



Directives *L'objet de ce projet est de créer une application Web permettant la gestion d'un parc de clefs pour l'Enssat.*

1 Présentation générale du sujet

L'Enssat est un établissement destiné à accueillir du public. Pourtant, de nombreux matériels coûteux, fragiles ou sensibles doivent être protégés. Certaines pièces (bureaux, labos, scolarité) sont donc fermées à clefs en permanence et d'autres (salles de TPs, TDs, amphithéâtre, ...) à certaines occasions. Ainsi, l'administration est souvent amenée à prêter une ou des clefs à des personnels, des extérieurs et des étudiants. Le suivi de ces emprunts est difficile à gérer pour un établissement de l'Enssat et le besoin d'un logiciel se fait sentir.

D'autre part, les portes sont équipées de canons (partie de la serrure acceptant la clef). Une clef peut exister en plusieurs exemplaires (généralement 4) et sont associées à un canon. Les canons et les clefs associées sont produites par des fournisseurs. Tous les canons ne s'adaptent pas à toutes les serrures. Il faut donc garder en mémoire la correspondance entre une porte et un canon. Ainsi, lorsque des clefs d'une porte ont été égarées, il est possible d'en commander de nouvelles auprès du fournisseur.

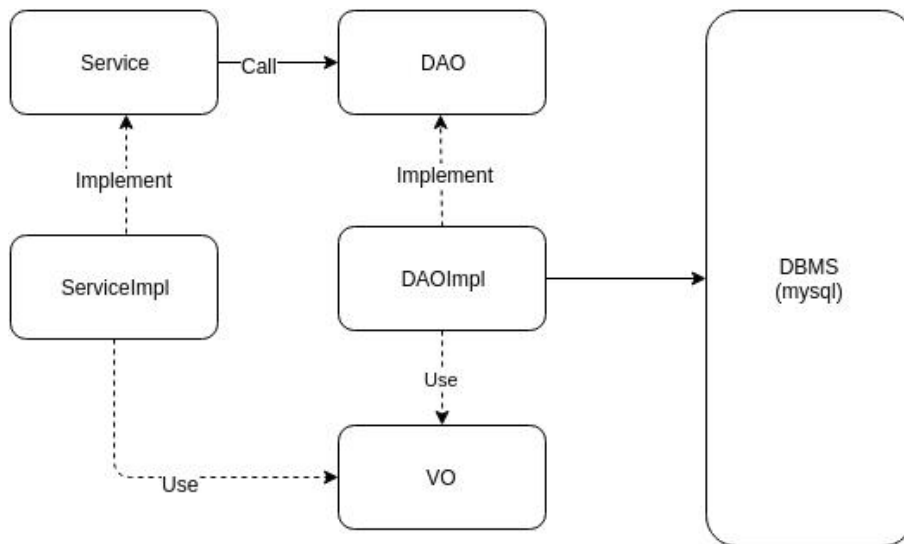
De plus, il existe des passes qui permettent d'ouvrir plus d'une porte. Les passes partiels ouvrent une série de portes. Les passes généraux ouvrent toutes les portes.

On vous demande de réaliser une application permettant de suivre le prêt des clefs et de gérer le stock de clefs (fournisseur? canon installé? sur quelle porte?).

2 Travail demandé

2.1 Contraintes techniques

1. L'application doit pouvoir tourner sur l'intranet de l'Enssat, sur le serveur apache. Elle devra donc être réalisée en PHP / HTML5.
2. Vous disposez d'un petit ensemble de classes PHP **simulant** une base de données. Vous ne créerez pas la base de données.
3. Ces classes sont organisées avec les schémas (design pattern) "DAO" (Data Access Object) et "VO" (Value Object). Les VO sont manipulés par des classes de services comme le service "emprunt" (donné) qui associe à une personne un trousseau de clef.
4. Les classes DAO n'offrent pas de persistance. Cependant, dans le futur de l'application, elle pourra être raccordée à une vraie BDD. Par conséquent, elles implémentent une interface et toute classe DAO d'accès à la future BDD devra implémenter cette interface.
5. La classe de service fournie (et celles que vous implémenterez) simulent les relations d'une BDD. Elle est juste présente pour vous permettre de développer la partie métier (pas la modélisation). Remarquez que l'on a utilisé le schéma "Injection de dépendance" pour lier le service avec nos implémentations spécifiques de DAO.



6. L'implémentation de l'application doit être modulaire.

7. Pour le développement, chaque équipe utilisera un projet GIT, sur gitlab. L'encadrant s'ajoutera à l'équipe de développeur pour connaître l'avancée du projet entre deux séances en présentiel. Cela suppose que vous l'invitez. Chaque développeur "signera" ses contributions dans le code (soit dans le cartouche du fichier, soit à la ligne).

3 Cahier des charges

Par définition, le cahier des charges est un document négociable. Il est toujours incomplet, incohérent et obscur. En effet, il est produit par un client qui n'a pas la connaissance de la technologie que vous allez employer. Pour toute question sur une fonctionnalité ou tout aménagement, contactez barreaud@enssat.fr.

Le logiciel gestion clés sait gérer simultanément les liens entre Clés + Portes + Personne.

Vous pourrez vous inspirer du logiciel décrit ici : <http://www.antibulle.com/gestion-de-cles.html>

3.1 Cas d'usage

3.1.1 Cas d'usage 1: correspondance salle-clef

Quelle est la clé (ou quelles sont les clés) qui ouvre(nt) la porte de la salle 101d ?

3.1.2 Cas d'usage 2 : imputabilité

Quelles sont les personnes qui ont ou ont eu accès à la salle 104dbis ?

3.1.3 Cas d'usage 3 : première utilisation

Vous pouvez enregistrer très facilement les trousseaux déjà prêtés (clés + emprunteurs + date du prêt etc...).

3.1.4 Cas d'usage 4 : import de fichiers

Vous pouvez importer des fichiers (Personnes, clefs, trousseau) au format CSV.

3.1.5 Cas d'usage 5 : mouvement de personnels / prêts aux étudiants

Quelles sont les clés que possède telle ou telle personne ?

3.1.6 Cas d'usage 6 : gestion des stocks

Quelles sont les clés à commander ? quelles sont les caractéristiques de la porte de la salle 202 ? (longueur du barillet par exemple, ce qui évite d'aller vérifier sur place en cas de besoin). Quel est le fournisseur de tel clé ?

3.1.7 Cas d'usage 7 : prêt de trousseau de clefs

Pour chaque prêt, un trousseau est constitué (il peut ne contenir qu'une clef). Le trousseau est associé à une personne et une date de retour peut lui être ajoutée. La relation entre la personne et le trousseau est alors dans l'état "Emprunté". Elle peut passer dans l'état "Perdu" ou "Rendu" (avec une date associée). Le logiciel produit un reçu (pdf) à signer par l'emprunteur.

3.1.8 Cas d'usage 8 : quelle est cette clef ?

Préciser si un clef est une clef simple (ouvre une porte), un passe partiel (dan ce cas donner les portes qu'elle ouvre) ou un passe total. Les clefs existent en plusieurs exemplaires. Elles ne sont pourtant pas numérotées (i.e. qu'il existe 4 exemplaires de la clef de la 101D et ils ont tous le même numéro).

3.1.9 Cas d'usage 9 : perte de clef

Si une personne rapporte qu'elle a perdu un trousseau, la relation de prêt doit alors passer dans l'état "Perdu".

3.1.10 Cas d'usage 10 : retour de clef

Quand une personne rapporte un trousseau, la relation de prêt doit alors passer dans l'état "Rendu". Le trousseau est physiquement défait et le stock de clefs actualisé.

3.1.11 Cas d'usage 11 : remplacement d'un canon de porte

Il faut pouvoir notifier toutes les personnes ayant accès à cette porte d'emprunter une nouvelle clef.

3.1.12 Cas d'usage 12: constituer un trousseau

Une personne souhaite accéder à un ensemble de pièces. L'application donne un ensemble de listes des clefs nécessaires. Ces listes peuvent contenir un ou des passes. Attention: pour les passes, préciser quelles portent sont concernées, pour ne pas donner accès accidentellement à une zone protégée. Une fois le trousseau validé, le stock des clefs est rectifié.

3.1.13 Cas d'usage 13: dupliquer un trousseau

Certaines salles sont souvent utilisées ensemble (salles de TD 1xxH, pour les Projets Management). Pour ne pas avoir à reconstituer des trousseaux à chaque PM, on pourra dupliquer une relation de prêt en changeant l'emprunteur, réactualisant la date de prêt et de retour attendu.

3.1.14 Cas d'usage 14: prolonger un emprunt

Dans le cas où la relation de prêt contient une date de retour attendu, il faut être capable de prolonger le prêt.

3.1.15 Cas d'usage 15: relance

Les emprunteurs n'ayant toujours pas rendu leur trousseau après la date prévue seront mis dans une liste visible par l'administrateur, pour une relance.

4 Conduite du projet

4.1 Organisation du développement

Les 6 (6) équipes de projet seront désignées par l'encadrant en séance. Elles doivent être équilibrées selon les origines des étudiants (MMI, INFO, RT).

Une équipe de projet désigne un de ses membres comme chef de projet.

Le chef de projet mène les discussions relatives

- au découpage en tâche (groupes de fonctionnalités),
- à l'évaluation de la durée de ces tâches
- à l'attribution de ces tâches,
- au séquençement des tâches,
- au échéances des tâches.

Le chef de projet peut organiser un point en début de chaque séance. Le chef de projet doit rendre compte de l'avancement à l'encadrant à chaque séance.

4.2 Délivrables

Les équipes de projet devront chacune fournir les livrables décrits dans la section suivante.

La démonstration du produit aura lieu sur rendez-vous avec l'encadrant. Le but de la démonstration (10 minutes) est de présenter les fonctionnalités implémentées et de justifier des choix techniques pertinents. Tous les membres de l'équipe doivent prendre la parole.

Il vous est demandé de fournir, une semaine avant votre démonstration:

un cahier de conception Ce cahier détaille les fonctionnalités élémentaires implémentées, explique les raisons qui ont conduit à leurs implémentations (fonctionnalité élémentaire, fondamentale, ...) et expose d'éventuelles spécificités d'implémentation (un algorithme ou une structure de donnée particulière).

un cahier de tests Ce cahier détaille comment un auditeur pourrait tester les fonctionnalités décrites dans le cahier de conception.

un cahier d'installation Ce cahier peut comporter une Foire aux questions pour guider l'installateur en cas de problèmes.

un cahier d'utilisation Ce cahier est à destination des utilisateurs du service. Il décrit dans un langage simple l'interface et le service.

un cahier de projet Ce cahier donnera une vue de la conduite du projet. Il donnera également une évaluation homme/heure pour chaque tâche.

4.3 Evaluation

L'évaluation porte sur la qualité de l'exécution, par sur la quantité: Vous pouvez ne pas implémenter toutes les fonctionnalités. Cependant, vous devez motiver vos choix et chaque fonctionnalité doit être documentée et testée.

Chaque équipe sera évaluée indépendamment de l'autre. Chaque équipe sera évaluée sur sa conduite de projet et la qualité de ses livrables.