

Protocole de communication entre l'application et le serveur

Liste des escapes games :

Requête client :

```
{  
  "request":"GET_ESCAPE"  
}
```

Réponse :

```
{"escapegames":  
  [  
    {"enigmes":  
      [  
        {  
          "reponse":"12.5",  
          "isresolve":false,  
          "nom":"A la piscine","infos":"Sur le couloir n°1, Paul part en crawl côté plongeur, mais avec un départ non plongé.\n\nSur le couloir n°2, Virginie part en même temps, en brasse coulée, du côté du petit bain. Ils nagent à vitesse constantes. Il se croisent une première fois à 5 m de l'extrémité de la piscine, côté petit bain. Puis tous deux, l'un après l'autre, font demi-tour et repartent l'un vers l'autre en conservant la même allure. Ils se croisent une deuxième fois à 2,5 m de l'extrémité côté plongeur. \n\nQuelle est la longueur de la piscine ?"},  
        {"reponse":"Informatique",  
          "isresolve":false,  
          "nom":"Le code pour les nuls",  
          "infos":"Dans quel rayon se trouve le livre \"Le code pour les nuls\" au learning centre"}  
      ]  
    }  
  ]  
}
```

```

    ],
    "nom": "Que des numeros 10 dans ma team",
    "temps": 60,
    "infos": "A bord d'un paquebot flambant neuf, une avarie survient.
L'équipage vous donne pour consigne de quitter le navire. Panique à bord ! Les passagers se
bousculent et se poussent pour atteindre au plus vite les canots de sauvetage. Dans la cohue, vous
prenez un mauvais coup et perdez connaissance. A votre réveil, il ne reste plus que vous à
bord"
  }
]
}

```

Création d'une partie

Requête client :

```

{
  "request": "CREATE_PARTIE",
  "escapegame": "Sherlock",
  "username": "Claude",
  "teamname": "TeamL"
}

```

La valeur de « escapegame » correspond au nom de l'escape game qu'il veut jouer, il donne aussi son nom de joueur et équipe.

Une partie seule est défini par un nom d'équipe vide, il reçoit alors :

```

{
  "reponse": "ok",
  "nom": "A la piscine",
  "temps": 60,
  "infos": "première énigme ",
  "idpartie": 1
}

```

Il reçoit alors la première énigme avec le nom de l'énigme et aussi le descriptif de celle-ci « infos » mais le temps total pour résoudre celui-ci. Mais aussi l'id de la partie primordiale pour continuer l'escape Game.

Sinon s'il joue en équipe il reçoit :

```
{  
    "reponse":"ok",  
    "idpartie":1  
}
```

Liste des salons

Si l'on veut jouer en équipe, nous arrivons sur une page avec la liste des salons de jeu en attentes de joueurs. L'application les récupère avec la requête :

```
{  
    "request":"GET_SALONS"  
}
```

Il reçoit de la part du serveur un tableau de partie en cours avec le nom de l'équipe, l'escape game qui va être réaliser, le nombre de joueur actuellement dans le salon et l'id de la partie.

```
{  
    "partieattente":  
        [  
            {"escapegame":"Sherlock",  
             "equipe":"PolyEscape",  
             "idpartie":1,  
             "joueurs":1}  
        ]  
}
```

Rejoindre une partie

Si l'on joue en équipe et que la partie n'a pas commencé, on peut rejoindre un salon pour cela. Il faut faire la requête suivante :

```
{  
    "request":"JOIN_TEAM",  
    "idpartie" = 1,  
    "username":"Pierre"
```

```
}
```

Il y a alors deux possibilités, si le salon est plein, il recevra alors :

```
{  
    "reponse":"ko"  
}
```

Sinon s'il réussit à rejoindre la partie, il recevra :

```
{  
    "reponse":"ok"  
}
```

Ainsi qu'un message du serveur qui sera envoyé à tous les membres du salon :

```
{  
    "reponse":"actualise",  
    "joueurs":[  
        {"ready":true,"username":"Polo"},  
        {"ready":false,"username":"Claude"}  
    ]  
}
```

Indiquer que l'on est prêt

Si l'on joue à plusieurs, on peut indiquer à tout moment que l'on est prêt, pour cela le client fait une requête au serveur :

```
{  
    "request":"SET_READY",  
    "idpartie" = 1,  
    "username":"Pierre"  
}
```

Le serveur gère lui-même s'il doit mettre le joueur prêt ou pas.

Le serveur répond à tous les membres du salon le message suivant :

```
{  
    "reponse":"actualise",  
    "joueurs":[
```

```
    {"ready":true,"username":"Polo"},  
    {"ready":false,"username":"Claude"}  
  ]  
}
```

Lorsque tout le monde est prêt, le serveur envoie à chaque joueur la première énigme qu'il doit résoudre sous la forme suivante :

```
{  
  "reponse":"enigme",  
  "nom":"A la piscine",  
  "temps":60,  
  "infos":"Enoncé de l'énigme"  
}
```

Repondre à une enigme

Requête client :

```
{  
  "request":"RESPONSE",  
  "reponse":"ceci est ma reponse",  
  "username":"Jean",  
  "idpartie":1  
}
```

Lorsqu'un joueur veut répondre à une énigme, celui-ci a uniquement besoin d'indiquer l'id de la partie, la réponse qu'il veut soumettre et son nom.

Dans ce cas 4 types de réponse possible, tout d'abord si l'utilisateur répond faux :

Si l'utilisateur répond juste et qu'il lui reste encore des énigmes à répondre :

```
{  
  "reponse":"ok",  
  "nom":"A la piscine",  
  "temps":60,  
  "infos":"Enigme suivante"}  
}
```

Si le joueur joue en équipe et qu'il a fini toutes ses énigmes mais pas ces équipiers il reçoit :

```
{  
    "reponse":"notyet"  
}
```

Et enfin si tout le monde a fini il reçoit :

```
{  
    "reponse":"finish",  
    "score":100  
}
```

Il faut savoir aussi lorsque le joueur répond correctement à une question, il indique à tous les joueurs de la partie à travers le message suivant :

```
{  
    "reponse":"success",  
    "joueurs":[  
        {"actual":3,"total":3,"username":"pierre"},  
        {"actual":1,"total":5,"username":"paul"}  
    ]  
}
```

Dans ce message, on indique pour chaque joueur le nombre d'énigmes total qu'il doit réaliser et ainsi que l'énigme actuel du joueur et son nom. Ce tableau est automatiquement mis à jour lorsqu'un joueur répond correctement.

De la même façon, lorsqu'un joueur découvre un indice tous les joueurs reçoivent le message :

```
{  
    "reponse":"indices",  
    "indices":[  
        {"indice":" indice de pierre","username":"pierre"}  
    ]  
}
```

Protocole de communication pour la demande d'aide

Demande d'aide de la part du joueur

Requete du client

```
{  
    "request":"HELP",  
    "enigme":" enigme actuelle du joueur ",  
    "username":"Jean",  
    "idpartie":1  
}
```

On envoie directement un message au maître du jeu :

```
{  
    "reponse":"request",  
    "username":"nom du joueur",  
    "enigme":"enigme actuelle du joueur",  
    "idpartie":1  
}
```

Puis on répond au joueur

```
{  
    "reponse":"ko"  
}
```

Si aucun maître du jeu n'est présent sinon on répond

```
{  
    "reponse":"ok"  
}
```

Indice de la part du maître du jeu

Lorsque le maître envoie son indice au joueur il doit faire la requete suivante

```
{  
  "request":"INDICE",  
  "indice":" indice pour le joueur joueur ",  
  "username":"Jean",  
  "idpartie":1  
}
```

Et nous renvoyons directement la requête au joueur