

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

Illustration

Nom, prénom

E-mail



Classe

Date de création

Table des matières

[1 Introduction 3](#_Toc125451210)

[1.1 Cadre, description et motivation 3](#_Toc125451211)

[1.2 Objectifs 3](#_Toc125451212)

[1.3 Planification initiale 3](#_Toc125451213)

[2 Analyse 3](#_Toc125451214)

[2.1 Maquettes 3](#_Toc125451215)

[2.2 Stories / tests d’acceptation 3](#_Toc125451216)

[2.3 Stratégie de test 3](#_Toc125451217)

[3 Implémentation 4](#_Toc125451218)

[3.1 Vue d’ensemble 4](#_Toc125451219)

[3.2 Choix techniques 4](#_Toc125451220)

[3.3 Points techniques spécifiques 4](#_Toc125451221)

[3.3.1 Point 1 4](#_Toc125451222)

[3.3.2 Point 2 4](#_Toc125451223)

[3.3.3 Point … 4](#_Toc125451224)

[4 Tests 4](#_Toc125451225)

[4.1 Tests effectués 4](#_Toc125451226)

[4.2 Erreurs restantes 4](#_Toc125451227)

[5 Conclusions 5](#_Toc125451228)

[6 Annexes 5](#_Toc125451229)

[6.1 Sources – Bibliographie 5](#_Toc125451230)

[6.2 Journal de bord du projet 5](#_Toc125451231)

# Introduction

## Cadre, description et motivation

Je fais ce projet dans le cadre de ma formation d’informaticien au Centre Professionnelle du Nord Vaudois

## Objectifs

2048 se joue sur une case 4×4, avec des tuiles de couleurs et de valeurs différentes qui peuvent être déplacées par une animation de défilement lorsque le joueur appuie sur les flèches directionnelles. Pour les valeurs les plus élevées, les cellules brilleront légèrement après avoir fusionné les deux cellules précédentes et changé de couleur.

## Planification initiale

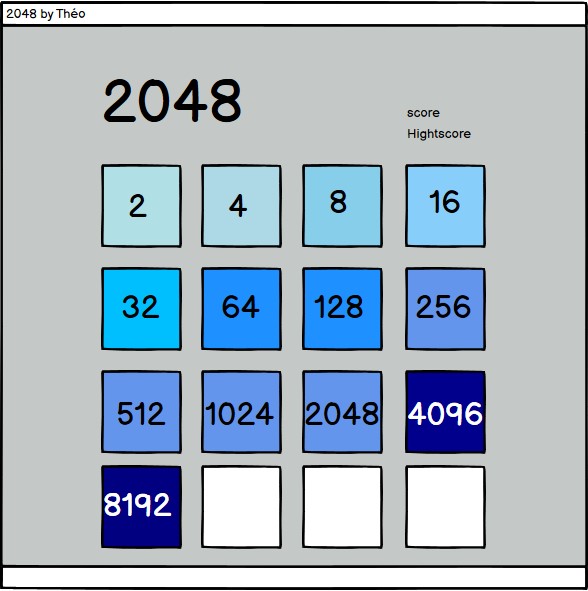
Sprint 1 : 10 février

Sprite 2 : 10 mars

Sprint 3 : 24 mars

# Analyse

## Maquettes



## Stories / tests d’acceptation

[https://icescrum.cpnv.ch/p/MA2023NUSS/#/project](https://icescrum.cpnv.ch/p/MA2023NUSS/" \l "/project)

## Stratégie de test

Les stories et les tests d’acceptation seront définis dans icescrum :

<https://icescrum.cpnv.ch/p/MA2023NUSS/#/project>

# Implémentation

## Vue d’ensemble

Le .py interagi avec un utilisateur qui va presser sur les touche puis avec un fichier texte qui note le meilleur score puis avec deux image une de win et une de loose

## Choix techniques

On m’a imposé un langage de programmation qui est python mais J’ai choisi mon environnement de développement qui est VSCode.

## Points techniques spécifiques

### Point 1

L’appelé de tasse\_4 a été compliquer à comprendre pour moi je vais donc l’expliquer :

La fonction tasse\_4 tasse tout vers la gauche faut donc lui envoyer les nombres dans un ordre persils puis les remettre dans le sens que l’on veut.

Comme ce ci :

[number[line][3], number[line][2], number[line][1], number[line][0]] = tasse\_4(number[line][3], number[line][2], number[line][1], number[line][0])

Ensuit il faut rappeler votre affichage pour moi c’est un fonction nommée display() donc on l’appelle de cette manière :

display()

Puis reste plus qu’à lui assignée une ou plusieurs touche pour pouvoir tasser dans diffèrent direction.

Comme cela :

#attraper les touches  
fen.bind("<Key>",lambda event:move(event))  
def move(event):  
 touche = event.keysym  
 if touche=="w" or touche == "W" or touche =="Up":  
 tasse\_up(event)  
 if touche=="a" or touche == "A" or touche =="Left":  
 tasse\_left(event)  
 if touche=="s" or touche == "S" or touche =="Down":  
 tasse\_down(event)  
 if touche=="d" or touche == "D" or touche =="Right":  
 tasse\_right(event)

### Point 2

Ici je vais expliquer comment écrire puis lire dans un fichier texte en python.

Tout d’abord nous devons crée un fichier txt ou l’on veut noter (ici l’exemple et un highscore) comme cela :

if score > int(hightscore):

        fichier = open("hightscore.txt", "w")

        fichier.write(str(score))

        fichier.close()

        hightscore = str(score)

Ici il y a une condition en plus car pour un meilleur score nous voulons tout d’abord écrire le score et si le score et plus petit que le meilleur score alors écrie le meilleur score. Pour cela il faut ouvrir le ficher texte avec une commande (ici la deuxième ligne) puis faire remarquer au programme que nous voulons écrire dedans à l’aide de la lettre « w » puis ensuit pressier ce que nous voulons noter (ici le score).

Ensuite pour que le meilleur score s’affiche nous devons lire dans le txt comme cela :

fichier = open("hightscore.txt", "r")

hightscore = fichier.readline()

Lire dans un fichier texte et tout simple il suffit d’ouvrir le fichier puis de préciser avec la lettre « r » que nous voulons lire le contenu et ensuite lire la ligne.

Voilà vous savez maintenant écrire et lire dans un txt en python.

# Tests

## Tests effectués

Les tests sont fait dans Icescrum :

<https://icescrum.cpnv.ch/p/MA2023NUSS/#/project>

## Erreurs restantes

La vérification de si on a perdu ne fonctionne que partiellement

# Conclusions

J’ai réussi a fait ce que je voulais et heureux d’avoir après plein de chose grâce à ce projet.

# Annexes

## Sources – Bibliographie

Pour les couleur : <https://www.blogduwebdesign.com/degrades-de-couleur/>

Lire et écrire en phyton : <http://pedagogix-tagc.univ-mrs.fr/courses/jgb53d-bd-prog/practicals/06_python/index.html#:~:text=Pour%20cela%2C%20on%20utilise%20la,m%C3%A9thodes%20pour%20lire%20ou%20%C3%A9crire>

## Journal de bord du projet

|  |  |
| --- | --- |
| **Date** | **Evénement** |
|  |  |

## Journal de travail

|  |  |  |
| --- | --- | --- |
| **Jour** | **Temps [h]** | **Description** |
| 01.02.2023 | 1h30 | Introduction du module plus faire la maquette |
| 03.02.2023 | 5h25 | Absent (rattraper) |
| 08.02.2023 | 1h30 | Code python |
| 10.02.2023 | 5h25 | Finir le code, remplir le word plus rendre le travail du sprint 1 |
| 22.02.2023 | 1h30 | Correction de tasse\_4 |
| 24.02.2023 | 5h25 | Terminer la correction puis le bon fonctionnement de tasse\_4 |
| 01.03.2023 | 1h30 | Ajoute de tasse\_4 dans mon code + l’appelé de tasse\_4 |
| 03.03.2023 | 5h25 | Finir le code en V0.2, remplir le word plus rendre le travail du sprint 2 |
| 08.03.2023 | 1h30 | Récrtion\_2048 |
| 10.03.2023 | 5h25 | Correction du sprint 2 |
| 15.03.2023 | 1h30 | Bug trouver dans la fonction tasse\_4. Bug résolu. |
| 17.03.2023 | 5h25 | Première version de l’image de win |
| 21.03.2023 | 1h30 | Réalisation de la fonction pour vérifier si on a gagner |
| 24.03.2023 | 5h25 | Vérifier si on a perdu + rendu du sprint 3 |