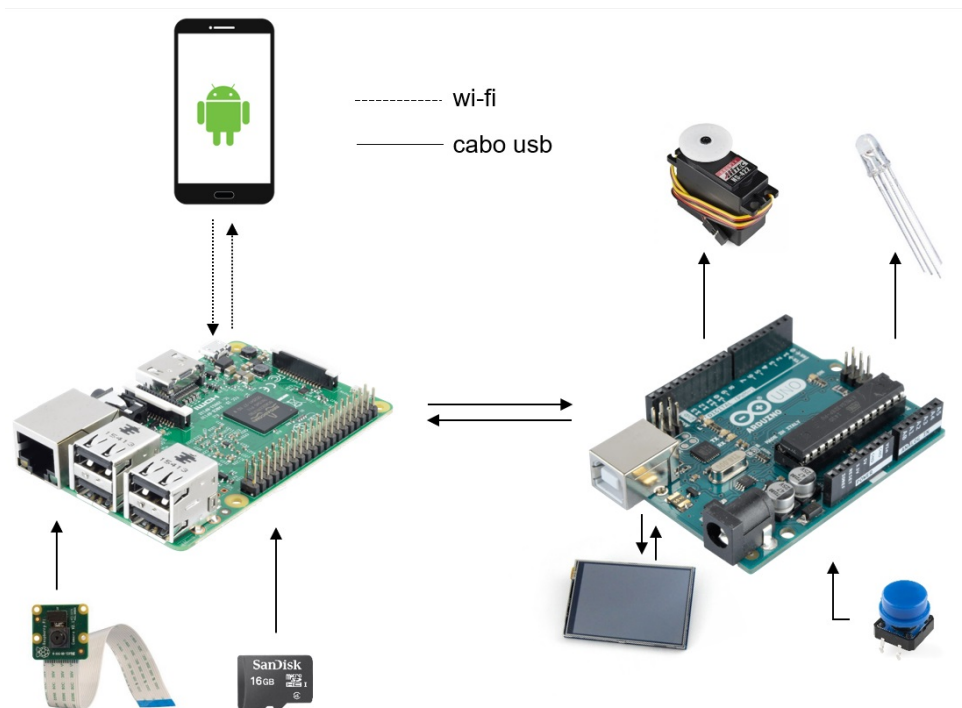
# 1 O Sistema

O objectivo deste trabalho é a criação de uma caixa de acesso restrito. A forma como se consegue é fornecendo a um terceiro um conjunto de códigos que poderão ser utilizados para a abertura da caixa sendo estes um código numérico de 4 dígitos, um código morse com 5 símbolos e um código QR.

É ainda possível que se controle remotamente o acesso à caixa, através de um controlador que neste caso é um telemóvel Android. Atráves de um toque é possível mandar abrir imediatamente a caixa sem que nenhum código seja introduzido. Para além disso é possível atualizar os códigos que são aceites pelo sistema.

Existe também um RaspberryPi que funciona como cérebro do sistema e dois Arduinos, Arduino One para I/O que controla um TFT e um outro, Arduino Two, que controla o motor que tranca/destranca a caixa, verifica se a caixa está aberta ou fechada e ainda que controla um led que indica a quem tenta abrir a caixa o estado da mesma.

A comunicação entre os componentes é feita através de cabo Serie/USB entre o RPi e os Arduinos, e por sockets sobre a rede/internet entre o Android e o RPi. Não existe comunicação directa entre os Arduinos e o Android. Outras alterações ao projecto podem ser vistas na wiki do moodle.



## 1.1 Protocolo de Comunicação

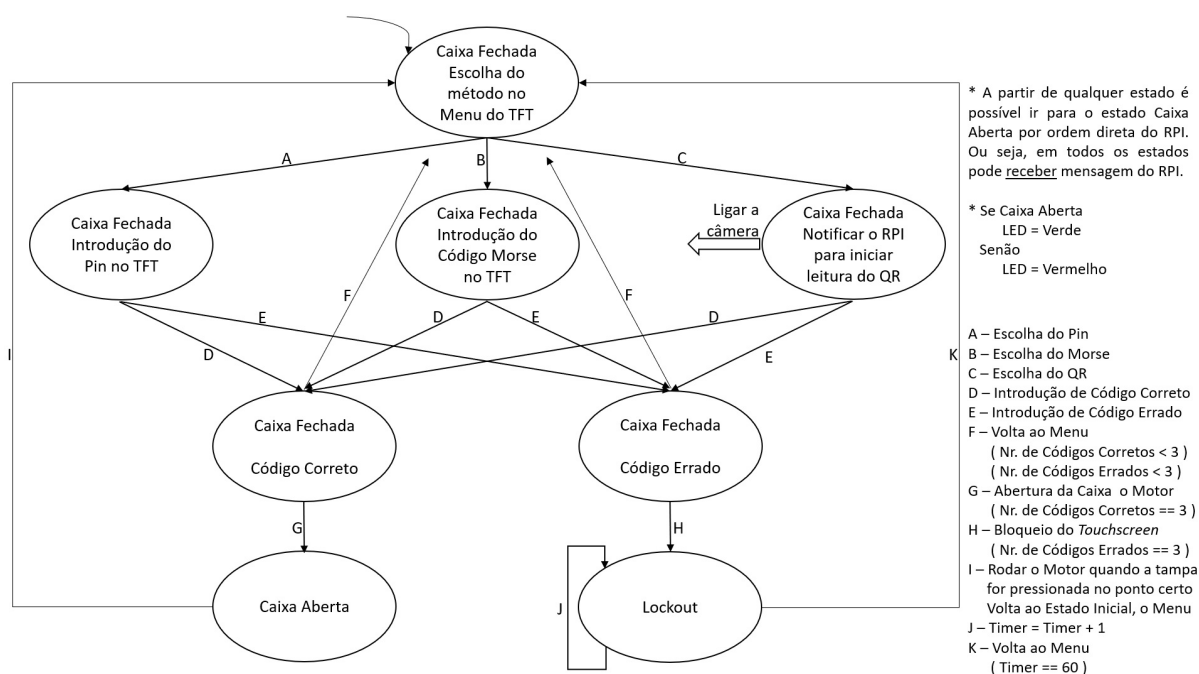
Os protocolos de comunicação entre os diferentes componentes estão explicados na wiki do moodle. As mensagens são curtas por causa da capacidade de computação do Arduino. Para além disso, este lê carácter a carácter. Ainda assim, as mensagens são suficientemente sucintas. Também para o Android as mensagens foram encurtadas para facilitar o envio através da rede.

## 1.2 Diagramas Atualizados

Segue-se um conjunto de diagramas que representam o que foi desenvolvido sobre o estado final previsto para o sistema. Nem todas as características foram desenvolvidas por motivos que nada tem a ver com problemas de implementação. Os diagramas surgem neste relatório pelas diferenças significativas relativamente aos projetados inicialmente.

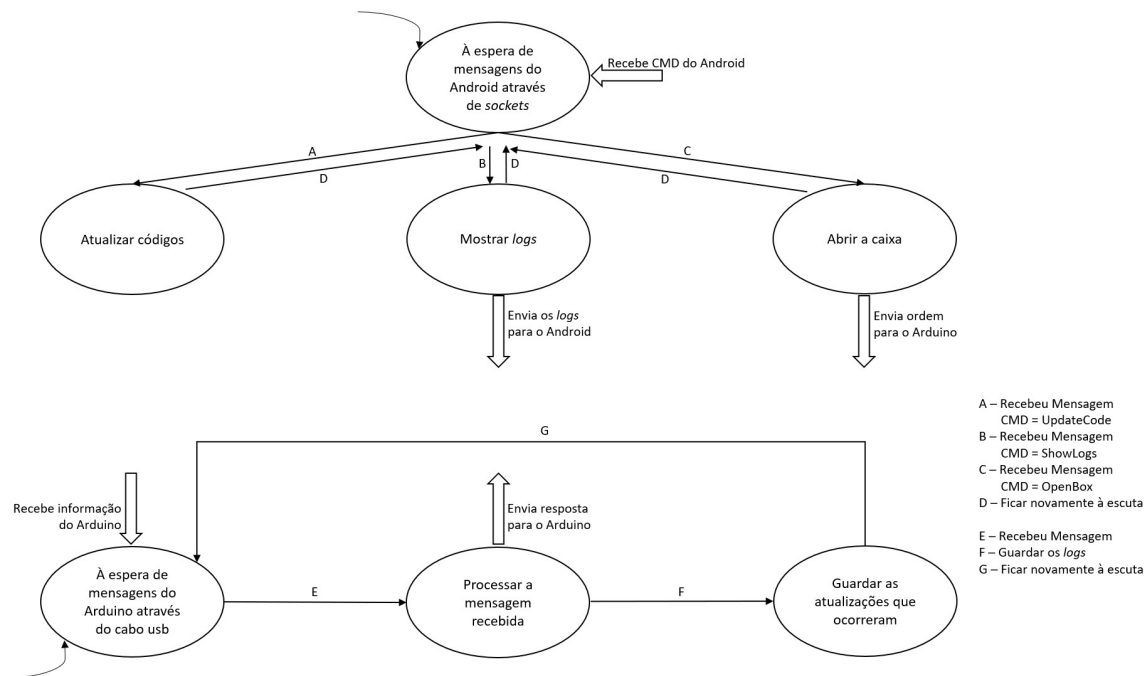
### 1.2.1 Arduino

Neste sistema são utilizados 2 Arduinos. Um controla um TFT que lhe ocupa todos os pins. Este é responsável pela seleção do código a introduzir seguido da leitura desse mesmo código. A exceção é a leitura do QR em que o Arduino apenas indica ao RPi que deve ligar a câmara. O outro controla todos os restantes componentes do sistema. É responsável pela manutenção da cor do led, a posição do motor que tranca a caixa, e por verificar se a tampa está ou não fechada para que seja possível trancá-la. Para saber se pode ou não destrancar a caixa comunica com o RPi do qual recebe ordens. Cada Arduino tem uma ligação directa ao RPi. Quando é necessário enviar mensagens entre Arduinos recorre-se ao RPi.



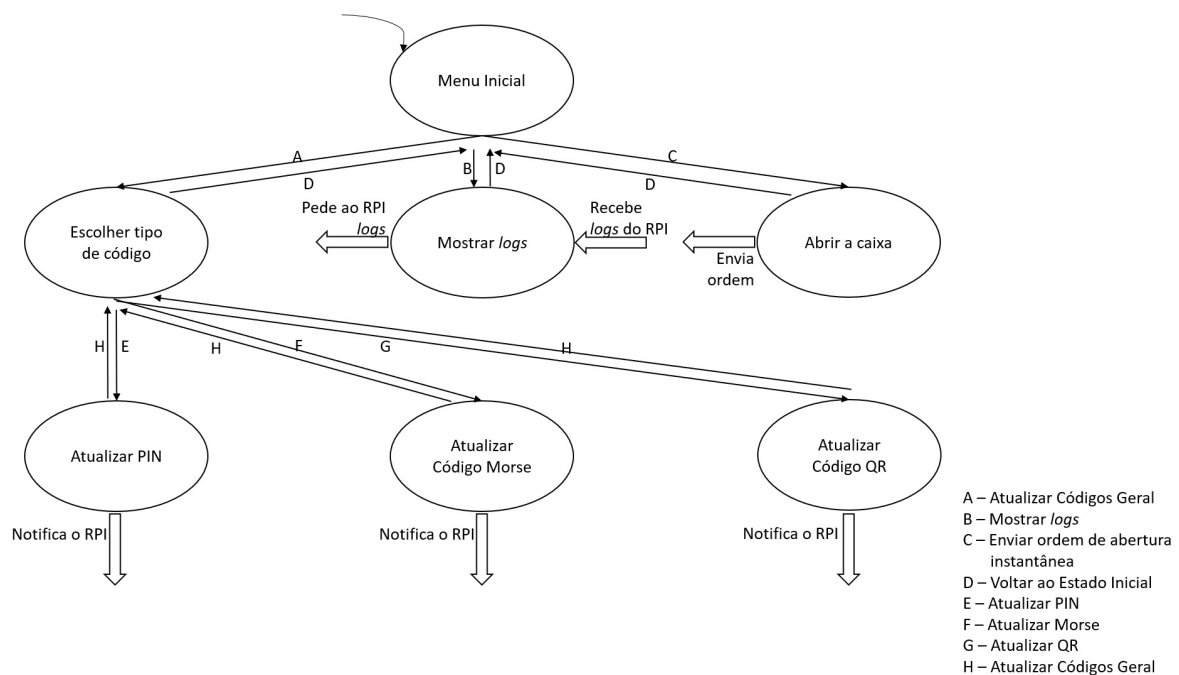
### 1.2.2 RaspberryPi

O RaspberryPi funciona como cérebro do sistema e serve de ponte de comunicação entre os Arduinos e o Android. Recebe comandos do controlador através de uma socket. Se necessário informa os Arduinos por cabo usb/porta serie. Recebe os códigos do Arduino One respondendo-lhe se é um código correto ou errado. Guarda o estado do sistema e guarda os códigos em utilização para que o mesmo possa saber se um código inserido é o correcto ou não e ainda a contagem de códigos corretos/errados inseridos. É ainda responsável por logar as interações com o sistema. Nem tudo foi implementado.

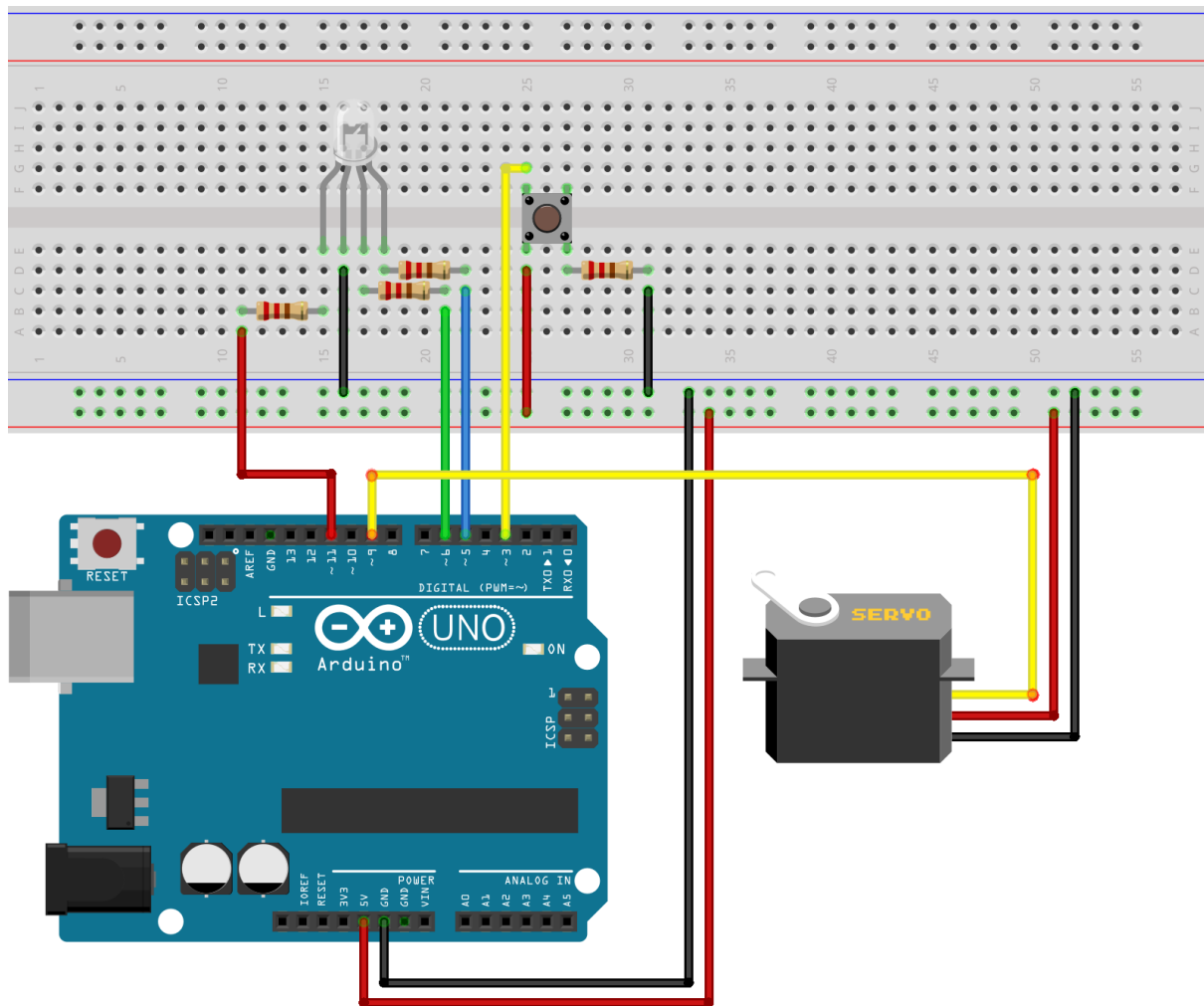


### 1.2.3 Android

Como controlador o Android tem a capacidade de a qualquer momento destrancar a caixa. No entanto não existe comunicação directa entre o Android e os Arduinos. Todas as mensagens enviadas pelo Android são dirigidas ao RPi para que este como cérebro do sistema faça com que os comandos enviados pelo Android sejam cumpridos. O Android é o único componente do sistema capaz de criar/alterar códigos para a abertura da caixa. Este força a sua posterior atualização no RaspberryPi.



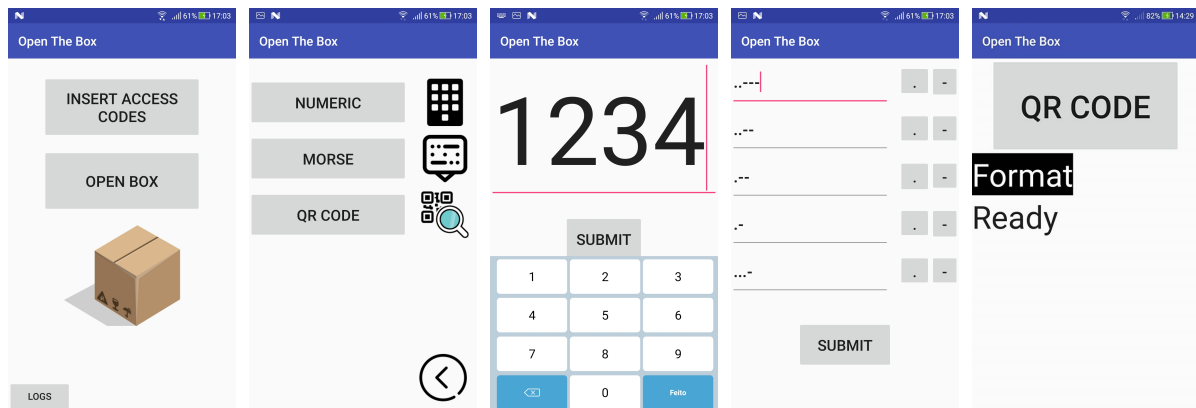
### 1.2.4 Montagem do Arduino Two



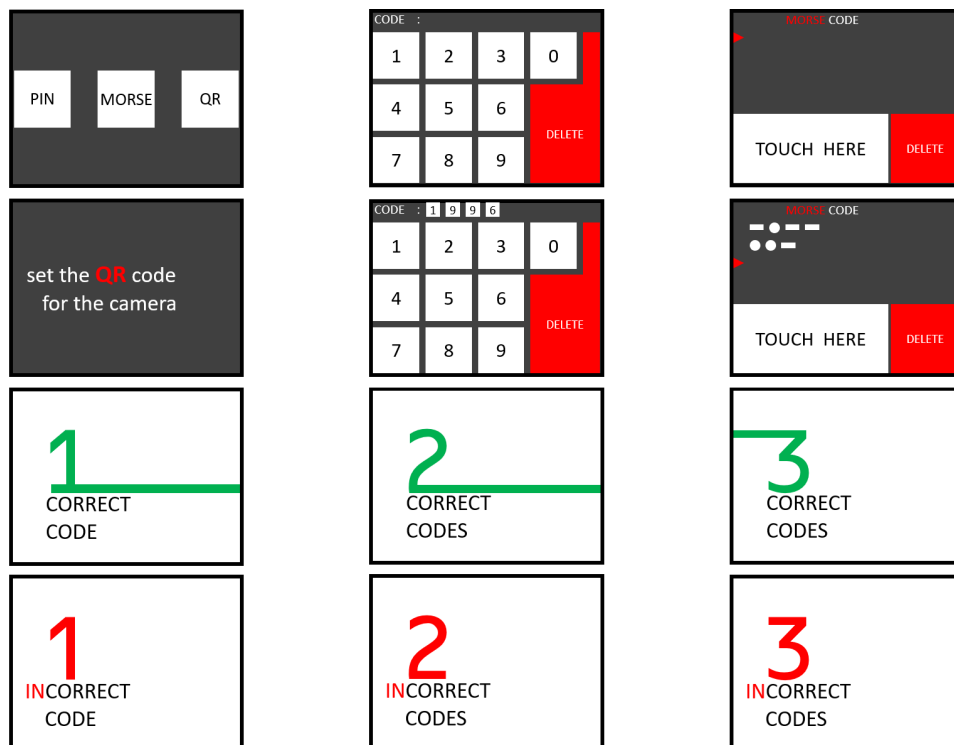
Este diagrama é apenas ilustrativo visto que no sistema final não é utilizada uma *breadboard* devido ao comprimento dos cabos necessários.

Todos os componentes conectam directamente ao arduino sendo que foi necessário soldar cabos para estender o alcance tanto do led como do botão. Do ponto de vista eléctrico é este o circuito existente. A única alteração é que a conexão azul do led não é feita visto que esta cor não é necessária.

## 1.2.5 Interface do Android



## 1.2.6 Interface do Arduino



## 1.3 Tabela de Custos Atualizada

Plataforma	Material	Qt.	Preço
Arduino	Arduino Uno rev3	2	40,00€
	Servo Motors Micro Servo	1	10,00€
	2.8" TFT Touchscreen	1	25,00€
	Tactile Button 12mm	1	01,00€
	LED - RGB Clear Common Cathode	1	01,50€
	Fonte 9V	2	10,00€
Raspberry	RaspberryPi 3	1	40,00€
	Camera Board	1	25,00€
	Cabo USB 2.0 A / USB B 1.8M	2	02,00€
	Cabo USB A <-> MicroUSB	1	01,50€
	Adaptador 5V USB-A	1	05,00€
Android	BQ Aquaris E5	1	135,00€
Caixa	Caixa Madeira 30x30x30	1	04,00€
		Total	300,00€

## 2 Requisitos Cumpridos e Não Cumpridos

### 2.1 Funcionais

O sistema deve ser capaz de ler um código QR	Não
O sistema deve ser capaz de ler input de um touchscreen	Sim
O sistema deve ser capaz de saber quando um botão foi pressionado	Sim
O sistema deve ser capaz de informar quando um código estiver correto	Sim
O sistema deve ser capaz de informar quando um código estiver errado	Sim
O sistema deve ser capaz de emitir uma luz verde quando a caixa estiver aberta	Sim
O sistema deve ser capaz de emitir uma luz vermelha quando a caixa estiver fechada	Sim

### 2.2 Não Funcionais

A caixa deve ser aberta num máximo de 5 segundos após indicação do Android	Talvez
Um <i>lockout</i> de 60 segundos deve ser aplicado após 3 falhas na autenticação	Sim
O <i>lockout</i> deve ser aplicado em menos de 10 segundos	Sim
O sistema terá pelo menos 3 formas de autenticação diferentes	Não

### 3 Dificuldades Encontradas

Como o diagrama de Gantt efetuado no planeamento inicial era demasiadamente genérico é difícil falar em cumprimento ou não dos prazos. Ainda assim é possível dizer que a componente do Arduino cumpre todos os requisitos que foram propostos inicialmente, assim como a do Android. No entanto, o componente RPi falhou largamente. O aluno responsável por esta componente sentiu muitas dificuldades apesar de nunca as ter partilhado. Sem nada o fazer prever informou os elementos do grupo que abandonaria o projeto, no dia anterior à entrega. Alegou demasiados "problemas de concorrência". Para além disto, não deixou qualquer base para que os restantes elementos continuassem/corrigissem o seu trabalho.

Tendo em atenção o sucedido tentamos no tempo restante desenvolver algumas funcionalidades dessa componente. Nesse tempo a principal preocupação foi aprimorar o que estava desenvolvido. Para além disso foi ignorado a leitura do código QR que se optou por considerar sempre certo quando o utilizador escolhe essa opção, desde que fosse introduzido na ordem correta. Foi também ignorado o envio de logs para o Android apesar de estar a ser armazenados num ficheiro do Raspberry.

No Arduino One as principais dificuldades foram sentidas inicialmente no processo de escolher as bibliotecas adequadas. Após isto, tudo ficou simples sendo apenas necessário analisar a posição de cada toque no ecrã. No entanto, esses toques levantaram alguns problemas na leitura do código morse uma vez que a sensibilidade do ecrã não é a melhor. Também existirem problemas de espaço tendo sido descartada a função que mostrava imagens no TFT por, sozinha, ocupar 15% memória total. Para substituir as informações que seriam passadas através de imagens, estas foram desenhadas no ecrã. No Arduino Two não foram sentidas dificuldades. A parte mais trabalhosa foi a montagem do sistema na caixa que obrigou a que fossem soldados cabos.

No Android toda a parte de criação de interfaces e interações correu dentro do tempo previsto e com a dificuldade esperada devido à inexperiência inicial em programação em Android. A partir do momento em que se começou a adaptar à plataforma e à linguagem não demorou muito tempo a ter a aplicação com um aspeto limpo e apresentável e, mais importante, com as interações necessárias. A maior dificuldade surgiu na parte da comunicação. Primeiramente por estar a tentar utilizar um smartphone que, pelo que se percebeu depois, era incapaz de fazer a comunicação. Ao não encontrar nenhuma solução para esse problema usou-se um outro smartphone e aí já ficou a comunicação do Android para um servidor num computador a funcionar correctamente. Outro problema que nesta parte foi a falta de comunicação no sentido inverso, que seria a comunicação da parte do Raspberry Pi para o Android.

Infelizmente, os testes de comunicação efetuados foram num curto espaço de tempo que não permitiu a resolução de todos os problemas.