Étienne Robert, Nicolas Mathieu, Bruno Théorêt, Frank Della Amegnignon, Joey Mallet, Tom Tran

cahier des charges

Sprint 3

Contents

[1- Introduction 2](#_Toc135079802)

[2- Description Générale 2](#_Toc135079803)

[3- Exigences fonctionnelles 2](#_Toc135079804)

[3.1. Inscription et authentification des utilisateurs 2](#_Toc135079805)

[3.2. Groupes 2](#_Toc135079806)

[3.3. Membres du groupe 3](#_Toc135079807)

[3.4. Liste de souhait et cadeaux 3](#_Toc135079808)

[3.5. Algorithme et pige 3](#_Toc135079809)

[3.6 Confidentialité 3](#_Toc135079810)

[4. Exigences non fonctionnelles 4](#_Toc135079811)

[4.1. Performance 4](#_Toc135079812)

[4.2. Sécurité 4](#_Toc135079813)

[4.3. Compatibilité 4](#_Toc135079814)

[5. Exigences techniques 4](#_Toc135079815)

[5.1. Backend 4](#_Toc135079816)

[5.2. Frontend 4](#_Toc135079817)

[5.3. Base de données 4](#_Toc135079818)

[6. Livrables 5](#_Toc135079819)

[7- Tâches à compléter pour le Sprint 3 5](#_Toc135079820)

[8- Schéma de Base de Données (mise à jour) 6](#_Toc135079821)

# 1- Introduction

L'objectif de ce document est de décrire en détail les exigences fonctionnelles, non fonctionnelles et techniques pour le développement d'une application web d'échange de cadeaux. L'application sera développée en utilisant ASP.NET Web API, Entity Framework Core et C# pour le backend, et React pour le frontend.

# 2- Description Générale

L'application permettra aux utilisateurs de s'inscrire, de se connecter, de créer des groupes d'échange de cadeaux, d'ajouter et de supprimer des membres à ces groupes, de lister les cadeaux qu’un membre désire recevoir et d’effectuer une pige entre les membres qui se verront attribuer un chouchou, c’est-à-dire un membre pigé à qui donner un ou plusieurs cadeaux. Le premier utilisateur à s’inscrire sera administrateur d’un groupe d’échange de cadeaux.

Pour gérer la pige des membres, nous allons mettre en place un algorithme au sein de notre application. Cette fonctionnalité sera exécutée au moment où le décide l’administrateur du groupe.

# 3- Exigences fonctionnelles

## 3.1. Inscription et authentification des utilisateurs

Les utilisateurs doivent être en mesure de s'inscrire et de se connecter à l'application. Les informations d'inscription comprendront le prénom, le nom, l'adresse courriel, le nom d’utilisateur et le mot de passe.

## 3.2. Groupes

Le premier utilisateur à s’inscrire sur le site sera l’administrateur. Les utilisateurs administrateurs peuvent créer des groupes d'échange de cadeaux. Chaque groupe doit avoir un nom, une description, une date d’échange, une date de pige, un budget de cadeau(x) et un lien web pour l’image du groupe.

## 3.3. Membres du groupe

Les utilisateurs administrateurs peuvent ajouter des membres à leurs groupes en utilisant l'adresse courriel de ces personnes, leur nom et prénom. Une invitation par courriel est envoyée au(x) personnes(s) invité(s). Les membres peuvent être supprimés des groupes par l'utilisateur administrateur, celui qui a créé le groupe. Un utilisateur administrateur peut aussi promouvoir un autre utilisateur comme administrateur.

Les utilisateurs recevant une invitation courriel seront invités à cliquer sur un lien web qui les dirigera vers une page d’inscription au groupe d’échange de cadeaux afin qu’ils complètent leur profil.

## 3.4. Liste de souhait et cadeaux

Les utilisateurs peuvent ajouter, modifier et supprimer des cadeaux de leur liste de souhait. Chaque cadeau a un nom, une description et un prix, un lien web pour l’image du cadeau et un lien web pointant vers un site de vendeur.

## 3.5. Algorithme et pige

L'algorithme de pige sera conçu pour attribuer un membre à chaque autre membre du groupe de manière aléatoire, mais en s'assurant que personne ne se pige lui-même. Cela créera une sorte de chaîne où chaque membre est lié à un autre membre unique.

Chaque membre recevra une notification par courriel lui indiquant le membre qu'il a pigé, ainsi que la liste de cadeaux souhaités de ce membre. Le membre est alors responsable de choisir un cadeau ou un groupe de cadeaux de cette liste pour offrir à la personne qu'il a pigée.

Il sera impossible d’inviter de nouveaux membres au groupe une fois que la pige aura été effectuée!

## 3.6 Confidentialité

Seul le membre qui pige un autre membre connaîtra l'identité de la personne pigée. Les autres membres du groupe ne seront pas informés de cette information pour maintenir l'aspect surprise de l'échange de cadeaux.

# 4. Exigences non fonctionnelles

## 4.1. Performance

L'application doit être capable de gérer un grand nombre d'utilisateurs et de groupes simultanément.

## 4.2. Sécurité

Les informations d'authentification des utilisateurs doivent être stockées de manière sécurisée. Seuls les utilisateurs authentifiés peuvent accéder aux fonctionnalités de l'application.

## 4.3. Compatibilité

L'application doit être compatible avec les navigateurs web les plus courants, y compris Chrome, Firefox, Safari et Edge.

# 5. Exigences techniques

## 5.1. Backend

Le backend sera développé en utilisant ASP.NET Web API. Entity Framework Core sera utilisé pour la gestion de la base de données.

## 5.2. Frontend

Le frontend sera développé en utilisant React. Il doit être responsive et offrir une bonne expérience utilisateur sur les appareils de bureau et mobiles.

## 5.3. Base de données

La base de données sera conçue pour stocker les informations des utilisateurs, des groupes, des membres du groupe et des cadeaux. La base de données qui sera utilisée sera Microsoft SQL Server.

# 6. Livrables

Code source du backend et du frontend.

Documentation du code.

Guide d'installation et d'utilisation de l'application.

# 7- Tâches à compléter pour le Sprint 3

Dans ce sprint 3 nous allons implémenter les fonctionnalités suivantes :

* Compléter la page d’aide (fait le 05/15/2023)
* Compléter Envoyer l’invitation (fait partiellement; elle s’envoie mais n’est pas listée dans les invitations en attente)
* Styler la liste de souhait et y ajouter un titre. Ajouter de la validation de champs (fait le 05/15/2023)
* Ajouter un champ EventDate dans la table Groups et refactorer le code front/backend (fait)
* Programmer la fonctionnalité de la pige des participants entre eux (fait le 05/14/2023)
* Récupérer le mot de passe (fait)
* Modifier l’onglet du groupe pour afficher la personne pigée (fait le 05/13/2023)
* Ajouter les vérifications que si la pige a été exécutée, on ne peut pas rajouter de participant au groupe (fait le 05/15/2023).
* Ajouter courriel vérification dans updateProfile et createProfile - vérifier que le courriel n’existe pas déjà (fait le 05/15/2023)
* Régler le problème de responsiveness pour le site web (fait le 05/15/2023)
* Optionnel : Ajouter le script de création de groupe et d’usagers avec Fluent API dans le backend pour remplacer les scripts SQL user.sql et groupusers.sql.
* Hash les mots de passe dans la base de données.
* Lorsqu’un utilisateur invité par courriel clique sur le lien web dans ce courriel et qu’il est redirigé à la page d’inscription, son prénom, nom, courriel sont automatiquement remplis dans le formulaire. Si cet utilisateur existe déjà dans le groupe (courriel unique par groupe), il sera redirigé vers la page de login.
* Fix : après la création d’un groupe, rediriger vers la page de liste des groupes.

# 8- Schéma de Base de Données (mise à jour)

A screenshot of a computer

Description automatically generated

**Crée une table indépendante : PendingNewUsers**

A screenshot of a computer

Description automatically generated