

# **CAHIER DES CHARGES : PROJET JEU**

Mercredi 5 février 2014

**Groupe A**

L3 info SPI

# Sommaire

<b>1</b>	<b>Présentation du projet</b>	<b>1</b>
1.1	Contexte . . . . .	1
1.2	Objectifs . . . . .	1
1.3	Utilisateur cible . . . . .	1
1.4	État de l'art . . . . .	1
<b>2</b>	<b>Contraintes initiales</b>	<b>1</b>
2.1	Contraintes de jeu . . . . .	1
2.2	Contraintes de conception . . . . .	2
2.3	Contraintes temporelles . . . . .	2
2.4	Contraintes matérielles . . . . .	2
<b>3</b>	<b>Spécification du projet</b>	<b>2</b>
3.1	Règles détaillées . . . . .	2
3.1.1	Éléments constituant le jeu . . . . .	2
3.1.2	Principe de fonctionnement du jeu . . . . .	2
3.2	Besoins fonctionnels . . . . .	3
3.3	Besoins non fonctionnels (optionnels) . . . . .	4
3.4	Explications additionnelles . . . . .	4
3.5	Acteur . . . . .	5
<b>4</b>	<b>Déroulement du projet</b>	<b>6</b>
4.1	Livrables . . . . .	6
4.2	Planning . . . . .	6
4.3	Équipe . . . . .	6
4.4	Outils de développement . . . . .	6
4.5	Diagramme WBS . . . . .	7
4.6	Diagramme de Gantt . . . . .	8
<b>5</b>	<b>Glossaire</b>	<b>8</b>

# 1 Présentation du projet

## 1.1 Contexte

Lors du second semestre de troisième année de licence SPI, il est demandé de réaliser un projet. Ce projet à thème imposé n'est autre que le développement d'un logiciel complexe demandant un travail en équipe.

Notre équipe de cinq étudiants aura donc du 24 janvier au 16 Mai 2014 pour mettre en œuvre les préceptes de Génie Logiciel vus au premier semestre afin de présenter un logiciel potentiellement commercialisable par le client, ici représenté par les enseignants.

## 1.2 Objectifs

L'objectif de ce projet est la conception d'une application permettant la création et l'aide à la résolution de puzzles de type picross (henjie).

Le joueur pourra commencer des puzzles de différentes tailles ; le but étant de terminer le plus vite possible la partie.

## 1.3 Utilisateur cible

Ce logiciel n'est destiné qu'à un seul joueur à la fois. Cependant, il permet à de nombreux joueurs d'apposer leur scores à la table des scores.

## 1.4 État de l'art

Les jeux déjà disponibles permettent différentes fonctionnalités :

- La création aléatoire et l'édition de grilles de picross de tailles diverses,
- La création d'une grille à partir d'une image,
- La résolution du puzzle avec ou sans aide à la résolution,
- La sauvegarde et le chargement d'une partie,
- L'enregistrement et la consultation des scores,
- Le partage de ses grilles déjà jouées,
- Cocher une case en cliquant dessus une fois, ce qui changera sa couleur,
- Marquer une case comme vide en cliquant dessus deux fois et en apposant une croix.

# 2 Contraintes initiales

## 2.1 Contraintes de jeu

- Le jeu est un picross, le joueur doit noircir les cases en fonction des informations données par les chiffres en tête de ligne et de colonne,

- Le jeu doit proposer plusieurs tailles de grille (5x5 10x10 15x15 20x20),
- L'application doit proposer un éditeur de grille.

## 2.2 Contraintes de conception

- Le langage de programmation est Ruby/GTK.
- Les paramètres de l'application doivent être externalisés afin de pouvoir être modifiés par le joueur (taille de la grille).
- Le logiciel doit être développé en Programmation Orientée Objet (POO).
- Le joueur doit pouvoir sauvegarder une partie pour la reprendre ultérieurement.

## 2.3 Contraintes temporelles

- Le client impose une réunion hebdomadaire entre les différents acteurs de notre groupe,
- Le projet final doit être présenté au client le 16 mai 2014.

## 2.4 Contraintes matérielles

- Le présent Cahier des Charges doit être remis au client le 16 mai 2014.
- Au terme de chaque réunion un compte rendu doit être remis au client.

# 3 Spécification du projet

## 3.1 Règles détaillées

### 3.1.1 Éléments constituant le jeu

**Grille** : La grille est constituée d'un carré dont le coté est un multiple de 5 cases. Il y a ainsi cinq tailles de grilles différentes, 5x5, 10x10, 15x15, et 20x20.

**Case** : Les cases peuvent être noircies, en cliquant dessus à l'aide de la souris. Il est possible de décocher la case en cliquant dessus à nouveau. Il est également possible de cliquer sur une case à l'aide de la souris pour indiquer que la case restera vide, qu'elle ne sera pas noircie.

**Indications extérieures** : Les indications extérieures sont des suites de chiffres sur les bords gauche et supérieur, indiquant le nombre de case à noircir, respectivement dans les lignes et dans les colonnes.

**Chronomètre** : le chronomètre indique le temps qui s'est passé depuis que le joueur a noirci la première case.

### 3.1.2 Principe de fonctionnement du jeu

**But du jeu** Le but du jeu est de noircir toutes les cases devant être noircies dans le plus court laps de temps possible.

**Début d'une partie** Au début d'une partie, la grille est vierge, et le chronomètre est à 00:00.

**Deroulement du jeu.** Le jeu commence lorsque le joueur noirci un case en cliquant dessus, ce qui enclenche le chronomètre.

**Le menu en cours de partie** En cours de partie, le joueur peut accéder à un menu spécial où se trouvent :

**Sauvegarde de la partie :** L'utilisateur peut sauvegarder sa partie en cours. Le joueur peut ensuite continuer à jouer ou quitter le jeu. ;

**Quitter la partie :** Renvoie le joueur au menu principal. Il devra avoir pris le soin de sauvegarder ou non sa partie. ;

**L'interface joueur** Le joueur noirci des cases ou les rend de nouveau blanches en cliquant dessus à l'aide de la souris. Il peut également cliquer sur un bouton **Aide** ou sur le bouton **Menu**.

### 3.2 Besoins fonctionnels

FONCTIONNALITE	REFERENCE
Nouvelle partie	F100
Gérer ses parties	F110
Sauvegarder une partie	F111
Charger une partie	F113
Supprimer une partie sauvegardée	F114
Quitter le jeu	F120
Choisir la taille de la grille	F130
Fournir une aide	F140
Créer des grilles de jeu	F150
Afficher le score	F160
Afficher les informations de la grille	F170
Afficher le menu du jeu	F180

- F100 : Lancer une nouvelle partie
- F110 : Le jouer a la possibilité de gerer ses parties à l'aide des fonctionnalités suivante
- F111 : Sauvegarder une partie à n'importe qu'elle moment
- F112 : Charger une partie à n'importe qu'elle moment
- F114 : Pouvoir supprimer une partie sauvegardée précédement
- F120 : Quitter le jeu de n'importe qu'elle endroit
- F130 : Choisir la taille de la grille lors d'une nouvelle partie ou de la création de grille
- F140 : Mise à disposition d'une aide pour le joueur, de différente degré

- F150 : Possibilité de création de grille par le joueur
- F160 : Afficher le score du joueur en cours de jeu ou sur le meuble principale
- F170 : Le joueur a la possibilité de consulter les informations de la grille qui sont disponibles
- F180 : Afficher le menu de base du jeu

### 3.3 Besoins non fonctionnels (optionnels)

FONCTIONNALITE	REFERENCE
Choisir la langue	F200
Affichage de statistiques/classement	F210
Mise en place d'un fond sonore	F220
Choix d'un mode de jeu	F230
Création de grille via une image	F240

- F200 : Le joueur pourra avoir la possibilité de choisir la langue du jeu parmi
- F210 : On pourra consulter les statistiques et classement disponible pour les joueurs ou les grilles proposées
- F220 : Mise en place d'un environnement sonore avec possibilité de le désactiver
- F230 : Le joueur aura le choix d'un mode de jeu lui proposant un enchaînement de partie ainsi qu'une gestion du score différents
- F240 : Le joueur aura la possibilité de créer des grilles à partir d'image qu'il importera

### 3.4 Explications additionnelles

Le joueur a le choix parmi différents modes de jeu.

- Mode Normal: Finir la partie en faisant le meilleur score possible.
- Mode Aléatoire: La taille et la difficulté de la grille est choisie aléatoirement.
- Mode Contre-la-montre: Finir la partie dans un temps imparti.
- Mode Aventure: Les parties s'enchaînent avec une difficulté croissante.

### 3.5 Acteur

La figure 1 représente le diagramme du cas d'utilisation avec les différentes fonctionnalités listées ci-dessus.

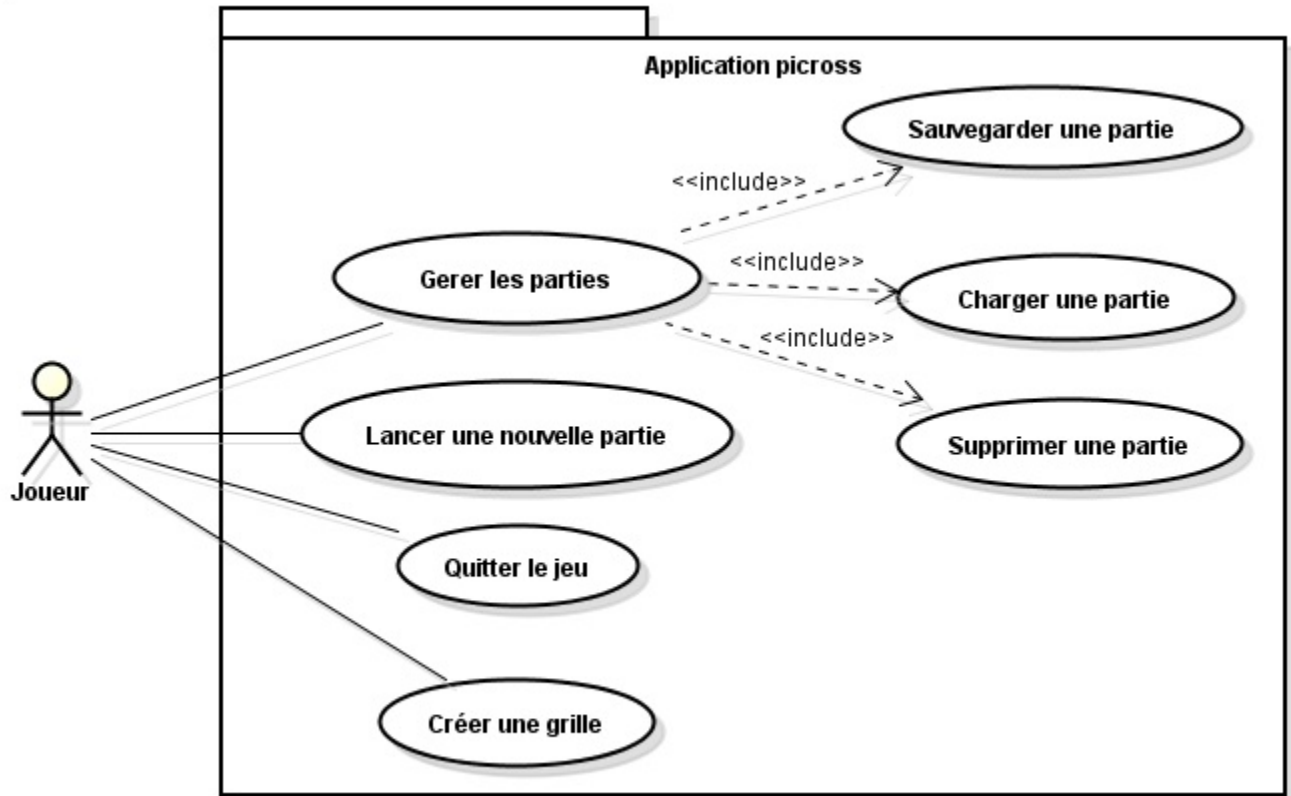


Figure 1: Diagramme du cas d'utilisation

## 4 Déroutement du projet

### 4.1 Livrables

Les livrables prévus sont :

- Le présent cahier des charges validé par le client ;
- Dossier de conception ;
- Manuel utilisateur ;
- Le jeu fonctionnel (livraison le jeudi 16 mai 2014).

### 4.2 Planning

Afin de mener à bien ce projet, il est mis à disposition des étudiants 16 séances de 3h (en plus du temps libre) pour permettre aux membres de l'équipe de se retrouver et au chef d'équipe de distribuer les tâches en respectant au mieux le planning mis en place au début du projet.

### 4.3 Équipe

Les membres de l'équipe sont Rémi TREMBLAIN (chef d'équipe), Erwan MARCAHND, Colas PICARD, Kévin CROILLERE et Anice KHOMANY.

### 4.4 Outils de développement

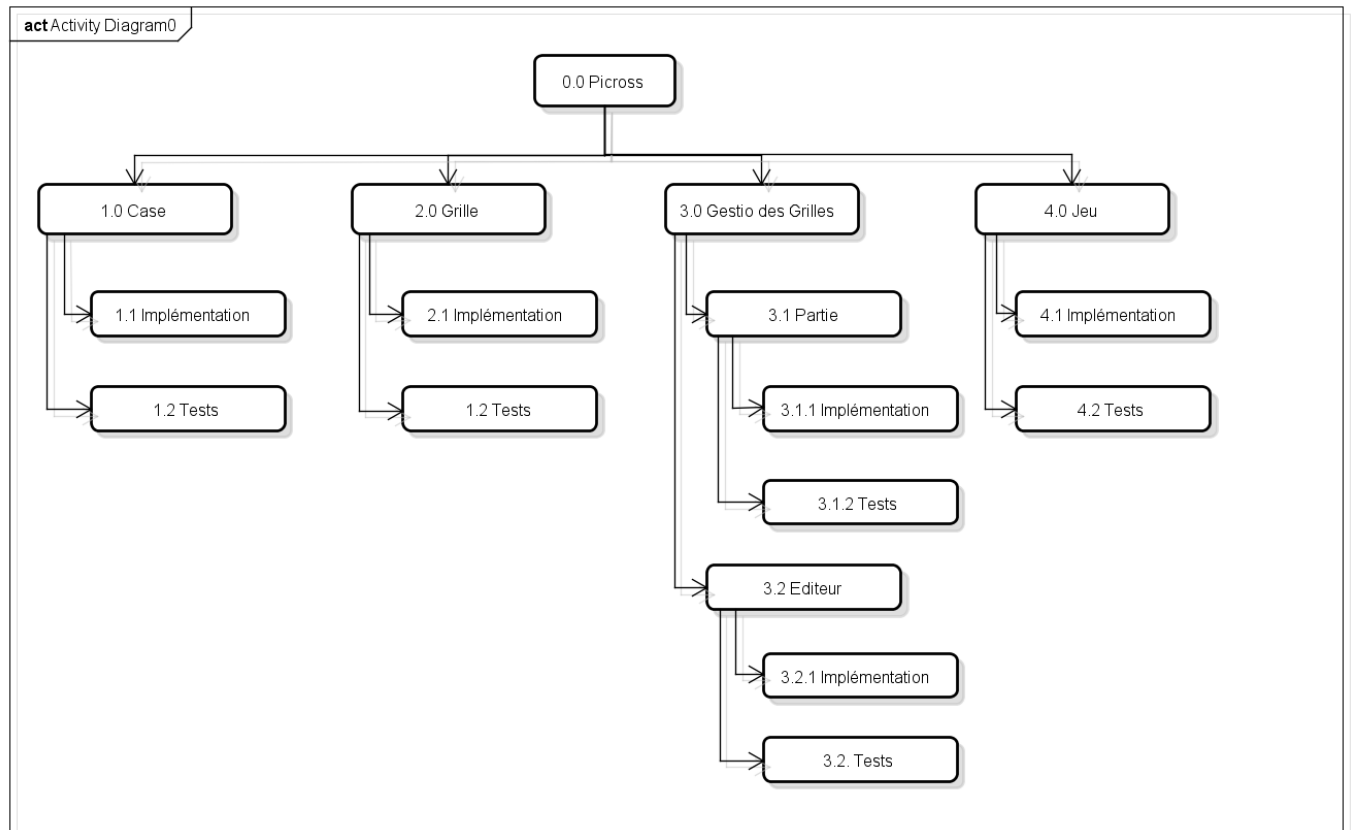
Voici les différents outils utilisés pour le projet :

- Le langage de programmation Ruby ;
- GTK pour Ruby pour le développement des interfaces graphiques ;
- YAML : outil qui permet la sérialisation de données, et qui sera donc utilisé pour la sauvegarde de données, telles que les grilles, les statistiques, la partie en cours ;
- GitHub pour la gestion de projet.



## 4.5 Diagramme WBS

La figure n 2 représente le diagramme WBS(Struture de découpage du projet), qui correspond à la décomposition hiérarchique du travail que notre équipe doit exécuter pour remplir les objectifs et produire les livrables voulus en respectant les délais.



powered by Astah

Figure 2: Diagramme WBS

## 4.6 Diagramme de Gantt

Il s'agit d'une représentation graphique de l'avancement temporel du projet, visible sur la figure n 3.

N	Nom	Durée	Date Début	Date Fin	Pred.
1	Rédaction du Cahier des charges	26 Jours	24/01/2014	28/02/2014	o
2	Rédaction du Dossier de Conception	7 jours	03/03/2014	11/03/2014	1
3	Developpement des fonctionnalités principales	30 jours	12/03/2014	22/04/2014	2
4	Rédaction du Manuel utilisateur	30 jours	12/03/2014	22/04/2014	2
5	Test des fonctionnalités principales	2 jours	23/04/2014	24/04/2014	3
6	Correction des erreurs	2 jours	25/04/2014	28/04/2014	5
7	Developpement des fonctionnalités optionnelles	7 jours	29/04/2014	7/05/2014	6
8	Test des fonctionnalités optionnelles	2 jours	08/05/2014	9/05/2014	7
9	Correction des erreurs	2 jours	12/05/2014	13/05/2014	8
10	Relecture du Projet	2 jours	14/05/2014	15/05/2014	6, 9
11	Remise du Produit	1 jour	16/05/2014	16/05/2014	6, 9

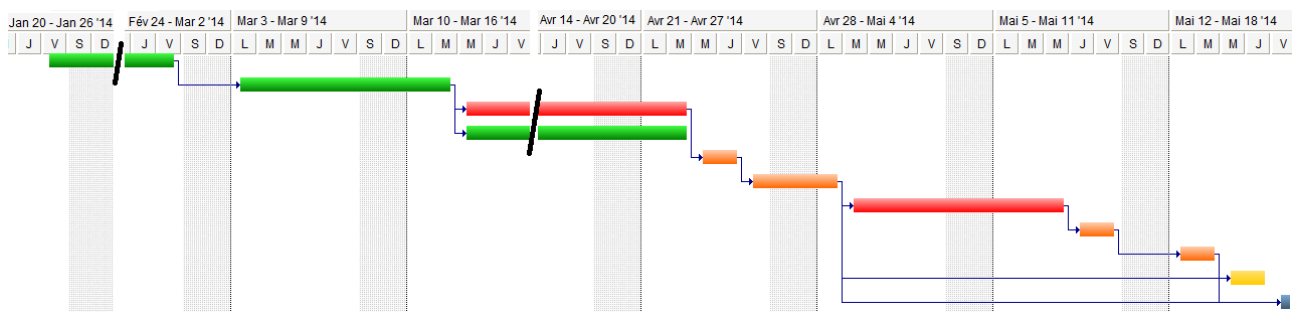


Figure 3: Diagramme de Gantt

## 5 Glossaire

**GTK** : Gimp ToolKit

**POO** : Programmation Orientée Objet

**YAML** : YAML ain't markup language (acronyme récursif)

Note : Les chiffres et pourcentages sont fournis à titre indicatif. Il pourront être modifiés pour une meilleure jouabilité.