

Xarxaria



Leandro Saraiva Maia
Chemin Du Bornalet 29
1530, Payerne
leandro.SARAIVA-MAIA@cpnv.ch

Johan Volland
Champ Derrey 4
1083, Mézières
Johan.VOLLAND@cpnv.ch

Table des matières

Introduction	4
Scénario	4
Cadre, description et motivation	4
Organisation	4
Objectifs	4
Analyse	5
Déroulement du jeu	5
Introduction	5
Choix	5
Le système de combat	5
Options	6
Schéma de décision	6
Maquette	6
Fichier externe (Trop grand)	6
Uses Cases	6
Scénarii	7
Cahier des charges détaillé	11
But	11
Définition de l'application	11
Analyse et conception d'application	11
Implémentation de l'application	11
Contrôle continu avec notre projet	11
Phase de validation	11
Stratégie de test	11
Budget initial	13
Planification	13
Historique	13
Conception	14
Caractéristiques des personnages	14
Points de vie (HP)	14
Points d'armure	14
Points d'agilité	14
Points de chance	14
Inventaire	14
Formatage du texte	14
MCD	17
MLD	17

Implémentation	17
Diagramme de classe	18
Base de donnée	18
Problèmes rencontrés	18
Mise en service	18
Connexion à la base de donnée	19
Dossier de conception	19
Dossier de réalisation	20
Description des tests effectués	20
Erreurs restantes	21
La 2ème Version (2.0)	22
Introduction	22
Les choix	22
Le système de combat	22
L'influence	22
Historique des actions	22
Glossaire	22
Pages "hub"	23
Mise en service	23
Installation	23
Liste des documents fournis	23
Conclusions	23
Annexes	24
Sources – Bibliographie	24
Journal de bord de chaque participant	25
Manuel d'Utilisation	28

1 Introduction

1.1 Scénario

Fichier externe (Trop grand) ([Disponible aussi sur google docs](#))

1.2 Cadre, description et motivation

Par groupe de deux personnes, nous devons réaliser une application en C#. Nous avons choisi de coder un livre interactif où le joueur pourra choisir la direction que prend l'histoire via des choix à la fin de chaque page.

Les diverses pages ainsi que leurs choix seront stockées dans une base de donnée.

Un système d'inventaire sera implémenté afin de permettre certains choix.

Un système de création de personnage, de combat ou d'influence sera ajouté si le temps nous le permet.

Ce projet nous permettra d'entraîner notre compréhension de la programmation orientée objet, le C# et la gestion de projets.

1.3 Organisation

	Saraiva Maia, Leandro	Voland, Johan	Julien Ithurbide
Chef de projet			X
Connexion DB		X	
Formatage du texte	X		
Scénario	X	X	
Image de la DB		X	

1.4 Objectifs

- Afficher du texte, des images.
- Changer de page en cliquant sur des liens
- Un menu pour voir l'inventaire et les caractéristiques du joueur
- Un petit scénario pour pouvoir compléter le jeu
- Avoir un du texte formaté dans le contenu des pages (cf chapitre texte formaté)

2 Analyse

2.1 Déroulement du jeu

2.1.1 Introduction

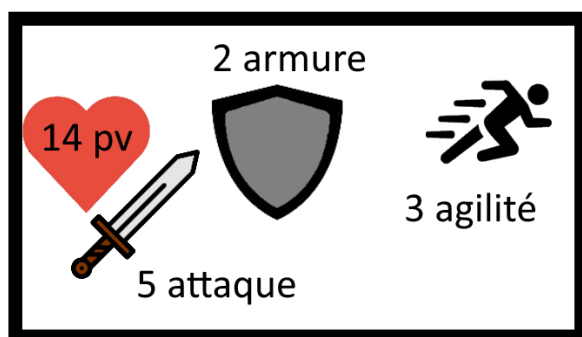
Le joueur est introduit au jeu par un texte explicatif, il choisit ses caractéristiques et son nom.

2.1.2 Choix

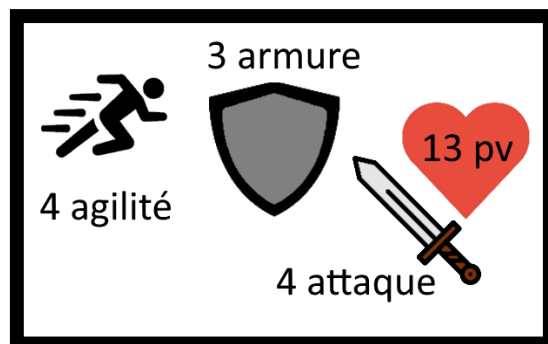
Tout au long de l'aventure, le joueur sera présenté à des pages. Une page est constituée d'une image et d'un texte. Le texte contient des choix représentés par un lien en bleu cliquable. Chaque choix aura les actions possibles suivantes :

- Changer de page
- Combat
- Obtenir/Perdre un objet d'inventaire
- Changer une caractéristique (pv, agilité, force, armure, chance)

2.1.3 Le système de combat



Personnage A



Personnage B

Les combats ne seront disponibles que dans la version 2.0. Un combat se déroule au tour par tour. Le personnage qui a le plus grand score d'agilité commence. Si les deux personnages ont les mêmes points d'agilité, le personnage avec le plus de points de chance commence. Si les deux personnages ont le même score de chance, le personnage qui commence est choisi aléatoirement.

A chaque tour, un personnage attaque l'autre avec le calcul suivant : points attaque - points armures de l'ennemi = pv retiré. Chaque personnage a une chance de faire un coup critique et d'infliger un point de dégât supplémentaire (1 chance = 5% de chance de coup critique).

Prenons le cas d'un combat entre le personnage A et le personnage B :

Comme B a plus d'agilité, c'est lui qui commence. B attaque alors A et lui retire des pv (4 points d'attaque de B - 2 points d'armure de A = 2 pv retiré à A).

Ensuite c'est à A d'attaquer B (5 points d'attaque de A - 3 points d'armure de B = 2 pv retiré à B).

Ainsi de suite jusqu'à ce qu'un personnage n'a plus de pv. L'armure, l'attaque et l'agilité des personnages ne changent pas, il y a uniquement les pvs qui changent.

2.1.4 Options

Un système de sauvegarde sera disponible. Des options diverses sont disponibles tels qu' agrandir/rétrécir le texte et l'alignement horizontal de celui-ci.

2.2 Schéma de décision

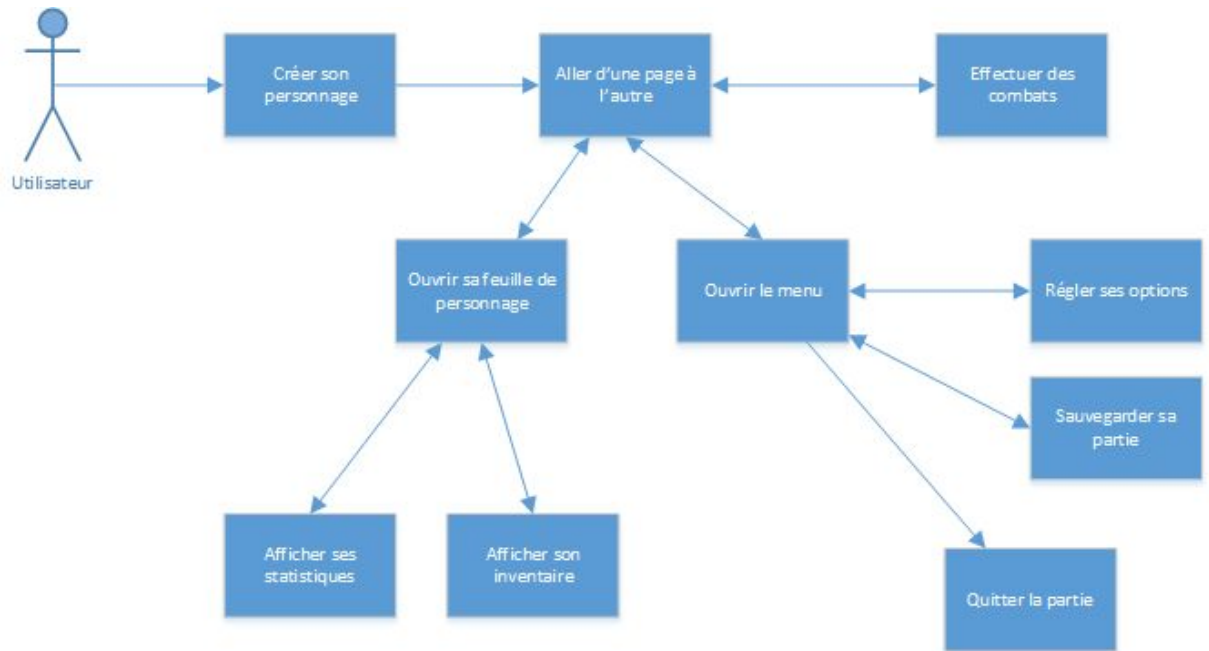
Le schéma de décision est un organigramme servant à indiquer, de façon général, le déroulement du scénario ainsi que les différentes actions possibles pour le joueur.

Fichier externe (Trop grand).

2.3 Maquette

Fichier externe (Trop grand)

2.4 Uses Cases



2.5 Scénarii

Créer son personnage		
Action	Particularité	Réaction
Appuyer sur les boutons des statistiques		Augmente/diminue la valeur
Appuyer sur le bouton "Commencer l'aventure"	Le champ "Nom du joueur" n'est pas rempli	Message d'erreur, le jeu ne commence pas
	Le champ "Nom du joueur" est rempli	Le jeu commence

<i>Aller d'une page à l'autre</i>		
Action	Particularité	Réaction
Entrée dans la page		Charge les données de la page
Cliquer sur un lien dans le texte	Le lien fait référence à une autre page	Transporte le joueur dans la page sélectionnée
	Le lien fait référence à un combat	Ouvre la fenêtre de combat et début du scénario "Effectuer des combat"
	Le lien fait référence à une modification de caractéristique	Modifie la caractéristique concernée
	Le lien fait référence à l'ajout/suppression d'un objet	Ajoute/supprime l'objet concerné
Cliquer sur le bouton "Feuille de personnage"		Ouvre la fenêtre du personnage et début du scénario "Ouvrir sa feuille de personnage"
Cliquer sur le bouton "Menu"		Ouvre la fenêtre du menu et début du scénario "Ouvrir le menu"

<i>Effectuer des combats</i>		
Action	Particularité	Réaction
Entrée dans la fenêtre de combat		Charge les statistiques des deux personnages
Cliquer sur le bouton "Prochain tour"		Les PV des personnages sont réduits (voir système de combat)
Les PV du joueur atteignent 0		Un message de fin de partie apparaît, l'application se ferme
Les PV de l'ennemi atteignent 0		La fenêtre de combat se ferme

<i>Consulter sa feuille de personnage</i>		
Action	Particularité	Réaction
Entrée dans la fenêtre du personnage		Charge le nom du joueur, ses statistiques actuelles et son inventaire
Cliquer sur le bouton "Continuer"		Ferme la fenêtre de personnage

<i>Ouvrir le menu</i>		
Action	Particularité	Réaction
Entrée dans le fenêtre du menu		Charge les boutons du menu
Cliquer sur "Continuer"		Ferme la fenêtre du menu
Cliquer sur "Sauvegarder"		La partie est sauvegardée
Cliquer sur "Options"		Ouvre la fenêtre d'option et début du scénario "Régler ses options"
Cliquer sur "Quitter"		Ferme l'application

<i>Régler ses options</i>		
Action	Particularité	Réaction
Entrée dans la fenêtre des options		Charges les données des options
Cliquer sur les boutons d'options		Applique les changements
Cliquer sur le bouton "Quitter"		Ferme la fenêtre d'option

2.6 Cahier des charges détaillé

2.6.1 But

Le but de ce module est de réaliser une application en C# par groupe de 2.

2.6.2 Définition de l'application

Tout d'abord il faut définir les tenants et les aboutissants de l'application choisis en effectuant une analyse complète. L'application doit ensuite être validée par l'enseignant. Il est important de prendre en compte les 8 semaines à disposition à raison de 5h15 par semaine. En plus d'utiliser les connaissances en programmation acquises, le projet devra inclure une nouvelle notion qui fera l'objet d'une analyse un peu plus poussée (Connexion à une base de données, utilisation du réseau, utilisation d'une API externe). A l'issue de cette phase qui durera une semaine, la proposition devra être remise à votre enseignant.

2.6.2.1 Analyse et conception d'application

Après acceptation de la proposition par l'enseignant, il faudra commencer une phase d'analyse et de conception. Cette phase est obligatoire avant de commencer l'implémentation de l'application et fera l'objet d'un rapport à transmettre à l'enseignant.

La planification du projet se fera selon la méthode agile sur github.

2.6.2.2 Implémentation de l'application

En parallèle de la phase d'analyse et de conception il faudra passer à la phase d'implémentation proprement dite.

2.6.2.3 Contrôle continu avec notre projet

Développer et tester le projet par itération successive.

2.6.3 Phase de validation

Délivrer le projet à l'enseignant. Le livrable sera composé de l'application et d'une documentation finale.

Une présentation orale du produit sera faite en classe.

2.7 Stratégie de test

Test	Testeurs	Résultat attendu	Résultat actuel
Les liens sont fonctionnels.	Johan, Leandro, famille, Rui	Quand on clique sur un lien, l'action demandée s'effectue correctement.	fonctionne
Les musiques se jouent correctement.	Johan, Leandro, famille, Rui	Les musiques se lancent et se jouent au bon moment.	fonctionne
Les caractéristiques se modifient correctement.	Johan, Leandro, famille, Rui	Quand on clique sur un lien ou lors d'un combat, les caractéristiques sont bien modifiées.	fonctionne
Les options sont fonctionnelles.	Johan, Leandro, famille, Rui	Quand on règle les options, les modifications s'effectuent correctement.	fonctionne
Le système de sauvegarde est fonctionnel.	Johan, Leandro, famille, Rui	On peut sauvegarder sa partie actuelle et par la suite la recharger sans qu'aucune donnée ne soit perdue.	fonctionne
Le système de combat est opérationnel.	Johan, Leandro, famille, Rui	Toutes les différentes fonctionnalités du combats fonctionnent correctement.	fonctionne
Les pages s'affichent correctement.	Johan, Leandro, famille, Rui	Le texte, l'image et le lien d'une page sont correctement affichés.	fonctionne

2.8 Budget initial

Aucun budget n'est alloué à ce projet.

2.9 Planification

Voir sur ce [lien](#)

2.10 Historique

Il n'y a eu aucune modification du cahier des charges..

3 Conception

3.1 Caractéristiques des personnages

Les personnages (joueurs / ennemis) possèdent tous les caractéristiques suivantes :

3.1.1 Points de vie (HP)

Représente la vie du personnage, si elle tombe à 0 le personnage est mort.

3.1.2 Points d'armure

Définit l'absorption des dégâts lors d'un combat. Notez que lorsqu'un personnage avec de l'armure prend des dégâts, la valeur d'armure ne change pas.

3.1.3 Points d'agilité

Définit qui commence à attaquer lors d'un combat. Cette caractéristique peut aussi être utilisé hors combat pour d'autres actions (escalader une tour, etc.)

3.1.4 Points de chance

Définit qui commence à attaquer lors d'un combat si les points d'agilité des deux personnages sont égaux. Cette caractéristique peut aussi être utilisé hors combat pour d'autres actions (duper des gardes, etc.)

3.2 Inventaire

Tout au long de l'histoire, le joueur va obtenir des objets. Ces objets seront ensuite utile (pièces d'or pour acheter un objet, manteau pour résister au froid, etc.). L'inventaire peut être accédé à tout moment dans le jeu en cliquant sur le bouton "Stats / Inventaire".

3.3 Formatage du texte

Chaque lien affiché dans le texte est un texte qui a été préalablement formaté.

Lors du chargement du texte, le programme vérifie si un caractère '<' ou '>' est dans le texte. Si c'est le cas, le programme va alors lire les valeurs contenu entre les balises. Chaque valeur est séparée par une virgule. La première valeur est "l'actionId", c'est un numéro qui permet d'identifier l'action à effectuer, voici la liste des "actionId" de la version 1.0:

- 0) Changement de page

- 1) Changement du nombre d'objets
- 2) Changement des points de vie
- 3) Définition des points de force
- 4) Changement des points d'armure
- 5) Changement des points d'agilité
- 6) Changement des points de chance
- 7) Affichage d'un message
- 8) Combat contre un ennemi
- 9) Changement des points de force

Notez que pour les points de force on définit une valeur, on ne soustrait ou n'ajoute rien. Nous avons choisi de faire ainsi pour être un peu plus réaliste. Acheter 20 dagues ne doit pas ajouter à chaque fois de la force. Les pièces d'armure trouvables dans le jeu se portent à des endroits différents (il n'y aura pas deux plastrons dans le jeu, chaque pièce d'armure trouvée se mettra dans une partie différente du corps, on trouvera une épaulière, un casque, des bottes, etc.) ce qui permet d'éviter le problème où le joueur achèterait 20 casques pour avoir beaucoup d'armure.

La deuxième valeur est le texte affiché dans la page qui sera cliquable (souligné et en bleu).

La troisième valeur est la valeur de l'action. Pour l'actionId du changement de page, cette valeur sera utilisée pour définir la page à laquelle on se rend lorsqu'on cliquera sur le lien. Pour l'actionId du changement de point de vie, cette valeur sera utilisée pour définir le changement de points de vie du joueur (si la valeur = 3 alors on ajoute 3 points de vie. Si la valeur = -2 alors on soustrait 2 points de vie).

Pour l'actionId du changement du nombre d'objets, l'action nécessite une valeur supplémentaire. Pour ce faire, la troisième valeur est elle-même séparée en plusieurs valeurs avec des points-virgules cette fois. Pour mieux saisir le formatage du texte, voici quelques exemples de ce qui serait affiché avec :

Contenu du texte dans la base de données	<2,Vous perdez deux points de vie,-2>
Ce qui sera affiché	Vous perdez deux points de vie
Ce qui va être fait	Fait perdre 2 points de vie au joueur

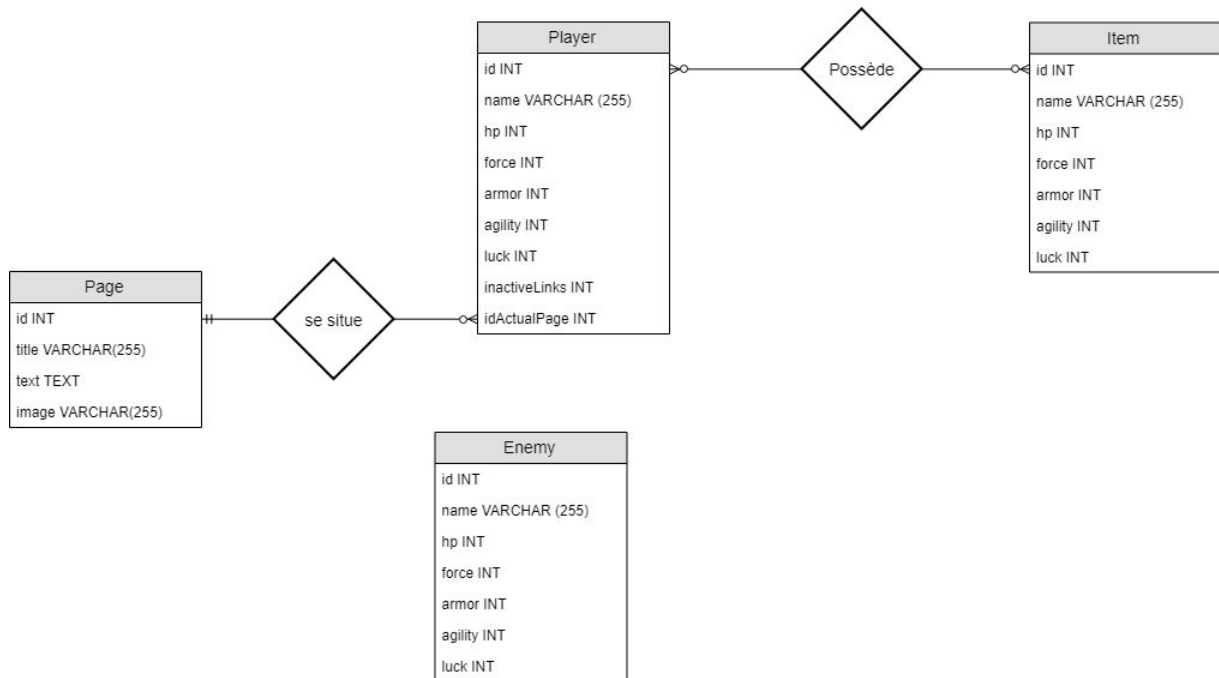
Contenu du texte dans la base de données	<1,Gagner trois pièces d'or,0;3>
Ce qui sera affiché	Gagner trois pièces d'or

Ce qui va être fait	Donne 3 pièces d'or au joueur (id de la pièce d'or = 0)
---------------------	--

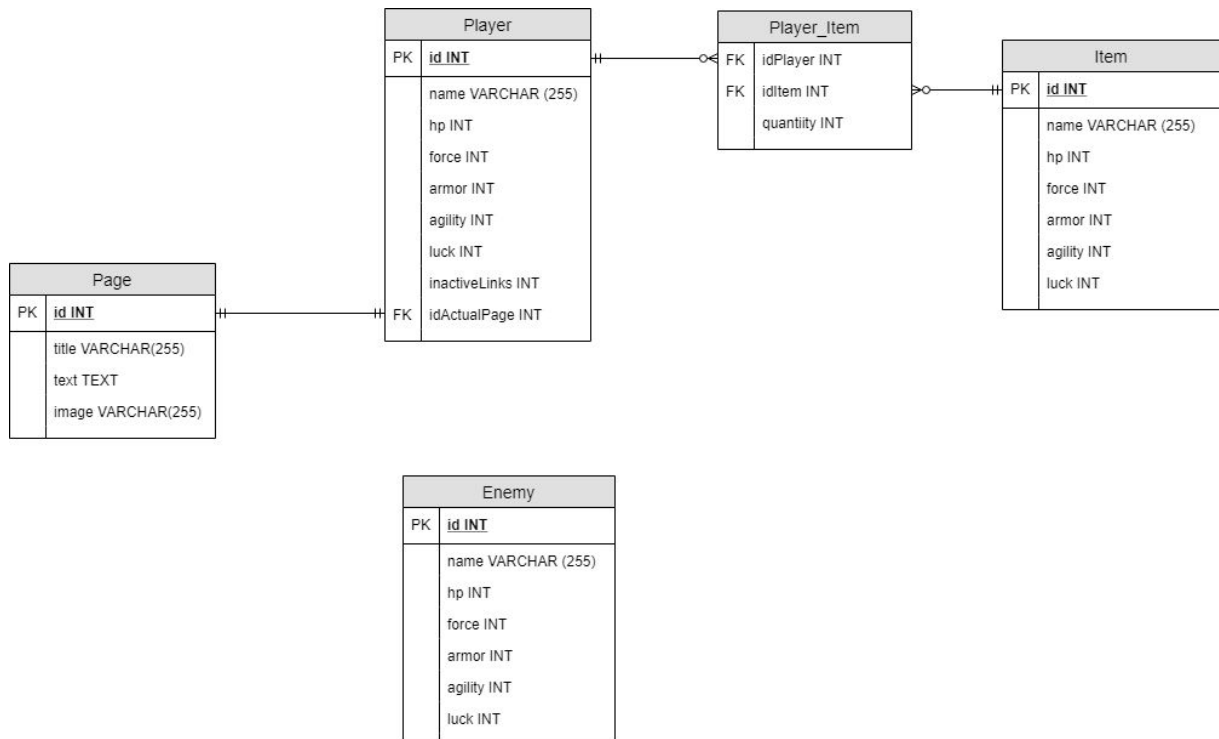
Contenu du texte dans la base de données	<1,Vous perdez votre manteau,8;-1>
Ce qui sera affiché	Vous perdez votre manteau
Ce qui va être fait	Soustrait le nombre de manteau du joueur de 1 (id du manteau = 8)

Contenu du texte dans la base de données	<0,Changer de page en cliquant ici,4>
Ce qui sera affiché	Changer de page en cliquant ici
Ce qui va être fait	Aller à la page 4

3.4 MCD



3.5 MLD



4 Implémentation

4.1 Diagramme de classe

Fichier externe

4.2 Base de donnée

Nous avons hébergé notre base de donnée sur internet grâce au service microsoft azure fourni par le CPNV pendant 1 mois.

Sur microsoft azure, il a fallu créer un serveur de base de donnée ainsi qu'une base de donnée. Il a fallu également modifier le pare-feu de ce serveur pour autoriser les connexions.

L'adresse de ce serveur est "xarxariadb.database.windows.net"

Cette application utilise une base de donnée sous microsoft SQL. Le langage est le Transact-SQL de microsoft. L'application se connecte donc sur le serveur de base de donnée grâce à l'adresse ci-dessus avec les informations de connexion suivantes :

Les informations de connexion utilisé dans l'application sont :

- Nom de l'utilisateur : xarxariaLogin
- Password : Pa\$\$w0rd
- Initial catalog : XarxariaDB

4.3 Problèmes rencontrés

4.3.1 Mise en service

Ceci s'appliquait uniquement lorsque la base de données était en local

Notre application nécessite une base de données. Nous avons tenté, pour effectuer des tests, de donner un exécutable de notre application à des collègues pour qu'ils essayent notre application, cependant ils rencontrèrent beaucoup de problèmes pour installer la base de données. En effet il faut installer deux logiciels :

- Microsoft sql server express 2017
 - Pour héberger un serveur de base de données
- SSMS
 - Pour lancer le script de création de la base de donnée

Beaucoup de nos testeurs ont rencontrés des problèmes lors de l'installation de Microsoft sql server express 2017. Certaines librairies microsoft n'était pas à jour et d'autres problèmes apparaissent également lors de la connexion au serveur de base de données.

C'est également une opération longue et fastidieuse.

4.3.2 Connection à la base de donnée

Ceci s'appliquait uniquement lorsque la base de données était en local

Nous avons perdu du temps pour connecter l'application avec la base de donnée. L'utilisateur crée n'avait pas les droits requis pour modifier les données de la base de données.

Lorsqu'une base de donnée est créée, l'utilisateur ayant les droits de modifications est l'utilisateur windows (Authentication windows).

Pour résoudre ce problème, nous avons procédé ainsi :

- Après avoir installé Microsoft sql server 2017
- Dans Microsoft SQL Server Management Studio 17 (SSMS)
- Se connecter à localhost avec l'authentification Windows
- Dans le dossier Sécurité > Connexions
- Faire clic droit et choisir "Nouvelle connexion"
- Dans l'onglet "General", cocher "Authentication SQL Server", insérer un nom, insérer un mot de passe et choisir la base de donnée liée
- Dans l'onglet "Server role", cocher sysadmin
- Dans l'onglet "User mapping", cocher la base de donnée lié et cocher "db_owner"
- Dans l'onglet "Status", mettre "Login" à Enabled et mettre "Permission to connect DB Engine" à GRANT
- Appuyer sur OK
- Faire clic-droit sur le server sql et choisir "propriétés"
- Dans l'onglet "Security" cocher "SQL Server and Windows Authentication mode"
- Déconnecter et reconnecter et c'est bon

4.4 Dossier de conception

Logiciels utilisés :

Programmation

- Visual Studio 2017

Base de données

- Microsoft SQL Server Express 2017
- Microsoft SQL Server Manager Studio

Documentation

- Visio 2016
- Word 2016
- Balsamiq
- Draw.io

Base de données :

Player : Contient toutes les informations relatives au joueur, comme ses caractéristiques par exemple.

Page : Contient toutes les informations servant à afficher correctement les différentes pages.

Item : Contient les différents objets du jeux ainsi que leurs caractéristiques.

Enemy : Comme la table "Player", contient les informations relatives aux différents ennemis trouvables dans le jeux.

4.5 Dossier de réalisation

Notre projet se trouve sur [github](#). Dans la branche master se trouve 3 dossiers :

- Code source
 - Contient la solution visual studio de la version 2.0
- Documents
 - Contient toutes les annexes de la documentation
- Livrable
 - Documentation.pdf
 - Diagramme de classe.png
 - Dossier contenant doxygen en html
 - Manuel d'utilisation.pdf
 - Schéma de décision.pdf
 - Maquette.pdf
 - Scénario.pdf

4.6 Description des tests effectués

Test	Testeurs	Résultat attendu	Résultat actuel
Les liens sont fonctionnels.	Johan, Leandro, famille, Rui	Quand on clique sur un lien, l'action demandée s'effectue correctement.	fonctionne
Les musiques se jouent correctement.	Johan, Leandro, famille, Rui	Les musiques se lancent et se jouent au bon moment.	fonctionne
Les caractéristiques se modifient correctement.	Johan, Leandro, famille, Rui	Quand on clique sur un lien ou lors d'un combat, les caractéristiques sont bien modifiées.	fonctionne
Les options sont fonctionnelles.	Johan, Leandro, famille, Rui	Quand on règle les options, les modifications s'effectuent correctement.	fonctionne

Le système de sauvegarde est fonctionnel.	Johan, Leandro, famille, Rui	On peut sauvegarder sa partie actuelle et par la suite la recharger sans qu'aucune donnée ne soit perdue.	fonctionne
Le système de combat est opérationnel.	Johan, Leandro, famille, Rui	Toutes les différentes fonctionnalités du combats fonctionnent correctement.	fonctionne
Les pages s'affichent correctement.	Johan, Leandro, famille, Rui	Le texte, l'image et le lien d'une page sont correctement affichés.	fonctionne

4.7 Erreurs restantes

Il ne reste aucune erreur.

4.8 La 2^{ème} Version (2.0)

Si tous les objectifs de la première version ont été complétés et qu'il reste du temps, une 2^{ème} version du projet sera faite.

Voici une liste exhaustive des fonctionnalités ajoutées lors de la deuxième version :

4.8.1 Introduction

Le joueur pourra choisir des objets de départ.

4.8.2 Les choix

Les choix non disponibles (par exemple: pas assez de force) seront grisés. Les combats deviendront obligatoires (On ne pourra pas juste éviter de cliquer sur le lien)

4.8.3 Le système de combat

Les armes auront des dégâts variables ((Dégâts de 4 à 6 au lieu de 5 par exemple)

La caractéristique d'agilité d'un personnage permet d'avoir une chance d'esquiver un coup, 1 point d'agilité = 5% de chance d'esquiver

Ces modifications permettront de créer des situations inattendues car on pourra savoir qui a le plus de chance de gagner mais pas qui gagnera à coup sûr.

4.8.4 L'influence

L'influence est une nouvelle caractéristique du joueur, elle fait en sorte que les actions ont des conséquences, par exemple :

Si lors de l'aventure, vous donnez des pièces à un mendiant, l'application se souviendra de cette action. Si vous recroisez ce mendiant, il peut se montrer amical ou hostile selon les actions antérieures.

L'influence agit comme un espèce de "karma"

4.8.5 Historique des actions

Fonctionnalité permettant de revenir facilement en arrière pour modifier son choix.

4.8.6 Glossaire

Un glossaire permettra d'en apprendre plus sur l'univers de Xarxaria. Contiendrait des descriptions des différents lieux et personnages

4.8.7 Pages “hub”

Page dédié pour une fonction, par exemple un magasin pour échanger ses pièces avec de l'équipement.

On peut aussi imaginer des événements aléatoires comme une attaque de bandit

5 Mise en service

5.1 Installation

Télécharger la release 2.0 disponible sur github [ICI](#)

(Voici également un [lien vers la release 1.0](#))

Une connexion internet est nécessaire.

Il suffit simplement de lancer l'exécutable “Xarxaria.exe” et le programme se lance.

5.2 Liste des documents fournis

- Manuel d'Utilisation au format .pdf
- Documentation au format .pdf
- Documentation disponible en ligne sur [google docs](#)
- Diagramme de classes au format .png
- Dossier Doxygen au format .html
- Schéma de décision au format .pdf
- Maquette au format .pdf
- Le scénario au format .pdf

6 Conclusions

Nous sommes parvenus à finir la version 1.0 et avons ajouté certaines fonctionnalités de la version 2.0 : Les dégâts variables et les chances d'esquive. De plus, certaines fonctionnalités pas présentes dans la version 1.0 et 2.0, comme des musiques ainsi qu'un bouton d'aide ont été ajoutées. Nous sommes parvenu à remplir notre objectif en proposant un produit fonctionnel. Néanmoins, certains points peuvent être améliorés. Actuellement, les choix sont basés sur le fair play des joueurs. Pour le moment, il n'y a aucune fonction qui vérifie que les joueurs ont les valeurs requises pour les tests (test d'agilité par exemple) ou encore, une fonction qui oblige un joueur à effectuer un combat.

Voici la liste des idées dans la conception qui n'ont pas été implémentés :

- L'influence
- Les dégâts variables
- Historique des actions
- Glossaire
- Des choix non disponibles si on a pas les caractéristiques requises
- Choisir des objets de départ

Pour la suite, nous avons des idées pour améliorer ce projet :

- Les actions et les tests beaucoup moins basés sur le fair play
- Une gestion de l'inventaire plus libre (on peut choisir les armes que l'on veut équiper).
- Repenser le schéma de décision, il y a quelques lacunes concernant les pages de "hub" où l'on peut effectuer plusieurs fois le même choix.

7 Annexes

7.1 Sources – Bibliographie

Images

- Google images (Il y a des watermark sur certaines images)

Effets sonore

- Click.wav : www.freesound.org
- Hover.wav : www.freesound.org

Musiques

- Drake_battle.mp3 : [Dragon's Dogma OST: 2-20 Dragon Battle](#)
- Boss_battle.mp3 : [Dragons Dogma OST\(43\) Mini Boss Battle Theme](#)
- Small_battle.mp3 : [Battle of Wesnoth - Battle](#)
- Ambiance.mp3 : [Battle of Wesnoth - Journey's End](#)

Autres

- [Extension pour les liens Rich text box](#) (modifié)

7.2 Journal de bord de chaque participant

Johan :

Date	Durée	Activité	Remarques
20.11.18	1h30	Choix du projet et 1ère analyse	Fait avec Leandro
21.11.18	25 min	Ouverture d'un google doc et arrangement de la doc (ajouter et supprimer des chapitres)	Fait avec Leandro
21.11.18	1h	Conception et création du MCD, mise en place de GIT	
22.11.18	-	Absence	Passage de la théorie voiture
27.11.18	1h30	Recherches diverses sur la base de données	Fait avec Leandro
28.11.18	1h30	Création et connection de la base de données	
29.11.18	40 min	Requête insert fonctionnel	Lecture des joueurs dans la base de données
29.11.18	1h	Conception	Fait avec Leandro
29.11.18	30 min	Création du MCD et du MLD	Il y a une erreur : dans Page, title est un VARCHAR et non un INT
04.12.18	10 min	Arrangement du MCD et du MLD	
04.12.18	45 min	Fonctions de connection à la base de données et création de la classe Page	
04.12.18	30 min	Arrangement du merge	fait avec Leandro
04.12.18	15 min	Création de la branche graphismes dans Git et ajout d'un dossier qui contiendra les images du jeu	
05.12.18	30 min	Création des Uses Cases	
05.12.18	1h	Création des scénarii	
06.12.18	30 min	Finition des scénarii et modification de la maquette	
06.12.18	1h25	Création de l'affichage de l'image	la base de données a été légèrement modifiée, la fonction n'est pas terminée
11.12.18	45 min	Finition de l'affichage de l'image	il faut redimensionner les images

11.12.18	20 min	Redimensionnement des images	
12.12.18	1h30	Création de pages	
13.12.18	2h15	Absence	Examen
18.12.18	1h30	Création de pages	
19.12.18	1h30	Création des pages 10 à 12	
19.12.18	3h	Ecriture du scénario	
20.12.18	1h45	Création de la fonction de sauvegarde	
20.12.18	45 min	Création de la release 1.0	fait avec Leandro
08.01.19	1h30	Ecriture du scénario	
09.01.19	1h20	Ajout de nouvelles page	
09.01.19	10 min	Ecriture du scénario	
10.01.19	1h30	Ecriture du scénario	
10.01.19	45 min	Modification du schéma de décisions et redéfinitions des statistiques des différents élément (armes, ennemis, etc)	fait avec Leandro
15.01.19	1h	Écriture du scénario	
15.01.19	30 min	Mise à jour de la base de données	
16.01.19	1h30	Mise à jour de la base de données	
17.01.19	1h30	Écriture du scénario	
18.01.19	1h30	Écriture du scénario	
22.01.19	1h30	Fin du scénario et ajout de nouvelles pages	
23.01.19	1h30	Ajout de nouvelles pages	
24.01.19	20 min	Ajout des dernières pages	
24.01.19	30 min	Test des différentes pages	
24.01.19	1h25	Mise à jour de la documentation	

Leandro :

Date	Durée	Activité	Remarques
20.11.18	1h30	Choix du projet et première analyse.	Fait avec Johan
21.11.18	25 min	Ouverture d'un google doc et arrangement de la doc (ajouter et supprimer des chapitres)	Fait avec Johan
21.11.18	25 min	Ouverture du git et planning initial	
21.11.18	40 min	Objectifs et conception version 2	
22.11.18	30 min	Ajout du schéma de l'application et du schéma de décision au projet git	Schéma version 1.0
22.11.18	1h	Création des formulaires dans le projet C#	Selon le schéma v1.0
22.11.18	1h	Création des système de liens cliquable	
27.11.18	1h30	Recherche diverses sur la base de données	Fait avec Johan
28.11.18	1h30	Formattage du texte avec lien cliquable	
29.11.18	40 min	Le texte change lorsqu'on clique sur un lien	
29.11.18	1h	Conception	Fait avec Johan
29.11.18	30 min	Déroulement et caractéristiques perso	
04.12.18	20 min	Ajustement de la maquette	Ajout du formulaire d'introduction et de combat
04.12.18	20 min	Analyse / conception divers	Surtout formatage
04.12.18	20 min	Planification semaine 3	
04.12.18	30 min	Arrangement du merge	Fait avec Johan
05.12.18	30 min	Problème rencontré pour la base de donnée	
05.12.18	1h	Avancement du scénario	
06.12.18	1h30	Formatage du texte, les balises ont maintenant des "action id" qui permettent de faire des actions	L'action crée est l'ajout d'objet dans l'inventaire du joueur
06.12.18	45 min	Finition du scénario 1.0	
11.12.18	1h	Les liens deviennent inactifs après avoir cliqué dessus	
12.12.18	45 min	Cahier des charges détaillé	

12.12.18	45 min	Ajout de commentaires dans le code	commentaires Doxygen / regions / par fichier
13.12.18	45 min	Ajout de nouveaux formulaires	Intégralement commentés Introduction, Start, LoadSelection, Combat, Options
13.12.18	1h30	Formulaire d'introduction fonctionnel	Ajout d'un joueur dans la base de données
13.12.18	1h30	Formulaire "Start" fonctionnel	Chargement d'une sauvegarde, fait au domicile
18.12.18	1h30	Refonte graphique des formulaires	
19.12.18	1h30	Résolution de bugs et implémentation de l'audio	
20.12.18	1h45	Finitions de l'audio et de l'aspect graphique	
20.12.18	45 min	Création de la release 1.0	fait avec Johan
08.01.19	1h30	Finition du formulaire d'introduction	Robustesse et ergonomie
09.01.19	1h30	Correction de bugs du formulaire principale	Text zoomable avec CTRL + molette. Le texte ne se remettait pas en haut lors du changement de page
10.01.19	2h15	Aspect du formulaire de combat	Bouton + son
15.01.19	1h30	Logique du formulaire de combat	Avancement
16.01.19	1h30	Logique du formulaire de combat + Base de donnée	Finition combat + Modification de la base de données avec Johan
17.01.19	1h30	Adaptation du code pour les objets dans la base de données	+ Game over
22.01.19	1h30	Liens inactifs dans la base de données	Le joueur ne peut plus sauvegarder et revenir pour cliquer sur des liens en boucle
23.01.19	1h30	Coup critique + maquettes	
24.01.19	2h15	Ajout de musiques de fond	

7.3 Manuel d'Utilisation

Fichier externe (Trop grand)