

DOCUMENTATION TECHNIQUE ET MANUEL DE JEU **KATARENGA & Co**

Kylian Lyenard  
Vincent Keller  
Johan Dumont-Basso

Table des matières

[1. Documentation technique 4](#_Toc199708037)

[1.1 Justification des choix techniques 4](#_Toc199708038)

[Choix Langage. 4](#_Toc199708039)

[Choix bibliothèque graphique. 4](#_Toc199708040)

[1.2 Structures de données 4](#_Toc199708041)

[Comment sont représentés dans le code : 4](#_Toc199708042)

[Général 4](#_Toc199708043)

[Les quadrants (double face, orientation…) 5](#_Toc199708044)

[Le plateau (4 emplacements configurables) 5](#_Toc199708045)

[Les pions (couleur, position, état…) 5](#_Toc199708046)

[Les camps 5](#_Toc199708047)

[Les joueurs 5](#_Toc199708048)

[1.3 Composants graphiques 5](#_Toc199708049)

[Comment sont dessinés les éléments 5](#_Toc199708050)

[Gestion des clics et menus 5](#_Toc199708051)

[Transitions entre les écrans 6](#_Toc199708052)

[Utilisation des images, sons… 6](#_Toc199708053)

[1.4 Algorithmes de déplacement 6](#_Toc199708054)

[Déplacement selon la couleur : 6](#_Toc199708055)

[1.5 Conditions de victoire 6](#_Toc199708056)

[Katarenga : occupation des deux camps ou élimination adverse. 6](#_Toc199708057)

[Congress : formation d’un bloc connecté. 6](#_Toc199708058)

[Isolation : dernier à pouvoir poser un pion. 6](#_Toc199708059)

[1.6 Communication réseau 7](#_Toc199708060)

[2. Manuel du jeu 7](#_Toc199708061)

[2.1 Prérequis / Installation 7](#_Toc199708062)

[2.2 Menu principal : 7](#_Toc199708063)

[2.3 Règles des jeux 8](#_Toc199708064)

[Règles communes aux 3 jeux : 8](#_Toc199708065)

[*Katarenga :* 8](#_Toc199708066)

[2.4 Utilisation de l'interface 10](#_Toc199708067)

# 1. Documentation technique

## 1.1 Justification des choix techniques

### Choix Langage.

Le projet est intégralement développé en C, ce qui permet :

- Un contrôle précis de la mémoire,

- Une grande portabilité (Linux, Windows, Mac),

- Des performances optimales pour le calcul et la gestion graphique.

### Choix bibliothèque graphique.

On a utilisé la bibliothèque SDL2 :

- Permet un rendu graphique multiplateforme (Windows, Linux, Mac),

- Très maniable.

- Gère aussi bien les graphismes que les entrées utilisateurs,

- Supporte l’affichage, les sons et le texte via les extensions SDL\_ttf, SDL\_image, SDL\_mixer.

- Expérience antérieure.

## 1.2 Structures de données

## Comment sont représentés dans le code :

Général

Pour les structures du Network ou les matrices, les pointeurs ont été favorisé car ils permettent de prendre moins d’espace disque lorsque passé en commentaire. Les parties n’ayant aucun rapport entre eux sont séparés dans des fonctions différentes, afin de supprimer les données inutilisées par une autre partie.

Les quadrants (double face, orientation…)

Les quadrants sont représentés implicitement à travers des matrices de taille 8x8 dans le code. Chaque case peut contenir une valeur indiquant la nature de la case (voir `board[x][y]`). L’orientation et les faces sont gérées par la valeur des cases et les fonctions d’affichage.

### Le plateau (4 emplacements configurables)

Le plateau est un tableau 2D (`int board[8][8]`). Chaque case peut changer de nature selon la configuration (quadrant, orientation, etc.).

### Les pions (couleur, position, état…)

Les pions sont gérés dans une matrice `pawns[8][8]` où chaque valeur indique :

- 0 : pas de pion

- 1 : pion joueur 1 (exemple : Bleu)

- 2 : pion joueur 2 (exemple : Rouge)

- >=3 : case spéciale ou état particulier (ex : prévisualisation)

### Les camps

Les camps sont des zones particulières du plateau, repérées par la valeur des cases dans la matrice du plateau.

### Les joueurs

Les joueurs sont des structures pour pouvoir y stocker le nombre de pions entrés dans un camp dans le jeu katarenga. Ils sont dans un array players[2] pour connaître le joueur actuel selon le tour.

## 1.3 Composants graphiques

### Comment sont dessinés les éléments

- Le plateau et les pions sont dessinés avec SDL2 via des rectangles de couleur.

- Les couleurs sont attribuées selon la valeur du pion ou de la case (fonction `getCellTheme`, `drawBoard`).

### Gestion des clics et menus

- La fonction `gridClick(int mouseX, int mouseY, int sWE)` convertit la position de souris en coordonnées du plateau.

- Les menus sont affichés avec SDL2 et gérés via des événements SDL.

### Transitions entre les écrans

- Gérées dans la boucle principale (`main.c`), via l’appel aux fonctions de menu et de jeu.

### Utilisation des images, sons…

- SDL\_image, SDL\_mixer, SDL\_ttf sont utilisés pour les images, sons et textes (voir initialisation dans `main.c`).

## 1.4 Algorithmes de déplacement

### Déplacement selon la couleur :

Déplacement selon la couleur :

- Bleu → roi

- Vert → cavalier

- Jaune → fou (1ère jaune rencontrée)

- Rouge → tour (1ère rouge rencontrée)

## 1.5 Conditions de victoire

### Katarenga : occupation des deux camps ou élimination adverse.

Pour gagner à Katarenga, il faut occuper les deux camps adverses ou éliminer tous les pions adverses. Le programme vérifie après chaque déplacement si un joueur remplit   
l’une de ces conditions en analysant la matrice du plateau.

### Congress : formation d’un bloc connecté.

Dans ce mode, le but est de former un bloc de pions connectés. Le code parcourt les lignes et colonnes pour vérifier si un joueur a réussi à aligner tous ses pions, ce qui signifie une victoire.

### Isolation : dernier à pouvoir poser un pion.

Le gagnant est le dernier joueur à avoir pu poser ou déplacer un pion. Après chaque coup, le programme regarde si l’autre joueur a encore des coups possibles : si ce n’est pas   
le cas, la partie est terminée.

## 1.6 Communication réseau

- Utilisation de TCP (`SOCK\_STREAM`) pour garantir la fiabilité des échanges.

- Architecture client/serveur.

- Un joueur crée un serveur, l’autre rejoint via l’IP.

- Protocole textuel simple : codes de commandes en début de message (`m` pour move, `c` pour camp, `n` pour name, etc.).

- Exemples de commandes :

- `m4725` : déplacement d’un pion de 47 à 25. (Au lieu d’avoir deux paramètres pour un pion, sa coordonnée x est la dizaine de ce nombre et l’unité est son axe vertical : 47 représentés le pion positionné en x = 4 et y = 7.

- `c02` : camp pour le pion 2.

- `nName` : envoi du nom.

# 2. Manuel du jeu

## 2.1 Prérequis / Installation

Lancement du jeu via le l’exécutable mis à disposition.

## 2.2 Menu principal :

Dans le menu principal vous aurez plusieurs options :

Local : Permet de créer une partie en localement avec un ami ou un ordinateur dont on pourra changer la difficulté

Online : Permet de faire une partie en ligne e

Charger une partie : Pour reprendre une partie qui aurai été arrêté prématurément

Quitter le Menu : Permet de quitter le Jeu

## 2.3 Règles des jeux

Règles communes aux 3 jeux :   
Les jeux se jouent tous en un contre un.

Déplacements :   
  
Un pion sur une case rouge se déplace comme une tour, sur une case jaune comme un fou, sur une case verte comme un cavalier, et sur une case bleue comme un roi. Il peut se déplacer jusqu'à n'importe quelle distance sans toutefois pouvoir dépasser la prochaine case de la couleur à partir de laquelle il a commencé dans le cas de la tour ou du fou.

#### Katarenga :

Initialisation du plateau :

On placera ici 8 pions d’une couleur sur une ligne et 8 autres pions sur la ligne du cadrant opposer à celle-ci.

Objectif :

Atteindre les deux camps adverses qui sont accessibles si un pion est sur la ligne opposé à celle de sa ligne de départ, le joueur propriétaire du pion peut alors utiliser un mouvement pour atteindre le dit camp en utilisant un tour de jeu, une autre condition de victoire s’apparente à réduire le nombre de pions de l’adversaire jusqu’à ce qu’il ne puisse plus réussir l’objectif énoncé précédemment.

On notera que la capture est interdite au premier tour.

#### Congres

Initialisation du plateau on positionnera les pions comme sur le schéma suivant :

A colorful game with black and white balls

AI-generated content may be incorrect.

Objectif :

A tour de rôle les joueurs pourront déplacer les pions comme indiqué dans la partie déplacement des pièces.  
Le but ici est de créer un bloc de pions de notre couleur en faisant en sorte qu’il y en a tout le temps un coller verticalement ou horizontalement à un autre de sa couleur.   
Pour cela on interdira toute capture

#### Isolation

Initialisation du plateau :   
Le plateau restera vide.

À tour de rôle chaque joueur pose un de ses pions sur une case vide de telle sorte que celui-ci ne soit pas en prise (*i.e.* ne puisse être potentiellement capturé) par un des pions déjà présents sur le plateau (de l'une ou l'autre couleur).

Il n'y a donc ni déplacements ni captures dans ce jeu.

Le vainqueur est le dernier joueur à pouvoir poser un pion.

## 

## 2.4 Utilisation de l'interface

Création d’une partie en ligne :

Pour crée une partie en ligne on met son nom puis on une adresse IP que vous donnez à l’autre personne avec laquelle vous voulez créer votre partie, et pour rejoindre une partie on va mettre l’IP dans la zone dédiée.