

# GEN-Yahtzee

## Fonctionnement général de l'applcatif

### Objectifs de base

Dans le cadre du cours de Génie Logiciel (GEN), nous avons pour projet, de concevoir une application client-serveur en adoptant la méthodologie de travail, Unified Process (UP). Nous avons choisi de réaliser une version digitale du Yahtzee, célèbre jeu de société.

### Utilisation de l'applcatif

Le client est accueilli sur une fenêtre d'authentification où il a la possibilité de spécifier l'adresse d'un serveur distant ainsi que ses identifiants, pour s'y connecter. Ensuite, il est redirigé sur la fenêtre principale du serveur. Pour jouer, il peut rejoindre une partie en attente de joueurs ou, en créer une nouvelle. Le cas échéant il doit spécifier le nombre de participants (2 au minimum) et leur temps à disposition pour chaque tour.

Le client a la possibilité de modifier son profil (mot de passe, avatar, description). Le serveur quant à lui dispose d'outils d'administrations permettant, d'ajouter, de modifier ou de supprimer un joueur. Il peut aussi configurer le moteur de jeu et ainsi permettre, par exemple, la création de partie composée de N joueurs.

L'interface de jeu permet d'effectuer les actions permises, de visualiser sa progression (et celle de ses adversaires) et d'obtenir un résumé des combinaisons gagnantes. Il est prévu aussi de mettre à disposition un système permettant de communiquer lors d'une partie. Le joueur a la possibilité d'observer une partie et visualiser les classements s'il ne désire pas jouer.

Lorsque qu'une partie débute, le serveur choisit aléatoirement le joueur qui commencera. A terme, le serveur enregistre les scores des différents joueurs et dresse un classement.

### Règles du jeu

Le Yahtzee se joue avec 5 dés et se finit une fois toutes les cases de la fiche de score remplies. Chaque joueur joue tout à tour et dispose de 3 lancers à chaque coup. L'objectif étant de réaliser des combinaisons qui rapportent des points. Le joueur a le choix de reprendre tous ou une partie des dés à chaque lancé, selon son gré, pour tenter d'obtenir la combinaison voulue. A chaque tour, le joueur doit obligatoirement inscrire les points qu'il a obtenu dans la fiche de score.

Le premier joueur lance les dés et choisit de les garder tous, de les rejouer tous ou rejouer seulement certains dés afin d'obtenir l'une des combinaisons.

Après le dernier jet de dé, il doit remplir la fiche de score. S'il n'obtient aucune combinaison ou n'est pas satisfait du score obtenu et pense pouvoir faire mieux, il a la possibilité d'inscrire 0 point dans la case de son choix, sachant qu'il ne pourra plus refaire cette combinaison par la suite. Le gagnant est celui qui obtient le plus de points. Pour compter les points, se référer à la fiche de score.

Lorsque le total intermédiaire est égal ou supérieur à 63 points, un bonus de 35 points supplémentaires est accordé, ce qui peut faire la différence au décompte final. Il faut donc être stratégique.

#### Contraintes

- Un joueur ne peut rejoindre une partie en cours (y compris dans le cas où il s'y est déconnecté involontairement);
- Une partie ne peut être composée de plus de 6 joueurs;
- Le protocole client-serveur n'est pas chiffré.

### Responsabilités client et serveur

#### Serveur

- Gestion de la base de données
- Instancier les parties
- Dresser les classements
- Mise en relation des joueurs
- Logique du jeu
- Configuration du serveur
- Valider authentications utilisateurs

#### Client

- Authentification auprès du serveur
- Mise-à-jours des vues

### Cas d'utilisation

#### S'inscrire sur un serveur

##### Scénario principal (succès)

1. Le joueur se rend sur le menu principal
2. Le joueur se rend sur le menu "S'inscrire..."
3. Le joueur enregistre l'adresse IP du serveur, un nom d'utilisateur et un mot-de-passe
4. Le joueur confirme l'inscription

5. Le serveur envoie un "acknowledge" 6. Affichage acknowledge

##### Autres scénarios (échecs, variantes)

#### Opération 3: Le serveur est introuvable

1. Le joueur est notifié
2. Le joueur corrige l'adresse IP

#### Opération 3: Le nom d'utilisateur existe déjà

1. Le joueur est notifié

Pré-requis : un serveur est lancé... ?  
On connaît son adresse ?  
Comment ?

Diagramme ?

Acteurs ?

Pourquoi enregistrer ?  
→ "Saisir" plutôt

2. Le joueur enregistre un nom d'utilisateur différent

**Opération 4: Le joueur annule la procédure**

1. Le joueur abandonne l'enregistrement

**S'authentifier sur un serveur**

**Scénario principal (succès)**

- (1. Le joueur se rend sur le menu principal) *Prérequis*  
2. Le joueur spécifie le serveur auquel il veut se connecter  
3. Le joueur spécifie son nom d'utilisateur et le mot-de-passe associé  
4. Le joueur confirme la connexion  
5. Le serveur ? *envoie acknowledgment*

**Autres scénarios (échecs, variantes)**

**Opération 2: Le serveur est introuvable** *(c'est ça)*

1. Le joueur est notifié  
2. Le joueur corrige l'adresse IP du serveur

**Opération 2: Le serveur ne répond pas**

1. Le joueur est notifié  
2. Le joueur renouvelle sa demande plus tard

**Opération 3: Le mot-de-passe est incorrect**

1. Le joueur est notifié  
2. Le joueur corrige son mot-de-passe

*Les cas pour échecs, les cas sont des variantes ?  
(A préciser au cas par cas)*

**Créer une partie**

**Scénario Principal (succès)**

- (1. Le joueur se rend dans l'espace prévu à cet effet) *Prérequis*  
2. Le joueur spécifie le nombre de joueurs admissible  
3. La partie est mise en attente tant qu'il n'y a pas assez de joueur  
(4.) *est créée, puis mise en attente.*

**Rejoindre une partie**

*Que voit le joueur ? A t'il rejoint la partie créée ?*

**Scénario Principal (succès)**

1. Le joueur se rend sur la liste des parties en attente  
2. Le joueur notifie au serveur quelle partie il veut rejoindre

*Que voit le joueur ?  
Uniquement la liste de parties ?*

**Observer une partie**

**Scénario Principal (succès)**

*Que voit le joueur une fois qu'il a rejoint la partie ?*

*Acknowledge ?*

*si 2 joueurs s'inscrivent simultanément avec 1 seul joueur attendu ?*

- Eventuellement le nombre de joueurs ayant rejoint la partie
- le nombre de joueurs attendu ?



1. Le joueur se rend sur la liste de toute les parties *→ en attente ? en action ?*
2. Le joueur notifie au serveur quelle partie il veut observer *pendant si le parti*

### Administrer les utilisateurs du serveur

#### Scénario Principal (succès)

1. L'administrateur se rend sur la fenêtre de configuration du serveur
2. L'administrateur sélectionne un utilisateur du serveur
3. L'administrateur accomplit l'action souhaitée (Suppression, ban, ...)

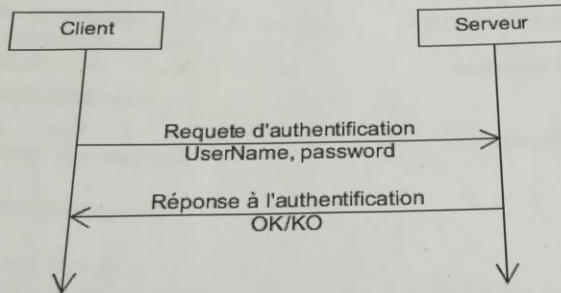
### Quitter le serveur

#### Scénario Principal (succès)

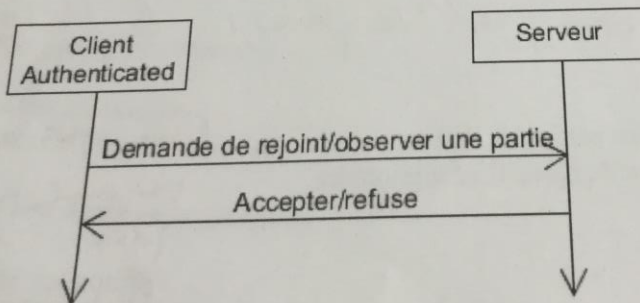
1. Le joueur quitte le serveur avec le dispositif prévu à cet effet
2. Le joueur est redirigé sur l'écran d'accueil

## Protocole client-serveur

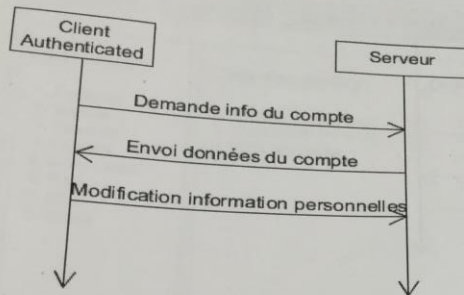
### Authentication



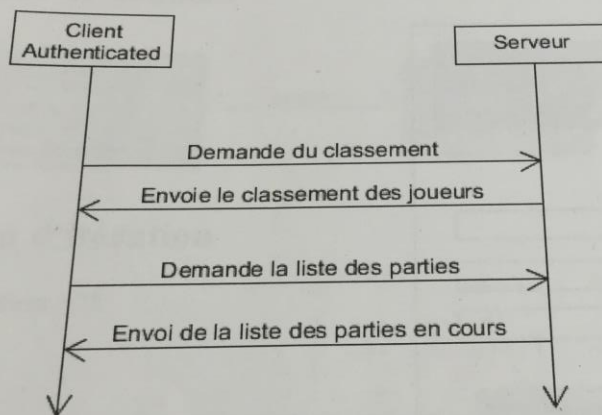
### Rejoindre/Observer une partie



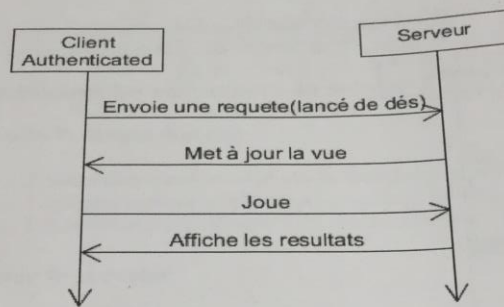
### Modification du profil



### Classement des joueurs



### Jouer une partie



## Ebauche UI

### Fenêtre d'authentification

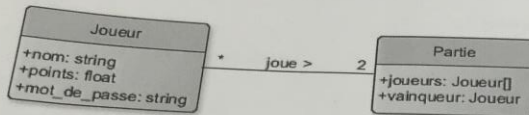
The sketch shows a rectangular window for authentication. At the top is a grey rectangular area labeled 'LOGO'. Below this, the text 'Authentification:' is followed by three input fields. The labels for these fields are 'IP Serveur:', 'Login:', and 'Mot de passe:'. At the bottom of the window are two buttons: 'S'inscrire' on the left and 'Entrer' on the right.

### Fenêtre principale

Règle du jeu Admin Editer profil Quitter

Classement:	Liste des parties:
1. Crabone	1. Nadine VS Eduardo 2/2 -> <0>
2. Nadine	2. Wasabi VS Paul VS ... 3/5 -> <0>
3. Eduardo	3. Elisa VS Curtis 2/3 -> <0>
4. Wasabi	
5. limayankée	
6. Paul	
7. Elisa	
8. Curtis	

## Base de données



## Plan d'itération

### Itération n°1

#### But

Développer une architecture (minimale) d'une application client-serveur, permettant l'échange de messages entre plusieurs clients et un serveur.

#### Dates

Du 21 avril 2016 jusqu'au 28 avril 2016.

#### Phase

Implémentation et conception.

#### Fonctionnalités

- Serveur pouvant gérer plusieurs clients simultanément ;
- Connexion d'un ou plusieurs clients sur le serveur ;

Pour chaque itération —

+ Effort escompté à l'avance  
+ Partage du travail / réajustement

Annouer : (pour chaque itération)

- Introduction ?
- Fonctionnalités ? / cas d'utilisation
- Autres (conception, etc...) 10/100



- Communication textuelle entre le client et le serveur;
- Gestion propre des états client (Connexion, déconnexion);
- Monitoring de la part du serveur, des clients associés. *c.a.d.?*

#### Démonstration

- Des clients doivent pouvoir se connecter au serveur;
- Visualisation des échanges entre les clients et le serveur (Par Wireshark).

#### Rôles

Fabien: Responsable normes et procédures  
 Kevin: Programmeur  
 Ibrahim: Analyste  
 Madolyne: Responsable documentation  
 Rosanne: Responsable des tests

#### Itération n°2

##### But

Implémenter le protocole client-serveur.

##### Dates

✓ Du 28 avril 2016 au 5 mai 2016.

##### Phase

Implémentation et conception.

#### Fonctionnalités

- Le client et le serveur communique selon le protocole mis en place;
- Le protocole contient toutes les interactions entre le client et le serveur t.q. la création d'une partie et le déroulement d'une partie;
- Ce protocole sert de base pour implémenter la mécanique du jeu.

#### Démonstration

- Le client et le serveur s'échange des messages selon le protocole établi (Visualisation par Wireshark).

#### Itération n°3

##### But

Implémentation du moteur et de la mécanique du jeu.

##### Dates

*→ Qu'est-ce que l'on pourra observer?*

*Par exemple - contenu de message envoyé  
 - message reçu et décodage.*

*leur propre type de message?*



Du 5 mai 2016 au 12 mai 2016 (avec une réserve d'une semaine). ✓

**Phase**

Implémentation et conception.

**Fonctionnalités**

- Plusieurs clients pourront jouer des parties;
- Le serveur peut accepter plusieurs parties simultanées.

**Démonstration**

- Une partie (en ligne de commande) entre 3 joueurs.

**Itération n°4**

**But**

Implémentation de la gestion des parties et du classement.

**Dates**

Du 12 mai 2016 au 19 mai 2016. ✓

**Phase**

Implémentation et conception.

**Fonctionnalités**

- Le client peut créer une partie;
- Le client peut rejoindre une partie;
- Le client peut observer une partie.

**Démonstration**

- Démonstrations des 3 fonctionnalités.

**Itération n°5 (+ 6)**

**But**

Implémentation de l'interface graphique

**Dates**

Du 19 mai 2016 au 2 juin 2016. ✓

**Phase**

2 semaines.

+ Quels cas d'usage?  
Quelles étapes?

#

Range la 6

(2 juin  
au 9 juin)

