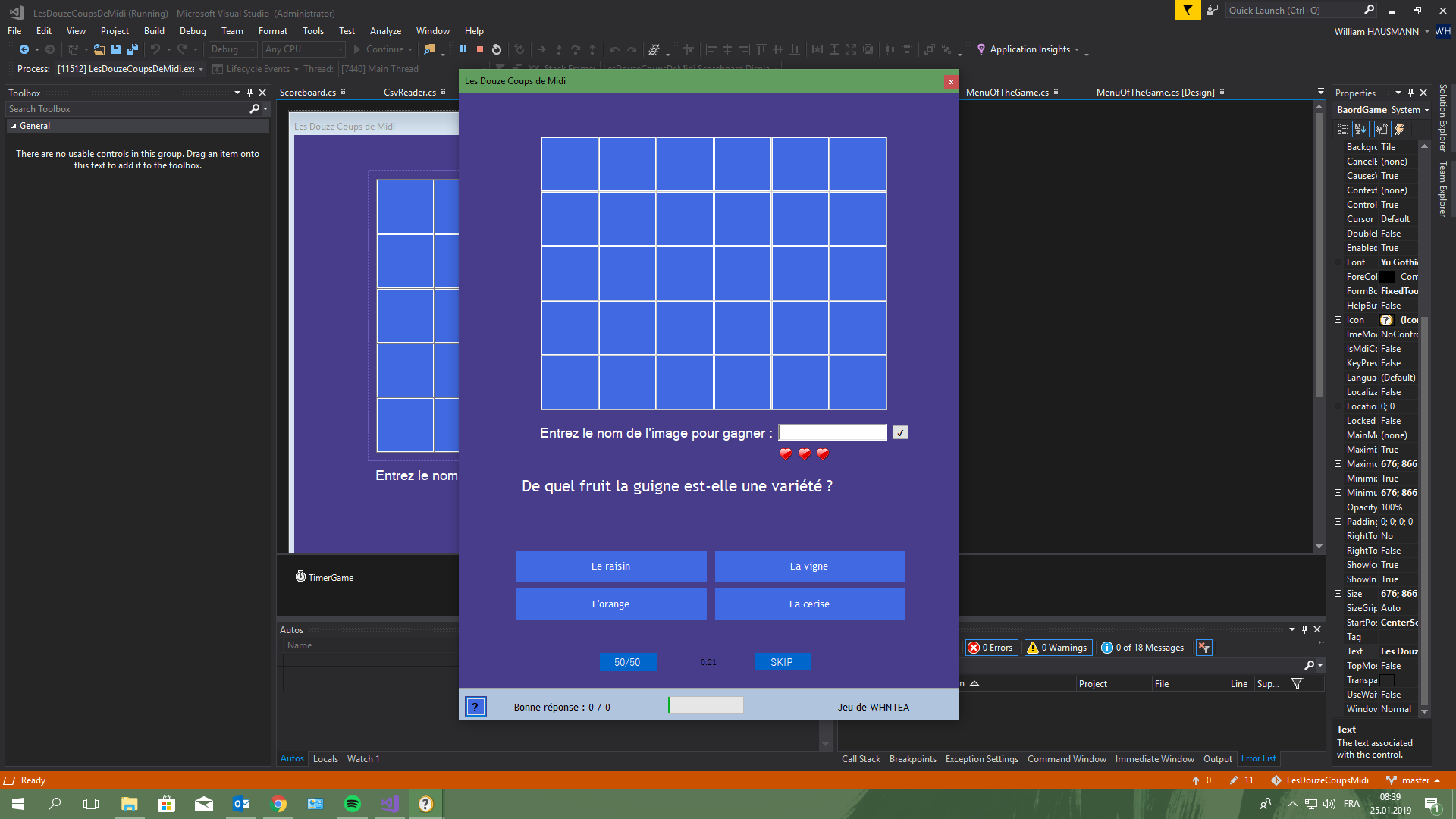


|  |
| --- |
| Les douze coups de midis |



Hausmann William

Chemin de la Cure 2a

1008, Prilly

william.hausmann@cpnv.ch

Esseiva Théo

Cossonay-gare 7b

1306, Daillens

theo.esseiva@cpnv.ch



SI-MI3a / SI-C3a

20/11/2018

Table des matières

[1 Introduction 3](#_Toc536369578)

[1.1 Cadre, description et motivation 3](#_Toc536369579)

[1.2 Organisation 3](#_Toc536369580)

[1.3 Objectifs 3](#_Toc536369581)

[2 Analyse 3](#_Toc536369582)

[2.1 Cahier des charges détaillé 4](#_Toc536369583)

[2.2 Stratégie de test 4](#_Toc536369584)

[2.3 Planification 4](#_Toc536369585)

[2.4 MCD 5](#_Toc536369586)

[2.5 Uses cases 5](#_Toc536369587)

[2.6 Maquette 6](#_Toc536369588)

[3 Conception 6](#_Toc536369589)

[3.1 Dossier de conception 6](#_Toc536369590)

[4 Réalisation 11](#_Toc536369591)

[4.1 Dossier de réalisation 11](#_Toc536369592)

[4.2 Description des tests effectués 12](#_Toc536369593)

[5 Mise en service 12](#_Toc536369594)

[5.1 Liste des documents fournis 12](#_Toc536369595)

[6 Conclusions 13](#_Toc536369596)

[7 Annexes 13](#_Toc536369597)

[7.1 Sources 13](#_Toc536369598)

# Introduction

## Cadre, description et motivation

Dans le cadre de notre 3ème année d’apprentissage d’informaticien au CPNV nous devons réaliser un projet libre en C ou C#.

Notre choix c’est porté sur un quizz car nous apprécions en faire entre amis et nous nous sommes inspiré du célèbre jeu télévisé « Les douze coups de midi ».

Le projet contiendra 30 questions auxquelles il faudra répondre parmi quatre réponses données. Il y aura une image séparée en 30 cases, à chaque fois que le joueur répond juste au question, une case contenant un bout de l’image s’affichera.

## Organisation

Eleve 1 : Esseiva théo, [theo.esseiva@cpnv.ch](mailto:theo.esseiva@cpnv.ch), 076/336/13/25

Eleve 2: Hausmann william, william.hausmann@cpnv.ch, 078/936/80/96

Responsable de projet : Benzonana, pascal, pascal.benzonana@cpnv.ch et 024 / 55 + 76079

Expert 1 : Benzonana, pascal, pascal.benzonana@cpnv.ch, 024 / 55 + 76079

Expert 2 : Ithurbide, julien, [Julien.ITHURBIDE@cpnv.ch](mailto:Julien.ITHURBIDE@cpnv.ch), 024 / 55 + 76123

## Objectifs

* Affiche aléatoire des questions dans le jeu.
* Affiche une image cachée aléatoirement.
* Vérification des réponses fausses et correctes au question.
* Vérification si le nom entré correspond à l’image cachée.
* Lecture du fichier des questions.
* Affichage d’une partie de l’image cachée aléatoirement.
* Implémentation du joker 50/50.
* Implémentation du joker Avis des autres joueurs.
* Implémentation du bouton aide
* Affichage du score
* Affichage du nombre de question passée et restante

# Analyse

Programmer un jeu en C# sur la plateforme Visual Studio.

Le jeu contiendra des questions stockée dans un fichier Excel auxquelles il faudra répondre. Au fur et à mesure des réponses correctes une image se révèlera peu à peu. Pour remporter la partie il faudra deviner l’image.

Le joueur à accès à des jokers. Un joker 50/50 qui supprime deux réponses fausses pour que le choix soit plus facile. Ainsi qu’un joker qui montre les pourcentages pour chaque réponse des choix des autres joueurs précédents.

A n’importe quel moment de la partie le joueur peut essayer de trouver l’image en rentrant un nom.

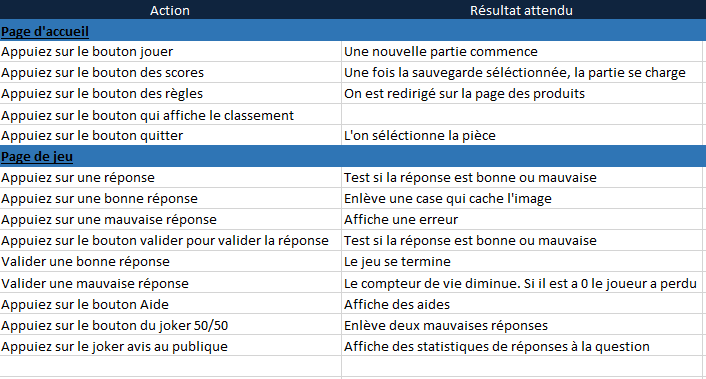
Si le joueur perd trois fois, le programme indique qu’il a perdu est se ferme.

## Cahier des charges détaillé

Votre travail sera évalué sur les critères suivants :

* Qualité fonctionnelle de l’application
* Qualité du code source
* Qualité de la documentation et de la présentation orale
* Méthode de travail / répartition des tâches

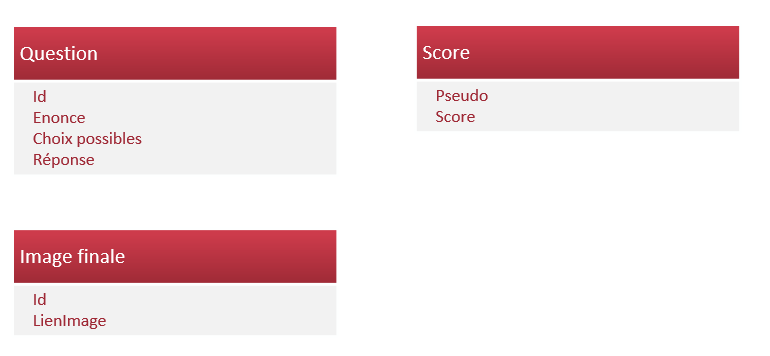
## Stratégie de test



## Planification

Voir le GitHub dans l’onglet Projet

https://github.com/WilliamWHN/LesDouzeCoupsMidi/projects

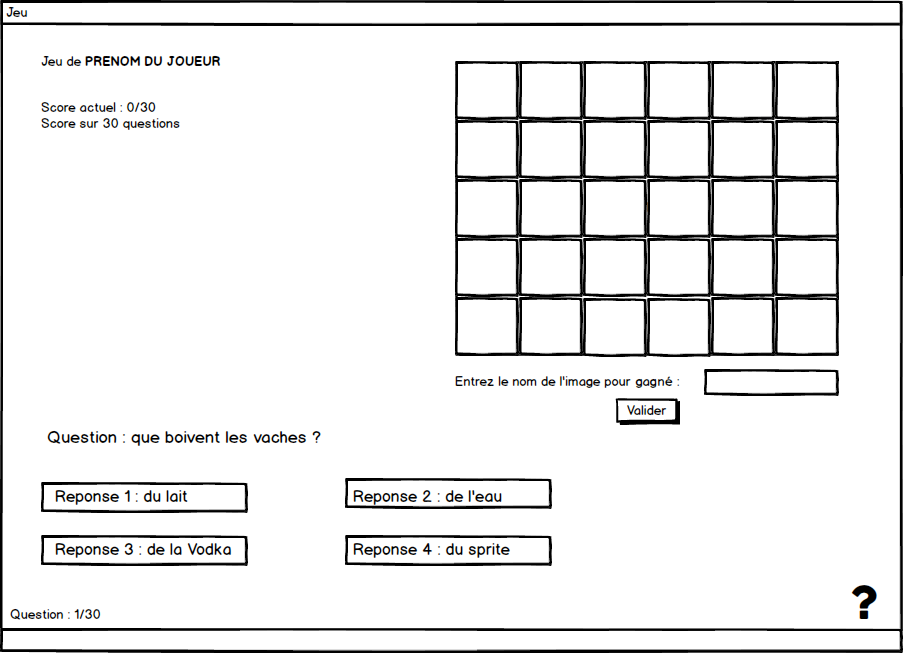


## MCD

## Uses cases



## Maquette



# Conception

## Dossier de conception

Matériel hardware utilisé :

* Processeur : Intel Core i7-6700, 3.40GHZ
* Ram : 16Go
* Type du système : Système d’exploitation 64 bits, processeur x64

Système d’exploitation : Windows 10

Les logiciels utilisés sont :

* Visual Studio 2017
* Word 2016
* Excel 2016
* Typora
* Visio 2016

|  |  |
| --- | --- |
| cmdPlay | |
| Largeur | 251 |
| Hauteur | 78 |
| Texte | Jouer |
| Click() | Lancer une partie |
|  | |

|  |  |
| --- | --- |
| cmdRules | |
| Largeur | 251 |
| Hauteur | 78 |
| Texte | Règles |
| Click() | Affiche les règles du jeu |
|  | |

|  |  |
| --- | --- |
| cmdScoreboard | |
| Largeur | 251 |
| Hauteur | 78 |
| Texte | Tableau des scores |
| Click() | Affiche les scores des joueurs précédents |
|  | |

|  |  |
| --- | --- |
| cmdCredits | |
| Largeur | 251 |
| Hauteur | 78 |
| Texte | Crédits |
| Click() | Affiche les crédits concernant le jeu |
|  | |

|  |  |
| --- | --- |
| cmdAnswer1 | |
| Largeur | 251 |
| Hauteur | 41 |
| Texte | Affiche une réponse par rapport à la question |
| Click() | Si la réponse est correcte cela enlève une case de l’image cachée et affiche une nouvelle question, sinon cela affiche juste une nouvelle question |
|  | |

|  |  |
| --- | --- |
| cmdAnswer2 | |
| Largeur | 251 |
| Hauteur | 41 |
| Texte | Affiche une réponse par rapport à la question |
| Click() | Si la réponse est correcte cela enlève une case de l’image cachée et affiche une nouvelle question, sinon cela affiche juste une nouvelle question |
|  | |

|  |  |
| --- | --- |
| cmdAnswer3 | |
| Largeur | 251 |
| Hauteur | 41 |
| Texte | Affiche une réponse par rapport à la question |
| Click() | Si la réponse est correcte cela enlève une case de l’image cachée et affiche une nouvelle question, sinon cela affiche juste une nouvelle question |
|  | |

|  |  |
| --- | --- |
| cmdAnswer4 | |
| Largeur | 251 |
| Hauteur | 41 |
| Texte | Affiche une réponse par rapport à la question |
| Click() | Si la réponse est correcte cela enlève une case de l’image cachée et affiche une nouvelle question, sinon cela affiche juste une nouvelle question |
|  | |

|  |  |
| --- | --- |
| cmdJFiftyFifty | |
| Largeur | 75 |
| Hauteur | 24 |
| Texte | 50/50 |
| Click() | Enlève deux mauvaise réponse des choix possibles |
|  | |

|  |  |
| --- | --- |
| cmdJSkip | |
| Largeur | 75 |
| Hauteur | 24 |
| Texte | SKIP |
| Click() | Passe une question est compte une bonne réponse |
|  | |

|  |  |
| --- | --- |
| cmdHelp | |
| Largeur | 75 |
| Hauteur | 24 |
| Texte | ? |
| Click() | Affiche les règles du jeu |
|  | |

|  |  |
| --- | --- |
| cmdValidate | |
| Largeur | 23 |
| Hauteur | 20 |
| Click() | Valide la réponse donnée dans la Textbox |
|  | |

|  |  |
| --- | --- |
| txtB(1-30) | |
| Largeur | 77 |
| Hauteur | 73 |
| Utilité | Cacher l’image de fin |
|  | |

|  |  |
| --- | --- |
| flwpTextBox | |
| Largeur | 482 |
| Hauteur | 384 |
| Utilité | Regroupe toutes les txtB(1-30) dans un flow layout panel |
|  | |

|  |  |
| --- | --- |
| rtxtbAnswerImage | |
| Largeur | 144 |
| Hauteur | 22 |
| Utilité | Pour que l’utilisateur puisse répondre |
|  | |

|  |  |
| --- | --- |
| rtxtbQuestion | |
| Largeur | 506 |
| Hauteur | 82 |
| Texte | Question choisie aléatoirement |
| Enter | Redirige la sélection de la RichTextbox sur le label du timer |
|  | |

|  |  |
| --- | --- |
| lblTimerGame | |
| Largeur | 31 |
| Hauteur | 14 |
| Texte | Affiche à chaque tic (1s) le chronomètre |
|  | |

|  |  |
| --- | --- |
| lblFinalQuestion | |
| Largeur | 311 |
| Hauteur | 22 |
| Texte | Entrez le nom de l’image pour gagner |
|  | |

|  |  |
| --- | --- |
| lblLife(1-3) | |
| Largeur | 19 |
| Hauteur | 23 |
| Image | Image de cœur en bitmap |
|  | |

|  |  |
| --- | --- |
| lblActualScore | |
| Largeur | 96 |
| Hauteur | 18 |
| Texte | NbBonneRéponse / NbQuestionRépondue |
|  | |

|  |  |
| --- | --- |
| lblPlayerName | |
| Largeur | 80 |
| Hauteur | 18 |
| Texte | Jeu de + « Nom du joueur » |
|  | |

|  |  |
| --- | --- |
| pbQuestion | |
| Largeur | 100 |
| Hauteur | 23 |
| Utilité | Montrer à l’utilisateur l’avancement dans le quizz |
|  | |

|  |  |
| --- | --- |
| rtxtbFooter | |
| Largeur | 666 |
| Hauteur | 49 |
| Utilité | Délimitation du footeur |
|  | |

# Réalisation

## Dossier de réalisation

Description de tous les fichiers

La documentation générale : Documentation générale du projet

Diagramme de séquence : Diagramme montrant comment le code fonctionne avec les différentes méthodes

Diagramme de classe : Diagramme résumant les différentes classes avec leurs attributs et leurs méthodes

Fichier des questions : Fichier Excel contenant les 300 questions avec leur 4 réponses

Un manuel d’utilisation : Manuel qui décris l’utilité de chaque bouton du menu et du plateau de jeu

Version du programme

Version 1.0

Système d’exploitation

Windows 10 Éducation

Programme

Visual Studio 2017

Word 2016

Excel 2016

Typora

Visio 2016

Notepad++

## Description des tests effectués

* Test du menu :
  + Si le bouton « Jouer » lance la partie
  + Si le bouton « Règles » affiche les règles
  + SI le bouton « Tableau des scores » Affiche les scores des joueurs
  + Si le bouton « Crédits » affiche les crédits
* Test du plateau de jeu
  + Si une question s’affiche bien aléatoirement
  + Si l’on peut cliquer sur une réponse
  + Si le programme vérifie si la réponse est bonne ou fausse
  + Si une case s’enlève en cas de bonne réponse
  + Si la réponse s’affiche en vert en cas de bonne réponse
  + Si la réponse entrée s’affiche en rouge et la bonne réponse en vert en cas de mauvaise réponse
  + Si quand une réponse a été donnée une autre question est chargée
  + Si le joker 50/50 enlève deux mauvaise réponse
  + Si le joker skip passe bien la question en contant la question comme bien répondue
  + Si l’on peut confirmer une réponse à l’image cachée
  + Si quand la réponse à l’image cachée est fausse, une vie s’enlève
  + Si quand la réponse à l’image cachée est correcte, le joueur remporte la partie
  + Si le bouton Aide affiche une Message Box contenant les règles

# Mise en service

## Liste des documents fournis

* Un dossier contenant le code
* Un dossier contenant le doc : La documentation générale, diagramme de séquence, Diagramme de classe, fichier Excel contenant les questions et un manuel d’utilisation.

# Conclusions

Planning

Au niveau du planning nous avons su bien gérer notre temps. Nous avons terminé le programme sans bug avec les fonctionnalités que nous voulions.

Point positif

L’interface est assez simple à comprendre et à prendre en main.

Grand choix de question.

Visibilité de la réponse correcte même si l’on répond faux.

Point négatif

Certaines images cachées on beaucoup de décor et l’entité é trouver se trouve au milieu de l’image.

Lors de la confirmation de la réponse de l’image cachée, si l’on appuie sur la touche « Enter », le programme fait un retour à la ligne. Il est obligé d’appuyer sur le bouton avec le logo « Vu ».

Difficultés particulières

Nous n’avons pas eu de difficultés particulières car nous avons eu de l’aide de la part de camarade et d’internet.

Comparaison

Au niveau des fonctionnalités nous avons rajouté des jokers. Un joker 50/50 qui enlève deux mauvaises réponses des quatre. Et un joker skip qui passe une question et compte comme une question correcte.

Nous avons ajouté un menu avec les possibilités de démarrer une partie, de consulter les règles du jeu, de voir les scores des joueurs précédents et enfin de voir les crédits (Programmeur, testeur).

# Annexes

## Sources