

RAPPORT

Projet Web 1A

Equipe :
ELHARTI Fadoua
DELRUE Pierre
LOUET Thibault
PERSON Victor

10 Mai 2016

Table des matières

1	Introduction	2
1.1	Introduction du sujet	2
1.2	L'équipe dans le projet	2
2	Choix techniques	3
2.1	Architecture du site	3
3	Base de données	4
3.1	Description des besoins	4
3.2	Schéma UML	4
4	Problèmes rencontrés et solutions	5
4.1	Gestion de la connexion	5
4.2	Gestion des formulaires	5
4.3	Gestion des privilèges	5
4.4	Gestion des informations apparaissant sur la page d'accueil	5
4.5	Gestion de la programmation	6
4.6	Le forum	6
5	Conclusion et perspectives d'évolution	7
5.1	Le projet finalisé	7
5.2	Le projet dans le futur	7
5.3	L'apport d'un tel projet	7

1 Introduction

1.1 Introduction du sujet

Le jeu en direct est en pleine explosion sur le web grâce à des sites comme Twitch ou Youtube qui permettent aux joueurs soit de retransmettre leur séance de jeu en direct soit de poster leur vidéo de gameplay. Cette tendance offre aux spectateurs une multitude de choix de joueurs. Cependant cela rend plus difficile pour un joueur de se démarquer des autres.

Grâce à notre site, nous offrons la possibilité à un joueur qui voudrait se professionnaliser de regrouper sa communauté à un même endroit. C'est pourquoi nous nous devons de proposer divers services à celui-ci.

Le site est axé autour de ces services : le stream en lui-même avec des informations importantes que le joueur voudrait faire passer à sa communauté, un forum où la communauté pourrait interagir entre-elle et avec le joueur, une grille de programme pour informer le téléspectateur des diffusions prévues par le joueur et la possibilité d'envoyer des messages directement au joueur.

Enfin, pour pouvoir avoir accès à ces services, un système de connexion s'est avéré nécessaire. Chaque compte se différencie selon son statut : Administrateur, Modérateur, Utilisateur. L'Administrateur a la possibilité de gérer les informations ainsi que la programmation. L'administrateur et les modérateurs peuvent gérer les utilisateurs.

1.2 L'équipe dans le projet

1. Thibault Louet : Mise en place de la base de données et feuille de style
2. Delrue Pierre : Mise en place du forum
3. Person Victor : Mise en place du reste du site et feuille de style
4. Elharti Fadoua : Mise en place de la feuille de style

Le travail a été effectué de manière collective grâce à GitHub qui nous a permis la transmission efficaces des fichiers et de leur mise à jour pendant la période de vacances. Chacun avait la possibilité de commenter les apports des autres pour mettre au jour des erreurs ou des manières plus efficaces de réaliser certaines actions. De même, les séances «physiques» nous ont permis de faire le point ensemble sur l'avancement du projet et du cap à tenir.

2 Choix techniques

Le site a été développé à l'aide des technologies suivantes :

1. SublimeText 2 pour le code en lui même
2. Github pour le partage des fichiers puis Git de l'ensiie pour le rendu
3. WampServer pour l'hébergement local du site
4. PHP 5.5 qui génère du HTML 5 pour toutes les pages du site
5. PostgreSQL 9.5 pour la base de données en place derrière le site
6. Javascript pour la validation des différents formulaires

Nous avons utilisé aucun Framework ou Template pour développer le site. Le rendu a été majoritairement testé à l'aide du navigateur Mozilla Firefox pendant le développement. Néanmoins, le site complet a été testé sur Mozilla Firefox, Google Chrome, Microsoft Edge et Safari. Ainsi, le site a été testé sur une large gamme de navigateurs. Cependant, ces tests ont été effectués sur ordinateur portable, aucun autre support n'a été utilisé pour aller sur le site. En conséquence, celui-ci ne doit pas être multisupport.

2.1 Architecture du site

Le site est composé de différentes pages php sur lesquelles peut naviguer l'utilisateur ou non. On différencie deux types de pages, les pages que l'utilisateur peut voir et les pages qui communiquent avec la base de donnée pour traiter diverses actions.

L'utilisateur a accès à la page d'accueil, la page de profil, la page de contact, aux diverses pages du forum, la page des messages privés, la page de connexion et de création de compte. Le site a été pensé pour qu'un utilisateur puisse accéder à n'importe quelle partie importe du site à tout moment. Nous considérons comme importantes la page d'accueil, la page principale du forum, la page de contact et la page de profil ainsi que les pages de connexion et de création de compte. En effet, ces pages contiennent toutes les informations que l'administrateur a définit comme nécessaire au projet (stream, informations, programmation, contact).

Les pages invisibles à l'utilisateur sont pour la plupart un moyen de communiquer avec la base de données pour créer, modifier, supprimer, visualiser des tuples qu'elle contient.

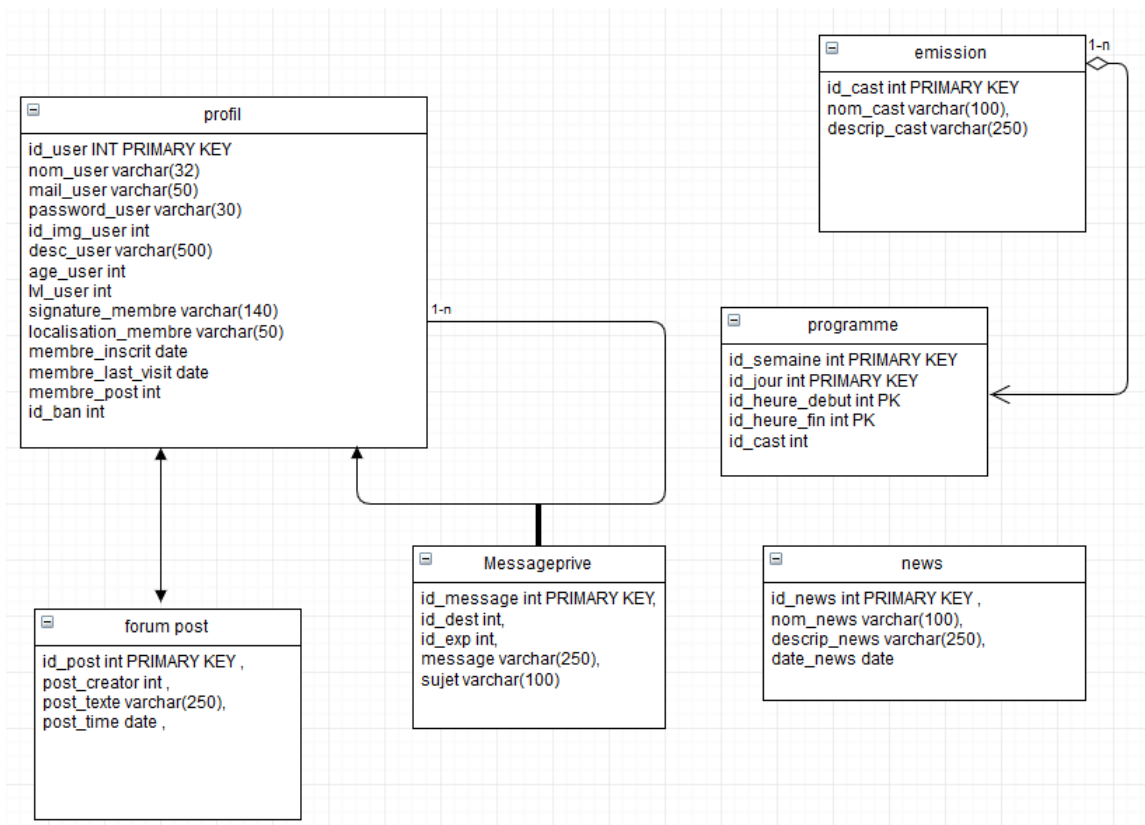
3 Base de données

3.1 Description des besoins

Pour le site il fallait pouvoir se connecter et pouvoir poster diverses informations de types différents. Ainsi, la base de données compte 8 tables :

1. Une table profil comprenant les informations des utilisateurs
2. Une table emission comprenant les informations liés aux émissions (blocs d'une même activité qui peut se retrouver dans plusieurs programmes)
3. Une table programme qui correspond à un bloc d'une même émission pendant une durée déterminée
4. 3 tables pour chaque catégorie du forum (présentation, règles générales et discussions générales) qui vont contenir les posts propres à ces catégories
5. Une table news qui contient les informations affichées sur la pag d'accueil
6. Une table messageprivé qui contient les messages privés envoyés entre les utilisateurs

3.2 Schéma UML



4 Problèmes rencontrés et solutions

4.1 Gestion de la connexion

La connexion est un point essentiel du site. L'utilisateur a la possibilité de naviguer sur le site sans avoir de compte mais n'aura alors pas accès à l'intégralité des fonctionnalités (messages privés, participation sur le forum, profil). La connexion et l'inscription se fait directement en cliquant sur le lien se connecter ou s'enregistrer accessible n'importe où sur le site. Une fois son compte créé, la connexion de l'utilisateur est assuré par la fonction `session_start()` qui garde dans sa variable `_SESSION` le nom de l'utilisateur. Ainsi, l'utilisateur doit cliquer sur se déconnecter pour être effectivement déconnecté du site.

4.2 Gestion des formulaires

Beaucoup d'informations doivent être transmises au serveur à partir de l'utilisateur. Ces informations prennent la forme de texte, chiffre, mot de passe. Il a alors été nécessaire de mettre à disposition de l'utilisateur des formulaires de saisie. Ceux-ci communiquent avec la base de données grâce à la méthode POST et sont sécurisés face aux injections SQL grâce à la fonction `pg_escape_string()` qui permet d'échapper tous les caractères d'une chaîne tapée par l'utilisateur. Enfin, pour empêcher le rechargement de la page lors d'erreur sur les formulaires, ceux-ci sont vérifiés à l'aide du javascript.

4.3 Gestion des privilèges

Ce type de site nécessite différents types de comptes : Administrateur, Modérateurs, Utilisateurs. Pour pouvoir gérer le type d'utilisateur connecté, la table profil a été agrémentée d'un champ `lvl_user` d'une valeur de 0 à 2. Ainsi, grâce à des conditionnelles dans les algorithmes communiquant avec la base de données, il est possible de choisir quoi afficher selon le type d'utilisateur qui est connecté (en particuliers le panel admin et le panel modérateur)

4.4 Gestion des informations apparaissant sur la page d'accueil

L'administrateur voulait pouvoir écrire des petites vignettes d'informations sur la page d'accueil sous sa vidéos. Il lui fallait alors avoir la possibilité de créer, modifier ou même supprimer celles-ci. La solution trouvée a été de stocker les informations dans une table à part entière que l'administrateur pourrait modifier grâce au panel admin et à des formulaires qu'il pourrait remplir. Pour l'affichage, on trie d'abord les informations par date décroissante puis on les affiche grâce à une liste et à un curseur.

4.5 Gestion de la programmation

L'administrateur voulait pouvoir mettre en ligne son «emploi du temps de jeu» pour que ses spectateurs ne ratent aucune diffusion. De même que pour les informations, il lui fallait avoir la possibilité de créer, modifier ou supprimer chaque temps de jeu. La solution mise en oeuvre ici a été de lui laisser seulement le choix de créer ou modifier des «blocs» d'émissions qu'il créerait séparément (lui laissant ainsi la possibilité de réutiliser ces blocs à divers moment. Pour ce faire, deux tables communiquent : emission et programme. L'utilisateur crée alors ses émissions avec leur titre et leur description grâce au panel admin. Puis celui-ci choisi parmi les émissions précédemment créées celle qu'il veut avec un certain créneau horraire. Tout chevauchement a été rendu impossible au préalable au travers du traitement de la base de données.

4.6 Le forum

Le site se voulait aggrémenté d'un forum où les utilisateurs connectés pouvaient communiquer. Comme celui se composait d'une partie statique (les 3 catégories principales), il a été décidé de créer 3 tables dans la base de donnée pour chacune d'entre elle qui contiendrait les poste propre à chaque catégories. L'utilisateur a alors la possibilité de poster un message qui apparaîtra a la suite des autres grâce au formulaire dédié en fin de page. Pour afficher les posts, il a suffit de créer un curseur trié par rapport à id_post sur la bonne table en fonction de l'endroit où l'utilisateur se trouve pour lui afficher les bons messages dans le bon ordre. Le résultat final est simplement un fil de posts sans véritable lien les uns des autres à cause d'un manque de temps pour vraiment avoir une structure de forum.

5 Conclusion et perspectives d'évolution

5.1 Le projet finalisé

Le projet de site pour streamer tel que nous l'avions défini a été dans la plupart des cas respectée. Néanmoins, le forum que nous avions voulu créer n'a pas abouti faute de temps. En effet, la structure de forum est une structure qui pourrait sembler simple aux premiers abords mais celle ci se révèle difficile à mettre en place en quelques semaines avec toutes les autres obligations imposées par le projet.

5.2 Le projet dans le futur

Le projet peut encore évoluer sur certains aspects. Tout d'abord comme nous le disions dans le paragraphe précédent, le forum pourrait totalement être mis en place. Ensuite, la base de données pourrait être mieux sécurisée avec un hachage des mots et passes au lieu de stocker ceux ci en clair. Une autre fonctionnalité pourrait aussi être ajoutée : la gestion de média qu'un utilisateur pourrait poster dans une section dédiée du site. Enfin, le design du site pourrait être amélioré à l'aide d'outils facilitant la conception d'une feuille de style.

5.3 L'apport d'un tel projet

Pour conclure, ce projet a été source de beaucoup d'expériences, tant au niveau technique qu'au niveau social, qui pourront être réutilisées dans le futur. Il a aussi été l'une des premières expériences de travail en groupe, façon de travailler totalement différente d'un travail seul, qui a ses avantages et ses inconvénients (organisation plus délicate, beaucoup plus d'idées, beaucoup plus de possibilité de travail) .