

MODÉLISATION

INTRODUCTION

Dans ce laboratoire, nous allons reprendre le travail accompli au laboratoire 1 et 2 et appliquer les concepts de modélisation UML vus en classe.

PÉNALITÉ DE RETARD

Pénalités de 10% par jour de retard, 5% par demi-journée.

TRAVAIL DEMANDÉ

Vous allez réaliser les diagrammes UML suivants avec LucidChart :

Diagramme de cas d'utilisation (lien):

Modélisez le logiciel au complet en utilisant un ou plusieurs diagramme en suivant les directives ci-dessous.

ACTEUR : Utilisateur du réseau social basé sur la proximité.

FONCTIONNALITÉS :

- "Création de Groupes" : L'acteur peut créer des groupes locaux, ajouter ou retirer des membres et supprimer des groupes.
- "Publication de Contenus" : L'acteur peut sélectionner des photos ou des vidéos à partager, les téléverser sur son profil ou un groupe spécifique et ajouter des commentaires ou des hashtags.
- "Messagerie" : L'acteur peut envoyer des messages privés, créer des conversations de groupe, et recevoir des notifications de nouveaux messages.
- "Découverte" : L'acteur peut activer ou désactiver la localisation pour trouver des utilisateurs et des événements à proximité, s'inscrire à des événements et interagir avec les publications d'autres utilisateurs.
- "Gestion du Profil" : L'acteur peut gérer son profil personnel en mettant à jour ses informations, en ajoutant des centres d'intérêt, en modifiant ses paramètres de confidentialité et en gérant ses abonnements.

Assurez-vous que les interactions clés entre l'utilisateur, le logiciel de réseau social, et les autres acteurs illustrent comment l'utilisateur se sert des fonctionnalités pour gérer et animer sa vie sociale en ligne, en se connectant avec des personnes et des événements locaux, en partageant des expériences et en personnalisant son expérience utilisateur.

Diagramme d'activités :

UN COMMERCE ENVOIE UNE NOTIFICATION

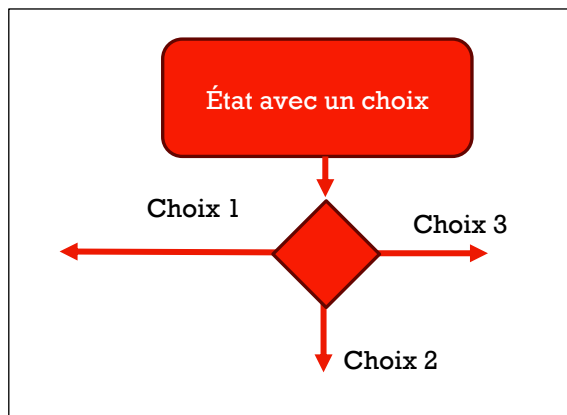
Modélisez un diagramme d'activité pour le processus d'envoi d'une notification par un commerce en suivant les instructions suivantes ci-dessous.

Voici les étapes :

1. **Démarrer le processus d'envoi de notification**
2. **Sélectionner le type de notification** (Promotion, Information, Urgence)

3. **Choisir le public cible** (Tous les clients, Clients VIP, Clients à proximité)
4. **Aperçu du public potentiel pour l'envoi de la notification** (Acceptent-ils les notifications ? Ont-ils préféré des types spécifiques de messages ?)
5. **Rédiger le contenu de la notification**
6. **Planifier l'envoi** (Immédiat, À une date/heure spécifique)
7. **Envoyer la notification**
8. **Confirmer la réception de la notification par le système**
9. **Enregistrer l'action dans le journal des activités du commerce**

Chaque étape où un choix doit être effectué (comme le type de notification, le public cible, les préférences des clients, et la méthode d'envoi) serait représentée par un état suivi d'un losange de choix vide avec les nom des choix sur les flèches:



Voici une description de ces décisions dans le diagramme :

- **Losange de choix pour le type de notification :**
 - Si 'Promotion', alors aller vers la rédaction de contenu promotionnel.
 - Si 'Information', procéder avec des informations générales.
 - Si 'Urgence', rédiger un message d'alerte immédiate.
- **Losange de choix pour le public cible :**
 - Si 'Tous les clients', sélectionner la base de données client complète.
 - Si 'Clients VIP', filtrer pour les clients marqués comme VIP.
 - Si 'Clients à proximité', utiliser les données de localisation pour filtrer les clients.
- **Losange de choix pour l'aperçu du public potentiel pour l'envoi de la notification :**
 - Si l'émetteur accepte, continuer avec la rédaction
 - Si non, revenir au choix de notification
- **Losange de choix pour la planification de l'envoi :**

- Si 'Immédiat', envoyer sans délai.
- Si 'Planifié', paramétrer le système pour envoyer à la date/heure définie.

UN UTILISATEUR REÇOIT UNE NOTIFICATION DE PROXIMITÉ

Modélisez l'utilisateur qui reçoit une notification de proximité avec un diagramme d'activités :

1. On assume, l'utilisateur est authentifié.
2. Le système vérifie les paramètres de localisation de l'utilisateur
3. Si la localisation est activée, continuez ; sinon, finissez le processus.
4. Le système cherche des notifications pertinentes basées sur la proximité
5. Si des notifications sont trouvées, continuez ; sinon, finissez le processus.
6. Le système vérifie les préférences de l'utilisateur
7. S'il y a une adéquation, alors l'utilisateur reçoit une liste de notifications les plus pertinentes
8. L'utilisateur décide d'ouvrir la notification ou de l'ignorer
9. Si l'utilisateur ouvre la notification, il est redirigé vers le contenu pertinent et un trajet est proposé ; sinon, le processus se termine.

CHANGER LE RAYON DE PROXIMITÉ

Modélisez le changement de valeur de rayon de proximité par l'utilisateur

- On assume, l'utilisateur est authentifié.
- L'utilisateur va dans le menu préférences
- Il choisit une des valeurs proposées
- Si la distance est de plus de 5km, une alerte apparaît pour signaler que la distance est difficilement accessible à pied.
- Enregistrer les modifications dans la base de données et rafraichir l'écran.

MODÉLISER LA CRÉATION D'UN NOUVEAU COMMERCE

- Connexion en tant qu'utilisateur
- Accès au panneau commerce. Si non inscrit en tant que commerce proposez de s'enregistrer comme commerce.
- Ajout d'un nouveau commerce
 - Par défaut si aucun n'était là
 - Sélection de l'option 'créer un nouveau commerce' dans l'application
- Renseignement des informations de base du commerce (nom, adresse, type d'activité)
- Configuration des détails du commerce (description, horaires, contact)
- Configuration de l'inventaire (ajout de produits ou services, gestion des stocks)
- Personnalisation de l'apparence de la page du commerce (logo, images, thème)
- Prévisualisation et validation du profil du commerce
- Publication du Profil du Commerce sur l'application pour qu'il soit accessible aux clients
- Ajout et planification d'envoi de notifications professionnelles

Diagrammes de classe :

Modélisez les classes suivantes. Suivez les énoncés ci-dessous, mais assurez de rajouter la visibilité et les signature des méthodes :

MODÉLISER L'UTILISATEUR :

Classe Utilisateur

- id : int (private)
- nom : String (private)
- prenom : String (private)
- dateDeNaissance : Date (private)
- adresse : String (private)
- courriel : String (private)
- preferences : Preferences (private)
- localisation : Localisation (private)
- recevoirNotification(notification : Notification) : void (public)
- modifierPreferences(preferences : Preferences) : void (public)
- suivreCommerce(commerce : Commerce) : void (public)
- consulterCommerces() : List<Commerce> (public)

MODÉLISER LE COURS :

Classe Notification

- id : int (private)
- titre : String (private)
- message : String (private)
- type : String (private)
- dateEnvoi : DateTime (private)
- statutLecture : boolean (private)
- envoyerUtilisateur(utilisateur : Utilisateur) : void (public)
- envoyerTous() : void (public)

MODÉLISER LE GESTIONNAIRE DE COMMERCE:

Classe GestionnaireDeCommerce : Hérite d'utilisateur

- idCommerce : int (private)
- nomCommerce : String (private)
- descriptionCommerce : String (private)
- adresseCommerce : String (private)
- creerNotification(notification : Notification) : void (public)
- modifierCommerce(informations : InformationsCommerce) : void (public)
- analyserStatistiques() : Statistiques (public)

MODÉLISER LA CLASSE PRÉFÉRENCES

La classe Préférences permettrait de gérer les préférences de l'utilisateur pour les alertes, les notifications promotionnelles, le rayon d'acceptation en kilomètres, le thème de l'application et le langage. Les méthodes permettent d'activer/désactiver les alertes et les notifications promotionnelles, et de changer le thème et la langue de l'application.