



CAHIER DES CHARGES

Projet d'algorithmique 2016-2017

Version: CDC_Groupe04_GDE_MBO_PPA_V1.1

Auteur: ISEN Toulon

ISEN Toulon - Yncrea
Maison du Numérique et de l'Innovation
Place Georges Pompidou
83000 Toulon

Description du document

Type		Version	Confidentialité	
Cahier des charges		1.0	Usage externe	
	Nom	Fonction	Date	Visa
	Rédacteur	ISEN Toulon		
	Vérificateur			
	Approbateur			
Destinataire		Fonction	Organisme	
Public			ISEN	

Révisions du document

[illegible]

Sommaire

1. Présentation du projet.....	7
A) Contexte du projet.....	7
B) Objectifs du projet.....	8
C) Description de l'existant.....	8
D) Acceptation du produit.....	8
2. Expression fonctionnelle des besoins.....	8
A) Besoins fonctionnels.....	9
B) Besoins non fonctionnels.....	11
3. Contraintes.....	12
A) Budgétaires.....	12
B) Temporelles.....	12
C) Normatives.....	12
4. Déroulement du projet.....	12
A) Planification.....	12
B) Documentation.....	12
C) Équipe projet et responsabilités.....	12

Ce document est un exemple de cahier des charges pour un projet de développement logiciel. Le cahier des charges formalise le besoin afin qu'il soit compris par l'ensemble des personnes qui vont être impliquées dans le projet. Il doit indiquer précisément les spécifications attendues.

Pour qu'un projet puisse être qualifié de réussi, il est impératif de définir précisément :

- les objectifs fixés,
- les ressources requises,
- la planification,
- les méthodes d'évaluation,
- les méthodes de contrôle.

Considéré dans sa globalité, un projet requiert les étapes suivantes :

- nomination d'une équipe et d'un responsable du projet,
- état de l'art et exploration des possibilités techniques,
- rédaction du cahier de charges
- mise en œuvre, suivi et évaluation du projet.

La structure suivie dans ce document s'inspire de ce que propose la norme NF EN 16271 (février 2013) qui a remplacé la norme NF X50-151 (septembre 2007).

"Le cahier des charges est un support indispensable de dialogue entre l'utilisateur et le concepteur. L'intérêt essentiel pour les deux parties est de ne pas découvrir des non-conformités fonctionnelles, d'usage ou réglementaires après réalisation. Il est en effet plus facile et moins coûteux d'agir dès la conception. " INRS, ED6231, mai 2016.

Index des illustrations

Illustration 1: Plateau de jeu Quixo.....	7
Illustration 2: Fonctionnement du Quixo.....	7

Index des tables

REFERENCES

Référence	Description	Nom
[1]		
[2]		

DEFINITIONS

Sans objet

ABBREVIATIONS

ISEN : Institut Supérieur de l'Electronique et du Numérique
 SQL : Structured Query Language

1. Présentation du projet

A) Contexte du projet



Illustration 1: Plateau de jeu Quixo

Le jeu du Quixo a connu sa première version commerciale en 1995. C'est un jeu de réflexion sur un plateau de 5*5 cubes. Deux adversaires sont opposés et ont chacun pour objectif d'aligner (ligne ou diagonale) 5 cubes de même symbole.

Les 25 cubes sont prisonniers du plateau : chaque cube possède 4 faces neutres, une face marquée d'une croix et une face marquée d'un rond. Au départ, les cubes sont disposés faces supérieures neutres. Chacun son tour, le joueur choisit un cube situé sur la partie extérieure du plateau. Il le replace avec sa marque sur la face supérieure, en poussant sur une rangée incomplète (principe du pousse – pousse).

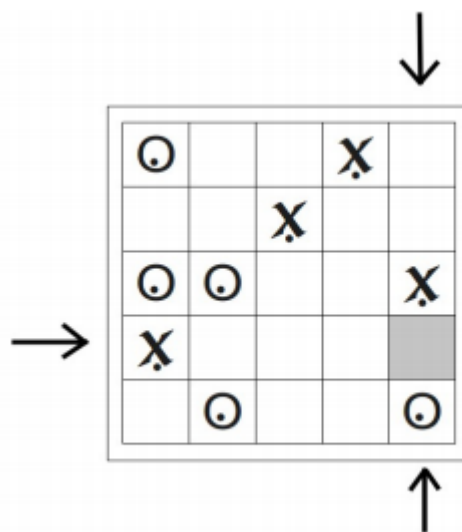


Illustration 2: Fonctionnement du Quixo

Le jeu est conçu avec des cubes en bois engendrant une consommation abusive de bois ce qui peut nuire à l'environnement.

B) Objectifs du projet

L'objectif du projet est de proposer une version logicielle du Quixo, pouvant se jouer seul contre l'ordinateur, à deux sur le même ordinateur ou à quatre, en deux contre deux sur le même ordinateur. Cela permettra de réduire à zéro le nombre de cubes en bois à produire et permettra également d'organiser des tournois entre joueurs sans avoir besoin du matériel physique adéquat.

C) Description de l'existant

Il n'y a pas de concurrents sur ce secteur, notre produit sera donc totalement innovant.

D) Acceptation du produit

Le logiciel sera considéré comme acceptable s'il vérifie tous les tests spécifiés dans le cahier de recette.

2. Expression fonctionnelle des besoins

Les exigences décrites ci-après ont pour but de décrire les caractéristiques du projet.

Elles sont verbalisées en trois parties : objet-verbe-attribut, où :

- objet désigne le sujet sur lequel porte l'exigence ;
- attribut désigne la caractéristique que le verbe impose à l'objet.

Numérotation

Les exigences sont numérotées sur quatre chiffres, de dix en dix pour pouvoir éventuellement insérer, après accord de toutes les parties concernées, de nouvelles exigences à côté de précédentes exigences.

Un numéro d'exigence abandonné ne pourra pas être réutilisé.

Les exigences sont de plusieurs types, chaque type décrit ci-après étant représenté par une lettre caractéristique. Cette lettre prefixera chaque exigence.

Les exigences peuvent être facultatives ou indispensables. Les lettres F et I suffixeront respectivement les numéros d'exigences décrites ci-dessus.

Types d'exigences

Les types d'exigences sont les suivants (avec entre parenthèses la lettre servant de préfixe au numéro d'exigence) :

- **exigences Fonctionnelles (EF)** : portent sur ce que le produit doit être capable de faire ;

- **exigences de Design (ED)** : portent sur tout ce qui a trait à la forme au travers de laquelle on interagira avec le produit ;
- **exigences d'Implémentation (EI)** : portent sur les contraintes techniques liées à la réalisation du produit ;
- **exigences de Performance (EP)** : portent sur des niveaux quantitatifs qui doivent être atteints dans des conditions à préciser ;
- **exigences de Maintenance (EM)** : portent sur tout le support qui est à mettre en œuvre pour assurer le bon fonctionnement du produit ;
- **exigences de Validation (EV)** : portent sur les actions qui permettent de valider des exigences ;
- **exigences Sécuritaires (ES)** : portent sur les mesures à prendre pour assurer la sécurité et la sûreté des informations ;
- **exigences Légales (EL)** : portent sur les mesures à prendre pour assurer la légalité des actions du produit.

Exemple : EF_0010_I sera l'exigence fonctionnelle indispensable 0010.

EP_0030_F sera l'exigence de performance facultative 0030.

A) *Besoins fonctionnels*

Les besoins fonctionnels décrivent les différentes fonctions que le logiciel devra assurer. Il est préférable de les classer par catégories, afin d'en simplifier la lecture.

Exigences fonctionnelles

EF_0010_I

Chaque joueur jouera sur une grille de 5 cases par 5 cases.

EF_0020_I

Chaque joueur choisi un symbole (croix ou cercle).

EF_0030_I

Chaque cube possédera trois faces différentes (croix, rond et neutre).

EF_0040_I

Chaque joueur à tour de rôle choisi un pion neutre ou de son signe qui se trouve à l'extrémité de la grille.

EF_0050_I

Chaque joueur repose son cube à l'extrémité d'une rangée incomplète.

EF_0060_I

Un joueur ne peut pas piocher un cube situé sur les 9 cases centrales.

EF_0070_I

Le joueur ne peut reposer un cube à l'endroit où il l'a pioché.

EF_0080_I

Le joueur ne peut pas piocher un cube ayant déjà été joué par l'adversaire.

EF_0090_I

Un joueur gagne lorsqu'il a aligné cinq mêmes faces de son symbole.

EF_0100_I

Si un joueur crée involontairement une ligne gagnante de signe adverse, alors il perd, même si dans le même temps il a réussi à compléter la sienne.

EF_0110_I

Un joueur ne peut pas passer son tour.

EF_0120_I

Un bouton « Menu » permettra de revenir au menu lors d'une partie.

EF_0130_I

Pour activer un bouton, l'utilisateur utilisera un clic gauche de souris.

EF_0140_I

Un bouton « Quitter » sur l'écran principal permettra de quitter le jeu.

EF_0150_I

Un bouton « Règles » sur l'écran principal permettra d'accéder aux règles du jeu.

EF_0160_I

Un bouton « 1 vs 1 » sur l'écran principal permettra d'accéder au jeu en mode 1 personne contre une autre classique.

EF_0170_I

Un bouton « 1 vs IA » sur l'écran principal permettra d'accéder au jeu en mode 1 personne contre l'ordinateur.

EF_0180_I

L'utilisateur accèdera au début du jeu à un menu principal.

EF_0190_I

En fin de partie le joueur accède à un menu affichant le gagnant et les scores.

EF_0191_I

Un bouton « Rejouer » permettra de recommencer une partie en gardant les scores des joueurs.

EF_0192_I

Un bouton « Menu » lors de l'affichage du gagnant et des scores permettra de revenir au menu principal.

EF_0200_F

Un bouton « 2 vs 2 » sur l'écran principal permettra d'accéder au jeu en mode 2 personnes contre deux autres.

Exigences de design

ED_0010_I

Le jeu s'affichera dans une fenêtre de taille 800x800.

ED_0020_I

Le jeu permettra de sélectionner un cube et le déplacer par le clic gauche.

ED_0021_F

Le jeu affichera les zones possibles de pose du cube.

ED_0030_I

Le jeu affichera la face neutre des cubes en début de partie.

ED_0040_I

Les trois symboles sur les faces sont croix ou rond ou neutre.

ED_0050_I

Le jeu affichera une case de couleur différente pour représenter une case vide.

ED_0060_I

L'utilisateur pourra modifier la taille de la fenêtre. Les éléments resteront en place.

ED_0070_I

L'utilisateur pourra suivre son score au fil des parties.

ED_0080_I

A la fin de la partie, le joueur pourra en rejouer une autre ou quitter le jeu.

ED_0090_I

Un bouton « Menu » permettra de revenir au menu lors d'une partie.

ED_0100_I

Affichage du joueur courant.

ED_0110_I

Au début d'une partie, les joueurs choisiront leurs symboles.

ED_0120_I

Le fond sera de couleur grise (#EFF0F4).

ED_0121_I

Les boutons seront de couleur bleu (#415F9D).

ED_0122_I

Les cubes seront de couleur bleu (#233B6E).

ED_0123_I

Les symboles seront de couleur grise (#EFF0F4).

ED_0124_I

Le texte sera de couleur grise (#EFF0F4) sur bleu et bleu (#415F9D) sur gris.

ED_0130_I

Le jeu sera entièrement joué à la souris à l'aide des clics gauches.

B) Besoins non fonctionnels

Les besoins non fonctionnels sont tous ceux qui n'ont pas trait aux fonctions du logiciel, comme le système d'exploitation sur lequel il devra fonctionner.

Exigences d'implémentation

EI_0010_I

Le logiciel devra fonctionner sous système d'exploitation Linux.

EI_0020_I

Le logiciel devra être écrit en langage C99.

Exigences de performance

EP_0010_I

Le logiciel devra consommer au plus 200 Mo de mémoire.

EP_0020_I

La phase de réflexion de l'ordinateur ne devra pas excéder 20 secondes.

Exigences de maintenance

EM_0010_I

Le logiciel sera maintenu par l'équipe projet pendant 4 mois.

EM_0020_I

Les sources du logiciel seront commentées au format Doxygen.

Exigences de validation

EV_0010_I

L'interface utilisateur sera validée manuellement.

Exigences sécuritaires

Aucune exigence sécuritaire.

Exigences légales

Aucune exigence légale.

3. Contraintes

A) Budgétaires

Le budget alloué pour ce projet sera de 30 000 €. Les moyens matériels mis à disposition sont trois ordinateurs reliés à internet.

B) Temporelles

La livraison du produit fini est prévue en mars 2017. La documentation relative à l'analyse du logiciel devra être livrée fin janvier 2017.

C) Normatives

Le code source du projet devra respecter la norme C99.

4. Déroulement du projet

A) Planification

Le projet se déroulera sur 4 mois.

La première semaine sera consacrée à l'analyse de l'existant et à la rédaction du CDC, du CDR, du CDCG et du CDCD.

B) Documentation

Le projet livré sera accompagné d'un cahier de conception général et détaillé, d'un cahier de recette, et d'un rapport final.

C) Équipe projet et responsabilités

L'équipe sera constituée d'un chef de projet, et de 2 développeurs.