

PROJET BEER GAMME

- cahier des charges -

Chapitre 1 : Contexte et périmètre du projet

1. Contexte

Afin de permettre aux étudiants d'IMT Mines d'Albi de mieux comprendre le principe de Chaîne Logistique, un jeu "Beer Game" est organisé par un enseignant. Ce jeu doit permettre aux élèves de prendre conscience des problèmes liés aux événements à risque qui peuvent perturber une chaîne logistique. Ils vont devoir ainsi adapter leur manière de jouer ainsi que prendre des décisions de manière autonome afin d'assurer un approvisionnement constant et adapté à la demande.

L'informatisation du jeu facilitera la mise en place et le déroulement du jeu. De plus, l'application sera disponible et adaptée au travail à distance.

2. Périmètre

Nous allons créer une application simulant une chaîne logistique. Celle-ci permet aux utilisateurs de jouer et à l'enseignant d'avoir la visibilité sur les actions des utilisateurs et le contrôle sur le jeu.

Une partie est composée d'une chaîne logistique dans laquelle chaque utilisateur devra se positionner puis gérer les commandes qu'il reçoit et qu'il doit fournir.

Il existe plusieurs rôles fonctionnels : les entreprises participant à la chaîne, les clients de fin de chaîne, les fournisseurs de début de chaîne et le superviseur de la partie.

Chapitre 2 : Système existant et objectifs du projet

1. Système existant

Le SI actuellement en place utilise des jetons pour modéliser les mouvements de matières entre les différents acteurs. Les données sont enregistrées à la main sur des feuilles de papier.

Une partie de Beer Game est divisée en plusieurs semaines se déroulant de la même façon:

- Lundi : un acteur découvre les commandes auxquelles il doit répondre cette semaine
- Mardi : un acteur décide la quantité qu'il va effectivement livrer pour toutes les commandes qu'il a reçues. Ces quantités et les restes à livrer sont enregistrés sur une fiche de suivi.
- Mercredi : un acteur réceptionne les produits que ses fournisseurs lui ont livré. Les quantités reçues sont enregistrées sur la fiche de suivi et dans son stock.
- Jeudi : un acteur passe un ou plusieurs commandes à un ou plusieurs fournisseurs. Les quantités commandées sont enregistrées sur la fiche de suivi.
- Vendredi : les commandes effectuées mardi sont envoyées.

2. Objectifs du projet

Le projet a pour objectif de simuler une chaîne logistique et d'apporter une vue globale sur cette simulation. L'objectif principal de l'application est pédagogique, elle doit permettre à des étudiants ou à des professionnels en formation de comprendre le principe de chaîne logistique.

Nous nous engageons à fournir une application Web permettant la simulation d'un jeu de Beer Game et les spécifications techniques de l'application que nous développons. Un guide d'utilisation ou des indications d'aide seront disponibles sur l'application.

Chapitre 3 : Caractéristiques fonctionnelles du futur système

1. Les données

Les données stockées pour chaque acteur sont :

- Le stock actuel :
 - Cette donnée est définie au début de la partie par le superviseur puis mise à jour quand des matières sont livrées par un fournisseur ou envoyées à un client.
 - Le stock de produit d'un acteur est un nombre entier positif ou nul.
 - Cette donnée est accessible par l'acteur et le superviseur.
- Les quantités de produit commandés à ses fournisseurs :
 - Cette donnée est initialisée à 0 au début de la partie puis mise à jour quand des produits sont commandés à un fournisseur.
 - La quantité de produit commandé d'un acteur est un nombre entier positif ou nul.
 - Cette donnée est accessible par l'acteur, le fournisseur cible de la commande et le superviseur.
- Les quantités en cours de livraison pour chacun de ses clients :
 - Cette donnée est initialisée par le superviseur au début de la partie puis mise à jour quand une commande est envoyée à un client.
 - La quantité en cours de livraison d'un acteur est un nombre entier positif ou nul.
 - Cette donnée est accessible par l'acteur, le client cible de la commande et le superviseur.
- Les restes à livrer pour chacun de ses clients :
 - Cette donnée est initialisée à 0 au début de la partie puis mise à jour quand une commande est envoyée à un client.
 - Le reste à livrer d'un acteur est un nombre entier positif ou nul.
 - Cette donnée est accessible par l'acteur et le superviseur.
- Le cumul des pénalités :
 - Cette donnée est initialisée à 0 au début de la partie puis mise à jour à chaque fin de semaine de jeu.
 - Le cumul des pénalités d'un acteur est un nombre positif ou nul.
 - Cette donnée est accessible par le superviseur.
- Le type d'acteur :
 - Cette donnée est une chaîne de caractères permettant d'identifier si l'acteur est contrôlé par un joueur ou une intelligence artificielle.
 - Les valeurs acceptées sont 'joueur' ou 'ia'. La valeur par défaut est 'ia'.
 - Cette donnée est modifiée quand un joueur prend le contrôle d'un acteur de la chaîne logistique.

Les données stockées pour chaque partie sont :

- L'identifiant de la partie :
 - Cette donnée est déterminée automatiquement à la création de la partie.
 - L'identifiant d'une partie est un nombre entier positif qui s'incrémente à chaque création de partie. Cet identifiant est unique.
 - Cette donnée est accessible par le superviseur et les joueurs.
- L'identifiant de la semaine :
 - Cette donnée est déterminée automatiquement au passage à une nouvelle semaine
 - L'identifiant d'une semaine est un nombre entier positif qui s'incrémente à chaque fin de semaine. Cet identifiant est unique.
 - Cette donnée est accessible par le superviseur et les joueurs.
- Les identifiants de tous les acteurs :
 - Cette donnée est initialisée quand un joueur sélectionne un acteur de la partie.
 - L'identifiant d'un acteur est un nombre entier positif. Cet identifiant est unique pour une partie donnée.
 - Cette donnée est accessible par le superviseur et les joueurs.
- La date de création de la partie :
 - Cette donnée est déterminée automatiquement à la création de la partie.
 - La date de création de la partie est un timestamp.
 - Cette donnée est accessible par le superviseur.

2. Les acteurs et cas d'utilisation du système cible

Les utilisateurs seront des élèves en école d'ingénieur ou des professionnels en formation. Ils jouent un acteur de la chaîne logistique.

Le superviseur d'une partie de jeu est un enseignant. Il doit avoir accès aux données des résultats ainsi que des graphiques résumant les actions des utilisateurs tout au long de la partie.

3. Hypothèses

- Les acteurs en début de chaîne peuvent commander une infinité de produits.
- Les matières fournies au premier acteur de la chaîne sont vendues en fin de chaîne sans transformation par les acteurs.
- Il n'y a pas de communication entre les joueurs.
- Les flux monétaires ne sont pas pris en compte.
- Les pénalités ne sont pas accessibles aux acteurs pendant la partie.
- Seule la création d'une partie nécessite la création d'un compte sur l'application.
- Les données de chaque partie sont conservées jusqu'à suppression par un administrateur.

4. Définition des processus

Le but du projet est de produire une application permettant de simuler le jeu du Beer Game. Pour cela, nous avons divisé le déroulement d'une partie de Beer Game en 3 stades:

- La création
- La partie en elle-même
- L'analyse de fin de partie

Pour chacun de ces stades, nous avons identifié les fonctions nécessaires au déroulement d'une partie et des fonctions optionnelles demandées par le client.

Fonctions Principales (nécessaire au déroulement d'une partie) :

Stade 1 : Création d'une partie

- Créer un lobby avec une chaîne logistique fixe
- Inviter un joueur dans le lobby
- Attribuer un rôle à un joueur
- Attribuer une Intelligence Artificielle aux rôles qui n'ont pas d'acteur

Stade 2 : Partie

- Visualiser les commandes reçues
- Passer une commande
- Effectuer une livraison
- Passer à l'étape de jeu suivante

Stade 3 : Analyse

- Afficher les graphiques d'évolution des pénalités et des stocks de tous les groupes (superviseur)

Autres fonctions :

Stade 1 : Création de la partie

- Choisir le nombre de joueurs
- Choisir l'état initial des stocks, livraison, délais de livraison et délais de communication
- Choisir le nombre de semaines à jouer

Stade 2 : Partie

- Visualiser les stocks
- Afficher la fiche de suivi
- Remplir la fiche de suivi
- Vérifier les valeurs dans la fiche de suivi (application)
- Voir toutes les fiches de suivi (superviseur)
- Voir les commandes en cours de livraison (superviseur)
- Voir les graphiques d'analyses (superviseur)
- Pouvoir prendre le contrôle d'un acteur (superviseur)

Chapitre 4 : Caractéristiques non fonctionnelles du futur système

1. Interface Homme-Machine

Un écran permettant de créer une partie. Cet écran permet d'inviter des acteurs à participer au jeu. Une fonction d'export des résultats des parties précédentes au format .csv doit être présente sur cet écran. Enfin, l'utilisateur peut voir l'ensemble des parties qu'il a créées et qui sont en cours d'exécution.

Un écran de visualisation de commande. Cet écran affiche l'ensemble des commandes de l'acteur (bons de commandes passés à ses fournisseurs et commandes en cours de livraison pour ses clients). C'est sur cet écran que sont effectuées les commandes décrites dans la partie 2.1 de ce document. Il doit permettre la sélection d'un fournisseur et la quantité de produits à commander.

Un écran contenant la fiche de suivi. L'écran doit afficher la fiche de suivi remplie par l'utilisateur depuis le début de la partie. C'est aussi sur cet écran qu'un acteur remplit la fiche de suivi tel que décrit dans la partie 2.1 de ce document. Le remplissage de la fiche de suivi doit être vérifié par le système et doit indiquer à l'utilisateur s'il a fait une faute dans le remplissage.

Un écran d'analyse de fin de partie. Cet écran contient le graphique d'analyse des pénalités et d'évolution des stocks pour chaque équipe.

2. Contraintes d'exploitations

- Volumétrie :

Le système d'information devra pouvoir accueillir au minimum l'ensemble des élèves d'une classe de génie industriel simultanément soit environ 60 personnes. Aussi, par mesure de sécurité il sera établi que le système d'information devra supporter un nombre de 100 connexions simultanées. La volumétrie des données stockées et échangées reste, quant à elle, à être définie.

- Performances :

Le temps de réponse envisagé doit être inférieur ou égal à 2 secondes.

- Disponibilité :

Pour la disponibilité, deux solutions sont envisagées :

→ Si cela est possible une disponibilité 24x7

→ Sinon une disponibilité minimale sur les horaires de cours c'est à dire de 8h à 17h30 du lundi au vendredi

- Coûts de mise en œuvre et de fonctionnement.

Considérant ce projet comme un projet de collaboration entre vous et de jeunes ingénieurs en formation, nous chiffrons ce dernier à 1560€ calculé à partir du taux horaire minimal en France qui est de 3,9€ multiplié par environ 80 heures de travail multiplié par 5 personnes qui sont les membres de l'équipe.

3. Contraintes d'organisations

Aucune contrainte d'organisation n'est à recenser. La gestion de la nouvelle application pourra tout à fait être assurée par le service informatique de l'école.

4. Contraintes de développement

Le projet sera dans un premier temps hébergé sur une machine de l'équipe via un serveur local. Dans un second temps, il est prévu que cet outil soit déployé sur les serveurs de l'école une fois son prototype validé mais la partie déploiement ne fait pas partie des exigences demandées et est hors scope du présent projet.

Concernant le développement de l'application, les langages utilisés seront :

- Systèmes de gestion de base de données : SQL
- Développement BackEnd : Python
- Développement FrontEnd : HTML, CSS, Javascript

Par ailleurs, pour le management du projet nous procéderons selon une méthodologie agile et envisageons avec vous des rendez-vous réguliers afin de valider les différents jalons du projet.

5. Délais de réalisation

Compte tenu du projet et de ses différentes phases, nous nous engageons à remettre le prototype à date du 25/02/2022.