

Projet

Système de gestion d'envoi de message en masse

1. Objectif

- L'entreprise **Locatis** est un bailleur gérant un parc de 5000 logements sur une moyenne agglomération semblable à celle de Dijon.
- Elle souhaite pouvoir informer de manière automatique et rapide ses locataires, ses partenaires et ses employés au moyen de l'envoi de message de type mail ou sms.
- L'entreprise **Locatis** veut développer une application informatique sur mesure adaptée au plus juste à ses besoins.

2. Contexte

- L'entreprise **Locatis** veut communiquer auprès de ses locataires dans plusieurs cas :
 - Communication liée à certains évènements :
 - Intervention technique dans les logements ou les bâtiments
 - Mise en route du chauffage
 - Enquête auprès des locataires
 - ...
 - Communication marketing et institutionnelles
 - Vente de logements
 - Logements à louer
 - Rappels sur les bonnes pratiques d'usage des logements
 - Les moyens de paiement du loyer
 - ...
 - Communication d'urgence
 - Pannes techniques dans les logements ou les bâtiments
 - ...
- Elle veut aussi communiquer auprès de ses salariés quand ils sont concernés par la communication locataire ou pour des besoins internes de gestion des ressources humaines :
 - Assemblées du personnel,
 - Informations générales
 - ...

- La société **Locatis** souhaite maintenir dans l'application des listes de diffusion pour l'ensemble des publics visés. Ces listes peuvent être créées :
 - par saisie manuelle
 - Par importation de fichiers textes csv ou xml.
 - Par importation et modification manuelle.
- Les destinataires seront classés par catégories afin de pouvoir les sélectionner sur ce critère lors d'un envoi ou pour d'autres opérations. Un même destinataire pouvant faire partie de plusieurs catégories.
- L'entreprise **Locatis** veut pouvoir préparer par avance des campagnes de communication et les planifier pour un ou plusieurs envois ultérieurs selon différents scénarios (Envoi unique, envoi journalier, envoi hebdomadaire etc ...) et pendant une période donnée (Par exemple, toutes les semaines pendant 1 mois, tous les mois pendant 1 an, tous les ans le 24 décembre etc ...)
- Les destinataires devront pouvoir être sélectionnés selon plusieurs critères (Par exemple le bâtiment où ils habitent, leur ancienneté, leur age etc ..)
- Une campagne devra pouvoir être créée par duplication d'une campagne existante.
- Le logiciel devra contrôler la présence et la validité d'adresses ou de numéro de mobile avant d'effectuer le lancement d'une campagne.
- Le logiciel devra disposer d'outils de recherche et de statistiques (Liste des messages envoyés à un destinataire particulier, liste des destinataires de plusieurs campagnes, etc ...)

3. Résultat attendu

- On souhaite mettre en place un logiciel multi-utilisateurs avec une interface graphique disposant de toutes les fonctionnalités nécessaires à la maintenance et l'utilisation des destinataires et le lancement de campagnes de communication en masse.
- On devra disposer de 3 natures d'utilisateur :
 - Administrateur : Création des utilisateurs, création des destinataires, gestion des droits. Droits des gestionnaires.
 - Gestionnaire : Création, planification et lancement des campagnes. Un gestionnaires n'est autorisé qu'aux catégories d'utilisateurs accordées par l'administrateur. Droits des utilisateurs.
 - Utilisateur : Consultation des statistiques. Planification ou lancement d'une campagne si l'administrateur ou un gestionnaire lui donne les droits.
 -

- On devra disposer d'accès différenciés par type d'utilisateur, sécurisés par mot de passe.
- On devra pouvoir consulter les campagnes et avertir le responsable lorsqu'elles seront lancées.

4. Environnement de développement

- Le logiciel sera développé en **Java** avec une interface graphique utilisant la bibliothèque **Swing**. Les données seront stockées dans une base de donnée **MySql** ou **MariaDb** accédée au travers de la bibliothèque **JDBC**.

Projet

Attendus et évaluation

Travail à réaliser (en groupe)

1. Analyse des besoins et rédaction du dossier de spécification des besoins
 - En partant de l'énoncé précédent et en le considérant comme l'expression initiale du besoin de votre client
 - Vous compléterez cette expression des besoins par autant d'entretiens que nécessaire avec le client (Votre enseignant tiendra ce rôle) afin de préciser tout ce qui vous semblera utile,
 - Vous rédigerez le cahier des charges (dossier de spécification des besoins) selon le format de la norme **IEEE Std 830-1998**,
 - Vous intégrerez à ce dossier Les diagrammes UML qu'il prévoit.
2. Sur la base du document produit à l'étape précédente vous concevrez et réaliserez une application répondant aux besoins exprimés :
 - En langage **Java** et utilisant la bibliothèque **Swing** pour réaliser l'IHM,
 - Utilisant les principes de la programmation orientée objet étudiés en cours (encapsulation, composition, héritage, polymorphisme, classes abstraites, interfaces et programmation générique) et les fonctionnalités spécifiques du langage **Java** tels que les exceptions, les collections etc ...
 - Documentée au moyen de **Javadoc**,
 - S'appuyant sur une architecture MVC (Modèle-Vue-Contrôleur) avec séparation des données, de la vue et du contrôleur,
 - Utilisant l'API **JDBC** afin d'assurer la persistance des données au moyen d'une base de données.
3. Au moyen du Framework **JUnit**, vous élaborerez et mettrez en œuvre un ensemble complet de tests unitaires afin de valider le fonctionnement de vos classes.
4. Vous gérerez le « versioning » et le travail collaboratif au moyen de **Git** et **Git Hub**
5. Vous produirez une documentation de l'application reprenant notamment :
 - Le rôle de chaque classe et chaque méthode (Un extrait de la **Javadoc** peut convenir)
 - Les principales fonctionnalités applicatives et leur mode d'emploi.
 - Un schéma de la base de données.

Projet

Attendus et évaluation

Les livrables

1. Dossier de spécification des besoins (Format PDF),
2. Codes sources **Java** de l'application (accompagnés des bibliothèques non standards utilisées : fichiers *.jar* ou *.zip*),
3. Export de la base de données (Fichier *.sql*),
4. Documentation au format PDF.

Les livrables sont à déposer sur l'**ENF du CNAM** dans l'espace prévu à cet effet avant la soutenance avant une date qui sera précisé ultérieurement.

Soutenance

Chaque binôme présentera le travail réalisé au cours d'une soutenance d'une durée de 20 à 25 minutes, tenant lieu d'examen final.

Le déroulé de la soutenance suivra le plan général suivant :

- Résumé du besoin exprimé tel que compris,
- Résumé du travail effectué : Choix réalisés, difficultés rencontrées ...
- Présentation rapide du code rendu (Architecture retenue, principales classes, base de données),
- Démonstration de quelques fonctionnalités

Chacun des membres du groupe devra s'exprimer de manière équivalente lors de cette soutenance.

Évaluation

Le travail réalisé sera évalué selon les critères suivants réparti en 4 catégories :

1. Cahier des charges :
 - Présentation, lisibilité et qualité rédactionnelle
 - Complétude
 - Respect des besoins exprimés
2. Code source
 - Qualité du code (Modularité, lisibilité, respect des conventions, règles d'écriture, commentaires)
 - Qualité de la mise en œuvre des notions du cours (Concepts de la POO, programmations abstraite et générique, patrons de conception, persistance des données)
 - Stratégie de test (Qualité du code, pertinence, couverture)
 - Qualité intrinsèque de l'IHM et de son codage (Séparation vues et données)
 - Publication et suivi des versions

Projet

Attendus et évaluation

Évaluation (suite)

3. Application réalisée
 - Adéquation aux exigences du cahier des charges
 - Ergonomie de l'IHM
 - Qualité de la documentation (Présentation, clarté, complétude, niveau d'aide à la prise en main de l'application)
4. Présentation
 1. Qualité de la présentation
 2. Qualité de la collaboration au sein du groupe.