

Celine Bonnet

Imane Bouka

Zoubir Lanseur

11 novembre 2014

Dossier de conception d'un jeu en réseau

Le « MORPION »

Sommaire

Expression du besoin	2
Présentation du produit	2
Règlement du jeu	2
Exigences fonctionnelles	3
Exigences Techniques	4
Architecture du système	5
Architecture logicielle	6
Critères pour la qualité	7
Manuel d'utilisation	8

Expression du besoin

Dans le cadre du module de génie logiciel, il est recommandé de réaliser un jeu en réseau mettant en application les compétences acquises durant le cours et utilisant les technologies « Client-Serveur ».

Afin de répondre à ce besoin, nous avons décidé de concevoir un « morpion » en réseau. A cela s'ajoute une petite customisation supplémentaire : rajout d'une fonctionnalité supplémentaire de type chat durant une partie du jeu.

Le dossier de conception suivant détaille toutes les exigences fonctionnelles et techniques et précise les différentes solutions retenues dans le cadre de la réponse au besoin.

Présentation du produit

Le « **morpion** » est un jeu de réflexion se pratiquant à deux joueurs au tour par tour et dont le but est de créer le premier un alignement sur une grille.

Règlement du jeu

Les joueurs inscrivent tour à tour leur symbole (ou leurs couleurs dans notre cas) sur une grille qui n'a pas de limites ou qui n'a que celles de l'écran sur lequel on joue. Dans notre cas, le premier qui parvient à aligner trois de ses symboles horizontalement, verticalement ou en diagonale gagne la partie.

En général, le « **morpion** » donne un avantage assez important à celui qui commence. Des formes évoluées existent, comme le ***Gomoku*** ou le ***Pente***, qui ajoutent à la notion d'alignement une notion de prise. Une partie dure environ 1 minute.

Exigences fonctionnelles

Exigences Fonctionnelles	
Besoins	Exigences
Le joueur est capable de commencer une partie	Le jeu est capable de tourner
	Le jeu est capable de détecter les joueurs
Le joueur est capable de reconnaître son tour	Le jeu attribue les tours aux joueurs
	Le jeu est capable de reconnaître les tours des joueurs (les priorités)
Le joueur est capable de jouer	Le joueur peut jouer lors de son tour
	Le jeu permet aux joueurs de voir leurs actions dans une partie
Le joueur est capable de communiquer avec son adversaire	Le jeu permet la communication entre les joueurs
	Le jeu met à disposition des joueurs une plateforme de chat
	Le jeu est capable de reconnaître l'émetteur et le destinataire
Le joueur est capable de rejouer une partie	Le jeu offre une option rejouer à la fin de chaque partie
Le joueur est capable de quitter la partie	Le jeu permet aux joueurs de se déconnecter

Exigences de performances	
Besoins	Exigences
Le maximum de joueurs est 2	Le jeu permet la connexion de deux joueurs en même temps et pas plus

Exigences Techniques

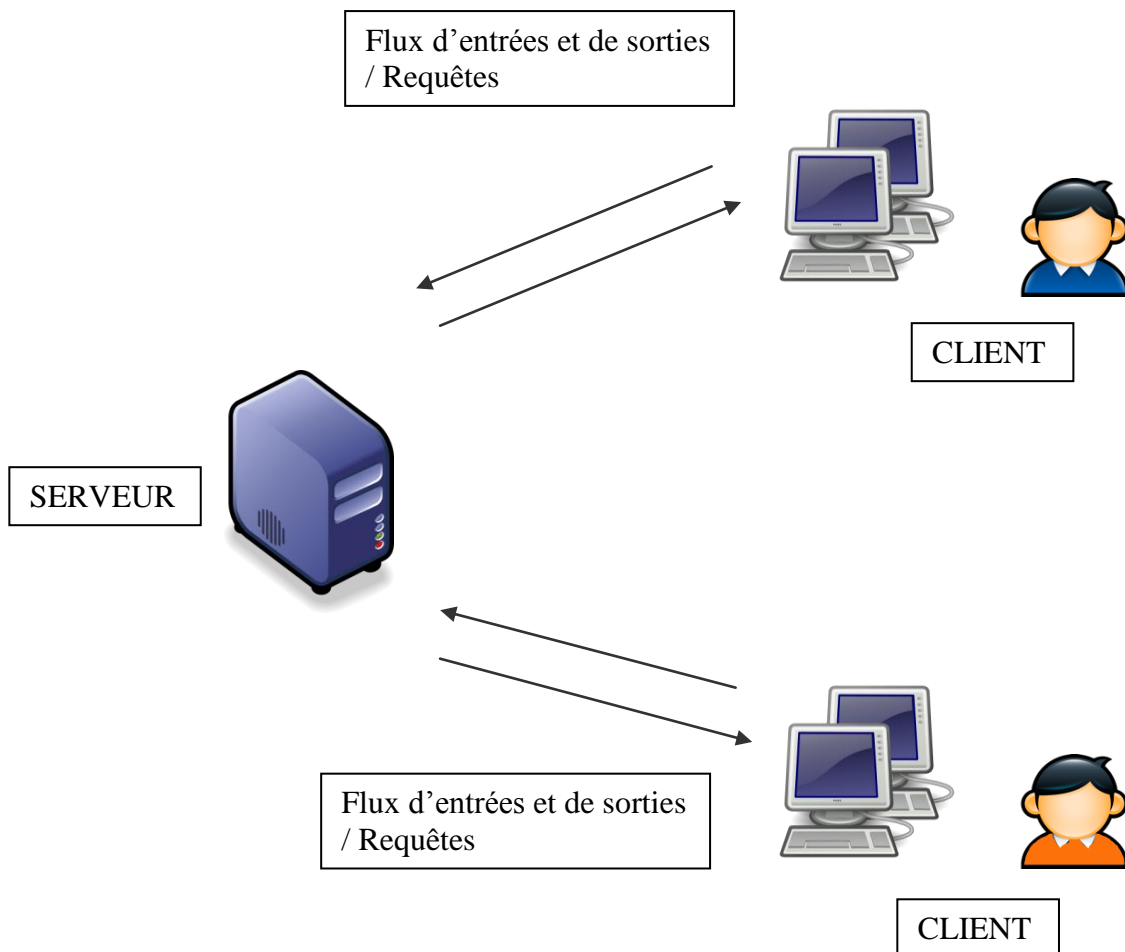
Dans cette partie nous allons essayer de transformer les exigences fonctionnelles en exigences techniques.

Exigences techniques	
Besoins	Exigences
Commencer une partie	Lancer le serveur
	Connecter les clients (2 maximum) (le bouton « Connect »)
Attribuer l'ordre de jeu	Le serveur donne la priorité au premier client connecté
	Seul le joueur à qui le tour est attribué peut jouer
Assurer la communication	La mise à disposition d'une interface de chat
	Le jeu utilise des sockets assurant le transfert des messages
	L'interface contient un bouton « Send » qui permet d'envoyer des messages
	Distinction des émetteurs/destinataires en mettant devant chaque message la désignation du joueur qu'il l'a envoyé
Rejouer une partie	L'interface contient un bouton « Rejouer » qui permet de relancer une nouvelle partie
Quitter une partie	L'interface contient un bouton « Disonnect » qui permet de finir une partie et de se déconnecter complètement du serveur de jeu.

Architecture du système

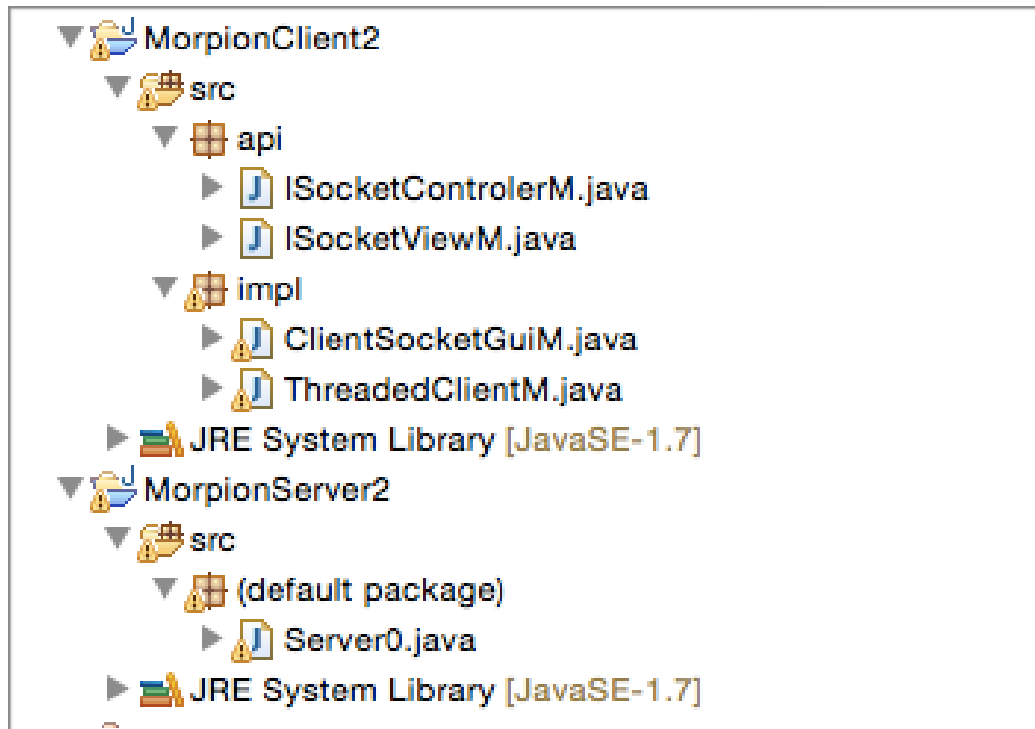
L'application Morpion a une structure réseau très simple. Cette dernière repose principalement sur l'illustration du modèle client/serveur de base. En effet, les clients s'échangent des flux de données (entrées, sorties) via des requêtes adressées au serveur de l'application.

La structure réseau de l'application Morpion



Architecture logicielle

L'application Morpion est structurée de la sorte:



Elle est constituée de deux projets principaux, un représente le côté serveur et l'autre le côté client.

L'architecture logicielle répond à l'architecture Modèle – Vue – Contrôleur.

Cependant dans notre application la classe modèle est en quelque sorte remplacée par la classe serveur car c'est cette dernière qui contient les informations relatives au jeu. Les classes Vue et Contrôleur sont elles du côté Client (ce sont les classes **ClientSocketGuiM** et **ThreadedClientM**).

Critères pour la qualité

Deux approches sont possibles pour la conception d'applications en mode client-serveur, soit :

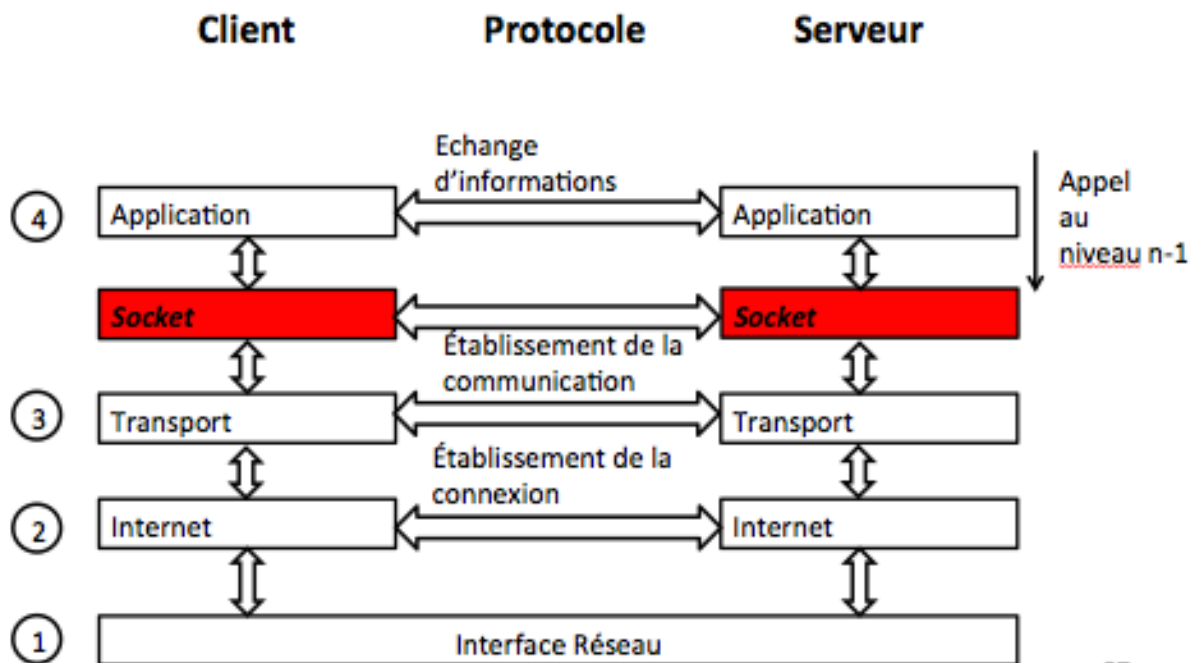
Orientation protocole applicatif: on définit un protocole (mots-clés, séquences) spécifique portant la sémantique applicative. (arch. Unix: SMTP, FTP, etc..)

Orientation application: on conçoit une application classique, et on la sépare en modules qui communiquent avec des appels de procédures à distance. (RPC, Corba, RMI)

Dans notre cas, nous avons choisi la première possibilité « protocole applicatif ». Donc le client et le serveur applicatif implémenteront des protocoles pour générer et analyser les données échangées, en conformité avec la sémantique applicative. Et Les protocoles s'empilent par dessus les protocoles internes de TCP/IP.

La structure en couches est la suivante :

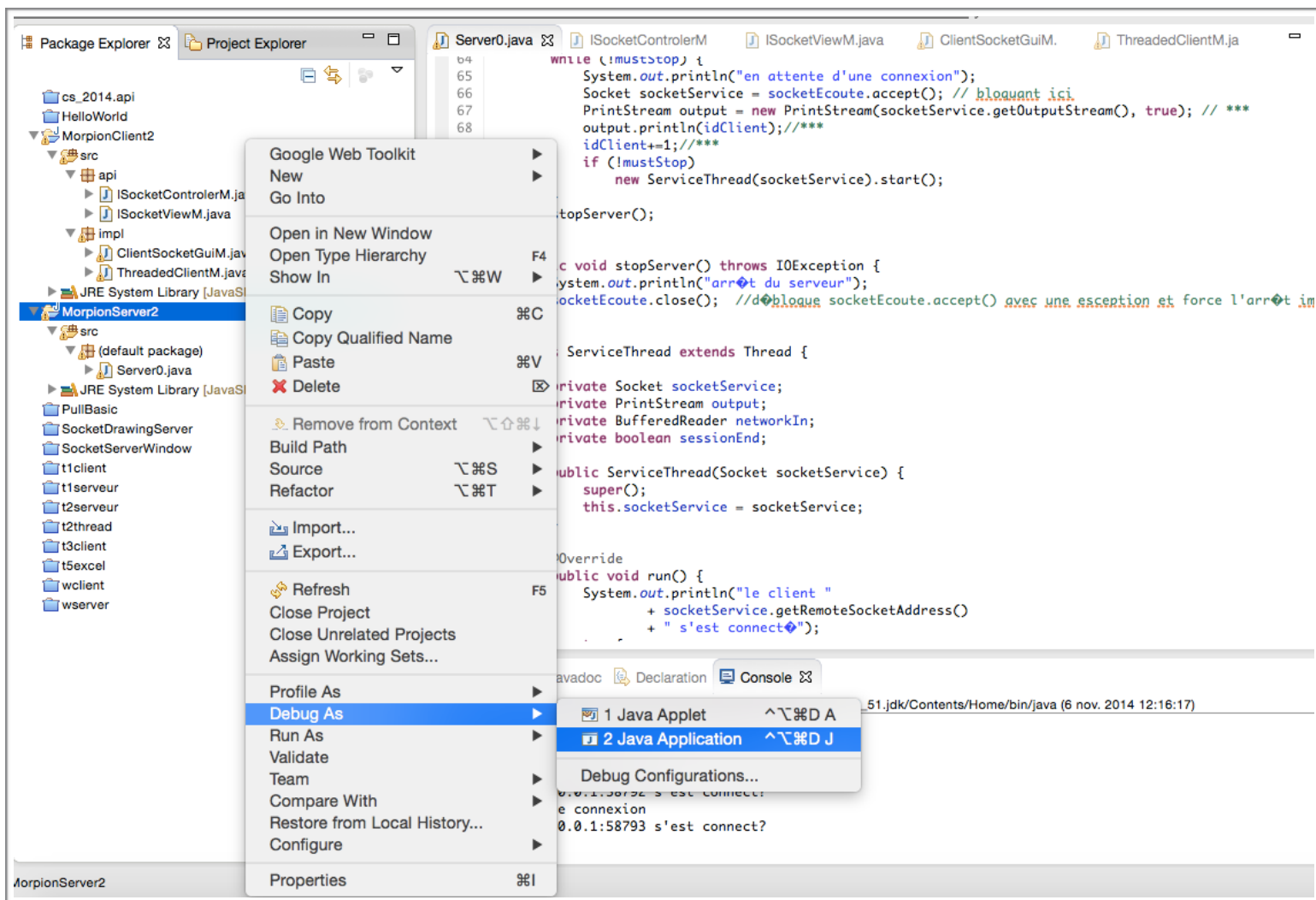
Structuration en couches



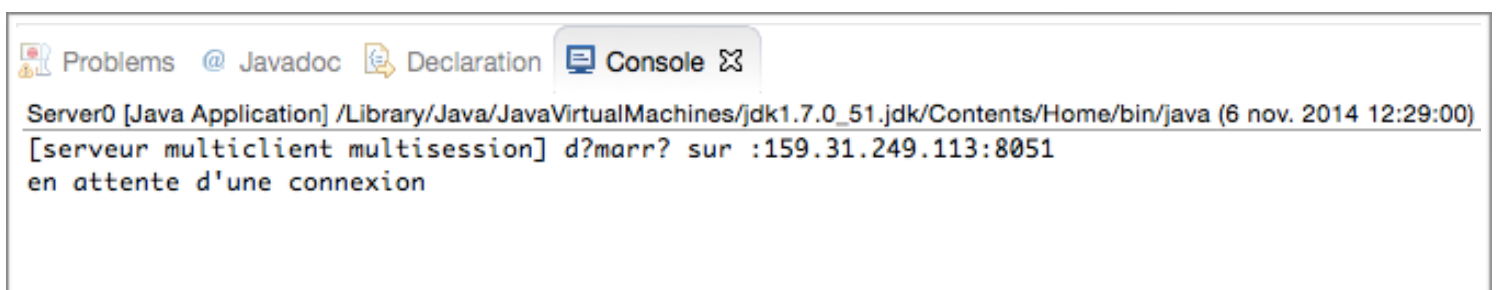
Manuel d'utilisation

Pour lancer l'application « Morpion » il faut suivre quelques étapes basiques et très simples:

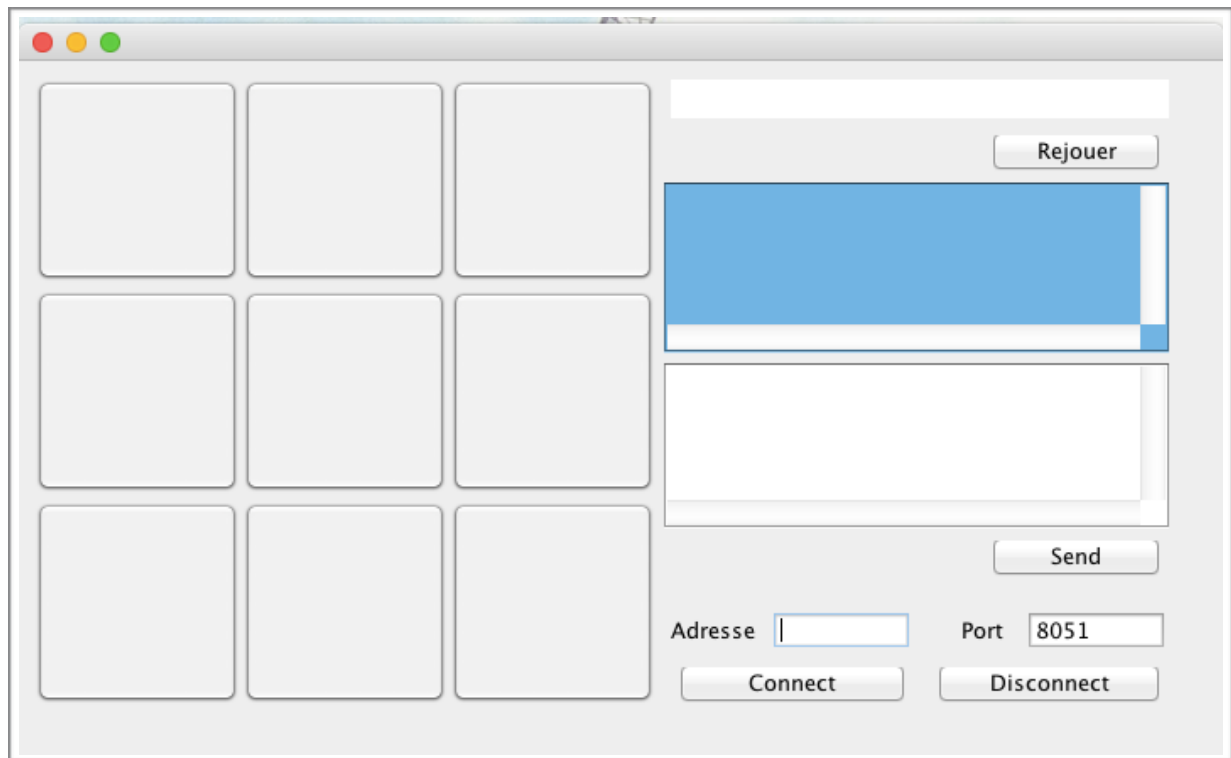
1. Lancer le serveur (faire tourner le projet « MorpionServer2 » ou tout simplement la classe « Server0 »)



Le serveur se met alors en attente des connexions des joueurs.

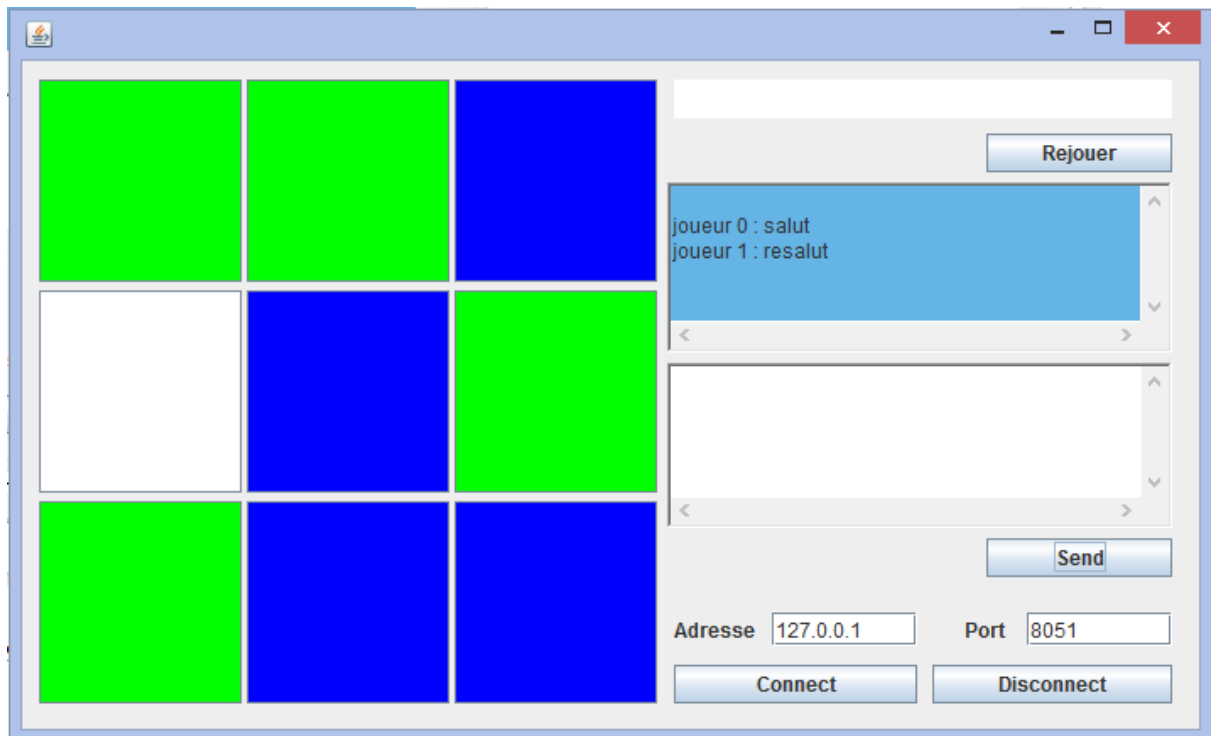


2. Lancer deux fois le client (faire tourner le projet « MorpionClient2 » deux fois) afin d'obtenir deux joueurs si on veut jouer sur la même machine et une seule fois si on joue sur deux machines différentes. On obtient alors une interface client de type:



Le client rentre ainsi l'adresse IP du serveur pour se connecter. Le port du serveur est le port 8051. Pour changer ce dernier il ne faut pas oublier de le changer dans le serveur.

3. Une fois les deux joueurs connectés ils peuvent commencer la partie en cliquant chacun à son tour sur les cases dans le but de réaliser un alignement horizontal, vertical ou diagonal de trois cases.



4. A la fin d'une partie, il y a un seul gagnant et les joueurs peuvent décider de rejouer encore en cliquant sur le bouton « Rejouer » ou bien de finir et quitter la partie en cliquant sur le bouton « Disconnect ».

5. Pour utiliser le chat il suffit d'écrire dans le champ en blanc et d'envoyer le message en cliquant sur le bouton « Send ». Le message est alors transmis aux deux clients sur le champ en bleu.

Note : Le morpion fonctionne bien en réseau. Nous l'avons testé sur plusieurs machines en réseau et l'application fonctionne bien. Cependant il peut arriver que le projet ne fonctionne pas à cause du pare-feu ou d'autres raisons qui nous échappent.