



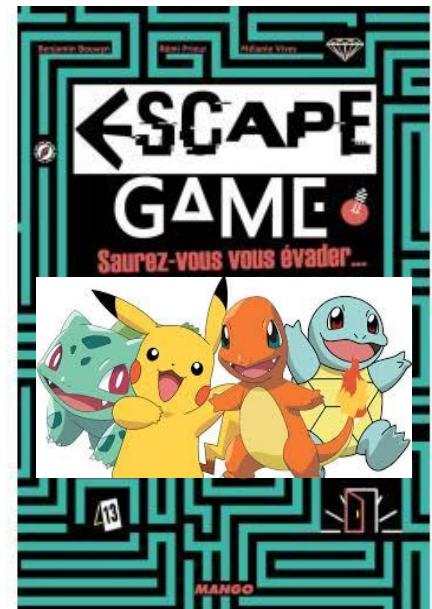
Pokemon Escape Game

24H du code – Sujet catégorie Embedded



Pokemon Escape Game

- “Pokemon Escape Game” est un défi de type “Escape Game” dans lequel un Pokémons doit avancer sur un parcours tout en résolvant des énigmes.
- Le but est la création de manière **ludique** d'une **application embarquée** sur un système avec un microcontrôleur de la famille **STM32**.
- L'application mettra en œuvre des technologies de « l'Internet des Objets »
 - des **entrées/sorties** du système, des **capteurs**
 - de la **connectivité**
 - des **protocoles de communication**
 - des interactions avec **un serveur distant – le Pokédex**
- Cette application embarquée pourra aussi interagir avec des applications sur smartphone et/ou PC en fonction des choix des concurrents.



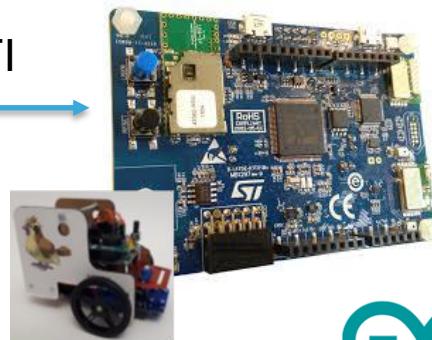
Synoptique globale

Cloud server (fourni par ST)

- Connexion en wifi
- Valide les réponses aux énigmes
- Surveille / affiche la progression des équipes



Equipement basé sur
STM32 B-L475E-IOT01A
programmé par Arduino



Poste de contrôle

- Intéraction avec l'équipe de programmeurs.
- Le pokémon peut être contrôlé, semi-autonome ou autonome (marge de progression)

BLE ou UART



On board Sensors
(LED, température,
accéléro, gyro,
QSPI memory ...)



Déplacement sur le plateau

- 2 servo-moteurs 360° à rotation continue, 2 roues à bande en caoutchouc, 1 roue à bille omnidirectionnelle
- Contrôle par PWM



Time of Flight
Suiveur de ligne
NFC reader
Loudspeaker

Les épreuves

- Les différentes épreuves mettront en oeuvre des périphériques du STM
- Les possibilités sont infinies !!



Salameche



Carapuce



Bulbizarre



Psykokwak



Aspicot

Choisis ton pokémon !



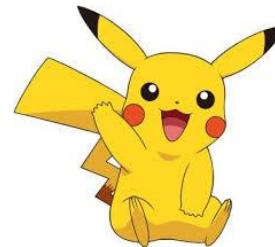
Chenipan



Roucool



Rattata



Pikachu



Rondoudou

Timeline

5

SAMEDI
10h-12h

SAMEDI APRES-MIDI JUSQU'À DIMANCHE MATIN 8h
15h

Découverte

Dressage du Pokémon: Entrainement !

Objectif Blinky: Hello MQTT!
Pokemon Heartbeat !

Objectif:
Pokemon Ready !

DIMANCHE
8h00-9h00

Chambre d'appel:
Check-List Entrainement:

DIMANCHE
9h00-10h30

Escape Game !

Winner:
...

Début de l'évaluation
Récupération des arènes

Entrée sur
l'Escape Game

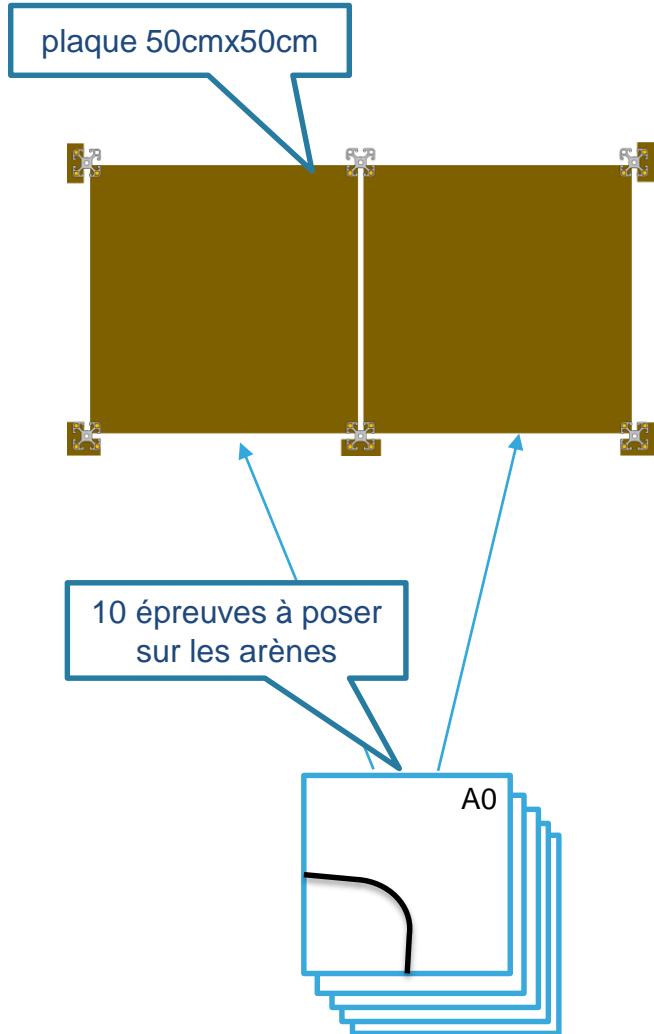
Fin des épreuves

- Les candidats reçoivent:
 - Le kit « Pokémons »
 - 2 cases de jeux (arènes)
 - L'ensemble des énigmes pour la phase d'entraînement
- Les objectifs de la phase de découverte sont:
 - Installation de l'**environnement Arduino**
 - Prise en main du **système embarqué**
 - Atteindre le niveau « **Blinky** »
 - Avoir une première idée des **librairies** à utiliser pour
 - Piloter le chip WiFi
 - Gérer la communication MQTT
 - Piloter les moteurs
- La phase « découverte » est **LE moment pour poser des questions aux organisateurs.**
- Ensuite, **seules des demandes de vérification du matériel** seront prises en compte.

Entrainement

7

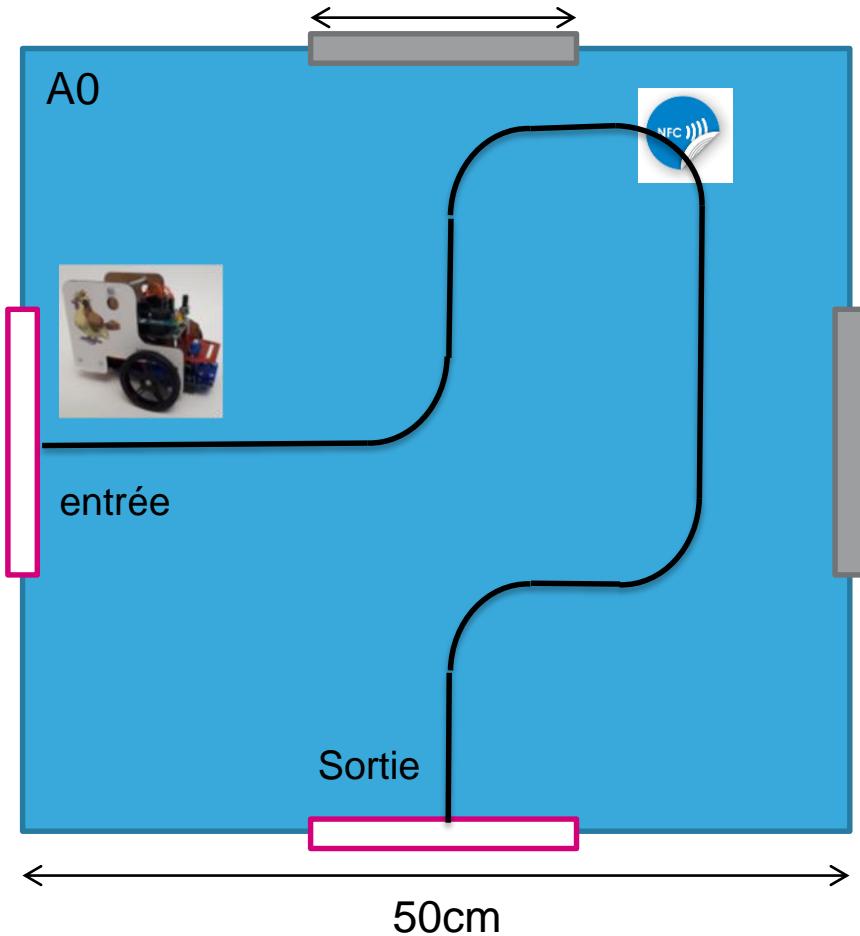
- Les équipes entraînent leurs Pokémons à résoudre les énigmes du parcours
 - Un broker MQTT est disponible pour valider de manière autonome les réponses.
 - Un dashboard est disponible pour suivre la progression des équipes:
<http://24hducode.spc5studio.com:1880/ui/#/0>
- Une épreuve est toujours résolue dans une arène (même à l'entraînement)
- Certaines énigmes donneront lieu à une **variante** lors de la phase finale de l'Escape Game
- Le jury peut poser des **questions aux équipes** pour évaluer leur progression
- A l'issue de cette phase les **cases de jeux et les arènes sont récupérées** par les organisateurs afin de composer le parcours final. A la fin des 24H, les Pokemon aussi seront récupérés.



Entrainement: déroulement d'une épreuve

8

- Chaque épreuve (également appelée énigme) se déroule sur le même principe et est numérotée de A0 à A11
- Les portes des arènes sont amovibles
- A chaque épreuve correspond un papier qui se pose sur le sol de l'arène.
- L'épreuve peut contenir une ligne noire pour guider le Pokémon. Il peut aussi être contrôlé à distance.
- Le Pokémon avance jusqu'à trouver le TAG NFC
- Ce livret contient les informations nécessaires à la réalisation de chaque épreuve. Le TAG NFC contient un identifiant d'épreuve, le nom et les informations éventuelles associées, ex: A3a:sport:avance sur 25cm
- Le Pokémon relève le challenge et envoie la réponse pour validation au Pokédex (maître du jeu, broker MQTT)
- Puis il continue sa route jusqu'à la sortie et l'arène suivante



Comment résoudre une énigme (1/3)

9

- Pour pouvoir résoudre une énigme (aussi bien en phase d'entraînement qu'en phase finale) le Pokémon doit impérativement être capable de lire le tag NFC contenant la définition de l'énigme. **Cette étape n'est pas optionnelle** (sauf pour l'arène A0 qui ne comporte pas de tag NFC).
- L'énoncé de l'énigme est une chaîne de caractère dont le format est le suivant:

A<id-arène>:<thème>:<énoncé>

- Par exemple, pour décrire une énigme sur le thème du sport dans l'arène « A5a », le tag NFC contiendra la chaîne de caractères :

A5a:sport:avance sur 25cm

- Une énigme peut donner lieu à une variante sur le même thème pour la phase finale de l'Escape Game. Dans l'exemple ci-dessus, une variante pourrait être :

A5b:sport:avance sur 35cm

Comment résoudre une énigme (2/3)

10

- La résolution de l'énigme est implémentée au choix du concurrent :
 - Résolution autonome par le système embarqué
 - Résolution par un système déporté (smartphone, PC, serveur...) qui retourne le résultat au Pokémon et/ou au serveur de validation (broker MQTT)
 - Résolution par le dresseur de Pokémon qui retourne le résultat au Pokémon
- **Dans tous les cas le résultat doit être communiqué au Pokédex (serveur internet avec broker MQTT).** Le format du résultat est imposé, il s'agit d'une chaîne de caractères composée comme suit :

A<id-arène>:<rponse>

- Par exemple pour communiquer la réponse « STM32 » comme solution de l'énigme dans l'arène « A1 », le Pokémon devra dire :

A1:STM32

Comment résoudre une énigme (3/3)

11

- L'envoi de la réponse au broker MQTT est le mode de communication imposé (sauf pour l'arène A0)
- Le format de code de retour du broker MQTT est le suivant :
 - OK : réponse correcte
 - KO : réponse incorrecte
 - ??? : réponse non comprise par le broker (problème de syntaxe)
 - Exemple : A1 STM32 est rejeté car le séparateur attendu est ‘:’ et non l'espace, ni ‘,’
- En cas de doute sur la communication MQTT les candidats peuvent émettre la chaîne:

Hello from X avec X=Pikachu, Psykokwak, ...

- Et recevront en retour

Hello X

(Ces essais « Hello » ne joueront pas sur le résultat final)

Chambre d'appel

12

- Note: Nous récupèrerons aussi les arènes à 8H pour monter le parcours final
- La chambre d'appel est le moment où le jury remplit une check-list avec les candidats pour :
 - Valider les énigmes résolues lors de la phase d'entraînement et attribuer les points gagnés
 - Chaque énigme résolue rapporte des points lorsqu'elle est validée par le jury (dans la phase de « chambre d'appel »)
 - Déterminer si le Pokémon est apte à s'attaquer à l'Escape Game
- La check-list est un formulaire rempli à partir du pokedex montrant la progression des équipes et des preuves apportées par les équipes (vidéo, rejouer une épreuve devant le jury). Ce formulaire papier est ensuite archivé tandis que les notes sont saisies dans la feuille de calcul qui permettra de donner le résultat final
- Il n'y a pas de classement à l'issue de cette étape: tous les concurrents peuvent marquer tous les points de la phase d'entraînement.
- *Les concurrents sont invités à livrer le code source au jury qui pourra:*
 - *Le revoir*
 - *L'utiliser à des fins de démonstration dans divers salons et évènements*

Escape Game (1/2)

13

- Le Pokémon doit **démarrer au signal** du jury (le dresseur du Pokémon peut l'activer comme bon lui semble : bouton, commande par bluetooth...).
- A partir de ce top départ, le Pokémon dispose de **6 à 8 minutes** pour sortir du parcours en résolvant un **maximum d'énigmes**. Au bout de ce temps le concurrent est arrêté. Le temps dépendra du nombre d'équipes
- Le mode de **déplacement du Pokémon est libre** (le dresseur peut intervenir ou non quand bon lui semble via du contrôle distant ou l'utilisation d'un bouton de contrôle) **tant que la locomotion est assurée de manière autonome par les moteurs du système**.
- Le Pokémon n'est pas autorisé à sortir du jeu (autrement que par la sortie...) pendant la durée de l'épreuve. Si des actions sont nécessaires (brancher une alimentation, charger un binaire...) elles doivent être effectuées **sans sortir le Pokémon de l'Escape Game**.

Escape Game (2/2)

14

- Dans chaque arène où une énigme est proposée, le Pokémon a le choix de :
 - résoudre l'énigme
 - ou passer l'énigme
 - Il n'y a pas de limite de refus d'énigme. Un Pokémon peut sortir de l'Escape Game et ne résoudre aucune énigme, mais il n'aura pas marqué de points.
- Toutefois le but premier est de résoudre un maximum d'énigmes de et chaque **énigme résolue de A1 à A10 rapporte deux points.**
- Les candidats seront classés:
 - Par ordre de **points marqués**
 - En cas d'égalité aux points, en fonction de **l'avancée dans le parcours**
 - Au temps pour départager ceux qui sont sortis
 - A la case atteinte pour départager ceux qui ne sont pas sortis
 - En cas d'égalité sur les deux premiers critères, le **choix du jury** départage les concurrents.

Comment gagner: notation du jury

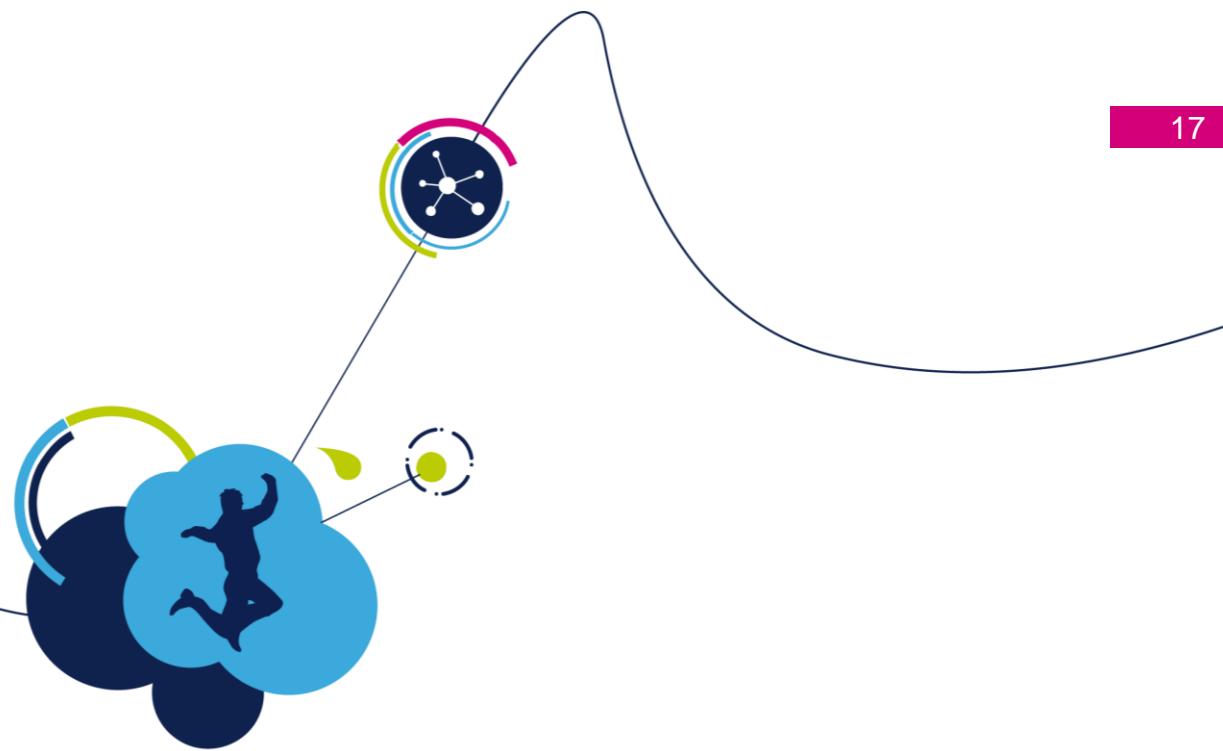
15

Le jury pourra départager des équipes sur plusieurs critères. Les candidats sont libres de leurs choix techniques mais l'esprit du jeu est de valoriser :

- L'utilisation des possibilités de la plate-forme, par exemple :
 - les connectivités WiFi et Bluetooth
 - le suiveur de ligne
- La capacité à produire un code unique (binaire unique) permettant de parcourir l'Escape Game et résoudre les énigmes.
 - Toutefois, les candidats sont autorisés à utiliser plusieurs binaires différents pour valider les énigmes, mais le temps en phase finale étant limité, cela serait pénalisant.
- La capacité à produire un module autonome
 - en termes d'alimentation électrique
 - en termes de déplacement
- La diversité des technologies mises en œuvre
 - Le but premier demeure la création d'une **application embarquée** sur un système avec un microcontrôleur de la famille **STM32**.
 - Toutefois les développements additionnels sur smartphone et/ou PC seront valorisés (monitoring, contrôle du Pokémon pour l'aider dans les situations difficiles...)
- Les aspects créatifs et ludiques
 - Utilisation des éléments de la plate-forme pour donner plus de personnalité au Pokémon...etc...

Notation finale et classement

- La notation finale est la somme des points:
 - Collectés à l'entraînement / 20 points
 - Obtenus au classement de l'Escape Game / 20 points
 - Attribués par le jury / 10 points
- Le vainqueur est l'équipe ayant obtenu le plus de points
- En cas d'égalité aux points, le vainqueur final est celui ayant obtenu le meilleur temps à l'Escape Game.



Les épreuves

Conseils Divers

18

- Le Kit Pokemon est relativement fragile, maniez-le avec précaution
- Vous pouvez travailler en parallèle en préparant à l'avance les prochaines épreuves
- Nous vous recommandons d'utiliser principalement l'alimentation externe et de bien charger votre batterie avant le passage final dans l'Escape Game !
- Si vous rencontrez des problèmes de connexion au WIFI ou que le WIFI est surchargé, passez par un téléphone mobile 4G configuré en Access Point.

Réveille ton Pokemon

19

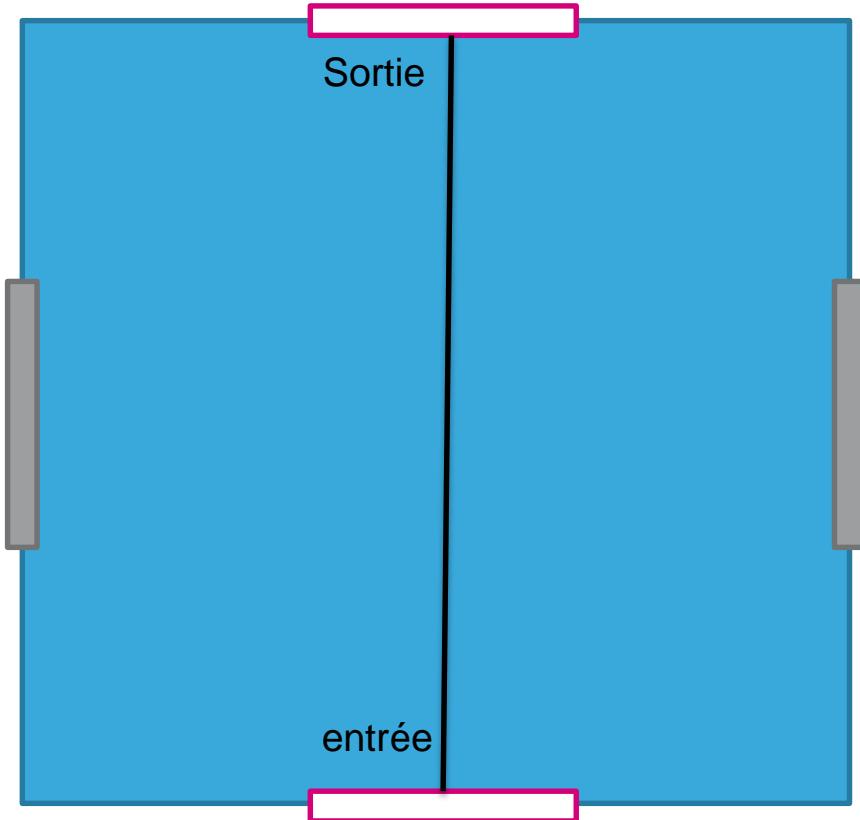
- Les premières étapes à suivre pour relever le challenge proposé sont:
- Installation de l'environnement logiciel basé sur Arduino tel que décrit dans l'annexe technique.
- Découverte et branchement du matériel également présenté en annexe technique
- Puis Compile et charge l'exemple Blinky pour symboliser les battements de cœur de ton Pokemon. Le battement pourra être conservé tout au long des épreuves pour vérifier que le Pokemon est bien réveillé.



Arène A0 – Pokemon GO !

20

- Ton Pokemon est réveillé, le « heartbeat » est en place, alors commence la phase d'entraînement et apprends à marcher !
- Tu pourras suivre la ligne ou bien être guidé à distance ...pour traverser l'arène de part en part. Utilise l'arène A1 par exemple.
- A toi de jouer: Pokemon GO !



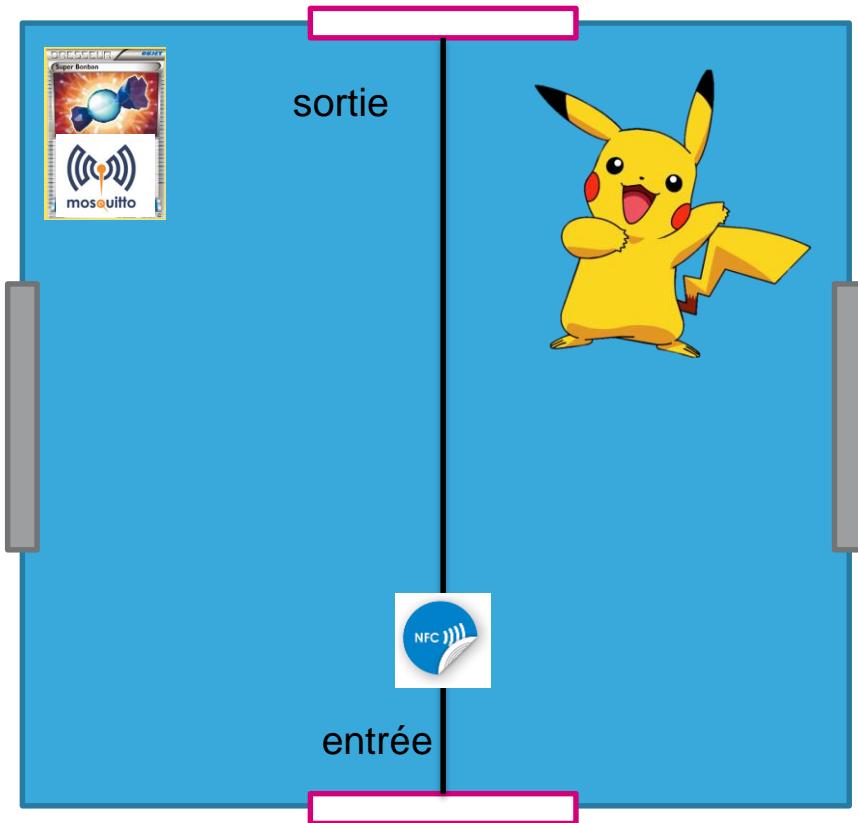
Arène A1 - « Pokemon evolution: savoir parler »

21



Arène A1- Pokemon evolution: « savoir parler »

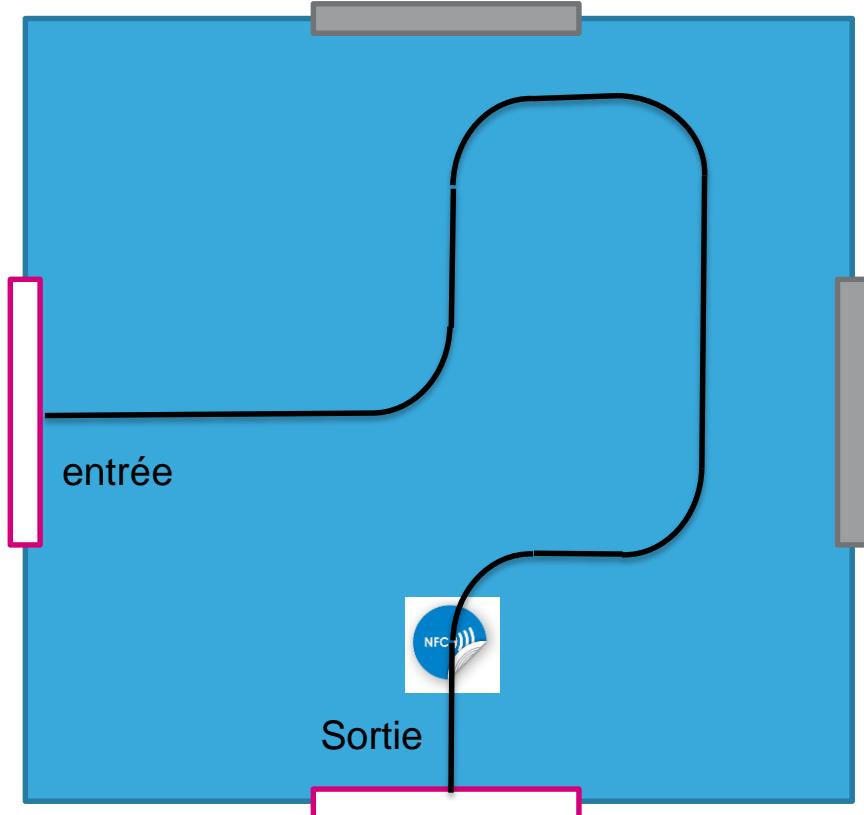
- Le pokemon possède 2 moyens de communication à activer:
 - Le lecteur NFC qui va permettre de découvrir les identifiants pour se connecter au Pokédex
 - Le WIFI pour pouvoir envoyer des messages au Pokédex et valider la résolution des énigmes
- Le pokemon doit avancer jusqu'à détecter et lire le TAG NFC, il y trouvera les informations propres à l'équipe (Le mot de passe et code du topic) puis devra envoyer son premier message au format:
 - A1>Hello 24h du code!
 - Les détails techniques sont fournis en annexe
- Ces identifiants sont à conserver et à utiliser tout au long des 24H, y compris lors de l'épreuve finale
- Validation de l'épreuve:
 - Par réception du message sur le Pokédex



Arène A2 – Pokemon GO crazy

23

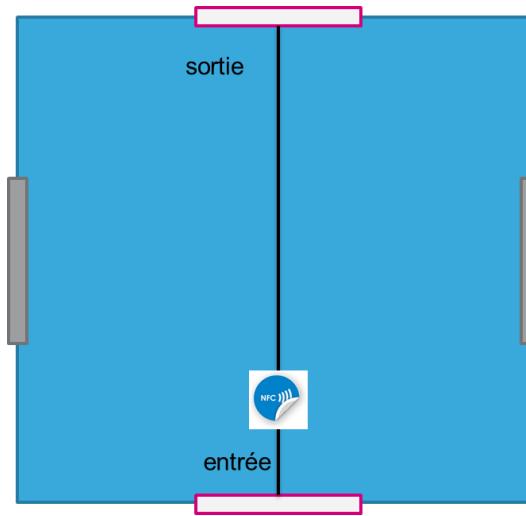
- Fonce sur le circuit, mais n'oublie pas d'attraper le code lors de ta course !
- Mot clé du Tag NFC: **Circuit**
 - Si le tag contient « A2:Circuit:1234 »
 - Le code « A2:1234 » doit être envoyé au Pokédex
- Validation de l'épreuve:
 - Respect du parcours et réception du message sur le Pokédex



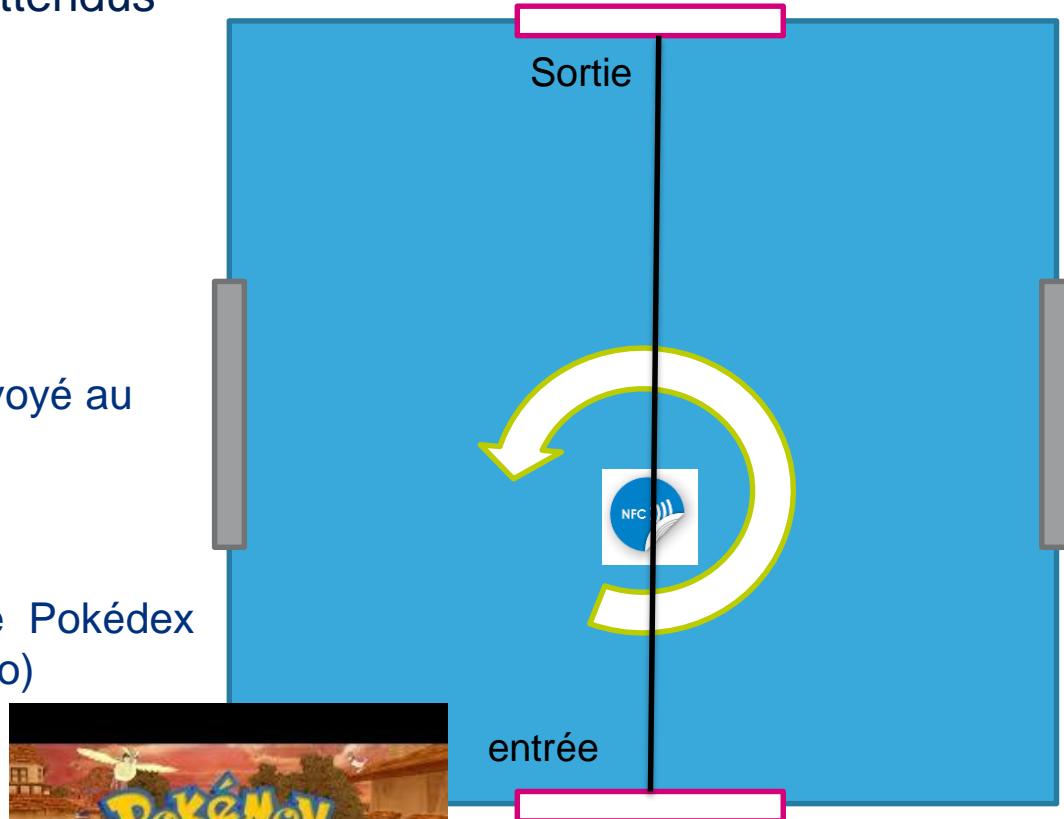
Arène A3 – Pokemon DJ

24

- La partition ton Pokemon lira et le jingle « Wild-pokemon-battle » il chantera.
 - Les notes de musique utilisées doivent être comprises dans la gamme située entre le **Do 261,63 Hz** et le **Si 493,88 Hz**.
- Mot clé du Tag NFC: **dj**
 - « *A3a:dj:PartitionA* »
 - « *A3a:PartitionA* » doit être envoyé au Pokédex
- Validation de l'épreuve:
 - Validation du message par le Pokédex
 - Prends aussi une vidéo de ton pokemon qui joue la musique
 - Lors de l'épreuve finale, le jury validera aussi par une écoute la mélodie jouée par le Pokemon.

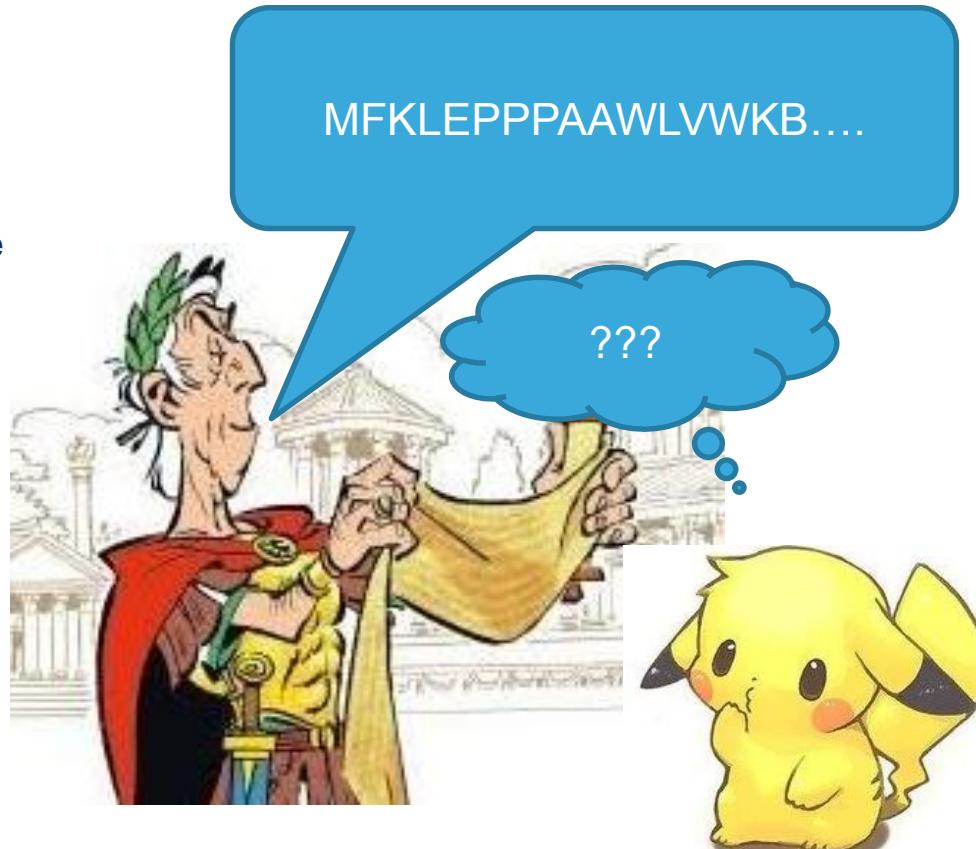
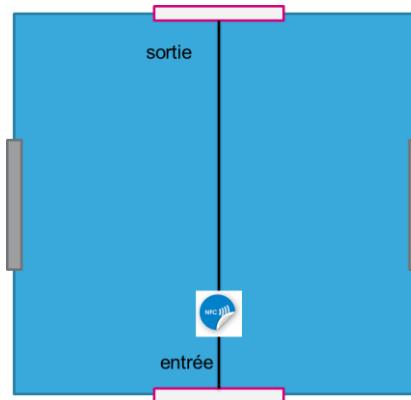


- Le pokémon trouve le TAG NFC et exécute le nombre de tours attendus dans le bon sens
- Mot clé du Tag NFC: **360**
 - « A4a:360:CCWn »
 - CCW = CounterClockWise
 - CW = Clock Wise
 - n: le nombre de tours
 - « A4a:CCWn » doit être envoyé au Pokédex
- Validation de l'épreuve:
 - Réception du message sur le Pokédex et validation visuelle (ou vidéo)



Arène A5a: « Le code de César »

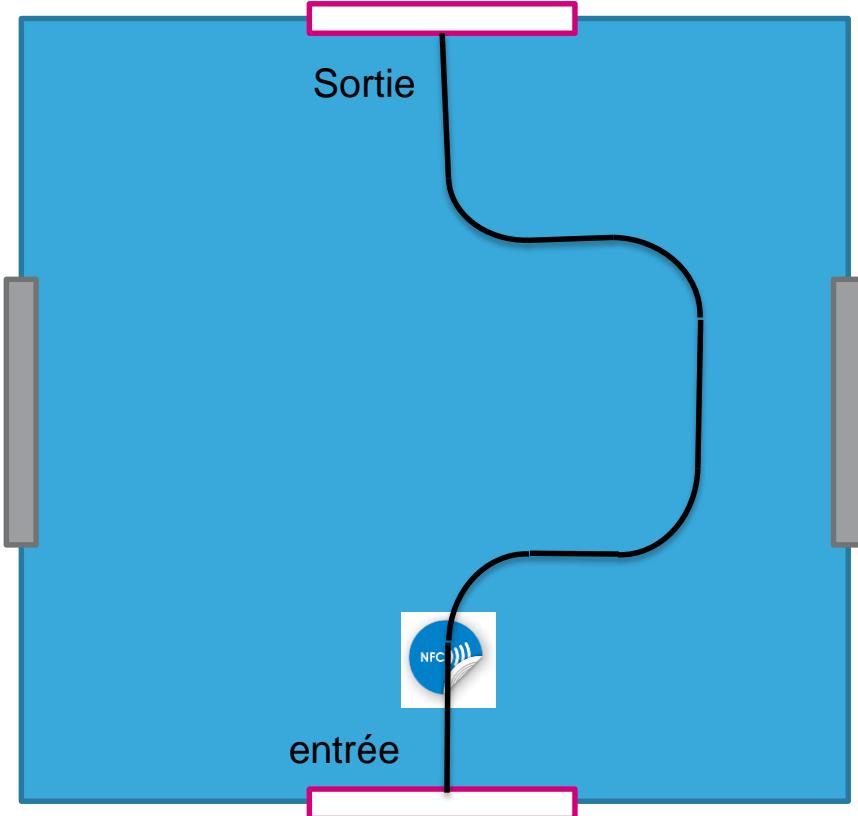
- Le TAG contient le code de César, à toi de le décrypter
- Mot clé du Tag NFC: **crypto**
 - « A5a:crypto:<CodeDeCesar> »
 - « A5a:<PhraseDécodée> » doit être envoyé au Pokédex
- Validation de l'épreuve:
 - Réception de la phrase décodée sur le Pokédex



Arène - A6 – Repos

27

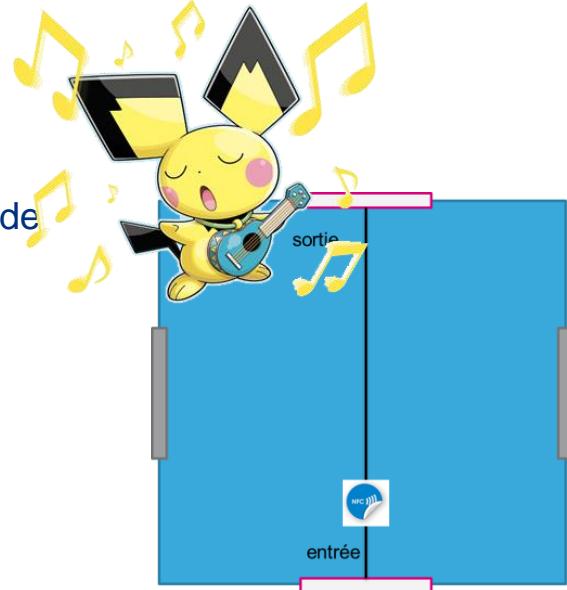
- Le Pokemon est fatigué, il s'arrête et attend le nombre de caresses (swipes) indiqué avant de repartir.
- Le tag NFC contient
 - « A6:swipe:5 »
 - Détecte **avec un capteur** 5 caresses d'un candidat avant de repartir.
 - « A6:5 » doit être envoyé au Pokédex
- Validation de l'épreuve:
 - Réception du message sur le Pokédex et vidéo



Arène - A7 – Pokemon SpyDJ (1/2)

28

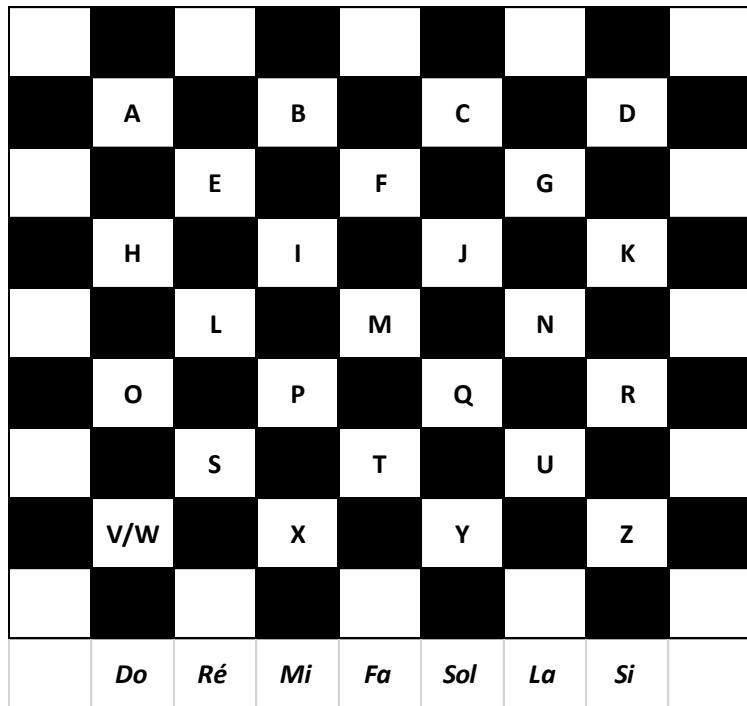
- A la cathédrale du Mans, ton Pokemon rencontrera les anges musiciens. Ton Pokemon devra apprendre à jouer sur l'échiquier de musique afin de jouer à ses amis des musiques codées.
- Le Pokemon doit donc encrypter puis jouer une mélodie à partir du message secret et de la clé de cryptage fournis par le tag NFC.
 - Le format de la partition doit être identique à celui de A3.
 - Les notes de musique utilisées doivent être comprises dans la gamme située entre le Si 246,94 Hz et le La 440,00 Hz.
 - La durée préconisée des notes est de 167ms, et celle du silence 187ms
- Format du TAG:
 - « A7a:spydj:<Message>:<Clé> »
 - « A7a:<MusiqueCodée> » doit être envoyé au Pokédex et la partition de musique doit être jouée
- Validation de l'épreuve:
 - Réception du message sur le Pokédex et vérification de la musique jouée (vidéo à l'appui)



Arène - A7 – Pokemon SpyDJ (2/2)

29

- La documentation détaillée sur l'échiquier musical est fournie en annexe
- Echiquier à musique:



Arène - A8 – Blind test

30

- L'arène Blind Test ne comporte pas de ligne, il faut donc trouver un moyen de le guider vers la sortie sans cette aide

- Quelques pistes ...

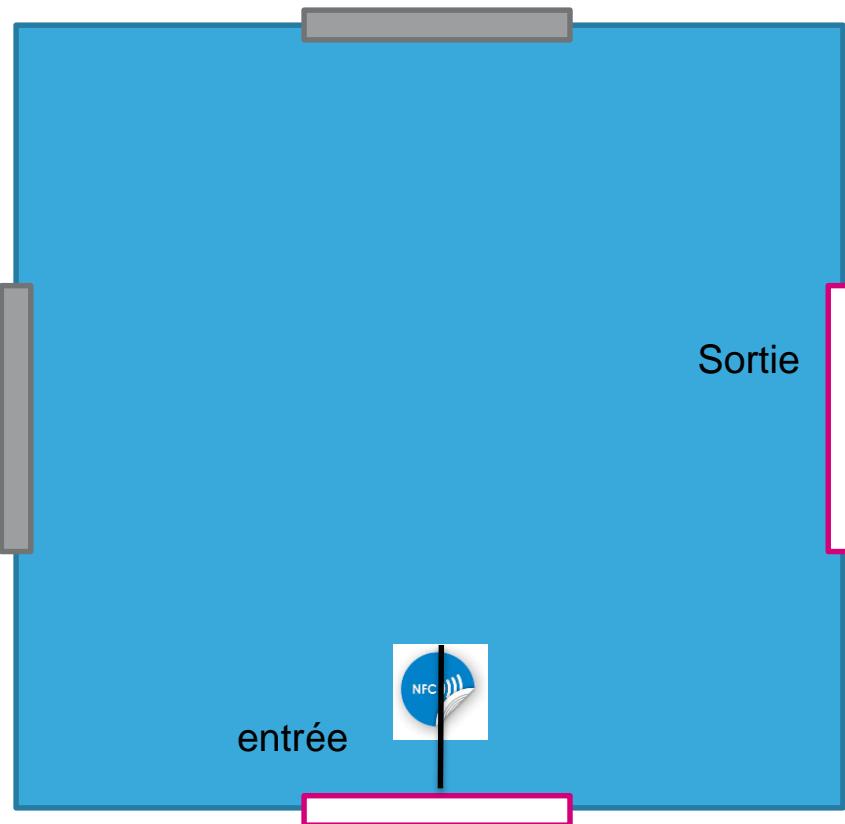
- Mesure de distance avec capteur ToF
- Contrôle - Commande par BLE (Bluetooth Low Energy)
Cf Annexe technique sur le BLE
- Tout est permis !

- Format du TAG:

- « A9:ble:blue »
- « A9:blue » doit être envoyé au Pokédex

- Validation de l'épreuve:

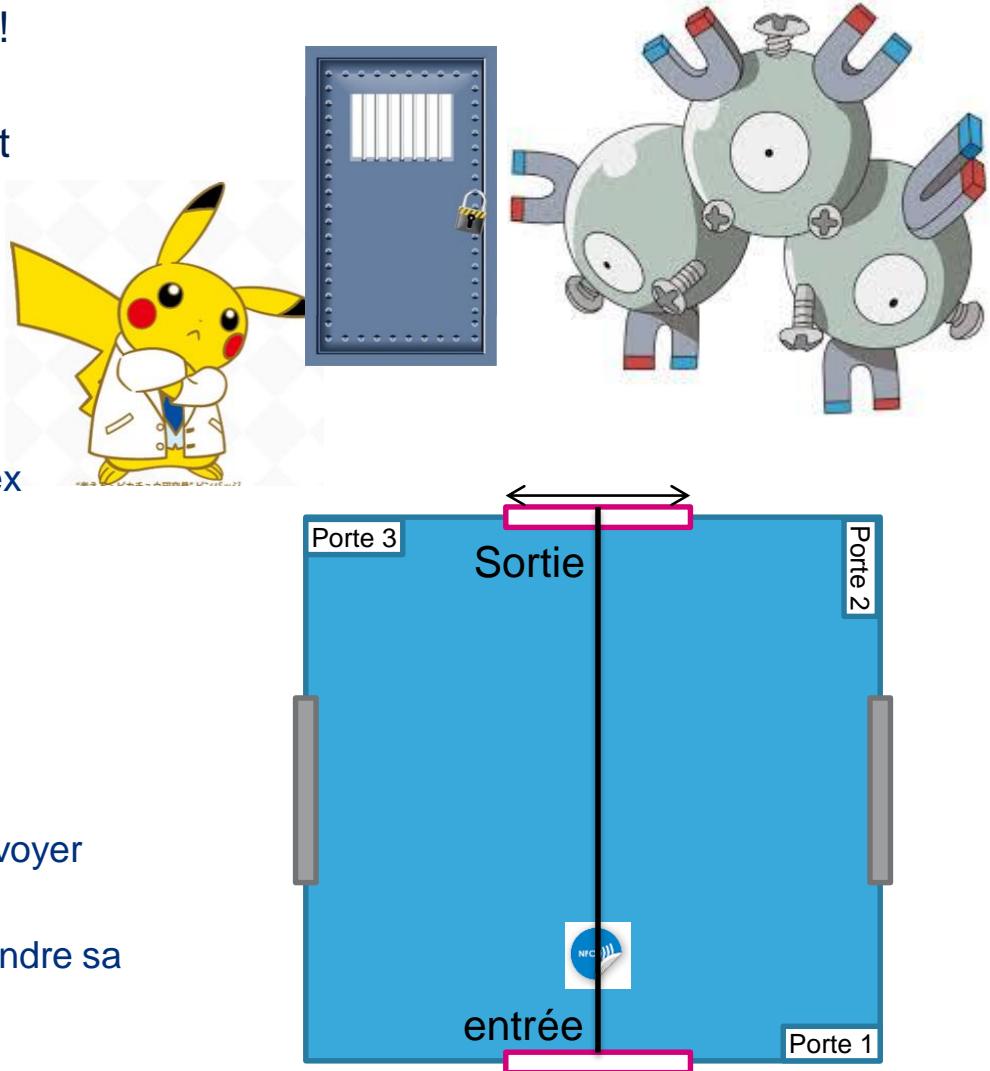
- Capacité à sortir de l'arène (vidéo à l'appui) et envoi du code sur le Pokédex



Arène - A9 – Libère Magnéton

31

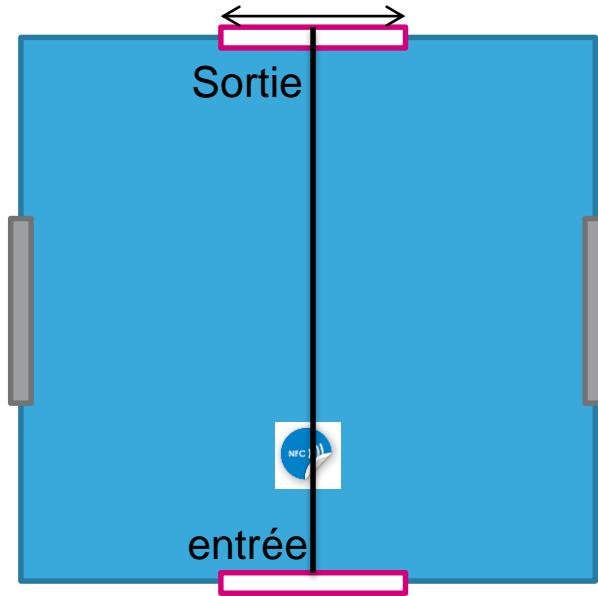
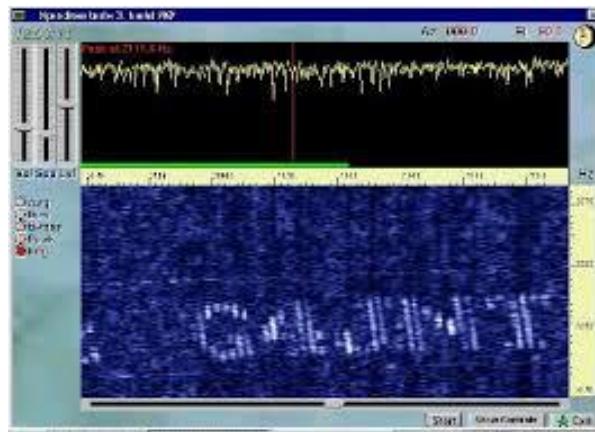
- Magnéton a été emprisonné ... Libère-le !
- Trouve la porte où est caché Magnéton et envoie l'info au Pokédex
- Format du TAG
 - « A9a:magnéton »
 - « A9a:porte1 » ou « A9a:porte2 » ou « A9a:porte3 » doit être envoyé au Pokédex
 - Lors de l'entraînement, place la porte à l'emplacement porte 2 lorsque tu es prêt à envoyer ta réponse au Pokédex. Lors de l'épreuve finale, la position sera aléatoire.
- Validation de l'épreuve:
 - Capacité à trouver magnéton et ne pas envoyer de mauvaise réponse au Pokédex
 - Capacité à optimiser son temps et à reprendre sa route après avoir trouvé magnéton



Arène - A10 – Au fond de ta mémoire

32

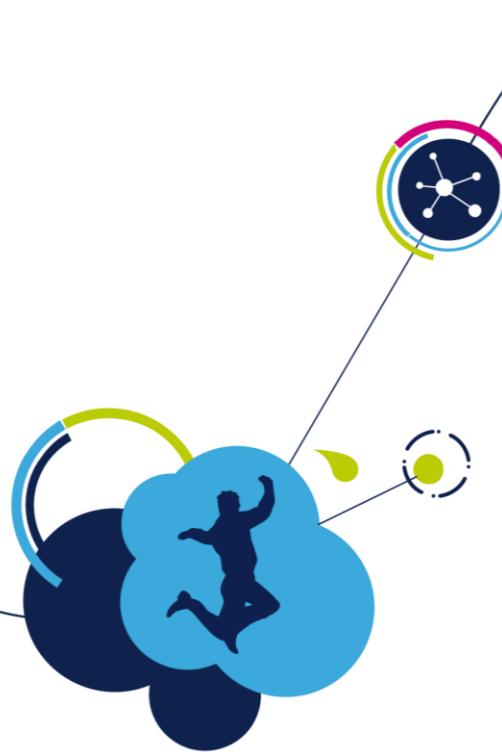
- Recherche au fond de ta mémoire ... un souvenir ... une image floue s'y trouve, mais à toi de redonner forme à cette image
- Format du TAG
 - « A9a:memory »
 - « A9a:<motdepasse> » doit être envoyé au Pokédex
- Tips: QSPI, Hellschreiber Android APP, buzzer
More Tips: 14x7 - 4,045ms



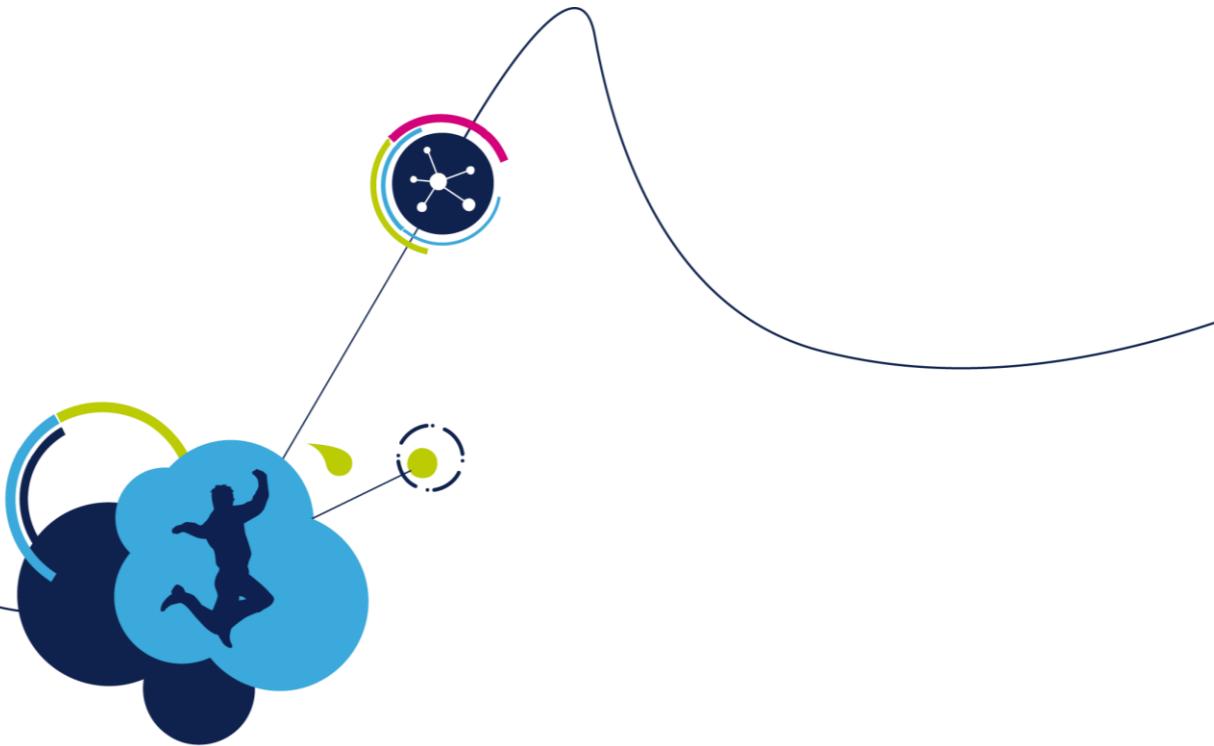
Dernière Arène – Sortie de l'escape Game

- Lors de l'épreuve finale, la dernière arène contiendra un TAG qui indique que le Pokemon a remporté l'Escape Game
- Format du TAG :
 - « A11:victoire »
- Lorsque le Pokemon y parviendra, laisse-le exprimer sa joie librement





Annexes Techniques

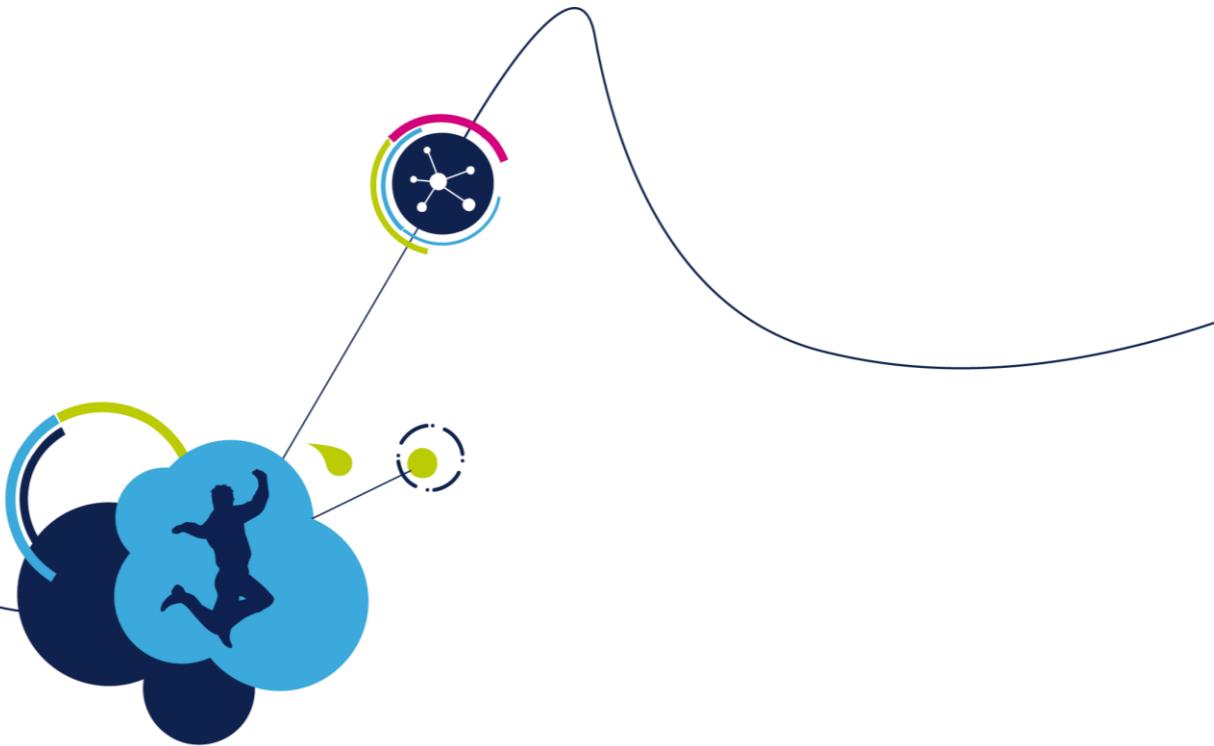


Matériel mis à disposition

Liste du matériel

36

- Ce livret qui tient lieu de documentation principale.
- Le kit Pokemon STM32 qui est décrit en annexe un peu plus loin.
- Les arènes : Au moins 2 arènes en bois seront fournies par équipe. Chaque arène est à monter et à relier par les équipes.
- Les épreuves: Un lot d'épreuves (de A0 à A10) sur papier qui se posent dans les arènes pour pouvoir s'entraîner. Les épreuves sont numérotées.
- L'environnement logiciel : qui doit être installé à partir des liens fournis dans l'annexe dédiée. En cas de problème, une version sur clé USB sera disponible.
- Attention à ne pas abîmer le matériel qui sera récupéré à la fin de l'évènement .



Environnement logiciel

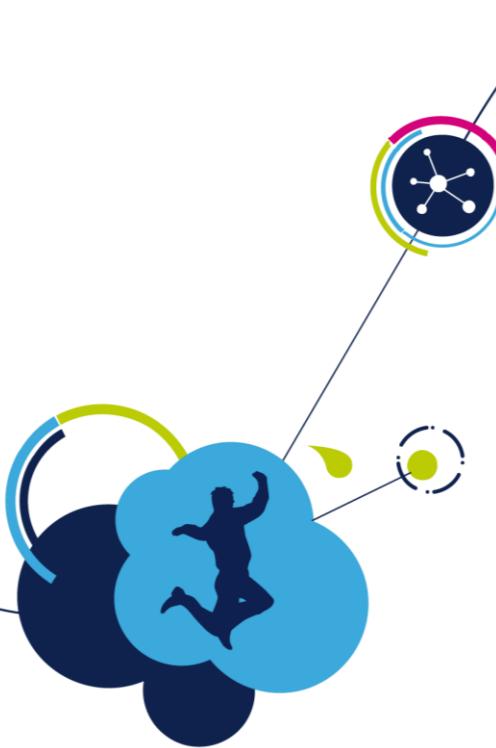
Environnement logiciel

38

- L'environnement à utiliser est Arduino :
 - <https://www.arduino.cc/>
- En addition, il faut ajouter le support du STM32 :
 - <https://github.com/stm32duino/wiki/wiki>
 - Sur la page Boards manager, vous trouverez les informations à utiliser.
- En fonction de votre environnement, il vous faudra peut être les drivers ST Link:
 - http://www.st.com/content/st_com/en/products/development-tools/software-development-tools/stm32-software-development-tools/stm32-utilities/stsw-link009.html
 - Il faut créer un compte myst.com pour les télécharger

Environnement logiciel suite

- Une fois que vous aurez réussi à faire battre le cœur de votre Pokémons, il faudra probablement installer de nouvelles librairies.
- Par exemple :
 - WiFi :
 - STM32duino_X-NUCLEO-53L0A1
 - <https://github.com/stm32duino/WiFi-ISM43362-M3G-L44>
 - MQTT :
 - PubSubClient by Nick O'Leary
 - <https://pubsubclient.knolleary.net/>
 - NFC :
 - STM32duino X-NUCLEO-NFC03A1
 - <https://github.com/stm32duino/x-nucleo-nfc03a1>
 - Non disponible dans le repo Arduino, il faut télécharger le zip depuis github et l'installer
 - Sketch -> Include Library -> Add ZIP library
 - Ne pas hésiter à explorer notre page GitHub :
 - <https://github.com/stm32duino>
- ...



Le kit Pokémon STM32

Le kit Pokémon STM32

41

- Le kit comprend :

- Un Pokémon
- Une alimentation 5V pour la phase d'entraînement
- Une batterie accompagnée de ses 2 cables (Rechargement et connexion au Pokémon)
- Un cable USB blanc de 1,8m



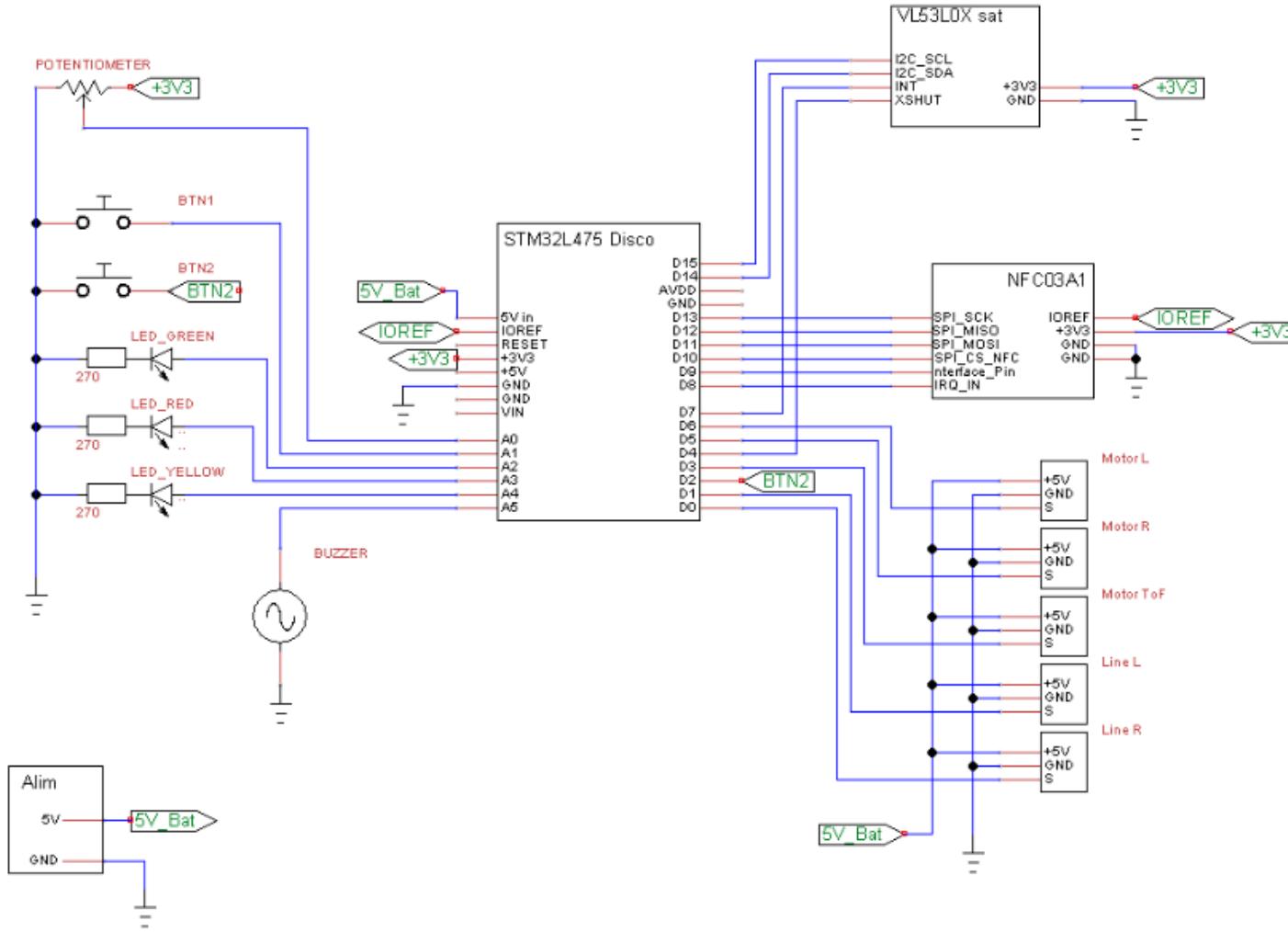
Anatomie du Pokémon

42

- Le Pokémon est constitué de ces éléments :
 - Une carte STM32L4 Discovery kit IoT node ([doc](#), [pinout](#)), incluant, entre autre :
 - Une QSPI : MX25R6435F
 - Un module BLE SPBTLE-RF
 - Un module WiFi : Inventek ISM43362-M3G-L44
 - Un magnétomètre : LIS3MDL
 - Deux servo moteurs à rotation continue pour le déplacement : ([fs90r](#))
 - **Attention aux valeurs min et max du pwm pour l'utilisation du moteur :**
ex de code : motor.attach(Pin, 900, 2100);
 - **Si des valeurs en dehors de cette plage sont utilisées, il y a un risque de deterioration du moteur, ne pas utiliser l'interface sans préciser les valeurs min et max**
 - Un lecteur de tag NFC ([X-NUCLEO-NFC03A1](#))
 - Deux suiveurs de ligne ([GT1140](#))
 - Un servo moteur ([fs90](#))
 - Un satellite Time Of Flight (ToF) ([X-NUCLEO-53L0A1](#))
 - Quelques leds, deux boutons et un potentiomètre

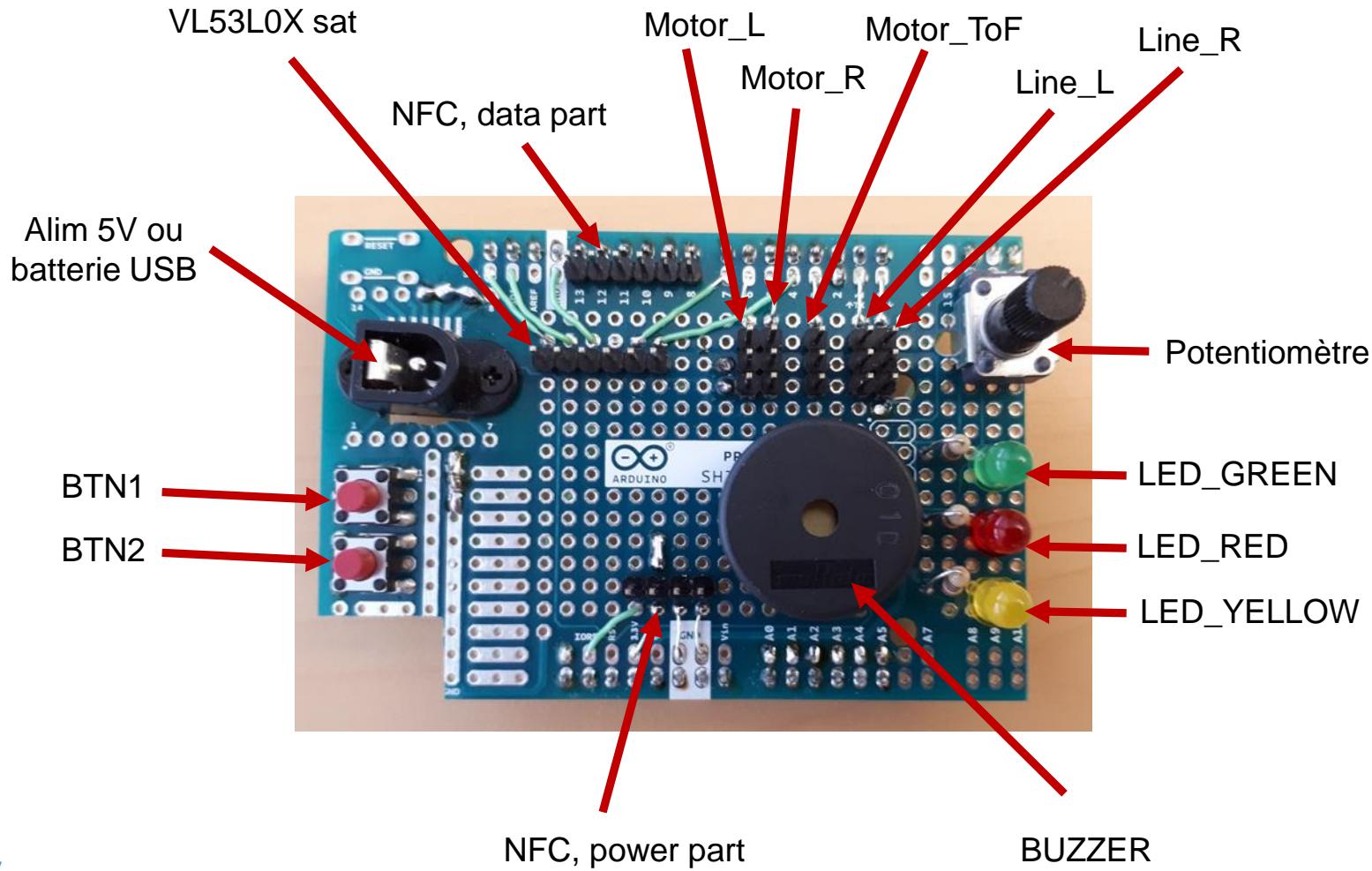
Pokémon aux rayons X

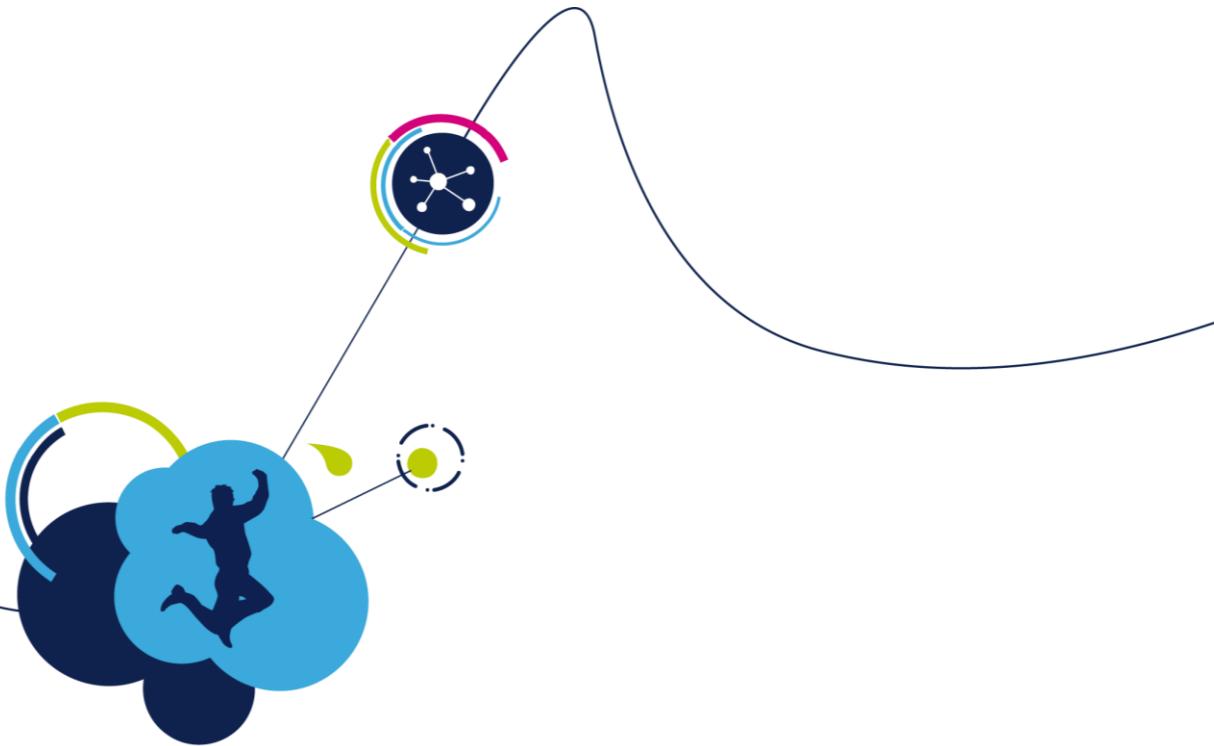
43



Pokémon vu de dessus

44





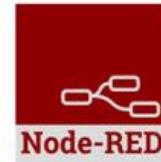
Communication avec le serveur

Technologies Utilisées

- Le serveur utilisé pour valider les réponses aux énigmes est un broker MQTT (Mosquitto) associé à un serveur NodeRED.
- Un dashboard est disponible pour suivre la progression des équipes.



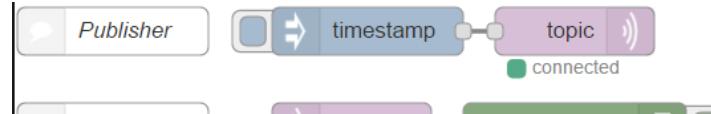
MQTT



HTTP



La communication
MQTT peut aussi être
déportée dans un
smartphone ou un PC
lié au STM32 par BLE...



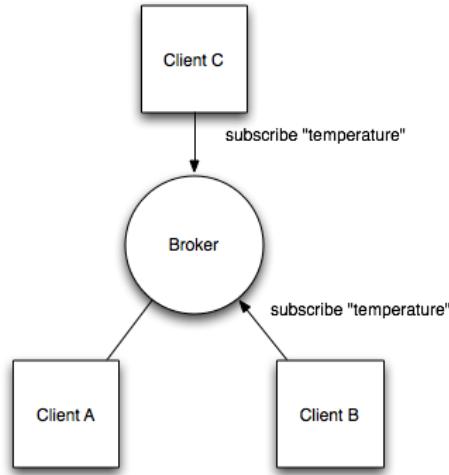
Protocole MQTT

- Chaque Pokémon doit être capable de mener un dialogue MQTT
 - Le protocole MQTT peut être géré par le device STM32 lui-même
 - Ou le protocole MQTT peut être déporté dans un smartphone ou un PC qui communique avec le STM32 (par BLE ou autre)
- MQTT est un protocole de transport client/serveur basé sur la notion de **publish** et de **subscribe**.
- Le protocole permet de transporter des **messages** (les données)
- Un message est publié (**publish**) par un client sur un **topic**
- Tous les clients qui s'enregistrent (**subscribe**) à ce topic reçoivent le **message**

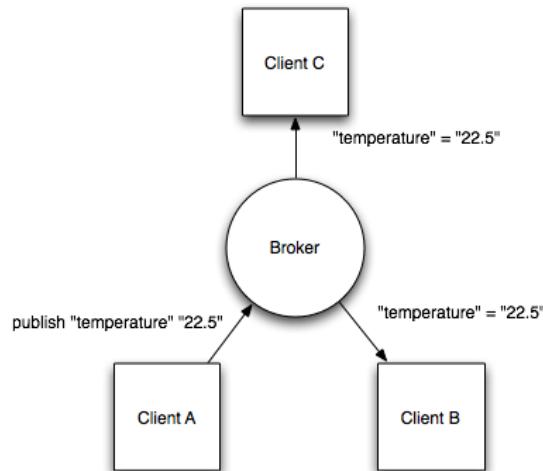
Le modèle de communication MQTT

For example, imagine a simple network with three clients and a central broker.

All three clients open TCP connections with the broker. Clients B and C subscribe to the `topic temperature`.



At a later time, Client A publishes a value of `22.5` for topic `temperature`. The broker forwards the message to all subscribed clients.

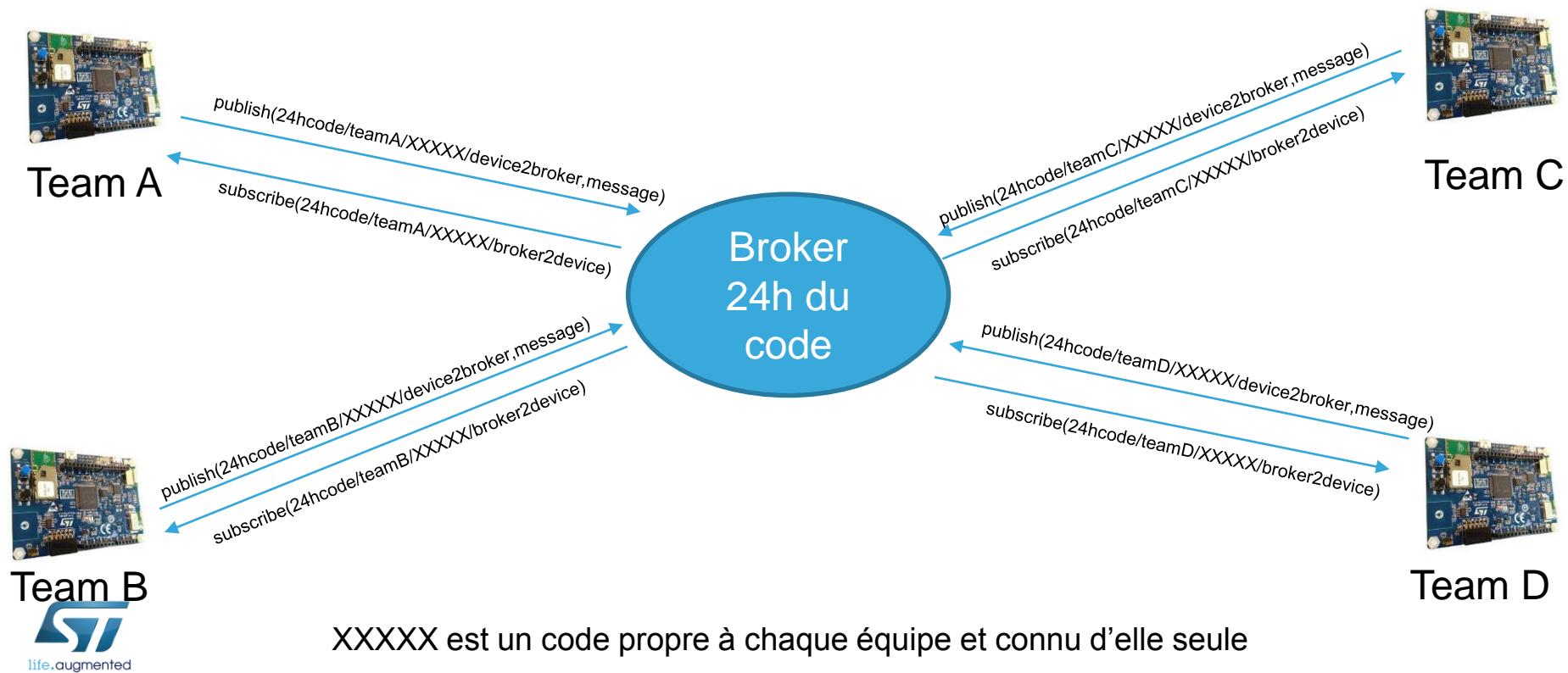


The publisher subscriber model allows MQTT clients to communicate one-to-one, one-to-many and many-to-one.

Le modèle de communication

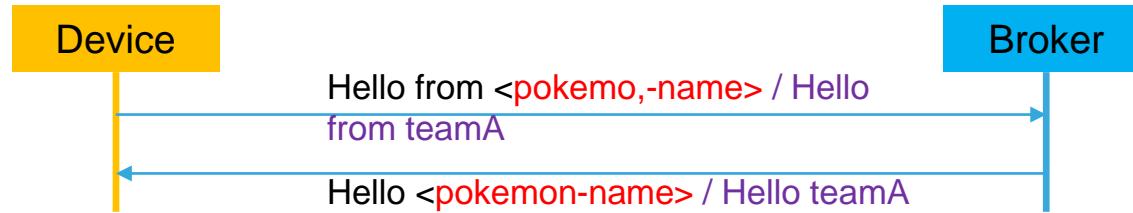
49

- Chaque Pokémon a 2 topics qui lui sont propres
 - Un topic sur lequel il publie les messages qu'il veut envoyer au broker: ce sont les réponses aux énigmes
 - Un topic sur lequel il s'enregistre pour recevoir les retours du broker (OK, KO, ???, Hello <pokemon-name>)



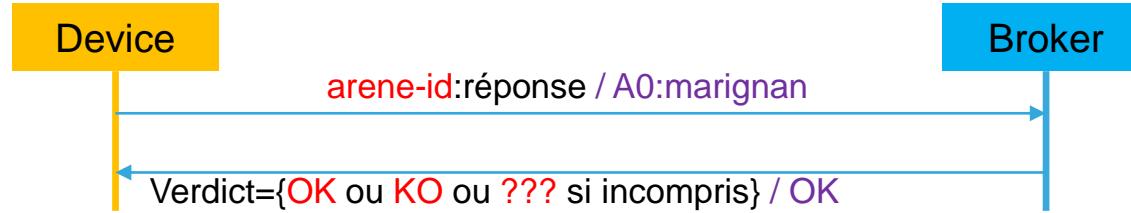
Messages MQTT à échanger

- Message « Hello from <pokemon-name> » pour vérifier la connexion



L'équipe sait que la connexion fonctionne (URL ok, identifiants OK, topics OK) s'il y a un retour.

- Envoyer la réponse pour l'arène **arene-id**: « **arene-id:réponse** »



Si l'équipe reçoit **OK** alors l'épreuve est validée.

Si l'équipe reçoit **KO** alors elle doit tenter une autre réponse.

Si l'équipe reçoit **???** elle doit vérifier le formatage de son message.

Si l'équipe ne reçoit rien elle doit vérifier la connexion.

Connexion MQTT

51

- La connexion au broker MQTT requiert un **login** et un **password** ainsi qu'un **client ID**
- Les logins et client ID des différentes équipes sont listés ci-dessous:
 - les mots de passe sont obtenus lors de l'épreuve A1.
 - La partie code équipe (XXXXXX) des topics est obtenue lors de l'épreuve A1.

Equipe/ clientID	MQTT credentials		MQTT topics		
	Broker login	Broker password	Code équipe	pub topic	sub topic
teamA	Bulbizarre	XXXXXXXX	XXXXX	24hcode/teamA/XXXXX/device2broker	24hcode/teamA/XXXXX/broker2device
teamB	Salameche	XXXXXXXX	XXXXX	24hcode/teamB/XXXXX/device2broker	24hcode/teamB/XXXXX/broker2device
teamC	Psykokwak	XXXXXXXX	XXXXX	24hcode/teamC/XXXXX/device2broker	24hcode/teamC/XXXXX/broker2device
teamD	Carapuce	XXXXXXXX	XXXXX	24hcode/teamD/XXXXX/device2broker	24hcode/teamD/XXXXX/broker2device
teamE	Chenipan	XXXXXXXX	XXXXX	24hcode/teamE/XXXXX/device2broker	24hcode/teamE/XXXXX/broker2device
teamF	Aspicot	XXXXXXXX	XXXXX	24hcode/teamF/XXXXX/device2broker	24hcode/teamF/XXXXX/broker2device
teamG	Roucool	XXXXXXXX	XXXXX	24hcode/teamG/XXXXX/device2broker	24hcode/teamG/XXXXX/broker2device
teamH	Rattata	XXXXXXXX	XXXXX	24hcode/teamH/XXXXX/device2broker	24hcode/teamH/XXXXX/broker2device
teamI	Pikachu	XXXXXXXX	XXXXX	24hcode/teaml/XXXXX/device2broker	24hcode/teaml/XXXXX/broker2device
teamJ	Rondoudou	XXXXXXXX	XXXXX	24hcode/teamJ/XXXXX/device2broker	24hcode/teamJ/XXXXX/broker2device

épreuve A1 pour obtenir XXXXXXXX et XXXXX

Accéder au serveur et aux dashboards

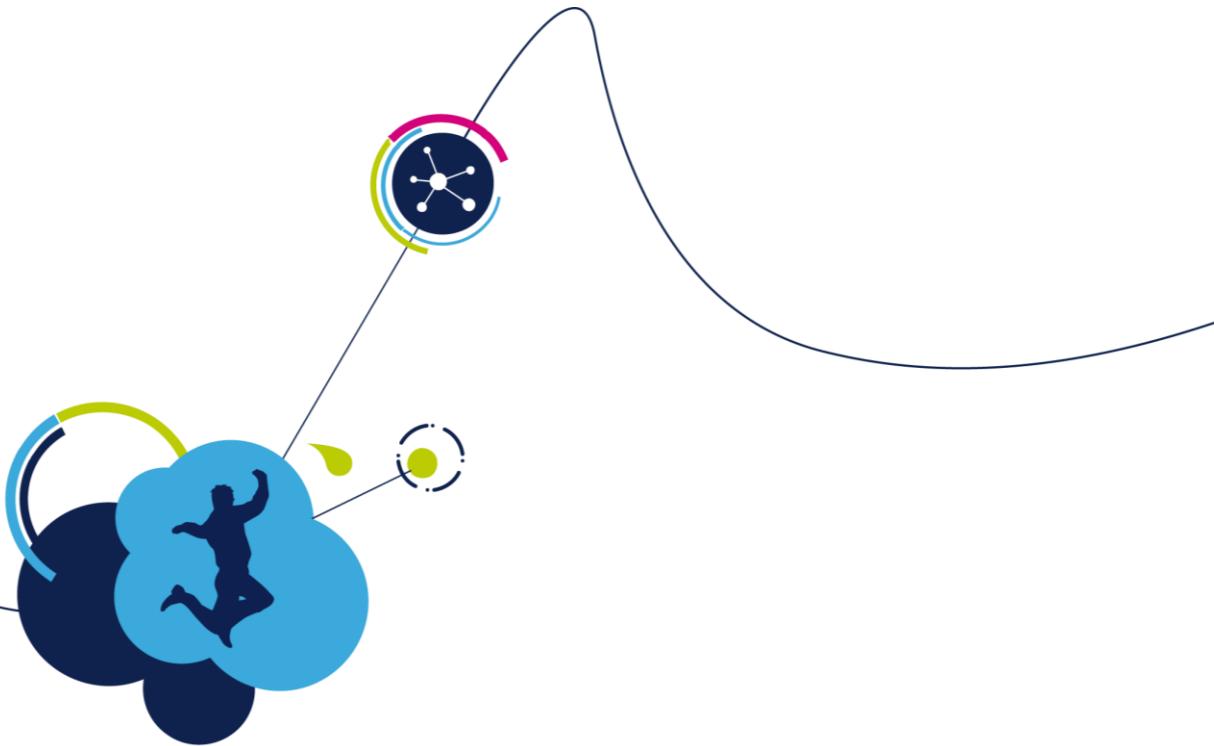
52

- Broker MQTT:
 - Adresse: **24hducode.spc5studio.com**
 - Port: **1883**
- Dashboards:
 - <http://24hducode.spc5studio.com:1880/ui/#/0>
- Connexion au WIFI de la CCI pendant les 24H
 - Nom du réseau : **24HDUCODE**
 - Clé : **2018#24hcode!**

Librairie MQTT Arduino

53

- Chaque équipe est libre d'utiliser la librairie de son choix voire de sous-traiter la communication MQTT à un PC ou smartphone relié au STM32 (par BLE par exemple)
- La communication MQTT depuis le STM32 Arduino a été vérifiée avec la librairie:
 - <https://github.com/knolleary/pubsubclient>
 - API: <https://pubsubclient.knolleary.net/api.html>



Echiquier musical

L'échiquier de musique décrypté ?

Au XIV^e siècle, on sait que Jean Le Bon et tous les grands personnages des cours européennes possédaient un échiquier de musique et qu'ils avaient à leur service des personnes sachant en jouer...

Il est bien évident que les sons produits par cet instrument que nous qualifierons élégamment "d'intimes", ne lui ont sans doute pas fait gagner l'engouement musical des puissants, mais il se pourrait que ce fût sa fonction de communication "délicate" qui était privilégiée et qui leur était très utile.



Communiquer avec quelqu'un sans que les messages ne soient interceptés par d'autres personnes est une préoccupation très ancienne. Ainsi est née la cryptographie, la science des codes secrets. La condition incontournable n'est pas tant de cacher le message, mais de se mettre d'accord sur la technique de dissimulation entre celui qui envoie le message et celui qui le reçoit.

Contrairement à la cryptographie qui transforme des messages de manière à les rendre incompréhensibles, l'échiquier de musique s'avère être un cas de sténographie (qui cache les messages dans un support) par séagramme musical (où les éléments codés ne sont ni des lettres, ni des chiffres ; le sens est véhiculé par la musique). Avec un échiquier de musique, il est possible de crypter et décrypter des messages sous la forme d'une mélodie musicale audible par tous.

Détail du dispositif acoustique. Un petit marteau métallique (tangente), vient frapper les cordes et reste en contact avec elles pour les faire résonner.

Aspect cryptologique

L'échiquier de musique, tel qu'il est représenté dans la peinture des anges musiciens de la cathédrale du Mans, possède un damier de 9 cases sur 9. Mais en raison de l'épaisseur du mécanisme sonore disposé à l'intérieur et qui occupe une ligne et une colonne à la périphérie du damier, l'instrument se retrouve en configuration efficace de 7 cases sur 7. Ceci correspond aux 7 touches musicales (*do, ré, mi, fa, sol, la, si*).

Dans le tableau ci-après sont distribuées sur les cases blanches les 26 lettres de l'alphabet. Le V et le W, identiques au Moyen Âge, se partagent la même case.



On constate qu'à chaque colonne de lettres correspond une note de musique. Et qu'à chaque répétition de la note correspond une lettre (exemple : deux notes *do* donnent la lettre *H* ; un *sol* donne la lettre *C* et ainsi de suite). On peut donc composer une mélodie en transcrivant le texte à dissimuler. Par ailleurs, l'insertion de chiffre au message ne pose pas de problème particulier. La notation numérique est réalisée en utilisant soit les chiffres romains, soit par correspondance entre les lettres et les nombres par les techniques anciennes de gématrie (*guematria*) ou isopséphie.

Cryptage : la clef

Mais le système est encore trop simple pour bien cacher le message. Il correspond historiquement aux procédés utilisés sous l'Antiquité comme les techniques de substitutions dites "codes de César".

Le code de César

Au 1^{er} siècle av. J.-C., les Romains inventent une méthode de cryptographie basée sur une substitution simple avec l'alphabet normal dans les communications du gouvernement. Il s'agit simplement de décaler les lettres de l'alphabet d'une quantité fixe. Par exemple, si on remplace A par D, on remplace B par E, C par F, D par G, etc. César écrit parfois en remplaçant les lettres latines par les lettres grecques. Ce système facile à "casser", fut malgré tout encore employé par les officiers sudistes pendant la guerre de Sécession, et même par l'armée russe en 1915.

Au Moyen Âge, une variable est alors introduite : la clef de chiffrement, qui rend le système bien plus compliqué pour celui qui chercherait à découvrir le message caché. Ainsi, on ne laisse plus la disposition des lettres immuable sur l'échiquier, mais leur organisation change à chaque fois selon la clef utilisée.

La clef de chiffrement peut être un mot ou un groupe de mots placés dans les premières cases. Ici, ce peut être idéalement les paroles des anges musiciens dans les phylactères peints à la cathédrale du Mans.

On distribue ensuite dans les cases restantes et dans l'ordre les lettres de l'alphabet non utilisées dans le mot ou le groupe de mot. On prend soin de ne pas répéter les lettres déjà placées, on évite ainsi les doublons, etc.

On observe que les lettres ont changé de place et ne correspondent plus aux mêmes notes de musique que précédemment. À présent, deux notes *do* donnent la lettre C ; un *sol* donne la lettre E et ainsi de suite.

On peut donc composer une mélodie comme celle que vous trouvez dans *Le Secret des anges*. La partition écrite peut voyager en toute impunité au regard de tous, personne ne peut soupçonner qu'un message est codé par la musique. Même si d'aventure, une personne soupçonne un message caché, il devra imaginer un système de distribution des lettres dans un tableau (échiquier), et il lui faudra de plus obtenir la clef de chiffrement (texte de la mélodie, ici *Ave Maria*), pour avoir la disposition précise des lettres correspondant aux notes de musique. Il y a donc une grande difficulté à retrouver, à partir de la mélodie, le message initial si l'on n'a pas la clef.



Exemple avec la clef de chiffrement *Ave Maria*.

Code de Platon

Platon (IV^e siècle av. J.-C.) aurait inséré dans ses écrits un schéma régulier de structure musicale correspondant à des symboles. Ce "code de Platon" démontrerait qu'il avait anticipé la révolution scientifique qui débuta au XVII^e siècle.

Musique cryptée de Bach

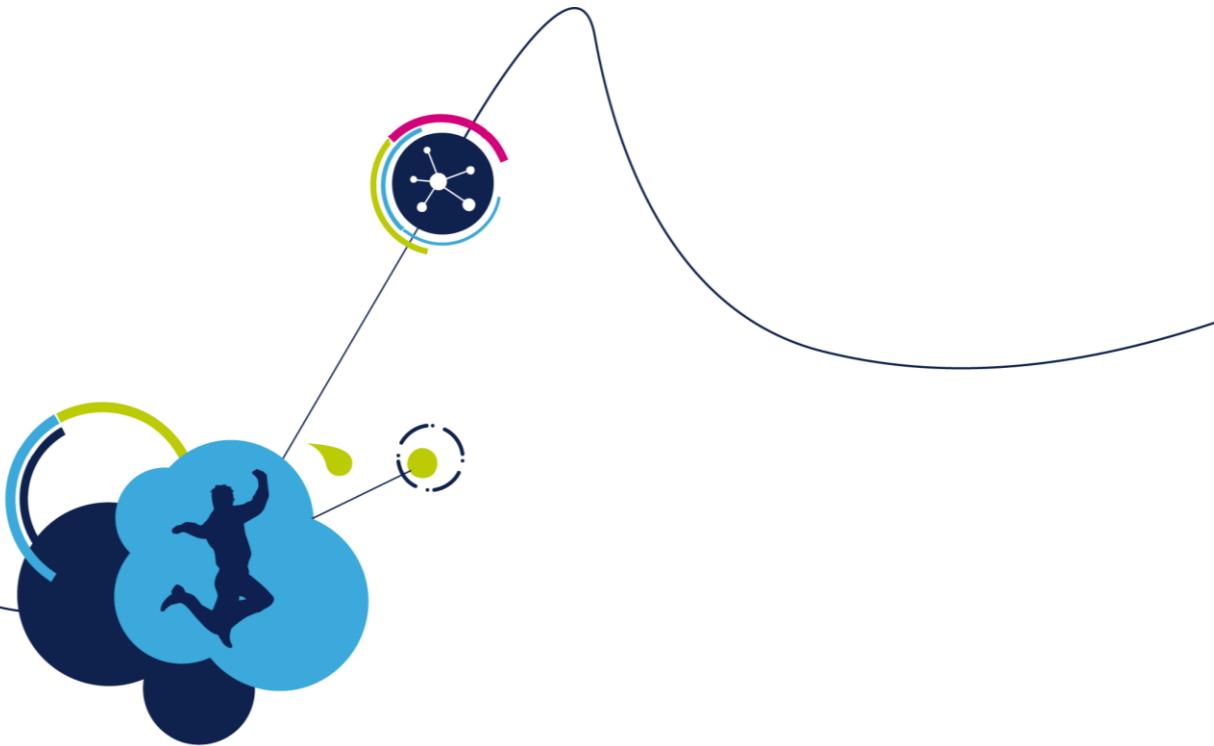
Décryptage

En faisant le processus inverse, on joue les notes de la mélodie sur l'échiquier de musique et on retrouve le message initial. L'instrument occupe bien une position centrale dans ce système de code secret. Ce dernier aurait été suffisamment robuste au Moyen Âge pour être sûr, car peu de gens savaient lire et écrire, encore moins déchiffrer la musique (la tradition orale permettait de s'affranchir facilement des supports écrits). Comme la récente découverte de codes musicaux dans les textes de Platon, les mélodies médiévales qui sont analysées et débattues depuis longtemps portent sans doute des messages que personne n'avait décelés jusqu'ici... Avis aux curieux ! De même, cette technique peut être parfaitement utilisée aujourd'hui pour de nouvelles compositions musicales.

De nombreux musiciens auraient intégré des messages dans leurs compositions. Le plus célèbre d'entre eux, Jean-Sébastien Bach (1685-1750), se serait largement inspiré d'un langage alphanumérique. Une sorte de correspondance entre des lettres et des nombres pour signer des œuvres ou pour souligner une attention particulière. Il ne faut pas oublier que l'instrument favori de Bach, sur lequel il passait le plus de temps, était le clavichord. C'est pour lui que Bach écrivit ses *48 Préludes et Fugues*. Il est conservé au Bach-Museum de Leipzig. Or, il se trouve justement que cet instrument est, à l'origine, un échiquier de musique !

The image shows two staves of musical notation in G clef and common time. The first staff contains the words "Ceci", "est", and "un". The second staff contains the word "secret.". The notes are chosen to represent the letters of the French phrase, with varying rhythms and note heads (solid or hollow) to distinguish between them. The music is set against a background of a grey gradient.

Exemple de cryptage de la phrase : *Ceci est un secret*, en notation musicale actuelle avec la clef Ave Maria et la disposition des lettres sur l'échiquier de musique ci-dessus. Le compositeur a toute liberté pour choisir le rythme des notes.

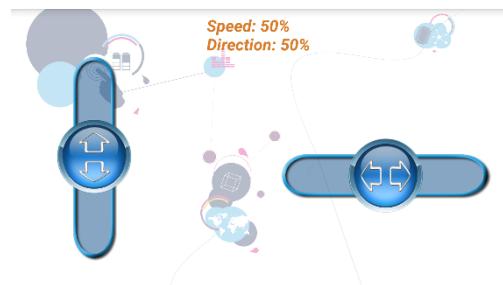


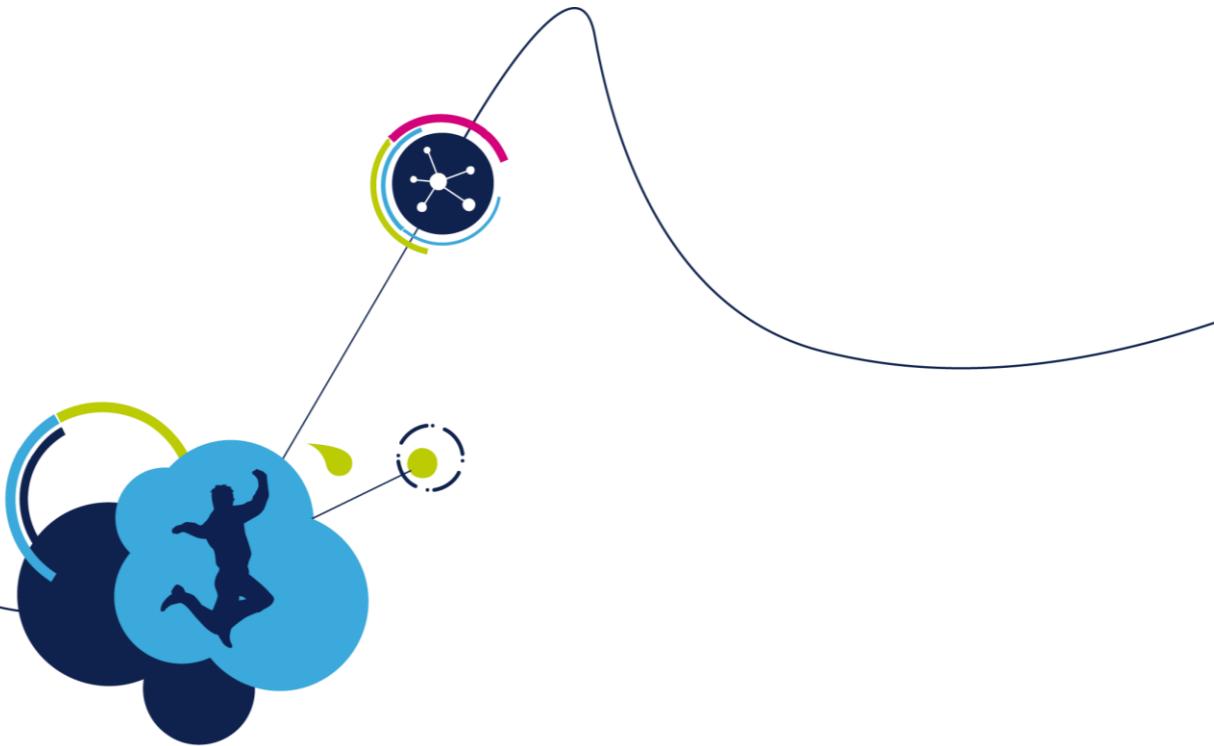
Bluetooth Low Energy

Bluetooth Low Energy

60

- Pour utiliser le BLE, nous vous proposons un apk Android :
 - NucleoCar_Rev2.apk
 - L'apk est disponible via la clef USB
 - Vous aurez bien sur besoin d'un smartphone Android avec le support du BLE.
- Vous pouvez bien sur utiliser d'autres applications de votre choix.





NFC

- Les tags NFC utilisés sont des tags de type 2, NXP NTAG213
- Pour lire le tag, utiliser la librairie STM32duino X-NUCLEO-NFC03A1 :
 - lire le champ URL / URI
 - le résultat est stocké dans une structure de type sURI_Info
 - L'information utile est dans le champ URI_Message

*****URI Reader*****

URI Information read successfully from the tag:

URI Information: [],
URI Protocol: [http://] ,
URI Message: [1234567890azerty]

