

# CAHIER DES CHARGES

## Projet Worms

Version: 2.3

Auteur: ORNON Thomas, PITUELLO Cédric

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

## Description du document

Type		Version	Confidentialité	
Cahier des charges		2.3	Usage externe	
Rédacteur   Vérificateur  Approbateur	Nom	Fonction	Date	Visa
	ORNON Thomas			
	PITUELLO Cédric			
Destinataire		Fonction	Organisme	
Public			ISEN	

## Révisions du document

[illegible]

# Sommaire

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

## Index des illustrations

Illustration 1: Screenshot d'une partie du jeu Worms ..... 6

## Index des tables

## REFERENCES

Référence	Description	Nom
[1]	Site web Monogame	<a href="http://www.monogame.net/">http://www.monogame.net/</a>
[2]	Site web Farseer	<a href="https://farseerphysics.codeplex.com/">https://farseerphysics.codeplex.com/</a>

## DEFINITIONS

Sans objet

## ABBREVIATIONS

ISEN : Institut Supérieur de l'Electronique et du Numérique

### 1. Présentation du projet

#### ***A) Contexte du projet***

Notre projet est la conception d'un jeux vidéo basé sur l'univers Worms. Le principe est simple : joué au tour par tour sur une carte 2D, vous incarnez un simple vers appelé « Worms ». Vous êtes armés d'une arme dévastatrice (ex : le fameux bazooka) et votre but est d'éliminer le ou les Worms adverses !

Vous disposez d'une équipe composée de plusieurs Worms et vous combattez contre une autre équipe. La victoire est vôtre lorsque votre équipe a survécu !

Le champ de bataille est une carte 2D destructible !  
Maintenant combattez !



Illustration 1: Screenshot d'une partie du jeu Worms Armageddon

## B) Objectifs du projet

L'objectif du projet est donc de développer un clone de Worms. Les aspects importants à gérer seront la physique des vers avec le décor, la gestion et l'animation des tirs, la map destructible et l'interface, entre autres. Le langage de programmation est C#. Nous utiliserons le framework Monogame, avec peut-être Velcro Physics (anciennement Farseeer Physics) pour le moteur physique.

## C) Description de l'existant

Le jeux Worms est sorti en 1995, édité par Ocean Software, sur Amiga et PC. A ce jour, ils existent une multitude de générations de jeux Worms.

## 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.

### 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\_0111\_I

Le worms peut aller à gauche

EF\_0112\_I

Le worms peut aller à droite

EF\_0113\_I

Le worms peut aller à sauter

EF\_0114\_I

Le worms est par défaut en attente

EF\_0121\_I

Le worms peut viser

EF\_0122\_I

Le worms peut tirer



- EF\_0210\_I  
Le programme gère les collisions
- EF\_0220\_I  
Le programme gère la gravité
- EF\_0230\_I  
Le programme gère la destruction de la carte
- EF\_0240\_I  
Le programme gère le déplacement des projectiles

### **Exigences de design**

- ED\_0111\_I  
Le worms a une animation quand il va vers la gauche
- ED\_0112\_I  
Le worms a une animation quand il va vers la droite
- ED\_0113\_F  
Le worms a une animation quand il attend (c.à.d quand il ne fait rien)
- ED\_0114\_F  
Le worms a une animation quand il saute
- ED\_0115\_I  
Le worms a une animation quand il tire
- ED\_0116\_F  
Le worms a une animation quand il meurt
  
- ED\_0121\_I  
Le projectile a une animation de déplacement
- ED\_0122\_F  
Le projectile a une animation d'explosion quand il y a impact
  
- ED\_0210\_I  
Le programme affiche une carte en jeu
- ED\_0220\_I  
Le programme met à jour la carte
- ED\_0310\_F  
Le programme a un menu accueil
- ED\_0320\_F  
Le programme a un menu Option
- ED\_0331\_F  
Le joueur peut afficher son inventaire en jeu
- ED\_0332\_F  
Le joueur peut se déplacer dans son inventaire en jeu
- ED\_0333\_F  
Le joueur peut choisir son arme en jeu
  
- ED\_0410\_F  
Le programme gère le bruit des armes en jeu
- ED\_0420\_F  
Une musique est jouée pendant la partie

## **B) Besoins non fonctionnels**

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

### **Exigences d'implémentation**

EI\_0010\_I

Le logiciel devra fonctionner sous système d'exploitation Windows.

EI\_0020\_I

Le logiciel devra être écrit en langage C#.

EI\_0030\_I

Le logiciel devra utiliser le FRAMEWORK MONOGAME.

EI\_0040\_F

Le logiciel devra utiliser le FRAMEWORK FARSEER.

### **Exigences de performance**

EP\_0010\_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 1 an.

### **Exigences de validation**

EV\_0010\_I

L'interface utilisateur sera validée manuellement.

EV\_0020\_I

Le jeu sera jouable à plusieurs sur le même ordinateur.

### **Exigences sécuritaires**

Aucune exigence sécuritaire.

### **Exigences légales**

Aucune exigence légale.

## **3. Contraintes**

### **A) Budgétaires**

Le budget alloué pour ce projet sera de 0€. Les moyens matériels mis à disposition sont trois ordinateurs reliés à internet.

### **B) Temporelles**

La livraison du produit fini est prévue en mai 2018. La documentation relative à l'analyse du logiciel devra être livrée fin décembre 2017.

### **C) Normatives**

Le code source du projet devra être documenter pour faciliter la compréhension et lecture du code.

## **4. Déroulement du projet**

### ***A) Planification***

Le projet se déroulera sur 9 mois.

La première semaine sera consacrée à l'analyse de l'existant et à la rédaction du CDC et du WBS dans un premier temps.

### ***B) Documentation***

Le projet livré sera accompagné d'un cahier de conception général et détaillé.

### ***C) Equipe projet et responsabilités***

L'équipe sera constituée d'un chef de projet et de 1 développeur.