YADUSTOCK PROJET TUTORE S3



IUT DIJON 2020

MARION TALLANDIER YOUSSEF EL KADI LOIS PAZOLA CORENTIN FAUTRE SAMUEL LACHAUD

SOMMAIRE

I - Introduction au sujetp 2
II - Etude des jeux existantsp 2-4
III - Les technologies envisagées
1. Présentation des technologiesp 4-6
2. Notre choixp 7
IV - Conception
1. Diagramme de classep 8 -13
2. Diagramme de cas d'utilisationp 14
3. Diagrammes de séquencep 15
V - Maquettesp 16-19

INTRODUCTION AU SUJET

Le projet Yadustock porte sur la création d'un serious game. Le joueur pourra contrôler ses propres décisions pour gérer l'entreprise que nous lui donnons entre les mains. Nous avons choisi que l'entreprise sera spécialisée dans la vente de jeux de cartes tel que les cartes Magic, Pokemon ou Yu-Gi-Oh. Ce choix s'explique par notre étude des serious game, rare son ceux qui propose ce genre de produits. En proposant des produits que les jeunes apprécient nous voulons les pousser à jouer à un serious game en s'amusant dans un milieu qu'ils connaissent.

Afin de profiter du jeu le joueur aura besoin d'une interface graphique lui permettant de gérer les parties les plus importantes d'une entreprise. Ainsi nous aurons un tableau de bord indiquant les informations sur les résultats du tour d'avant et les décisions prises par le joueur sur le tour actuel. Une page stock lui permettra de visualiser la quantité de chacun de ses produits et d'en fixer le prix de vente. Une page marché lui permettra d'acheter les produits auprès de son fournisseur. Enfin la page compte lui permettra de consulter son argent disponible. De plus cette page propose un achat de boost qui lui permettront d'améliorer ses résultats. L'interface doit être claire et intuitive pour que l'utilisateur puisse prendre en main facilement le mécanisme du jeu. Des images pour remplacer les mots sont nécessaires pour rendre plus agréable l'expérience de jeu.

Nos principaux objectifs se basent sur la bonne expérience du joueur, nous allons donc nous focaliser sur la simplification des actions menées par celui-ci en offrant une interface claire et agréable à utiliser. Nous voudrions par la suite ajouter davantage d'image pour remplacer des mots ou des textes ou tout simplement pour les accompagné et rendre le jeu plus agréable pour le joueur. Notre second objectif est d'offrir quelques possibilités pour pimenter l'expérience du joueur tel qu'un système de banque pour pouvoir emprunter et une page d'étude de marché que le joueur devra consulter pour choisir les boosts correspondants à la situation. Ainsi les boosts ne seront pas toujours positifs en fonction de ses choix.

ETUDE DES JEUX EXISTANTS

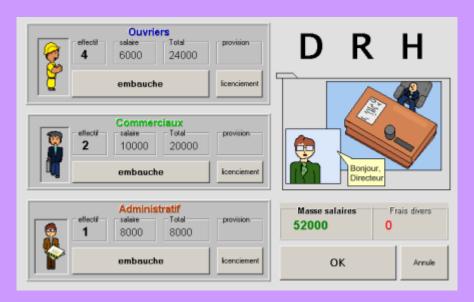


Figure 1 : Image du jeu Cartel Euro 3000



Figure 2: image du jeu Industry Manager Future Technologies



Figure 3: image du jeu Let's Play Capitalism 2

Durant nos recherches nous avons découvert beaucoup de serious game existants. Ces jeux avaient tous une interface très différente malgré le même principe de jeux. Mais grâce à ça nous avons pu remarquer que la plupart des produits proposés n'était pas toujours très intéressant à s'occuper d'après nous. Nous pensions que pour pousser des jeunes à s'intéresser à ce genre de jeu nous devrions utiliser des produits qu'ils connaissent et apprécient. Cela nous a pousser dans une recherche très rapide car étant nous même dans cette tranche d'âge il a été très simple de trouver un produit que beaucoup de jeunes apprécient.

LES TECHNOLOGIES ENVISAGEES

PRESENTATION DES TECHNOLOGIES

Pour notre projet un large choix de technologies s'offrait à nous, nous avons donc fait un tableau comparatif pour nous aider à choisir la meilleure pour remplir nos principaux objectifs. Notre tableau nous a donc permis de tirer ces informations :

LANGAGES DE PROGRAMMATION

Java + JavaFX		
Points positifs	Points négatifs	
 Accessible à l'entièreté de l'équipe 	 Java est un langage très lourd 	
 JavaFX est facile à prendre en main 		
 Programmes multi-plateformes 		
 JavaFX possède des fonctions 		
graphiques		

C# + .NET		
Points positifs	Points négatifs	
 Langage récent et grandement 	 Langage en apprentissage pour 	
documenté	chacun des membres de l'équipe	
 La création d'IHM est très intuitive et 		
très complète		
 Gagne beaucoup de temps à la 		
programmation		

Python + PyQT		
Points positifs	Points négatifs	
 Facilement lisible 	 Langage non-maitrisé par l'équipe 	
 PyQT possède des fonctions 	et nécessiterais donc un temps	
graphiques	d'apprentissage.	
 Rapide est léger 	 Syntaxe difficile à utiliser 	
 Possibilité de générer l'IHM 	 Non multi-plateforme 	
depuis un logiciel tiers (QT)	 Difficile de générer l'application 	
	finie	

IDE

Visual Studio		
Points positifs	Points négatifs	
 IDE Très complet proposant 	■ IDE très lourde à utiliser, très	
beaucoup de fonctionnalités	fournis.	
	■ Met du temps à compiler	

Netbeans		
Points positifs	Points négatifs	
 Implémente bien Java et JavaFX 	■ Met beaucoup de temps à la	
 Très facile de faire le produit final 	compilation les gros projets.	
(.jar)		

QT Creator		
Points positifs	Points négatifs	
 Générer facilement le code de 	 Code généré en XML, 	
l'IHM	transformation en format PyQT	
 Incorporation facile du CSS 	difficile.	
 Large gamme de paramètres pour 	 Lourd à utiliser 	
configurer l'IHM	 Difficultés de travail dans 	
	différentes versions de QT	

EDITION D'IMAGE

Gimp		
Points positifs	Points négatifs	
 Edition poussé possible 	■ Non intuitif	

PhotoFiltre		
Points positifs	Points négatifs	
 Intuitif, facile d'utilisation 	 Ne permet pas d'édition poussé 	

LOGICIELS DE PARTAGE DE FICHIERS

SVN (TORTOISE)		
Points positifs	Points négatifs	
 Intégration avec l'environn 	ement Peu pratique d'utilisation	
Windows	■ Peu intuitif	
 Privé 		
■ Hébergé à l'IUT		

Dépôt Git (GitHub)		
Points positifs	Points négatifs	
 Très permissif 	 Stocké sur des serveurs non 	
■ Bien organisé	possédés par l'iut.	
 Facile d'utilisation 	 L'organisation des fichiers et les 	
 Géré depuis un client Git et non 	commit sont plus longs à effectuer	
depuis les dossiers Windows		
 Disponible depuis un navigateur 		
web		
■ Privé		

NOS CHOIX

À la suite de l'analyse du tableau nous avons pris les décisions ci-dessous :

Nous utiliserons le langage C#/.NET pour la réalisation du produit final. Nos besoins étant énormément basé sur l'IHM ce langage est le plus approprié. De plus nous sommes en cours d'apprentissage de ce langage et même si cela paraît être un point négatif il s'avère en réalité être positif pour nous finalement. Cela nous permettra de nous améliorer et d'avoir des conseils assez facilement par des professeurs.

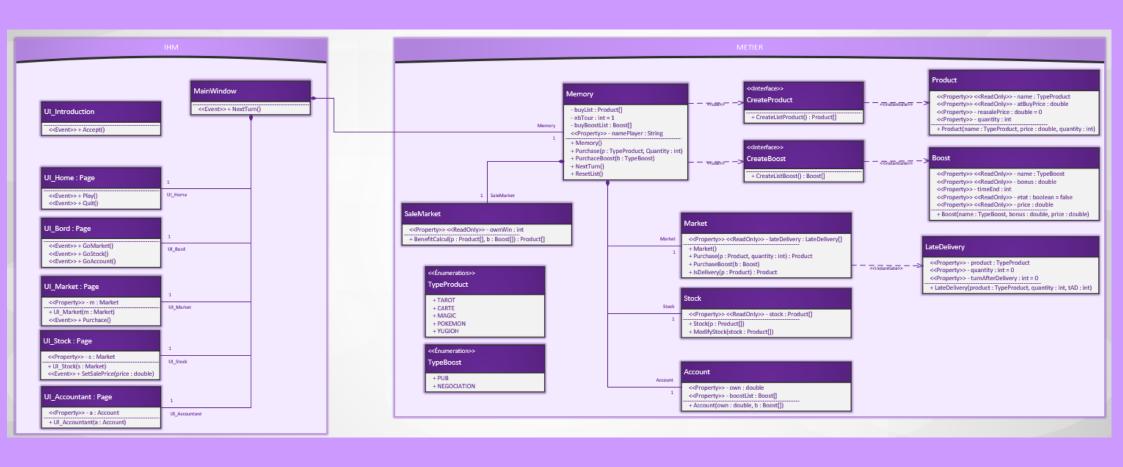
Nous utiliserons Visual Studio pour réaliser le produit car il s'agît du meilleur IDE pour C#. De plus tous les membres du groupe le possèdent et nous en apprenons l'usage en cours.

Nous utiliserons PhotoFiltre pour l'édition d'image car il s'agît du logiciel le plus simple d'utilisation. De plus certains membres de l'équipe connaissent déjà très bien celui-ci. Ayant choisi un style pixel art pour le jeu il est très simple pour nous de réaliser tous nos graphismes sur ce logiciel. Les maquettes seront très facilement réalisables dessus.

Nous utilisons un dépôt git qui a déjà été crée pour la réalisation de ce dossier et la suite du projet. Pour tous les membres il est plus simple d'utilisation qu'un dépôt SVN et nous avons beaucoup plus de choix de personnalisation pour documenter le dépôt. Un lien a été envoyé au professeur suivant notre projet tutoré afin qu'il puisse suivre de lui-même notre avancement si nécessaire.

CONCEPTION

DIAGRAMME DE CLASSE



IHM

MainWindow

Fenêtre qui comprendra toutes les différentes pages nécessaires au jeu.

L'event « next turn » permettra de passer au prochain tour et d'actualiser toutes les pages avec les informations nécessaires.

UI_Introduction

Page pop-up permettant au joueur de lire les explications et règle du jeu.

L'event « accept » permet de fermer le pop-up pour pouvoir jouer.

UI_Home

Page d'accueil permettant de jouer ou de quitter.

L'event « Play » permet de commencer le jeu.

L'event « Quit » permet de quitter le jeu.

UI_Bord

Page regroupant les informations du dernier tour du tour actuel. Permet de passer au tour suivant.

UI_Market

Page permettant d'acheter les différents produits.

L'event « purcharse » permet de valider les achats.

UI_Stock

Page permettant de voir l'état des stocks et de modifier le prix des produits.

L'event « SetSalePrice » permet de modifier le prix d'un produit.

UI_Accountant

Page permettant de voir l'état des comptes, voir les boosts possédés et les boots achetables.

METIER

Memory

Cette classe sauvegarde toutes les décisions du joueur pour modifier tout le nécessaire après l'action de fin de tour.

Create product

Interface permettant de créer un produit lors de la fin du tour pour l'ajouter au stock.

Create boost

Interface permettant de créer un boost lors de la fin du tour pour l'ajouter aux boosts possédés.

Product

Classe de produit regroupant les attributs d'un produit pour pouvoir regrouper par catégorie par la suite (cartes).

Boost

Classe des boost regroupant les attributs et les fonctions d'un boost.

Market

Classe du marché regroupant les attributs de celui-ci pour effectuer l'achat auprès du fournisseur. Celui-ci vend le produit au prix d'achat.

Stock

Classe du stock regroupant les attributs de celui-ci pour modifier les prix et vérifier le stock. Celui-ci contient le prix de vente du produit.

Accountant

Classe du compte du joueur regroupant l'argent possédé et les boosts possédés.

LateDelivery

Classe permettant de gérer les retards de livraison s'ils existent.

SaleMarket

Classe permettant de calculer le bénéfice ou déficit obtenu par le joueur au tour venant de se finir.

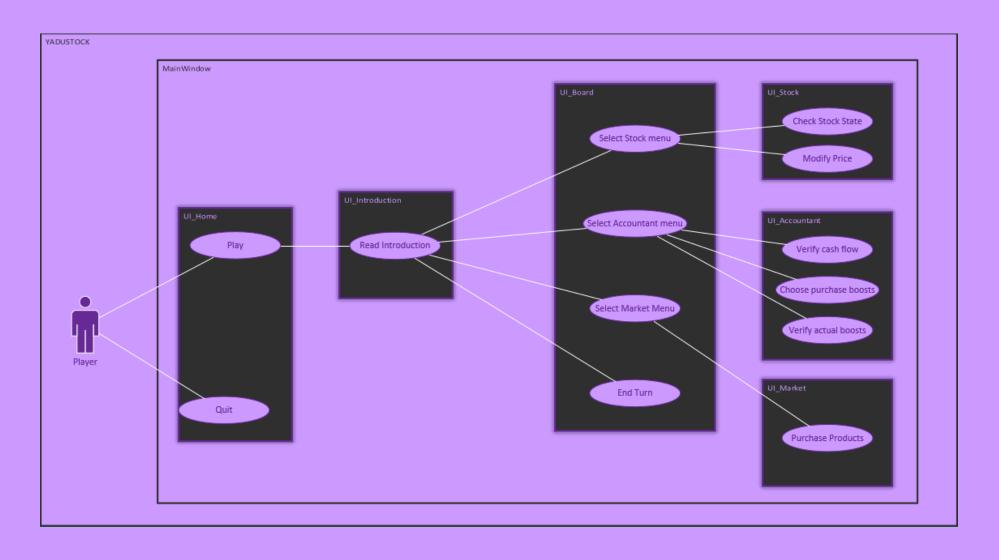
Type Product

Enumération regroupant tous les produits existants dans le jeu.

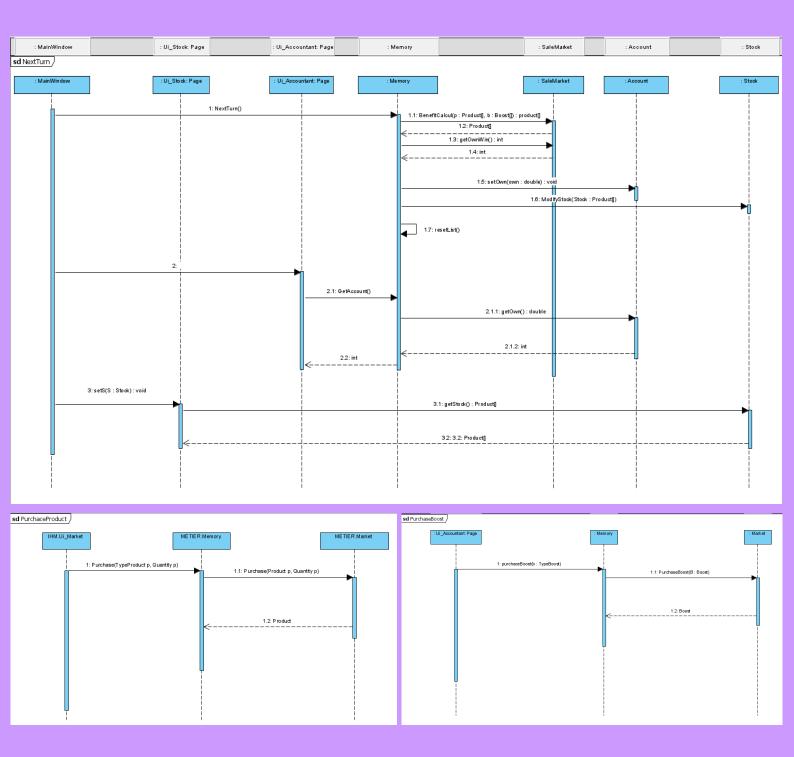
Type Boost

Enumération regroupant tous les boosts existants dans le jeu.

DIAGRAMME DE CAS D'UTILISATION



DIAGRAMMES DE SEQUENCE



MAQUETTES

A présent vous pouvez découvrir les maquettes et également comprendre le style choisi pour notre rapport.

Le fond du jeu est de la même couleur que le fond de ce rapport.

PAGE D'ACCUEIL

YADUSTOCK

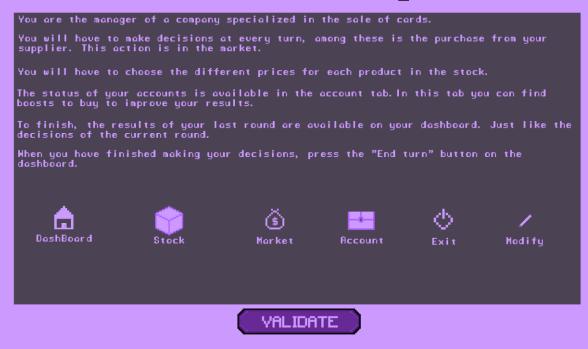


A Purple Dream's Game

Nous avons choisi un style pixel art pour la simplicité de réalisation mais également le plaisir de pouvoir personnalisé nous même les petites images du jeu, choisir des couleurs assez vives et originales.

PAGE D'INTRODUCTION

Welcome on our game !



Cette page contiendra les explications et règles du jeu. Les logos sont présentés pour précisé à quoi ils correspondent pour la suite.

TABLEAU DE BORD



YOUR BORD



Comme préciser au début du rapport le tableau de bord contient les informations principales du tour d'avant et du tout actuel.

PAGE STOCK



STOCK

PRODUCT	QUANTITY	PRICE
Pokemon Card	0	0 /
Yu-Gi-Oh Card	0	0 /
Magic Card	0	0 /
Tarot Card	0	0 /
VALIDATE		

Cette page contient l'état du stock et permet la modification du prix de vente.

PAGE MARKET



Cette page permettra au joueur d'acheter auprès du fournisseur tout en ayant les informations du prix total et la quantité qu'il possède actuellement.

PAGE ACCOUNT



Sur cette page le joueur peut vérifier ses comptes et acheter les boosts. Il peut également consulter ceux qu'il possède.

VALIDATE

Ainsi vous avez vu toutes les pages que le jeu de base contiendra avec les fonctionnalités de base. Comme vous avez pu le remarquer nous avons décidez de faire un jeu en anglais. Le monde de l'entreprise pouvant souvent nous amener à des situations d'échanges internationaux nous pensons que proposer aux joueurs de se familiariser avec la langue anglaise pourrait être un bon point. Nous avons essayé le plus possible d'ajouter des images pour remplacer le texte, comme le montre le menu du haut sur chaque page. Le sigle € suivi du chiffre d'argent qu'il possède permet d'avoir l'information sur l'argent disponible à chaque instant. Le tour actuel lui est également indiqué sur chaque page.

Pour rendre le jeu plus interactif nos boutons changeront lors du click :

Boost name

