

Saé 2.01 – Développement d'une application

Chifoumi – Dossier d'Analyse et conception v1

Sommaire :

Version initiale :

- 1-Compléments de spécifications externe
- 2-Diagramme des cas d'utilisations
- 3-Scénarios
- 4-Diagramme de classe UML

Version 0 :

- 1-Liste des fichiers sources
- 2-Tests des méthodes

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.

2-Diagramme des Cas d'Utilisation

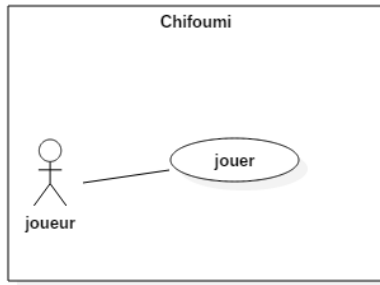


Figure 1 : Diagramme des Cas d'Utilisation du jeu Chifoumi

3-Scénarios

(a) Exemple Scénario

Titre : Jouer 2 coups
Résumé : Le joueur démarre une partie et joue 2 coups
Acteur : Utilisateur (acteur principal)
Pré-condition : Le jeu est démarré et se trouve à l'état initial.
Post-condition : néant
Date de création : 01/05/2021
Créateur : ...

Date de mise à jour : --

Utilisateur	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.
9. Choisit une figure	10. Affiche la figure du joueur dans la zone d'affichage du dernier coup joueur.
	11. Choisit une figure.
	12. Affiche sa figure dans la zone d'affichage de son dernier coup.
	13. Détermine le gagnant et met à jour les scores.
	14. Affiche les scores.
15. Choisit une Nouvelle Partie	16. Réinitialise les scores.
	17. Réinitialise les zones d'affichage des derniers coups.
	19. Affiche les scores et les zones d'affichage des derniers coups.

Tableau 1 : Scénario nominal

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.

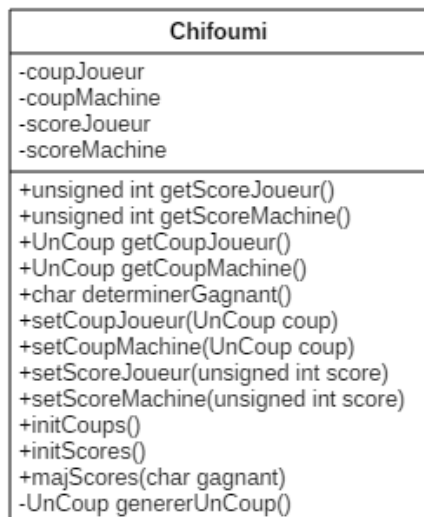


Figure 2 : Diagramme de Classes UML du jeu Chifoumi

(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	Mémoire la dernière figure choisie par le joueur. Type énuméré enum unCoup {pierre, ciseau, papier, rien};	UnCoup	papier
coupMachine	Mémoire 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 scoreMachine 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

Version 0 :

1-Liste des fichiers sources de cette version (et rôle de chacun)

Chifoumi.h : interface de l'application non graphique Chifoumi

```
    /** Une définition de type énuméré
public:
    enum UnCoup {pierre, papier, ciseau, rien};
    /** Méthodes 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);
        /* Mise à jour des scores en fonction des règles de gestion actuelles :
        - 1 point pour le gagnant lorsqu'il y a un gagnant
        - 0 point en cas de match nul */
    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
```

Chifoumi.cpp : corps de l'application non graphique Chifoumi

Main.cpp : fichier source contenant la boucle principale qui exécute l'application non graphique et la boucle secondaire qui se met en attente de évènements

V0.pro : fichier contenant les chemins d'inclusions et l'arborescence des fichiers

Chifoumi_dossierAnalyseConception_V0 : dossier d'analyse & conception

2-Tests des méthodes :

Test	Résultat	Validation
Méthodes get() associées aux attributs 'score'	Score Joueur : 0 score Machine : 0	Ok
Méthodes get() associées aux attributs 'coup'	Coup Joueur : Rien Coup Machine : rien	Ok
Méthodes set() associées aux attributs 'score'	Score Joueur : 1 Score Machine : 2	Ok
Méthode initScores()	Score Joueur : 0 Score Machine : 0	Ok
Méthodes set() et get() associées aux attributs 'coup'/'choix'	Coup Joueur : pierre Coup Machine : ciseau	Ok
Quelques tours de jeu pour tester l'identification du gagnant et la maj des scores	Coup Joueur : pierre Coup Machine : ciseau Score Joueur : 1 Score Machine : 0	Ok
Appel du Constructeur	Scores à 0 CoupsJoueurs à rien	Ok

