Ressources FiveM

- Utilisez la tablette XelBob-tab
- Ressources GTA V - Installation
- Ressources GTA V - Méthodes

Utilisez la tablette XelBob-tab

XelBob-tab est le produit de l'association entre Xelyos et Bob's&Co. Cette tablette utilisable sur Fivem vous permet d'afficher le contenu de sites internet directement sur votre serveur.

Afficher plus d'info



Ressources GTA V - Installation

Cette ressources n'est plus fonctionnelle pour le moment !

Installer la ressource

Pour récupérer la clé vous permettant d'utiliser l'api de Mail RP, rendez vous sur la page :

Récupérer sa clé API

Voici deux méthodes pour installer la ressources sur votre serveur :

Avec git :

Rendez vous dans le dossier `resources` de votre serveur, et utiliser la commande suivante :

```
git clone https://github.com/XelyosTeam/fivem-resources.git [xelyos]
```

Un dossier va allors de créer sous le nom **[xelyos]**, vérifiez bien que dans ce répertoire vous trouverez le répertoire **mailrp**.

Sans git :

Rendez vous sur le git : **Last releases Git Xelyos Resources FiveM** et télécharger le fichier `.zip`. Déplacez le fichier zip téléchargé dans votre répertoire `resources`, et faîtes *extraire-ici*, renommer le nouveau fichier installer en **[xelyos]**

Activer la ressource

Lier à l'API :

Pour lié votre serveur FiveM avec l'api de Mail RP, vous devez vous rendre dans le fichier `resources/[xelyos]/mailrp/server.lua`, au niveau de la ligne 6, remplacez **apiToken** par votre clé.

Activer la ressource :

Pour activer la ressource sur votre serveur, ajouter dans votre `server.cfg` les lignes suivantes :

```
# Xelyos solutions
ensure mailrp
```

Si vous utilisez déjà des ressources Xelyos, vous n'avez qu'à ajouter la ligne à la suite des autres.

Utiliser la ressource

La fonction callback :

```
function requestAction(statusCode, resultData)
  -- Action qui suit la requête
end
```

Lorsque vous allez faire appel à l'API, un résultat est envoyé en fonction de la requête que vous utilisez. Pour exploiter cette réponse, vous devrez utilisez une fonction *callback*, dans cette fonction écrivez les actions qui devront suivre la requête. Vous pouvez renommer la fonction *callback* comme vous le voulez.

Variable	Définition
statusCode **(obligatoire)**	Résultat du status de la requête (200 : réussite | 403 : échec)
resultData **(obligatoire)**	Données retournées au format JSON

Appelez le code depuis un fichier serveur :

Pour utiliser l'api depuis un fichier serveur, vous devez utiliser la structure suivante :

```
function requestAction(statusCode, ResultData)
  ...
end

TriggerEvent('xelyos:MailRP', function(mailRP)
    mailRP:methode(requestAction)
end)
```

Afficher la liste des méthodes

Vous trouverez la liste des méthodes ci-dessous !

Appelez le code depuis un fichier client :

Pour utiliser l'api depuis un fichier serveur, vous devez utiliser la structure suivante :

```
function requestAction(statusCode, ResultData)
  ...
end

TriggerServerEvent('xelyos:MailRP', function(mailRP)
    mailRP:methode(requestAction)
end)
```

Afficher la liste des méthodes

Exemple d'utilisation - Récupérer le nom de son serveur

```
function showServerInfo(statusCode, resultData)
    print("Status", statusCode) -- Status de la requête
    print('Result', resultData) -- Informations retournées
  print('Server name', resultData.name) -- Récupération du nom du serveur
    print('Server count api request', resultData.api.count) -- Récupérer le nombre de requêtes effectuées au total pour le serveur
end

AddEventHandler('onResourceStart', function(resourceName)
    if GetCurrentResourceName() == resourceName then
      -- Lors de l'initialisation de la ressource
        TriggerEvent('xelyos:MailRP', function(mailRP)
            mailRP:getServer(showServerInfo)
        end)
    end
end)
```

Ressources GTA V - Méthodes

Cette ressources n'est plus fonctionnelle pour le moment !

Chaque méthode possède dans ses paramètres une variable appelée `callBack`, cette variable correspond à la fonction que vous utilisez pour le traitement des données renvoyées par l'API. Chaque variable utilisée dans les différentes méthodes sont expliquées sur le document API.

---

getServer()

Utilité : Récupérer les informations sur son serveur

Doc API : Gestion du serveur - Informations générales

Appel :

```
mailRP:getServer(callBack)
```

---

findSubdomains()

Utilisé : Récupérer la liste des sous domaines de votre serveur

Doc API : Gestion des sous domaines - Liste des sous domaines

Appel :

```
mailRP:findSubdomains(callBack)
```

---

findSubdomain(name)

Utilisé : Récupérer les informations d'un sous domaine avec ses adresses mails

Doc API : Gestion des sous domaines - Détails d'un sous domaine

Appel :

```
mailRP:findSubdomain(callBack, name)
```

---

addSubdomain(name, public, manager, carnet)

Utilisé : Créer un nouveau sous domaine

Doc API : Gestion des sous domaines - Ajouter un sous domaine

Appel :

```
mailRP:addSubdomain(callBack, name, public, manager, carnet)
```

---

findAddress(email)

Utilisé : Récupérer les informations d'une adresse mail

Doc API : Gestion des adresses - Récupérer une adresse mail

Appel :

```
mailRP:findAddress(callBack, email)
```

---

addAddress(name, firstname, email, password)

Utilisé : Créer une adresse mail

Doc API : Gestion des adresses - Créer une adresse mail

Appel :

```
mailRP:addAddress(callBack, name, firstname, email, password)
```

---

sendEmail(from, to, subject, content, confirmOpened, attachments)

Utilisé : Envoyer un mail depuis une adresses à un ou plusieurs destinataires

Doc API : Gestion des mails - Envoyer un mail

Appel :

```
mailRP:sendEmail(callBack, from, to, subject, content, confirmOpened, attachments)
```