Groupe : Wiliam Nunesse , Thibault Michel, Victor Sorrenti, Julien Garcia

Cesi EXIA

Dossier de Conception

Projet Web

*Sommaire*

1. *Introduction*
2. *Choix technologique*
3. *Base de Données*
4. *Le site web*
5. *Nomenclature*
6. *Conclusion*
7. *Introduction :*

Le but de ce Projet web est de constituer un site web pour le BDE du CESI. L’objectif étant de faciliter l’organisation et la communication de manifestation au sein de l’école et de consacrer également une partie à la vente en ligne de goodies. Plusieurs rôles seront mis en place comme « membre du BDE » ou « salariés du CESI » qui auront différents droits sur le site.

1. *Choix technologique :*

Pour réaliser ce projet nous avons utilisé différents logiciels. Pour conceptualiser la base de données (BDD) nous avons utilisé JMerise, c’est un logiciel qui permet de créer graphiquement une BDD et de sortir un script SQL, afin de la rendre fonctionnelle. Nous avons donc utilisé phpMyAdmin en passant par WAMP car c’est un outil que nous avons beaucoup utilisé tout au long de nos années a l’EXIA, il nous est donc familié et facile d’utilisation. Pour la conception du site nous avons utilisé « sublime text 3 » qui est un éditeur de code que nous utilisons depuis le début de nos études. Nous avons aussi utilisé Github afin de partager notre travail entre nous.

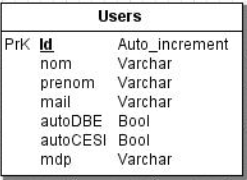
1. *Base de Données :*

Afin de récupérer et stocker les nombreuses données du site nous avons dû créer une BDD. Pour cela nous avons listé tous les éléments dont nous avions besoin :

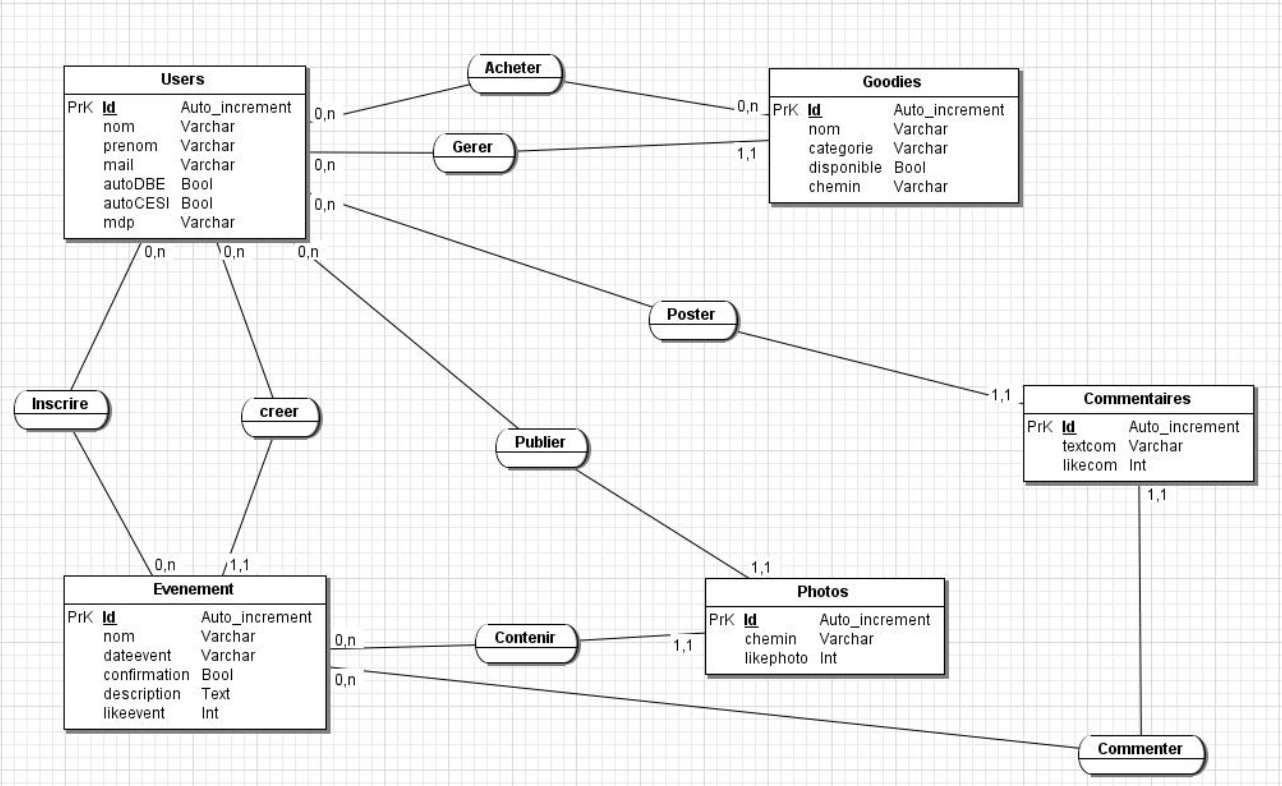
* Les utilisateurs (user)
* Les Evènements
* Les photos
* Les commentaires
* Les Goodies

Puis nous avons construit les tables avec leurs attributs :

Exemple : La table Users



Nous avons donc fini par réaliser un modèle conceptuel de données (MCD) reliant toutes les tables entre elles :



Le logiciel JMerise peut créer un script SQL à partir d’un MCD, ce qui nous a donc permis de la créer rapidement sur phpMyAdmin

Script SQL :

--

-- Structure de la table `acheter`

--

DROP TABLE IF EXISTS `acheter`;

CREATE TABLE IF NOT EXISTS `acheter` (

`Id` int(11) NOT NULL,

`Id\_Goodies` int(11) NOT NULL,

PRIMARY KEY (`Id`,`Id\_Goodies`),

KEY `FK\_Acheter\_Id\_Goodies` (`Id\_Goodies`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Structure de la table `commentaires`

--

DROP TABLE IF EXISTS `commentaires`;

CREATE TABLE IF NOT EXISTS `commentaires` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`textcom` varchar(255) NOT NULL,

`likecom` int(11) DEFAULT '0',

`Id\_Evenement` int(11) NOT NULL,

`Id\_Users` int(11) NOT NULL,

PRIMARY KEY (`Id`),

KEY `FK\_Commentaires\_Id\_Evenement` (`Id\_Evenement`),

KEY `FK\_Commentaires\_Id\_Users` (`Id\_Users`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=latin1;

--

-- Structure de la table `evenement`

--

DROP TABLE IF EXISTS `evenement`;

CREATE TABLE IF NOT EXISTS `evenement` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`nom` varchar(255) NOT NULL,

`dateevent` varchar(255) DEFAULT 'pas de date',

`confirmation` tinyint(1) DEFAULT '0',

`description` text NOT NULL,

`likeevent` int(11) DEFAULT '0',

`Id\_Users` int(11) NOT NULL,

PRIMARY KEY (`Id`),

KEY `FK\_Evenement\_Id\_Users` (`Id\_Users`)

) ENGINE=InnoDB AUTO\_INCREMENT=6 DEFAULT CHARSET=latin1;

--

-- Structure de la table `goodies`

--

DROP TABLE IF EXISTS `goodies`;

CREATE TABLE IF NOT EXISTS `goodies` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`nom` varchar(255) NOT NULL,

`categorie` varchar(255) NOT NULL DEFAULT 'Accessoires',

`disponible` tinyint(1) NOT NULL DEFAULT '1',

`chemin` varchar(255) DEFAULT NULL,

`Id\_Users` int(11) NOT NULL,

PRIMARY KEY (`Id`),

KEY `FK\_Goodies\_Id\_Users` (`Id\_Users`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Structure de la table `inscrire`

--

DROP TABLE IF EXISTS `inscrire`;

CREATE TABLE IF NOT EXISTS `inscrire` (

`Id` int(11) NOT NULL,

`Id\_Evenement` int(11) NOT NULL,

PRIMARY KEY (`Id`,`Id\_Evenement`),

KEY `FK\_Inscrire\_Id\_Evenement` (`Id\_Evenement`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Structure de la table `photos`

--

DROP TABLE IF EXISTS `photos`;

CREATE TABLE IF NOT EXISTS `photos` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`chemin` varchar(255) NOT NULL,

`likephoto` int(11) DEFAULT '0',

`Id\_Users` int(11) NOT NULL,

`Id\_Evenement` int(11) NOT NULL,

PRIMARY KEY (`Id`),

KEY `FK\_Photos\_Id\_Users` (`Id\_Users`),

KEY `FK\_Photos\_Id\_Evenement` (`Id\_Evenement`)

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Structure de la table `users`

--

DROP TABLE IF EXISTS `users`;

CREATE TABLE IF NOT EXISTS `users` (

`Id` int(11) NOT NULL AUTO\_INCREMENT,

`nom` varchar(255) NOT NULL,

`prenom` varchar(255) NOT NULL,

`mail` varchar(255) NOT NULL,

`autoDBE` tinyint(1) NOT NULL DEFAULT '0',

`autoCESI` tinyint(1) NOT NULL DEFAULT '0',

`mdp` varchar(255) NOT NULL,

PRIMARY KEY (`Id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=latin1;

--

-- Contraintes pour les tables déchargées

--

--

-- Contraintes pour la table `acheter`

--

ALTER TABLE `acheter`

ADD CONSTRAINT `FK\_Acheter\_Id` FOREIGN KEY (`Id`) REFERENCES `users` (`Id`),

ADD CONSTRAINT `FK\_Acheter\_Id\_Goodies` FOREIGN KEY (`Id\_Goodies`) REFERENCES `goodies` (`Id`);

--

-- Contraintes pour la table `commentaires`

--

ALTER TABLE `commentaires`

ADD CONSTRAINT `FK\_Commentaires\_Id\_Evenement` FOREIGN KEY (`Id\_Evenement`) REFERENCES `evenement` (`Id`),

ADD CONSTRAINT `FK\_Commentaires\_Id\_Users` FOREIGN KEY (`Id\_Users`) REFERENCES `users` (`Id`);

--

-- Contraintes pour la table `evenement`

--

ALTER TABLE `evenement`

ADD CONSTRAINT `FK\_Evenement\_Id\_Users` FOREIGN KEY (`Id\_Users`) REFERENCES `users` (`Id`);

--

-- Contraintes pour la table `goodies`

--

ALTER TABLE `goodies`

ADD CONSTRAINT `FK\_Goodies\_Id\_Users` FOREIGN KEY (`Id\_Users`) REFERENCES `users` (`Id`);

--

-- Contraintes pour la table `inscrire`

--

ALTER TABLE `inscrire`

ADD CONSTRAINT `FK\_Inscrire\_Id` FOREIGN KEY (`Id`) REFERENCES `users` (`Id`),

ADD CONSTRAINT `FK\_Inscrire\_Id\_Evenement` FOREIGN KEY (`Id\_Evenement`) REFERENCES `evenement` (`Id`);

--

-- Contraintes pour la table `photos`

--

ALTER TABLE `photos`

ADD CONSTRAINT `FK\_Photos\_Id\_Evenement` FOREIGN KEY (`Id\_Evenement`) REFERENCES `evenement` (`Id`),

ADD CONSTRAINT `FK\_Photos\_Id\_Users` FOREIGN KEY (`Id\_Users`) REFERENCES `users` (`Id`);

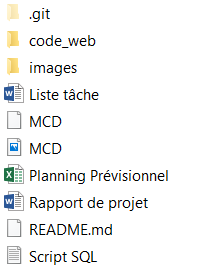
COMMIT;

1. *Le site web :*

Le site web a été conçu dans plusieurs langages. L’HyperTextMarkup Language (HTML) un langage balise pour le « dur » de la page, les feuilles de style en cascade (CSS) pour la front-end du site et l’HyperText Preprocessor (PHP) qui permet la liaison à la BDD et la dynamisation du site pour le back-end. Nous avons aussi utilisé du SQL pour communiquer avec la BDD grâce à des requêtes SQL.

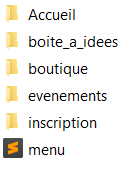
1. *Nomenclature :*

Suite à l’utilisation de différents langages nous avons dû (plus ou moins) structurer notre code et donc séparer les différents langages. Exemple de la structure notre code :

Ici les deux fichiers principaux sont « code\_web » et « images ».

Le dossier « images » contient toutes les images présentent sur le site, et la BDD leurs chemin d’accès.

Le dossier « code\_web » est composé comme ci-dessous :



Tous les dossier sont composés d’une ou plusieurs page (.php) ainsi que leur fiche de style (.css). La fiche menu.css est en dehors car elle est utilisée dans toutes les pages (utile pour la barre de navigation du site)

Toutes les pages sont alors liées les unes aux autres via leurs chemin dans les dossiers.



ou



1. *Conclusion :*

Pour conclure, ce projet nous as permit d’approfondir nos connaissances acquises durant l’UE développement web, comme les langages HTML, CSS, PHP, Javascript mais aussi l’utilisation de certains logiciels. Ce projet nous a aussi permit de se remémorer les notions (déjà acquises) de BDD.

Lien GitHub du Projet : https://github.com/Vicso/ProjetWeb