

## **Le Mans Université**

Licence Informatique 3ème année

Module Génie Logiciel 2

**Système d'aide à la résolution de puzzle : HASHI**

### **Cahier des charges**

#### **Groupe 4**

**Aaron AMANI, Axel JOURRY, Cindy CALVADOS, Clément JANVIER, Collins Soares, Florian DREUX, Rayyan LAJNEF, Thomas MALABRY, Willhem LIBAN**

13 Janvier 2021

## I. Introduction

- A - Présentation du projet
- B - Objectifs du projet
- C - Critères d'acceptabilités

## II. Gestion de projet

- A - Mise en place des rôles
- B - Diagramme de Gantt

## III. Besoins

- A - Besoins utilisateurs
- B - Scénario d'utilisation

## IV. Fonctionnalités

- A - Fonctionnalités Principales
- B - Maquette

## V. Contraintes

- A - Exigences Fonctionnelles
- B - Exigences des clients
- C - Les différents risques

## VI. Divers

- A - Glossaire

# I. Introduction

## **A - Présentation du projet**

Dans le cadre du semestre 6 de notre licence Informatique à l'Université du Mans et afin de valider notre module de Génie Logiciel 2, nous avons eu la consigne de créer un système d'aide à la résolution de puzzle, pouvant notamment être utilisé dans un jeu de Hashi. Ce dernier se doit d'avoir une difficulté suffisamment complexe, afin de justifier un système d'aide.

Le Hashi, de son nom complet Hashiwokakero, a été créé en 1990 dans le magazine Puzzle Communication publié par Nikoli. Le Hashi se joue à un joueur et se constitue d'une grille, remplie partiellement de cercles numérotés. L'objectif du joueur est de relier chacun de ces cercles de telle façon que le nombre de traits partant d'un cercle soit égal au chiffre inscrit dans celui-ci. De plus, il faut que chaque cercle soit relié entre eux de sorte à former une unique composante connexe. Chaque trait part à un point cardinal d'un cercle, il n'y a donc aucun trait allant en diagonale. Il ne peut donc pas y avoir de liaison directe entre deux cercles qui se trouvent sur des colonnes et des lignes différentes.

## **B - Objectifs du projet**

Les objectifs du projet sont multiples. Notre travail principal dans ce projet est la création d'un jeu de type Hashi ainsi que le codage d'un système d'aide à la résolution de puzzle. Ce projet, étant réalisé dans un cadre universitaire, a également pour but de nous enseigner les fondamentaux de la réalisation d'un projet en équipe de 9 (et non d'un petit groupe de 2 ou 3 comme nous en avons l'habitude).

Ce projet nous permet aussi de mettre en application les différentes compétences que nous avons assimilées les semestres précédents, telles que la conception de modèles UML, le Ruby ou encore GTK.

## **C - Critères d'acceptabilités**

Lors de la réalisation de ce projet et pour que celui-ci soit valide, plusieurs critères sont à respecter :

- Le jeu doit être ergonomique.
- Le joueur doit pouvoir accéder à différents niveaux d'aide au cours d'une partie
- Le joueur doit pouvoir sauvegarder la partie
- L'application doit être facilement utilisable

## II. Gestion de projet

### A - Mise en place des rôles

Avec un effectif plus conséquent que ce à quoi nous étions habitués, nous devons dès le début du projet définir des rôles précis pour le bon déroulement du projet. Suite au conseil des enseignants, nous avons donc désigné un chef de projet, chargé de veiller à la communication avec nos enseignants et de l'organisation du projet. Nous avons aussi assigné un documentaliste, qui est chargé de la mise à jour des documents relatifs au projet : cahier des charges, diagramme de classe, diagramme de Gantt.

### B - Diagramme de Gantt

Hashi

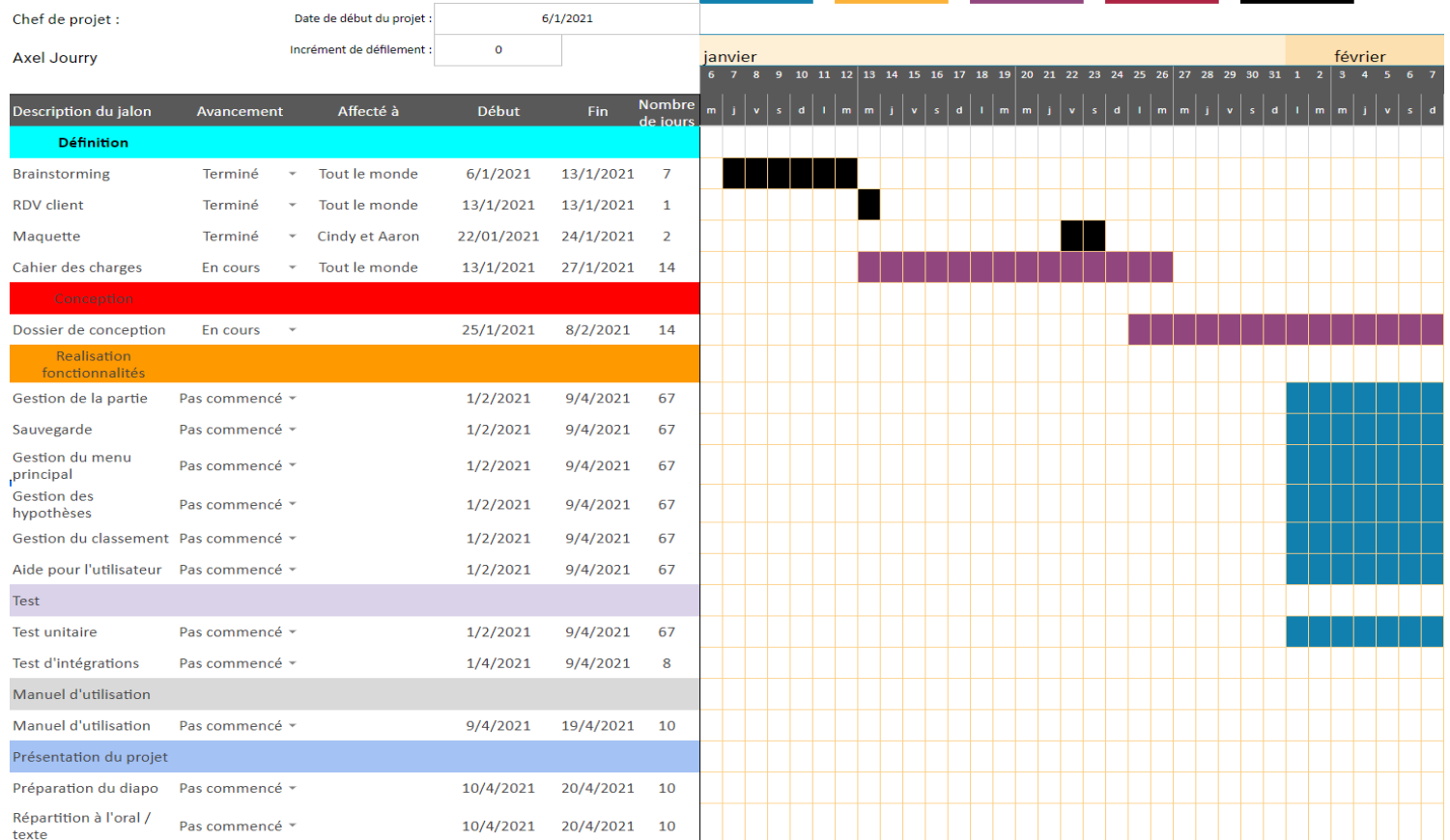


Figure 1. Diagramme de Gantt

## III. Besoins

### A - Besoins utilisateurs

Les utilisateurs seront des personnes de tout âge à partir du moment où ces derniers savent compter (appelées "joueur" pour la suite).

Pour les joueurs, le jeu doit permettre :

- de se détendre, de s'amuser
- de faire travailler son esprit
- de progresser
- de choisir la difficulté.

Pour les joueurs, le système d'aide à la résolution doit :

- aider le jeu à être plus intuitif
- proposer des aides plus ou moins explicites aidant à résoudre un puzzle de type Hashi
- proposer une (ou plusieurs) aide(s) à la résolution à chaque niveaux

### B - Scénario d'utilisation

**Le Jeu du Hashi** : Jeu sur ordinateur basé sur le langage Ruby. Permet la résolution de puzzles de type Hashi , tout en proposant une aide à la résolution pour pouvoir débloquer le joueur et l'aider à s'améliorer.

Pour le déroulement de cette partie nous allons suivre **Tanguy**, un ami des développeurs qui tente le jeu pour la première fois et qui n'a jamais joué à un Hashi.

**Tanguy** obtient les fichiers sources du jeu et lance le `.sh` contenu dans les dossiers. La page d'accueil se lance, et plusieurs boutons apparaissent devant lui. Il clique sur le bouton "Choix niveau". Sur le menu de sélection de niveau, plusieurs niveaux de difficultés lui sont proposés : Facile (8X8), Intermédiaire (12X12) et Difficile (15X15). Il choisit de prendre le niveau Facile.

Malheureusement **Tanguy** se rend compte, une fois devant sa grille, qu'il ne connaît même pas les règles. Il décide donc de retourner sur le menu principal, car il se souvient avoir vu un mode tutoriel. Pour ce faire, il y arrive en cliquant sur le bouton "Pause", un menu apparaît sur lequel sont affichées plusieurs options telles que "Recommencer" et "Quitter". **Tanguy** quitte sa partie, celle-ci est automatiquement sauvegardée.

De retour sur le Menu principal, **Tanguy** choisit le mode tutoriel. Une succession d'images lui est alors fournie afin de lui expliquer les règles. Une fois le tutoriel terminé, **Tanguy** retourne sur le Menu Principal afin de reprendre sa partie. Pour ce faire, il clique de nouveau sur le bouton "Choix niveau". La liste des différents niveaux lui sont encore proposés. Il sélectionne alors le même niveau que précédemment. Le jeu lui précise alors qu'une sauvegarde existe déjà pour ce niveau. Il peut ainsi choisir entre "Nouveau" et "Charger". **Tanguy** choisit la dernière option et reprend donc sa partie là où il l'avait laissée.

Après quelques minutes de jeu, **Tanguy** est perdu et ne sait pas quoi jouer, il décide de faire appel au système d'aide. En cliquant sur le bouton d'aide, une fenêtre d'aide apparaît. Celle-ci lui propose un indice spécifique à sa situation sans lui donner la solution. Après avoir été mis sur la piste, **Tanguy** peut continuer à jouer.

En jouant, après plusieurs coups, **Tanguy** se rend compte qu'il s'est trompé, et souhaite revenir en arrière. Il [clique donc sur le bouton retour](#) afin d'effacer le coup qu'il vient d'effectuer.

En continuant sa grille, **Tanguy** hésite entre deux coups qui lui semble correct, il [clique donc sur le bouton hypothèse](#), qui lui permettra de retourner à la position présente quand il le souhaite. Après plusieurs coups, il se rend compte que son choix n'était pas le bon, il [clique donc sur le bouton de fermeture d'hypothèse](#), il essaie alors le deuxième choix qu'il avait.

Après quelques minutes, **Tanguy** termine la grille. Le jeu le félicite et lui propose de [sauvegarder son score](#), ce qu'il accepte, il entre alors son nom et appuie sur le [bouton "Valider"](#).

Le jeu affiche alors le classement pour ce niveau. **Tanguy** peut ensuite quitter cette fenêtre grâce au bouton ["Menu principal"](#).

Une fois de retour sur [l'écran d'accueil](#), **Tanguy** clique sur le bouton ["A propos"](#) pour voir ce qu'il y a dedans. Il y voit le nom de ses amis développeurs et des informations concernant l'application. Ayant terminé son essai de l'application, il retourne au menu principal afin de sortir du jeu en appuyant sur le [bouton "Quitter"](#).

## IV. Fonctionnalités

### A - Fonctionnalités Principales

Suite à l'étude du scénario d'utilisation ci-dessus, nous arrivons à une liste de fonctionnalités principales ainsi que des catégories dans lesquelles les ranger.

Référence	Fonctionnalités
BF 01	<ul style="list-style-type: none"><li>→ Gestion de la partie:<ul style="list-style-type: none"><li>◆ 1.1 Gestion du chronomètre.</li><li>◆ 1.2 Annulation des actions ( undo / redo )</li><li>◆ 1.3 Mise en pause du jeu</li></ul></li></ul>
BF 02	<ul style="list-style-type: none"><li>→ Gestion de la sauvegarde:<ul style="list-style-type: none"><li>◆ 2.1 Chargement d'une partie existante</li><li>◆ 2.2 Destruction d'une sauvegarde</li></ul></li></ul>
BF 03	<ul style="list-style-type: none"><li>→ Gestion du Menu principal :<ul style="list-style-type: none"><li>◆ 3.1 Nouvelle partie.<ul style="list-style-type: none"><li>● 3.1.1 Choix mode du jeu<ul style="list-style-type: none"><li>○ 3.1.1.A Choix difficulté du jeu</li></ul></li></ul></li><li>◆ 3.2 Reprise d'une partie<ul style="list-style-type: none"><li>○ Voir BF 2.1</li></ul></li><li>◆ 3.3 Quitter le jeu.</li></ul></li></ul>
BF 04	<ul style="list-style-type: none"><li>→ Gestion de l'aide<ul style="list-style-type: none"><li>◆ 4.1 Demande d'aide<ul style="list-style-type: none"><li>● 4.1.1 Recherche de l'aide à afficher</li><li>● 4.1.2 Affichage d'une aide spécifique à la situation du joueur</li></ul></li></ul></li></ul>

BF 05	→ Gestion des Hypothèses: <ul style="list-style-type: none"> <li>◆ 5.1 Création d'une nouvelle hypothèse</li> <li>◆ 5.2 Validation d'une hypothèse</li> <li>◆ 5.3 Annulation d'une hypothèse</li> </ul>
BF 06	→ Gestion du classement: <ul style="list-style-type: none"> <li>◆ 6.1 Classement par niveau <ul style="list-style-type: none"> <li>● 6.1.1 Classement en fonction du temps</li> </ul> </li> </ul>
BF 07	→ Interactions grille-utilisateur : <ul style="list-style-type: none"> <li>◆ 7.1 Interaction avec les noeuds</li> <li>◆ 7.2 Indices sur les coups possibles</li> </ul>

## **BF01 - Gestion de la partie**

Comme son intitulé l'indique, cette fonctionnalité a pour responsabilité la gestion d'une partie. Cela englobe le lancement de la partie et l'initialisation de la grille selon la taille sélectionnée par l'utilisateur.

### 1.1 - Gestion du chronomètre

Lors du lancement d'une grille , un chronomètre est lancé, il correspond au temps passé sur celle-ci. Autrement dit, il mesure le temps nécessaire à la résolution de la grille. Le chrono s'arrête automatiquement à la fin du niveau ou si le jeu est mis en pause.

### 1.2 - Annulation des actions ( undo / redo )

L'utilisateur peut choisir d'annuler une action et de revenir à l'état antérieur (undo), tant qu'il n'effectue pas de nouvelles actions il peut revenir à l'action avant le premier undo ( redo ).

### 1.3 - Mise en pause du jeu

Cette fonctionnalité permet de mettre en pause le chronomètre et d'afficher un menu temporaire recouvrant la grille (afin d'éviter toute tricherie qui consisterait à analyser le plateau sans que le chrono soit décompté). Plusieurs actions seront disponibles :

- recommencer la grille
- quitter la partie



## **BF02 - Gestion de la sauvegarde**

Cette fonctionnalité gère les actions liées à la sauvegarde de partie. Une sauvegarde automatique est réalisée à chaque coup.

### 2.1 - Chargement d'une partie existante

Permet à l'utilisateur de reprendre là où il s'est arrêté au moment de la sauvegarde

### 2.2 - Destruction d'une sauvegarde

Permet de supprimer une sauvegarde donnée

## **BF03 - Gestion du Menu principal :**

Le menu principal permet à l'utilisateur d'accéder à de nombreuses fonctionnalités, telles que la création d'une nouvelle partie, l'accès aux parties sauvegardées, aux modes du jeu et à l'affichage du classement.

### 3.1 - Nouvelle partie

Ouvre un menu permettant de lancer une nouvelle partie.

#### 3.1.1 Sélection Jeu/Tutoriel

Une fois la partie lancée, l'utilisateur peut choisir entre deux modes :

- "Normal"

Le mode normal correspond à une partie classique.

- 3.1.1.A Choix difficulté du jeu

Avant de commencer, l'utilisateur choisit le type de grille qu'il souhaite résoudre : 8x8, 12x12 ou 15x15.

- "Tutoriel"

Le mode tutoriel lance une grille facile au joueur en le guidant automatiquement sur les actions qu'il doit effectuer, afin qu'il puisse saisir le fonctionnement du jeu et en comprendre ses règles.

### 3.2 - Reprise d'une partie

Pouvoir charger une partie qui a été sauvegardée précédemment. Cela permet à l'utilisateur de reprendre une partie où il s'est arrêté. Voir BF 2.1

### 3.3 - Quitter le jeu

Ferme le jeu.

## BF04 - Gestion de l'aide

Le système d'aide ouvre une fenêtre qui donne au joueur un indice spécifique b

### 4.1 Demande d'aide

#### - 4.1.1 Recherche de l'aide à afficher

Le jeu regarde la grille et recherche l'aide la plus appropriée pour la situation de celle-ci.

#### - 4.1.2 Affichage d'une aide spécifique à la situation du joueur :

Le jeu ouvre une fenêtre pop-up pour afficher l'aide choisie, celle-ci ne donne pas la solution mais bien une indication concernant un ou plusieurs points de la grille pouvant aider le joueur.

## BF05 - Gestion des Hypothèses

Le jeu dispose d'un *mode hypothèse*, ce mode permet de jouer sur une copie de la grille originale, permettant ainsi au joueur de tester différents chemins qui lui semblaient corrects.

### 5.1 - Création d'une nouvelle hypothèse

Lance le *mode hypothèse*, la partie originale reprendra à ce point lorsque le joueur décidera de quitter le *mode hypothèse*.

### 5.2 - Validation d'une hypothèse

Met fin au *mode hypothèse* et applique celle-ci à la grille.

### 5.3 - Annulation d'une hypothèse

Permet au joueur de reprendre sa partie au point où elle était avant le lancement du *mode hypothèse*.

## BF06 - Gestion du classement:

Un système de classement est disponible, celui-ci est mis à jour dès qu'une partie se termine. à la fin de chaque partie, les joueurs peuvent entrer leur nom afin que le classement prenne en compte le nom des joueurs y figurant.

### 6.1 - Classement par niveau

#### 6.1.1 : Classement en fonctions du temps pour finir une grille

Classement des joueurs par grille jouées ( facile, moyen, difficile ) et en fonction du temps (moins de temps on met pour finir une grille, mieux c'est). Ce classement permet de comparer son temps sur une grille donnée par rapport à ses précédents essais ainsi que par rapport à d'autres joueurs.

## **BF07 - Interactions grille-utilisateur**

### 7.1 - Interaction avec les nœuds

7.1.1 En survolant un nœud avec la souris, il passe en surbrillance, ainsi que tous les ponts qui y sont reliés directement.

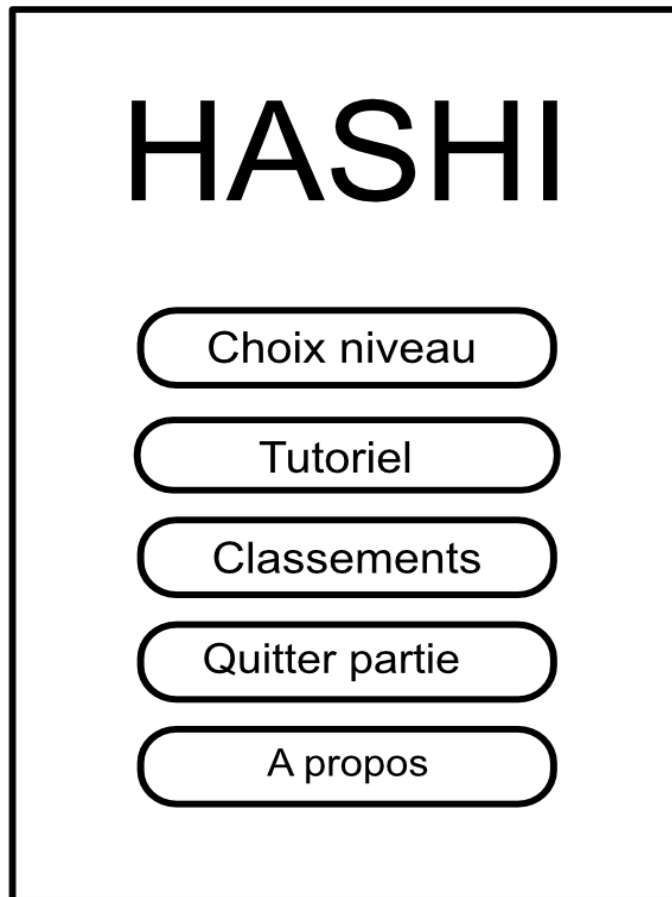
7.1.2 : Maintenir un clic sur un nœud met également en surbrillance tous les nœuds reliés à celui-ci.

### 7.2 - Indices sur les coups possibles

A aucun instant, le jeu ne fournira un coup précis au joueur. Mais plus l'utilisateur utilisera le bouton indice sans jouer, et plus le système lui donnera un indice précis lui permettant d'avancer. Le système peut donner des aides textuelles (par exemple, rappeler une règle du jeu), des indices sous formes d'affichage à l'écran (élément en surbrillance), ou bien un mélange des deux. On peut notamment penser à un système indiquant au joueur le nombre et l'emplacement de ses erreurs. Les aides proposées sont des techniques génériques, mais l'aide que le joueur recevra sera toujours une aide qui lui est utile dans sa situation actuelle (le jeu ne teste pas si l'aide fait partie de la solution au puzzle, le système ne la connaissant pas, mais propose une aide utile ).

## B - Maquettes

### 1 - Menu principal



#### Choix de niveau :

- Ouvre une fenêtre de choix de niveau. (Cette page est divisée en sous-catégorie de difficultés, Facile, Moyen et Difficile. Chaque catégorie à une liste de niveaux.)

#### Tutoriel :

- Succession d'images mimant un niveau pour apprendre les bases du jeu

#### Classements :

- Ouvre le menu des classements, ouvre une nouvelle fenêtre.

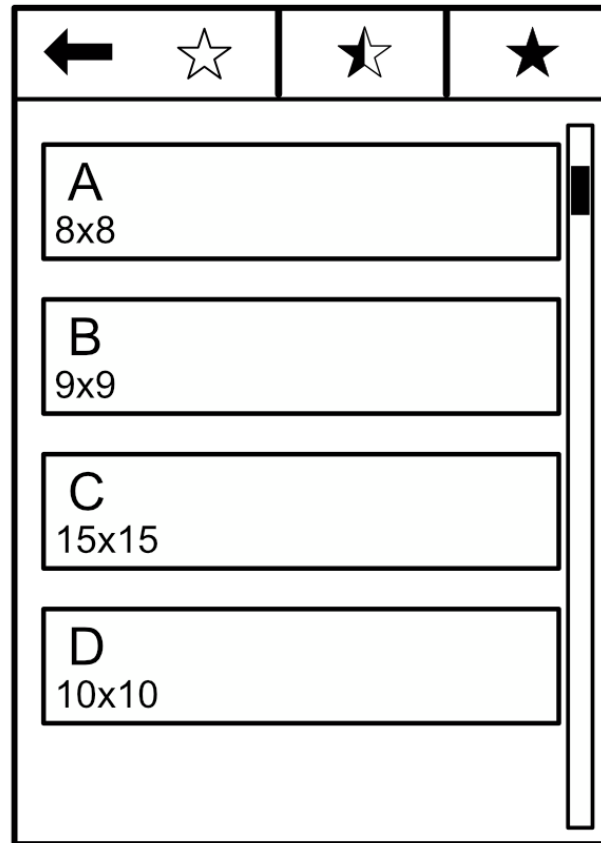
#### A propos :

- Page d'information sur les développeurs du jeu.

#### Quitter :

- Quitte le jeu.

## 2 - Sélection du plateau

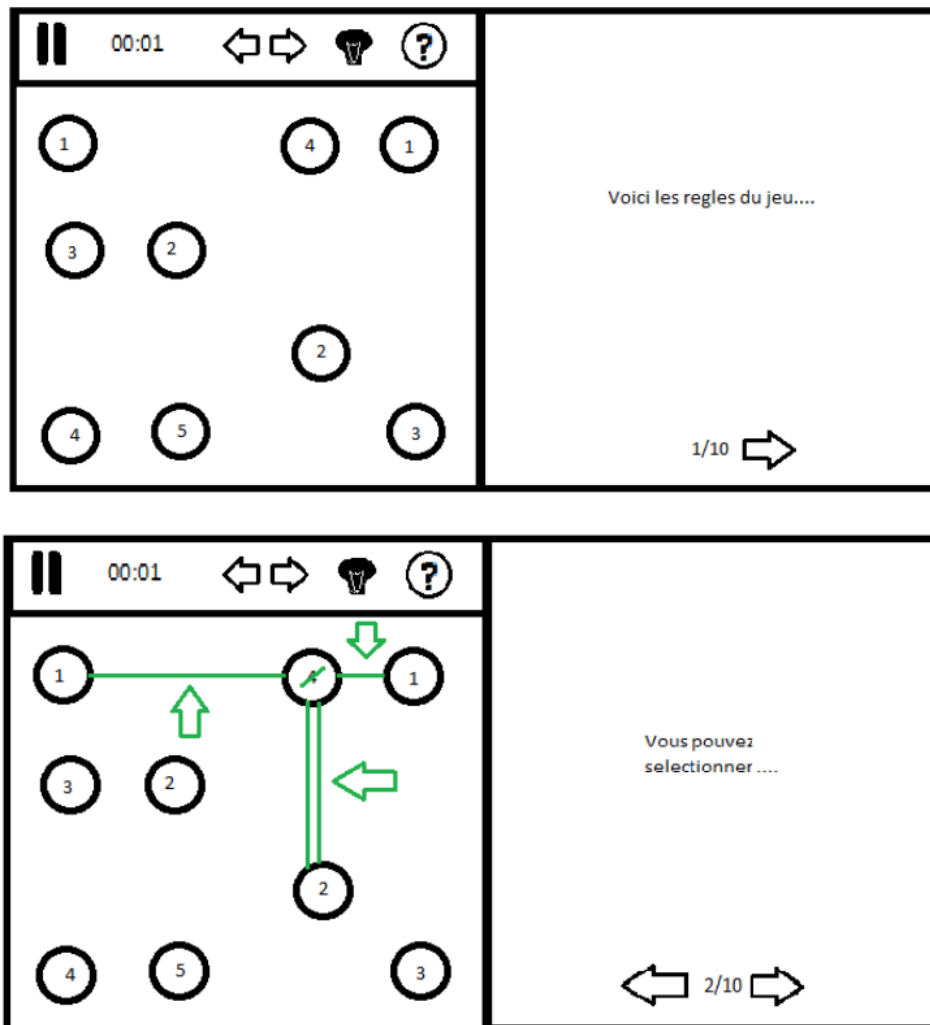


Affichage des niveaux en liste divisée en onglets représentant les différentes difficultés.  
(facile, medium, difficile)

Retour :

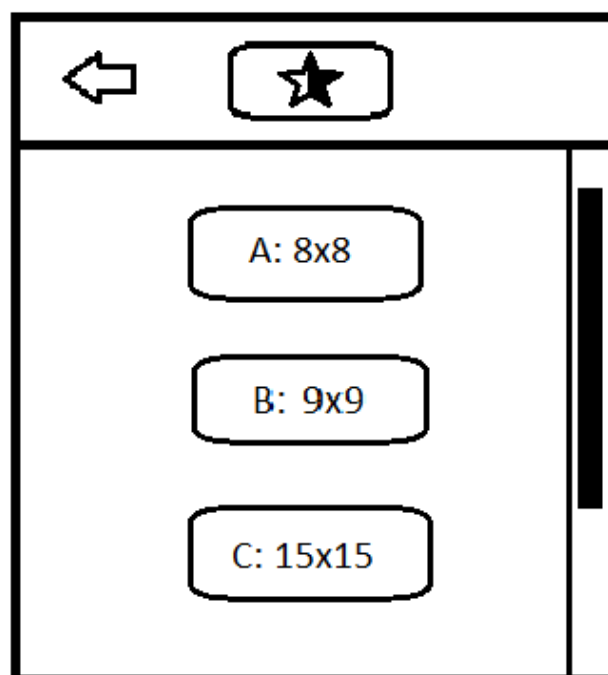
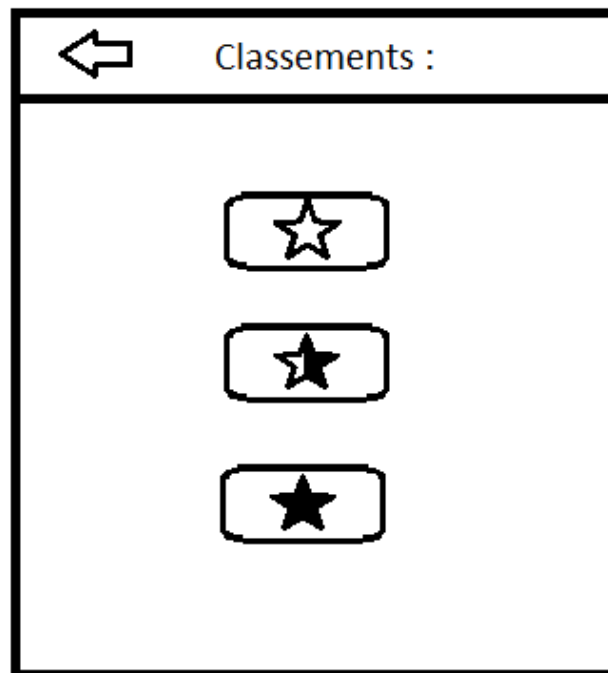
- Retour au menu principal

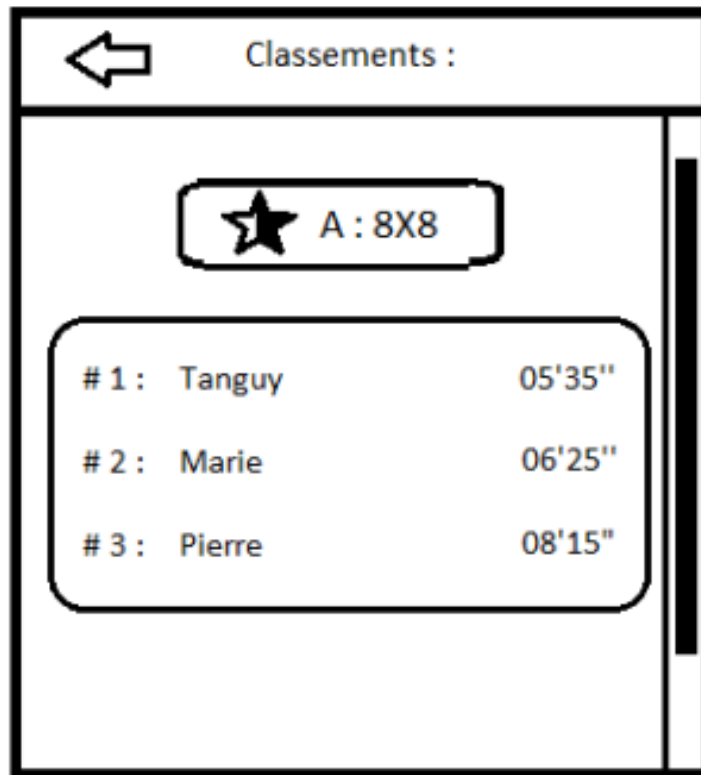
### 3 - Ecran tutoriel



Succession d'images servant à expliquer le fonctionnement du jeu. Le joueur peut naviguer entre les pages afin de comprendre les règles.

#### 4 - Classements





Possibilité de choisir le niveau pour lequel on souhaite consulter les classements.

Liste des scores du niveau :

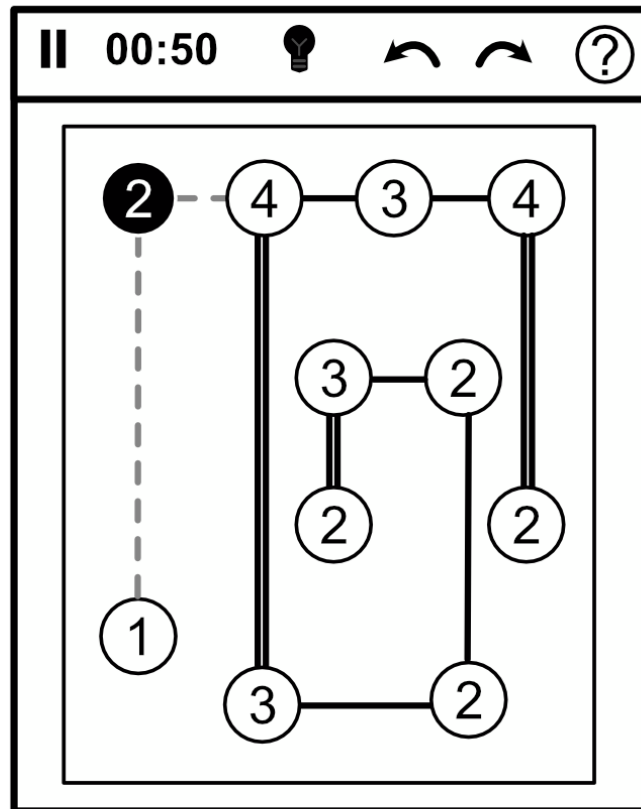
- Accès à la liste des scores après avoir cliqué sur un niveau.

Retour :

- Retour au menu principal



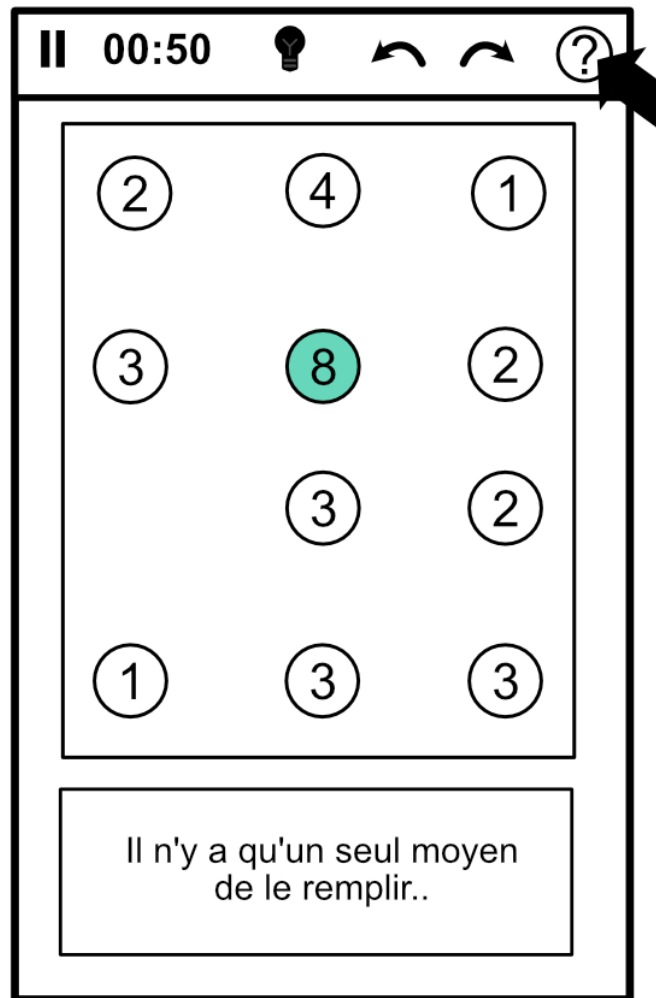
## 5 - Plateau de jeu



### Grille :

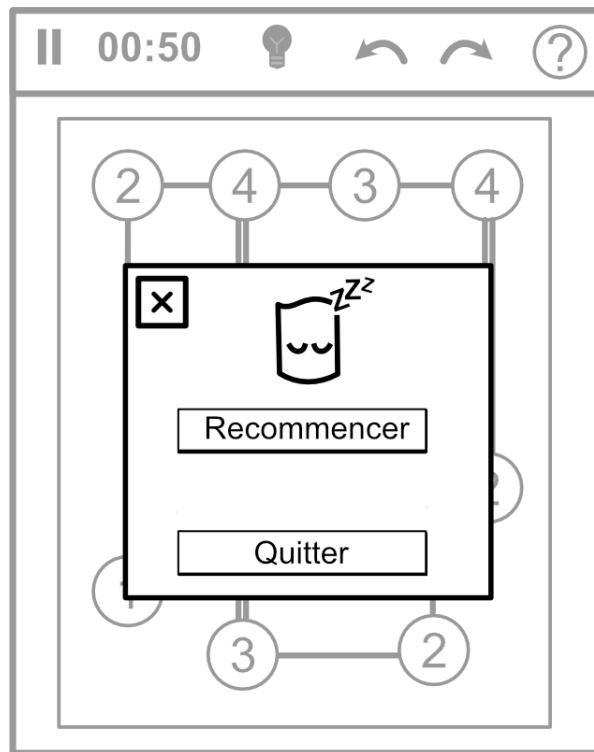
- La grille où se déroule la partie.
- Bouton pause pour afficher le menu de pause
- Le temps total de la partie
- Bouton hypothèse : Déclenche le mode hypothèse, permettant au joueur de jouer sans conséquence directe sur sa partie. (Contient un bouton valider permettant de valider l'hypothèse.)
- Undo / Redo pour revenir en arrière ou retourné en avant
- Bouton indice pour demander au jeu un indice sur le coup suivant à jouer.

## 6 - Menu d'aide



Lorsque le joueur appuie sur le bouton d'aide, le jeu affiche sous le plateau de jeu une fenêtre contenant un indice ayant un rapport avec l'état actuel du plateau. L'indice n'affiche jamais la réponse exacte mais a pour objectif d'orienter le joueur vers une action fonctionnant **dans l'état actuel du plateau**.

## 7 - Menu Pause



Petite fenêtre avec un bouton recommencer et quitter.

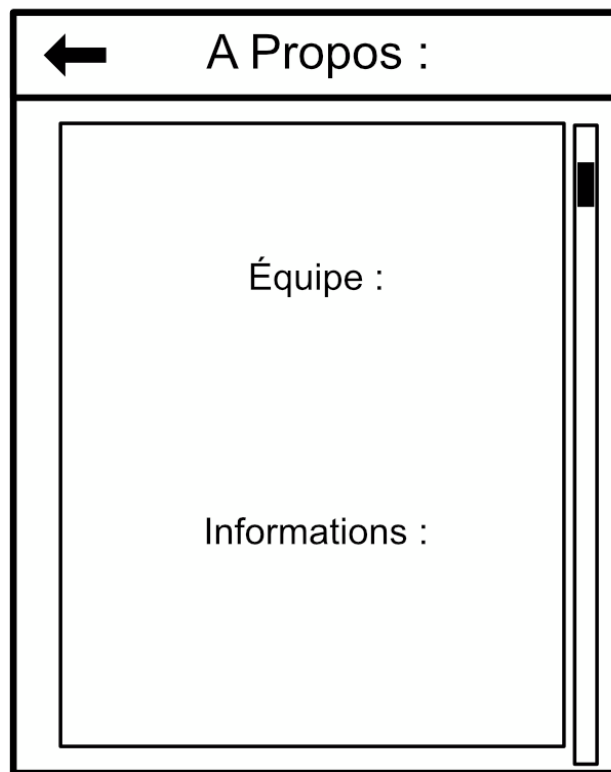
Recommencer :

Ferme le menu pause puis redémarre le niveau et le chronomètre

Quitter :

Ferme le niveau et le sauvegarde automatiquement.

## 8 - Menu A Propos



Fenêtre affichant les informations relatives à l'équipe de développement .

Retour :

- Retour au menu principal

# V. Contraintes

## A - Exigences Fonctionnelles

Pour la réalisation du projet, les exigences suivantes doivent être respectées :

- L'application doit être développée en ruby à l'aide du gem Gtk
- L'application doit correspondre en tout point au jeu Hashi avec un système de solution et d'indice associé au jeu
  - Système d'indice qui donne des techniques de jeu
  - Système d'hypothèse

## B- Exigences Clients

Les clients ont également des exigences concernant le rendu du projet :

- L'application doit être rendue pour le 23 Avril 2021 au plus tard
- L'application sera envoyée sous le format d'une image virtualbox
- L'application doit correspondre aux différentes exigences du cahier des charges
- L'application doit être un jeu du Hashi associé à un système de solution

## C- Les Risques

Pour le bon déroulement du projet, les risques suivants devront être évités au maximum :

- Retard pour le rendu de l'application qui peut être dû à:
  - Mauvaise organisation.
  - Participants ne faisant pas le travail demandé
  - Conflit entre les participants
- Mauvaise compréhension de la demande du client.
- Exigences du cahier des charges non respecté.
- Ajout de fonctionnalités non demandées par le client.

# VI. Divers

## A - Glossaire

Ruby :

Langage de Programmation Objet

Gtk :

Gem associé à ruby permettant de faire une interface graphique

Système de résolution :

Outil associé au jeu du Hashi qui permet à l'utilisateur de sortir de situation où il est bloqué. Il contient le système d'hypothèse et le système d'indice qui correspond aux techniques de jeu

Hypothèse :

Système qui permet à l'utilisateur de tenter différents coups. Le système lance un plateau superposé au plateau courant . S'il se rend compte qu'il est bloqué, il peut revenir au moment où il a lancé ce mode sur. Sinon le plateau du mode hypothèse devient alors le plateau courant

Plateau :

Correspond à la zone de jeu du Hashi pour l'utilisateur.