

Giftlist

Rapport

Projet Tutoré

S3

Lise Jolicoeur – Eugène Leclerc – Yvain Raynaud

Table des matières

Introduction.....	3
Rapport sur l'analyse et la conception de la base de données	5
Rapport sur l'implémentation	7
Code SQL.....	7
Documentation technique.....	8
Conclusion	8

Introduction

Dans le cadre du projet tutoré du S3, nous avons été chargés de concevoir une application de gestion groupée de cadeaux de Noël. Cette application devra permettre l'organisation des achats de cadeaux de Noël à l'aide d'un système de groupes et de listes de cadeaux. L'utilisateur pourra choisir les cadeaux qu'il souhaite, les organiser dans des listes et partager ces listes avec les membres des groupes dont il fait partie. Ces membres pourront indiquer l'achat d'un des cadeaux, ce qui sera signalé aux autres utilisateurs tout en restant une surprise pour le destinataire.

Les acteurs concernés par cette utilisation seront d'un côté les administrateurs, chargés de mettre en place l'application et de la maintenir, et de l'autre côté les utilisateurs, qui bénéficieront des fonctionnalités de l'application. Ces utilisateurs peuvent bénéficier de droits supplémentaires s'ils sont membres ou propriétaires d'un groupe. Ce seront principalement des familles ou des groupes d'amis souhaitant une plateforme pour gérer leurs cadeaux de Noël.

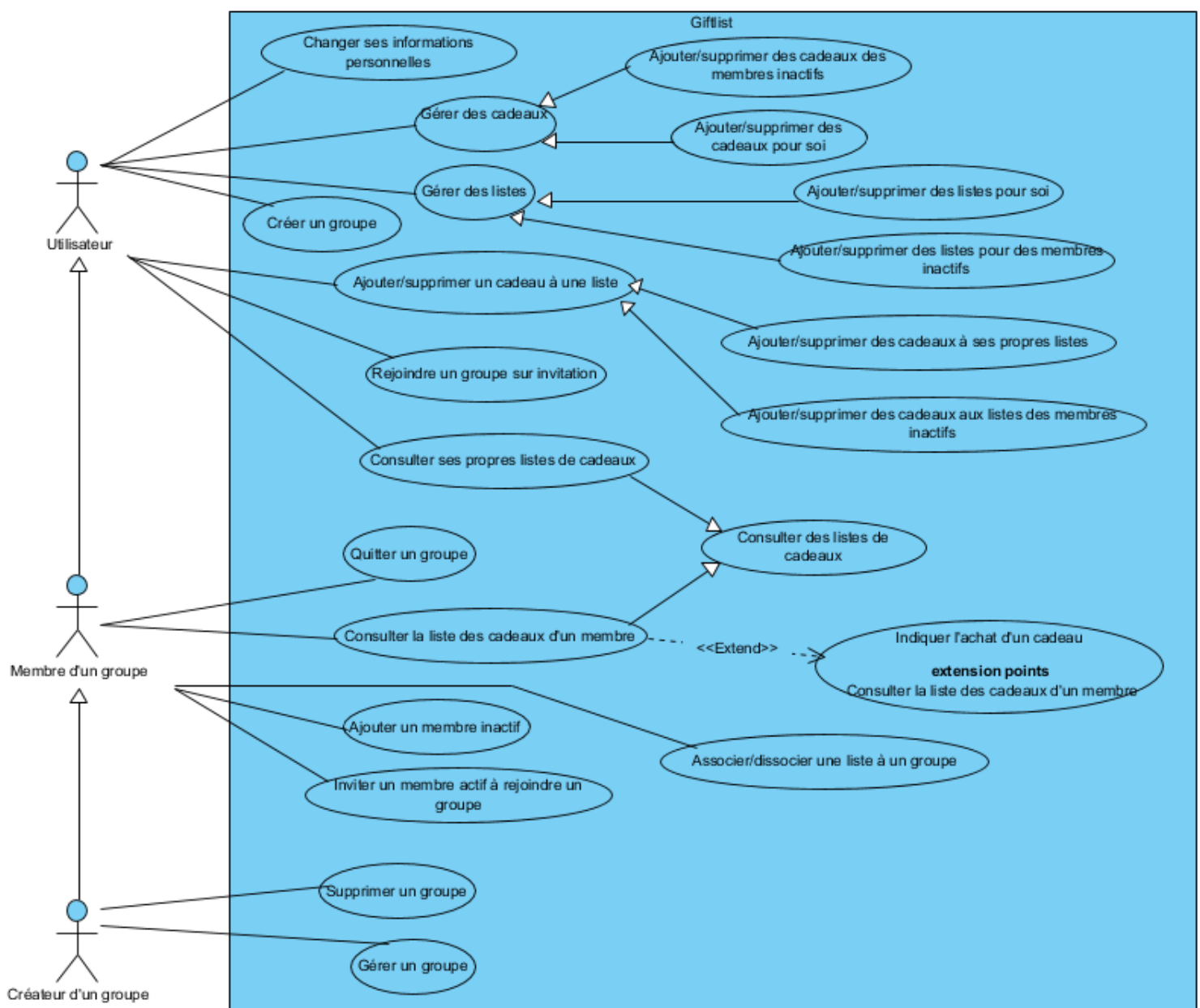


Figure 1 : diagramme de cas d'utilisation

Les cas d'utilisation sont les suivants :

- **Création de compte** : l'utilisateur peut créer un compte à partir de son nom, son prénom, son login et son mot de passe. Il peut aussi créer un compte pour un utilisateur inactif, comme un enfant, à partir du nom et prénom de la personne et qu'il gèrera lui-même
- **Création de groupe** : l'utilisateur inscrit peut créer des groupes auxquels il donne un nom et en supprimer
- **Invitation dans un groupe** : l'utilisateur membre d'un groupe peut inviter d'autres utilisateurs dans ce groupe
- **Création de cadeau** : l'utilisateur inscrit peut créer des cadeaux (nom, image, prix, ...) et en supprimer
- **Création d'une liste** : l'utilisateur inscrit peut créer des listes auxquelles il donne un nom et en supprimer
- **Ajout de cadeaux dans une liste** : l'utilisateur inscrit peut ajouter ses cadeaux dans les listes qu'il a créées et en retirer
- **Partage d'une liste** : l'utilisateur membre d'un groupe peut partager une ou plusieurs de ses listes dans un ou plusieurs groupes dont il est membre
- **Achat d'un cadeau** : l'utilisateur membre d'un groupe peut indiquer l'achat d'un cadeau d'une liste partagée dans ce groupe. Les autres membres seront informés, mais pas le destinataire du cadeau.
- **Consulter des listes de cadeaux** : chaque utilisateur peut consulter ses listes de cadeaux et celles des autres membres de ses groupes.
- **Changer ses informations personnelles** : chaque doit pouvoir changer son mail et son mot de passe.
- Chaque utilisateur peut faire les mêmes opérations sur les listes, cadeaux, groupe pour les utilisateurs inactifs.

Rapport sur l'analyse et la conception de la base de données

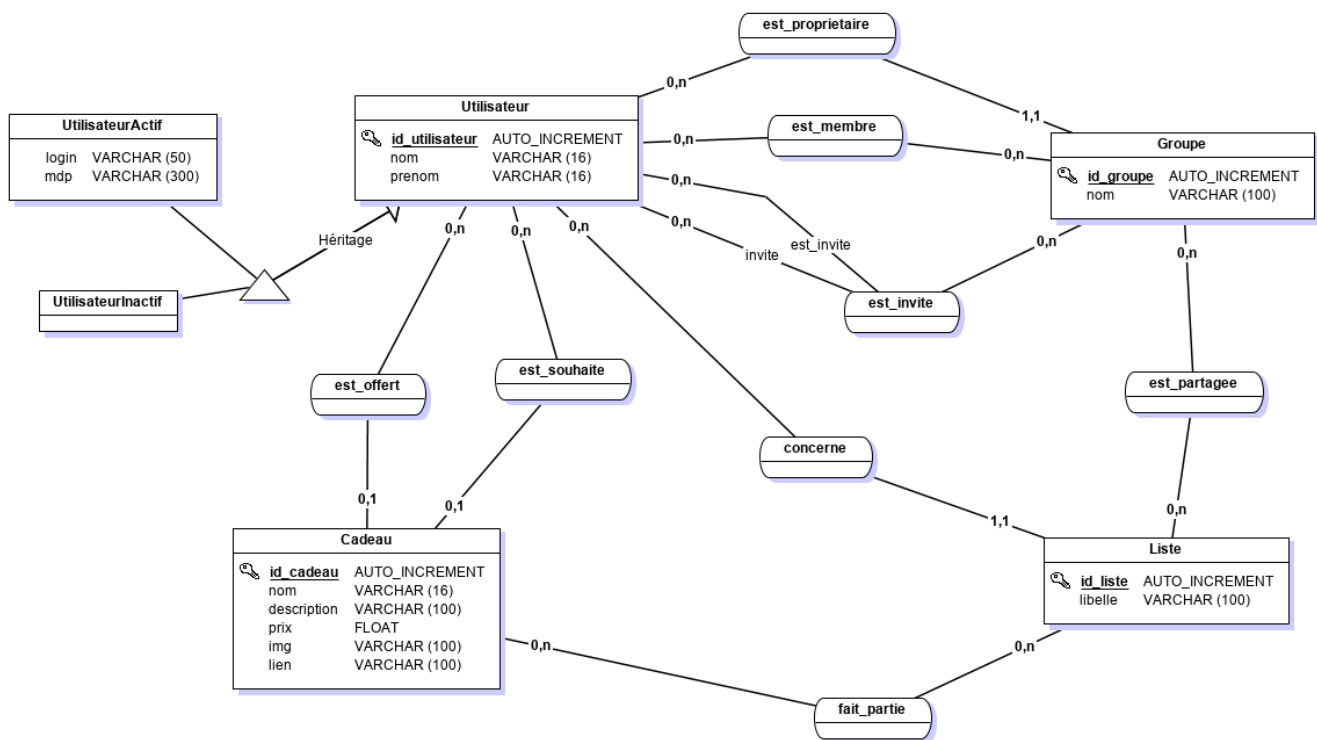


Figure 2 : modèle conceptuel des données

Nous avons pris plusieurs décisions lors de la conception de la base de données. Comme il est indiqué dans le schéma, nous avons différencié les utilisateurs actifs (normaux) des utilisateurs inactifs (enfants, personnes âgées) tout en gardant une base commune pour les liens avec les autres entités. Nous avons aussi choisi de garder des cadeaux uniques : si deux personnes veulent le même ordinateur portable, il sera présent deux fois dans la base de données. Toutefois, compte tenu de la faible chance que deux utilisateurs souhaitent le même exact cadeau (puisqu'ils indiquent nom, description, prix, image et lien), nous avons considéré qu'il était judicieux de procéder ainsi. De plus, cela simplifiera l'ajout d'un cadeau par l'utilisateur qui aura seulement besoin de rentrer les informations de son cadeau pour le créer plutôt que de chercher parmi plusieurs dizaines, centaines voire milliers de cadeaux existants. Nous avons également choisi d'indiquer l'achat d'un cadeau par un id utilisateur dans la table cadeau. Cet id nous permettra de garder une trace de qui a acheté le cadeau.

Loading...

Figure 3 : dictionnaire des données

Cadeau

Column	Type	Null	Default	Comments
id_cadeau (<i>Primary</i>)	int(11)	No		
nom	varchar(16)	No		
description	varchar(100)	Yes	NULL	
prix	float	Yes	NULL	
img	varchar(100)	Yes	NULL	
lien	varchar(100)	Yes	NULL	
id_utilisateur_est_offert	int(11)	Yes	NULL	
id_utilisateur_est_souhaite	int(11)	Yes	NULL	

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_cadeau	2	A	No	
Cadeau Utilisateur FK	BTREE	No	No	id_utilisateur_est_offert	2	A	Yes	
Cadeau Utilisateur0 FK	BTREE	No	No	id_utilisateur_est_souhaite	7	A	Yes	

Groupe

Column	Type	Null	Default	Comments
id_groupe (<i>Primary</i>)	int(11)	No		
nom	varchar(100)	No		
id_utilisateur	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_groupe	2	A	No	
Groupe Utilisateur FK	BTREE	No	No	id_utilisateur	2	A	No	

Liste

Column	Type	Null	Default	Comments
id_liste (<i>Primary</i>)	int(11)	No		
libelle	varchar(100)	No		
id_utilisateur	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_liste	7	A	No	
Liste Utilisateur FK	BTREE	No	No	id_utilisateur	6	A	No	

Utilisateur

Column	Type	Null	Default	Comments
id_utilisateur (<i>Primary</i>)	int(11)	No		
nom	varchar(16)	No		
prenom	varchar(16)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_utilisateur	10	A	No	

UtilisateurActif

Column	Type	Null	Default	Comments
id_utilisateur (<i>Primary</i>)	int(11)	No		
login	varchar(50)	No		
mdp	varchar(300)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_utilisateur	7	A	No	

UtilisateurInactif

Column	Type	Null	Default	Comments
id_utilisateur (<i>Primary</i>)	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_utilisateur	2	A	No	

est_invite

Column	Type	Null	Default	Comments
id_utilisateur (<i>Primary</i>)	int(11)	No		
id_groupe (<i>Primary</i>)	int(11)	No		
id_utilisateur_est_invite (<i>Primary</i>)	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_utilisateur	2	A	No	
				id_groupe	2	A	No	
				id_utilisateur_est_invite	7	A	No	
est_invite_Groupe0 FK	BTREE	No	No	id_groupe	2	A	No	
est_invite_Utilisateur1 FK	BTREE	No	No	id_utilisateur_est_invite	6	A	No	

est_membre

Column	Type	Null	Default	Comments
id_groupe (<i>Primary</i>)	int(11)	No		
id_utilisateur (<i>Primary</i>)	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_groupe	2	A	No	
				id_utilisateur	4	A	No	
est_membre_Utilisateur0 FK	BTREE	No	No	id_utilisateur	4	A	No	

est_partagee

Column	Type	Null	Default	Comments
id_liste (<i>Primary</i>)	int(11)	No		
id_groupe (<i>Primary</i>)	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_liste	7	A	No	
				id_groupe	8	A	No	
est_partagee_Groupe0 FK	BTREE	No	No	id_groupe	2	A	No	

fait_partie

Column	Type	Null	Default	Comments
id_liste (<i>Primary</i>)	int(11)	No		
id_cadeau (<i>Primary</i>)	int(11)	No		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id_liste	7	A	No	
				id_cadeau	10	A	No	
fait_partie_Cadeau0 FK	BTREE	No	No	id_cadeau	10	A	No	

[Print](#)

Rapport sur l'implémentation

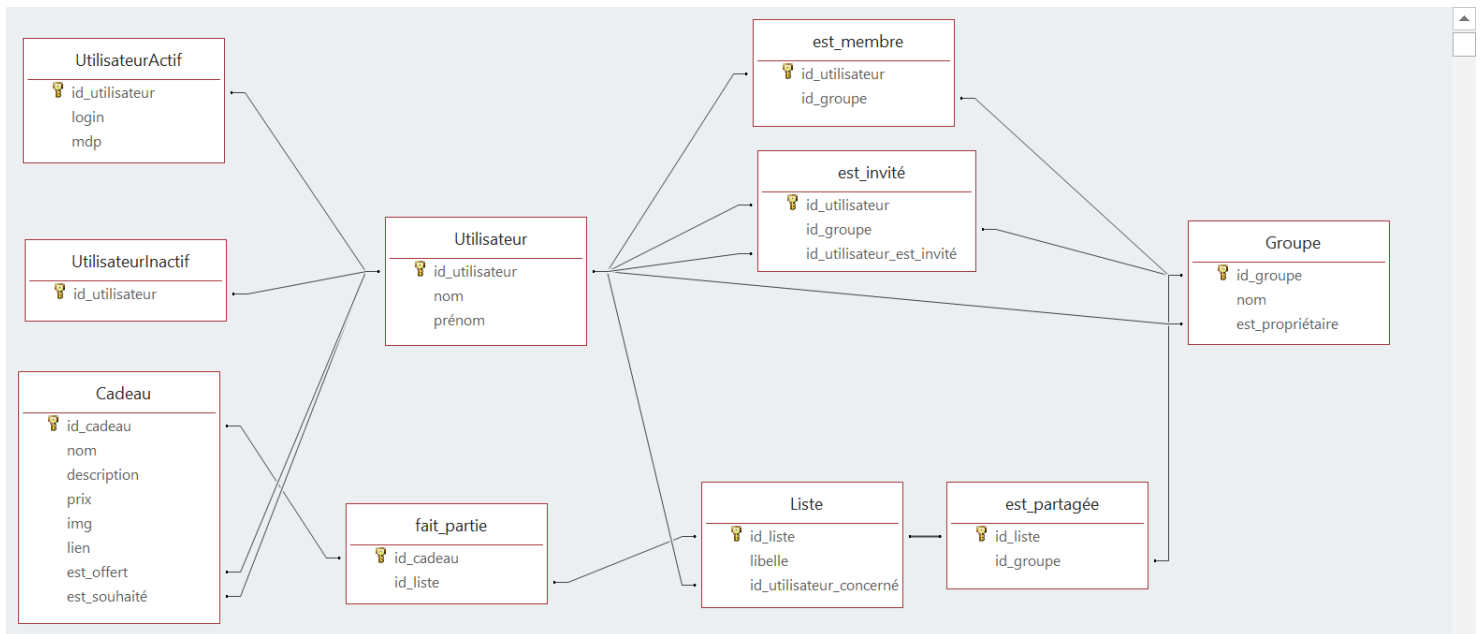


Figure 4 : schéma relationnel des données

Code SQL

Notre code est entièrement sauvegardé sur GitHub. Voici les liens pour y accéder :

Code pour la création de la base de données :

https://github.com/Raynaudy/tutore_s3_cadeaux/blob/master/sql/BDD_create.sql

Code pour l'insertion des données :

https://github.com/Raynaudy/tutore_s3_cadeaux/blob/master/sql/codeSQLInsertionDonnees.sql

Code pour réaliser les fonctionnalités de l'application :

https://github.com/Raynaudy/tutore_s3_cadeaux/blob/master/sql/codeSQLRequetesFonctions.txt

Documentation technique

Avec le code fourni dans ce rapport, il est possible d'implémenter notre base de données dans plusieurs plateformes différentes comme sqldeveloper ou phpMyAdmin. Pour répondre à notre besoin qui est de faire un site web, nous avons décidé d'implémenter notre base de données directement dans phpMyAdmin. phpMyAdmin permet de facilement utiliser la base de données et de facilement visualiser les données présentes.

Au niveau technique, il suffit d'installer phpMyAdmin sur son ordinateur et de se créer un compte.

Voici deux liens pour être guidé lors de l'installation de phpMyAdmin, sous Linux ou Windows :

- <https://itx-technologies.com/fr/blog/2108-installer-et-acceder-a-phpmyadmin-sous-ubuntu>
- <https://longwatchstudio.com/blog/2018/05/02/installer-phpmyadmin-sous-windows-10-iis/>

Ensuite, il faut copier le code de création de la base de données dans un fichier sql et l'importer dans phpMyAdmin. Une fois la base de données créée, il faut copier-coller le code d'insertion dans la feuille SQL de phpMyAdmin et lancer les requêtes. Après ces deux étapes courtes, la base de données est prête à être utilisée.

Conclusion

Après de multiples tests, nous pensons avoir su répondre à toutes les contraintes imposées. En effet, nous avons effectué diverses requêtes en imaginant les différents cas possibles et les différents besoins des utilisateurs et tout fonctionnait parfaitement. Cependant, nous ne pourrions en être certain seulement une fois que le développement applicatif sera bien avancé. Ce projet a permis à toute l'équipe une rapide remise à niveau sur la conception de base de données ainsi que sur le langage SQL. Nous avons même découvert la généralisation en SQL que nous n'avons pas eu l'occasion d'utiliser l'année dernière.