

2021 - 2022

Version 2

S2.01 - Développement d'une application

Nicolas DARGAZANLI

Alexandre MAURICE

Guillaume TRITSCH

Table des matières

[Présentation de la version 3](#_Toc104138138)

[Détails des fichiers 3](#_Toc104138139)

[main.cpp 3](#_Toc104138140)

[modele.h 3](#_Toc104138141)

[modele.cpp 3](#_Toc104138142)

[presentation.h 3](#_Toc104138143)

[presentation.cpp 3](#_Toc104138144)

[chifoumivue.h 3](#_Toc104138145)

[chifoumivue.cpp 3](#_Toc104138146)

[chifoumivue.ui 3](#_Toc104138147)

[/Image 4](#_Toc104138148)

[Réalisation du modèle MVP 4](#_Toc104138149)

[Présentation 4](#_Toc104138150)

[Vue 4](#_Toc104138151)

[Test de la version 5](#_Toc104138152)

# Présentation de la version

Cette version consiste à réaliser une version du programme basé sur la version v1, en utilisant le modèle MVP.

# Détails des fichiers

## main.cpp

Programme permettant de vérifier le bon fonctionnement de l’interface et le corps des fichier intitulé chifoumi. Ce fichier est un fichier de test fournis lors de la présentation de la SAE.

## modele.h

Ce fichier est l’interface de la classe chifoumi qui met à disposition la gestion d’une partie de chifoumi entre un joueur et l’ordinateur.

Cela comprend donc des attributs de classe gérant le score et le coup de chaque joueur, et des méthodes pour générer un coup, les comparer, gérer les scores, …

## modele.cpp

Ce fichier est le corps de la classe chifoumi. Elle contient le corps de chaque méthode de la classe.

## presentation.h

Ce fichier comporte l’interface de la classe Presentation qui gère la relation entre le modèle et la vue.

## presentation.cpp

Ce fichier comporte les définitions des sous programmes de la classe Presentation.

## chifoumivue.h

Ce fichier contient l’interface de la classe ChifoumiVue qui gère la fenêtre graphique et les interactions avec. Pour fonctionner la classe utilise la classe Chifoumi et le fichier chifoumivue.ui.

## chifoumivue.cpp

Ce fichier regroupe les définitions des sous programmes et slots déclaré dans l’interface.

## chifoumivue.ui

Ce fichier contient toutes les informations de l’interface utilisateur.

## /Image

Le dossier image contient des images utilisables par le programme, toutes ne sont pas utilisé.

# Réalisation du modèle MVP

Pour la réalisation du modèle MVP j’ai privilégié les relations entre la vue et la présentation par des appels de sous programmes dans la présentation par des sous programmes de la vue. Les signaux et slots sont interne à la vue. J’ai utilisé cette version du modèle car il laisse une plus grande marge au niveau de la vue.

## Présentation

|  |  |  |
| --- | --- | --- |
| Déclaration | Type | Rôle |
| explicit **Presentation**(  Chifoumi \*m,  QObject \*parent = nullptr) | méthode constructeur | Constructeur de la classe, elle prend en paramètre le modèle. |
| void **deroulerUnTour**(  Chifoumi::UnCoup) | Méthode publique | Prend en paramètre le coup du joueur et déroule le tour grâce au modèle, puis met à jour la vue grâce au sous-programme MAJInterace dans la vue. |
| void **reinitialiser**() | Méthode publique | Réinitialise les coups et les scores grâce au modèle, puis met à jour la vue grâce au sous-programme MAJInterace dans la vue. |
| Chifoumi\* **getModele**() | Méthode publique | Retourne le modèle utilisé. |
| ChifoumiVue\* **getVue**() | Méthode publique | Retourne la vue utilisé. |
| void **setModele**(Chifoumi \*m) | Méthode publique | Indique le modèle à utiliser. |
| void **setVue**(ChifoumiVue \*v) | Méthode publique | Indique la vue à utiliser. |

## Vue

|  |  |  |
| --- | --- | --- |
| Déclaration | Type | Rôle |
| void **reinitialiser**() | Slot privée | Appelle le sous-programme reinitialiser de la présentation. Puis actualise les diffèrent éléments de la fenêtre. |
| void **jouerPierre**() | Slot privée | Appelle le sous-programme deroulerUnTour en passant en paramètre le coup du joueur (ici pierre). |
| void **jouerCiseau**() | Slot privée | Appelle le sous-programme deroulerUnTour en passant en paramètre le coup du joueur (ici ciseau). |
| void **jouerPapier**() | Slot privée | Appelle le sous-programme deroulerUnTour en passant en paramètre le coup du joueur (ici feuille). |
| void **afficherGagnant**(  char joueurGagnant) | Méthode privée | Affiche le gagnant et actualise les couleurs des différents éléments de la fenêtre. |
| void **afficherPoint**(  unsigned int jGauche,  unsigned int jDroit) | Méthode privée | Affiche les points. |
| void **afficherCoup**(  Chifoumi::UnCoup coupJGauche, Chifoumi::UnCoup coupJDroit) | Méthode privée | Affiche les coups. |
| void **majCouleur**(  QString propJGauche,  QString propJDroit) | Méthode  privée | Met à jour les couleurs du joueur et de la machine. |
| QPixmap **pixmapDeCoup**(  Chifoumi::UnCoup) | Méthode privée | Renvoie le QPixmap du coup passé en paramètre. |
| void **MAJInterface**(  char joueurGagnant, Chifoumi::UnCoup coupJGauche, Chifoumi::UnCoup coupJDroit, unsigned int scoreJGauche,  unsigned int scoreJDroit) | Méthode publique | Met à jour l’interface en appelant les sous-programmes de la vue afficherGagnant, afficherPoin et afficherCoup. |
| Presentation \***getPresentation**() | Méthodes publique | Retourne la présentation utilisé par la vue. |
| void **setPresentation** (  Presentation \*p) | Méthode  publique | Indique la présentation à utiliser par la vue. |
| **ChifoumiVue**(  Presentation \*p,  QWidget \*parent = nullptr) | méthode constructeur | Constructeur de la classe ChifoumiVue prenant en paramètre la présentation à utiliser. |
| ~***ChifoumiVue***() | méthode destructeur | Destructeur par défaut de la classe. |

# Test de la version

Remarque :

Lors de l’utilisation de Qt pour cette version, une erreur d’affichage des *« cout »* a empêché le déroulement des tests. Nous avons donc utilisé VScode pour cette version de la SAE. C’est donc pourquoi aucun fichier Qt ne se trouve dans le dossier programme de ce livrable.

Les tests à réaliser sont des tests de non-régression pour vérifier que l’utilisation du modèle MVP pour structurer le programme n’a pas provoqué une défaillance de fonctionnalité déjà testé précédemment.

Toutefois seul les tests de la version 1 seront de nouveau réalisé car le modèle n’a pas été modifié.

|  |  |  |  |
| --- | --- | --- | --- |
| Objet du test | Oracle | Résultat | Conclusion |
| Test du bouton rejouer | Actualisation des couleurs et du bouton\*. Réalisation des scores et des coups. | Actualisation des couleurs et du bouton. Réalisation des scores et des coups. | Test passé |
| Test du bouton pierre | Attribution du coup pierre au joueur et déroulement du tour. | Attribution du coup pierre au joueur et déroulement du tour. | Test passé |
| Test du bouton ciseau | Attribution du coup ciseau au joueur et déroulement du tour. | Attribution du coup ciseau au joueur et déroulement du tour. | Test passé |
| Test du bouton feuille | Attribution du coup feuille au joueur et déroulement du tour. | Attribution du coup feuille au joueur et déroulement du tour. | Test passé |

\*Le bouton doit passer du texte « Lancer la partie » à « Relancer la partie » au premier clique