



ECOLE NATIONALE SUPÉRIEURE DE  
L'ELECTRONIQUE ET DE SES  
APPLICATIONS

PROJET LATÉRAL TRANSVERSAL

# Tales of Kornwal

*OUAZZAGHTI Reda*  
*et*  
*ZOUHDI Zakaria*

Projet de troisième année supervisé par  
M. GRANIER et Prof. GOSSELIN

29 Septembre 2016

# Table des matières

<b>1</b>	<b>Objectif</b>	<b>2</b>
1.1	Présentation générale . . . . .	2
1.2	Règles du jeu . . . . .	3
<b>2</b>	<b>Description et conception des états du jeu</b>	<b>4</b>
2.1	Description des états . . . . .	4
2.2	Conception logiciel . . . . .	5
<b>3</b>	<b>Changelog</b>	<b>7</b>
3.1	Livrable, Jalon 1 . . . . .	7
3.2	Livrable, Jalon 1 . . . . .	7

# Chapitre 1

## Objectif

### 1.1 Présentation générale



By David Revoy / Blender Foundation - Own work, CC BY 3.0

Tales of Kornwal est un jeu vidéo basé sur les mêmes règles de Fallout Tactics, *i.e* un jeu d'aventure doté d'un système de combat en tour par tour, permettant aux joueur de progresser et interagir avec un univers à l'allure originale à mi-chemin entre le médiéval-fantastique et le post-apocalyptique.

Les interactions seront basées sur un système de gestion d'inventaire et d'au moins une caractéristique, qui feront office de modificateurs lors d'actions entreprises par le personnage (*e.g* : la caractéristique "Force" influera grandement sur les dégâts infligés par un ennemi ou par le héros, ainsi que l'utilisation de telle ou telle arme).

## 1.2 Règles du jeu

Le jeu pourra posséder plusieurs aspects dépendant de l'étude du cahier des charges :

- Déplacement d'un personnage sur une "zone" de la mappemonde, accédant aux différentes cases nord-sud-est-ouest de la map en cliquant sur l'une des extrémités de l'écran.
- Système de combat tour par tour : lors d'une rencontre avec un ennemi, la map se vide de tous les sprites autres que le personnage joueur et ses adversaires, laissant donc place au duel entre le héros et l'ennemi. Le joueur commencera en premier (sauf modification) et disposera de deux choix possibles : se déplacer d'une case dans la zone, ou attaquer l'ennemi, faisant baisser son capital de points de vie. Le nombre de points de vie retirés dépendra de la caractéristique FORCE du personnage, ainsi que de son ARME équipée. Le tour se termine après que l'une des actions suivante a été effectuée, laissant le tour à l'ennemi (qui se déroulera de la même manière).
- Système d'interaction avec les personnages joueur ou non-joueur (discussions, interface d'échange d'objets)

## Chapitre 2

# Description et conception des états du jeu

### 2.1 Description des états

Un état du jeu est défini par les éléments suivants :

- **Les éléments d'espace** (classe Space), élément immobiles permettant de définir les zones où les personnages peuvent se déplacer (*e.g* herbe, sable, neige...). Sur les bords de la zone (définie par la grille, constituée des éléments fixes Space et Wall), l'attribut booléen "changemap" est "True", ce qui permet au personnage de changer de zone.
- **Les éléments de mur** (classe Wall), définissant les zones où un personnage ne peut se déplacer (*e.g* mur)
- **Les personnages** (classe Character), éléments mobiles se déplaçant sur la grille (ou plus exactement sur les cases définies par les classes Space). Ces éléments peuvent être contrôlés soit par l'humain (Personnage joueur), soit par les IA (Personnages non joueurs). Les personnages possèdent trois status possibles :
  - **YOURTURN** : Le tour du personnage. Il aura le droit de dépenser des points de mouvements pour se déplacer sur la map, ou des points d'actions pour endommager les adversaires.
  - **HISTURN** : Le tour des autres personnages. Le personnage doit rester immobile et subir les actions des autres personnages jusqu'à ce que son tour arrive.
  - **DEAD** : Le personnage ne peut plus bouger, son tour n'arrivant jamais. Il est mort.

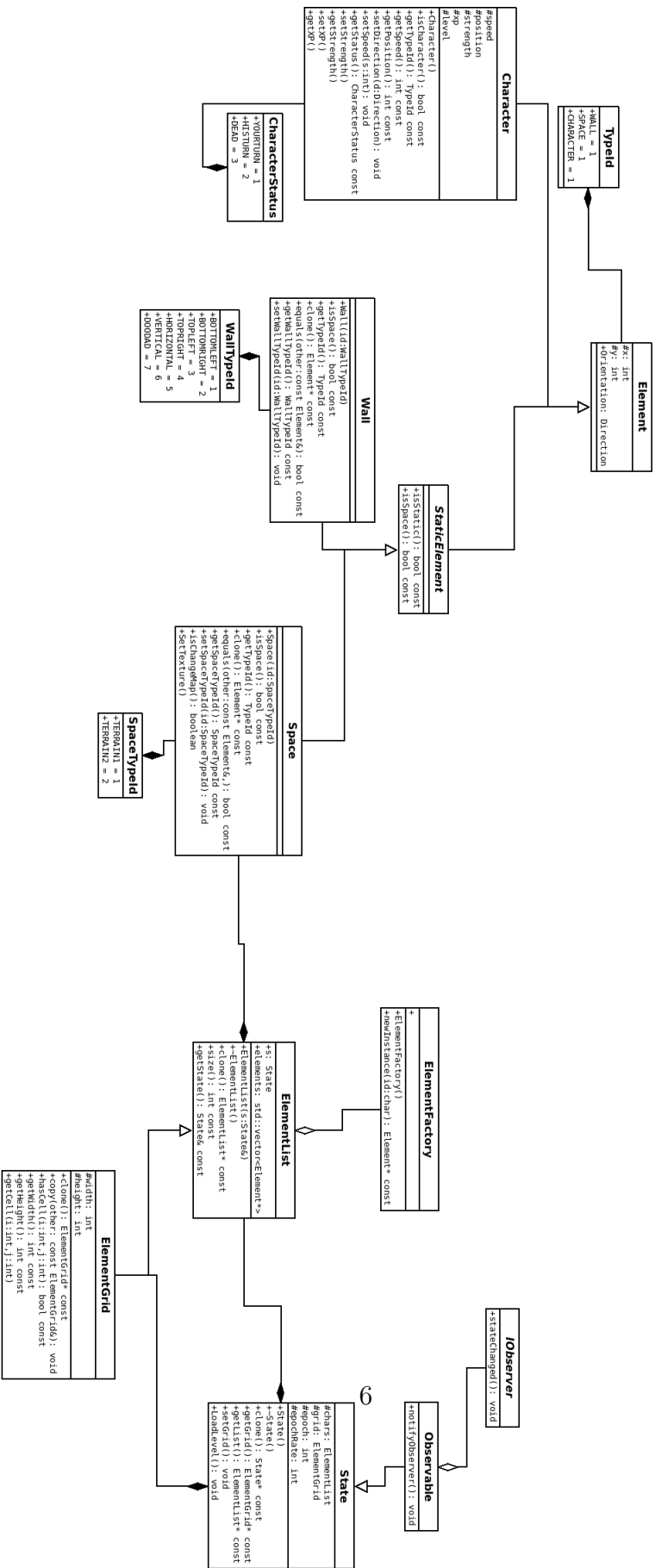
- Aux états fixes et mobiles nous rajoutons les états de **vitesse** (de mise à jour du jeu) et d'**époque** (l'état à un instant donnée). La vitesse étant donc un nombre d'époque donné par seconde.

## 2.2 Conception logiciel

La version 1.final a également été l'objet d'ajout de classes telles que :

- **ElementFactory** : Toute la hiérarchie d'ElementFactory sert à instancier de manière standardisée tout les éléments héritiers de la classe d'**Element** : ElementList sera principalement utilisée pour les personnages (=classe Character) tandis qu'ElementGrid instanciera les terrains (=classes Wall et Space)
- **Observers** : La hiérarchie des classes Observers (suivant le Pattern Observer) permet l'implémentation d'écouteurs, qui à leur tour permettront de permettre de réagir à un changement d'état (cette partie sera développée plus en profondeur durant le développement de la partie render)

La page suivante présente le diagramme d'état.



# Chapitre 3

## Changelog

### 3.1 Livrable, Jalon 1

- **Date** : Mercredi 28 Septembre 2016
- **Version** : 1.1
- **Nature de l'ajout sur le rapport** :
  - Ajout des textures et des effets sonores
  - Page de garde
  - Sommaire
  - Section 1.1 : Présentation générale
  - Section 2.2 : Règle du jeu
- **Modifications** : Aucune modification pour l'instant.

### 3.2 Livrable, Jalon 2

- **Date** : Jeudi 6 Octobre 2016
- **Version** : 1.final
- **Nature de l'ajout sur le rapport** :
  - Ajout du Chapitre 3 et des sections correspondantes.
- **Modifications** : Aucune modification des sections précédentes pour l'instant.