

PPE3 : NOTAGAME

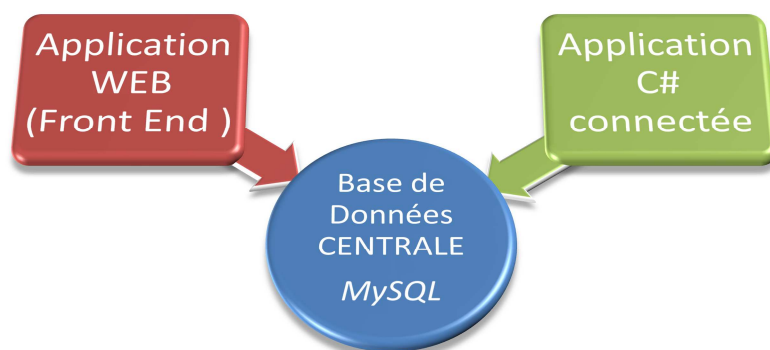
Contexte :

L'association **NotaGame** est spécialisée dans la promotion des jeux-vidéos. Elle souhaite que tous les jeux-vidéos compatibles pour des consoles de jeux puissent être notés de manière collaborative via un principe de commentaire et de note attribués par des utilisateurs.

La société 1FO@SIO leur a déjà fourni une première version V1 (principe de développement par itération : méthode AGILE). Cette version V1 repose sur 2 applicatifs :

- un **site Internet** en M.V.C
 - qui permet une inscription par email et pseudo d'un futur commentateur
 - qui permet à un utilisateur (pseudo) de consulter tous les jeux et leurs commentaires et de mettre un commentaire (max 100 caractères) sur un jeu précis.
 - qui possède un mode administrateur qui permet de visualiser les commentaires postés et de les supprimer s'il estime que ces commentaires ne sont pas *corrects* (charte de déontologie de l'association)
- une **application C#** en M.V.C et en mode connecté pour la base de données
 - qui permet de faire le CRUD (Create Read Update Delete) sur les tables CONSTRUCTEUR et SUPPORT de la base de données.
- une **Base de données commune** aux 2 applications :
 - table JeuxVideos (**idJV**, nomJV, anneeSortie, classification, editeur, description) : jeux videos
// classification correspond à un âge minimum conseillé
 - table Support (**idS**, nomS, anneeSortie, caractéristiques, idC#) : console de jeux
 - table Compatible (**idJV#**, **idS#**) : permet de vérifier la compatibilité d'un jeu en fonction d'une console
 - table Constructeur (**idC**, nomC) : liste des constructeurs de supports
 - table Users (**idU**, email, pseudo, communautaire) : liste des utilisateurs qui ont posté un commentaire.
la communauté est une information pouvant être (geek, joueur occasionnel, etc...)
 - table Commentaire (**idJV#**, **idU#**, libelle) : liste des commentaires mis sur les jeux par les utilisateurs.

Contrainte : données **CENTRALISEES** sur un même serveur de Base de Données :



Votre proposition de Service Informatique :

Vous êtes embauché par le **SSII 1FO_SIO** en tant que développeur/concepteur d'applications. Vous serez affecté à une équipe de développement de 3 personnes détaillée ci-dessous.

Vous êtes encadré par les 2 responsables M.O.E (Maîtrise d'œuvre):
Frédérique de Robien et Carine Autret (1fo.sio.49@gmail.com)

Les outils à votre disposition pour ces missions :

- Outil de développement : Eclipse ou autre / VisualStudio
- Technologies Web : CSS3, PHP5 et JQuery.
- Gestion de projet en groupe : GitHub (pour gérer le *versionning* et le travail collaboratif),
- planification selon digramme de GANTT (Microsoft Project par exemple) ou gestion en tâches (Trello par exemple).

Liste des documents fournis :

- *Fichier descriptif du projet* : **Projet NOTAGAME.pdf**
- *Fichier Technique* : **DocumentationTechnique NotaGame.docx**
- *Script de la BDD* : **bd_ppe3_notagame.sql**
- *Code fourni de l'application WEB existante* : <https://github.com/fderobien/PPE3NOTAGAME>
- *Code fourni pour l'application C# existante* : https://github.com/autret/PPE3_NotaGame_C-

PARTIE I : Vérifier que l'application est opérationnelle

Réponses écrites et formalisées : 1 document PDF

- Pour le web : les réponses doivent être complétées dans la documentation technique fournie

A destination de la SSII pour le vendredi 16 Septembre 2016 (fin séance)

A envoyer par mail à 1FO.SIO.49@gmail.com

1. Analyser la base de données en répondant aux questions suivantes :
 - a. expliquer l'utilité de la table Compatible ? Que vérifie-t-elle ?
 - b. Pourquoi identifier l'utilisateur par son email : quel est l'avantage de cette solution et aussi son inconvénient ?
 - c. expliquer l'utilité de la table Commentaire ? Que vérifie-t-elle ?
2. Fournir la **schématisation conceptuelle** de cette base de données.
3. Implanter la base de données et les applicatifs existants donnés dans un environnement client-serveur (ferme du lycée : serveur de BD, serveur web) et mettre en place les outils pour la gestion du travail de groupe (GitHub au moins).
4. Fournir un **jeu d'essai complet** afin de vérifier si l'application V1 fournie est opérationnelle. Lister tous les tuples nécessaires et faites progressivement une montée en charge.
5. Phase d'analyse de l'existant :
 - a. Élaborer un **cas d'utilisation** (exemple en ANNEXE I) de cette version V1 afin d'identifier les acteurs et les fonctionnalités associées.
 - b. Pour le **cas d'utilisation** « Affichage des commentaires pour un jeu » : lister et expliquer les différents fichiers et méthodes utilisés lors du clic sur le bouton d'option correspondant au jeu voulu. Un schéma vaut mieux qu'un long discours ! (exemple en ANNEXE II)

PARTIE II : VERSION 1 : finir les fonctionnalités manquantes

Partie II à faire valider par un maître d'œuvre

Avant le vendredi 30 septembre 2016 (fin séance)

(Version1.2) : C# : **Compléter le CRUD** sur les tables manquantes (sauf COMMENTAIRE, la gestion des commentaires étant faite sur le web en connexion administrateur) dans l'application lourde C#.

(Version1.3) : Web : **Modifier la sélection du jeu :**

Cas d'utilisation « La notation des jeux en web » : la sélection du jeu (bouton d'option) devient peu ergonomique lors de la montée en charge de la base de données. Modifier cette sélection du jeu en conséquence.

(Version1.4) : Web : **Affichage du pseudo associé au commentaire :**

Pour les deux Cas d'utilisation « Consultation des commentaires par jeux-vidéos » et « Suppression des commentaires » (avec connexion administrateur) : l'affichage des commentaires est à compléter du pseudo de l'utilisateur l'ayant posté.

PARTIE III : VERSION 2 : les nouveaux besoins.

Présentation orale finale (fonctionnelle)

Semaine du BTS BLANC du 21 au 25 novembre 2016


L'association NotaGame souhaite faire évoluer son application. Voici ces nouveaux besoins précisés dans plusieurs missions.

Pour chaque mission, cela constituera une sous version V2.1 (Version 2 mission1).

Certaines missions ont des répercussions sur les 2 applicatifs (Site et C#)


Mission1 : (Version2.1) : Evolution des données

- La gestion de la communauté n'est pas satisfaisante. Nous voulons vérifier qu'il n'y a pas de doublons ni d'erreur de saisie.
- On constate que les joueurs organisent leurs jeux-vidéos par genre (action, aventure, jeu de rôle, sport, famille, stratégie...) et souhaitent dans leurs recherches retrouver cette notion de genre pour filtrer les jeux.

 Analyser ces besoins et Proposer une évolution de la base de données. Mettre en œuvre une solution en listant les différentes tâches identifiées sur les 2 applicatifs et réparties entre les 3 collaborateurs.


Mission2 : (Version2.2) : contrôle du formulaire utilisateur


NotaGame souhaite que les différents champs soient vérifiés au niveau applicatif (via du javascript) : email correct, pseudo de 5 à 10 caractères maximum

 Proposer une solution pour mettre en œuvre ce besoin en listant les différentes tâches identifiées sur les 2 applicatifs et réparties entre les 3 collaborateurs.

Mission3 : (Version2.3): Modération des commentaires

NotaGame trouve un peu gênant que les commentaires non corrects soient quand même affichés sur le site. Il souhaite que lorsqu'un utilisateur poste un commentaire celui-ci ne soit pas affiché de suite : il est en quelque sorte "En attente". Ce n'est qu'en mode administrateur que l'on pourra valider ou refuser le commentaire. S'il est validé, il s'affichera sur le site et s'il est refusé, pas d'affichage mais est conservé dans la BDD (il n'y a donc plus de suppression de commentaires comme dans la V1).

 Réaliser un DET (Diagramme État Transition) afin de bien comprendre tous les états pris par le commentaire (aide en ANNEXE III)

 Proposer une solution pour mettre en œuvre ce besoin en listant les différentes tâches identifiées sur les 2 applicatifs et réparties entre les 3 collaborateurs.

Mission4 : (Version2.4): Comparaison des jeux-vidéos

La version V1 permet de consulter via le site internet les jeux-vidéos et les commentaires. L'objectif final de cet ensemble applicatif est de proposer un outil de comparaison de ces jeux en utilisant les notes attribuées par les utilisateurs. Permettre cette saisie de note en même temps que le commentaire, note comprise entre 0 et 5.

D'une part, proposer une amélioration de la consultation afin que l'internaute puisse obtenir une liste classée des meilleurs jeux (tous genres, tous constructeurs confondus ou sélectionnés selon différents tris comme le genre, le constructeur, l'appareil compatible...).

D'autre part, l'application lourde C# permet l'ensemble des modifications des données. L'administrateur qui s'occupe de ces mises à jour souhaite pouvoir aussi consulter le comparatif des jeux-vidéos selon trois tris :

- Aucun tri : liste des 5 meilleurs jeux tout confondu
- Tri par genre : liste des 5 meilleurs jeux par genre
- Tri par appareil compatible : liste des 5 meilleurs jeux par appareil

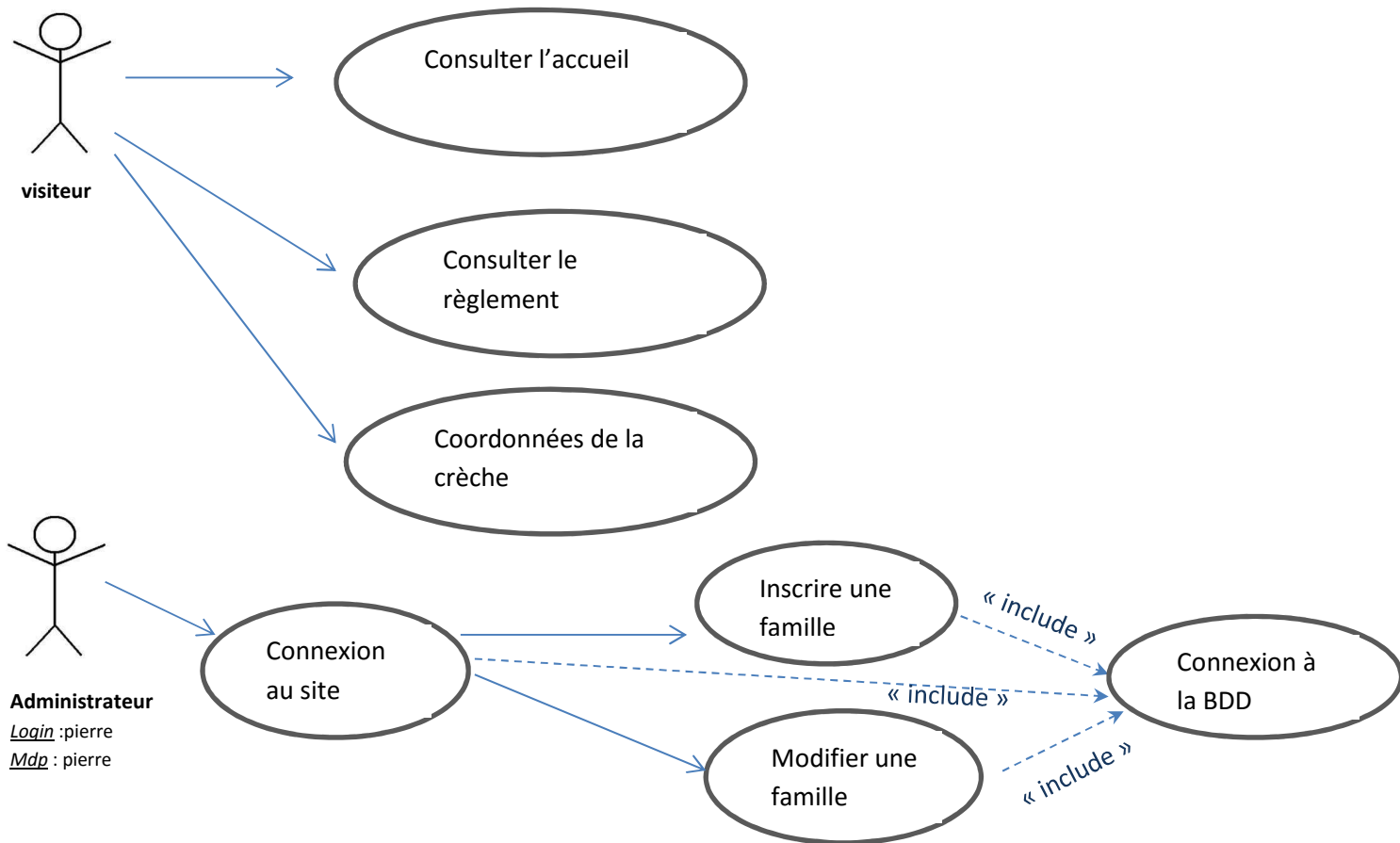
Ces données comparatives obtenues doivent être présentées dans l'applicatif C# (autre que dans un contrôle de type dataGridView) et permettre l'édition des résultats dans un document pdf.



Proposer une solution pour mettre en œuvre ces besoins en listant les différentes tâches identifiées sur les 2 applicatifs et réparties entre les 3 collaborateurs.

ANNEXE I :

Voici un cas d'utilisation d'un site internet concernant une crèche qui accueille des enfants :



Explication : l'acteur « visiteur » peut accéder à 2 fonctionnalités : consulter l'accueil et consulter le règlement. Il voit aussi en permanence les coordonnées de la crèche quel que soit la page.

L'acteur « administrateur » accède à 2 fonctionnalités : inscrire une famille et modifier un famille. Pour y accéder, il doit au préalable être identifié sur le site. Ces 3 fonctionnalités nécessitent une connexion à la BDD.

ANNEXE II :

Exemple de détail de l'enchaînement d'appel des différentes pages pour réaliser la fonctionnalité attendue : (Schéma réalisé avec FreeMind)



ANNEXE III :

Diagramme Etat-Transition de l'objet AVION :

