

CodeTris



(Une image originale représentant le projet)

Alexandre King – MID4
Avenue de Valmont 28b, 1010 Lausanne
88h
Aurélien Curchod

Table des matières

1	SPÉCIFICATIONS	3
1.1	TITRE	3
1.2	DESCRIPTION	3
1.3	MATÉRIEL ET LOGICIELS À DISPOSITION	3
1.4	CAHIER DES CHARGES	3
1.5	LES POINTS SUIVANTS SERONT ÉVALUÉS	3
1.6	VALIDATION ET CONDITIONS DE RÉUSSITE	3
2	PLANIFICATION INITIALE	3
3	ANALYSE	4
3.1	OPPORTUNITÉS	4
3.2	DOCUMENT D'ANALYSE ET CONCEPTION	4
3.3	CONCEPTION DES TESTS	4
3.4	PLANIFICATION DÉTAILLÉE	5
4	RÉALISATION	5
4.1	DOSSIER DE RÉALISATION	5
4.2	POINTS SUPPLÉMENTAIRES	6
5	TESTS	6
5.1	DOSSIER DES TESTS	6
6	CONCLUSION	6
6.1	BILAN DES FONCTIONNALITÉS DEMANDÉES	6
6.2	BILAN DE LA PLANIFICATION	6
6.3	BILAN PERSONNEL	7
7	DIVERS	7
7.1	JOURNAL DE TRAVAIL	7
7.2	WEBOGRAPHIE	7
8	ANNEXES	7

1 RÉSUMER

//à voir si mettre ici ou plus loin

2 SPÉCIFICATIONS

2.1 Titre

CodeTris : Le bloc de la programmation

2.2 Description

CodeTris est un jeu éducatif qui intègre des questions de programmation dans le jeu d'arcade Tetris.

L'objectif de ce projet est de créer une expérience ludique et éducative, encourageant les joueurs à renforcer leurs compétences en programmation tout en jouant à un jeu classique.

Ce jeu pourrait être utilisé par exemple comme accroche lors des modules de programmation de 1ère année.

2.3 Matériel et logiciels à disposition

- PC de l'ETML
- Visual Studio 2022
- Github
- Suite office

2.4 Cahier des charges

Voir annexes

2.5 Les points suivants seront évalués

- Le rapport
- Les planifications (initiale et détaillée)
- Le journal de travail
- Le code et les commentaires
- Les documentations de mise en œuvre et d'utilisation

2.6 Validation et conditions de réussite

- Compréhension du travail
- Possibilité de transmettre le travail à une personne extérieure pour le terminer, le corriger ou le compléter
- Etat de fonctionnement du produit livré

3 PLANIFICATION INITIALE

//Début TPI

- Ce paragraphe présente le planning d'origine (date de début, date de fin, vacances et congés, liste hiérarchique des tâches ou GANTT, jalons, durée totale)
- Toutes les mises à jour subies par le planning sont à reporter (avec date de mise à jour) et peuvent déboucher sur plusieurs versions de plannings.

4 ANALYSE

4.1 Opportunités

- Mise en œuvre d'un jeu avec C# console
- Utilisation de GitHub
- Jeu éducatif pour des élèves de première année

4.2 Document d'analyse et conception

Menu (principal, Options, score)

Jeu

Score

DB (qu'est qui est mis en db, pourquoi ?) (MCD, MLD, dictionnaires de données)

UML (classe, event)

- Ce paragraphe décrit le fonctionnement de manière détaillée.
 - Autant que possible de manière graphique, imagée, tableaux, etc.
 - Tous les cas particuliers devraient y être spécifiés...
- Il s'agit d'y présenter les fonctionnalités à développer :
 - Découpage en étapes, en modules, en fonctionnalités, etc.
 - Schémas de navigation, schémas événementiels, structogramme, pseudocode, etc.
- Si le projet inclut une base de données :
 - Dictionnaire des données
 - Modèle conceptuel des données, modèles logique des données.

4.3 Conception des tests

Tableau des tests à réaliser lors du projet. Les résultats seront inscrits dans la [partie test](#) de ce document.

Nom du Test	Fonctionnalité testée	Description	Condition de réussite	Importance (0 = bas, 5 = haut)
Menu	Menu	Le joueur peut se déplacer dans le menu ainsi que quitter le jeu via le menu principal	Le joueur peut accéder à toutes les parties du menu et des sous-menus	4
Options	Menu d'options	Différentes options peuvent être sélectionnées. Celles-ci impacteront le jeu	Les options en jeu correspondent bien à celles choisies par l'utilisateur	3
Pseudo	Pseudo du joueur	Le joueur possède un pseudo unique. Il peut le définir lors du lancement du jeu	Le pseudo défini par le joueur est unique. Lors du premier lancement du jeu, il le définit	3
Déplacement des pièces	Déplacement et pivotement des pièces	Le joueur doit pouvoir déplacer les pièces horizontalement, les faire pivoter, et accélérer leur descente	Le joueur peut utiliser les touches choisies dans le menu « options » pour effectuer les différentes actions	5

Questions	Questions	Des questions apparaissent lors de la complétion d'une ligne	Les questions s'affichent et le joueur peut y répondre. Elles proviennent de la DB	4
Score	Changement du score	Incrémentation du score lors d'une bonne réponse à une question	Si le joueur répond correctement à une question, le score augmente en fonction de la difficulté	2
Game over	Activation au bon moment	Lorsqu'une pièce a atteint le haut de la zone, la partie est terminée	Le game over s'active uniquement lorsqu'une pièce touche le haut de la zone. La partie est ensuite sauvegardée en DB	3
DB	Bon fonctionnement de la DB avec le jeu	La connexion avec la DB est faite correctement. Des informations doivent pouvoir être récupérées et insérées	Les informations sont correctement récupérées et insérées. Un fichier de log est prévu en cas de problème	4

Les tests seront faits au fil du temps et dans un ordre logique (certains tests ne pouvant être fait avant d'autres dû à un ordre d'implémentation des fonctionnalités). Les tests peuvent être réalisés soit par une personne interne au projet (développeur, CDP, expert), soit par une personne externe (camarade, famille, profs, ect...), soit via un test unitaire. Le personne/outils ayant réalisé le test sera mentionné dans le tableau de résultat.

4.4 Planification détaillée

//Début TPI

- A ce stade, après l'analyse complète du projet, un planning détaillé et complet (avec tâches, sous-tâches, dépendances, durée, ...) peut être finalisé.
- Le planning détaillé doit s'inscrire dans le planning initial. Il faut que l'on puisse situer cette planification détaillée par rapport à la planification initiale.

5 RÉALISATION

5.1 Dossier de Réalisation

//Expliquer mise en place de git avec visual studio, mise en place du projet, framework, etc...

Mise en place du git -> projet VS2022 (framework)->Classes->Code important -> Gestion Thread (+optimisation)-> DB

Se baser sur le git P_Appro2 pour ordre et éléments à voir

- Cette partie permet de reproduire ou reprendre le projet par un tiers.
- Pour chaque étape, il faut décrire sa mise en œuvre. Typiquement :
 - Versions des outils logiciels utilisés (OS, applications, pilotes, librairies, etc.)
 - Configurations spéciales des outils (Equipements, PC, machines, outillage, etc.)

- Code source commenté des éléments logiciels développés.
- Modèle physique d'une base de données.
- Arborescences des documents produits.
- Il faut décrire le parcours de réalisation et justifier les choix.

5.2 Points supplémentaires

Ajout supplémentaire	Date	Description

- Historique des modifications demandées (ou nécessaires) aux spécifications détaillées.
- Date, raison, description, etc.

6 TESTS

6.1 Dossier des tests

Cette partie résumera les tests effectués et permettra d'effectuer un bilan de ceux-ci.

Nom du test	Date du test	Passation	Commentaire
Menu			
Action dans le jeu			
Score			
Game over			
DB			

- On dresse le bilan des tests effectués (qui, quand, avec quelles données...) sous forme de procédure. Lorsque cela est possible, fournir un tableau des tests effectués avec les résultats obtenus et les actions à entreprendre en conséquence (et une estimation de leur durée).
- Si des tests prévus dans la stratégie n'ont pas pu être effectués :
 - raison, décisions, etc.
- Liste des bugs répertoriés avec la date de découverte et leur état:
 - Corrigé, date de correction, corrigé par, etc.

7 CONCLUSION

7.1 Bilan des fonctionnalités demandées

//tableau bilan final (qu'est qui a été fait, qu'est-ce qui reste à faire, pourcentage et temps restant)

- Il s'agit de reprendre point par point les fonctionnalités décrites dans les spécifications de départ et de définir si elles sont atteintes ou pas, et pourquoi.
- Si ce n'est pas le cas, estimer en « % » ou en « temps supplémentaire » le travail qu'il reste à accomplir pour terminer le tout.

7.2 Bilan de la planification

//comparaison entre planif et réalité, expliquer les deltas importants

- Distinguer et expliquer les tâches qui ont généré des retards ou de l'avance dans la gestion du projet. Indiquer les différences entre les planifications initiales et détaillées avec le journal de travail.

7.3 Bilan personnel

- Si c'était à refaire:
 - Qu'est-ce qu'il faudrait garder ? Les plus et les moins ?
 - Qu'est-ce qu'il faudrait gérer, réaliser ou traiter différemment ?
- Qu'est-ce que ce projet m'a appris ?
- Suite à donner, améliorations souhaitables, ...
- Remerciements, signature, etc.

8 DIVERS

8.1 Journal de travail

- Date, activité (description qui permet de reproduire le cheminement du projet), durée, liens et références sur des documents externes. Lorsqu'une activité de recherches a été entreprise, il convient d'énumérer ce qui a été trouvé, avec les références.

8.2 Webographie

//tableau avec nom du site et lien de la page concernée

- Références des sites Internet consultés durant le projet.

9 ANNEXES

//git hub (public), installateur ?, JRNLTRV, DB, doc utilisateur (mode d'emploi) ?,

- Listing du code source (partiel ou, plus rarement complet)
- Guide(s) d'utilisation et/ou guide de l'administrateur
- Etat ou « dump » de la configuration des équipements (routeur, switch, robot, etc.).
- Extraits de catalogue, documentation de fabricant, etc.