

## COMPTE RENDU – SÉANCE 7

Lors de cette séance, j'ai ajouté au programme de l'ESP32 des lignes de code permettant au propriétaire de la boîte de réinitialiser le code à distance. J'ai donc ajouté au fichier .html une zone de saisie avec envoi (formulaire) qui permet à l'utilisateur de saisir un code à 4 chiffres, ensuite récupéré et stocké dans une variable par l'ESP32.

```
<p>  
<form method="GET" action="/code">  
  <label for="code">Nouveau code :</label>  
  <input type="text" name="password" id="code" placeholder="Ex: 1234" maxlength="4"/>  
  <input type="submit" value="Envoyer" onclick="submitMessage()"/>  
</p>
```

Ces lignes permettent de créer un espace de saisie et d'envoyer l'information obtenue dans l'url de la page web sous la forme suivante : `http://adresse_ip_esp/code=1234` (si le nouveau code est 1234 par exemple).

Cette information est ensuite récupérée par l'esp32 et stockée dans une variable "code" :

```
server.on("/code", HTTP_GET, [](AsyncWebServerRequest *request){  
  String Code;  
  Code = request->getParam("password")->value();  
  code = Code;  
  Serial.println(code);  
  request->send(SPIFFS, "/index.html", String(), false, processor);  
});
```

La requête ci-dessus va chercher la valeur du paramètre ayant le nom "password" du fichier.html.

On obtient alors le résultat suivant :

**LOZA BOX**

**Voulez-vous autoriser l'ouverture de la boîte ?**  
État de la boîte : **FERME**

**Voulez-vous réinitialiser le code ?**  
Code actuel : **1234**  
Nouveau code :

On a ici une zone de texte pour saisir le nouveau code.



Une fenêtre nous indique ensuite que le code a bien été réinitialisé.



La page s'actualise en mettant à jour l'affichage du code actuel (donc le code qui vient d'être réinitialisé par le propriétaire).

```
Connecting to WiFi..  
Connecting to WiFi..  
Connecting to WiFi..  
Adresse IP: 192.168.0.34  
4325
```

Ce nouveau code est ensuite "sauvegardé" dans une variable sur arduino et affiché sur le moniteur série.

Nous avons ensuite essayé avec Elisa de rassembler tous les composants sur une même carte (ESP32). Pour cela il a donc fallu trouver les bibliothèques adaptées à la carte. Tout semble maintenant être compatible (le programme parvient à compiler seulement un message apparaît pour la bibliothèque du liquid crystal IC2 mais ne semble pas gêner l'exécution), néanmoins, bien que la carte soit une v1, elle semble fonctionner avec les pins d'une v2.

Nous avons également décidé de ne pas mettre de caméra dans la boîte ni d'alarme ne trouvant finalement pas ça très utile.