



Cahier des charges N° 2018-L3INFO

du groupe 4

Conception d'une application  $\pi$ Cross

permettant la création et l'aide à la résolution de Puzzle de type Picross

**Benoit Combasteix, Valentin Lion, Martin Lebourdais, Arthur Prodhomme, Benjamin Di Carlo,  
Nathan Oualet, Ouassim Messagier, Samed Oktay**

---

## Sommaire

---

1Contexte.....	3
2Objectifs.....	3
3Expression des besoins.....	3
4Faisabilité technique.....	3
5Fonctionnalités.....	3
5.1Sauvegarde/Chargement.....	3
5.2Difficulté.....	4
5.3Aide.....	4
5.4Checkpoint.....	4
5.5Didacticiel.....	4
5.6Interface.....	4
5.7Modes de jeu.....	5
5.8Jouabilité.....	5
6Contraintes.....	6
6.1Planning prévisionnel.....	6
7Livrables.....	7
8Interface Homme Machine.....	7
8.1Menu de départ.....	7
8.2Menu Principal.....	8
8.3Menu de sélection des différents modes de jeu .....	9
8.4Interface de Jeu.....	9
8.5Interface de fin.....	10

## 1 Contexte

Le principe est de développer une application permettant la création et l'aide à la résolution de puzzles de type Picross suffisamment complexe pour justifier un travail en équipe et la mise en œuvre des préceptes de Génie Logiciel.

Le Picross est un jeu de réflexion solitaire qui consiste à découvrir un dessin sur une grille en noircissant des cases d'après des indices logiques laissés sur le bord de la grille. Par exemple, sur une ligne, les chiffres 5 et 6 indiquent qu'il y a de gauche à droite un bloc de 5 cases puis un bloc de 6 cases. Ces deux blocs sont au minimum séparés d'une case. Concernant les colonnes, les chiffres 2 et 3 indiquent de haut en bas, un bloc de 2 cases puis 3 cases, toujours séparés par au minimum 1 case.

## 2 Objectifs

Les objectifs pédagogiques sont d'apprendre à travailler en équipe, de pratiquer les différentes phases de la conception d'un logiciel (analyse des besoins, cahier des charges, modélisation, documentation) et de « vendre » le logiciel développé sous la forme d'une présentation orale devant les commanditaires.

## 3 Expression des besoins

- Mettre en place une application permettant la création de puzzles de type Picross
- Gérer un système d'aide permettant la résolution

## 4 Faisabilité technique

- Développement d'une application en langage « Ruby »
- Utilisation d'une méthode de sérialisation « Marshall »

## 5 Fonctionnalités

### 5.1 Sauvegarde/Chargement

La sauvegarde est automatique, elle s'effectue lorsque le jeu est quitté par l'utilisateur.

L'utilisateur a la possibilité de charger une partie existante, pour cela il se rend dans le menu JOUER et choisit CHARGER UNE PARTIE.

## 5.2 Difficulté

3 niveaux de difficulté :

- Facile : Somme des chiffres > moitié taille de la grille
- Moyen : Somme des chiffres = moitié taille de la grille
- Difficile : Somme des chiffres < moitié taille de la grille

## 5.3 Aide

Le système d'aide indique une ligne/colonne facilement « remplissable ».

Sur les grilles difficiles, l'aide enlève/barre le numéro déjà rempli.

Ce système a un coût, suivant l'aide proposée, du temps est rajouté au chronomètre.

Aide	Coût( en s )	Nombre maximum
Indique ligne/colonne à remplir	30	5
Indique plus gros chiffre des différentes lignes/colonnes	15	3
Indique case à colorier	120	3

## 5.4 Checkpoint

Le checkpoint permet de revenir à une situation antérieure, ce système est associé aux hypothèses.

Une hypothèse permet de formuler une réponse pour tester différentes solutions sans être certain de leur réussite. Plusieurs hypothèses peuvent être formulées qui sont représentées en différentes couleurs. Il est possible de les annuler.

## 5.5 Didacticiel

Le didacticiel montre le fonctionnement du jeu pour un nouvel utilisateur en le guidant pas à pas dans la résolution d'une grille.

## 5.6 Interface

Sur la première page de l'application, l'utilisateur rentre son pseudo, ensuite soit il joue soit il regarde le scoreboard. Si il choisi la première option plusieurs options s'offrent à lui :

- nouvelle partie
- charger une partie

Si il choisit de faire une nouvelle partie alors :

- choix du mode de jeu

Puis :

- choix de la difficulté

Et :

- choix de la taille de la grille

Enfin il va pouvoir jouer sur la grille qui sera affichée.

## 5.7 Modes de jeu

L'utilisateur peut choisir différents modes de jeu :

- un mode entraînement sans score
- un mode aventure qui consiste à réussir une grille pour débloquer la suivante tout en suivant un scénario. Ce mode est chronométré et comporte des Picross plus difficiles en fin de niveau (un « boss »).
- un mode compétitif pour se mesurer aux autres utilisateurs. Ce mode rassemble le classement des meilleurs temps réalisés.

## 5.8 Jouabilité

L'utilisateur peut jouer avec la souris ou le clavier. Voici un petit tableau pour visualiser les différentes touches :

	Souris	Clavier
Case coloriée	clique gauche	N + clique gauche
Case blanche	clique gauche sur une case déjà coloriée	B + clique gauche
Croix	clique droit	C + clique gauche

L'utilisateur peut surligner un ensemble de case en faisant défiler sa souris. En faisant cela, il peut voir à côté de sa souris le nombre de case qu'il est en train de surligner. Dans le cas où il surligne en commençant à côté d'une ou plusieurs cases déjà coloriées alors il voit apparaître en dessous le nombre totale de cases qui sera colorié, comme ceci : 5/7.

Concernant les hypothèses (changement de couleur des cases), l'utilisateur doit cliquer gauche sur une case hypothèse puis la couleur change automatiquement.

## 6 Contraintes

Les contraintes de ce projet sont multiples, en effet nous devons rendre le livrable avant le 12 avril 2018. Le langage de programmation nous est imposé il s'agit de Ruby.

### 6.1 Planning prévisionnel

Afin de respecter la deadline imposée, Voici notre planning prévisionnel sous forme de diagramme de Gantt.

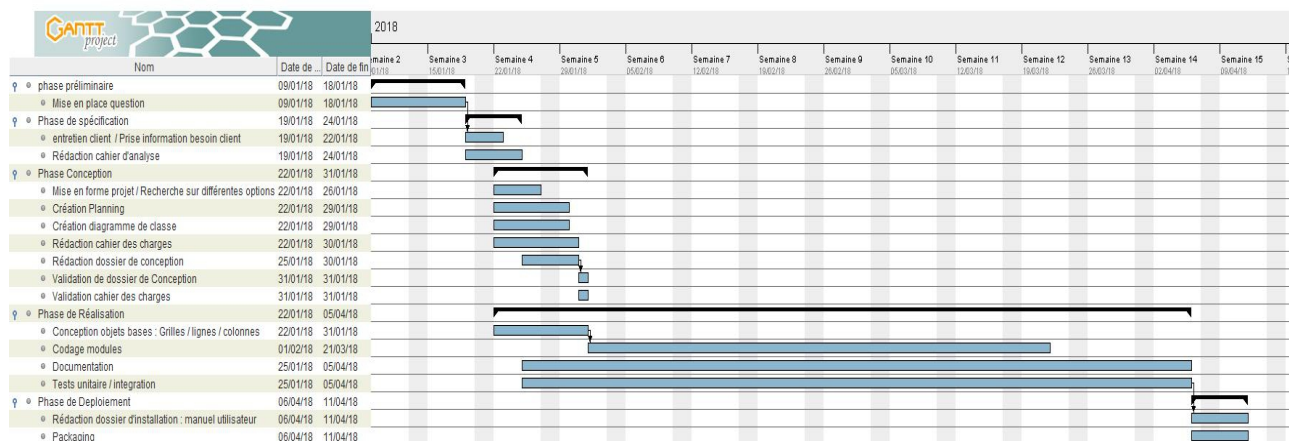


Illustration 1: Diagramme de GANTT

Puis nous avons fais un diagramme de PERT pour modéliser de manière plus général ce planning.

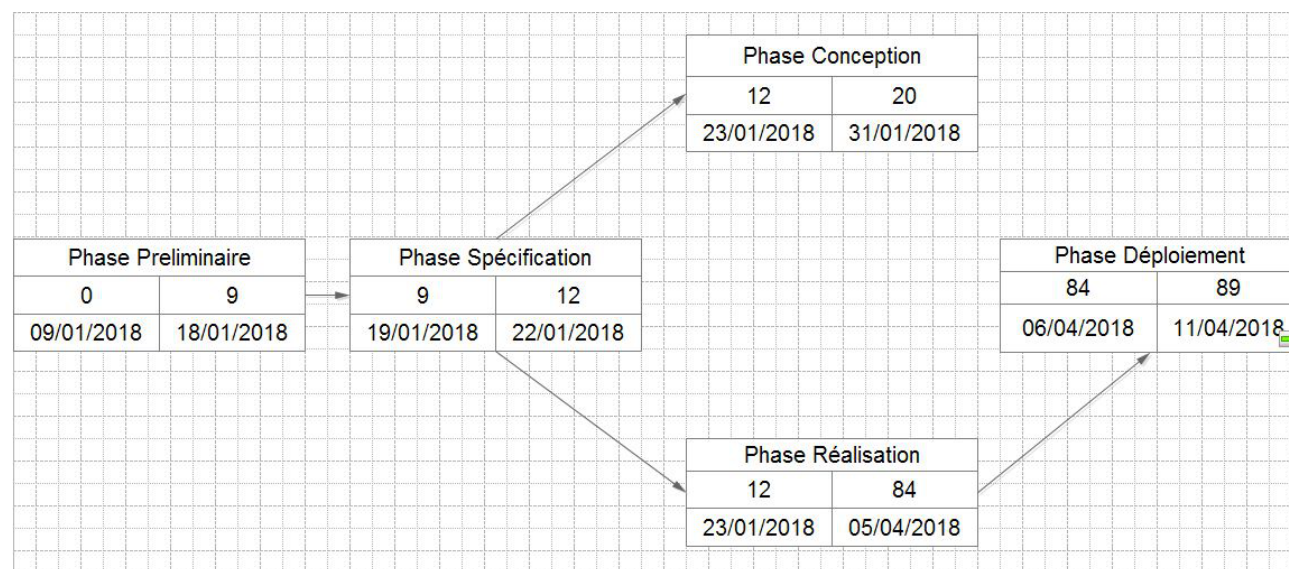


Illustration 2: Diagramme de PERT

## 7 Livrables

Lors de la période du projet et avant la date de livraison seront livrés :

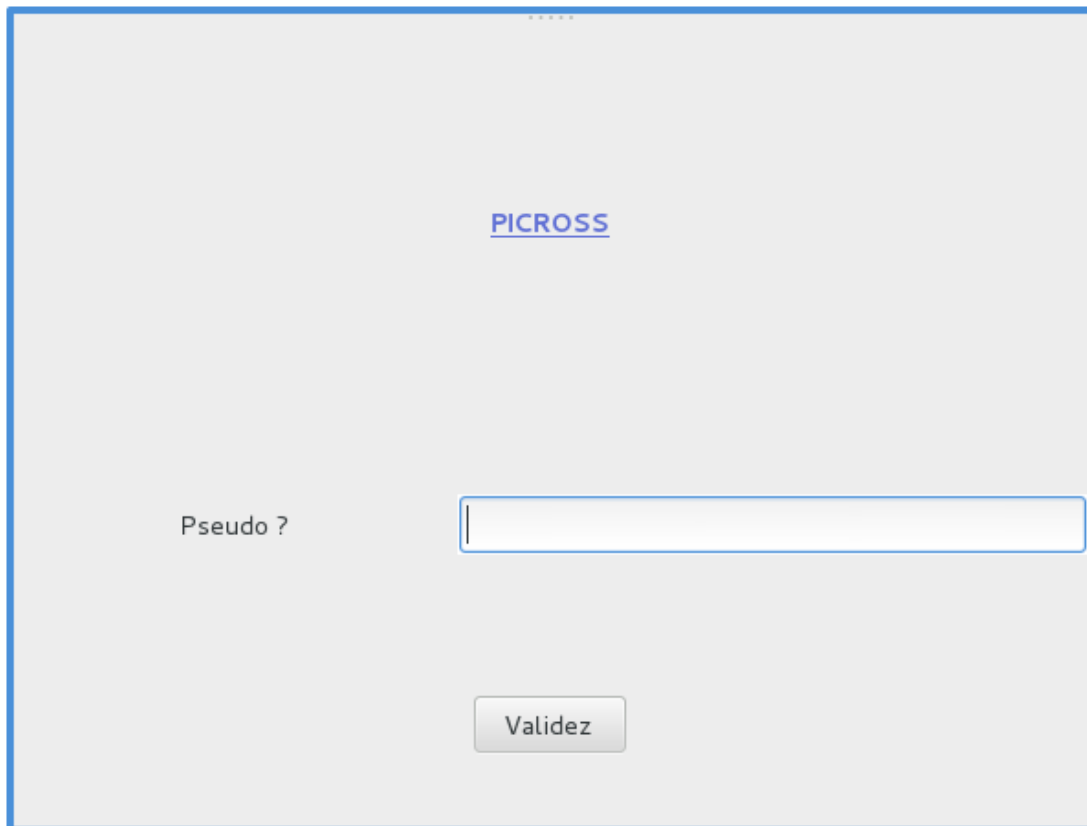
- Un cahier des charges qui sera préalablement validé
- Le logiciel qui respecte ce cahier des charges
- Un manuel d'utilisateur

## 8 Interface Homme Machine

Les interfaces présentées ci-dessous représentent l'interface fonctionnelle c'est à dire que seulement les différentes fonctions disponible à l'utilisateur sont présente mais cela ne représente le modèle décoré.

### 8.1 Menu de départ

Le menu de départ comporte le logo du jeu et une demande de pseudo qui sera nécessaire a la sauvegarde et à l'entrée des highscores.

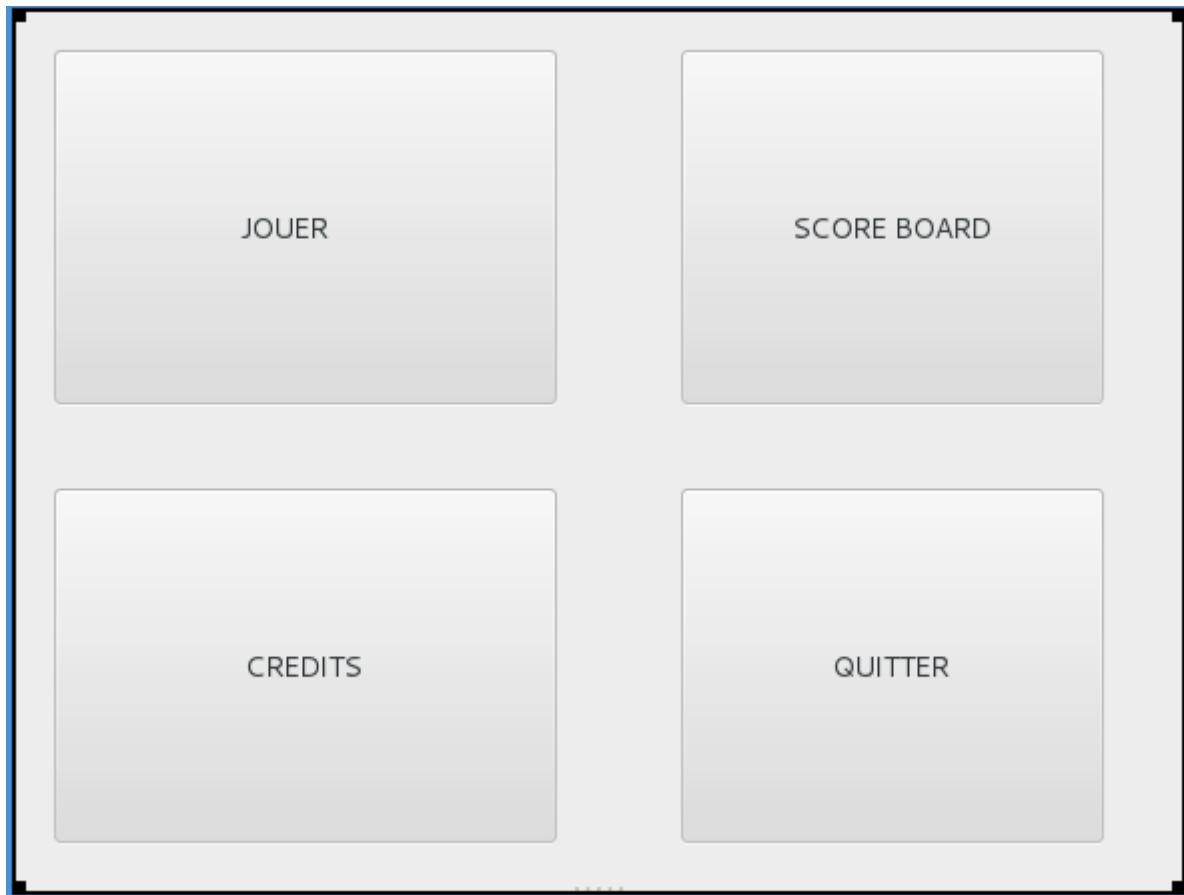


*Illustration 3: Menu I*

## 8.2 Menu Principal

Le menu principal comporte 4 boutons :

- le premier « JOUER » qui envoie l'utilisateur sur le menu de sélection des différents modes de jeu
- le second « SCOREBOARD » qui l'enverra sur l'interface des scoreboard
- le troisième « CREDITS » qui emmènera vers l'interface des crédits
- le dernier bouton « QUITTER » qui enverra vers l'interface de remerciement.

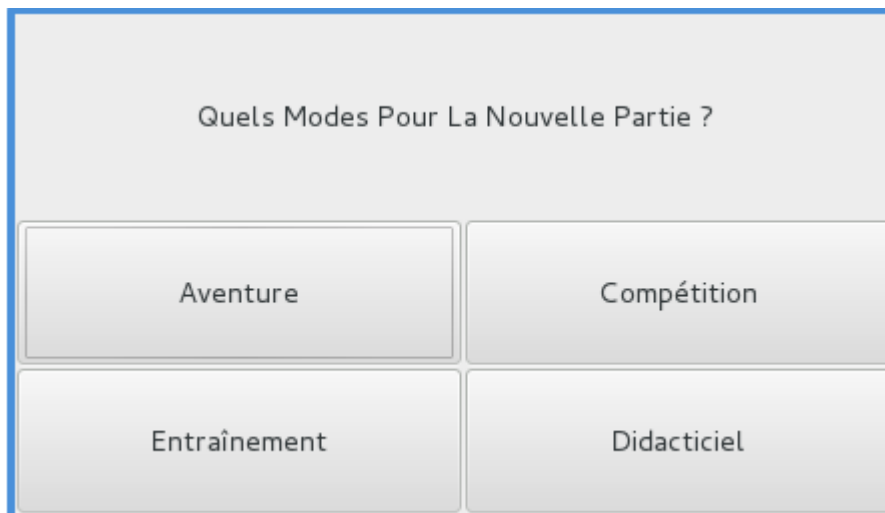


*Illustration 4: Menu II*



### 8.3 Menu de sélection des différents modes de jeu

Le menu de sélection des différents modes de jeu comporte 4 boutons qui emmène chacun sur l'interface de jeu qui leurs correspondent.



*Illustration 5: Menu III*

### 8.4 Interface de Jeu

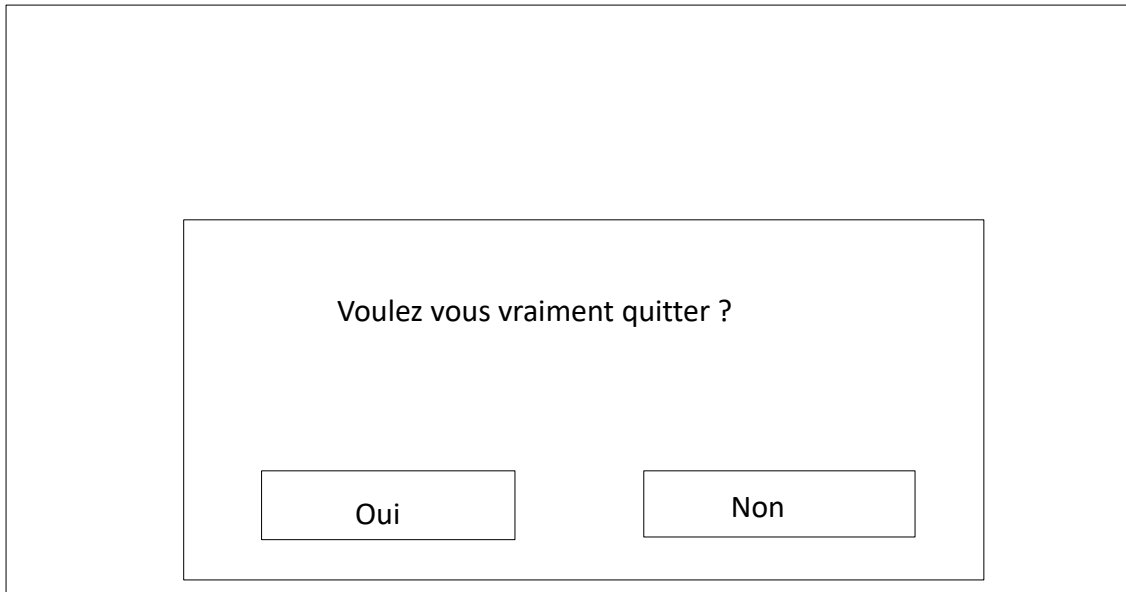
L'interface de jeu comporte la zone de la grille ou on pourra jouer au jeu du Picross, une zone ou est afficher le temps écoulé, une zone avec les différentes aides disponible avec leurs pénalités ainsi que cinq boutons pour recommencer la grille, pour quitter le jeu, pour sauvegarder la progression, pour charger une partie et pour passer en mode hypothèse.



*Illustration 6: Menu IV*

### 8.5 Interface de fin

Cette dernière interface demande à l'utilisateur s'il veut vraiment quitter, si oui l'interface se ferme, si non l'utilisateur est envoyé vers le menu principal.



The diagram shows a rectangular window representing a dialog box. Inside this window is a smaller, centered rectangle. Within this inner rectangle, the text "Voulez vous vraiment quitter ?" is centered at the top. Below the text, there are two rectangular buttons positioned side-by-side. The left button contains the text "Oui" and the right button contains the text "Non".

*Dessin 1: Menu V*