

Accueil PAM - TAKUZU . . . . .	2
Spécifications PAM - TAKUZU . . . . .	4
1_ACCUEIL . . . . .	5
ACC_CU01_Gérer la page d'accueil . . . . .	6
2_JEU . . . . .	7
JEU_CU01_Compléter la grille . . . . .	8
JEU_CU02_Réinitialiser la grille . . . . .	9
JEU_CU03_Vérifier la grille . . . . .	10
JEU_CU04_Valider la grille . . . . .	11
JEU_CU05_Aide à la saisie . . . . .	12
JEU_CU06_Affichage des règles . . . . .	13
JEU_CU07_Résolution de la grille . . . . .	14
3_RG . . . . .	15
ACC_01 - Page d'accueil . . . . .	16
JEU_01 - Page de jeu . . . . .	17
JEU_02 - Etat d'une case . . . . .	19
JEU_03 - Liste des messages . . . . .	20
JEU_04 - Affichage des valeurs d'une case . . . . .	21
JEU_05 - Pré-remplissage d'une grille . . . . .	22
JEU_06 - Règles du jeu . . . . .	23
JEU_07 - Valeur temporelle du chronomètre . . . . .	24
JEU_08 - Pop-up d'affichage des règles du jeu . . . . .	25
JEU_09 - Auto-complétions d'une case . . . . .	26
JEU_10 - Mémoire d'une case . . . . .	27
JEU_11 - Règles de Résolution . . . . .	29

## Accueil PAM - TAKUZU

Ceci est la page d'accueil du wiki pour le PAM - TAKUZU

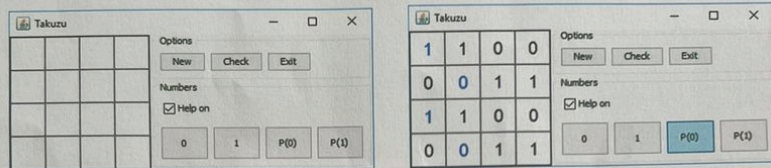
Objectif du sujet:

Énoncé du sujet:

### Projet PAM – IDI septembre 2024

#### Réalisation d'une application de TAKUZU

Il s'agit de réaliser une application de TAKUZU (ou BINERO) qui sera utilisée pour des tests de comportements d'agents logiciels. Ce PAM permettra au groupe projet de mettre en œuvre des connaissances apportées par les cours de Programmation, Conception des applications - UML, UX Design, Systèmes distribués et Intelligence artificielle.



L'application doit permettre à un utilisateur de remplir la grille suivant différents modes (notamment de vérification), mais aussi de réaliser un remplissage automatique (complet ou pas à pas).

Quelques spécifications techniques pour démarrer :

- ✓ L'application sera développée sous Python et Tkinter pour l'IHM,
- ✓ L'IHM sera conçue de façon à offrir à l'utilisateur au moins trois espaces : l'espace de jeu avec la grille, un espace de paramétrage selon les modalités de jeu et un espace de guide et de jeu automatique.
- ✓ Chaque case sera supervisée par un agent (thread), de façon à ce qu'elles soient autonomes (l'application sera donc distribuée)
- ✓ Les règles du jeu seront vérifiables pour guider l'utilisateur ; il faudra donc les représenter et les coder.

Le développement de l'application se fera de façon agile, mettant en œuvre au moins 4 sprints. A la fin de chaque sprint, selon l'avancée du développement de l'application des objectifs fermes seront discutés et validés pour le sprint suivant.

Ce projet contient une large dimension de créativité. Le groupe projet veillera cependant à bien respecter les spécifications et les contraintes du client, car celui-ci poursuivra le développement de l'application pour ses activités de recherche.



## Spécifications PAM - TAKUZU

# 1\_ACCUEIL

**Objectif :** L'objectif de ce cas d'utilisation est de décrire le comportement et l'affichage de la page d'accueil du jeu Takuzu depuis une interface graphique Tkinter.

**Pré-requis :** L'utilisateur déclenche l'ouverture de l'IHM de jeu Takuzu.

**Scénarii :**

- [ACC\\_CU01\\_Sélectionner taille de grille](#)

## ACC\_CU01\_Gérer la page d'accueil

Pré-requis : 

L'utilisateur a déclenché l'ouverture de l'interface du jeu Takuzu.

Description du scénario : 

### 1. Présentation :

La modale d'accueil s'affiche à l'ouverture du jeu telle que définie selon la règle suivante :  ACC\_01 - Page d'accueil


### 2. Actions disponibles :

L'utilisateur peut interagir avec la modale via plusieurs actions distinctes :


#### 2.1 Sélection de la taille de la grille :

- L'utilisateur saisi la taille de la grille en cochant l'une des options suivantes :
  - **6x6** : définit le paramètre N de la taille de la grille à N = 6.
  - **8x8** : définit le paramètre N de la taille de la grille à N = 8.
  - **10x10** : définit le paramètre N de la taille de la grille à N = 10.

#### 2.2 Sélection du chronomètre (facultative) :

- L'utilisateur peut choisir si la partie doit être chronométrée en cochant l'option "**Activer le chronomètre**" :
  - **Case à cocher "Activer le chronomètre"** : Désactivée par défaut. Si l'utilisateur coche la case, la valeur passe de `False` à `True`, activant ainsi le chronomètre pour la partie avec une valeur telle que définie selon la règle  JEU\_07 - Valeur temporelle du chronomètre.

#### 2.3 Bouton "Jouer" :

- L'utilisateur peut cliquer sur le bouton "**Jouer**" pour démarrer la partie.
- La modale d'accueil se ferme et l'application redirige vers la page de jeu telle que définie dans la règle  JEU\_01 - Page de jeu avec les paramètres sélectionnés.

#### 2.4 Icône de fermeture :

- L'utilisateur peut choisir de fermer la modale en cliquant sur l'icône de fermeture.
- La modale se ferme sans redirection vers la page de jeu.

### 3. Comportement en attente :

La modale reste visible tant que l'utilisateur n'a pas effectué l'une des actions décrites ci-dessus.

## 2\_JEU

**Objectif :** L'objectif de ce cas d'utilisation est de décrire le comportement et l'affichage de la page du jeu Takuzu depuis une interface graphique Tkinter.

**Pré-requis :** L'utilisateur a sélectionné une grille de taille **N** sur la page d'accueil.

**Scénarii :**

- JEU\_CU01\_Compléter la grille
- JEU\_CU02\_Réinitialiser la grille
- JEU\_CU03\_Vérifier la grille
- JEU\_CU04\_Valider la grille

## JEU\_CU01\_Compléter la grille

**Pré-requis :** L'utilisateur est présent sur la modale du jeu Takuzu.

**Description du scénario :**

- **1. Affichage de la grille :**

- **1.1** La grille présente sur la modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)
- **1.2** Chaque grille vierge est générée avec environ **25%** des cases pré-remplies par défaut, selon les règles définies [JEU\\_05 - Pré-remplissage d'une grille](#) et les modalités d'affichage décrites ici [JEU\\_04 - Affichage des valeurs d'une case](#)

- **2. Actions disponibles sur la grille :**

L'utilisateur peut cliquer sur les cases de la grille. Le comportement varie selon l'état de la case sélectionnée :

- **2.1** Si la case sélectionnée est à l'état '**vide**' selon la règle [JEU\\_02 - Etat d'une case](#) alors elle sera initialisée à '**0**' selon la règle d'affichage suivante [JEU\\_04 - Affichage des valeurs d'une case](#)
- **2.2** Si la case sélectionnée est à l'état '**0**' selon la règle [JEU\\_02 - Etat d'une case](#) alors elle sera initialisée à '**1**' selon la règle d'affichage suivante [JEU\\_04 - Affichage des valeurs d'une case](#)
- **2.1** Si la case sélectionnée est à l'état '**1**' selon la règle [JEU\\_02 - Etat d'une case](#) alors elle sera initialisée à '**vide**'.
- **2.1** Si la case fait partie des cases pré-remplies par défaut à l'**étape 1.2**, aucune action n'est disponible pour cette case.

- **3. En attente d'une action :**

La modale reste ouverte jusqu'à ce que l'utilisateur effectue une nouvelle action.



## JEU\_CU02\_Réinitialiser la grille

**Pré-requis :** L'utilisateur est présent sur la modale du jeu Takuzu.

**Description du scénario :**

- **1. Présentation :**

La modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)

- **2. Actions disponibles :**

L'utilisateur a cliqué sur le bouton "Réinitialiser" présent dans la modale :

- **2.1** Si aucune valeur n'a été saisie par l'utilisateur dans une case, celle-ci reste inchangée.
- **2.2** Si une valeur a été saisie par l'opérateur: La case est remise à son état initial (vide) tel que décrit dans la règle [JEU\\_02 - Etat d'une case](#).

- **3. Message de confirmation :**

Le message correspondant à l'ID 1 de la règle [JEU\\_03 - Liste des messages](#) s'affiche dans la zone prévue à cet effet sur la modale.

- **4. En attente d'une action :**

La modale reste ouverte jusqu'à ce que l'utilisateur effectue une nouvelle action.

## JEU\_CU03\_Vérifier la grille

**Pré-requis :** L'utilisateur est présent sur la modale du jeu Takuzu.

**Description du scénario :**

- **1. Présentation :**

La modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)

- **2. Actions disponibles :**

L'utilisateur a cliqué sur le bouton "**Vérifier**" présent dans la modale :

- **2.1** Si, après vérification selon la règle [JEU\\_06 - Règles du jeu](#), aucune erreur n'est détectée dans les valeurs saisies, alors le message correspond à l'**ID 6** de la règle [JEU\\_03 - Liste des messages](#) s'affiche dans la zone dédiée.
- **2.2** Si, après vérification selon la règle [JEU\\_06 - Règles du jeu](#), une erreur est détectée dans les valeurs saisies, alors le message pour lequel l'**ID\_Message\_Erreur** correspondant à l'ID de la règle [JEU\\_03 - Liste des messages](#) s'affiche dans la zone dédiée.

- **3. En attente d'une action :**

La modale reste ouverte jusqu'à ce que l'utilisateur effectue une nouvelle action (modification des valeurs, nouvelle vérification, etc.).

## JEU\_CU04\_Valider la grille

**Pré-requis :** L'utilisateur est présent sur la modale du jeu Takuzu.

**Description du scénario :**

- **1. Présentation :**

La modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)

- **2. Actions disponibles :**

L'utilisateur a cliqué sur le bouton "Valider" présent dans la modale :

- **2.1** Si, après vérification selon la règle [JEU\\_06 - Règles du jeu](#), aucune erreur n'est détectée dans les valeurs saisies, alors les messages correspondant à l'**ID 9** et **ID 11** (*optionnel*) de la règle [JEU\\_03 - Liste des messages](#) s'affiche dans la zone dédiée et le chronomètre s'arrête tel que défini selon la règle [JEU\\_07 - Valeur temporelle du chronomètre](#).
- **2.2** Si, après vérification selon la règle [JEU\\_06 - Règles du jeu](#), une erreur est détectée dans les valeurs saisies, alors le message correspond à l'**ID 10** de la règle [JEU\\_03 - Liste des messages](#) s'affiche dans la zone dédiée.
- **2.3** L'IHM renvoi vers la page d'accueil telle que décrite selon la règle [ACC\\_01 - Page d'accueil](#).

- **3. En attente d'une action :**

La modale reste ouverte jusqu'à ce que l'utilisateur effectue une nouvelle action.

## JEU\_CU05\_Aide à la saisie

**Pré-requis :** L'utilisateur est présent sur la modale du jeu Takuzu, et la grille est incomplète.

**Description du scénario :**

- **1. Présentation :**

La modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)

- **2. Actions disponibles :**

L'utilisateur a cliqué sur le bouton "**Aide**" présent dans la modale :

- **2.1** Si une case évidente peut être autocomplétée selon les règles [JEU\\_11 - Règles de Résolution](#) , alors la case est remplie automatiquement par la valeur correcte ( 0 ou 1 ) selon la règle d'auto-complétions suivante : [JEU\\_09 - Auto-complétions d'une case](#) .
  - **2.2** Si aucune case évidente ne peut être complétée, alors un message informatif correspondant à l'**ID11** selon la règle [JEU\\_03 - Liste des messages](#) s'affiche dans la zone prévue à cet effet.
- **3. Mise en évidence :**
    - **3.1** La mise en évidence de la case complétée reste active tant qu'aucune action utilisateur (*saisie manuelle, vérification, etc.*) n'est effectuée.
  - **4. En attente d'une action :**
    - **4.1** La modale reste ouverte et l'utilisateur peut continuer à interagir avec la grille ou utiliser d'autres options (*réinitialisation, vérification, validation*).

## JEU\_CU06\_Affichage des règles

**Pré-requis :** L'utilisateur est présent sur la modale du jeu Takuzu.

**Description du scénario :**

- **1. Présentation :**

La modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)

- **2. Actions disponibles :**

Si l'utilisateur clique sur le bouton "[?]" présent dans l'en-tête de la modale de jeu, alors :

- **2.1** Une pop-up d'affichage des règles s'affiche telle que définie selon la règle [JEU\\_08 - Pop-up d'affichage des règles du jeu](#)
- **2.2** . Si l'utilisateur clique sur le bouton "Fermer" alors la pop-up se ferme et l'IHM renvoi vers la partie en cours selon la règle : [JEU\\_01 - Page de jeu](#)

- **3. En attente d'une action :**

La pop-up d'affichage des règles reste ouverte jusqu'à ce que l'utilisateur effectue une nouvelle action.

## JEU\_CU07\_Résolution de la grille

**Pré-requis :** L'utilisateur est présent sur la page de jeu du Takuzu et la grille est incomplète.

**Description du scénario :**

- **1. Présentation :**

La modale de jeu s'affiche telle que définie selon la règle suivante : [JEU\\_01 - Page de jeu](#)

- **2. Actions disponibles :**

L'utilisateur clique sur le bouton "**Résolution**" :

- **2.1** Les threads en mémoire de chaque cases définis selon la règle [JEU\\_10 - Mémoire d'une case](#) identifient les valeurs évidentes selon les règles [JEU\\_11 - Règles de Résolution](#) .
- **2.2** Chaque case est remplie automatiquement par la valeur correcte ( 0 ou 1 ) selon la règle d'auto-complétions suivante : [JEU\\_09 - Auto-complétions d'une case](#) .
- **2.3** Une fois le processus de complétion terminé,
  - Si la grille est correctement complétée, alors un message informatif correspondant à l'**ID9** selon la règle [JEU\\_03 - Liste des m](#) [essages](#) s'affiche dans la zone prévue à cet effet.
  - Sinon, un message informatif correspondant à l'**ID11** selon la règle [JEU\\_03 - Liste des messages](#) s'affiche, signalant qu'il n'y a plus de résolution possible.
- **3. En attente d'une action :**
  - **3.1** La modale reste ouverte et l'utilisateur peut continuer à interagir avec la grille ou utiliser d'autres options (*réinitialisation, vérification, validation*).

## 3\_RG

**Objectif :** L'objectif de ce cas d'utilisation est de décrire les règles de gestions ainsi que les descriptions des écrans du jeu Takuzu.

**Scénarii :**

- ACC\_01 - Page d'accueil
- JEU\_01 - Page de jeu
- JEU\_02 - Etat d'une case
- JEU\_03 - Liste des messages
- JEU\_04 - Affichage des valeurs d'une case
- JEU\_05 - Pré-remplissage d'une grille
- JEU\_06 - Règles du jeu

## ACC\_01 - Page d'accueil

La structure de la modale d'accueil du jeu Takuzu est décrite ci-dessous :

Type		Valeur	Description
Titre principal		TAKUZU	Nom du jeu
Bouton		[Icône de fermeture]	Cliquable et déclenche la fermeture de la modale comme décrit à l' <b>étape 3</b> du scénario <a href="#">ACC_CU01_Gérer la page d'accueil</a>
	Titre	Sélectionner la taille de la grille :	
	Case à cocher	6x6	<ul style="list-style-type: none"> <li>Cochée par défaut.</li> </ul> Cliquable et déclenche l' <b>étape 2.1</b> du scénario <a href="#">ACC_CU01_Gérer la page d'accueil</a> avec comme paramètre de taille de grille N = 6.
	Case à cocher	8x8	Cliquable et déclenche l' <b>étape 2.1</b> du scénario <a href="#">ACC_CU01_Gérer la page d'accueil</a> avec comme paramètre de taille de grille N = 8.
	Case à cocher	10x10	Cliquable et déclenche l' <b>étape 2.1</b> du scénario <a href="#">ACC_CU01_Gérer la page d'accueil</a> avec comme paramètre de taille de grille N = 10.
	Titre	Activer le chronomètre	
	Case à cocher		<ul style="list-style-type: none"> <li>Décochée par défaut.</li> </ul> Cliquable et déclenche l' <b>étape 3</b> du scénario <a href="#">ACC_CU01_Gérer la page d'accueil</a> en initialisant la variable <i>timer_var</i> avec l'une des valeurs suivantes : <div>False</div> si décochée. <div>True</div> si cochée.
	Bouton	Jouer	Cliquable et déclenche l' <b>étape 2.3</b> du scénario <a href="#">ACC_CU01_Gérer la page d'accueil</a>



## JEU\_01 - Page de jeu

Type		Valeur	Description
Titre principal		TAKUZU - grille NxN	Nom du jeu avec précision de la taille N de la grille sélectionnée par l'utilisateur.
Bouton		[Accueil]	Cliquable et déclenche l'ouverture de la page d'accueil telle que définie selon la règle <a href="#">ACC_01 - Page d'accueil</a>
Bouton		[?]	Cliquable et déclenche l'ouverture de la pop-up d'affichage des règles du jeu définie selon le scénario <a href="#">JEU_CU06_Affichage des règles</a>
	Chronomètre	Valeur temporelle définit selon la règle <a href="#">JEU_07 - Valeur temporelle du chronomètre</a>	Affiche le temps écoulé depuis le début de la partie. <ul style="list-style-type: none"><li>Déclenché au lancement de la partie</li><li>Arrêté lorsque l'utilisateur clique sur “Valider”</li></ul>
	Grille	NxN	Grille de taille NxN, avec numérotation de 1 à N pour les lignes et colonnes.
	Bouton	Réinitialiser	Cliquable et déclenche la Réinitialisation des valeurs saisies par l'utilisateur tel que défini selon le scénario <a href="#">JEU_CU02_Réinitialiser la grille</a> .
	Bouton	Vérifier	<ul style="list-style-type: none"><li>Si la grille est incomplète : Cliquable et déclenche la Vérification des valeurs saisies par l'utilisateur tel que défini selon le scénario <a href="#">JEU_CU03_Vérifier la grille</a>.</li><li>Sinon : Non cliquable.</li></ul>
	Bouton	Valider	<ul style="list-style-type: none"><li>Si la grille est incomplète : Non cliquable.</li><li>Sinon : Cliquable et déclenche la Validation de la grille tel que définie selon le scénario <a href="#">JEU_CU04_Valider la grille</a></li></ul>
	Bouton	Aide	<ul style="list-style-type: none"><li>Si la grille est complète : Non cliquable.</li><li>Sinon : Cliquable et déclenche l'Aide à la saisie de l'opérateur telle que définie selon le scénario <a href="#">JEU_CU05_Aide à la saisie</a></li></ul>
	Bouton	Résolution	<ul style="list-style-type: none"><li>Si la grille est complète : Non cliquable.</li><li>Sinon : Cliquable et déclenche la résolution de la grille telle que définie selon le scénario <a href="#">JEU_CU07_Résolution de la grille</a></li></ul>
	Titre	Message	
	Zone de Texte	Message issu de la table <i>MESSAGE_ID</i>	Titre Message



## JEU\_02 - Etat d'une case

Ci-dessous se trouve la liste des différentes valeurs possibles pour une case.

Valeur	Description
'vide'	Valeur par défaut d'une case non saisie par l'utilisateur initialisée à '-1'
0	Valeur initialisée à '0'
1	Valeur initialisée à '1'

## JEU\_03 - Liste des messages

Ci-dessous se trouve la liste des messages ainsi que leurs ID présents dans la table *MESSAGE\_ID*.

ID	Message	Description
1	"La grille a été réinitialisée."	
2	"Il y a une erreur dans la colonne" + <code>str(col + 1)</code>	ID de la colonne correspondant à la variable " <code>str(row + 1)</code> " pour un nombre allant de 1 à N avec N = taille de la grille
3	"Il y a une erreur dans la ligne" + <code>str(row + 1)</code>	ID de la ligne correspondant à la variable " <code>str(row + 1)</code> " pour un nombre allant de 1 à N avec N = taille de la grille
4	"Il y a une erreur dans la colonne " + <code>str(col + 1)</code> + "le nombre de 0 et de 1 est différent"	ID de la colonne correspondant à la variable " <code>str(row + 1)</code> " pour un nombre allant de 1 à N avec N = taille de la grille
5	"Il y a une erreur dans la ligne " + <code>str(row + 1)</code> + "le nombre de 0 et de 1 est différent"	ID de la ligne correspondant à la variable " <code>str(row + 1)</code> " pour un nombre allant de 1 à N avec N = taille de la grille
6	"Il n'y a pas d'erreur"	
7	"Il y a une erreur, deux colonnes sont identiques"	
8	"Il y a une erreur, deux lignes sont identiques"	
9	"Félicitation, vous avez correctement complété la grille !"	
10	"Malheureusement, la grille est fausse vous devez recommencer !"	
11	"Aucune case évidente à compléter pour l'instant."	
12	"Temps final du chronomètre : {minutes}:{seconds}"	( <i>optionnel</i> ) message affiché ssi le chronomètre est actif

## JEU\_04 - Affichage des valeurs d'une case

Ci-dessous se trouve la description de l'affichage des valeurs d'une case en fonction de leurs états.

Pour les cases **initialisées par défaut**, les valeurs seront affichées comme indiqué ci-dessous :

Valeur	Description	Couleur
0	Valeur initialisée par défaut à '0'	'black'
1	Valeur initialisée par défaut à '1'	'black'

Pour les cases **saisies par l'utilisateur**, les valeurs seront affichées comme indiqué ci-dessous :

Valeur	Description	Couleur
0	Valeur initialisée à '0' par l'utilisateur	'green'
1	Valeur initialisée à '1' par l'utilisateur	'green'

Pour les cases **saisies par l'aide**, les valeurs seront affichées comme indiqué ci-dessous :

Si la case initialisée est correcte :

Valeur	Description	Couleur
0	Valeur initialisée à '0' par l'aide	'blue'
1	Valeur initialisée à '1' par l'aide	'blue'

Si la case initialisée était incorrecte :

Valeur	Description	Couleur
0	Valeur initialisée à '0' par l'aide	'red'
1	Valeur initialisée à '1' par l'aide	'red'

## JEU\_05 - Pré-remplissage d'une grille

### Objectif :

Le pré-remplissage automatique de la grille vise à faciliter le démarrage de la partie en respectant les règles fondamentales du jeu Takuzu. 25% des cases seront remplies aléatoirement avec les valeurs "1" et "0", tout en garantissant que l'opérateur dispose de combinaisons évidentes pour débiter sa résolution.

### Détails de la règle :

#### 1. Proportion de remplissage :

- 25% des cases de la grille doivent être pré-remplies avec des valeurs "1" et "0".

#### 2. Respect des règles de Takuzu :

Le pré-remplissage doit respecter les règles du jeu Takuzu afin de garantir que la grille soit valide dès le départ :

- **Pas plus de deux chiffres identiques consécutifs** : Aucune rangée ou colonne ne doit contenir plus de deux "1" ou "0" consécutifs.

#### 3. Facilitation de la résolution :

Le pré-remplissage doit être conçu pour offrir à l'opérateur des combinaisons évidentes dès le début, favorisant ainsi la résolution :

- **Détection des évidences** : La disposition des "1" et "0" doit permettre de repérer immédiatement certaines combinaisons évidentes, telles que :
  - Des séquences de deux "1" ou deux "0" déjà présentes, nécessitant automatiquement un "0" ou un "1" dans la case suivante pour respecter la règle "pas plus de deux chiffres identiques consécutifs".
  - Des séquences de deux "1" ou deux "0" déjà présentes et séparées par une case 'vide', nécessitant automatiquement un "0" ou un "1" dans la case vide pour respecter la règle "pas plus de deux chiffres identiques consécutifs".

#### 4. Grilles valides mais non triviales :

Le pré-remplissage doit rendre le début du jeu plus accessible sans pour autant éliminer le défi. La grille ne doit pas être résolue d'avance ou trop facile, mais les combinaisons initiales doivent orienter l'opérateur vers les premiers placements corrects.

#### 5. Aléatoire contrôlé :

Le pré-remplissage doit être pseudo-aléatoire, c'est-à-dire que la répartition des "1" et "0" sera générée aléatoirement tout en respectant les contraintes ci-dessus pour assurer une grille valide et équilibrée dès le départ.

## JEU\_06 - Règles du jeu

Ci-dessous se trouve la liste des différentes règles du TAKUZU.

ID	Description	ID_Message_Erreur
1	Il ne doit pas y avoir 3 cases identiques à la suite sur une colonne.	2
2	Il ne doit pas y avoir 3 cases identiques à la suite sur une ligne.	3
3	Deux colonnes ne peuvent pas être identiques.	7
4	Deux lignes ne peuvent pas être identiques.	8
5	Une colonne doit contenir le même nombre de 0 et de 1.	4
6	Une ligne doit contenir le même nombre de 0 et de 1.	5

## JEU\_07 - Valeur temporelle du chronomètre

**Description :** Cette règle de gestion décrit le format de la valeur temporelle du chronomètre ainsi que ses valeurs initiale et finale.

**Format de la valeur temporelle :**

- **Format :** MM:SS
  - **MM :** Minutes (00 à 59)
  - **SS :** Secondes (00 à 59)

**Valeur initiale :**

Le chronomètre commence à **00:00** au démarrage de la partie lorsque l'utilisateur clique sur le bouton "**Jouer**" tel que définit selon le scénario [☰ ACC\\_CU01\\_Gérer la page d'accueil](#) .

**Valeur finale :**

Le chronomètre s'arrête et affiche le temps total écoulé au moment de la validation lorsque l'utilisateur clique sur le bouton "**Valider**" tel que définit selon le scénario [☰ JEU\\_CU03\\_Vérifier la grille](#) .



## JEU\_08 - Pop-up d'affichage des règles du jeu



Ci-dessous se trouve la liste des différentes règles du TAKUZU disponible sur l'IHM du jeu pour l'utilisateur.

Type	Valeur	Description
Bouton	[Fermer]	Cliquable et déclenche la fermeture de la pop-up telle que définit à l' <b>étape 2.2</b> du scénario <a href="#">JEU_CU06_Affichage des règles</a>
Titre principal	Règles du jeu	
Liste		
	Il ne doit pas y avoir 3 cases identiques à la suite sur une colonne.	
	Il ne doit pas y avoir 3 cases identiques à la suite sur une ligne.	
	Deux colonnes ne peuvent pas être identiques.	
	Deux lignes ne peuvent pas être identiques.	
	Une colonne doit contenir le même nombre de 0 et de 1.	
	Une ligne doit contenir le même nombre de 0 et de 1.	

## JEU\_09 - Auto-complétions d'une case

**Description :** Cette règle de gestion décrit le comportement et l'affichage d'une case auto-complétée.

**Affichage de la case :**

- Lorsqu'une case est autocomplétée, elle est mise en évidence par une valeur affichée telle que définie dans la règle  JEU\_04 - Affich  
 .

**État de la case :**

- Une case autocomplétée reste **modifiable** par l'utilisateur, permettant une modification de la valeur si nécessaire.
- Ces cases sont traitées par l'IHM de la même manière que les saisies manuelles lors des vérifications et validations futures.

**Durée de la mise en évidence :**

- La mise en évidence reste visible tant qu'aucune action n'est réalisée par l'utilisateur. Cela inclut :
  - La saisie d'une autre case.
  - L'utilisation du bouton "**Vérifier**".
  - Une modification manuelle de la case autocomplétée.
- Une fois l'une de ces actions effectuée, la case reprend l'apparence standard des saisies utilisateur.

**Messages et retours visuels :**

- Si la case autocomplétée est modifiée par l'utilisateur, l'indicateur visuel disparaît, et la case est considérée comme modifiée manuellement.

## JEU\_10 - Mémoire d'une case

### Objectif :

Décrire la structure de la mémoire associée à chaque thread représentant une case de la grille Takuzu. Cette mémoire, mise à jour en continu, permet à chaque case d'apprendre son environnement immédiat (cases voisines) et d'évaluer sa propre valeur en respectant les règles du jeu.

### Description de la mémoire :

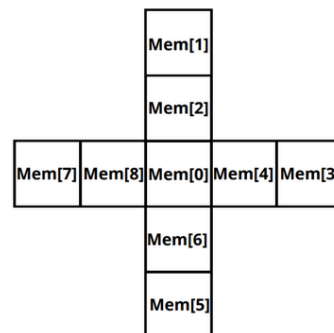
Chaque case est représentée par un thread indépendant disposant de sa propre mémoire structurée comme suit :

#### 1. Structure de la mémoire :

- `Mem[0]` : **Valeur de la case** (vide, 0 ou 1).
- `Mem[1]` à `Mem[8]` : **Valeurs des cases voisines** : (Cf image)
- `Mem[9]` : **Nombre total de 1 sur la ligne** contenant la case.
- `Mem[10]` : **Nombre total de 0 sur la ligne** contenant la case.
- `Mem[11]` : **Nombre total de 1 sur la colonne** contenant la case.
- `Mem[12]` : **Nombre total de 0 sur la colonne** contenant la case.
- `Mem[13]` : **Indice de la ligne** de la case (pour localisation dans la grille).
- `Mem[14]` : **Indice de la colonne** de la case (pour localisation dans la grille).

Mem = [Mem[0], Mem[1], Mem[2], Mem[3], Mem[4], Mem[5], Mem[6], Mem[7], Mem[8],  
Mem[9], Mem[10], Mem[11], Mem[12], Mem[13], Mem[14]]

Mem[0] = Valeur de la case.  
Mem[9] = Nombre total de 1 sur la ligne.  
Mem[10] = Nombre total de 0 sur la ligne.  
Mem[11] = Nombre total de 1 sur la colonne.  
Mem[12] = Nombre total de 0 sur la colonne.  
Mem[13] = Valeur de la ligne.  
Mem[14] = Valeur de la colonne.



### Extension de la mémoire pour une grille 10x10 [↗](#)

Pour une grille de jeu de taille 10x10, chaque case doit connaître l'ensemble des valeurs de sa ligne et de sa colonne afin de respecter les contraintes spécifiques à cette dimension élargie. Cela nécessite une mémoire de thread étendue par rapport aux autres grilles.

### Structure de la mémoire pour une grille 10x10 :

La structure de mémoire d'une case est composée des éléments suivants :

#### 1. Les 15 premières valeurs :

Identiques à la mémoire décrite pour les grilles standard :

- `Mem[0]` : Valeur de la case (vide, 0 ou 1).
- `Mem[1]` à `Mem[8]` : Valeurs des 8 cases voisines immédiates.

- `Mem[9]` à `Mem[12]` : Comptes des 1 et 0 sur la ligne et la colonne.
- `Mem[13]` et `Mem[14]` : Indices de ligne et de colonne.

## 2. Extension pour les grilles 10x10 :

Pour permettre à chaque case de connaître toutes les valeurs de sa ligne et de sa colonne, 36 valeurs supplémentaires sont ajoutées :

- **9 cases au-dessus dans la colonne :**

`Mem[15]` à `Mem[23]` : Valeurs des 9 cases situées au-dessus dans la même colonne, de haut en bas.

- **9 cases à droite dans la ligne :**

`Mem[24]` à `Mem[32]` : Valeurs des 9 cases situées à droite dans la même ligne, de gauche à droite.

- **9 cases en dessous dans la colonne :**

`Mem[33]` à `Mem[41]` : Valeurs des 9 cases situées en dessous dans la même colonne, de haut en bas.

- **9 cases à gauche dans la ligne :**

`Mem[42]` à `Mem[50]` : Valeurs des 9 cases situées à gauche dans la même ligne, de droite à gauche.

## Fonctionnement :

### 1. Initialisation :

- Lorsqu'une grille est générée ou réinitialisée, chaque thread initialise sa mémoire :
  - Les cases voisines sont identifiées (ou définies comme "null" si elles n'existent pas, par exemple pour une case en bordure).
  - Les compteurs de ligne et de colonne sont calculés (nombre de 0 et de 1).

### 2. Mise à jour continue :

- À chaque action de l'utilisateur ou résolution automatique, la mémoire de la case est mise à jour de façon indépendante :
  - Si une valeur est modifiée dans une case voisine, `Mem[1]` à `Mem[8]` sont ajustées.
  - Les compteurs de ligne et de colonne (`Mem[9]` à `Mem[12]`) sont recalculés en fonction des nouvelles valeurs.
- Cela garantit que chaque case dispose d'une vue cohérente et actualisée de son environnement.

# JEU\_11 - Règles de Résolution

## Objectif :

Ces règles déterminent comment chaque case de la grille décide de sa valeur en se basant sur les informations contenues dans sa mémoire (`Mem`) et les contraintes du Takuzu. Les décisions sont prises en respectant les règles fondamentales de résolution.

---

### 1. Résolution basée sur les valeurs voisines immédiates :

- **Règle 1** : Si les 2 valeurs au-dessus sont identiques, alors `Mem[0]` prend la valeur opposée.
  - **Règle 2** : Si les 2 valeurs à droite sont identiques, alors `Mem[0]` prend la valeur opposée.
  - **Règle 3** : Si les 2 valeurs en dessous sont identiques, alors `Mem[0]` prend la valeur opposée.
  - **Règle 4** : Si les 2 valeurs à gauche sont identiques, alors `Mem[0]` prend la valeur opposée.
- 

### 2. Résolution basée sur la colonne :

- **Règle 5** : Si les 2 valeurs de la colonne entourant la case sont identiques, alors `Mem[0]` prend la valeur opposée.
  - **Règle 9** : Si le nombre de `1` dans la colonne est atteint, alors `Mem[0] = 0`.
  - **Règle 10** : Si le nombre de `0` dans la colonne est atteint, alors `Mem[0] = 1`.
- 

### 3. Résolution basée sur la ligne :

- **Règle 6** : Si les 2 valeurs de la ligne entourant la case sont identiques, alors `Mem[0]` prend la valeur opposée.
  - **Règle 7** : Si le nombre de `1` dans la ligne est atteint, alors `Mem[0] = 0`.
  - **Règle 8** : Si le nombre de `0` dans la ligne est atteint, alors `Mem[0] = 1`.
- 

### 4. Cas spécifiques à la grille 10x10 :

- **Règle 27** :  
Si dans la ligne il y a 3 valeurs vides (dont `Mem[0]`), et :
  - Une de ces valeurs est voisine de `Mem[0]` et est égale à `0`.
  - Il reste un seul `1` dans la ligne.
  - **Alors l'autre valeur vide prend la valeur `0`.**
- **Règle 28** :  
Si dans la ligne il y a 3 valeurs vides (dont `Mem[0]`), et :
  - Une de ces valeurs est voisine de `Mem[0]` et est égale à `1`.
  - Il reste un seul `0` dans la ligne.
  - **Alors l'autre valeur vide prend la valeur `1`.**
- **Règle 29** :  
Si dans la colonne il y a 3 valeurs vides (dont `Mem[0]`), et :
  - Une de ces valeurs est voisine de `Mem[0]` et est égale à `0`.
  - Il reste un seul `1` dans la colonne.
  - **Alors l'autre valeur vide prend la valeur `0`.**
- **Règle 30** :  
Si dans la colonne il y a 3 valeurs vides (dont `Mem[0]`), et :

- Une de ces valeurs est voisine de `Mem[0]` et est égale à `1`.
  - Il reste un seul `0` dans la colonne.
  - Alors l'autre valeur vide prend la valeur `1`.
- 

#### 5. Résolution combinée avec des valeurs vides dans la colonne :

- **Règle 15** : Si les 2 valeurs en dessous et celle au-dessus sont vides, et qu'il reste un seul `1` dans la colonne :
    - Si la valeur 2 cases au-dessus est `0`, alors `Mem[0] = 1`.
  - **Règle 16** : Si les 2 valeurs en dessous et celle au-dessus sont vides, et qu'il reste un seul `1` dans la colonne :
    - Si la valeur 2 cases au-dessus est `1`, alors les valeurs en dessous et 2 au-dessus prennent `0`.
  - **Règle 17** : Si les 2 valeurs en dessous et celle au-dessus sont vides, et qu'il reste un seul `0` dans la colonne :
    - Si la valeur 2 cases au-dessus est `0`, alors les valeurs en dessous et 2 au-dessus prennent `1`.
  - **Règle 18** : Si les 2 valeurs en dessous et celle au-dessus sont vides, et qu'il reste un seul `0` dans la colonne :
    - Si la valeur 2 cases au-dessus est `1`, alors `Mem[0] = 0`.
- 

#### 6. Résolution combinée avec des valeurs vides dans la ligne :

- **Règle 23** : Si les 2 valeurs à gauche et celle à droite sont vides, et qu'il reste un seul `1` dans la ligne :
  - Si la valeur 2 cases à droite est `0`, alors `Mem[0] = 1`.
- **Règle 24** : Si les 2 valeurs à gauche et celle à droite sont vides, et qu'il reste un seul `1` dans la ligne :
  - Si la valeur 2 cases à droite est `1`, alors les valeurs à droite et 2 à gauche prennent `0`.
- **Règle 25** : Si les 2 valeurs à gauche et celle à droite sont vides, et qu'il reste un seul `0` dans la ligne :
  - Si la valeur 2 cases à droite est `0`, alors les valeurs à droite et 2 à gauche prennent `1`.
- **Règle 26** : Si les 2 valeurs à gauche et celle à droite sont vides, et qu'il reste un seul `0` dans la ligne :
  - Si la valeur 2 cases à droite est `1`, alors `Mem[0] = 0`.