S2.01 Développement d'une application

Répertoire GitHub : https://github.com/eetchever004/Chifoumi

ETCHEVERRY Eliott

TD1 - TP2

GOUAUD Romain

BUT S2 2021-2022

LABASTIE Esteban

SOMMAIRE :

Saé 2.01 – Développement d'une application

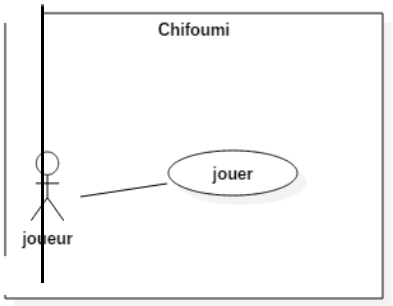
Chifoumi – Dossier d'Analyse et conception

1. Compléments de spécifications externes.

On précise **uniquement** les points qui vous ont semblé flous ou bien incomplets. Rien de plus à signaler dans cette étude.

1.1

2. Diagramme des Cas d'Utilisation



1.2

Figure 1 : Diagramme des Cas d'Utilisation du jeu Chifoumi

3. Scénarios

(a) Exemple Scénario

Cas d'utilisation	**JOUER**	
Résumé	Le joueur joue une partie.	
Acteur primaire	Joueur	
Système	Chifoumi	
Intervenants		
Niveau	Objectif utilisateur	
Préconditions	Le jeu est démarré et se trouve à l'état initial.	
Postconditions		
Date de création		
Date de mise à jour		
Créateur		
Opérations	**Joueur**	**Système**
1	Démarre une nouvelle partie.	
2		Rend les figures actives et les affiche actives.
3	Choisit une figure.	
4		Affiche la figure du joueur dans la zone d'affichage du dernier coup joueur.
5		Choisit une figure.
6		Affiche sa figure dans la zone d'affichage de son dernier coup.
7		Détermine le gagnant et met à jour les scores.
8		Affiche les scores. Retour à l'étape 3.
Extension		
3.A	**Le joueur demande à jouer une nouvelle partie.**	
3.A.1	Choisit une nouvelle partie	
3.A.2		Réinitialise les scores.
3.A.3		Réinitialise les zones d'affichage des derniers coups.
3.A.4		Retour à l'étape 3.

Tableau 1 : Scénario nominal

(b) Remarques :

- *Le scénario est très simple.*
- *L'objectif est de mettre en évidence les actions de l'utilisateur, celles du système, sachant que ces actions sont candidates à devenir des méthodes du système*

1.3

4. Diagramme de classe (UML)

(a) Le diagramme de classes UML du jeu se focalise sur les classes **métier**, cad celles décrivant le jeu indépendamment des éléments d'interface que comportera le programme.

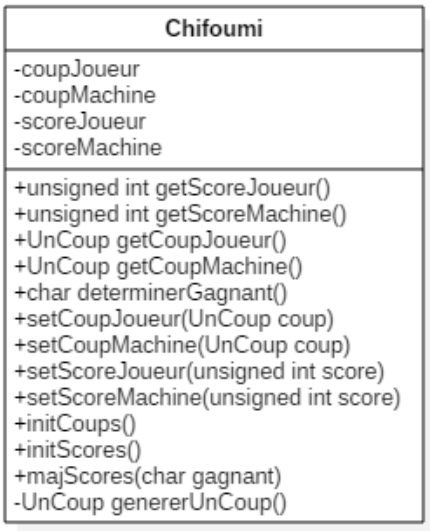


Figure 2 : Diagramme de Classes UML du jeu Chifoumi

(b) Dictionnaire des éléments de la **Classe Chifoumi**

Nom attribut	Signification	Type	Exemple
scoreJoueur	Nbre total de points acquis par le joueur durant la partie courante	unsigned int	1
scoreMachine	Nbre total de points acquis par la machine durant la partie courante	unsigned int	1
coupJoueur	Mémorise la dernière figure choisie par le joueur. Type énuméré enum unCoup {pierre, ciseau, papier, rien};	UnCoup	papier
coupMachine	Mémorise la dernière figure choisie par la machine.	UnCoup	Ciseau

Tableau 2 : Dictionnaire des éléments - Classe Chifoumi

(c) Dictionnaire des méthodes : intégrées dans l'interface de la classe : cf Figure 3

```
using namespace std;
class Chifoumi
{
   ///* ---- PARTIE MODèLE ----------------------------
      ///* Une définition de type énuméré
   public:
      enum UnCoup {pierre, papier, ciseau, rien};

      ///* Méthodes publiques du Modèle
   public:
      Chifoumi();
      virtual ~Chifoumi();

      // Getters
      UnCoup getCoupJoueur();
         /* retourne le dernier coup joué par le joueur */
      UnCoup getCoupMachine();
         /* retourne le dernier coup joué par le joueur */
      unsigned int getScoreJoueur();
         /* retourne le score du joueur */
      unsigned int getScoreMachine();
         /* retourne le score de la machine */
      char determinerGagnant();
         /* détermine le gagnant 'J' pour joueur, 'M' pour machine, 'N' pour match nul
            en fonction du dernier coup joué par chacun d'eux */

       ///* Méthodes utilitaires du Modèle
   private :
      UnCoup genererUnCoup();
   /* retourne une valeur aléatoire = pierre, papier ou ciseau.
      Utilisée pour faire jouer la machine */

      // Setters
   public:
      void setCoupJoueur(UnCoup p_coup);
         /* initialise l'attribut coupJoueur avec la valeur
            du paramètre p_coup */
      void setCoupMachine(UnCoup p_coup);
          /* initialise l'attribut coupmachine avec la valeur
            du paramètre p_coup */
      void setScoreJoueur(unsigned int p_score);
         /* initialise l'attribut scoreJoueur avec la valeur
            du paramètre p_score */
      void setScoreMachine(unsigned int p_score);
          /* initialise l'attribut coupMachine avec la valeur
            du paramètre p_score */

      // Autres modificateurs
       void majScores(char p_gagnant);
         /* met à jour le score du joueur ou de la machine ou aucun
            en fonction des règles de gestion du jeu */
      void initScores();
         /* initialise à 0 les attributs scoreJoueur et scoreMachine
            NON indispensable */
      void initCoups();
         /* initialise à rien les attributs coupJoueur et coupMachine
            NON indispensable */

       ///* Attributs du Modèle
    private:
      unsigned int scoreJoueur;   // score actuel du joueur
      unsigned int scoreMachine;  // score actuel de la Machine
      UnCoup coupJoueur;          // dernier coup joué par le joueur
      UnCoup coupMachine;         // dernier coup joué par la machine
};
```

Figure 3 : Schéma de classes = Une seule classe Chifoumi

(d) Remarques concernant le schéma de classes

1. On ne s'intéresse qu'aux attributs et méthodes métier. Notamment, on ne met pas, pour l'instant, ce qui relève de l'affichage car ce sont d'autres objets du programme (widgets) qui se chargeront de l'affichage. Par contre, on n'oublie pas les méthodes getXXX(), qui permettront aux objets métier de communiquer leur valeur aux objets graphiques pour que ceux-ci s'affichent.
2. On n'a mis ni le constructeur ni le destructeur, pour alléger le schéma.
3. D'autres attributs et méthodes viendront compléter cette vision ANALYTIQUE du jeu. Il s'agira des attributs et méthodes dits DE CONCEPTION nécessaires au développement de l'application.

1.3.1

5. Implémentation et tests

5.1 Implémentation
Liste des fichiers de cette version :
- chifoumi.h : déclaration des variables et des sous-programmes du chifoumi
- chifoumi.cpp : définition des corps des sous programmes du chifoumi
Respectivement spécification et corps de la classe Chifoumi décrite au paragraphe 4.

5.2 Test

Test avec le programme fourni main.cpp

coupJoueur	coupMachine	Résultat attendu
pierre	ciseau	Incrémenter scoreJoueur
papier	ciseau	Incrémenter scoreMachine
ciseau	ciseau	Aucun changement de score
pierre	papier	Incrémenter scoreMachine
papier	papier	Aucun changement de score
ciseau	papier	Incrémenter scoreJoueur
pierre	pierre	Aucun changement de score
papier	pierre	Incrémenter scoreJoueur
ciseau	pierre	Incrémenter scoreMachine

6. Classe Chifoumi : Diagramme états-transitions

(a) Diagramme états-transitions -actions du jeu

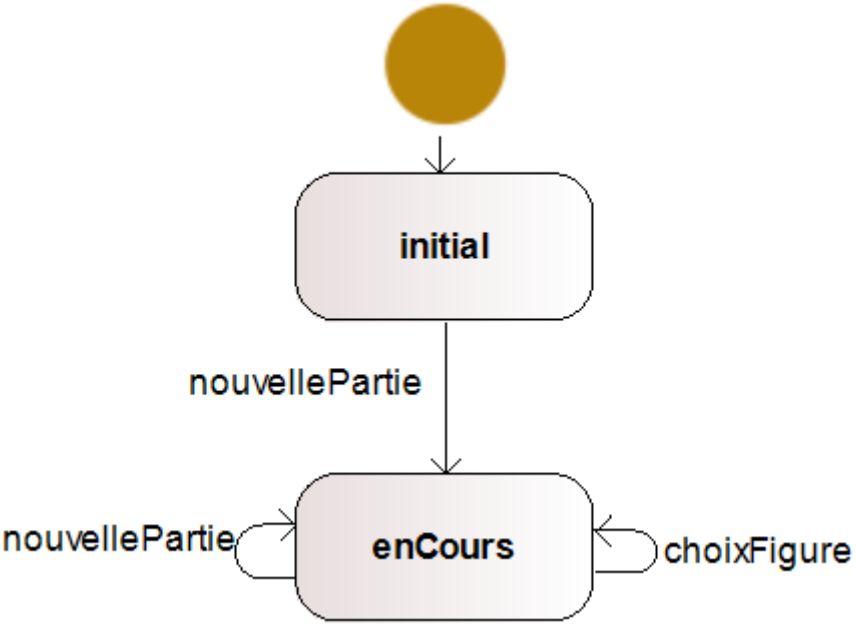


Figure 4 : Diagramme états-transitions

(b) Dictionnaires des états, événements et Actions

Dictionnaire des états du jeu

nomEtat	*Signification*
initial	Etat de début d'une partie, les boutons de figure sont inactifs
enCours	Etat lorsqu'une partie est en cours, les boutons de figure sont actifs

Tableau 3 : États du jeu

Dictionnaire des événements faisant changer le jeu d'état

nomEvénement	*Signification*
choixFigure	Le joueur choisit une figure à jouer en appuyant sur un des boutons figure
nouvellePartie	Une nouvelle partie est démarrée en appuyant sur le bouton de nouvelle partie

Tableau 4 : Evénements faisant changer le jeu d'état

Description des actions réalisées lors de la traversée des transitions

Activation des boutons de figure	Les boutons de figures sont activés afin de permettre au joueur de les presser
Jouer un coup	Le joueur joue soit la pierre, soit la feuille, soit le ciseau, la machine choisie au hasard sa figure. Un gagnant est déterminé et les scores sont mis à jour, sur l'interface graphique, les figures des coups des joueurs sont affichées.
Initialiser une partie	Remise à zéro des scores, les figures des coups des joueurs ne sont plus affichées

Tableau 5 : Actions à réaliser lors des changements d'état

(c) Préparation au codage :

Table correspondant à la version matricielle du diagramme états-transitions du jeu :
- en *ligne : les **événements*** faisant changer le jeu d'état
- en *colonne : les **états*** du jeu

Événement / *nomEtatJeu*	choixFigure	nouvellePartie
Initial	--	Activation des boutons de figure
En cours	Jouer un coup	Initialiser une partie

Tableau 6 : Matrice d'états-transitions du jeu chifoumi

7. Éléments d'interface

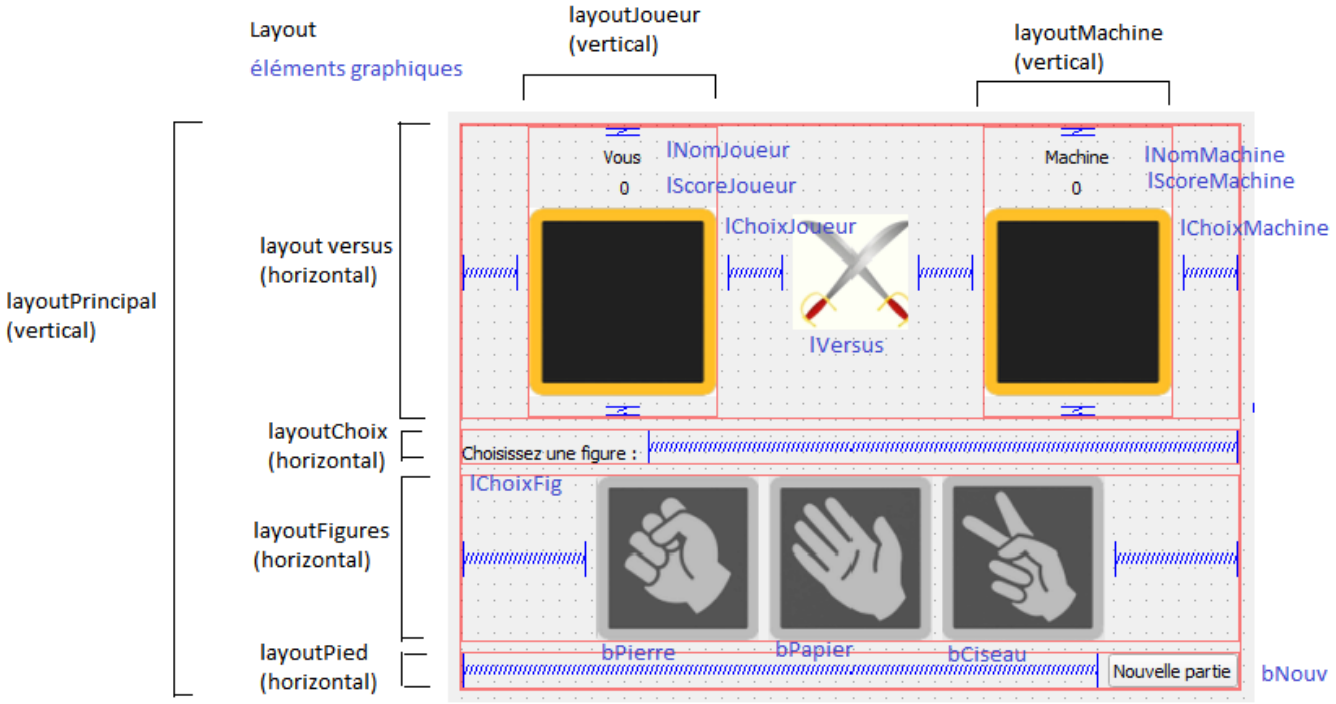


Figure 5 : éléments d'interface

8. Implémentation et tests

8.1 Implémentation
Liste des fichiers de cette version :
- chifoumivue.h : déclaration des variables et des sous-programmes du chifoumi
- chifoumivue.cpp : définition des corps des sous programmes du chifoumi
- chifoumivue.ui : fichier graphique QtDesigner
- main.cpp : fichier main de l'application

4 connexion :
3 pour jouer chaque coup avec comme signal le clic de la souris et comme slot les boutons de figure
1 pour débuter une nouvelle partie avec comme signal le clic de la souris et comme slot le bouton de nouvelle partie

8.2 Test

Test : état initial

Activité	Résultat attendu	Résultat obtenu	Test passé
Choix pierre	Rien	Rien	Passé
Choix papier	Rien	Rien	Passé
Choix ciseaux	Rien	Rien	Passé
Choix nouvelle partie	Déblocage des boutons de figure	Déblocage des boutons de figure	Passé

Tableau 6 : Tests de l'état initial

Test : état enCours

Activité	**Résultat attendu**	**Résultat obtenu**	**Test passé**
Choix nouvelle partie	Réinitialisation des score Changement des affichages joueur et machine par l'image par défaut	Réinitialisation des score Changement des affichages joueur et machine par l'image par défaut	Passé
Choix joueur : pierre Choix machine : pierre	Affichage de 2 pierres Pas d'incrémentation des scores	Affichage de 2 pierres Pas d'incrémentation des scores	Passé
Choix joueur pierre Choix machine : papier	Affichage d'une pierre côté joueur Affichage d'un papier côté machine Incrémentation du score machine Pas d'incrémentation pour le joueur	Affichage d'une pierre côté joueur Affichage d'un papier côté machine Incrémentation du score machine Pas d'incrémentation pour le joueur	Passé
Choix joueur pierre Choix machine : ciseaux	Affichage d'une pierre côté joueur Affichage d'un ciseau côté machine Pas d'incrémentation du score machine Incrémentation du score pour le joueur	Affichage d'une pierre côté joueur Affichage d'un ciseau côté machine Pas d'incrémentation du score machine Incrémentation du score pour le joueur	Passé
Choix joueur papier Choix machine : pierre	Affichage d'un papier côté joueur Affichage d'une pierre côté machine Pas d'incrémentation du score machine Incrémentation du score pour le joueur	Affichage d'un papier côté joueur Affichage d'une pierre côté machine Pas d'incrémentation du score machine Incrémentation du score pour le joueur	Passé
Choix joueur papier Choix machine : papier	Affichage de 2 papiers Pas d'incrémentation des scores	Affichage de 2 papiers Pas d'incrémentation des scores	Passé
Choix joueur papier Choix machine : ciseaux	Affichage d'un papier côté joueur Affichage d'un ciseau côté machine	Affichage d'un papier côté joueur Affichage d'un ciseau côté machine	Passé

	Incrémentation du score machine Pas d'incrémentation pour le joueur	Incrémentation du score machine Pas d'incrémentation pour le joueur	
Choix joueur ciseaux Choix machine : pierre	Affichage d'un ciseau côté joueur Affichage d'une pierre côté machine Incrémentation du score machine Pas d'incrémentation pour le joueur	Affichage d'un ciseau côté joueur Affichage d'une pierre côté machine Incrémentation du score machine Pas d'incrémentation pour le joueur	Passé
Choix joueur ciseaux Choix machine : papier	Affichage d'un ciseau côté joueur Affichage d'un papier côté machine Pas d'incrémentation du score machine Incrémentation du score pour le joueur	Affichage d'un ciseau côté joueur Affichage d'un papier côté machine Pas d'incrémentation du score machine Incrémentation du score pour le joueur	Passé
Choix joueur ciseaux Choix machine : ciseaux	Affichage de 2 ciseaux Pas d'incrémentation des scores	Affichage de 2 ciseaux Pas d'incrémentation des scores	Passé

Tableau 7 : Tests de l'état enCours

Version 2

v2 effectuée sur la v1

1. Liste des fichiers sources

Fichiers	Rôle
chifoumi.pro	Fichier projet, répertoriant les fichier du chifoumi
chifoumi.cpp	Corps du modèle (les traitement effectués pour le fonctionnement du jeu)
chifoumi.h	Entête du modèle (les traitement effectués pour le fonctionnement du jeu)
chifoumipresentation.cpp	Corps de la présentation (Fait le liens entre le modèle et la vue)
chifoumipresentation.h	Entête de la présentation (Fait le liens entre le modèle et la vue)
chifoumivue.cpp	Corps de la vue (Ce qui est affiché à l'utilisateur)
chifoumivue.h	Entête de la vue (Ce qui est affiché à l'utilisateur)
chifoumievue.ui	Fichier Qt Designer de la vue
main.cpp	Programme principal
ressourcesChifoumi.qrc	Fichier ressource contenant les images nécessaire lors de l'affichage du chifoumi

Tableau 8 : Liste des fichiers sources v2

2. **Présentation des nouveaux fichiers .h**
   Le fichier chifoumivue.h contient toutes les méthodes apportant des modifications graphiques : changements d'images, mise à jour des labels, boutons,...

   Le fichier chifoumi.h contient toutes les données du jeu : score et coup du joueur, score et coup de la machine, … Ainsi que les méthodes permettant de les manipuler : getters et setters

   Le fichier chifoumipresentation.h contient tous les traitements du jeu : choix des coups, affichage des fenêtres des versions futures, fonctionnalités du menu déroulant.

   L'utilisation de ce modèle MVP permet d'organiser le code et d'éviter d'avoir à changer le modèle ou la présentation en cas de changements graphiques.

3. **Explication des points importants des CPP**
   Mise en oeuvre du modèle MVP

4. **Résultats des tests propres à la version**

Même tests que la v1

Version 3

1. **Liste des fichiers sources**

Fichiers	Rôle
chifoumi.pro	Fichier projet, répertoriant les fichier du chifoumi
chifoumi.cpp	Corps du modèle (les traitement effectués pour le fonctionnement du jeu)
chifoumi.h	Entête du modèle (les traitement effectués pour le fonctionnement du jeu)
chifoumipresentation.cpp	Corps de la présentation (Fait le lien entre le modèle et la vue)
chifoumipresentation.h	Entête de la présentation (Fait le lien entre le modèle et la vue)
chifoumivue.cpp	Corps de la vue (Ce qui est affiché à l'utilisateur)
chifoumivue.h	Entête de la vue (Ce qui est affiché à l'utilisateur)
chifoumievue.ui	Fichier Qt Designer de la vue
main.cpp	Programme principal
ressourcesChifoumi.qrc	Fichier ressource contenant les images nécessaire lors de l'affichage du chifoumi

Tableau 9 : Liste des fichiers sources v3

2. **Présentation des nouveaux fichiers .h**
   Aucun nouveau fichier pour cette v3

3. **Explication des points importants des CPP**
   /

4. **Résultats des tests propres à la version**

Tests interface

Activité	**Résultat attendu**	**Résultat obtenu**	**Test passé**
A Propos (f1)	affichage du MessageBox depuis la barre de menu ou le raccourci	affichage du MessageBox depuis la barre de menu ou le raccourci	Passé
Fichier >> Quitter (alt f3)	fermeture de l'application depuis le menu déroulant ou le raccourci	fermeture de l'application depuis le menu déroulant ou le raccourci	Passé

Tableau 10 : Tests de l'interface v3

Version 4

1. Diagramme état-transition

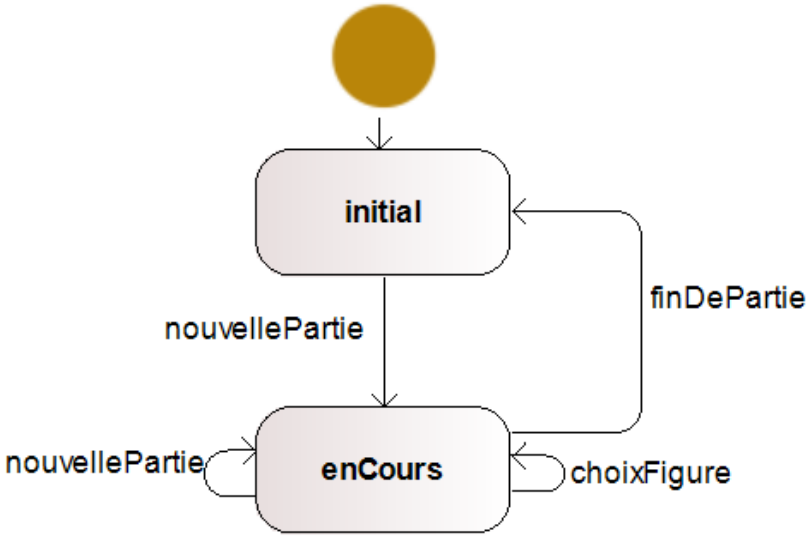


Figure 6 : état-transition v4

2. Dictionnaire d'événements, d'actions et d'état associés

Dictionnaire des états du jeu

nomEtat	*Signification*
initial	Etat de début d'une partie, les boutons de figure sont inactifs
enCours	Etat lorsqu'une partie est en cours, les boutons de figure sont actifs

Tableau 11 : États du jeu

Dictionnaire des événements faisant changer le jeu d'état

Nous avons pensé à ajouter un troisième état pour la fin de la partie mais nous nous sommes éloignés de cette idée car cet état n'aurait pas été utile au déroulement du jeu.

nomEvénement	*Signification*
choixFigure	Le joueur choisit une figure à jouer en appuyant sur un des boutons figure
nouvellePartie	Une nouvelle partie est démarrée en appuyant sur le bouton de nouvelle partie
finDePartie	Affiche un message lorsque la limite de score est atteinte

Tableau 12 : Evénements faisant changer le jeu d'état

Description des actions réalisées lors de la traversée des transitions

Activation des boutons de figure	Les boutons de figures sont activés afin de permettre au joueur de les presser
Jouer un coup	Le joueur joue soit la pierre, soit la feuille, soit le ciseau, la machine choisie au hasard sa figure. Un gagnant est déterminé et les scores sont mis à jour, sur l'interface graphique, les figures des coups des joueurs sont affichées.

Initialiser une partie	Remise à zéro des scores, les figures des coups des joueurs ne sont plus affichées
Affichage message fin de partie	Affichage du message personnalisé contenant le nom du joueur et son score en fin de partie.

Tableau 13 : Actions à réaliser lors des changements d'état

3. Version matricielle du diagramme états-transitions + identification des éléments d'interface supplémentaires éventuellement nécessaires

elementinterface	*bPierre*	*bCiseau*	*bPapier*	*bNouv*	---
Événement / nomEtatJeu	**choixFigure**			**nouvellePartie**	**finDePartie**
initial	---			enCours / actvité3	---
enCours	enCours / actvité1			enCours / actvité2	initial / activité4
actvité3	mettre les scores à 0				
	mettre les coups à rien				
	afficher ces éléments de jeu sur l'interface				
	rendre les boutons des figures actifs				
	mettre le focus sur bNouv				
actvité2	mettre les scores à 0				
	mettre les coups à rien				
	afficher ces éléments de jeu sur l'interface				
	mettre le focus sur bNouv				
actvité1	mettre le coup du joueur à la figure choisie				
	faire jouer la machine				
	déterminer le gagnant				
	mettre à jour les scores en fonction du gagnant				
	affiche ces éléments de jeu sur l'interface				
	mettre le focus sur boutonNvllePartie				
	vérifier si la partie est terminée				
activité4	afficher le message personnalisé de fin de partie				
	griser les boutons de figures				

Tableau 14 : états-transitions matriciel v4

4. Décrire les nouveaux éléments d'interface

5. Liste des fichiers sources

Fichiers	Rôle
chifoumi.pro	Fichier projet, répertoriant les fichier du chifoumi
chifoumi.cpp	Corps du modèle (les traitement effectués pour le fonctionnement du jeu)

chifoumi.h	Entête du modèle (les traitement effectués pour le fonctionnement du jeu)
chifoumipresentation.cpp	Corps de la présentation (Fait le lien entre le modèle et la vue)
chifoumipresentation.h	Entête de la présentation (Fait le lien entre le modèle et la vue)
chifoumivue.cpp	Corps de la vue (Ce qui est affiché à l'utilisateur)
chifoumivue.h	Entête de la vue (Ce qui est affiché à l'utilisateur)
chifoumievue.ui	Fichier Qt Designer de la vue
main.cpp	Programme principal
ressourcesChifoumi.qrc	Fichier ressource contenant les images nécessaire lors de l'affichage du chifoumi

Tableau 15 : Liste des fichiers sources v4

6. **Présentation des fichiers .h**
   Aucun nouveau fichier pour cette v4

7. **Explication des points importants des CPP**
   La limite de score est enregistrée dans le modèle est est à 5 par défaut

8. **Résultats des tests propres à la version**

Tests interface

Activité	**Résultat attendu**	**Résultat obtenu**	**Test passé**
fin de partie	affichage de la fenêtre de fin de partie et passage à l'état initial avec maj des éléments du jeu	affichage de la fenêtre de fin de partie et passage à l'état initial avec maj des éléments du jeu	Passé

Tableau 16 : Tests interface v4