Il est plus important de documenter la nature des informations échangées que le code en tant que tel puisque le protocole fait partie de l’interface publique de votre client ou serveur.

Protocole

Nature et séquence des messages échangés

Michaël Turcotte

Liste des messages échangés

Ici vous listez les messages ou les variables échangées par les différentes instances logicielles. Pour chaque variable qui prend une liste définie de valeurs possibles, on liste ces valeurs. Lorsque la liste des variables associées dépend du message envoyé, on liste les autres variables associées.

Lorsqu’un message possède une structure complexe tel qu’un objet ou un tableau, on décrit complètement cet objet, ainsi que la liste des valeurs possibles ou ses limites.

## Envoi de message d’Actions (en énumération)

***MESSAGES d’ACTION***

|  |  |
| --- | --- |
| Messages | Valeurs possibles |
| action : string | changerRole |
| partieTerminee |
| partieProlongee |
| partieGagnant |
| butMarque |
| defenseurPasseLigneCentre |
| tirer |
| miseEnEchec |
| perdreRondelle |

## Actualisation de variables de salon

***VARIABLES DE SALON***

|  |  |
| --- | --- |
| Variables | Valeurs possibles |
| pointageJoueur1 : int | Entre 0 et 1000 |
| pointageJoueur2 : int | Entre 0 et 1000 |
| nomJoueur1 : String | Entre 1 et 25 caractères |
| nomJoueur2 : String | Entre 1 et 25 caractères |
| tempsRestant : int | Entre 0 et 20 |

## Actualisation de tableaux de salon

***TABLEAUX DE SALON***

|  |  |  |
| --- | --- | --- |
| Tableaux | Composition par objet de tableau | Valeurs possibles |
|  |  |  |
|  |  |
|  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Actualisation d’objets de salon

***OBJETS DE SALON***

|  |  |  |
| --- | --- | --- |
| Objet | Composition | Valeurs possibles |
| attaquant : Attaquant | positionX : unsigned int | positionX : unsigned int |
| positionY : unsigned int | positionY : unsigned int |
| rotation : unsigned float | rotation : unsigned float |
| enPossenssion : boolean | enPossenssion : boolean |
| defenseur : Defenseur | positionX : unsigned int | positionX : unsigned int |
| positionY : unsigned int | positionY : unsigned int |
| rotation : unsigned float | rotation : unsigned float |
| enPossenssion : boolean | enPossenssion : boolean |
| rondelle : Rondelle | positionX : unsigned int | positionX : unsigned int |
| positionY : unsigned int | positionY : unsigned int |
| vitesse: unsigned float | vitesse: unsigned float |

Séquence des requêtes et messages

Dans ce diagramme, on illustre une séquence probable de messages échangés entre le client et le serveur.

**CLIENT A**

**CLIENT B**

**SERVEUR**

\*

else

gagnant

gagnant

If pointageJoueur1 > pointageJoueur2

prolongation = true

prolongation = true

If pointageJoueur1 == pointageJoueur2

tempsRestant != 0 && !prolongation

perdreRondelle

miseEnEchec

echangerRole

positionRondelle

positionRondelle

tirer

renitialiserPosition

renitialiserPosition

renitialiserPosition

renitialiserPosition

echangerRole

If defenseurPasseLigneCentre

miseAJourPointage

miseAJourPointage

If butMarque

defenseur

attaquant

CONNECT

CONNECT