Khadra

Table des matières

LISTE DE FIGURES.....6

INTRODUCTION GENERALE.....7

CHAPITRE I ETUDE PREALABLE ET SPECIFIACTION DES BESOINS.....9

1 Introduction.....10

2 Cadre de projet.....10

3 Etude du cas.....10

4 Critique :.....11

5 Solution proposée :.....11

6 Spécification des besoins.....11

6.1 Les besoins fonctionnels :.....12

7 Conclusion.....13

CHAPITRE II ANALYSE ET CONCEPTION.....13

1 Introduction.....14

2 Présentation de l'UML.....14

2.1 Définition :.....14

2.2 Description de la vue statistique (Diagramme de classe 1) :.....16

2.3 Diagramme de séquence et de cas d'utilisation:.....16

2.4 Diagramme de flux :.....19

3 Conclusion.....19

CHAPITRE III REALISATION.....20

1 Introduction.....21

2 Technologies Utilisé.....21

2.1 Framework Spring (SpringBoot) :.....21

2.1.a SpringData:.....22

2.1.b Spring Web et Mvc:.....22

2.1.c Spring Mail :.....23

2.2 Avantage du Spring Boot :.....23

2.3 Technologies supportées :.....24

2.4 Design pattern :.....24

2.4.a Model-View-Controler (MVC):.....25

2.4.b Solide principe :.....27

2.5 Ajax (Asynchronous JavaScript and XML):.....28

2.5.a Qu'est-ce que l'AJAX ?.....28

2.5.b Les formats de données :.....29

2.6 Thymleaf :.....29

2.7 Tomcat :.....30

2.8 Maven :.....31

2.8.a A quoi sert Maven ?.....31

2.8.b Repository Maven :.....31

2.8.c Projet Object Model(POM) :.....31

2.8.d Git Hub :.....33

3 Outils de développement :.....34

4 Environnement de développement.....36

4.1 Netbeans :.....36

5 Description de l'application.....37

5.1 Les interfaces de l'application :.....37

5.1.a Accueil et authentification :.....37

5.1.b Salle :.....39

5.1.c Utilisateur:.....41

5.1.d Reservation:.....43

5.1.e Contact :.....46

6 Conclusion :.....46

CONCLUSION GENERALE ET PERSPECTIIVES.....47

ANNEXE.....48

REFERENCES BIBLIOGRAPHIQUES.....50

LISTE DE FIGURES

Figure II. 1 «Diagramme de classe1 »....15
Figure II. 2: « Diagramme de classe 2 »....16
Figure II. 3: « Diagramme de séquence et cas d'utilisation de réservation dans un intervalle» 17
Figure II. 4: « Diagramme de séquence et cas d'utilisation d'authentification »....18
Figure II. 5: « diagramme de flux »....19
Y
Figure III. 1: « rôle de ces éléments »....26
Figure III. 2: « Le schéma de la structure générique d'une architecture MVC »....27
Figure III. 3: «fichier POM.xml»....33
Figure III. 4:« WampServer »....34
Figure III. 5:«MariaDB»....35
Figure III. 6: « interface de NetBeans IDE 8.2 »....37
Figure III. 7 :« L'interface authentification»....37
Figure III. 8:« l'authentification de l'administrateur »....38
Figure III. 9: « Liste et formulaire des salles»....38
Figure III. 10:« Liste et formulaire des utilisateurs»....39
Figure III. 11:« Formulaire d'ajoute des salles»....39
Figure III. 12:« L'interface de la modification des salles »....40
Figure III. 13:«L'interface de la modification des utilisateurs »....42
Figure III. 14:«L'interface de la suppression un utilisateur »....43
Figure III. 15:« L'interface de la réservation sur l'intervalle»....44
Figure III. 16:« L'interface de la réservation sur calendrier »....44

Figure III. 17:« L'authentification de l'utilisateur »....45
Figure III. 18: « Formulaire de la réservation »....45
Figure III. 19:« L'interface de réservation »....46
Figure III. 20: « Formulaire de contact »....46

INTRODUCTION GENERALE

Le monde informatique est un monde très vaste, son apparition correspond à un changement total de culture, jour après jour l'informatique occupe une place intéressante dans l'administration, puisqu'elle permet de mémoriser, de traiter et de diffuser l'information nécessaire dans un délais très court, qui permettra donc de prendre les bonnes décisions dans le temps voulu, la tendance actuelle des entreprises et des organisations est l'informatisation de leur système d'informations ou une partie de ce système, dans cette vision les facultés de l'université essaie de mettre en œuvre des applications pour ces différents services

En effet, les facultés sont confrontées à des problèmes persistants liés à la réservation des locaux, mais le gros problème est que la gestion des locaux sont des tâches difficile et chronophage.

Mon projet vient d'un coté pour répondre à ce besoin professionnel, et d'un autre coté pour appliquer les connaissances théoriques.

Dans cet objectif, mon travail consiste à réaliser une application web pour administrer les locaux de la faculté. Pour la mise en œuvre de mon application j'ai utilisé Maria DB, ainsi que l'environnement Netbeans pour la réalisation de l'application.

Structure du Mémoire :

Ce mémoire est décomposé en trois chapitres : Le premier chapitre est consacré à

l'Etude préalable et spécification des besoins. Le deuxième chapitre est dédié à la présentation du langage UML (Les définition, les descriptions, et l'utilité de différents diagrammes) ainsi que la modélisation du système étudié, en utilisant les différents diagrammes (diagramme de cas d'utilisation, diagramme de séquence, diagramme de classe et diagramme de flux) d'UML. Le dernier chapitre propose la phase de réalisation et de mise en œuvre de l'application. Finalement, je termine mon mémoire par une conclusion générale et des perspectives.

CHAPITRE I ETUDE PREALABLE

Rapport-Gratuit.com

1 Introduction

L'étude préalable possède des caractéristiques propres qui la différencient clairement d'autres étapes du mon projet**.**

2 Cadre de projet

Ce chapitre généralise la méthode de réservation des locaux.

3 Etude du cas

L'étude de cas peut-être utilisée dans les études pour se pencher sur un cas en particulier. Elle vise l'étude approfondie d'un cas spécifié comme un sujet spécifique.

Mon projet permet de chercher un local de travail (éventuellement pour l'enseignement), de consulter le calendrier de disponibilités et d'effectuer une demande de réservation pour toutes les salles disponibles

Donc pour cela, je cite quelques étapes pour expliquer la méthode de réservation aux dessous :

- L'utilisateur Spécifie une salle dans un intervalle de temps, c'est-à-dire, la Date début et la Date fin.
- Rechercher une salle pour faire une réservation
- L'administrateur Spécifie le Nombre de participants dans un champ obligatoire : Nom salle et le type d'utilisateur soit enseignant soit administrateur.
- Trouver une date.
- Ou activer la réservation pour un calendrier c'est à dire spécifier le nombre de semaine pour réserver une salle.

4 Critique :

Les réservations sont actuellement faites soit sur papier soit directement dans l'administration.

Notre analyse a mis en évidence certaines insuffisances dans le système d'information

actuel, qui sont principalement liées au processus de réservation et d'annulation:

Dans le processus de réservation, l'utilisation d'un moyen au sein de la faculté pour échanger des informations et vérifier la disponibilité des salles, l'équipement et la capacité d'une salle, ce qui est nécessaire pour la bonne coordination des données:

Cette fonctionnalité peut causer un problème (chevauchement) avec un autre enseignent lors de

réservation surtout si deux utilisateurs prennent la même salle au même moment.

5 Solution proposée :

La prise en compte des problèmes évoqués précédemment a aboutit à l'élaboration d'une proposition. Cette dernière introduit des avantages grâce une application web qui gère les salles.

Elle consistera à mettre sur pied une application web afin de faciliter la réservation des locaux.

Des réservations en ligne pour gagner le temps et la rapidité dés la réception de la

Réservation, un email est envoyé à l'utilisateur pour confirmation. Une interface administrateur

sera aussi créée afin de permettre la gestion des réservations et des utilisateurs, pour ainsi remplacer le travail qui a été fait manuellement.

6 Spécification des besoins

L'analyse et la spécification des besoins représentent la première phase du cycle de développement d'un logiciel. Elle sert à identifier les acteurs réactifs du système et leur associer chacun l'ensemble d'actions avec lesquelles il intervient dans l'objectif de donner un résultat optimal et satisfaisant au client.

Ainsi, dans ce chapitre, je commence en premier lieu par une spécification des besoins auxquels doit répondre l'application, passant ensuite à l'analyse de ces besoins à travers l'introduction des acteurs et les diagrammes de cas d'utilisation relatifs à ces acteurs.

6.1 Les besoins fonctionnels :

Les contrôleurs proposés par mon application se résument en quatre gestions :

1. gestion des salles : il gère les salles (ajouter, modifier, supprimer, rechercher..)

2. gestion de la réservation : il gère la réservation (ajouter, modifier, supprimer, rechercher..)

3. gestion de profil : il gère le profil (ajouter, modifier, supprimer, rechercher..)

4. gestion de message : il gère l'envoie de message

Les besoins non fonctionnels décrivent toutes les contraintes auxquelles est soumis le système pour sa réalisation et son bon fonctionnement.

B1. **Ergonomie et souplesse**

L'application doit offrir une interface conviviale et ergonomique exploitable par l'utilisateur en envisageant toutes les interactions possibles à l'écran du support tenu.

B2. **Rapidité**

L'application doit optimiser les traitements pour avoir un temps de génération de schéma raisonnable.

B3. **Efficacité**

L'application doit être fonctionnelle indépendamment de toutes circonstances pouvant entourer l'utilisateur.

B4. **Maintenabilité et scalabilité**

Le code de l'application doit être lisible et compréhensible afin d'assurer son état évolutif et extensible par rapport aux besoins du marché.

7 Conclusion

Dans ce chapitre j'ai cité quelques critiques et j'ai proposé les solutions pour remédier ces problèmes et enfin présenter la spécification des besoins

CHAPITRE II ANALYSE ET CONCEPTION

1 Introduction

Cette partie est consacrée aux étapes fondamentales pour le développement de mon application pour administrer les locaux de faculté. Pour la conception et la réalisation de mon application, j'ai choisis de modéliser avec le formalisme UML (Unified Modeling Language) qui offre une flexibilité marquante qui s'exprime par l'utilisation des diagrammes.

2 Présentation de l'UML

2.1 Définition :

UML (Unified Modeling Language), se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. Il véhicule en particulier: [1]

Les concepts des approches par objets : classe, instance, classification, etc.

Intégrant d'autres aspects : associations, fonctionnalités, événements, états, séquences, etc.

UML définit neuf types de diagrammes devisés en deux catégories:

- Diagrammes statiques (structurels) : diagramme de classe, d'objet, de composant, de déploiement et de diagramme de cas d'utilisation.
- Diagrammes dynamique (comportementaux) : diagramme d'activité, de séquence, d'état-transition et de diagramme de collaboration. [2]

Pour la modélisation des besoins, j'ai utilisé les diagrammes UML suivant : Diagramme de cas d'utilisation, diagramme de séquence, diagramme de classe et diagramme de flux.

2.1 Description de la vue statistique (Diagramme de classe 1) :

Le diagramme de classes est considéré comme le plus important de la modélisation orienté objet. Il s'agit d'une vue statique du fait qu'on ne tient pas compte du facteur

temporel dans le comportement du système. Le diagramme de classe permet de modéliser les classes du système et leurs relations indépendamment d'un langage de programmation particulier

L'intérêt majeur de ce diagramme est de représenter les entités du système d'information

Les diagrammes de classe retenus à la fin de la conception sont les suivants :

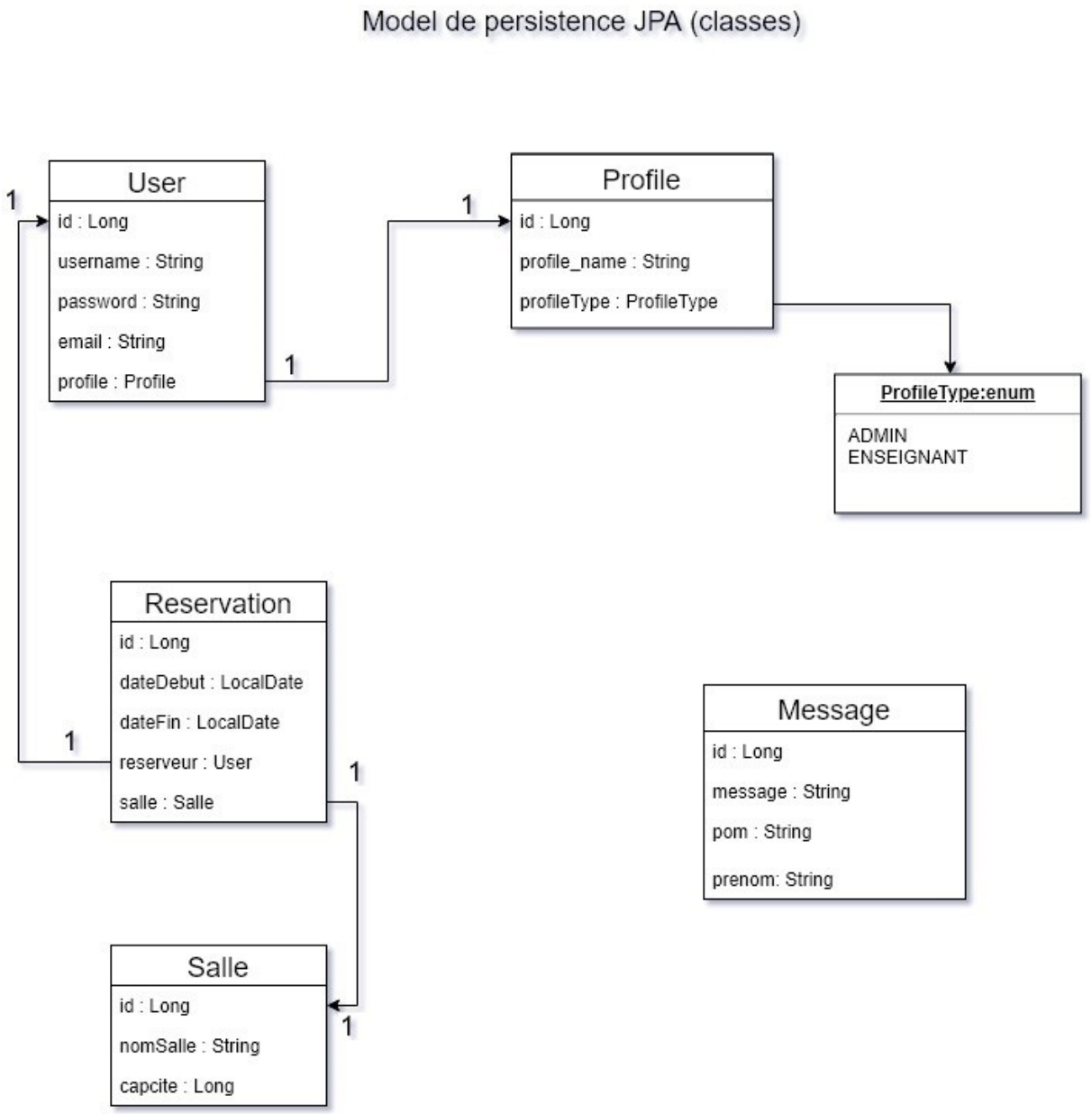

Figure II. 1 «Diagramme de classe1 »

2.2 Description de la vue statistique (Diagramme de classe 2) :

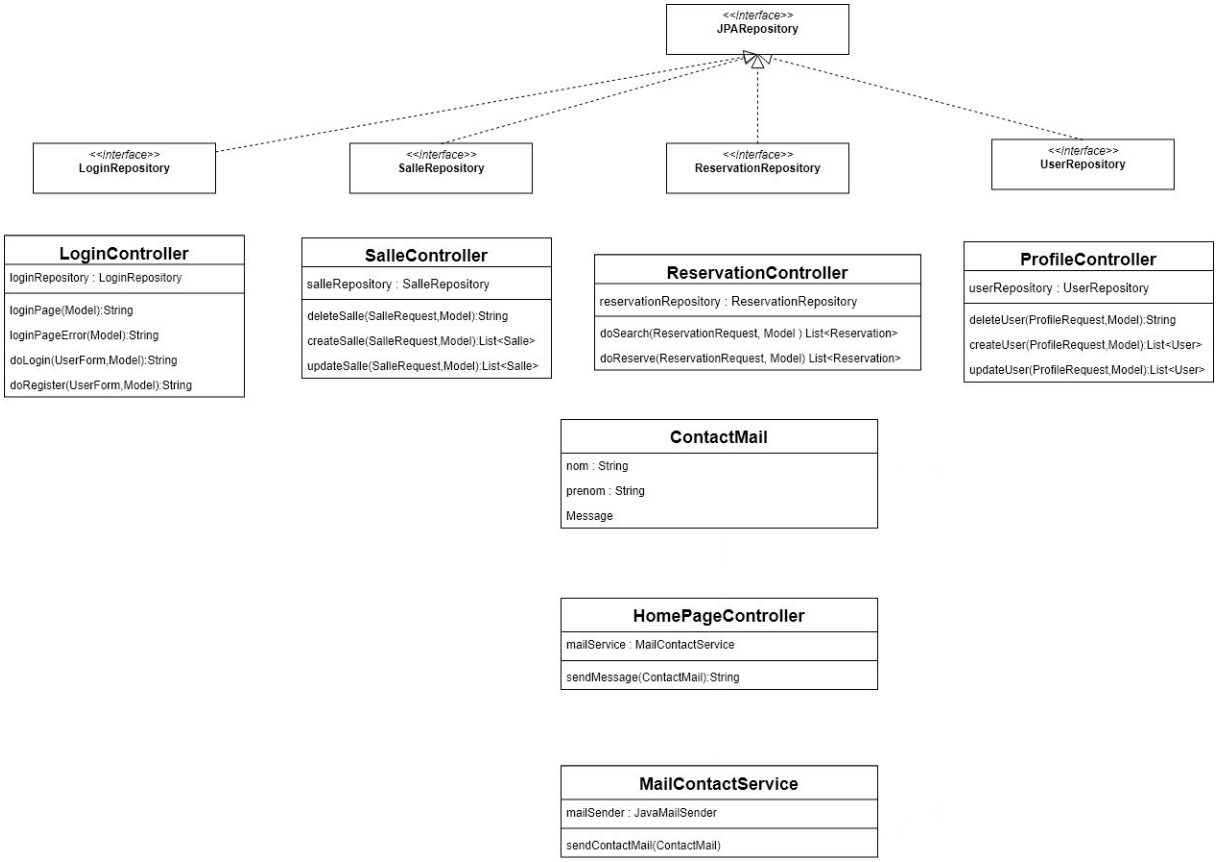

Figure II. 2: « Diagramme de classe 2 »

2.3 Diagramme de séquence et de cas d'utilisation:

Un diagramme de cas d'utilisation est un graphe d'acteurs, un ensemble de cas d'utilisation englobés par la limite du système, des associations de communication entre les acteurs et les cas d'utilisation, et des généralisations entre cas d'utilisation.

Il est destiné à représenter les besoins des utilisateurs par rapport au système. Identification des acteurs : Les acteurs d'un système sont les entités externes à ce système qui interagissent avec lui.

- **Diagrammes de séquence :** les diagrammes de séquence montrent des interactions entre objet selon un point de vue temporel. Généralement ces diagrammes définissent les objets acteurs et le système.

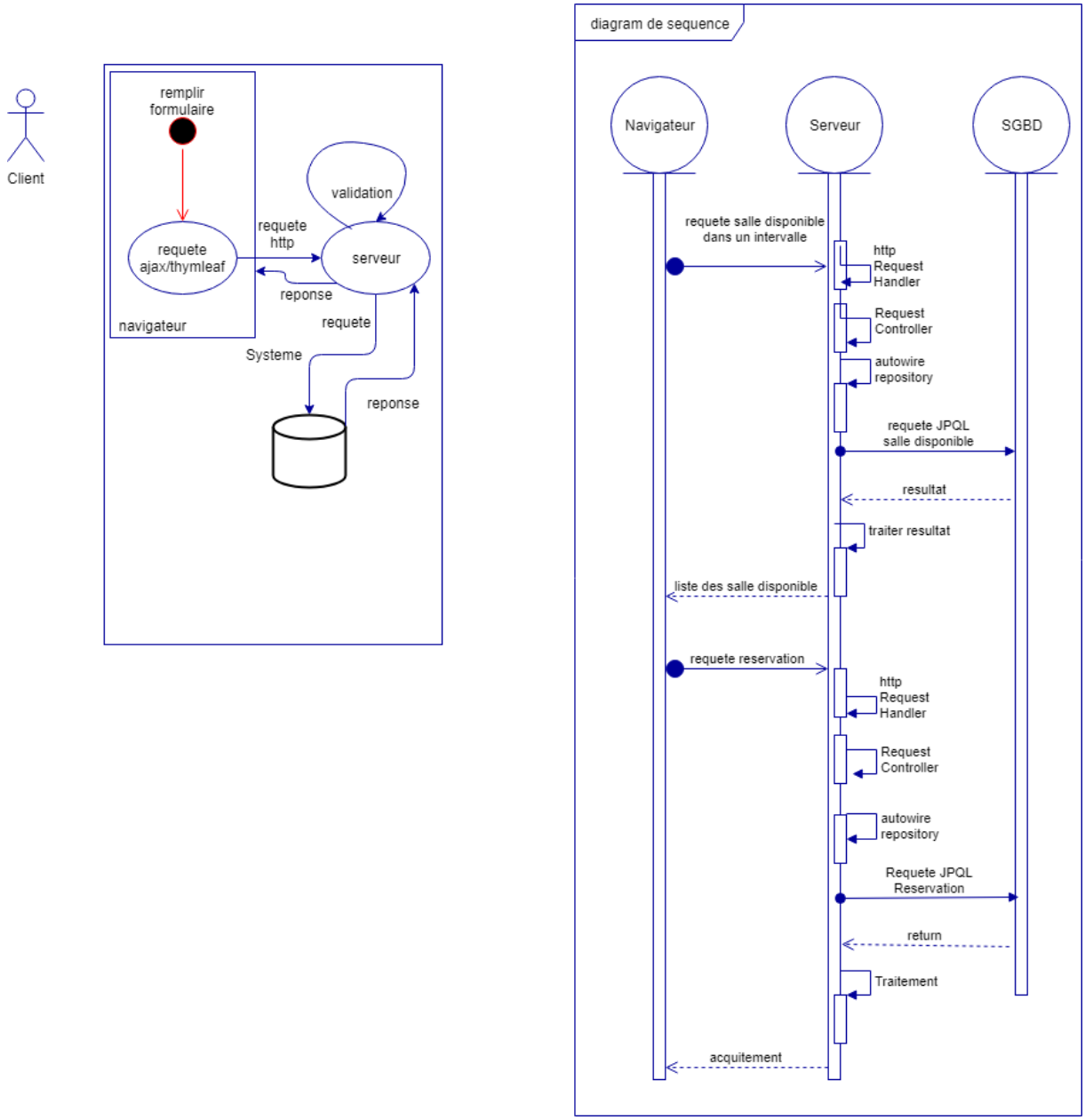

Figure II. 3: « Diagramme de séquence de réservation dans un intervalle»

La description de la vue dynamique de mon application est réalisée à travers les différents diagrammes de séquences. En effet, un diagramme de séquence décrit les

interactions entre les objets du système. Je dois donc, utiliser les différents composants de l'architecture MVC pour mieux éclaircir les tâches.

2.4 Diagramme de séquence et cas d'utilisation:

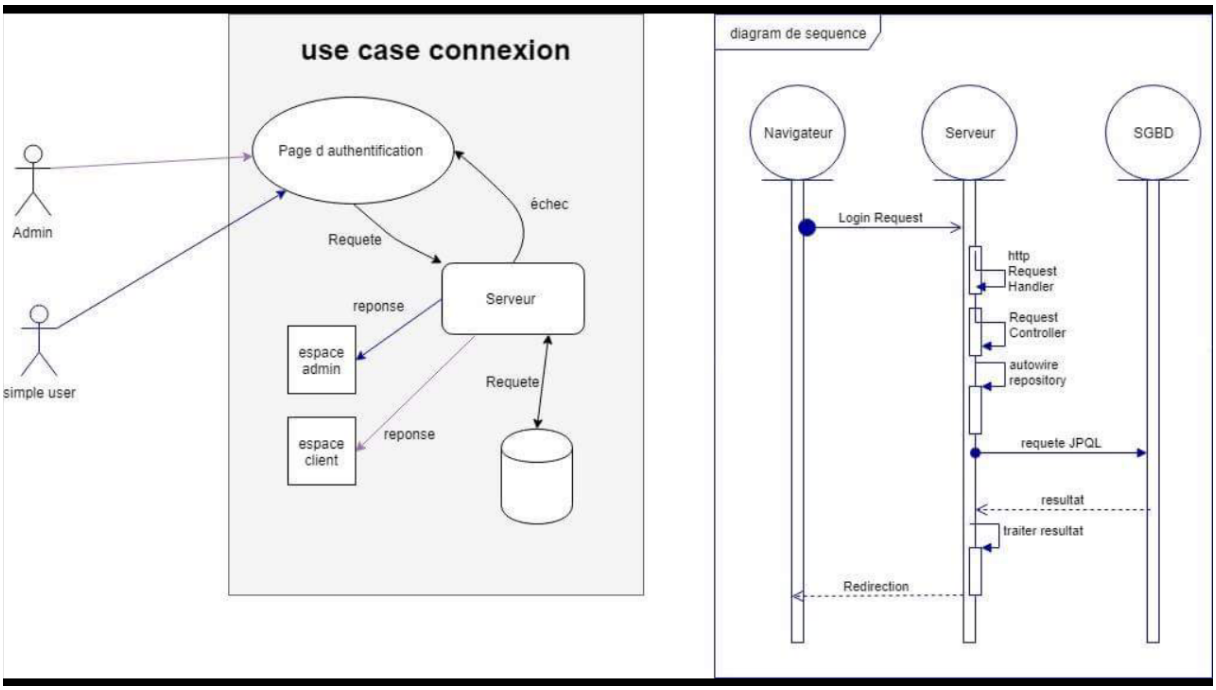

Figure II. 4: « Diagramme de séquence et cas d'utilisation d'authentification »

Diagramme de cas d'utilisation :

- Résumé : ce cas d'utilisation permet à l'administrateur et à l'utilisateur de faire une connexion.
- Acteur : Administrateur, simple user.

Diagramme de séquence :

- Résumé : ce cas d'utilisation permet d'envoyer une requête de connexion depuis le client vers le serveur.
- Envoyer la requête.
- Traitement et manipulation dans la base de données.

- Validation.

Rapport-Gratuit.com

2.5 Diagramme de flux :

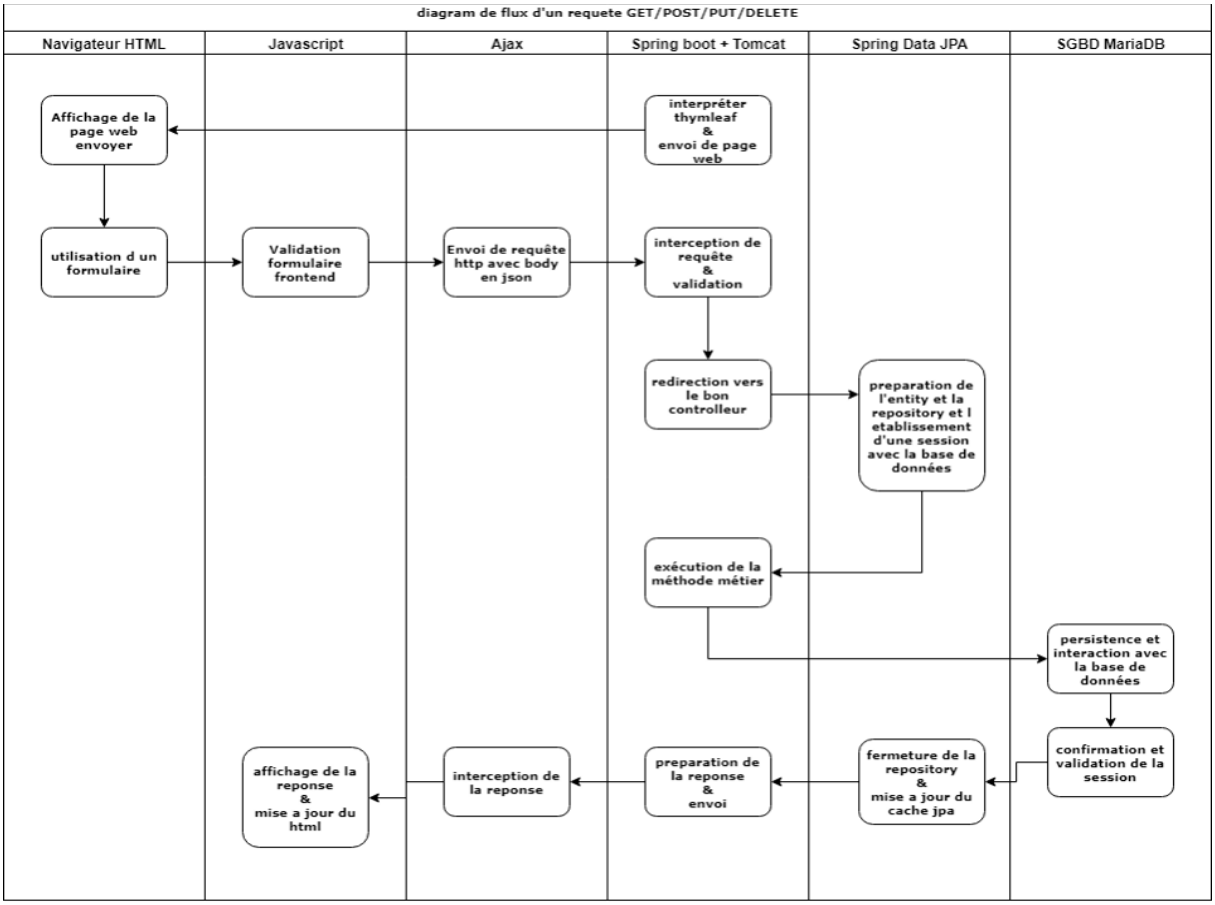

Figure II. 5: « diagramme de flux »

3 Conclusion

Dans ce chapitre, j'ai pu concevoir un système d'information en se basant sur les diagrammes du langage UML à savoir le diagramme de cas d'utilisation, le diagramme de séquence et le diagramme de classe et diagramme de flux.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

CHAPITRE III REALISATION

1

1

1 Introduction

Dans ce chapitre, consacré à la réalisation de mon application d'administrer les locaux des facultés, j'ai présenté les outils de développement adoptés; soient les Framework tels que springboot et la gestion de bibliothèque maven, ainsi que le design pattern et enfin l'environnement utilisé qui est Netbeans 8.2.

2 Technologies Utilisés

2.1 Framework Spring (SpringBoot) :

Spring est un Framework open-source destiné aux applications java (J2EE) pour les applications 3-tiers dont il facilite le développement et les tests.

Ce Framework est très apprécié par la communauté Java pour sa souplesse d'utilisation et sa modularité. En effet, il n'est pas nécessaire d'implémenter la totalité de Spring pour pouvoir utiliser Spring, et c'est ce qui en fait son principal point fort, on utilise que ce qu'on a besoin.

Outre, ce Framework est un Framework robuste et très puissant qui peut gérer de grand nombre de connexion pour des sites énormes qui reçoivent des millions de visiteurs et des milliers de transactions par seconde.

➢ ***Spring Boot :***

On peut utiliser Spring Boot pour créer des applications Java qui peuvent être démarrées en utilisant java -jar ou des déploiements de guerre plus traditionnels. Nous fournissons également un outil de ligne de commande qui exécute des "scripts de spring".

Spring Boot est un projet ou un micro Framework qui a notamment pour but de faciliter la configuration d'un projet Spring et de réduire le temps alloué au démarrage d'un projet. ... Spring a regroupé les dépendances Maven de Spring dans des « méga dépendances » afin de faciliter la gestion, ce qui est nommé en 'starter'.

Il facilite la création d'applications autonomes de production que vous pouvez exécuter. Nous adoptons une vision avisée de la plate-forme Spring et des bibliothèques tierces, afin que vous puissiez commencer avec un minimum d'agitation. La plupart des applications Spring Boot nécessitent très peu de configuration Spring. [4]

2.1.a SpringData*:*

Spring Data est découpé en une couche commune à toutes les sources de données sous-jacentes (Neo4j, MongoDB, JPA…), appelée Spring Data Commons, à laquelle s'ajoute une couche propre à la source de données.

Spring Data a pour mission de fournir un modèle de programmation Spring familier et cohérent pour l'accès aux données tout en conservant les caractéristiques spéciales du magasin de données sous-jacent.

Il facilite l'utilisation des technologies d'accès aux données, des bases de données relationnelles et non relationnelles, des cadres de réduction de la carte et des services de données en nuage. C'est un projet parapluie qui contient de nombreux sous-projets spécifiques à une base de données donnée. Les projets sont développés en travaillant avec de nombreuses entreprises et développeurs qui sont derrière ces technologies passionnantes.

Spring Data JPA, qui fait partie de la famille Spring Data, permet de mettre facilement en place des référentiels basés sur JPA. Ce module traite de la prise en charge améliorée des couches d'accès aux données basées sur JPA. Il est plus facile de créer des applications alimentées par Spring qui utilisent des technologies d'accès aux données.[5]

2.1.b Spring Web et Mvc:

Ces deux modules permettent d'utiliser Spring avec une application web et d'implémenter une architecture MVC.

Ce Framework, grâce à sa couche d'abstraction, ne concurrence pas d'autres Framework dans une couche spécifique d'un modèle architectural Modèle-Vue-Contrôleur mais s'avère un Framework multicouches pouvant s'insérer au niveau de toutes les couches ; modèle, vue et contrôleur.

Le Framework et en particulier le module Spring MVC permettent d'implémenter naturellement un Modèle Vue Contrôleur. L'utilisation des fichiers de configuration XML en font une remarquable implémentation facilement maintenable et évolutive.

Dans Spring MVC, les contrôleurs sont chargés d'appeler les différentes méthodes nécessaires de la classe métier et de transmettre à la vue le modèle contenant toutes les données indispensables à génération de celle-ci. A partir des fichiers de configuration, les requêtes http sont redirigées par un dispatcher vers le contrôleur adéquat. Il existe différents types de contrôleurs qui permettent de rediriger les requêtes selon différents critères :

- adresse URL complète.
- paramètres passés dans l'URL.
- utilisateur connecté à l'application.

De plus, Spring MVC offre la possibilité de développer ses propres types de contrôleur afin de pouvoir rediriger des adresses Url plus complexes. Enfin, Spring MVC autorise l'utilisation des expressions régulières pour la redirection des URL.

Le passage du modèle à la vue se fait par l'intermédiaire d'une map d'Object triée par mot clé. La vue y accède en utilisant le mot clé de l'Object souhaité. [6]

2.1.c Spring Mail :

Est un projet de la famille spring qui permet de gérer l'aspect messagerie dans un projet via des servers smtp tel que Google ou Hotmail afin de les intégrés avec le moindre coup, il peut aussi être intégrer avec notre propre serveur smtp au juste titre de bien le configurer. [7]

2.2 Avantage du Spring Boot :

- Il est facile de développer l'application basée sur Spring avec Java ou Groovy.
- Il diminue énormément du temps et augmente la productivité.

- Il évite d'écrire plusieurs de codes d'expression standard (boilerplate Code), des Annotations et des configurations XML.
- Il est très facile d'intégrer des applications Spring Boot avec ses écosystèmes de Spring comme Spring JDBC, Spring ORM, Spring Data, Spring Security etc.
- Il suit l'approche « Configuration par défaut » afin de diminuer le temps et l'effort de développement.
- Il fournit des serveurs intégrés (Embedded HTTP servers) comme Tomcat, Jetty .... afin de développer et de tester des applications web à la manière la plus facilement et la base de données stockées dans la mémoire data bases facilement.
- Il fournit de tester des applications Spring Boot (Java ou Groovy) à partir de l'invite de commande (command prompt) facilement et rapidement en utilisant les outils de Build comme Maven et Gradle. [8]

2.3 Technologies supportées :

- **Core:** Spring Security, JTA, Spring Cache, Spring Session.
- **Web:** Spring MVC, Web Socket, Jersey, Mobile.
- **Moteurs de Templates:** Freemaker, Thymeleaf, Groovy.
- **Data base :**
  - **SGBDR** : SprinData JPA,JDBC,JOOQ.
  - **NOSQL**: Redis, MongoDB, Elasticsearch, and Cassandra.[9]

2.4 Design pattern :

Un Design Pattern est une solution à un problème récurrent dans la conception d'applications orientées objet. Un patron de conception décrit alors la solution éprouvée pour résoudre ce problème d'architecture de logiciel. Comme problème récurrent on trouve par

exemple la conception d'une application où il sera facile d'ajouter des fonctionnalités à une classe sans la modifier.

A noter qu'en se plaçant au niveau de la conception les Design Patterns sont indépendants des langages de programmation utilisés.

Les Design Patterns étant largement documentés et connus d'un grand nombre de développeurs ils permettent également de faciliter la communication. Si un développeur annonce que sur ce point du projet il va utiliser le Design Pattern Observateur il est compris des informaticiens sans pour autant rentrer dans les détails de la conception (diagramme UML, objectif visé...). [10]

2.4.a Model-View-Controler (MVC):

Après l'évaluation de la technologie, les sites web et les applications ont progressivement évolué, les attentes des utilisateurs et des clients également. De ce fait, mon site web utilise l'architecture MVC. L'architecture MVC (modèle, vue et contrôleur) est un concept très puissant qui intervient dans la réalisation d'une application ou un site web. Son principal intérêt est la séparation des données (modèle), de l'affichage (vue) et des actions (contrôleur).

Le motif est composé de trois types de modules ayant trois responsabilités différentes : les modèles, les vues et les contrôleurs.

- Un modèle (Model) contient les données à afficher.
- Une vue (View) contient la présentation de l'interface graphique.
- Un contrôleur (Controller) contient la logique concernant les actions effectuées par l'utilisateur.

Ce motif est utilisé par de nombreux frameworks pour applications web tels que Ruby on Rails, Django, ASP.NET MVC, Spring, Struts, Symfony, Apache Tapestry ou Angular Js. On peut citez d'autre tel que androïde pour les applications mobiles et swing ou java fx pour les applications desktop qui se base sur MVC.

MVC n'est pas le seul design pattern dans son genre on peut citez MVVM ou MVP, mais vu sa simplicité et sa réputation il reste dans la majorité des projets le choix favorable du développeur.

- ✓ ***Comment fonctionnent une architecture MVC ?***

Le pattern MVC permet de bien organiser son code source. Il va vous aider à savoir quels fichiers créer, mais surtout à définir leur rôle. Le but de MVC est justement de séparer la logique du code en trois parties que l'on retrouve dans des fichiers distincts.

- ➢ ***Modèle :***

Cette partie gère les données de site web. Son rôle est d'aller récupérer les informations dans la base de données, de les organiser et de les assembler pour qu'elles puissent ensuite être traitées par le contrôleur.

- ➢ ***Vue :***

Cette partie se concentre sur l'*affichage*. Elle ne fait presque aucun calcul et se contente de récupérer des variables pour savoir ce qu'elle doit afficher. On y trouve essentiellement du code HTML mais aussi quelques boucles et conditions PHP ou JavaScript très simples, pour afficher par exemple une liste de messages.

- ➢ ***Contrôleur :***

Cette partie gère la logique du code qui prend des décisions. C'est en quelque sorte l'intermédiaire entre le modèle et la vue : le contrôleur va demander au modèle les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la vue. Le contrôleur contient exclusivement du code au langage Framework serveur (java pour Spring, php pour Laravel).

C'est notamment lui qui détermine si le visiteur a le droit de voir la page ou non (gestion des droits d'accès). [11]

La figure suivante schématise le rôle de chacun de ces éléments.

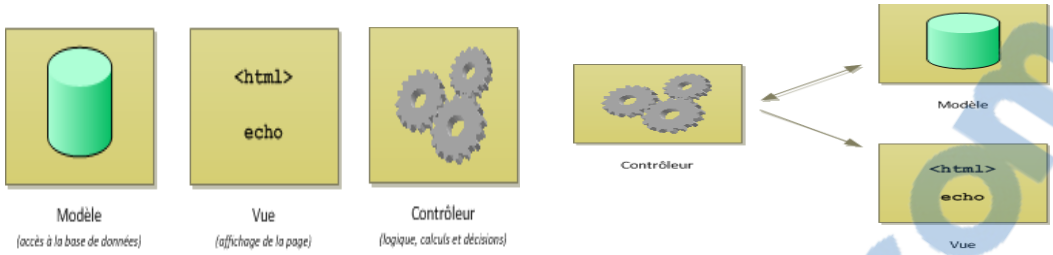

Figure III. 1: « rôle de ces éléments »

Il est important de bien comprendre comment ces éléments s'agencent et communiquent entre eux.

- ✓ ***Choix de l'architecture MVC :***

Nous avons choisi de travailler avec l'architecture MVC, car elle permet de bien séparer la logique de la présentation. La vue n'aura aucune logique d'imbriquer. Aussi, étant donné que tout est très bien séparé, il est très facile d'ajouter et de modifier au code sans que le reste ne s'effondre. C'est un pattern qui se prête très bien au développement.

Figure III. 2: « Le schéma de la structure générique d'une architecture MVC »

2.4.b Solide principe :

En programmation informatique, SOLID est un acronyme représentant cinq principes de bases pour la programmation orientée objet.

Ils sont censés apporter une ligne directrice permettant le développement de logiciel plus fiable et plus robuste.

- ➢ *Responsabilité unique (single responsibility principle)*

Rapport-Gratuit.com

Une classe, une fonction ou une méthode doit avoir une et une seule responsabilité (« une classe ne doit changer que pour une seule raison ») par exemple une class Cahier ne devrai en aucun cas représenter à l'intérieure un livre, avoir une fonction qui lui permet de récupérer une connexion à la base de données.

- *Ouvert/fermé (open/closed principle)*

Une classe doit être ouverte à l'extension, mais fermée à la modification (affirme qu'une classe doit être à la fois ouverte (à l'extension) et fermée (à la modification). C'est le « o » de SOLID. « Ouverte » signifie qu'elle a la capacité d'être étendue. « Fermée » signifie qu'elle ne peut être modifiée que par extension, sans modification de son code source.

L'idée est qu'une fois qu'une classe a été approuvée via des revues de code, des tests unitaires et d'autres procédures de qualification, elle ne doit plus être modifiée mais seulement étendue.

En pratique, le principe ouvert/fermé oblige à faire bon usage de l'abstraction et du polymorphisme.

- *Substitution de Liskov (Liskov substitution principle)*

Une instance de type T doit pouvoir être remplacée par une instance de type G, tel que G sous-type de T, sans que cela ne modifie la cohérence du programme.

- *Ségrégation des interfaces (en) (interface segregation principle)*

Préférer plusieurs interfaces spécifiques pour chaque client plutôt qu'une seule interface générale.

- *Inversion des dépendances (dependency inversion principle)*

Il faut dépendre des abstractions, pas des implémentations, il fait référence à une forme spécifique de découplage des modules logiciels. Il correspond au « D » de l'acronyme SOLID. En suivant ce principe, la relation de dépendance conventionnelle que les modules de haut niveau ont, par rapport aux modules de bas niveau, est inversée dans le but de rendre les premiers indépendants des seconds.

Les deux assertions de ce principe sont :

- Les modules de haut niveau ne doivent pas dépendre des modules de bas niveau. Les deux doivent dépendre d'abstractions.
- Les abstractions ne doivent pas dépendre des détails. Les détails doivent dépendre des abstractions. [12]

2.5 Ajax (Asynchronous JavaScript and XML):

2.5.a Qu'est-ce que l'AJAX ?

AJAX est l'acronyme d'*Asynchronous JavaScript and XML*, ce qui, transcrit en français, signifie « JavaScript et XML asynchrones ».

Derrière ce nom se cache un ensemble de technologies destinées à réaliser de rapides mises à jour du contenu d'une page Web, sans qu'elles nécessitent le moindre rechargement visible par l'utilisateur de la page Web. Les technologies employées sont diverses et dépendent du type de requêtes que l'on souhaite utiliser, mais d'une manière générale le JavaScript est constamment présent.
D'autres langages sont bien entendu pris en compte comme le HTML et le CSS, qui servent à l'affichage, mais ceux-ci ne sont pas inclus dans le processus de communication. Le transfert de données est géré *exclusivement* par le JavaScript, et utilise certaines technologies de formatage de données, comme le XML ou le JSON, mais cela s'arrête là. [13]

L'AJAX est un vaste domaine, dans le sens où les manières de charger un contenu sont nombreuses.

2.5.b Les formats de données :

L'AJAX est donc un ensemble de technologies visant à effectuer des transferts de données. Dans ce cas, il faut savoir *structurer nos données*. Il existe de nombreux formats pour transférer des données, nous allons voir ici les quatre principaux :

- Le format texte est le plus simple, et pour cause : il ne possède aucune structure prédéfinie. Il sert essentiellement à transmettre une phrase à afficher à l'utilisateur, comme un message d'erreur ou autre. Bref, il s'agit d'une chaîne de caractères, rien de plus.

- Le HTML est aussi une manière de transférer facilement des données. Généralement, il a pour but d'acheminer des données qui sont déjà formatées par le serveur puis affichées directement dans la page sans aucun traitement préalable de la part du JavaScript.

- Un autre format de données proche du HTML est le XML, acronyme d'Extensible Mark up Language. Il permet de stocker les données dans un langage de balisage semblable au HTML. Il est très pratique pour stocker de nombreuses données ayant besoin d'être formatées, tout en fournissant un moyen simple d'y accéder.

Le plus courant est le JSON, acronyme de JavaScript Object Notation. Il a pour particularité de segmenter les données dans un objet JavaScript, il s'agit d'une représentation des données sous forme d'objet JavaScript. Il est très avantageux pour de petits transferts de données segmentées et est de plus en plus utilisé dans de très nombreux langages.

2.6 Thymleaf :

Thymleaf est un moteur de Template, sous licence Apache 2.0, écrit en Java pouvant générer du XML/XHTML/HTML5. Thymleaf peut être utilisé dans un environnement web (utilisant l'API Servlet) ou non web. Son but principal est d'être utilisé dans un environnement web pour la génération de vue pour les applications web basées sur le modèle MVC. [14]

2.7 Tomcat :

Tomcat est un serveur HTTP à part entière. De plus, il gère les servlets et les JSP (par un compilateur Jasper compilant les pages JSP pour en faire des servlets). Il a été écrit en langage Java. [15]

Il peut donc s'exécuter via la machine virtuelle Java sur n'importe quel système d'exploitation la supportant.

Caractéristiques :

D'après l'application du projet, les caractéristiques de Thymeleaf sont les suivantes :

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

- C'est un moteur de template écrit en Java traitant les fichiers XML, XHTML et HTML5.
- Thymleaf permet de traiter à la fois les fichiers appartenant à un site web ou non. Il n'y a pas dépendance vis-à-vis de l'API Servlet.

Plusieurs modes de Template sont disponibles :

- XML
- XHTML 1.0 and 1.1
- HTML5

Le support de l'internationalisation des textes.

La mise en œuvre d'un cache performant et configurable permet de réduire les entrées/sorties.

Thymeleaf est extrêmement extensible et peut être utilisé comme Framework de Template.

Une documentation très complète contenant de nombreux exemples est disponible.

2.8 Maven :

Couramment appelé Maven, Apache Maven est un outil de gestion et d'automatisation de production des projets logiciels Java en général et Java EE en particulier. Maven est géré par l'organisation Apache Software Foundation. L'outil était précédemment une branche de l'organisation Jakarta Project.

Maven utilise un paradigme connu sous le nom de Project Object Model (POM) afin de décrire un projet logiciel, ses dépendances avec des modules externes et l'ordre à suivre pour sa production. Il est livré avec un grand nombre de tâches prédéfinies, comme la compilation de code Java ou encore sa modularisation. [16]

2.8.a A quoi sert Maven ?

Maven est un outil permettant d'automatiser la gestion de projets Java.Il offre entre autres les fonctionnalités suivantes :

- Compilation et déploiement des applications Java (JAR, WAR).
- Gestion des librairies requises par l'application.
- Exécution des tests unitaires.
- Génération des documentations du projet (site web, PDF, Latex).
- Intégration dans différents IDE (Eclipse, JBulder)

2.8.b Repository Maven :

Afin de bien gérer les dépendances, maven s'appuie sur des repositories de librairies (jar). Ces repositories peuvent être locaux à la machine ou accessibles via HTTP. Ainsi, à sa première exécution, Maven télécharge les différents plugins dont il a besoin et les installe dans le répertoire maven/repository situé dans le répertoire de travail de l'utilisateur. Ainsi, ces mêmes librairies peuvent être réutilisées entre les différents projets.[17]

2.8.c Projet Object Model(POM) :

Chaque projet ou sous-projet est configuré par un POM qui contient les informations nécessaires à maven pour traiter le projet (nom du projet, numéro de version, dépendances vers d'autres projets, bibliothèques nécessaires à la compilation, noms des contributeurs, etc.).

Ce POM se matérialise par un fichier pom.xml à la racine du projet. Cette approche permet l'héritage des propriétés du projet parent. Si une propriété est redéfinie dans le POM du projet, elle recouvre celle qui est définie dans le projet parent. Ceci introduit le concept de réutilisation de configuration.

Le fichier POM du projet principal est nommé POM parent. Il contient une description détaillée de mon projet, avec en particulier des informations concernant la version nage et la gestion des configurations, les dépendances, les ressources de l'application, les tests, les membres de l'équipe, la structure et bien plus.[18]

✓ ***Démarrage rapide :***

Si nous sommes développeur Java, nous pouvons utiliser start.spring.io pour générer un projet de base, suivre l'exemple de démarrage rapide ci-dessous ou lire le guide de démarrage de la documentation de référence.

```
pom.xml [pfe-scolarite]
Source  Graph  Effective  History
8     <version>0.0.1-SNAPSHOT</version>
9     <packaging>jar</packaging>
10
11    <name>pfe-scolarite</name>
12    <description>Demo project for Spring Boot and MariaDB</description>
13
14    <parent>
15        <groupId>org.springframework.boot</groupId>
16        <artifactId>spring-boot-starter-parent</artifactId>
          <version>2.0.0.M7</version>
18        <relativePath/> <!-- lookup parent from repository -->
19    </parent>
20
21    <properties>
22        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
23        <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
24        <java.version>1.8</java.version>
25    </properties>
26
27    <dependencies>
28        <dependency>
29            <groupId>org.springframework.boot</groupId>
30            <artifactId>spring-boot-starter-mail</artifactId>
              <version>1.5.10.RELEASE</version>
32        </dependency>
```

1

2

```
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
    </plugins>
</build>
```

3

Figure III. 3: «fichier POM.xml»

➢ ***spring-boot-starter-parent***

spring-boot-starter-parent est un projet disponible dans **Spring Boot**. Les bibliothèques tributaires sont déclarées dans **spring-boot-starter-parent**, mon projet en seulement hérite. Nous devons uniquement déclarer **<parent>** dans le fichier **pom.xml** de mon projet.

- ***spring-boot-starter-web***

Les autres **"Starter"** fournissent simplement des dépendances que vous êtes probable d'avoir besoin quand vous développez un type particulier de l'application.

Par exemple, quand nous développons une application web, nous avons besoin d'une dépendance **spring-boot-starter-web**.

- ***spring-boot-maven-plugin***

Spring-boot-maven-plugin est un plugin qui fournit des bibliothèques nécessaires qui soutient mon projet d'être capable d'exécuter directement sans déploiement sur le serveur web. Il aide à créer un fichier jar qui est exécutable. .[18]

2.8.d Git Hub :

Git Hub (exploité sous le nom de Git Hub, Inc.) est un service web d'hébergement et de gestion de développement de logiciels, utilisant le logiciel de gestion de versions Git.

Git Hub propose des comptes professionnels payants, ainsi que des comptes gratuits pour les projets de logiciels libres. Le site assure également un contrôle d'accès et des fonctionnalités destinées à la collaboration comme le suivi des bugs, les demandes de fonctionnalités, la gestion de tâches et un wiki pour chaque projet.

- ✓ ***Fonctionnalités :***

Git Hub est centré vers l'aspect social du développement. En plus d'offrir l'hébergement de projets avec Git, le site offre de nombreuses fonctionnalités habituellement retrouvées sur les réseaux sociaux comme les flux, la possibilité de suivre des personnes ou des projets ainsi que des graphes de réseaux pour les dépôts (en anglais *repository*).

Il offre aussi la possibilité de créer un wiki et une page web pour chaque dépôt. Le site offre aussi un logiciel de suivi de problèmes (de l'anglais *issue tracking system*). Git Hub propose aussi l'intégration d'un grand nombre de services externes, tels que l'intégration continue, la gestion de versions, badges, chat basés sur les projets, etc.… [19]

3 Outils de développement :

Les Outils utilisé sont les suivants :

WampServer :

WampServer est une plateforme de développement Web sur Windows qui vous permet de créer des applications Web dynamiques avec Apache2, PHP, MySQL et MariaDB. WampServer installe automatiquement tout ce dont vous avez besoin pour développer intuitivement des applications Web. Vous pourrez régler votre serveur sans même toucher à ses fichiers de configuration. Mieux encore, WampServer est disponible gratuitement (sous licence GPML) dans les versions 32 et 64 bits. Wampserver n'est pas compatible avec Windows XP, SP3 ou Windows Server 2003. [20]

Maria DB :

Figure III. 4:« WampServer »

Maria DB est un système de gestion de base de données édité sous licence GPL. Il s'agit d'un fork communautaire de MySQL : la gouvernance du projet est assurée par la fondation Maria DB, et sa maintenance par la société Monty Program AB, créateur du projet. Cette gouvernance confère au logiciel l'assurance de rester libre. [21]

Figure III. 5:«Maria DB»

Java :

Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au *SunWorld*.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java. [22]

JavaScript :

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs avec l'utilisation (par exemple) de Node.js. C'est un langage orienté objet à prototype, c'est-à-dire que les bases du langage et ses principales interfaces sont fournies par des objets qui ne sont pas des instances de classes, mais qui sont chacun équipés de constructeurs permettant de créer leurs propriétés, et notamment une propriété de prototypage qui permet d'en créer des objets héritiers personnalisés. En outre, les fonctions sont des objets de première classe. Le langage supporte le paradigme objet, impératif et fonctionnel. [23]

XML :

L'***Extensible Markup Language*** (**XML**), « langage de balisage extensible» est un métalangage informatique de balisage générique qui dérive du SGML. Sa syntaxe est dite « extensible » car elle permet de définir différents espaces de noms, c'est-à-dire des langages avec chacun leur vocabulaire et leur grammaire, comme XHTML, XSLT, RSS, SVG… Elle est reconnaissable par son usage des chevrons (<, >) encadrant les noms des balises. L'objectif initial de XML est de faciliter l'échange automatisé de contenus complexes (arbres, texte riche…) entre systèmes d'informations hétérogènes (interopérabilité) [24]

4 Environnement de développement

4.1 Netbeans :

NetBeans est un environnement de développement intégré (EDI) un environnement libre et facile à utiliser. Il supporte plusieurs langages de programmation à savoir JAVA, C++, PHP, et bien d’autres. Il offre toutes les facilités d'un IDE moderne. Il fourni plusieurs outils tel qu’un éditeur de texte doté d’un pré-compilateur avancé (éditeur en couleurs, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web), un gestionnaire de projets. Ainsi que des outils de débogage et de test des programmes. C’est un outil qui facilite énormément la phase de développement et des testes.

Compilé en Java, NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Développement Kit JDK est requis pour les développements en Java.

NetBeans constitue par ailleurs une plate forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate forme. [25]

La figure 9 présente une capture écran de Netbeans 8.2, utilisé dans l'implémentation.

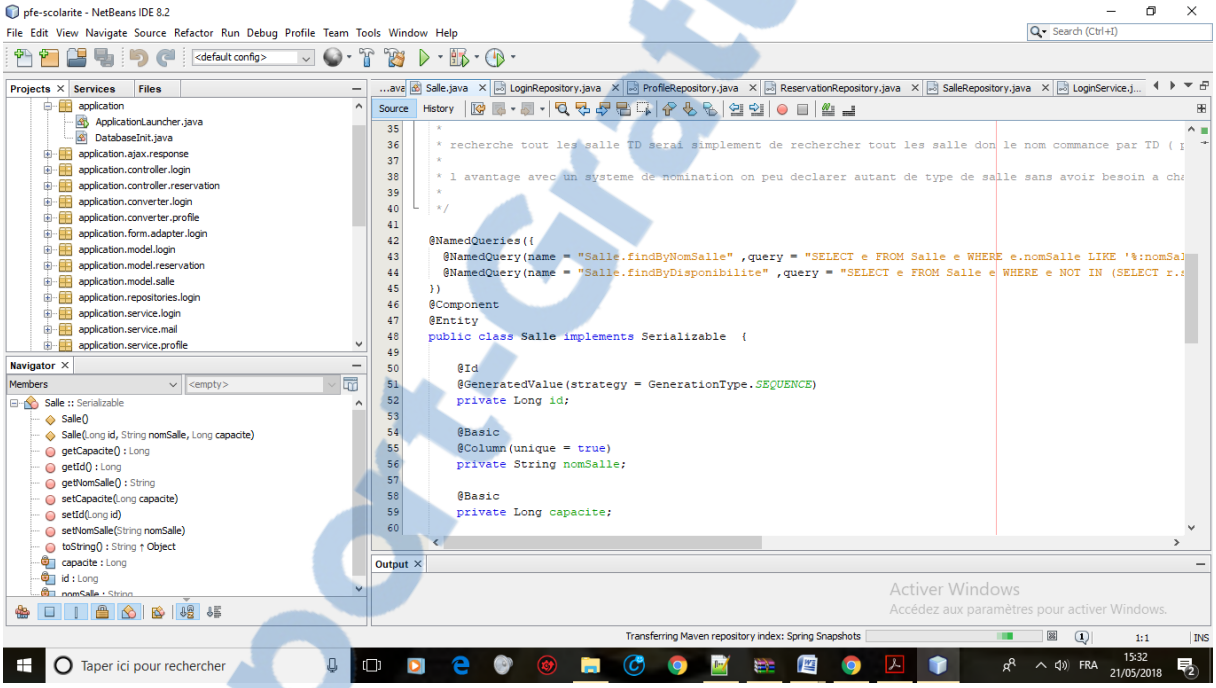

Figure III. 6: « interface de NetBeans IDE 8.2 »

III.5 Description de l'application

III.5.1 Les interfaces de l'application :

Rapport-Gratuit.com

5.1.a Accueil et authentification :

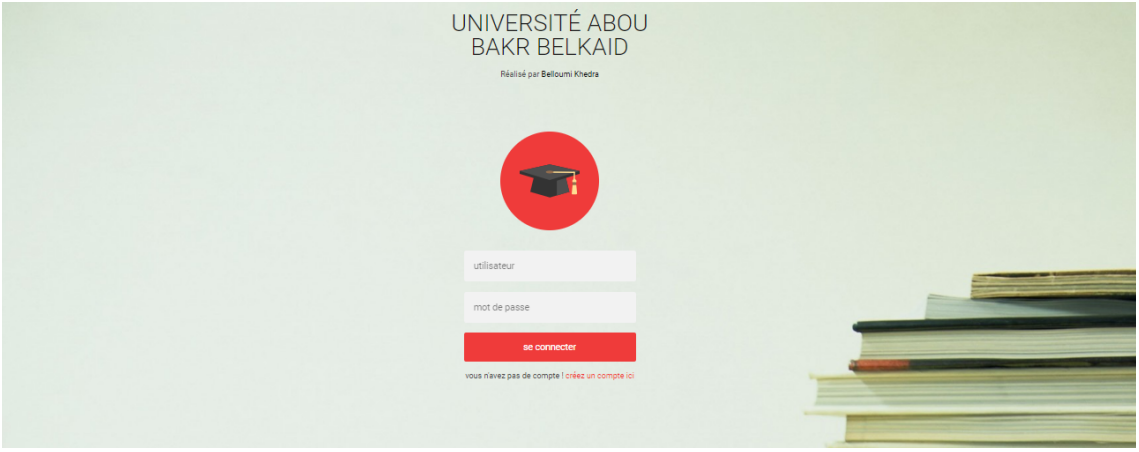

Figure III. 7 :« L'interface authentification»

Dans le cas où l'administrateur fait l'authentification, il permet de saisir

- Nom d'utilisateur : admin

Et

- Mot de passe : ******

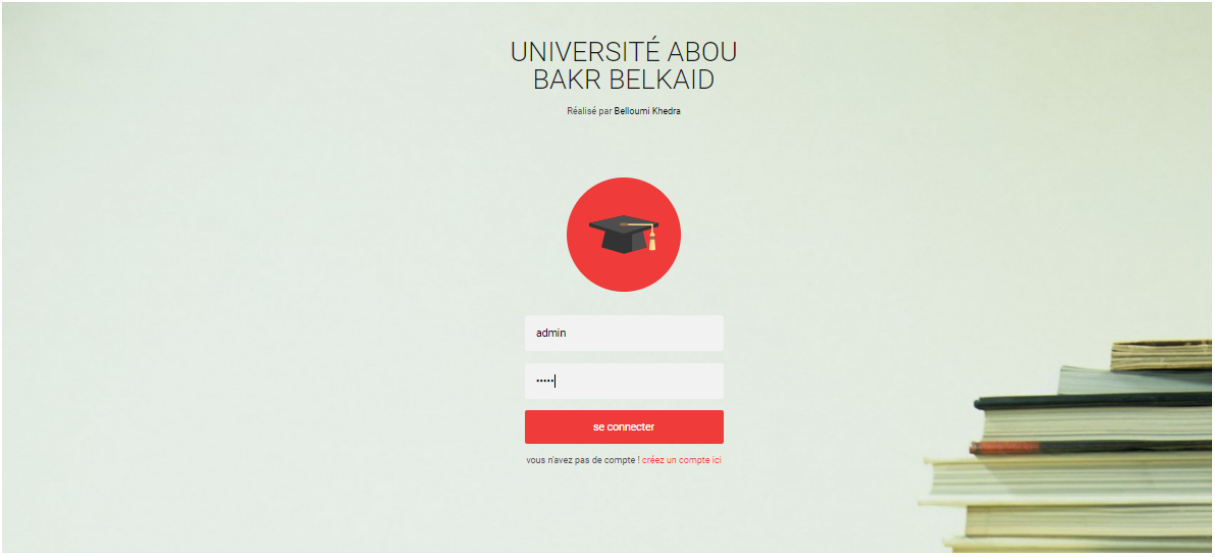

Figure III. 8:« l'authentification de l'administrateur »

S'affiche : l'interface principale qui sera affichée permet d'ajouter une salle, un utilisateur et de faire la réservation comme ci-dessous :

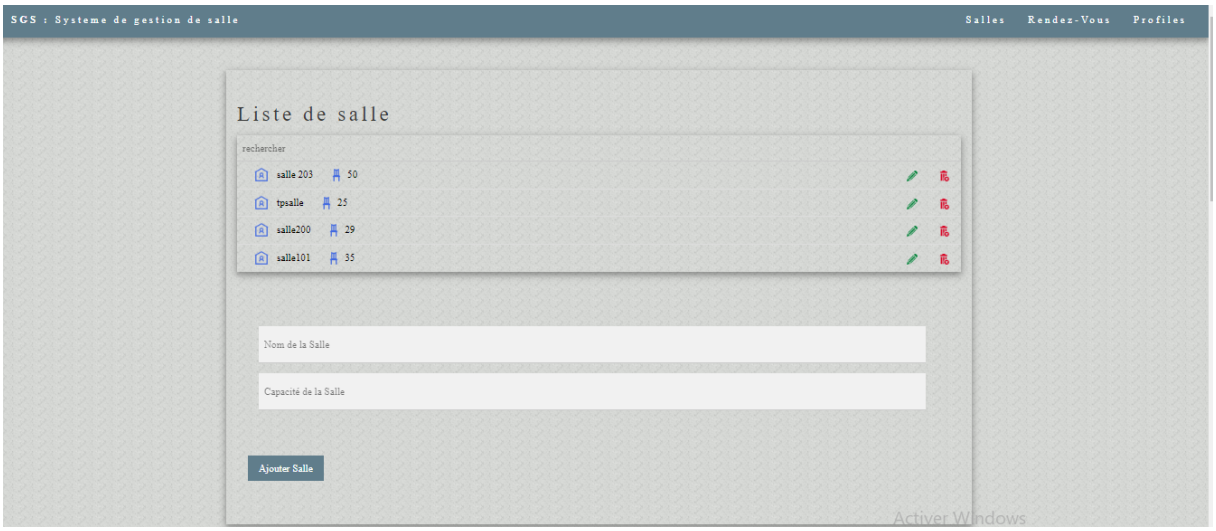

Figure III. 9: « Liste et formulaire des salles»

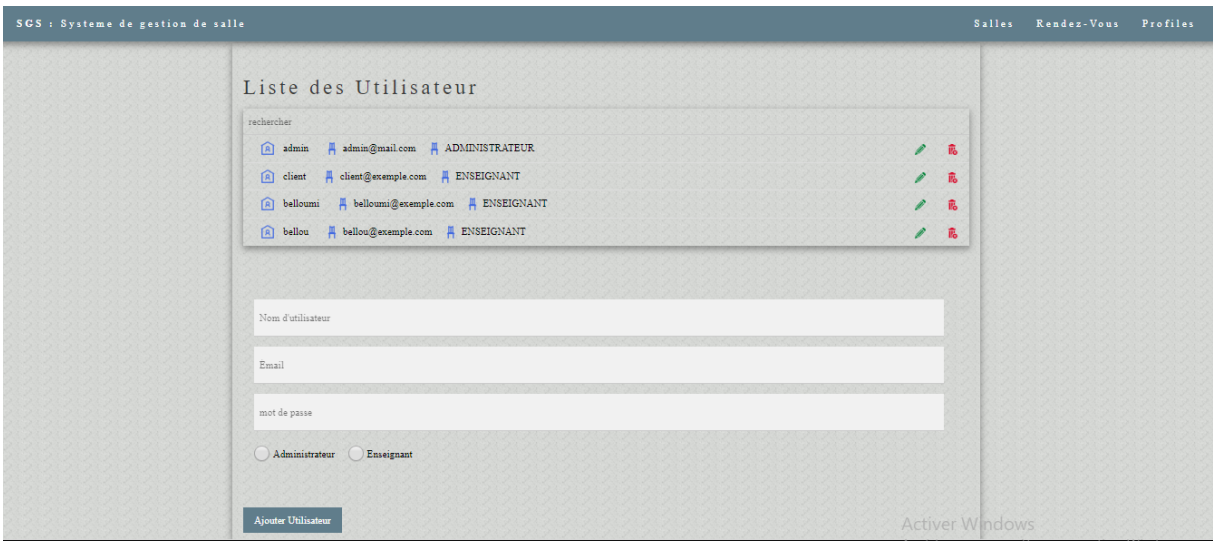

Figure III. 10:« Liste et formulaire des utilisateurs»

5.1.b Salle :

➢ L'ajout d'une salle :

Renseigner les champs « nom de la salle »et « capacité de la salle ».

- Clique sur « Ajouter salle ».

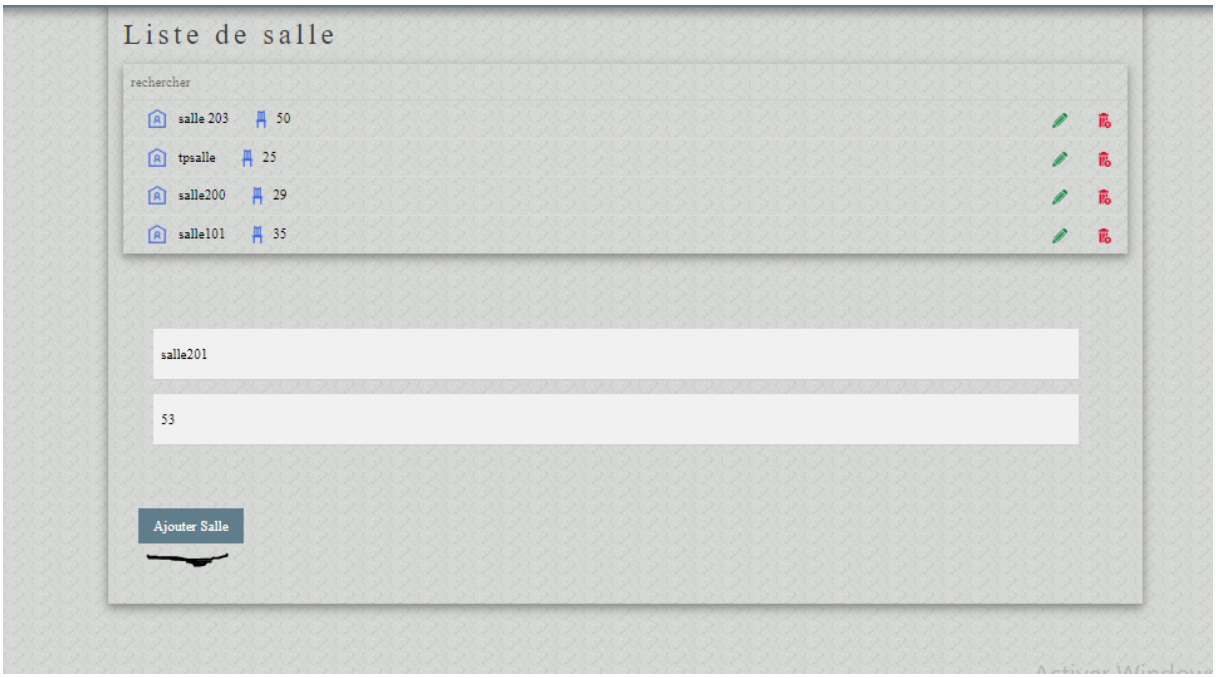

Figure III. 11:« Formulaire d'ajoute des salles»

➢ Modifier les informations de la salle :

- Clique sur le bouton« Modifier ».S'affiche **:**

Figure III. 12:« L'interface de la modification des salles »

➢ Supprimer une salle réservée **:**

- Clique sur le bouton « Supprimer ».

S'affiche **:**

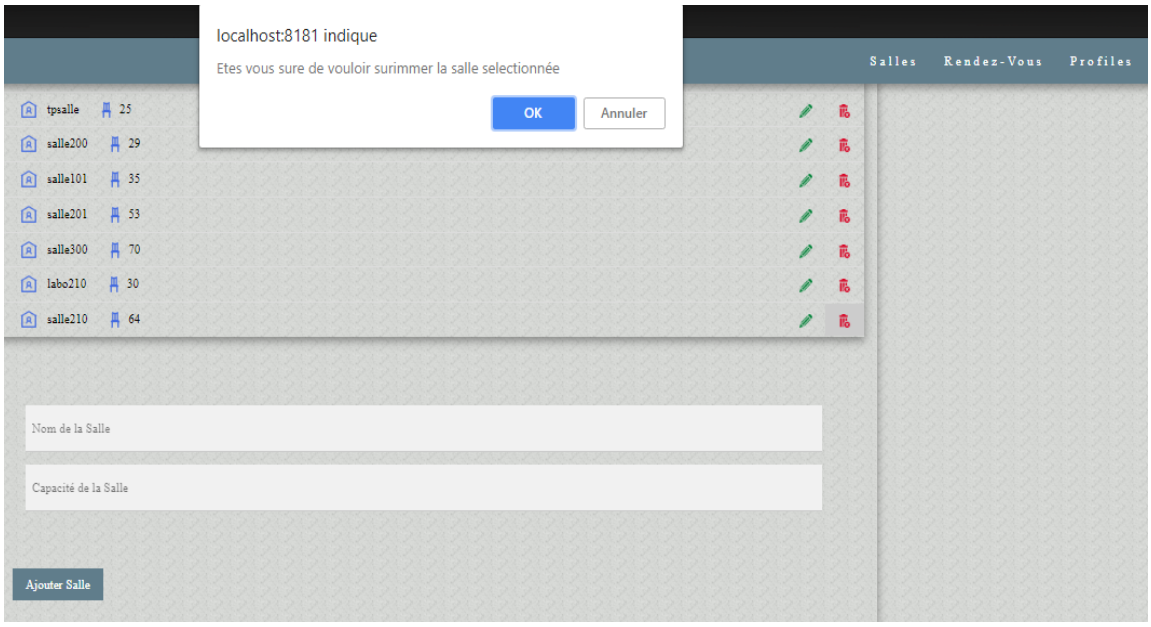

Figure III.11 : « Interface de suppression »

5.1.c Utilisateur:

➢ L'ajout d'un utilisateur :

- Renseigner les champs « nom d'utilisateur»« email» et « mot de passe».
- Clique sur le bouton «Administrateur » ou « Enseignant »
- Clique sur « Ajouter utilisateur ».

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

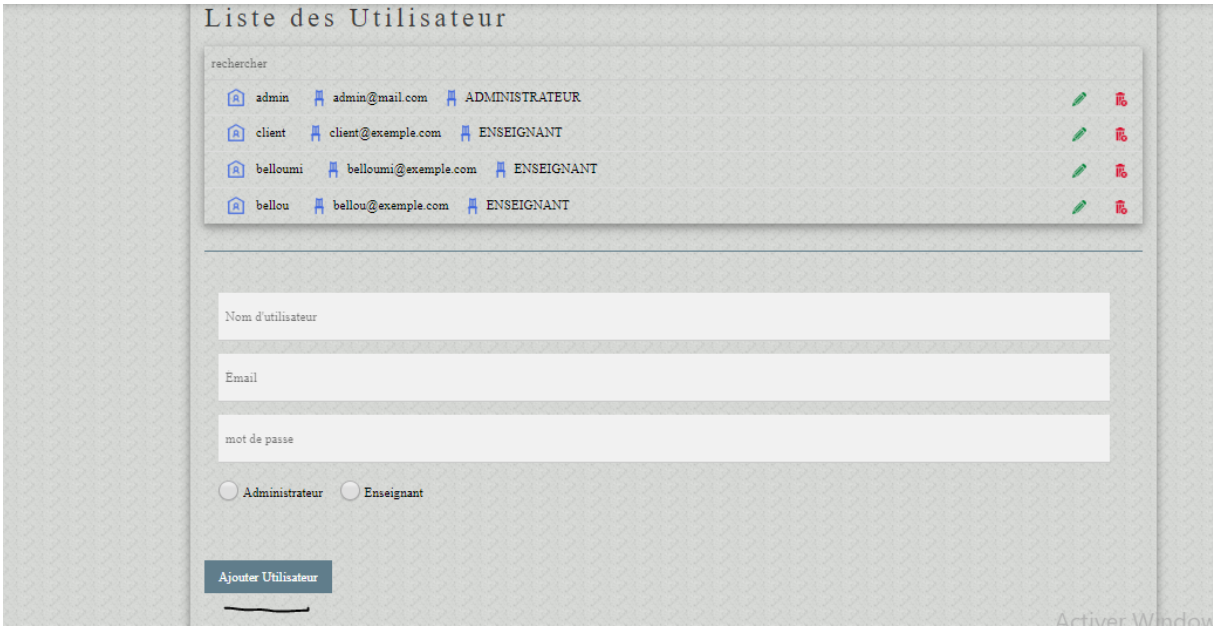

Figure III. 12: «formulaire d'ajout d'un utilisateur »

➢ Modifier les informations de l'utilisateur :

- Clique sur le bouton« Modifier ».

S'affiche :

Figure III. 13:«L'interface de la modification des utilisateurs »

➢ Supprimer un utilisateur **:**

- Clique sur le bouton « Supprimer ».

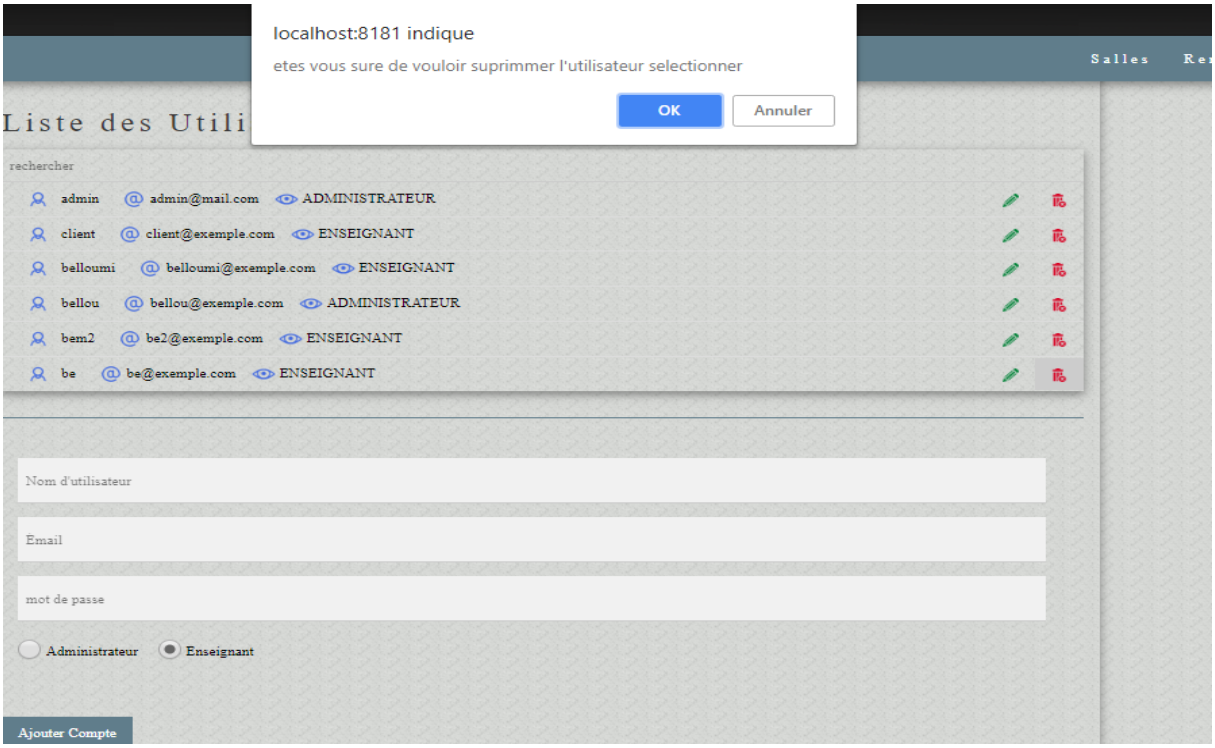

Figure III. 14:«L'interface de la suppression un utilisateur »

5.1.d Reservation:

➢ Saisir « Nom d'utilisateur » et «nom d'une salle» et si on fait une réservation sur un intervalle

S'affiche :

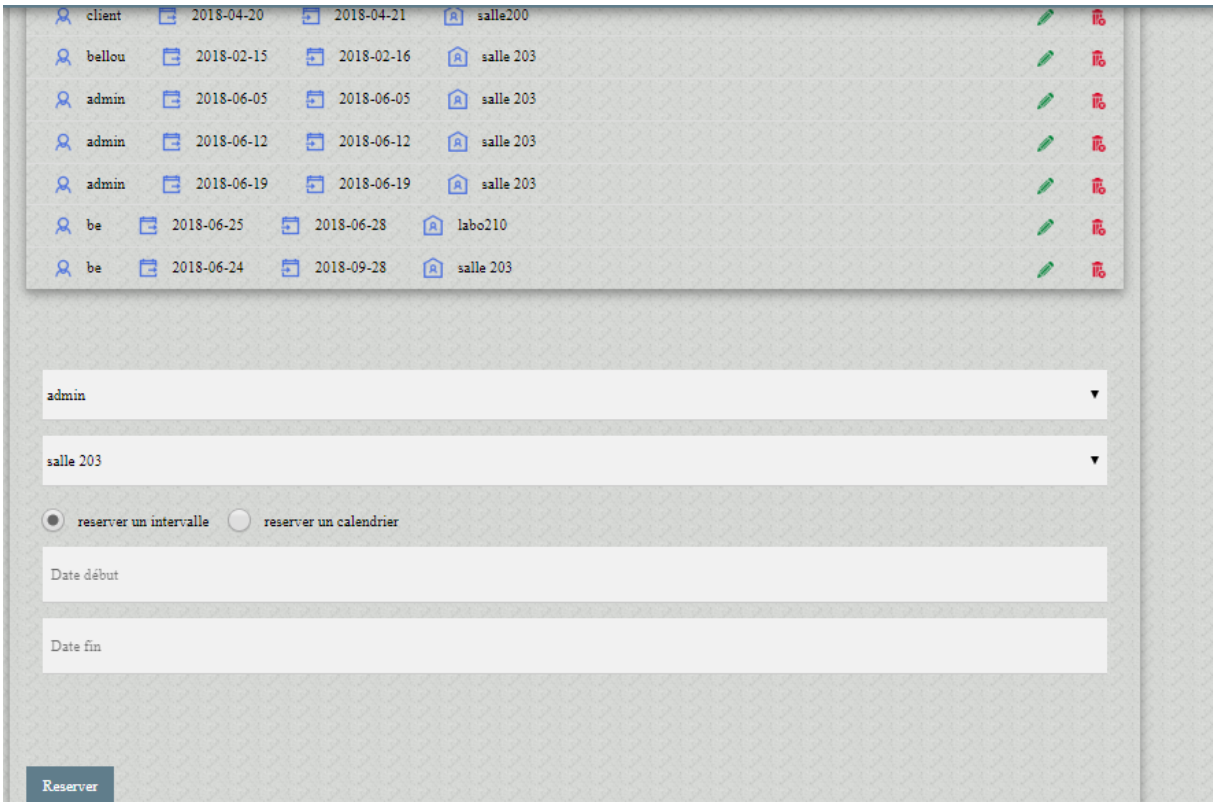

Figure III. 15:« L'interface de la réservation sur l'intervalle»

Si on fait une réservation sur un calendrier s'affiche :

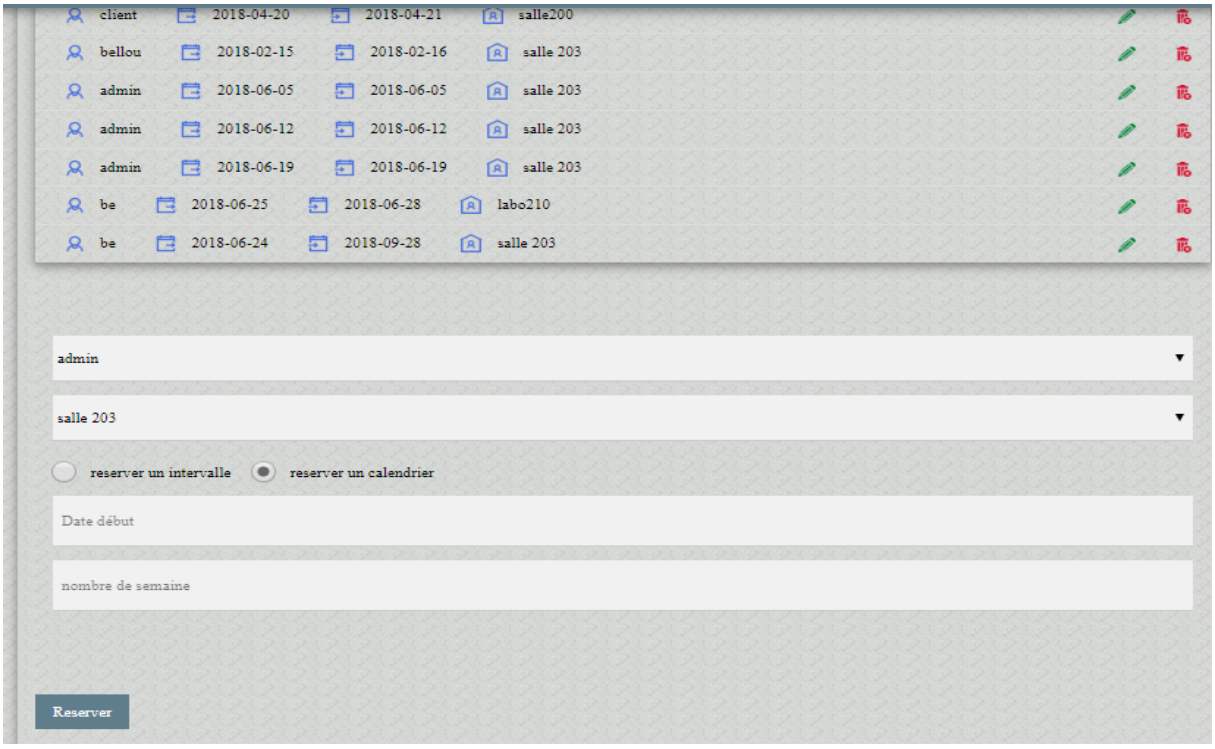

Figure III. 16:« L'interface de la réservation sur calendrier »

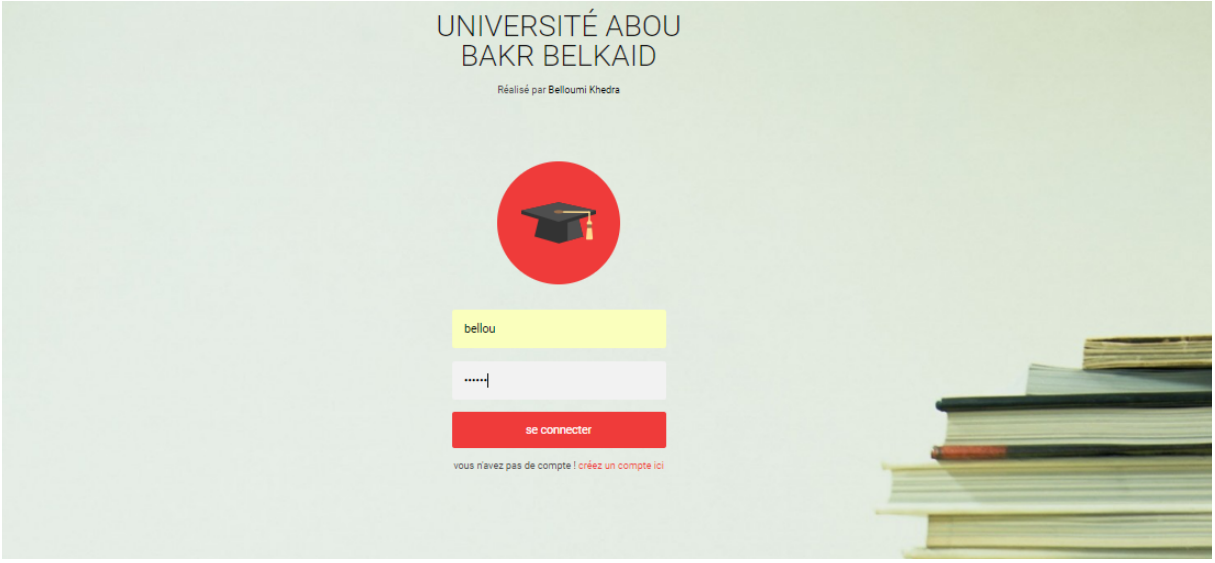

Figure III. 17:« L'authentification de l'utilisateur »

L'interface principale affichée permet de rechercher une salle pour faire la réservation par l'utilisateur et d'envoyer un message à l'administrateur comme sous dessous :

Figure III. 18: « Formulaire de la réservation »

- Réservation :

  - Recherche une salle disponible :
  - Renseigner les champs.
  - Cliquer sur le bouton « Rechercher une salle disponible »

S'affiche

Figure III. 19:« L'interface de réservation »

5.1.e Contact :

Figure III. 20: « Formulaire de contact »

- Renseigner les champs.
- Cliquer sur le bouton « Envoyer un Message » à l'administrateur.
- Afficher une interface pour mentionner que le message est envoyé.

Si un autre utilisateur vient réserver sur l'intervalle, vous n'aurez que des chambres non réservées.

6 Conclusion :

Dans ce chapitre j'ai, au premier lieu, présenté les différents technologies et outils utilisés pour implémenter mon application. Par la suite, j'ai présenté quelques interfaces montrant les différentes étapes de déroulement de l'application.

Rapport-Gratuit.com

CONCLUSION GENERALE & PERSPECTIVES

L'objectif de mon projet était de créer une application web pour administrer les locaux de la faculté, offrant un ensemble de tâches telles que : gestion des salles et des profiles, réservations des salles, plannings, l'accès à la scolarité.

Plusieurs technologies ont été nécessaires pour la réalisation de notre projet, on citera donc le langage JAVA qui permet de développer des applications client-serveur, le langage XML qui permet de décrire des données à l'aide de balises et de règles que l'on peut personnaliser, JAVASCRIPT pour rendre les pages web dynamique, MARIADB est offre des performances optimisées ainsi que des fonctionnalités supplémentaires.

Après le passage par les différentes étapes de développement, l'application a abouti à un logiciel fonctionnel qui répond globalement aux critères imposés dans ce domaine.

Le présent travail nous a permis d'acquérir des connaissances et de maitriser et se perfectionner dans le domaine de la programmation web, et de consoliser nos connaissances en conception logicielle

Comme perspectives à ce travail, nous proposons d’enrichier notre application par l’ajout d’un service collaboratif, ce forum, permettant une interactivité entre les différents acteurs du département.

ANNEXE

Annexe-1 : Composants Spring : annotations

Les premières versions de Spring utilisaient uniquement des configurations par fichier XML (déclaration des Beans et de leur dépendances). Les versions plus récentes de Spring ajoutent de plus en plus de configuration par annotation des classes, jusqu'à faire des systèmes sans configuration xml avec SpringBoot.

Les principales annotations à connaître pour démarrer :

annotations des classes : @Component, @Service, @Repository

annotations d'attributs : @Autowired

annotations de paramètres : @Value

annotations de configuration : @EnableAutoConfiguration, @ComponentScan (dans les dernières versions de Spring, remplace le fichier de configuration xml principal)

Annexe-2 : FORMULAIRE DE DEMANDE DE LOCAUX PAR LES ENSEIGNENT

CADRE A REMPLIR PAR L'ENSEIGNENT

DEMANDE DE MISE A DISPOSITION DE LOCAUX

□ En mon nom propre (indiquer, nom, prénom et numéro d'étudiant) :

Ou

□ Au nom de l'administration:

- Date (s) de la réservation :

- Horaires :

Journée □ 1/ 2 journée □

- Nombre de participants attendus :

t-gratuit.com
NDIAL DU MÉMOIRES

REFERENCES BIBLIOGRAPHIQUES

Débutez l'analyse logicielle avec UML 23/05/2018Le professeur Carina Roels

[1]https://openclassrooms.com/courses/2035826-debutez-lanalyse-logicielle-avec-uml/2035851-uml-c-est-quoi

Modélisation objet et diagrammes UML statiques 2016 by Aurélien Tabard

[2]http://tabard.fr/courses/2016/gl/3_UML-Statique.pdf

Cas d'utilisation Diagrammes de séquence Chantal Reynaud2003-2004

[3]https://ufr-segmi.parisnanterre.fr/servlet/com.univ.collaboratif.utils.LectureFichiergw?ID_FICHIER=1348818741386

Spring(Framework). Posted on **9 octobre 2009** by **Thibaut**

[4] https://fr.wikipedia.org/wiki/Spring_(framework)

Spring Data JPA - Reference Documentation. Oliver GierkeThomas DarimontChristoph StroblMark PaluchJay BryantVersion 2.0.8.RELEASE,2018-06-1

[5] http://projects.spring.io/spring-data/

Spring Grow. 2014 Alexander Perucci

[6] http://www.alexanderperucci.com/portfolio_tags/spring-mvc/

Guide to Spring Mail by baeldung le April 19, 2018

[7] http://www.baeldung.com/spring-email

Avantage spring Thibaut Tropardy 28 août 2008

[8] https://w3blog.fr/2008/08/28/les-avantages-de-spring-mvc/

Technologies supportés Nathalie Bertrand le 2018

[9]http://www.hypershell.com/fr/technologies/expertises-et-solutions/technologies-supportees/

Design Patterns Erich Gamma le 21 octobre 1994

[10] https://www.commentcamarche.com/contents/474-design-patterns

MVC Mathieu Nebra 15 janvier 2018

[11] https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur

Solide principle Robert C. Martin a condensé, en 2002

[12] http://blog.xebia.fr/2011/07/18/les-principes-solid/

Ajax Joe Kellyle 15 juin 2018

[13] https://fr.wikipedia.org/wiki/Ajax_(informatique)

Thymleaf Sébastien NORMAND.le 5 November 2017

[14] https://www.thymeleaf.org/documentation.html

Tomcat Douglas MBIANDOU le 27 Décembre. 2017

[15] https://fr.wikipedia.org/wiki/Apache_Tomcat

Apache Maven Project Gokan EKINCI le 23 janvier 2015

[16] https://maven.apache.org/

Repository Ron Amadeo, *Ars Technica* le 19 Jun 2018

[17] https://www.merriam-webster.com/dictionary/repository

Démarrage rapide Arquillian Guides Dan Allen le 2009-2018

[18] http://arquillian.org/guides/getting_started_fr/

Git Hub présentation Zulie Garnierle 11 janvier 2016

[19] http://simplonline.co/13-ressources/11-github-presentation
Wampserver Christian Pc 2012-2017

[20] https://sourceforge.net/projects/wampserver/

MariaDB Michael Widenius le 8 mai 2018

[21] https://fr.wikipedia.org/wiki/MariaDB

Java James Gosling le novembre 2016

[22] https://fr.wikipedia.org/wiki/Java_(langage)

JavaScript Claude DELANNOY le 28 juin 2018
[23] https://fr.wikipedia.org/wiki/JavaScript

Structurez vos données avec Xml Ludovic Roland 27/03/2018

[24]https://openclassrooms.com/courses/1766341-structurez-vos-donnees-avec-xml/1766421-quest-ce-que-le-xml

Netbeans Henri Laugié le 1 juin 2018

[25]https://fr.wikipedia.org/wiki/NetBeans

Résumé

Compte tenu de la gestion des facultés qui se caractérisent aujourd'hui par leur organisation et leur gestion, ils s'efforcent de les moderniser et de les gérer par la mise en place de programmes informatiques et de mécanismes pratiques.

Le but de ce travail est de faire une conception et la réalisation d'un nouveau système de gestion des locaux pédagogiques permettant de faire des réservations. Notre contribution se manifeste par une application web qui sera dédiée à cette tâche ; plusieurs modules ont été nécessaires à la réalisation. Ce projet de fin d'études nous a permis de dégager plusieurs satisfactions : à commencer par mettre en application nos connaissances théoriques, maitriser plus le domaine de développement et de présenter un outil efficace à notre faculté pour automatiser les réservations et ainsi éviter les chevauchements

Mots clés : réaliser un système de gestion des locaux, réservation des salles, gérer les salles.

Abstract

Given the management of faculties that are now characterized by their organization and management, they strive to modernize and manage them through the implementation of computer programs and practical mechanisms.

The purpose of this work is to design and implement a new management system for educational premises to make reservations. Our contribution is manifested by a web application that will be dedicated to this task; several modules were necessary for the realization. This end-of-studies project has given us several satisfactions: starting with applying our theoretical knowledge, mastering the field of development and presenting an effective tool to our faculty to automate bookings and thus avoid duplication

Key words: to realize a system of management of the premises, reservation of the rooms, to manage the rooms.

ملخص

ونظرا لإدارة الكليات التي تتميز الآن بتنظيمها وإدارتها ، فإنها تسعى إلى تحديثها وإدارتها من خلال تنفيذ برامج الكمبيوتر والآليات العملية.

الغرض من هذا العمل هو تصميم وتنفيذ نظام إدارة جديد للأماكن التعليمية للقيام بالتحفظات. تتجلى مساهمتنا في تطبيق ويب سيتم تخصيصه لهذه المهمة. كانت عدة وحدات ضرورية لتحقيقها. قدم لنا مشروع نهاية الدراسات هذا العديد من الرضا: بدءًا من تطبيق معرفتنا النظرية ، إتقان مجال التطوير وتقديم أداة فعالة لأعضاء هيئة التدريس لأتمتة الحجوزات وبالتالي تجنب الازدواجية.

الكلمات الرئيسية: لتحقيق نظام إدارة المباني ، حجز الغرف ، لإدارة الغرف

.

Rapport-Gratuit.com