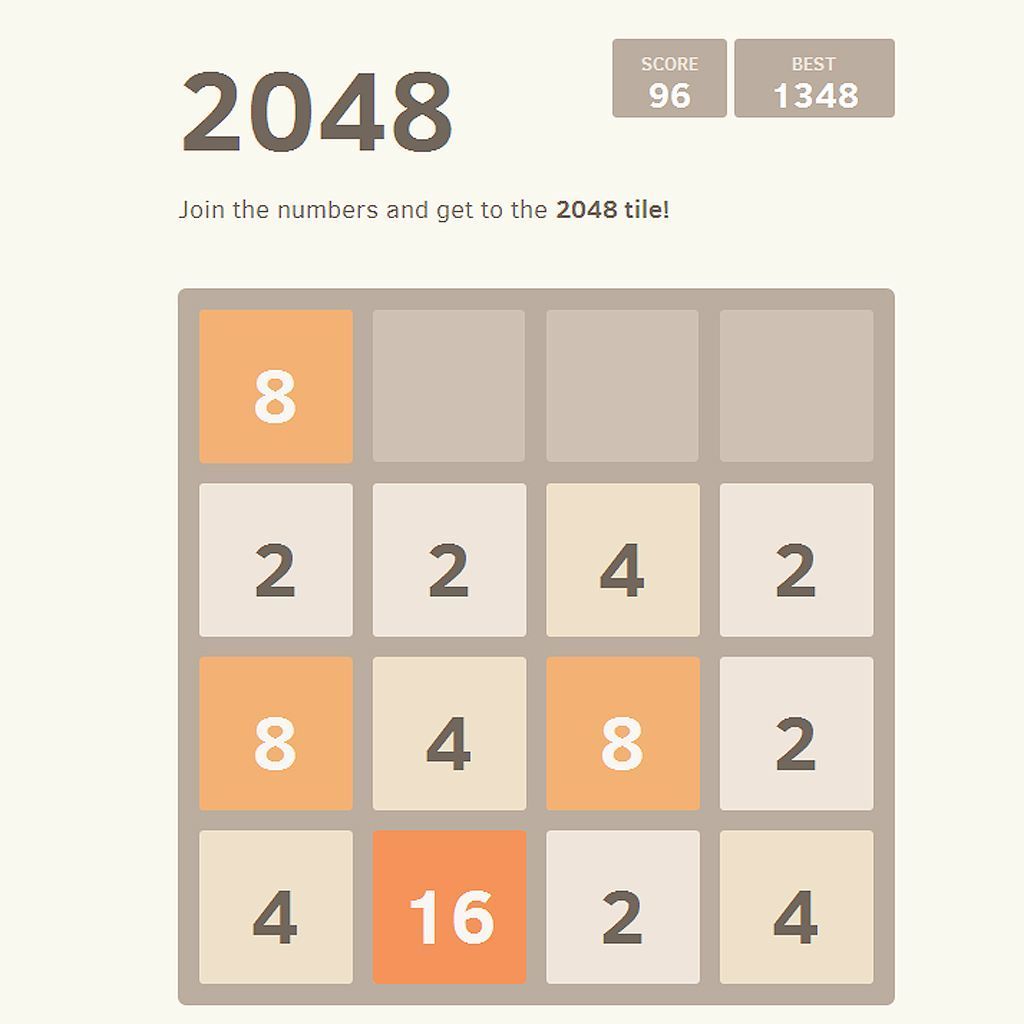


|  |
| --- |
| **2048** |



Fonseca Diogo Christophe

Christophe.FONSECA-DIOGO@cpnv.ch



Table des matières

SI-C1a

23.02.2023

[1 Introduction 3](#_Toc129249413)

[1.1 Cadre, description et motivation 3](#_Toc129249414)

[1.2 Objectifs 3](#_Toc129249415)

[1.3 Planification initiale 3](#_Toc129249416)

[2 Analyse 3](#_Toc129249417)

[2.1 Maquettes 3](#_Toc129249418)

[2.2 Stories / tests d’acceptation 4](#_Toc129249419)

[2.3 Stratégie de test 4](#_Toc129249420)

[3 Implémentation 4](#_Toc129249421)

[3.1 Vue d’ensemble 4](#_Toc129249422)

[3.2 Choix techniques 4](#_Toc129249423)

[3.3 Points techniques spécifiques 5](#_Toc129249424)

[3.3.1 Point 1 5](#_Toc129249425)

[3.3.2 Point 2 5](#_Toc129249426)

[3.3.3 Point … 5](#_Toc129249427)

[4 Tests 5](#_Toc129249428)

[4.1 Tests effectués 5](#_Toc129249429)

[4.2 Erreurs restantes 5](#_Toc129249430)

[5 Conclusions 5](#_Toc129249431)

[6 Annexes 5](#_Toc129249432)

[6.1 Sources – Bibliographie 5](#_Toc129249433)

[6.2 Journal de bord du projet 6](#_Toc129249434)

[6.3 Journal de travail 6](#_Toc129249435)

# Introduction

## Cadre, description et motivation

Le projet 2048 a été imaginé comme projet par le CPNV pour le module ICT-431.

Ce choix a été établit car nous n'allons plus utiliser le langage de programmation C mais le Python. Le jeu 2048 est un jeu qui se joue avec une grille de 4x4, le but du jeu est de tasser les tuiles semblables pour pouvoir augmenter le numéro de la tuile jusqu’à atteindre le nombre 2048 pour gagner.

Ce projet pourra lourdement solliciter mes compétences en programmation Python.

## Objectifs

Objectifs généraux

* Approfondir mes connaissances en Python.
* Savoir gérer un projet.
* Etre capable de faire une documentation détaillée.

Objectifs SMART

* Pouvoir avoir une grille de jeu 4x4.
* Afficher correctement les puissances de deux sur la grille du jeu.
* Que chaque nombre aient une couleur de fond différente dans la grille.
* A chaque clic sur la flèche directionnelle je veux pouvoir ajouter les nombres adjacents qui ont la même puissance de deux.
* Que le jeu prenne fin du moment que le nombre atteigne 2048.

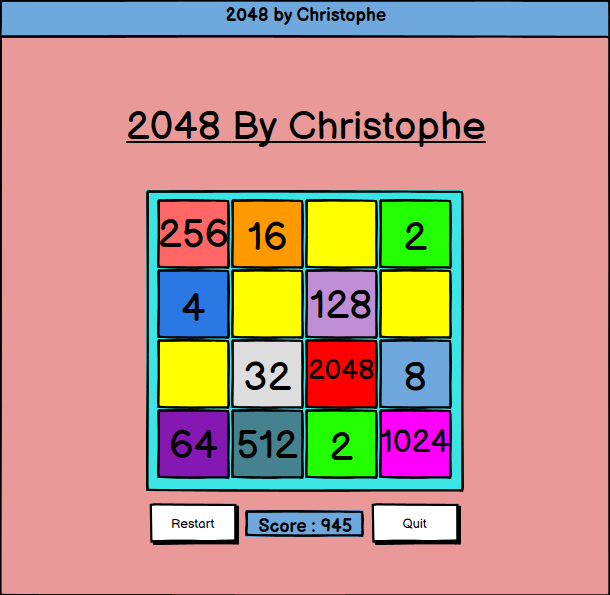
## Planification initiale

Planification réalisé sur Icescrum.

# Analyse

Ce paragraphe présente l’analyse avec les maquettes, des stories qui ont été créés et la stratégie de test.

## Maquettes



## Stories / tests d’acceptation

Les stories se trouvent sur Icescrum : <https://icescrum.cpnv.ch/p/2048CHRIST/#/backlog/all>

## Stratégie de test

Pour le développement du jeu 2048, le jeu sera programmé en python et sera installé sur la même machine où j’écrirai le code.

Je préparerai

* Le code de la grille du jeu 2048 sur PyCharmEdu.
* La fonction de tassage sur un autre fichier python dans PyCharmEdu pour pouvoir tester les différents cas qui seront possible sur le jeu 2048.

Sur la machine où je développe le jeu 2048 je ferai une série de tests unitaires pour voir si ma fonction n’a pas de problèmes, d’intégrations pour vérifier que si j’ajoute la fonction à la grille le code ne subisse pas une erreur, fonctionnels pour vérifier que le système fait ce qu’on attend de lui, et systèmes pour voir si le code fonctionne après avoir ajouté toutes les fonctionnalités dans le jeu.

Les tests seront faits par moi et certains de mes collègues de classe. Les tests de mes collègues seront faits sur d’autres machines Windows 10 pour que l’on puisse vérifier si le jeu fonctionne bien et s’adapte aux différentes configurations.

Les tests seront faits sur des machines Windows.

# Implémentation

## Vue d’ensemble

Le système réagit dès que l'utilisateur clique sur les flèches du clavier ou les touches WASD. Il réagira aussi lorsque l’utilisateur cliquera sur le bouton pour refaire une partie.

## Choix techniques

Pour ce projet, le matériel utilisé est le pc fixe fourni par le CPNV.

En terme de sauvegarde, nous avons un accès sur un serveur distant de fichiers pour que l'on puisse travailler en dehors du CPNV (Perso).

La machine est sous le système d'exploitation Windows 10.

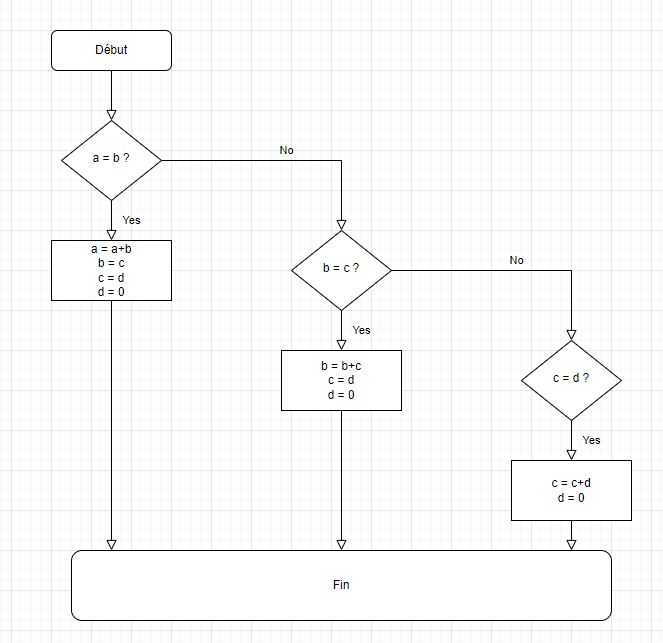
Les logiciels utilisés sont les suivant Pycharm Edu 2022.2, Visual Studio Code.

Pour le code, nous avons utilisé Python avec la version 3.8. Python est un langage de programmation interprété, multiparadigme et multiplateformes.

Nous avons utilisé aussi un module en Python qui se nomme Tkinter, Tkinter est principalement utilisé pour la conception d'interface graphique.

## Points techniques spécifiques

Voici le diagramme de flux qui compose la base de la fonction de tassage, cependant le programme a été codé avec un tableau



# Tests

## Tests effectués

<https://icescrum.cpnv.ch/p/2048CHRIST/#/project>

## Erreurs restantes

# Conclusions

Développez en tous cas les points suivants:

* Objectifs atteints / non-atteints
* Comparaison entre ce qui avait prévu et ce qui s’est passé, en termes de planning et (éventuellement) de budget
* Points positifs / négatifs
* Difficultés particulières
* Suites possibles pour le projet (évolutions & améliorations)

# Annexes

## Sources – Bibliographie

* W3shcool : <https://www.w3schools.com/python/default.asp>

Nous avons utilisé l’aide de W3school pour les boucles For, les dictionnaires et les listes.

* Aidé par des camarades de classe (Xavier, Brad, Judah)
* Tkinter 8.5 reference : a GUI for Python : <https://anzeljg.github.io/rin2/book2/2405/docs/tkinter/key-names.html>

Nous avons utilisé l’aide de Tkinter 8.5 reference : a GUI for Python pour le nom des keybinds.

## Journal de travail

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Jour** | **Semaine** | **Temps [Min]** | **Type** | **Description** | **Remarques/problèmes/déductions** |
|  |  |  |  |  |  |
| 17.févr | 6 | 20 | Documentation | Acceptation des tests et fin du sprint 2 sur IceScrum |  |
| 17.févr | 6 | 15.00 | Implémentation | Ajout de commentaires dans le code |  |
| 17.févr | 6 | / | Tests | Test de la finalité de la V02. | Aucun problème. Fin de la V02. |
| 17.mars | 6 | 30.00 | Implémentation | Compter le nombre de mouvement terminé | Solution trouvée avec l'aide de ANDOLFATTO Frederique |
| 17.mars | 6 | 20.00 | Implémentation | Compter le nombre de mouvement | Nouvelle version donnée par ANDOLFATTO Frederique mais rencontre un problème de mouvement négatif |
| 16.mars | 6 | 60.00 | Implémentation | Compter le nombre de mouvement | Gros problème, en réfléction pour trouver la solution |
| 16.mars | 6 | 30.00 | Implémentation | Ajout de la flèche du bas pour pouvoir faire le tassage dans le bas | Aidé par Judah |
| 16.mars | 6 | 30.00 | Implémentation | Ajout de la flèche du haut pour pouvoir faire le tassage dans le haut | Aidé par Judah |
| 16.mars | 6 | 30.00 | Documentation | Mise à jour de la documentation | Ajout des liens vers les sites internet pour le point 6.1 |
| 16.mars | 6 | 30.00 | Documentation | Mise à jour de la documentation | Documentation point 6.1 |
| 10.mars | 5 | 30.00 | Implémentation | Création du compte Github et implémentation des premiers commits et push | Github = Avoir différentes version de ses projets. |
| 09.mars | 5 | 30.00 | Documentation | Mise à jour de la documentation | Documentation du point 3.3 et 4 |
| 09.mars | 5 | 30.00 | Documentation | Mise à jour de la documentation | Documentation du point 3.1 à 3.2 |
| 03.mars | 4 | 80.00 | Implémentation | Ajout de la flèche de droite pour pouvoir faire le tassage dans le côté droit |  |
| 03.mars | 4 | 80.00 | Implémentation | Ajout de la flèche de gauche pour pouvoir faire le tassage dans le côté gauche | Bloqué car le tableau ne voulait pas changer en fonction de la flèche. Aidé par Brad. |
| 03.mars | 4 | 20.00 | Implémentation | Ajout de la détéction des flèches et du WASD |  |
| 03.mars | 4 | 5.00 | Tests | Test de la fonction pour voir si elle fonctionne correctement |  |
| 02.mars | 4 | 80.00 | Implémentation | Implémentation de la fonction sur la grille |  |
| 02.mars | 4 | 60.00 | Documentation | Stratégie de test terminé dans la documentation |  |
| 02.mars | 4 | 60.00 | Documentation | Diagramme de flux |  |
| 24.févr | 3 | 60.00 | Tests | Test de la fonction de tassage | Test réussis |
| 24.févr | 3 | 30.00 | Analyse | Recherche pour créer une fonction de tassage |  |
| 24.févr | 3 | 30.00 | Documentation | Création du sprint 2. |  |
| 23.févr | 3 | 90.00 | Documentation | Mise à jour de la documentation | Documentation du point 1 au point 2.2. |
| 23.févr | 3 | 5.00 | Documentation | Création de la documentation détaillée pour le projet 2048. |  |
|  |  |  |  |  |  |
| 09.févr | 2 | 10.00 | Documentation | Fin du sprint 1. | Tout est terminé pour le sprint 1. |
| 09.févr | 2 | / | Tests | Test de la finalité de la V1.5. | Aucun problème. Fin de la V1.5. |
| 09.févr | 2 | 30.00 | Implémentation | Ajouts des couleurs selon le nombre dans la case. |  |
| 09.févr | 2 | 10.00 | Documentation | Création des tests pour les stories du premier sprint. | Avancement du projet sur IceScrum. |
| 09.févr | 2 | 10.00 | Documentation | Création du premier sprint sur IceScrum. | Avancement du projet sur IceScrum. |
| 09.févr | 2 | 5.00 | Documentation | Ajout des différentes estimations dans les stories sur IceScrum. | Avancement du projet sur IceScrum. |
| 09.févr | 2 | 25.00 | Documentation | Ajout des différentes stories sur IceScrum. | Avancement du projet sur IceScrum. |
| 03.janv | 1 | 25.00 | Implémentation | Ajout du dictionnaire. |  |
| 03.janv | 1 | 10.00 | Implémentation | Enlever le tableau inutile de la V01 et changer la taille de la fenêtre. |  |
|  |  |  |  |  |  |
| 03.janv | 1 | / | Tests | Test de la finalité de la V01. | Aucun problème. Fin de la V01. |
| 03.janv | 1 | 5.00 | Implémentation | Changement de place pour que les grilles ne soient pas superposées. |  |
| 03.janv | 1 | 30.00 | Implémentation | Ajout de la 2 grille et ajouter toutes les valeurs possibles. |  |
| 03.janv | 1 | 20.00 | Analyse | Annalyse du code mis sur le commun. | Récupération du tableau de base. |
| 03.janv | 1 | 10.00 | Implémentation | Ajout des 2 chiffres dans la grille et changer le nom de la fenêtre, changer les noms de variables. | Changement de nom de la fenêtre et des variables. |
| 03.janv | 1 | 40.00 | Documentation | Création du sprint 1. |  |
|  |  |  |  |  |  |