

# Projet PC2R 2012

## Bataille Navale Royale

### 1 Introduction

*Jeudi 20 décembre 2012 : Cher journal, c'est demain que je dois rendre mon projet de PC2R. Je suis très impatient car je me suis donné à fond pour impressionner mon chargé de TD et ainsi avoir une bonne note. Je vais y passer la nuit afin de peaufiner quelques détails.*

*Vendredi 21 décembre 2012, 7h30 : Cher journal, nos enseignants nous ont convoqués à 14h afin de présenter oralement nos projets. Ce n'était pas prévu et leur courriel était n'était pas très cordial, je me demande s'il n'y a pas eu de la triche. Enfin, pour ma part, je suis très content de pouvoir présenter mon travail à mes enseignants adorés. Je vais bien m'habiller car il fait très froid aujourd'hui, et le vent souffle très fort.*

*Vendredi 21 décembre 2012, 14h30 : Cher journal, j'ai eu du mal à arriver à Jussieu, le métro s'est arrêté sans qu'on nous donne de raison. J'ai dû terminer à pied, mais j'ai les pieds mouillés. Il semblerait que la Seine sorte un peu de son lit. Je profite d'une petite pause dans le couloir du bâtiment X (je ne savais même pas qu'il existait !) pour t'écrire avant d'aller présenter mon projet, mais ce n'est pas facile car les lumières vacillent. J'espère que cette tempête sera bientôt finie !*

*Samedi 22 décembre 2012 : Cher journal. Je suis fort contrarié. Comme il n'y avait plus d'électricité, je n'ai pas pu présenter mon projet. En plus, je n'ai pas pu rentrer chez moi car l'eau continue à monter. Elle est maintenant presque au dernier étage ! Et ma maman m'avait promis un délicieux gâteau si je ramenaient une bonne note... Il y avait une réserve de nourriture à l'étage, donc au moins, nous sommes rassasiés. Nous nous occupons comme nous pouvons. Nos enseignants s'affairaient sur un programme compliqué en Esterel. Ça a l'air de bien les amuser car ils ricanent tout le temps, mais ils ne veulent pas nous le montrer (c'est sûrement pour noter automatiquement nos projets.)*

*Dimanche 23 décembre 2012 : Visiblement, c'est vraiment une tempête hors du commun. Impossible de capter la radio ou un quelconque réseau. Certains étudiants cèdent à la panique et parlent de la fin du monde, cela n'aide pas à améliorer la situation. Heureusement que nos enseignants gèrent parfaitement la situation, et surtout qu'ils connaissent l'existence du dispositif de caissons de survie du campus. Et heureuse coïncidence, il y en avait juste assez pour tous les étudiants de PC2R sur le toit du bâtiment X. Je suis dans un petit caisson à deux places avec mon binôme. Visiblement, beaucoup d'étudiants sont claustrophobes, car il a fallu que nous aidions nos enseignants à les faire rentrer de force dans leurs caissons (mais c'est pour leur bien). Nous avons de la nourriture pour tenir un bon moment, mais j'espère que nous n'en aurons pas besoin ! Il y a aussi un ordinateur, mais impossible de comprendre son fonctionnement, il semble ne lancer qu'un programme affichant une sorte de grille avec un curseur.*

*Mardi 25 décembre 2012 : Cher journal, c'est aujourd'hui que j'aurais dû recevoir mes présents du papa Noël, mais je suis coincé avec mon binôme dans ce caisson de survie. Quelle déveine ! Comme nous avons terminé nos livres, nous avons commencé à explorer le caisson. Nous avons découvert un petit hublot. Visiblement la Seine est encore montée et nous sommes submergés. Par contre, impossible de distinguer quoi que ce soit tellement l'eau est trouble. Mon binôme s'inquiète beaucoup, il dit qu'une telle montée des eaux est impossible. j'espère qu'il va réussir à garder son calme.*

*Mardi 31 décembre 2012, 20h00 : L'ordinateur s'est activé. Sur l'écran, les enseignants de PC2R sont apparus. Nous avons essayé de parler mais ils ne semblaient pas nous entendre. Ils nous ont demandé d'ouvrir une boîte logée dans un compartiment précis et recommandé de bien profiter du repas de fête qu'elle contenait. Puis ils se sont mis à ricaner et la connexion s'est coupée. Cher journal, je ne comprends plus la situation. Mon binôme commence vraiment à perdre son sang froid.*

*Mercredi 1er janvier 2013, 00h00 : Le caisson s'est déplacé tout seul. Il nous semble que nous sommes descendus très profond. Par le hublot tout est complètement noir. Au moment où j'écris ces lignes, l'écran de l'ordinateur s'est modifié. Il indique :*

- > BIENVENUE DANS LE DISPOSITIF DE SAUVETAGE COOPÉRATIF*
- > VOUS ÊTES DANS L'ÉQUIPE 2*
- > PLACEZ-VOUS*

*Sur la grille, deux cases sont en surbrillance.*

*Mon binôme a appuyé sur une touche de direction, le caisson s'est déplacé ! Visiblement, nous devons piloter ces caissons pour sauver notre vie, mais quelle est cette histoire d'équipe et de coopération ? !*

*Mercredi 1er janvier 2013, 01h00 : Après de nombreux déplacements, impossible de comprendre comment se diriger. Il semblerait qu'on ne puisse se mouvoir qu'à l'horizontale et dans un espace restreint. Nous avons arrêté de nous acharner, peut être que les enseignants vont nous réapparaître et nous guider ?*

*Mercredi 1er janvier 2013, 02h00 : Au bout d'un moment l'ordinateur a affiché :*

- > PLACEMENT TERMINÉ.*

*À ce moment, d'autres cases sont apparues en surbrillance sur l'écran. En même temps, nous avons pu entendre la voix d'autres étudiants. La plupart sont en plein délire. Certains pensent qu'ils sont dans un jeu vidéo. D'autres crient au complot mondial. De beaucoup, on n'entend que des sanglots ou des morceaux de phrases insensées.*

*Mercredi 1er janvier 2013, 02h10 : L'ordinateur affiche :*

- > A VOUS DE JOUER.*
- > PRESSEZ (HAUT/BAS/GAUCHE/DROITE) POUR DÉPLACER LE MODULE DE SAUVETAGE.*
- > PRESSEZ (ENTRÉE) POUR ACTIVER LE MODULE DE SAUVETAGE.*

*Visiblement, les autres n'ont pas ce message, c'est à moi de choisir.*

*Le curseur est sur notre caisson, je presse ENTRÉE pour activer le module de sauvetage.*

*Mercredi 1er janvier 2013, 02h20 : Après 10 minutes, l'ordinateur a affiché :*

- > ACTIVATION DU MODULE DE SAUVETAGE.*

*Puis une grande lumière rouge a envahi le caisson, suivie d'une odeur de brûlé désagréable. Et mon binôme a disparu. Je suppose qu'il a été télé-porté dans une zone de sauvetage, ou quelque chose comme ça, par le module de sauvetage !*

*Ce tour-ci je déplace le curseur sur la seconde case de notre caisson, au prochain tour je presserai ENTRÉE pour activer le module de sauvetage. Dans les hauts parleurs, j'entends les collègues de mon équipe me traiter de fou, et me dire que je n'ai rien compris. Ils auraient voulu que je tire sur les cases vides, je crois qu'ils déraillent complètement.*

*Mercredi 1er janvier 2013, 02h40 : J'ai activé le module de sauvetage sur ma case. L'écran affiche :*

- > ACTIVATION DU MODULE DE SAUVETAGE.*
- > VOUS N'AVEZ PAS RÉUSSI VOTRE SAUVETAGE.*

*Saperlipopette, je crois que je viens de comprendre. Les solutions données en ME, les canettes de coca gratuites, les examens trop courts et trop faciles... Tout ça c'était pour nous amadouer ! Si vous avez découvert ce journal, vous connaissez maintenant les coupables de la grande apocaly*

## 2 Travail principal

Ce projet consiste à implanter le jeu *Bataille Navale Royale*, une variante de la bataille navale. Ce jeu se déroule de la façon suivante :

1. Entre 2 et 4 joueurs doivent placer chacun une flotte de sous-marin dans une aire carrée donnée. L'aire est discrétisée en une grille de 16 par 16 cases. Chaque joueur doit y placer 2 sous-marins occupant 1 case, 2 sous-marins occupant 2 cases adjacentes, 2 occupant 3 cases consécutives sur une ligne horizontale ou verticale, et 1 occupant 4 cases de la même façon. Si deux joueurs occupent une même case, on suppose que la chance qu'ils soient à la même profondeur est nulle, cela ne pose donc pas de problème.
2. Une fois le placement terminé, chaque joueur pilote un petit drone, se déplaçant de case en case sur la grille à la surface de l'eau. Le jeu se déroule au tour par tour, chaque joueur ayant la main une fois par tour (à tour de rôle). À chaque tour, un joueur peut déplacer le drone d'un nombre de cases égal à 17 moins le nombre de parties de ses sous-marins non touchées (soit 1 au départ et 16 lorsqu'il ne lui reste plus qu'une partie de sous-marin non détruite). À la place d'une des actions de déplacement, il peut choisir d'activer son drone. Dans ce dernier cas, un laser surpuissant s'active, endommageant tous les sous-marins à la verticale du drone. Le drone communique alors au joueur s'il a touché un sous-marin ou non (mais il ne peut savoir s'il en a touché plusieurs ou s'il a mis un coup fatal).
3. Lorsque tous les sous-marins d'un joueur sont complètement détruits, celui-ci ne peut plus jouer. Le dernier joueur ayant au moins une partie d'un sous-marin non détruite gagne la partie.

### Travail à réaliser :

1. Programmer un serveur et un client dans deux langages différents, communiquant par le protocole décrit dans ce document. Chaque client devra gérer pour un joueur la connexion au serveur, le placement des vaisseaux et la phase de déplacement et d'activation du drone. Votre client devra produire une représentation visuelle de la carte et du drone ainsi que des résultats des actions entreprises. Le serveur devra garder état de tous les joueurs, gérer l'interface entre le joueur et son drone ainsi que la synchronisation des tours de jeu et la décision de fin de partie.
2. Rédiger un petit rapport de 4 à 8 pages, avec une limite de deux captures d'écran et des marges et tailles de fonte raisonnables.

### 2.1 Protocole

Le serveur écoute sur son port 2012 en TCP. Les deux parties parlent un protocole texte qui prend la forme d'une succession de commandes.

Chaque commande est composée d'une suite de chaînes terminées par des caractères '/'. La dernière chaîne est suivie d'un retour chariot Unix '\n'. La première chaîne est le nom de la commande, les autres ses arguments. Chaque nom de commande n'a qu'un seul rôle dans le protocole (le même nom de commande ne peut signifier deux choses différentes à deux moments différents).

La suite "\" dans un argument est utilisée pour échapper le caractère '/', et "\" pour échapper le caractère '\\'.

**Conseil:** tirez parti des possibilités des langages utilisés pour rendre automatique et sûre la lecture et l'interprétation des commandes.

### 2.2 Connexion

Lorsque le client se connecte au serveur, il envoie sans attendre la commande **CONNECT** suivie du nom du joueur précisé par l'utilisateur. Le serveur répond par une commande **WELCOME** suivie du nom du joueur (qui peut être un peu différent dans le cas où plusieurs joueurs auraient demandé le même nom).

Les parties se jouent de deux à quatre joueurs. La connexion du second joueur ouvre une fenêtre de connexion de 30 secondes. La partie démarre à la connexion du quatrième joueur ou bien à la fin de la fenêtre de 30 secondes si moins de quatre joueurs se sont connectés.

À la fin de l'attente, tous les joueurs reçoivent la liste des joueurs participants sous la forme d'une commande **PLAYERS** suivie des noms des joueurs.

**Note:** si la commande de connexion est mal formée, le serveur envoie **ACCESSDENIED** au lieu de **WELCOME** et termine la connexion.

**Exemple de connexion:** où chaque colonne représente la communication entre un joueur et le serveur. La partie gauche représente les commandes envoyées par le client ( $\rightarrow$ ) la partie droite celle répondues par le serveur ( $\leftarrow$ ).

CONNECT/Bertrand/ $\rightarrow$	
$\leftarrow$ WELCOME/Bertrand/	
$\vdots$	
	CONNECT/René/ $\rightarrow$
	$\leftarrow$ WELCOME/René/
$\vdots$	$\vdots$
$\leftarrow$ PLAYERS/Bertrand/René/	$\leftarrow$ PLAYERS/Bertrand/René/

## 2.3 Préparatifs

Une fois les joueurs connectés, il faut gérer le positionnement des sous-marins. La grille est indiquée entre 0 et 15 horizontalement et entre A et P verticalement. Le point en bas à gauche est  $(A, 0)$  et le point en haut à droite est  $(P, 15)$ .

Le serveur va demander successivement la position de chaque sous-marin. Il demande à placer un sous-marin avec la commande **SHIP** suivie du nombre  $n$  de cases occupées par le vaisseau. Le client répond avec une commande **PUTSHIP** suivie de la suite des cases occupées sous la forme **PUTSHIP**/ $x_1/y_1/\dots/x_n/y_n/$ . Les positions doivent se suivre en ligne horizontale ou verticale, ne doivent pas dépasser de la grille, et deux sous-marins du même joueur ne peuvent occuper la même case. Si tout cela est respecté, le serveur renvoie une commande **OK**, et passe au sous-marin suivant. Sinon, il renvoie **WRONG** et renvoie la même commande **SHIP**. Une fois tous les vaisseaux placés, le serveur envoie la commande **ALLYOURBASE**.

**Exemple d'échange:** le premier placement du joueur n'est pas bon car en dehors de la grille, le serveur lui redemande donc de le placer.

	$\leftarrow$ SHIP/1/
PUTSHIP/99/Z/ $\rightarrow$	
$\leftarrow$ WRONG/	
	$\leftarrow$ SHIP/1/
PUTSHIP/6/H/ $\rightarrow$	
$\leftarrow$ OK/	
$\vdots$	
	$\leftarrow$ SHIP/2/
PUTSHIP/3/B/4/B/ $\rightarrow$	
$\leftarrow$ OK/	
$\vdots$	
$\leftarrow$ ALLYOURBASE/	

## 2.4 Déroulement

Une fois la flotte placée, le client patiente pendant que les autres joueurs terminent leurs placements en attendant une commande du serveur.

Une fois que tous les joueurs ont placé leur flotte, le serveur démarre la boucle de jeu. Les joueurs ont la main une fois chacun par tour, l'ordre étant déterminé par le temps passé à placer la flotte (le plus rapide joue en premier).

Quand c'est le tour d'un joueur, le serveur lui envoie une commande **YOURTURN**/ $x/y/a/$  où  $(x, y)$  est la position du drone et  $a$  le nombre d'actions possibles pour ce tour.

Le client répond avec une commande **ACTION** suivie des  $a$  actions, qui peuvent être U, D, L et R pour déplacer le drone d'une case (resp. en haut, en bas, à gauche et à droite) ou E pour activer le laser (limitée à une occurrence par commande **ACTION**). Le serveur effectue alors les actions une par une. Lorsqu'une commande E est rencontrée, il y a deux possibilités.

- Sous la position du drone ne se trouvaient aucun sous-marin, dans ce cas le joueur reçoit la commande **MISS**/ $x/y/$  où  $(x, y)$  est la position où le laser a été activé sans succès.
- Sous la position du drone se trouvaient un ou plusieurs sous-marins, dans ce cas le joueur reçoit la commande **TOUCHE**/ $x/y/$  où  $(x, y)$  est la position où le laser a été activé avec succès et chacun des joueurs dont un sous-marin a été touché reçoit la commande **OUCH**/ $x/y/$ .

**Note :** Si la commande est mal formée, le joueur passe son tour.

**Exemple d'échange:** au tour 1, le joueur se déplace en bas ; au tour 2 il tente d'activer le laser et réussit à toucher un concurrent ; entre ses tours 2 et 3, un concurrent touche un de ses sous-marin, il aura donc une action supplémentaire au tour suivant ; au tour 3 il se déplace à gauche, active le laser et rate.

	←	YOURTURN/8/H/1/
ACTION/D/	→	
	←	YOURTURN/8/G/1/
ACTION/E/	→	
	←	TOUCHE/8/G/
	←	OUCH/6/H/
	←	YOURTURN/8/G/2/
ACTION/L/E/	→	
	←	MISS/7/G/
	←	YOURTURN/7/G/2/
	:	

## 2.5 Fin de partie

Lorsqu'un joueur meurt (tous ses sous-marins sont complètement détruits), tous les joueurs reçoivent la commande **DEATH** suivie du nom du joueur concerné. Ce dernier ne reçoit alors plus aucune commande **YOURTURN**. Lorsqu'il ne reste qu'un survivant, il est décrété vainqueur et tout les joueurs reçoivent son nom en paramètre d'une commande **AWINNERIS**.

Le serveur attend alors la réponse de chaque joueur qui peut être **BYE** auquel cas la connexion avec joueur est terminée ou **PLAYAGAIN** auquel cas le joueur est directement inscrit pour la prochaine partie.

Une fois que tous les joueurs ont répondu, le serveur relance une phase de connexion pour accepter d'éventuels nouveaux arrivants. Vous avez le droit d'ajouter un temps limite de réponse, par exemple de 30 secondes.

**Note :** penser à prendre en charge correctement une déconnexion brutale.

## 3 Extensions

Les extensions sont à réaliser dans l'ordre. Il ne s'agit pas d'exercices bonus. Implanter au moins la première est nécessaire pour obtenir une bonne note.

### 3.1 Discussion instantanée

Votre client doit afficher une petite zone de discussion instantanée façon IRC. Lorsque le joueur a écrit son message et l'envoi, le client envoie une commande **TALK** au serveur suivie du message. Le serveur transmet alors à tous les clients la commande **HEYLISTEN** suivie du nom du joueur puis du message. Chaque client doit alors afficher le message dans sa console de discussion.

**Note 1 :** le client peut parler dès l'obtention de son nom par la commande **WELCOME** et à n'importe quel moment de la partie. Vous pouvez si vous le souhaitez établir une limite de fréquence au dessus de laquelle les messages ne seront plus relayés et le joueur recevra un message d'avertissement en retour.

**Note 2 :** le serveur peut envoyer lui-même des messages (par exemple en utilisant le nom **"(broadcast)"**). Cela peut être utile pour ajouter des décomptes ou des commentaires.

### 3.2 Comptes utilisateurs

En plus du mode anonyme, on veut permettre à un joueur de réserver son nom, en protégeant son utilisation par un mot de passe.

Pour ceci, on autorise deux commandes de connexion supplémentaires :

- **REGISTER/n/p/** se connecte sous le nom *n* et réserve ce nom pour de futures connexions, en le protégeant par le mot de passe *p*. Si l'utilisateur existe déjà dans la base, le client reçoit une commande **ACCESSDENIED** et la connexion est terminée.
- **LOGIN/n/p/** se connecte sous le nom *n*, qui doit être enregistré dans la base d'utilisateurs et dont le mot de passe doit être *p*. Si l'utilisateur n'existe pas ou si le mot de passe ne correspond pas, le client reçoit une commande **ACCESSDENIED** et la connexion est terminée.

**Note 1 :** le choix d'implantation de la base d'utilisateurs est laissée libre (fichier texte, base SQLite, donnée sérialisée, etc.)

**Note 2 :** en option, vous pouvez établir un protocole plus sûr que la transmission du mot de passe en clair.

### 3.3 Mode spectateur

On rajoute une commande de connexion **SPECTATOR**. Un spectateur n'intervient pas dans la partie mais peut voir toutes les actions de tous les joueurs. Le serveur utilise pour cela le jeu de commandes suivantes :

1. La commande **PLAYERS** comme pour les joueurs, indiquant le début de partie et donnant la liste des joueurs.
2. **PLAYERSHIP**/ $n/x_1/y_1/\dots/x_n/y_n/$  où  $n$  est le nom du joueur et le reste est identique à la commande **PUTSHIP**.
3. **PLAYERMOVE**/ $n/x/y/$  pour indiquer que le joueur  $n$  s'est déplacé en  $(x, y)$ .
4. **PLAYEROUCH**/ $n/x/y/$  pour indiquer qu'un sous-marin du joueur  $n$  a été touché en  $(x, y)$ .
5. Les commandes **DEATH** et **AWINNERIS** comme pour les joueurs.

Si un spectateur arrive en cours de partie, le serveur doit lui renvoyer à ce moment toutes les commandes qu'ils aurait reçues s'il était arrivé au tout début afin de rattraper son retard.

Les spectateurs ont le droit de s'exprimer dans la discussion avec la commande **TALK**, le nom qui apparaît est simplement "(spectateur)". Ils reçoivent aussi les commandes **HEYLISTEN**.

### 3.4 Serveur HTTP de statistiques

On veut, en accédant au port 2092 de la machine où est lancé le serveur depuis un navigateur Web, obtenir une page HTML indiquant pour chaque joueur inscrit dans la base de données son nombre de victoires et son nombre de défaites.

**Indication :** pour vous éviter la lecture de la spécification du protocole HTTP, interrogez quelques sites Web directement avec la commande telnet pour vous donner une première idée de ce que doit renvoyer votre serveur.

## 4 Pour aller plus loin

Si et seulement si vous avez terminé ce qui précède, vous pouvez étendre le jeu à votre gré. Voici quelques suggestions.

- Ajouter des éléments au jeu : sonars, bonus, réparations, missiles entre drones, etc.
- Améliorer l'infrastructure du jeu : protocole binaire, serveur gérant des parties en parallèle, etc.
- Implanter un client et/ou spectateur Web/HTTP.
- Peaufiner l'aspect visuel/sonore du client.

Pour chacune de vos extensions, détaillez le principe et l'implantation dans le rapport.