

Rapport du Projet de Fin d'Année

Présenté par :

AKHACHKHOUCH Ahmed

LOUKHNATI Mohamed Khalil

Spécialité : Génie Informatique

Thème :

Développement d'une application web pour la gestion des équipements culturels de la commune d'Agadir

Encadré par :

- **Mr. Youness SALMI** : Encadrant à l'Entreprise
- **Mr. Abdelaziz WALID** : Encadrant à l'ENSA

Entreprise : **La commune d'Agadir**

Soutenu le : **15 Juillet 2020**, devant les membres du jury :

Mr. Hicham ELBOUJAOUTI

Mr. Mohamed ELYAAKOUBI

Mr. Abdelaziz WALID

Remerciements :

Avant tout, nous voudrons remercier tous ceux qui nous ont beaucoup appris au cours de ce stage, et qui ont eu la gentillesse de faire de ce stage un moment très profitable.

Nous tenons à remercier également M. Abdelaziz WALID, notre encadrant au sein de l'école, pour sa collaboration, ses conseils et son suivi, son soutien, sa disponibilité et ses précieux conseils durant la période de ce stage.

Nous remercions notre encadrant au sein de la commune, Mr. Youness SALMI, Ingénieur en génie informatique, chef du service informatique, pour son accueil ses encouragements et son aide précieuse.

Nos remerciements vont aussi à tous les membres du jury : M. Mohamed ELYAAKOUBI, M. Hicham ELBOUJAOUI et M. Abdelaziz WALID, qui ont acceptés d'évaluer notre travail, et au corps administratif de l'ENSA pour leurs efforts considérables.

Nos remerciements à tous ceux et celles qui nous ont apporté leur contribution.

Résumé:

Le présent rapport constitue le fruit de notre travail accompli dans le cadre du projet de fin d'année au sein de la commune d'Agadir. Durant ce stage nous avons eu la chance de travailler sur un projet informatique visant, entre autres, à étudier, concevoir, et développer une solution de gestion des équipements culturels de la commune d'agadir.

Ce projet a pour objectif d'optimiser la procédure de gestion des équipements culturels de la commune d'Agadir. Afin d'atteindre cet objectif, nous avons proposé une solution digitalisée basée sur deux modules. Le premier module représente le front-office qui sera destiné aux demandeurs qui souhaitent réserver des équipements culturels pour organiser des événements à caractère culturel. Quant au deuxième module, il offre le service du back-office qui est destiné aux gestionnaires afin de traiter les demandes de réservations soumis par les demandeurs ainsi que gérer les équipements culturels. En outre, il offre aux décideurs la possibilité de faire la supervision des activités grâce à un tableau de bord.

Dans le but de gérer au mieux ce projet, nous avons opté pour l'approche de développement agile (Scrum). Ensuite, planifier les différents sprints nécessaires pour la réalisation de ces modules. En plus, dans l'intention de mener à bien cette mission, on était obligé de commencer par une montée en compétence pour bien maîtriser le métier. Par la suite, un sprint de préparation (sprint 0) a été initié pour effectuer une analyse fonctionnelle globale afin de planifier les différents sprints nécessaires pour la réalisation de la solution. Ensuite, une étude technique a été planifiée et effectuée dans le sprint de préparation avant d'exécuter les sprints qui suivent.

Après avoir terminé l'exécution du sprint de préparation, on a passé à l'exécution des autres sprints suivant l'ordre de leur numérotation. L'exécution de chaque sprint fait l'objet d'une analyse, conception, et réalisation du module. Le développement de chaque module de l'application a été effectué en s'appuyant sur un ensemble d'outils et de technologies comme Angular 9, Spring Boot, Spring Security, JQuery, Bootstrap, etc...

Notre projet, c'est accompli par le développement d'une application web, qui a subi des tests fonctionnels afin d'en assurer le bon fonctionnement. Afin d'étendre les fonctionnalités de notre application, nous proposons comme perspective, un module supplémentaire qui sera consacré à l'évaluation des événements organisés au sein des équipements culturels. Ce dernier sera destiné aux responsables d'équipements culturels pour faire l'évaluation. En plus, il aidera également les gestionnaires et les décideurs à prendre les bonnes décisions.

Mots-clés: Approche de développement Agile, Scrum, Sprint, Angular 9, Spring Boot, Spring Security, JQuery, Bootstrap.

Abstract:

This report is the fruit of our work accomplished as part of the end-of-the-year project at the AGADIR municipality. During this internship, we had the chance to work on a software engineering project aiming, among other things, to study, design, and develop a management solution for equipment demands.

The purpose of this project is to optimize the equipment demand procedure, manage demands as well and manage permissions given by the municipality's services. Therefore, this application offers a solution based on two modules. The first module represents the front-office which will be a destination to the applicants who want to apply in order to reserve a cultural equipment for a purpose of organizing a cultural event. The second module is the back-office that is destined for the manager to manage the applications and also manage the cultural equipment. Furthermore, it offers to the decision maker the possibility to supervise and track what's going on with the app via a dashboard.

In order to better manage this project, we decided to follow the Scrum development method and assign sprints for our modules. In order to carry out this mission, we started by educating ourselves on the priorate subjects related to our project and improve our skillset. A sprint preparation (sprint 0) was initiated to perform a global functional analysis to build a global model on which all future sprints will be based. A technical study was planned in the preparation sprint before working on the following sprints.

After finishing the preparation sprint, we started to buckle down the other sprints in the order of their numbering. Each sprint contained design phase, followed by the conception and then the implementation. The development of each module of the application was carried out relying on a set of tools and technologies as Angular 9, Spring Boot, Spring Security, jQuery, Bootstrap, etc...

Our end product is a web application that had several functional tests to ensure the functionate of the app. To expend the functionalities delivered by our app, we plan to add another module that evaluate the events organized in the cultural equipment of our client, with the emergences of a new actor who is the equipment responsible, he will be the one who evaluate the events, which is going to help tremendously the managers, and decision makers.

Keywords: Agile approach, Scrum, Sprint, Angular 9, Spring Boot, Spring Security, jQuery, Bootstrap.

Table d'abréviation:

Abréviation	Désignation
API	Application Programming Interface
UML	Unified Modeling Language
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
XML	eXtensible Markup Language
JSP	Java Server Pages
DAO	Data Access Object
SGBD	Système de Gestion de Base de Données
J2EE	Java 2 Entreprise Edition
MVC	Model View Controller
IOC	Inversion of Control
GPL	General Public License
POM	Project Object Model
WWW	World Wide Web
URL	Uniform Resource Locator
CMS	Content management system
AJAX	Asynchronous JavaScript And XML
PHP	HyperText Preprocessor
JS	Java Script
SQL	Structured Query Language
REST	Representational State Transfer
IDE	Integrated Developement Environement
TCP	Transmission Control Protocol
IP	Internet Protocol
HTTP	Hypertext Transfer Protocol

Table des matières

Introduction générale :	1
Chapitre I : Contexte général du projet	
1-Présentation de l'organisme d'accueil :	3
1-1 Cadre général du sujet de stage :	3
1-2 La Commune d'Agadir :	3
1-2-1 Situation géographique, démographique et économique de la ville d'Agadir :	3
2-2 Organigramme de la commune d'agadir :	5
2-Presentation du projet :	7
2.1- Etude de l'existant :	7
2.2- Problématique :	8
2.3- Description du projet :	9
2.4- Solution proposée :	9
2.5- Méthodologie de développement :	10
Chapitre II : Analyse et spécification - Sprint de préparation	
1-La méthodologie Scrum :	15
1-1 Définition :	16
1-2 Les rôles Scrum :	16
2-Identification des acteurs :	17
3-Identification des besoins	18
3-1 Les besoins fonctionnels	18
3-2 Les besoins non fonctionnels	18
4-Backlog Scrum :	18
5-Diagramme de cas d'utilisation global :	19
6-Planification du projet :	20
6-1- Planification des Sprints :	21
6-2- Planning prévisionnel d'enchaînement des tâches :	21
Conclusion	22
Chapitre III : Etude Technique	
1-Introduction :	24
2- Architecture de la solution :	24
2-1- Architecture logique :	24
2-2- Architecture applicative :	25
3- Choix des outils technologiques :	26

3-1- Environnement de développement Intégrés :.....	27
3-2- Outils et technologies adoptées :	28
Conclusion	32

Mise en œuvre du Sprint 1 : Front Office

1-Le Backlog du sprint :.....	34
2-Règles de métier :	35
3-Analyse :	35
4-Conception :.....	36
4-1 Diagramme de cas d'utilisation :	36
4-2 Description Textuelle :	37
4-2-1 Cas d'utilisation « Importer calendrier » :.....	37
4-2-2 Cas d'utilisation « Consulter les demandes » :.....	39
4-3- Diagramme de classe :.....	40
4-4 Diagramme de séquence :	40
5- Réalisation :.....	43
Conclusion :	47

mise en œuvre du sprint 2 : Back Office

1- Le Backlog du sprint :.....	49
2- Analyse.....	50
3- Conception :.....	51
3-1 Diagramme de cas d'utilisation :.....	51
3-2 Description Textuelle :	52
3-3- Diagramme de classe :.....	55
3-4- Diagramme de séquence :	55
4- Réalisation :.....	43
Conclusion	65
Conclusion général.....	66
Webographie	67

Liste des figures

FIGURE 1: METHODE CLASSIQUE VS METHODE AGILE	11
FIGURE 2: ITERATION SELON LA METHODE SCRUM	15
FIGURE 3: BACKLOG SCRUM DU SPRINT DE PREPARATION	19
FIGURE 4: DIAGRAMME DE CAS D'UTILISATION DU SPRINT DE PREPARATION	20
FIGURE 5: DIAGRAMME DE GANTT DU PLANNING PREVISIONNEL	22
FIGURE 6: LE MODEL MVC	24
FIGURE 7: ARCHITECTURE MVC	25
FIGURE 8: BACKLOG SCRUM DU SPRINT 1	34
FIGURE 9: MAQUETTE DU SPRINT 1	36
FIGURE 10: DIAGRAMME DE CAS D'UTILISATION DU SPRINT 1	37
FIGURE 11: DIAGRAMME DE CLASSE DU SPRINT 1	40
FIGURE 12: DIAGRAMME DE SEQUENCE DE « REMPLIR UNE DEMANDE »(BOITE NOIRE)	41
FIGURE 13: DIAGRAMME DE SEQUENCE DE « CONSULTER UNE DEMANDE »(BOITE NOIRE)	42
FIGURE 14: DIAGRAMME DE SEQUENCE DE « REMPLIR UNE DEMANDE »(BOITE BLANCHE)	43
FIGURE 15: DIAGRAMME DE SEQUENCE DE « CONSULTER UNE DEMANDE »(BOITE BLANCHE)	44
FIGURE 16: INTERFACE D'INSCRIPTION	45
FIGURE 17: INTERFACE D'AUTHENTIFICATION	45
FIGURE 18: INTERFACE DE CREATION D'UNE DEMANDE	46
FIGURE 19: INTERFACE DES DEMANDES	47
FIGURE 20: INTERFACE D'UNE DEMANDE	48
FIGURE 21: BACKLOG SCRUM DU SPRINT 2 (GESTIONNAIRE)	50
FIGURE 22: BACKLOG SCRUM DU SPRINT 2 (DECIDEUR)	51
FIGURE 23: MAQUETTE DU SPRINT 2	52
FIGURE 24: DIAGRAMME DE CAS D'UTILISATION DU SPRINT 2	53
FIGURE 25: DIAGRAMME DE CLASSE DU SPRINT 2	55
FIGURE 26: DIAGRAMME DE SEQUENCE DE « TRAITER UNE DEMANDE »(BOITE NOIRE)	56
FIGURE 27: DIAGRAMME DE SEQUENCE DE « CONSULTER LES STATISTIQUES »(BOITE NOIRE)	57
FIGURE 28: DIAGRAMME DE SEQUENCE DE « TRAITER UNE DEMANDE »(BOITE BLANCHE)	58

FIGURE 29: DIAGRAMME DE SEQUENCE DE « CONSULTER LES DEMANDEURS »(BOITE BLANCHE).....	59
FIGURE 30: DIAGRAMME DE SEQUENCE DE « CONSULTER LES STATISTIQUES »(BOITE BLANCHE).....	60
FIGURE 31: INTERFACE DES DEMANDEURS	61
FIGURE 32: INTERFACE DES RESERVATIONS.....	61
FIGURE 33: INTERFACE DES EQUIPEMENTS	62
FIGURE 34: INTERFACE DE MODIFICATION D'UN EQUIPEMENT.....	62
FIGURE 35: GRAPHE : DEMANDES.....	64
FIGURE 36: GRAPHE : DEMANDES PAR DEMANDEUR.....	65
FIGURE 37: GRAPHE : LES DEMANDES PAR ETAT D'AVANCEMENT.....	65
FIGURE 38: GRAPHE : AUTORISATIONS PAR GESTIONNAIRE	66
FIGURE 39: GRAPHE : AUTORISATIONS PAR EQUIPEMENT.....	66

Liste des tableaux

TABLEAU 1 : LES COMMUNES LES PLUS PEUPLEES A SOUSS MASSA	4
TABLEAU 2 : L'ORGANOGRAMME DE LA COMMUNE URBAINE D'AGADIR	6
TABLEAU 3 : PLANNIFICATION DES SPRINTS.....	21
TABLEAU 4 : REGLES DU METIER DU SPRINT 1.....	35
TABLEAU 5 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « REMPLIR LA DEMANDE »	38
TABLEAU 6 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « CONSULTER UNE DEMANDE ».....	39
TABLEAU 7 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « TRAITER UNE DEMANDE »	54
TABLEAU 8 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « CONSULTER LES DEMANDEURS ».....	55

Introduction Générale :

L'état a principalement un rôle prépondérant à l'égard de la culture, ce qui signifie l'offre publique en matière culturelle qui est littéralement ancrée dans la création et la construction des équipements. Pour les communes, les équipements contribuent à définir la fonction culturelle des villes, en dressant un panorama des activités culturelles qu'offrent les villes à leurs habitants. Cependant, le défi d'une meilleure et optimale gestion de ces équipements persiste.

Ça nous ramène à ce présent travail dont l'intitulé est : « Développement d'une application Web pour la gestion des équipements culturels de la commune d'Agadir » qui représente l'analyse, la conception et la réalisation d'une solution digitalisée qui permet d'optimiser la gestion des équipements culturels.

Le présent rapport est découpé en cinq chapitres :

- ✓ Chapitre 1 : Contexte général du projet ;
- ✓ Chapitre 2 : Sprint de préparation : Analyse et Spécification ;
- ✓ Chapitre 3 : Etude Technique ;
- ✓ Chapitre 4 : Mise en œuvre du sprint 1 : Front Office ;
- ✓ Chapitre 5 : Mise en œuvre du sprint 2 : Back Office ;
- ✓ Conclusion générale.

Le premier chapitre est un chapitre introductif dans lequel nous effectuons une description succincte de la commune d'Agadir. Ensuite, nous exposons le cadre général du projet qui inclut l'étude de l'existant qui nous a permis de dégager une idée claire sur la procédure actuelle de gestion des équipements culturels, ainsi que la solution proposée pour combler ses lacunes et la méthodologie choisie pour mieux gérer ce projet.

Le deuxième chapitre est dédié à l'analyse des besoins où nous commençons par présenter tout d'abord, l'équipe Scrum ainsi que les rôles de chaque membre. Puis, une description du cycle de vie Scrum. Ensuite, nous identifions les principaux acteurs et les besoins fonctionnels et non fonctionnels. Après une longue discussion avec le client, nous avons pu établir le backlog de produit qui nous a permis par la suite à planifier la réalisation des différents sprints

Le troisième chapitre est dédié à l'étude technique du projet et la mise en place de l'environnement de développement. Au début de ce chapitre, nous commençons par une brève description de l'architecture logique et logicielle de la solution, puis les choix techniques qui inclut principalement l'environnement de développement intégré ainsi que les outils et les technologies adoptés.

Les deux derniers chapitres seront dédiés aux modules de notre application où nous nous intéressons à l'exécution des différents sprints planifiés pour en assurer les réalisations. Finalement, nous clôturons, ce rapport par une conclusion générale dans laquelle nous évaluerons les résultats atteints et nous exposerons les éventuelles perspectives du présent projet.

Chapitre1 : Contexte général du projet

Ce chapitre mis l'accent sur le contexte général du projet. D'abord, une vue globale sera élaborée sur l'historique et les services de l'organisme d'accueil. Par la suite, une description du cadre général du projet par la précision de la problématique et la solution proposée. Pour terminer vers la fin par la présentation de la méthodologie et démarche visée pour gérer au mieux le projet, ainsi que le choix des outils et technologies.

1- Présentation de l'organisme d'accueil :

Dans ce chapitre nous nous intéressons à la présentation de la commune d'Agadir au sein de laquelle nous avons effectué notre stage.

1-1 Cadre général du sujet de stage :

Notre projet intitulé « Développement d'une application Web pour la gestion des équipements culturels de la commune d'Agadir » qui a été réalisé au sein de la division des affaires sociales, culturels et sportifs dans la commune d'agadir et sous supervision du service informatique, Ce thème a été proposé par notre encadrant dans la commune ayant le profil d'ingénieur d'état en génie informatique et qui occupe le poste de chef de service informatique.

1-2 La Commune Urbaine d'Agadir :

La ville d'Agadir est la capitale de la région Souss Massa. Elle est la Cinquième agglomération urbaine au Maroc. Selon le recensement officiel de la population de 2014, cette Commune compte 421 844 habitants avec un taux d'accroissement de la population de 3,7 %.

La Commune Urbaine d'Agadir est limitée à l'Ouest par l'océan atlantique, les communes rurales d'Aourir et Taghazout au Nord, la commune de Drarga à l'Est et au Sud-est et les communes de Dcheira, Inezgane et Ait melloul au sud.

La Commune Urbaine d'Agadir est divisée en quatre annexes (Ex-communes) :

- Agadir centre
- Anza
- Bensergao
- Tikiouine

1-2-1 Situation géographique, démographique et économique de la ville d'Agadir :

a- Situation géographique :

Bordée par l'océan Atlantique à l'ouest et limitée au nord par les derniers reliefs du Haut-Atlas occidental, la ville d'Agadir est située sur un plateau continental étroit qui s'ouvre sur la plaine du Souss Massa.

Le développement de la ville est déterminé par son environnement physique et naturel :

Bloquée au nord et à l'est par la barrière naturelle de la montagne et par le parc national du Souss Massa au sud, l'agglomération se développe aujourd'hui sur la plaine de part et d'autre de l'oued Souss.

Sur le plan géologique, Agadir est située sur « l'accident sud-atlasique » séparant la chaîne du Haut-Atlas de l'Anti-Atlas à l'origine du séisme de 1960 qui détruisit la majeure partie de la ville.

b- Situation démographique :

Les enjeux démographiques dans la capitale du Souss occupent une place essentielle dans l'analyse du processus de développement en cours : Au terme du dernier recensement de 2014, la population légale de la région Souss Massa s'est établie à 2.676.847 habitants (dont 4914 de nationalité étrangère). Elle représente près de 7,9 % de la population totale du Maroc. Cette population s'est accrue de 352 700 habitants au cours de la décennie précédente (2004-2014). Le taux d'accroissement annuel moyen qui s'ensuit est de l'ordre de 1,4 %, dépassant ainsi la moyenne nationale (1,2 %).

A l'échelle des communes urbaines du Grand Agadir (Agadir, Anza, Inezgane, Aït Melloul, Ben Sergao, Dcheïra et Tikiouine), ce taux a été de 5,76 % soit un résultat largement supérieur à la moyenne urbaine nationale (3,6 %). Plusieurs communes proches d'Agadir ont été soumises à une croissance urbaine particulièrement importante.

Aujourd'hui, celle-ci tend à se déplacer dans les communes rurales périphériques de l'agglomération où se développent des centres urbains comme Lqliaa, Dcheira, Drarga.

Nom	Provinces/Préfectures	Population
Agadir (chef-lieu)	Préfecture Agadir Ida-Outanane	421 844 (2014)
Aït Melloul	Préfecture d'Inezgane-Aït Melloul	171 847 (2014)
Inezgane	Préfecture d'Inezgane-Aït Melloul	130 333 (2014)
Dcheira El Jihadia	Préfecture d'Inezgane-Aït Melloul	100 336 (2014)
Oulad Teïma	Province de Taroudant	89 387 (2014)
Lqliâa	Préfecture d'Inezgane-Aït Melloul	83 235 (2014)
Taroudant	Province de Taroudant	80 149 (2014)
Tiznit	Province de Tiznit	74 699 (2014)
Drarga	Préfecture Agadir Ida-Outanane	70 793 (2014)

Tableau 1 : Les communes les plus peuplées à Souss Massa

e- Situation économique :

Pôle de croissance et de développement, Agadir s'affirme aujourd'hui comme une métropole économique incontestable. Ce succès a été récemment mis en évidence dans le cadre des études du SNAT (1) dans lesquelles le Souss Massa et sa capitale ont été donnés comme champion et détenteur du record national des aires de croissance principales.

En s'appuyant sur sa position de pôle régional entre le sud et le nord du Maroc et la diversité de ses fonctions (industrielle, administrative, universitaire), la ville a su capter une part importante de l'attractivité nationale en tirant partie des atouts naturels et économiques dont elle disposait mais aussi du dynamisme de sa population.

La performance économique, de la ville et de la région du Souss Massa depuis les 20 dernières années, la place aujourd'hui comme deuxième pôle de croissance derrière l'aire métropolitaine centrale. Le tourisme, la pêche et l'industrie agro-alimentaire sont les piliers de ce développement économique remarquable.

2-2 Organigramme de la commune d'agadir :

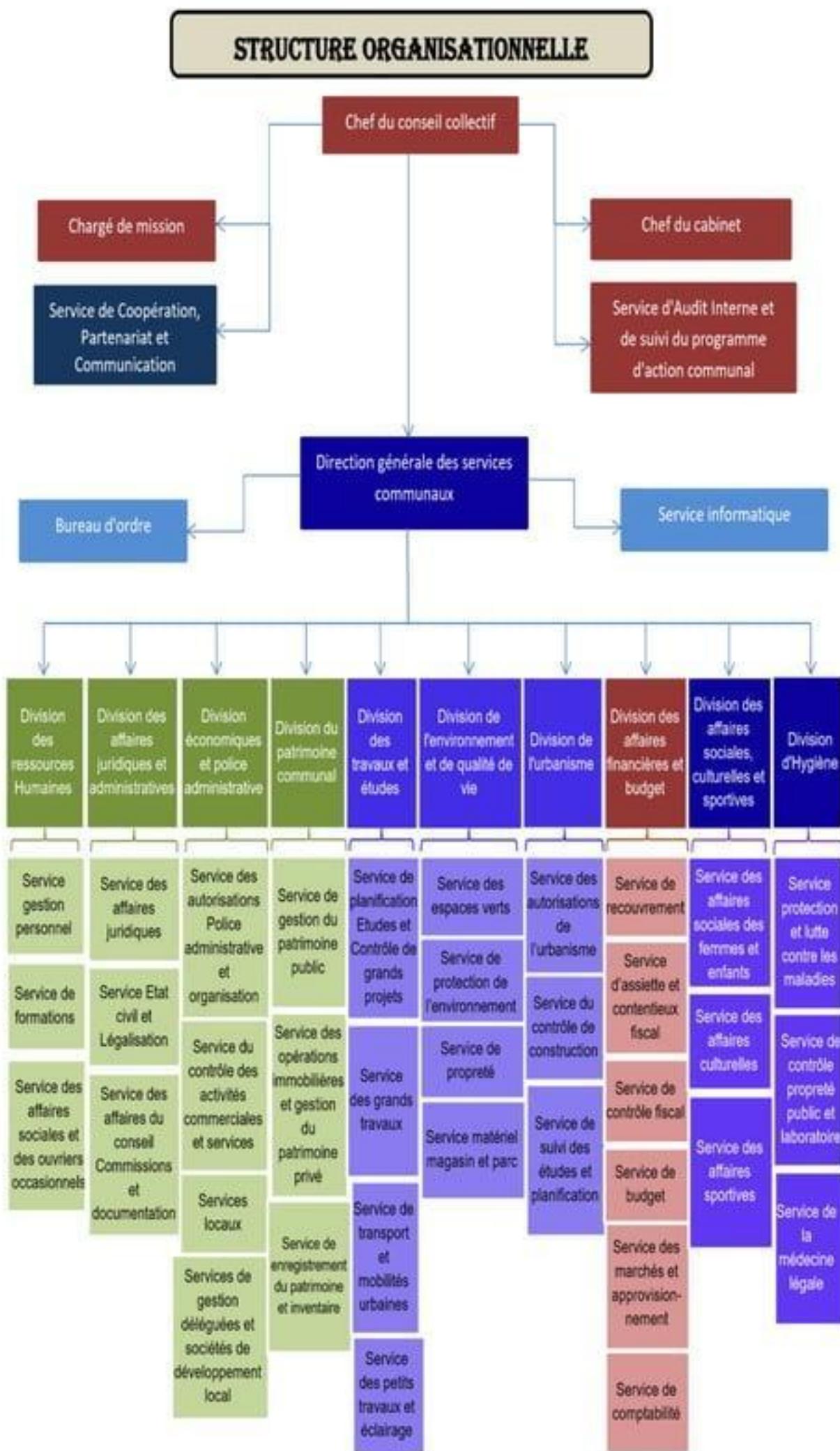


Tableau 2 : L'organigramme de la commune d'Agadir

2- Présentation du projet :

La gestion des réservations des équipements représente une procédure qui prend du temps et d'effort pour toutes les administrations communales marocaines, alors nous présentons dans ce chapitre tout d'abord l'étude de l'existant et après illustrer la problématique pour en déduire enfin des solutions proposées.

2.1- Etude de l'existant :

La division des affaires sociales, culturelles et sportives contient un service des affaires culturelles. Parmi ces tâches il y a la gestion des équipements culturels, qui sont réservés par une personne morale (Association, parti politique, organisation...) ou physique afin de les utiliser pour un événement culturel. Les équipements culturels disponibles actuellement à Agadir sont :

- Complexe Mohamed Khair Eddine.
- Complexe Mohamed Mohamed jamal addorra.
- Complexe Culturel Bensergao.
- Complexe Culturel Tikiouine.
- Complexe Culturel Hadj Habib.
- Complexe Culturel Taddart-Anza.
- Dar Al Hay Haut Anza.
- Dar Al Hay ibn Zeydoun.
- Dar Al Hay Al Fidya.
- Dar Al Hay Essalam.
- Dar Al Hay El Houda.
- Dar Al Hay Zitoune.
- Conservatoire musical Agadir.
- Musée du Patrimoine Amazigh.
- Musée Mémoire d'Agadir.
- Théâtre de Verdure.
- Bibliothèques communales d'agadir.

- Salle de mémoire d'Agadir.
- Centre d'accueil.

En Total, Il y a 6 complexes, 6 Dar Al Hay, 2 Musées, plusieurs bibliothèques, un conservatoire musical, un théâtre, une salle, et un centre d'accueil.

Pour qu'un individu réserve un espace culturel, il doit se déplacer vers le complexe Khair Dine où se trouve le département des ressources humaines de service des affaires culturelles, le gestionnaire reçoit la demande de la réservation d'un équipement culturel avec les pièces jointes nécessaires. Le gestionnaire traite le dossier, par exemple, si l'équipement est déjà réservé à la date demandée. Si c'est accepté le demandeur reçoit un appel informant que la demande de réservation est validée. Ce dernier doit se déplacer vers le complexe Khair Eddine pour récupérer l'autorisation sous format papier qui lui permettra d'utiliser cet équipement pendant une période définie.

Pour tracker l'état d'avancement du dossier de la demande de réservation, le demandeur doit se déplacer vers Khair Dine, et contacter l'un des gestionnaires du service des affaires culturelles.

Les décisions sont prises par les gestionnaires, mais s'il y a une complication, ils contactent le décideur ' « élite communal » afin de valider, ce dernier peut aussi annuler des réservations.

Le processus de gestion des équipements culturels actuels est basé sur le traitement manuel non digitalisé.

Cette gestion est faite quasiment en papier, ou fichier Word. Ces fichiers continuent des informations à propos de la demande de réservation d'un document, ou des autorisations garanties au demandeur.

Les gestionnaires sont installés en (COMPLEXE Khair Eddine « centrale »), ils reçoivent des lettres de demandes, contactent le responsable décideur et aussi les responsables des équipements avant de former une décision à propos de la demande, si c'est accepté, ils forment une autorisation, le demandeur de toute façon est notifié à la fin.

2.2- Problématique :

Selon notre étude du processus actuelle, nous avons identifié que ce processus est long, et fastidieux ce qui influence le rendement et les prestations. Il y a plusieurs contraintes qui gênent et rendent le travail moins efficace et plus épuisant. Parmi ces contraintes, nous pouvons citer :

- La difficulté de gestion d'emploi des équipements.

- L'historique est mal structuré.
- L'interaction entre le demandeur et le gestionnaire n'est pas optimisé, et parfois gênante.
- L'absence de tracker ses dossiers de demandes de réservations.
- Moins de clarté sur l'avancement du travail ou des tâches à faire (qui fait quoi) en gardant un historique structuré.
- Manque d'études et de statistique pour avoir une idée générale sur l'état actuelle et les états précédentes.
- Procédure administratif complexe.
- Trop de paperasse.
- Les demandes de réservations sont déposées sans vérification préalable de la disponibilité des équipements, ce qui engendre un grand nombre de refus.

2.3- Solution proposée :

Après l'étude que nous avons effectuée et selon le cahier de charges, nous opérons pour une application qui informatise la procédure d'une demande, et plus précisément la gestion des demandes effectuant par les demandeurs. Concrètement, nous devions réaliser une plateforme WEB qui permettrait à la division des ressources culturelles et sociales de gérer les réservations des demandeurs d'une manière facile et efficace, superviser, et historier son activité de manière automatique, ainsi que garder une vue d'ensemble sur son activité et certaines statistiques relatives à cette dernière, et ce tout en gardant à l'esprit que l'accès à ladite plateforme devrait être sécurisé par une procédure d'authentification.

2.4- Description du projet :

L'objectif principal du projet est de mettre en place une plateforme numérique pour la gestion des réservations concernant les équipements culturels.

Cette plateforme contiendra plusieurs fonctionnalités relatives à la gestion des équipements, qui peuvent être divisées entre des fonctionnalités pour les demandeurs et d'autres pour les gestionnaires, sans oublier les statistiques qui englobe le déroulement du travail.

2.5- Méthodologie de développement :

Après l'apparition de la méthodologie d'Agile, le choix n'est plus évident. Quelle est la différence entre la méthodologie classique et la méthodologie d'Agile ? et Quelle sera la méthode la plus convenable pour notre projet ?

2.5.1- Méthode Classique :

Depuis toujours, les projets sont gérés avec la méthode dite « classique » qui se caractérise par recueillir les besoins, définir le produit, le développer et le tester avant de le livrer. On parle alors ici d'une approche prédictive « cycle en cascade ».

Le cycle en cascade consiste à prévoir des phases séquentielles où il faut valider l'étape précédente pour passer à la suivante. Le chef de projet doit alors s'engager sur un planning précis de réalisation du projet en prévoyant des jalons de débuts et fins de phases ainsi que les tâches à effectuer.

Il faut tout faire bien du premier coup, dit Access Dev. Du côté des risques du projet, ils sont identifiés tardivement car il faut patienter jusqu'à la fin du développement pour effectuer la phase de test. « Plus le projet avance, plus l'impact des risques augmente : il sera toujours plus difficile et coûteux de revenir en arrière lorsqu'on découvre une anomalie tardivement. »

2.5.2- Méthode Agile :

La méthodologie Agile s'oppose généralement à la méthodologie traditionnelle. Elle se veut plus souple et adaptée, et place les besoins du client au centre des priorités du projet.

Les méthodes agiles utilisent un principe de développement itératif qui consiste à découper le projet en plusieurs étapes qu'on appelle itérations.

Ces itérations ne sont rien d'autre que des mini-projets définis avec le client en détaillant les différentes fonctionnalités qui seront développées en fonction de leur priorité. « Le chef de projet établit alors un macro-planning correspondant aux tâches nécessaires pour le développement de ces fonctionnalités. »

Le but est d'assumer le fait que l'on ne peut pas tout connaître et anticiper quel que soit notre expérience. « On découpe alors le projet en itérations plutôt que de tout prévoir et planifier en sachant que des imprévus arriveront en cours de route. »

2.5.3- Méthodologie classique ou Méthodologie Agile ?

A l'origine, cette approche « Agile » a été créée pour les projets de développement web et informatique. Ce qui est notre cas. Puisque planifier la totalité du projet dans les moindres détails avant de le développer est contre-productif, et ça aide aussi améliorer la relation entre le client et l'équipe, pour s'assurer que nous sommes sur le bon chemin.

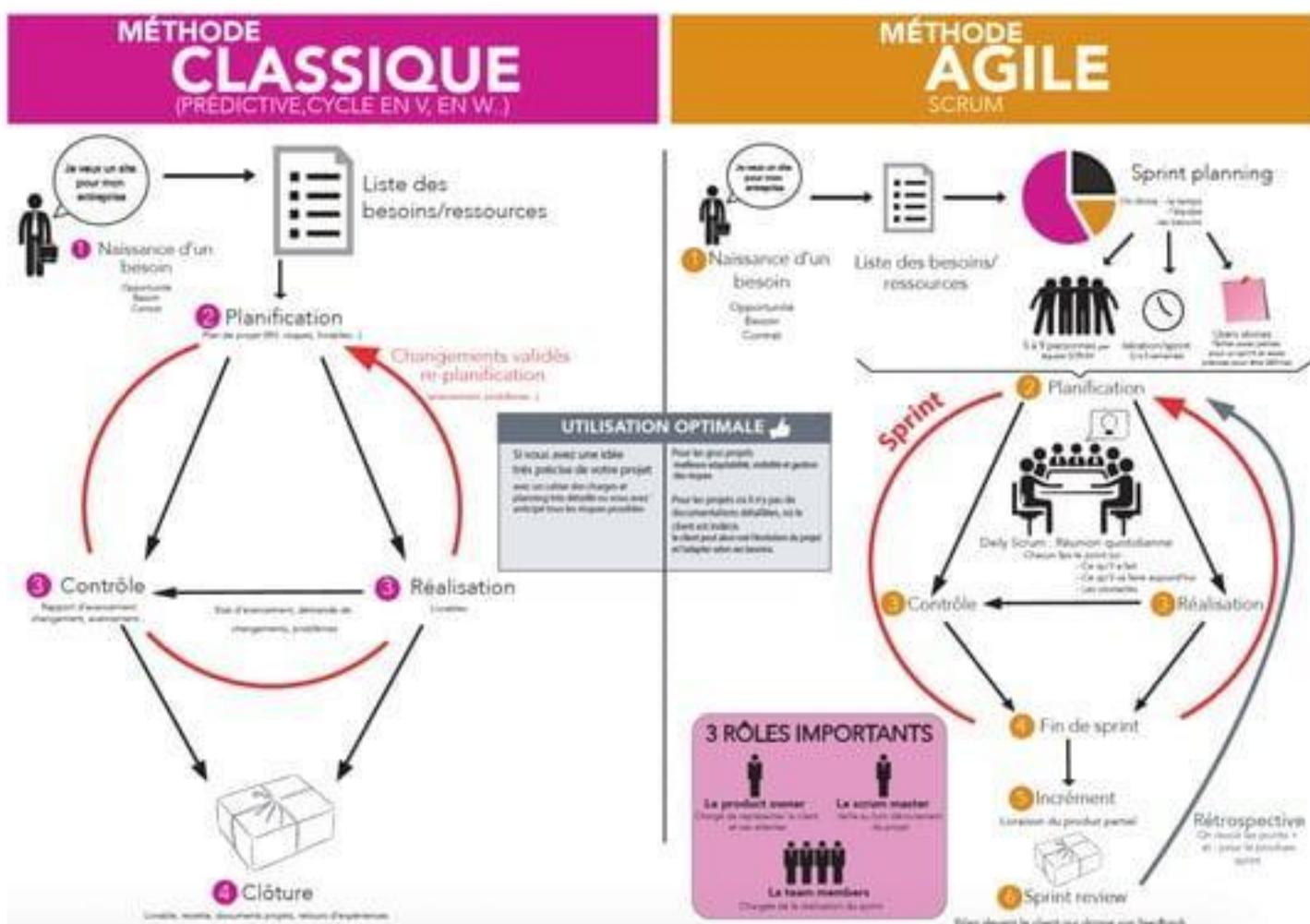


Figure 1 : Méthode classique Vs Méthode Agile

2.5.4- Méthode Scrum :

L'approche Scrum suit les principes de la méthodologie Agile, c'est-à-dire l'implication et la participation active du client tout au long du projet.

Scrum se compose de plusieurs éléments fondamentaux : des rôles, des événements, des artefacts, des règles. Il s'agit d'une approche empirique (c'est-à-dire qui se base sur l'expérience), dynamique et participative de la conduite du projet. Nous avons décidé d'utiliser SCRUM pour les raisons suivantes :

Compréhension du travail et des tâches à accomplir.

Appliquer Scrum, c'est subdiviser votre projet en plusieurs petites parties réalisables. Ça aide en visibilité, et nous obligent à nous demander si toutes les tâches doivent vraiment être effectuer. Nous pouvons ainsi optimiser en continu les étapes qui nous mènent à notre objectif final. Notre projet est composé de plusieurs factures, la meilleure façon de l'entamer est de le subdiviser, puisqu'il n'y a pas une relation rigide entre ces factures.

- **Transparence :**

Chacun des membres de l'équipe doivent savoir ce que les autres accomplissent et le résultat qu'ils peuvent en attendre. Chacun annonce son avancement en une particulière fonctionnalité. Ça aide à visualiser l'avancement du projet, et aussi les problèmes qui doivent être résolu. C'est avantageux pour la majorité des projets.

- **Délais intégrés :**

Chacun sait quand il doit agir et les autres membres de l'équipe savent quand ils peuvent attendre quelque chose de lui. Dans notre cas, les tâches sont connectées, donc savoir le délai des tâches subdivisées est primordial

- **Réactivité :**

La méthode Scrum s'oppose à la boîte noire ou (effet tunnel), ce moment entre la définition du cahier des charges et la livraison du résultat final. Ce qui se passe entre temps, le client n'en a aucune idée, la relation client-prestataire est cloisonnée, verrouillée, chacun à sa place.

Scrum se découpe en plusieurs sprints ; un sprint correspond à un cycle de 1 à 3 semaines. En amont, on définit le back log, c'est-à-dire tous les éléments nécessaires au travail de l'équipe de développement. À la fin du sprint, celle-ci fournit un livrable qui est exploitable immédiatement. Une livraison rapide et régulière permet de voir si le produit correspond aux attentes et d'intégrer rapidement les changements au sprint suivant.

Cela permet un feedback plus intéressant avec notre client La commune. Il n'y a pas plusieurs applications semblables chez notre client, donc nous avons besoins de plusieurs interactions avec les gestionnaires de la commune pour bien comprendre leurs besoins.

- **L'adaptabilité :**

Anticiper et s'entourer des meilleures compétences ne seront pas un rempart contre les changements que connaîtra le produit. L'idée initiale est immanquablement amenée à changer en cours de développement, puis avec les retours des utilisateurs qui vont faire évoluer notre réflexion.

En segmentant la production comme le fait la méthode Scrum, il y a plus de flexibilités, nous pouvons changer les fonctionnalités d'une façon plus pratiques selon les retours des gestionnaires de la commune, puisque nous avons constaté que l'idée n'est pas très claire pour nos clients.

Conclusion :

Tout au long de ce chapitre, nous avons présenté l'organisme d'accueil de la commune d'Agadir et ses principales activités. Par ailleurs, nous avons pu dégager le contexte général du projet et présenter le choix de la méthodologie de développement. Le chapitre suivant sera consacré à l'étude préliminaire.

Chapitre2 : Analyse et spécification – Sprint de préparation

Nous consacrons ce chapitre à l'étude des besoins qui constitue une phase d'analyse du projet. Nous allons présenter tout d'abord, l'équipe SCRUM ainsi que les acteurs principaux de l'application. Puis, nous allons présenter le Backlog de produit et la planification des sprints. Enfin, nous allons clôturer le chapitre par la modélisation du diagramme de cas d'utilisation global du projet ainsi que le diagramme de GANTT prévisionnel.

1- La méthodologie Scrum :

1-1 Définition :

Scrum est la méthodologie suivie par la société Sopra Banking Software, pour la gestion de ses projets. C'est une méthodologie agile itérative basée sur des itérations de courtes durées appelées Sprints. Lorsqu'on dit Scrum, il faut comprendre les mots clés suivants :



Figure 2 : Itération selon la méthode Scrum

Le Scrum Master

- S'assure que les principes et les valeurs de Scrum sont respectés ;
- Facilite la communication au sein de l'équipe ;
- Cherche à améliorer la productivité et le savoir-faire de son équipe.

L'équipe

- Pas de rôle bien déterminé : architecte, développeur, testeur ;
- Tous les membres de l'équipe apportent leur savoir-faire pour accomplir les tâches ;
- Taille de 6 à 10 personnes en général et pouvant aller jusqu'à 200 personnes.

Le Product Owner

- Expert métier, définit les spécifications fonctionnelles ;
- Etablit la priorité des fonctionnalités à développer ou à corriger ;
- Valide les fonctionnalités développées ;
- Joue le rôle du client.
- Les sprints
- Le cycle de vie Scrum est rythmé par des itérations de quelques semaines, les sprints.

Le product backlog

- Le référentiel des exigences initiales est dressé et hiérarchisé avec le client. Il constitue ce que l'on nomme le Product Backlog. Il ne doit pas nécessairement contenir toutes les fonctionnalités attendues dès le début du projet, il va évoluer durant le projet en parallèle avec les besoins du client.
- User Story : Les fonctionnalités décrites portent le nom de User Stories et sont décrites en employant la terminologie utilisée par le client.

Une User Story ou Story contient généralement les informations suivantes :

- ID – un identifiant unique
- Nom – un nom court (entre 2 et 10 mots), descriptif de la fonctionnalité attendue par le client (ex. Export / Import Standard Sales Item). Le nom doit être suffisamment clair pour que les membres de l'équipe et le Product Owner comprennent de quelle fonction il s'agit. Le nom ne doit pas introduire d'ambiguités.
- Importance – un entier qui fixe la priorité des Stories. La priorité d'une story peut être changée en cours de réalisation du projet.
- Estimation – La quantité de travail nécessaire pour développer, tester, et valider cette fonctionnalité. L'unité de mesure peut être un nombre de jours idéaux (jours à 100% dédiés à la fonctionnalité) ou un nombre de points. Les estimations se font en relatif en comparant les estimations des stories terminées avec la story à estimer.
- Démo – Un test relativement simple (ex : exporter un objet en XML puis l'effacer de la base, l'importer depuis le XML, à la fin l'objet doit être dans la base). Ce test constitue un test de validation.
- Notes – toute autre information : clarifications, références documentaires...

Le sprint planning meeting

On organise, avant chaque sprint, une réunion de planification, le sprint planning meeting-. Ce planning sélectionne dans le Product Backlog les exigences les plus prioritaires pour le client. Elles seront développées, testées et livrées au client à la fin du sprint. Elles constituent le sprint Backlog, un sous-ensemble du Product Backlog.

1-2 Les rôles Scrum :

Mon équipe se compose d'un Scrum master (Mohamed Khalil LOUKHNATI), un Product Owner (SALMI Youness) et deux développeurs (Mohamed Khalil LOUKHNATI, Ahmed

AKHACHKHOUCH). Le Product Owner a rédigé les user stories dans le Product Backlog en leur attribuant des priorités en fonction de leurs valeurs métier. L'équipe Scrum se réunit lors de chaque cérémonial de planification de Sprint afin de déterminer les User Stories pouvant être traitées durant le Sprint. Celles ayant été choisies sont ensuite placées dans le Sprint Backlog puis découpées en tâches. Nous avons organisé aussi des réunions quotidiennes de 15 min pour discuter l'avancement de notre travail. À l'issue du Sprint, nous avons possédé un produit potentiellement livrable qui fait l'objet d'une démonstration lors de la revue de Sprint. Le cycle se termine enfin par la rétrospective de Sprint comme illustré dans la figure 2.

2- Identification des acteurs :

Un acteur est une personne ou un autre système informatique qui attend un ou plusieurs services offerts par l'application. Il interagit avec le système par envoi ou réception des messages.

Nous identifions trois acteurs :

Demandeur :

Qui aura comme options.

- Envoyer des demandes.
- Tracker ces demandes.

Agent de division des Affaires Sociales Culturelles et Sportives (gestionnaire) :

Qui aura comme tâches.

- Paramétrage de l'application.
- Gestion des demandes.

Décideur :

Qui aura comme options.

- Consulter les statistiques et l'avancements de chaque demande.

3- Identification des besoins

L'identification des besoins consiste à traduire les objectifs du projet en un ensemble de fonctionnalités ciblées par l'outil à réaliser. Ceci procurera une compréhension plus approfondie des tâches à mettre en œuvre.

Cette plateforme répondra aux deux types de besoins :

3-1 Les besoins fonctionnels

Notre application doit satisfaire les besoins fonctionnels suivants :

- Gestion des demandeurs.
- Gestion des demandes de réservation des équipements culturels déposées par les demandeurs.
- Gestion des équipements culturels de la commune.

3-2 Les besoins non fonctionnels

Ce sont les besoins qui permettraient d'améliorer la qualité des services du site comme la convivialité et l'ergonomie des interfaces et l'amélioration du temps de réponse.

Parmi ces besoins nous pouvons citer :

- **La sécurité** : Il faut avoir une interface d'authentification qui permet à chaque utilisateur de se connecter pour consulter son profil.
- **La convivialité** : Le site doit être facile à utiliser. Il doit présenter un enchaînement logique entre les interfaces et un ensemble de liens suffisants pour assurer une navigation rapide et un texte compréhensible, visible et lisible.
- **La disponibilité** : le site doit toujours être disponible lorsque n'importe quel utilisateur désire consulter le site.

4- Backlog Scrum :

Le backlog Scrum est destiné à recueillir tous les besoins du client que l'équipe projet doit réaliser. Il contient donc la liste des fonctionnalités intervenant dans la constitution d'un produit, ainsi que tous les éléments nécessitant l'intervention de l'équipe projet, qui vont être traiter à fur et à mesure. Ça regroupe tout ce qu'il y a à faire. Le Backlog continent plusieurs User stories.

L'User Story est censée illustrer un besoin fonctionnel exprimé par les types d'utilisateurs. Elle vise ainsi à assurer que l'on délivre bien de la valeur pour nos usagers.

USER STORIES				
ID	Priorité	En tant que	Je veux que	Afin de
U_S1	1	Demandeur	Remplir un formulaire de demande de réservation	Créer une demande de réservation
U_S2	1	Demandeur	Afficher la liste de mes demandes	Tracer l'avancement de chaque demande.
U_S5	1	Agent	Afficher les demandes avec un filtre.	Gérer les demandes
U_S4	1	Agent	Historique des demandes et réservations de chaque demandeur	Avoir une bonne idée sur les précédentes interactions avec un certain demandeur.
U_S5	1	Agent	Accepter ou refuser la demande de réservation	Préparer une autorisation
U_S6	2	Décideur	Afficher les statistiques concernant les demandes et les agents	Tracker l'avancement du projet.

Figure 3 : BackLog Scrum du sprint de préparation

- (1) Priorité élevée.
- (2) Priorité moyenne.
- (3) Priorité basse.

5- Diagramme de cas d'utilisation global :

Un diagramme de cas d'utilisation c'est une représentation simple des interactions de l'utilisateur avec le système qui permet de montrer la relation entre l'utilisateur et les différents cas d'utilisation auxquels il participe. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ce qui rend ce diagramme le meilleur choix pour une bonne communication entre le développeur (concepteur) et le client.

Nous commençons par offrir une vue d'ensemble des différentes fonctionnalités offertes par notre application, à l'aide d'un diagramme de cas d'utilisation global, ensuite nous allons détailler les cas d'utilisation les plus prioritaires. La figure illustre le diagramme de cas d'utilisation général :

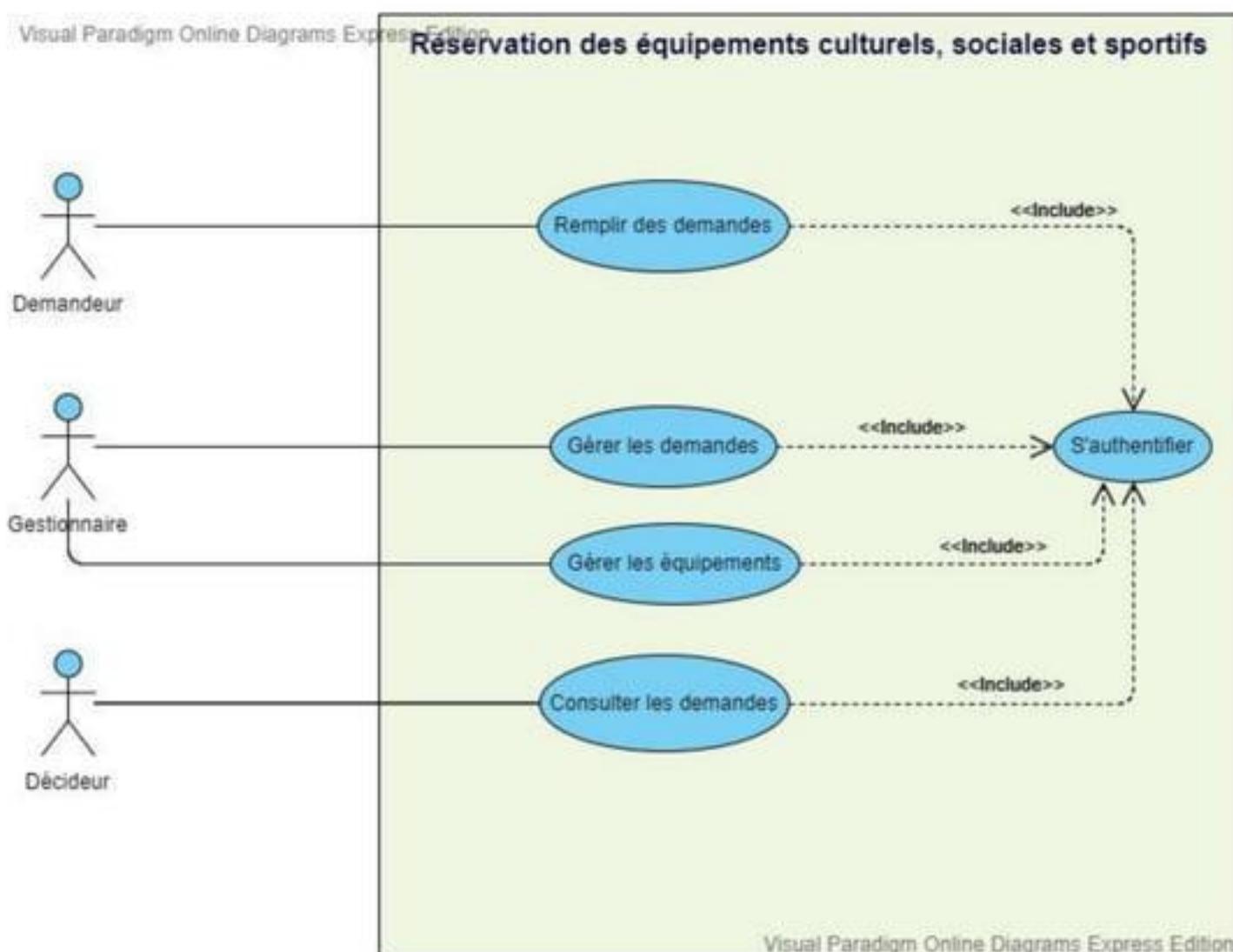


Figure 4 : DIAGRAMME DE CAS D'UTILISATION DU SPRINT DE PREPARATION

L'agent de division des affaires sociales, culturelles et sportives (Gestionnaire) aura accès à toutes les fonctionnalités de l'application. Tandis que le demandeur, il pourra uniquement consulter les demandes effectuées par lui-même et ses autorisations.

Le décideur aura accès à consulter toutes les demandes, les autorisations au cas de validation, ainsi que les statistiques périodiques.

6- Planification du projet :

La clé principale de la réussite d'un projet est un bon planning. En effet, le planning sert à subdiviser le travail et séparer les tâches à réaliser, ainsi que prévoir le déroulement du projet tout au long des phases constituant le cycle de développement.

6-1- Planification des Sprints :

Le tableau suivant présente la feuille de route de projet (Roadmap en SCRUM) qui décrit le déroulement des sprints du projet.

Planification des sprints		
Nom de l'activité	Durée	Date
Sprint 0 : Sprint de préparation	5 semaines	11/03/2020-14/04/2020
Sprint 1 : Sprint de Demandeur	4 semaines	15/04/2020-14/05/2020
Sprint 2 : Sprint de Gestionnaire et Décideur	5 semaines	15/05/2020-24/06/2020

Tableau 3 : PLANNIFICATION DES SPRINTS

6-2- Planning prévisionnel d'enchaînement des tâches :

La figure suivante illustre le diagramme GANTT, l'outil permettant de modéliser la planification des tâches nécessaires à la réalisation de notre projet. Il permet le suivi des délais d'un projet.

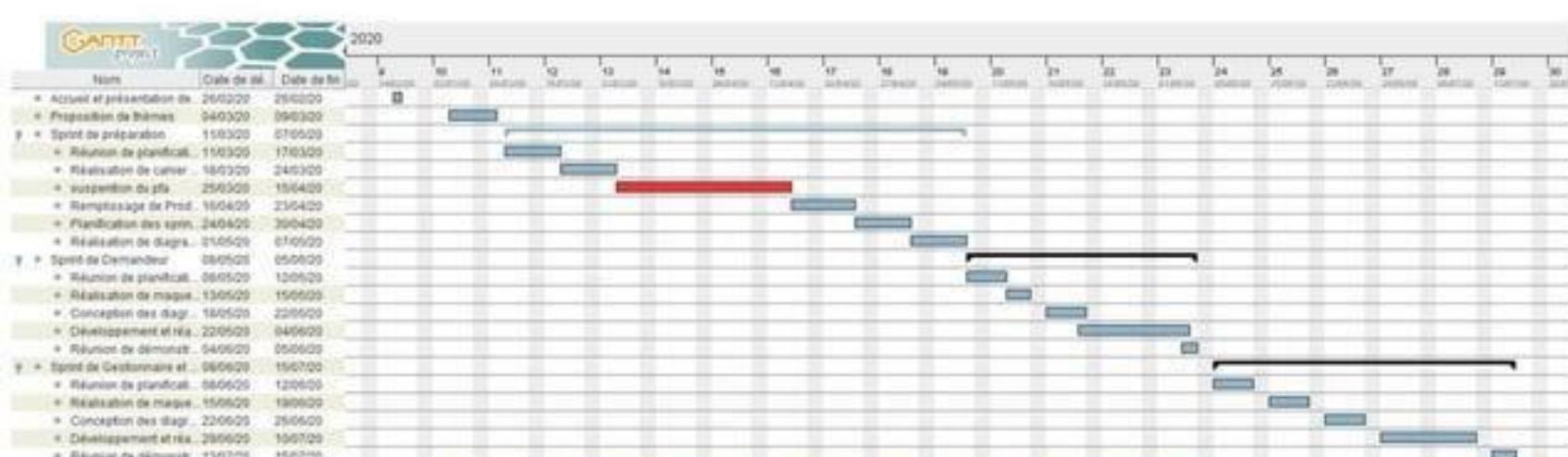


Figure 5 : DIAGRAMME DE GANTT DU PLANNING PREVISIONNEL

(La période en rouge) : la suspension du projet en raison de l'indisponibilité de la division des affaires culturelles, sociales et sportives à cause de la pandémie Covid-19.

Conclusion

Ce chapitre nous a permis de bien délimiter le projet et d'avoir une vision plus claire sur le sujet. Nous avons identifié les acteurs et le Backlog produit. Par la suite, nous avons décrit le cas d'utilisation global. Enfin, nous avons planifié et organisé le temps consacré à la réalisation du projet en identifiant les sprints et le diagramme de GANTT. Dans le chapitre suivant, nous allons entamer la phase d'étude technique qui rentre aussi dans le sprint de préparation.

Chapitre3 : Etude Technique

Nous consacrons ce chapitre à l'initialisation du projet et la mise en place de l'architecture logique et logicielle, l'environnement de développement ainsi que les outils technologies adoptées.

1-Introduction :

Vu que nous allons travailler avec des nouvelles technologies pour la première fois, nous avons consacré la première période à s'auto documenter. Ensuite, nous avons suivi des formations d'initiation sur quelques technologies avant d'entamer le projet, et en parallèle on a des tâches hebdomadaires dans le même sens, pour bien comprendre les exigences du notre projet.

2- Architecture de la solution :

2-1- Architecture logique :

Pour la réalisation de notre projet, nous avons opté à un modèle d'architecture MVC, afin de garantir une assurance de la modularité, séparation des préoccupations et la rapidité de développement. MVC littéralement Modèle Vue Contrôleur est une architecture qui organise l'interface Homme-Machine d'une manière à ce que le développement puisse se faire en couches indépendantes. Il impose la séparation entre les données, la présentation et les traitements ce qui donne trois parties fondamentales dans l'application finale : le modèle de données, le contrôleur et la vue. Pour la conception des interfaces homme-machine, nous avons utilisé Spring MVC 2 qui implémente le modèle MVC, la figure suivante représente les interactions entre les différentes couches.

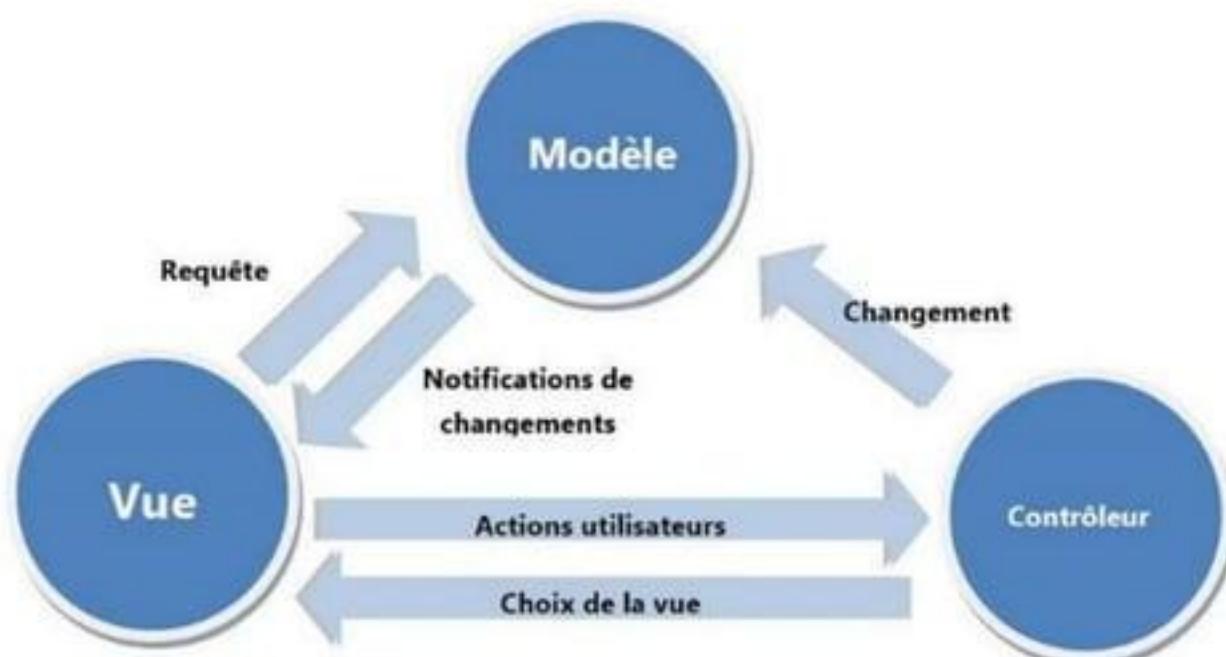


Figure 6 : LE MODEL MVC

2-2- Architecture applicative :

La définition de l'architecture de l'application constitue une étape importante dans le processus de conception. Elle dépend d'un certain nombre de facteurs dont on cite les exigences en matière de performance, les perspectives, d'évolutivité, modularité et extensibilité. La Figure 14 illustre une représentation de l'architecture logicielle de l'application.

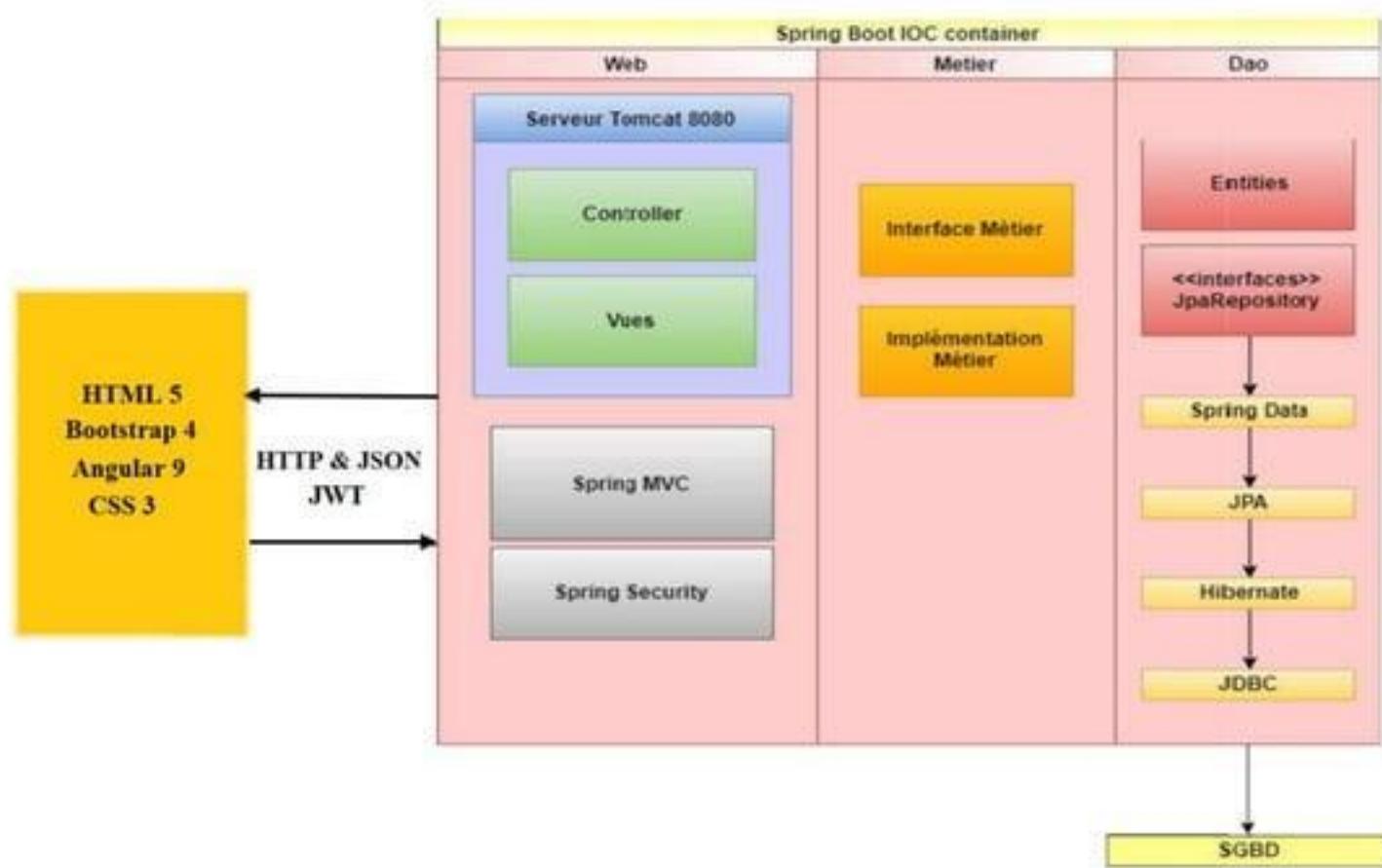


Figure 7 : ARCHITECTURE applicative

Cette architecture favorise une séparation claire des responsabilités, la réutilisabilité et l'évolutivité grâce à l'utilisation du Framework Spring Boot. Afin de bien comprendre cette architecture, nous allons procéder à l'explication de différentes couches :

- Couche présentation :

La couche de présentation assure la logique de navigation. Elle se compose d'une partie frontale en HTML5, Angular9 et Bootstrap pour la représentation graphique des informations et une partie en backend avec un contrôleur frontal de type RestController qui assure l'échange des données en format JSON. Aucun traitement métier n'est à implémenter dans cette couche. Le Framework Spring implémente le patron de conception MVC qui divise la couche présentation en trois parties de la façon suivante :

➤ Le Model :

Représente les données manipulées ainsi que les méthodes pour les récupérer. En effet, il contient le logique métier de l'application.

➤ La Vue :

Correspond à l'interface graphique avec laquelle l'utilisateur interagit et qui permet d'afficher des éléments à l'écran. La vue n'effectue aucun traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle et envoyés par le contrôleur. Dans notre application, elle est implémentée par des pages Html utilisant la technologie Angular.

➤ Le Contrôleur :

Permet de gérer les événements de synchronisation pour faire des mises à jour au niveau du modèle ou de la vue.

- Couche Métier :

C'est dans cette couche qu'on doit retrouver tous les traitements métiers de l'application. Ces traitements seront implémentés sous forme de services Spring. Spring est un conteneur léger permettant de simplifier l'intégration des différentes couches via l'utilisation des interfaces JAVA. En plus, Spring est totalement portable sur tous les serveurs d'application et totalement intégrables avec toutes les technologies choisies pour le développement de ce projet.

- Couche Persistance :

C'est la couche qui permet de sauvegarder et faire appel aux données persistantes de l'application. Elle est implémentée via des services d'accès aux données qui feront appel à SGBD.

3- Choix des outils technologiques :

Avant d'entamer le premier sprint afin d'implémenter la première partie de notre application, une étude technologique a été tout d'abord réalisée. Dans le cadre de cette étude, les outils et technologies qui sont proposés, sont comme suit :

3-1- Environnement de développement Intégrés :

- **STS :**

Spring Tool Suite est un environnement de développement basé sur Eclipse qui est personnalisé pour développer des applications Spring. Il fournit un environnement prêt à utiliser pour implémenter, déboguer, exécuter et déployer des applications Spring, y compris les intégrations pour Git, Maven, AspectJ...



- **Visual studio Code :**

Un éditeur de code open-source, gratuit et multi-plateforme (Windows, Mac et Linux), développé par Microsoft, l'IDE propriétaire de Microsoft. Principalement conçu pour le développement d'application avec JavaScript, TypeScript et Node.js, l'éditeur peut s'adapter à d'autres types de langages grâce à un système d'extension bien fourni.



- **Eclipse :**

Eclipse IDE est un environnement de développement intégré libre (le terme Eclipse désigne également le projet correspondant, lancé par IBM) extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions.



3-2- Outils et technologies adoptées :

Node.js

Une plateforme de développement JavaScript. Ce n'est pas un serveur, ce n'est pas un Framework, c'est juste le langage Javascript avec des bibliothèques permettant de réaliser des actions comme écrire sur la sortie standard, ouvrir/fermer des connections réseau ou encore créer un fichier.



Spring Boot

Un module récent de l'écosystème Spring. Il permet de simplifier le développement en Spring (configuration Maven, programme principal, etc...).



Hibernate

Hibernate est un Framework open source gérant la persistance des objets en base de données relationnelle. Hibernate est adaptable en termes d'architecture, il peut donc être utilisé aussi bien dans un développement client lourd, que dans un environnement web léger de type Apache Tomcat ou dans un environnement Java EE complet : WebSphere, JBoss Application Server et Oracle WebLogic Server



Maven

Apache Maven est un outil pour la gestion et l'automatisation de production des projets logiciels Java en général et Java EE en particulier. L'objectif recherché est comparable au système make sous Unix : produire un logiciel à partir de ses sources, en optimisant les tâches réalisées à cette fin et en garantissant le bon ordre de fabrication. Il est semblable à l'outil Ant, mais fournit des moyens de configuration plus simples, eux aussi basés sur le format XML. Maven est géré par l'organisation Apache Software Foundation. Précédemment Maven était une branche de l'organisation Jakarta Project.



MySQL

MySQL est un SGBDR. Il est distribué sous une double licence GPL et propriétaire. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde, autant par le grand public (applications web principalement) que par des professionnels, en concurrence avec Oracle, Informix et Microsoft SQL Server



Spring Security

Spring Security est un Framework de sécurité léger qui fournit une authentification et un support d'autorisation afin de sécuriser les applications Spring. Il est livré avec des implémentations d'algorithmes de sécurité populaires.

Grâce à spring-security-core nous avons pu mettre en place la sécurité sur l'application web avec une liste de droits d'accès différents. Grâce à ces rôles, le serveur est capable d'interpréter, en fonction de l'utilisateur connecté, les rôles qui lui ont été affectés et ainsi lui permettre de visualiser et avec interagir uniquement certaines parties de l'application



PowerDesigner

PowerDesigner (anciennement PowerAMC) est un logiciel de conception créé par la société SAP, qui permet de modéliser les traitements informatiques et leurs bases de données associées.



CSS

CSS

Les feuilles de style en cascade, généralement appelées CSS, forment un langage informatique qui décrit la présentation des documents HTML et XML. Les standards définissant CSS sont publiés par le W3C. Introduit au milieu des années 1990, CSS devient couramment utilisé dans la conception de sites web et bien pris en charge par les navigateurs web dans les années 2000.



Bootstrap



Bootstrap est une collection d'outils utile à la création du design (graphisme, animation et interactions avec la page dans le navigateur etc...) de sites et d'applications web. C'est un ensemble qui contient des codes HTML et CSS, des formulaires, boutons, outils de navigation et autres éléments interactifs, ainsi que des extensions JavaScript en option. C'est l'un des projets les plus populaires sur la plate-forme de gestion de développement GitHub.

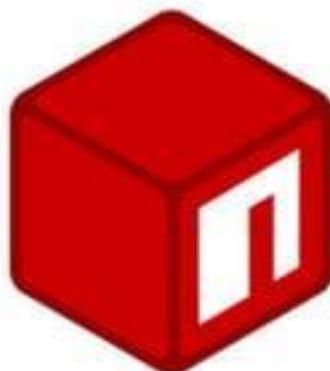
Angular 9



Angular9 est destiné à simplifier la création d'applications Web progressives (PWA), étant donné qu'à ce jour, la prise en charge de ces applications dans le Framework nécessite beaucoup d'expertise de la part des développeurs. Rappelons que le concept d'applications Web progressives, fruit d'une collaboration entre Google et Mozilla, vise à permettre le développement d'applications Web offrant sur un mobile une expérience de qualité supérieure, à peu près la même qu'une application native.

Npm

npm est le gestionnaire de paquets officiel de Node.js. Depuis la version 0.6.3 de Node.js, npm fait partie de l'environnement et est donc automatiquement installé par défaut. npm fonctionne avec un terminal et gère les dépendances pour une application. Il permet également d'installer des applications Node.js disponibles sur le dépôt npm. NPM est un outil indispensable lorsqu'il s'agit de travailler avec NodeJS, effectivement, le noyau de ce dernier est très petit et se contente de fournir les fonctionnalités les plus basiques de la plateforme. Tout le reste, et tout ce qui fait la force de NodeJS, est fournie grâce au gestionnaire de paquets npm. En général npm installe, puis fait appel à d'autres gestionnaires de paquets qui feront le travail à sa place et collecteront puis installeront les différents paquets.



JQuery

JQuery est un framework JavaScript sous licence libre qui permet de faciliter des fonctionnalités communes de Javascript. Depuis sa création en 2006 et notamment à cause de la complexification croissante des interfaces Web, jQuery a connu un large succès auprès des développeurs Web et son apprentissage est aujourd'hui un des fondamentaux de la formation aux technologies Web. Il est à l'heure actuelle la librairie front-end la plus utilisée au monde.



GitHub

GitHub est un service en ligne qui permet d'héberger les repositories de code. GitHub est un outil gratuit pour héberger du code open source, et propose également des plans payants pour les projets de code privés. C'est le numéro 1 mondial et il héberge plus d'une dizaine de millions de repositories.



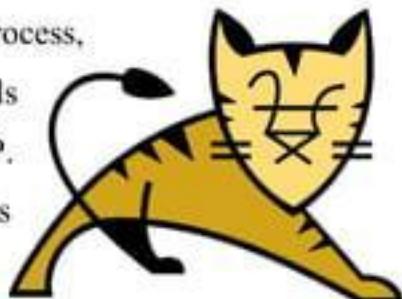
Jackson

On a choisi JSON comme type de données transmises entre le serveur et l'application web, donc nous avons besoin d'un Framework qui peut convertir les objets java en objets JSON lorsque l'envoie des données de l'applications vers le serveur et vice versa lorsque la réception des données. Pour cela nous avons utilisé Jackson pour faire cette conversion à travers des annotations spécifiques.



Tomcat

Apache Tomcat est un conteneur web libre de servlets et JSP Java EE. Issu du projet Jakarta, c'est un des nombreux projets de l'Apache Software Foundation.²³ Il implémente les spécifications des servlets et des JSP du Java Community Process, est paramétrable par des fichiers XML et des propriétés, et inclut des outils pour la configuration et la gestion. Il comporte également un serveur HTTP. 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). Tomcat a été écrit en langage Java. Il peut donc s'exécuter via la machine virtuelle Java sur n'importe quel système d'exploitation la supportant.



Conclusion

Au cours de ce chapitre nous avons présenté l'architecture logique, logicielle de notre application, l'environnement de travail. Dans le chapitre suivant, nous allons illustrer la réalisation du premier module et nous allons entamer la conception du premier sprint.

Chapitre3 : Mise en œuvre du sprint 1 : Front Office

Ce chapitre fait l'objet d'une présentation du premier release du projet qui est la procédure à suivre pour réserver un équipement. L'étude de chaque sprint couvre l'analyse, la conception, la réalisation et les tests fonctionnels.

1- Le Backlog du sprint :

ID du scénario utilisateur	Pour qu'on puisse	Les tâches	Estimation par jours
SC_1	Gérer les demandes effectuées par les demandeurs	Développer l'interface de créer un compte Demandeur	6
		Développer l'interface d'authentification	5
		Développer l'interface contenant les équipements	En cliquant sur un équipement dans la liste des équipements, ses informations globales s'affichent
		Développer l'interface de créer une demande de réservation	Consulter La disponibilité de chaque équipement
		Développer l'interface pour visualiser les demandes des réservations	L'historique de toutes les réservations demandées
		Développer l'interface du profile du demandeur	L'état de traitement de chaque demande

Figure 8 : BACKLOG SCRUM DU SPRINT 1

2- Règles de métier :

Elément concerné	Règle
Etat de la demande	<p>Chaque demande est caractérisée par un statut qui décrit l'état d'avancement de la demande :</p> <ul style="list-style-type: none"> • Non-traitée. • En train d'être traitée. • Acceptée. • Rejetée.

Tableau 4 : REGLES DU METIER DU SPRINT 1

3- Analyse :

Afin de concrétiser les exigences demandées, nous avons réalisé des maquettes, pour qu'on puisse avoir une idée plus précise sur l'ergonomie du module.

Bonjour Demandeur
A

Liste des demandes				
Id	Titre	Date de demande	Date de Reservation	Etat
1	Demande 1	20/04/2020	20/05/2020	Initial
2	Demande 2	21/04/2020	21/05/2020	Validée
3	Demande 3	20/04/2020	29/04/2020	Rejetée
4	Demande 4	23/04/2020	20/05/2020	Initial
5	Demande 5	29/04/2020	04/05/2020	Initial
6	Demande 6	01/04/2020	20/04/2020	Validée
7	Demande 7	20/04/2020	15/05/2020	Initial

1 / 2 Visual Paradigm Online Diagrams Express Edition

Figure 9 : MAQUETTE DU SPRINT 1

4- Conception :

4-1 Diagramme de cas d'utilisation de Sprint1 :

Afin de remplir une demande avec succès, il faut d'abord importer les fichiers nécessaires

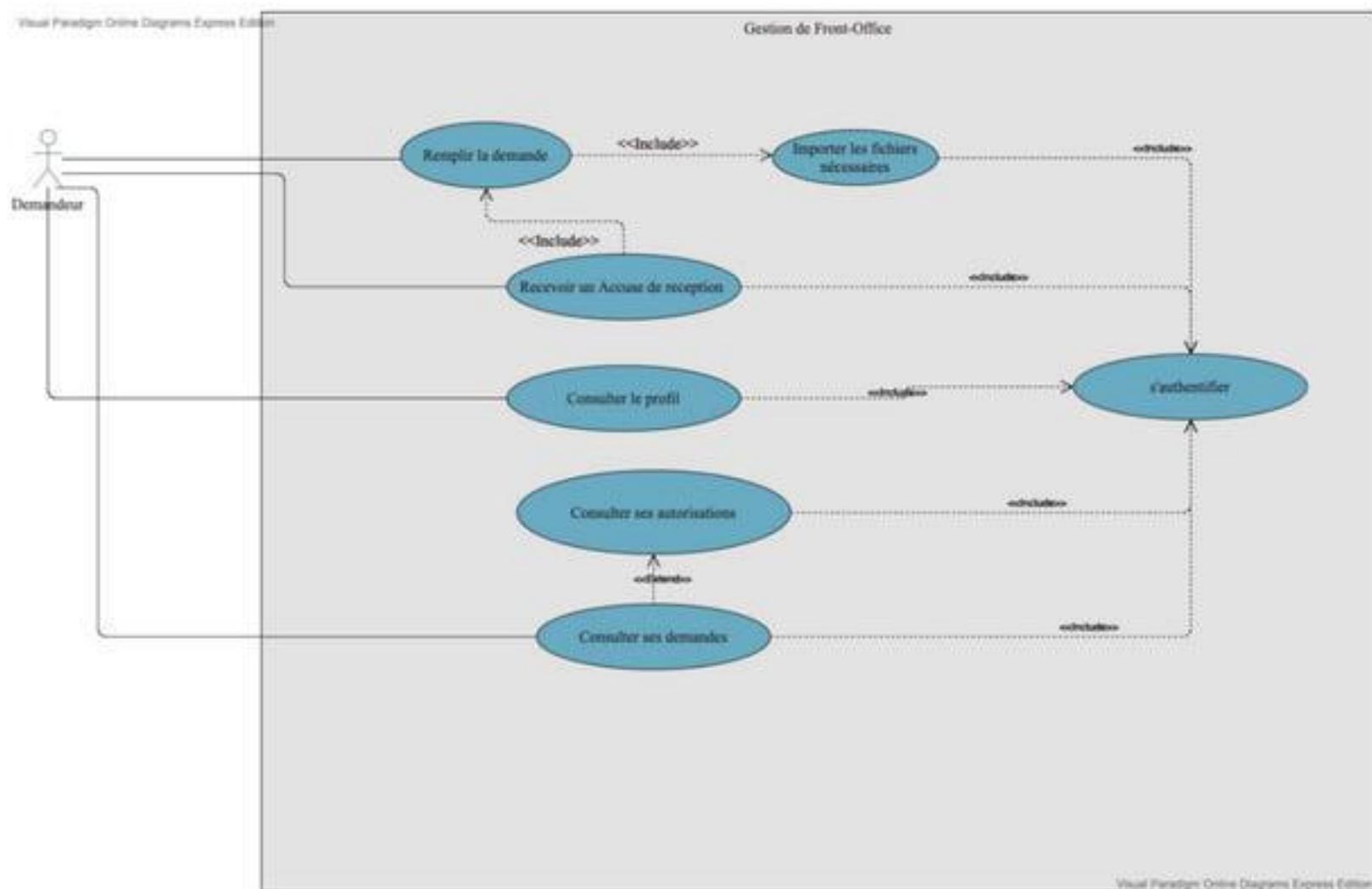


Figure 10 : DIAGRAMME DE CAS D'UTILISATION DU SPRINT 1

4-2 Description Textuelle :

4-2-1 Cas d'utilisation « Remplir une demande » :

UC_001- Remplir la demande	
Type	Core Use Case
Description brève	<ul style="list-style-type: none"> • Ce cas d'utilisation décrit les données nécessaires pour déposer une demande de réservation
Acteur principal	<ul style="list-style-type: none"> • Demandeur
Pré conditions spécifiques	<ul style="list-style-type: none"> • L'utilisateur doit être authentifié et son rôle soit « Demandeur »
Scénario nominal :	
<ol style="list-style-type: none"> 1. L'utilisateur sélectionne « Créer une demande » dans le « navbar ». 2. Le système présente un formulaire à remplir. 3. L'utilisateur remplit les données nécessaires. 4. L'utilisateur click sur le Button « Submit ». 5. Le système affiche un message signalant le succès de l'opération. 	
Post-conditions spécifiques	<p>Les données d'une demande sont importées :</p> <ul style="list-style-type: none"> ✓ Sauvegarde des données dans la base de données.
<p>Le système peut rencontrer les exceptions ci-dessous :</p> <ul style="list-style-type: none"> ➤ L'utilisateur ne remplit pas un champ obligatoire. 	

Tableau 5 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « REMPLIR LA DEMANDE »

4-2-2 Cas d'utilisation « Consulter les demandes » :

UC_002- Consulter les demandes	
Type	Core Use Case
Description brève	<ul style="list-style-type: none"> Ce cas d'utilisation décrit étapes pour consulter les informations d'une demande
Acteur principal	<ul style="list-style-type: none"> Demandeur
Pré-conditions spécifiques	<ul style="list-style-type: none"> L'utilisateur doit être authentifié et son rôle soit « Demandeur »
Scénario nominal : <ol style="list-style-type: none"> L'utilisateur sélectionne « Mes demandes » dans le « navbar ». Le système présente la liste de toutes les demandes effectuées par cet utilisateur avec l'état de chaque demande. L'utilisateur click sur le Button « Voir détail » au droit d'une demande. Le système affiche toutes les informations contenues dans la demande. 	
Le système peut rencontrer les exceptions ci-dessous : <ul style="list-style-type: none"> ➤ L'utilisateur ne remplit pas un champ obligatoire. 	

Tableau 6 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « CONSULTER UNE DEMANDE »

4-3- Diagramme de classe :

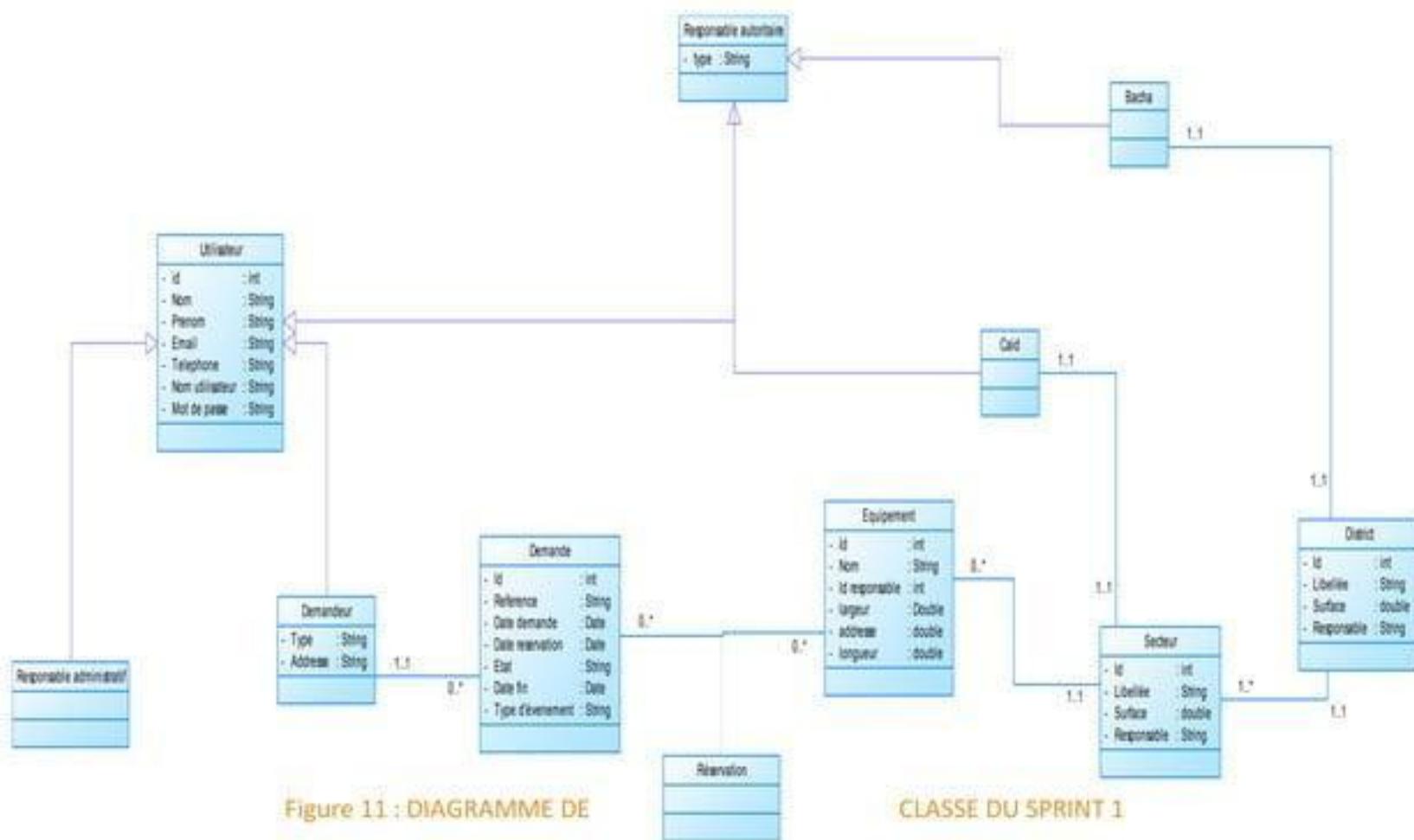


Figure 11: DIAGRAMME DE CLASSE DU SPRINT 1

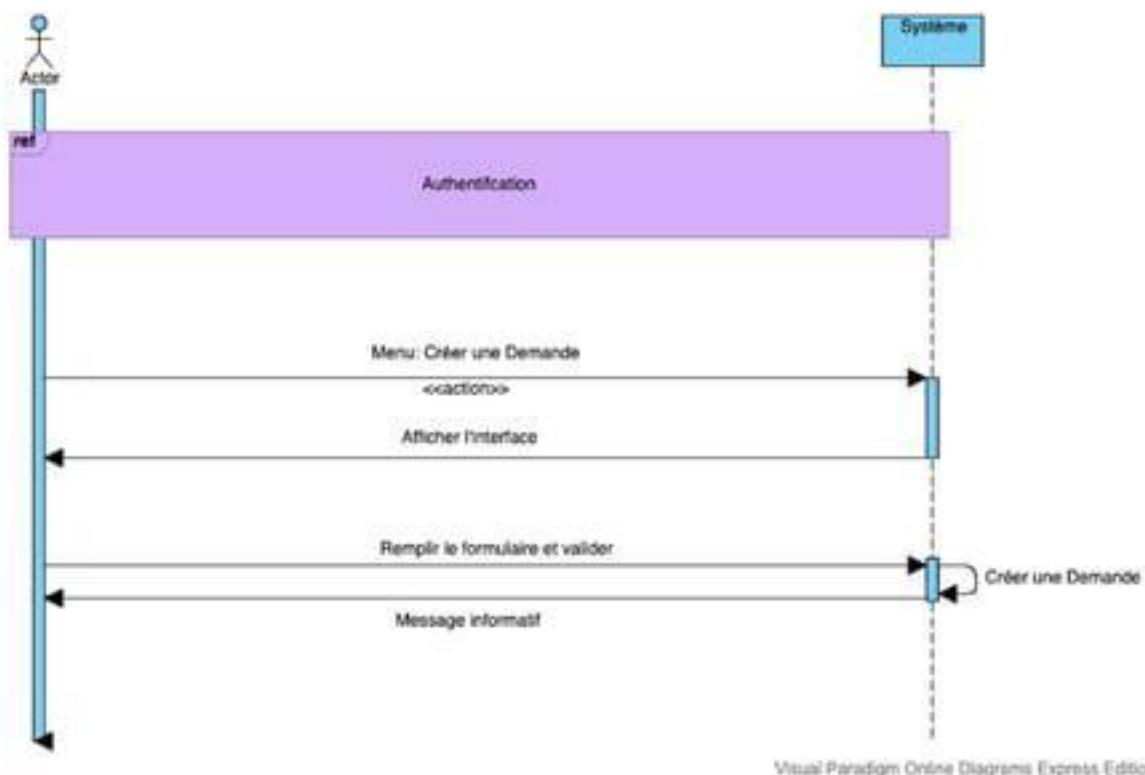
4-4 Diagramme de séquence :

4-4-1 Boîtes noires :

a. « Remplir une demande »

Le diagramme ci-dessous décrit les étapes détaillées du remplissage d'une demande afin de réserver un équipement :

a. Remplir une demande



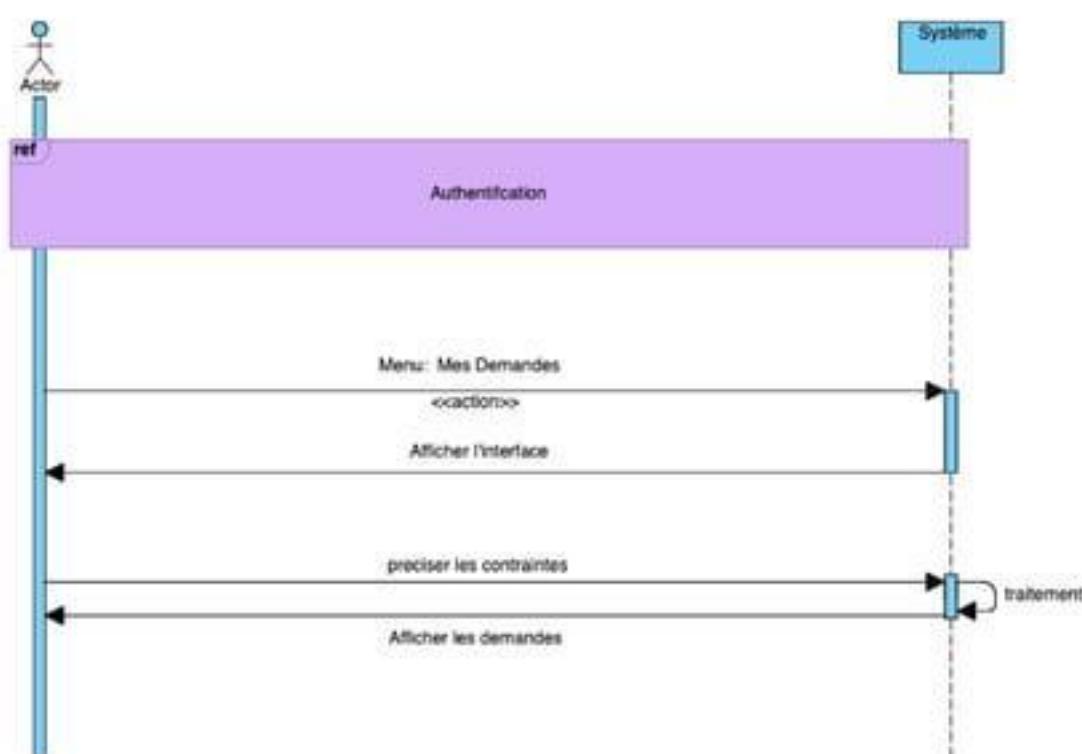
Visual Paradigm Online Diagrams Express Edition

Figure 12 : DIAGRAMME DE SEQUENCE DE « REMPLIR UNE DEMANDE »(BOITE NOIRE)

b. « Consulter les demandes »

Le diagramme ci-dessous décrit les étapes détaillées pour afficher les demandes d'un demandeur particulier, avec une possibilité de trier la recherche :

b. Consulter les demandes



Visual Paradigm Online Diagrams Express Edition

Figure 13 : DIAGRAMME DE SEQUENCE DE « CONSULTER UNE DEMANDE » (BOITE NOIRE)

4-4-2 Boites blanches :

a. « Remplir une demande »

Le diagramme ci-dessous décrit les étapes détaillées du remplissage d'une demande afin de réserver un équipement :

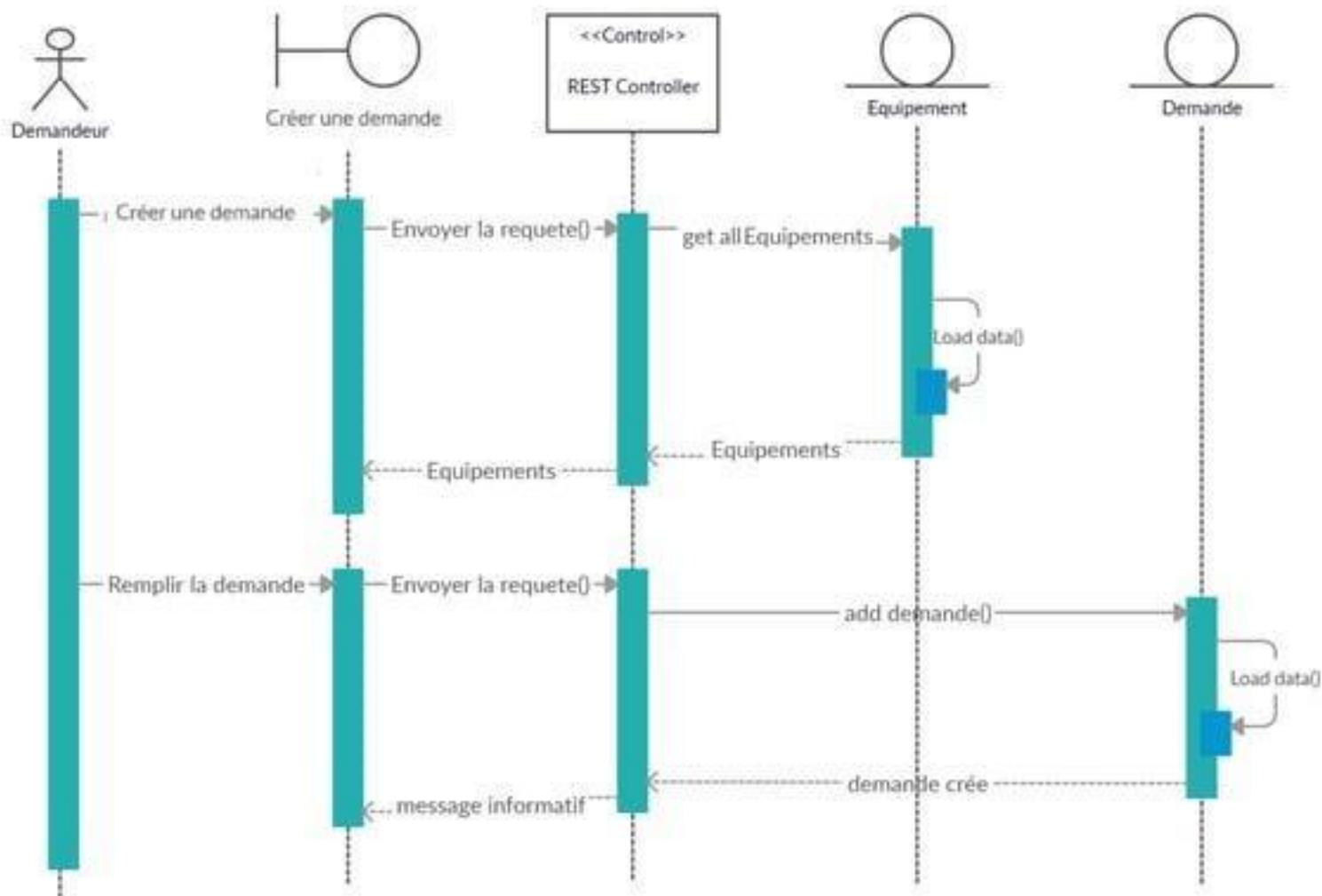


Figure 14: DIAGRAMME DE SEQUENCE DE « REMPLIR UNE DEMANDE »(BOITE BLANCHE)

b. « Consulter les demandes »

Le diagramme ci-dessous décrit les étapes détaillées pour afficher les demandes d'un demandeur particulier, avec une possibilité de trier la recherche :

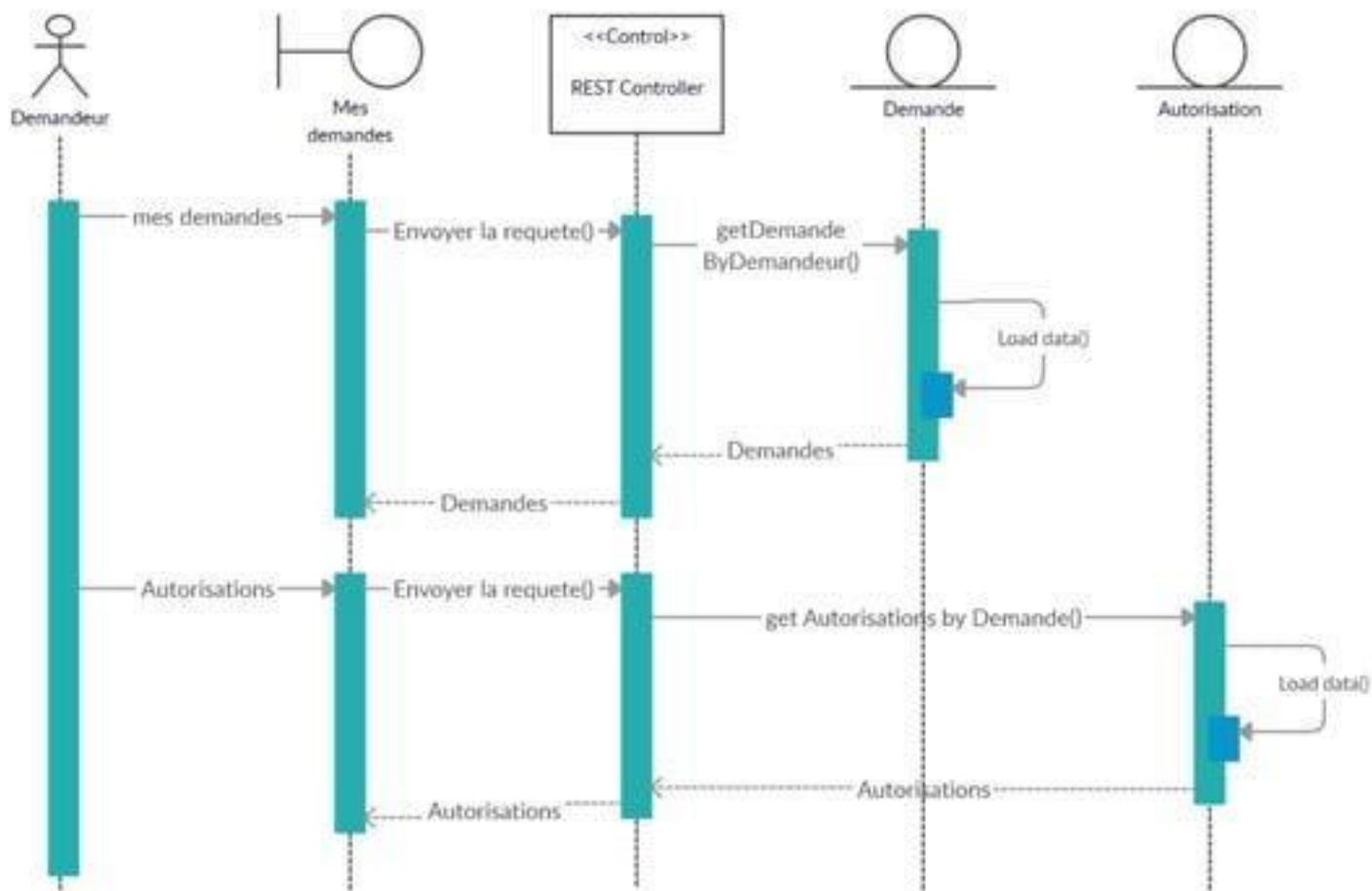


Figure 15: DIAGRAMME DE SEQUENCE DE « CONSULTER UNE DEMANDE »(BOITE BLANCHE)

5- Réalisation :

- Interfaces d'authentification :

Ci-dessous l'interface de « SignUp » qui concerne l'inscription des demandeurs :

Create a new account

Username :

Password :

Confirm Password :

nom :

prenom :

telephone :

Email :

type :

adresse :

S'INSCRIRE

Figure 16: INTERFACE D'INSCRIPTION

L'interface ci-dessous illustre un formulaire de connexion des demandeurs :

Home Actualité Listes des équipements

Sign up Log in

Username

Password

CONNEXION

[Forgot Password?](#)

Figure 17: INTERFACE D'AUTHENTIFICATION

- Interfaces d'un demandeur connecté :

L'image ci-dessous présente une interface pour déposer une demande de réservation :

The screenshot shows a web-based form for creating a reservation request. The interface is divided into two main sections: a blue sidebar on the left and a white main content area on the right.

Blue Sidebar (Left):

- Icon:** A white icon of a document with a grid pattern.
- Title:** **Remplir ma demande**
- Text:** Pour bien compléter votre demande avec succès, vous devrez remplir tous les champs

Main Content Area (Right):

Text: Monsieur mohamedi remplir Svp les champs de votre demande

Form Fields:

- saisir l'objet de la demande :**
- Préciser le type de l'événement :**
- Importer l'obligation légalisée :** Aucun fi...r choisi
- estimer un nombre de personnes qui se présenteront :**
- choisir l'équipement désiré :**
- choisir la date de réservation :**
- Description :**

Buttons:

- SUBMIT** button at the bottom of the form.

Figure 18: INTERFACE DE CREATION D'UNE DEMANDE

L'image ci-dessous présente une interface de consultation des demandes :

Figure 19: INTERFACE DES DEMANDES

Identifiant de la réservation:	53
La date de demande :	2020-06-21
La date de réservation :	2020-06-28
L'état de demande :	EN COURS

DÉTAIL

En cliquant sur une demande particulière, elle réagit comme suite :

En cliquant sur le bouton Détail :

Identifiant de la réservation :	86
la date de demande :	2020-06-27
l'état de demande :	initial
la date de réservation :	2020-07-24
L'équipement désiré :	dakha
Type d'événement :	hebdomadaire
L'objet :	festival
le nombre de présence estimé :	100
Description :	festival culturel

[Listes des équipements](#)
[Mon Profile](#)
[Mes demandes](#)
[Créer une demande](#)

[Log Out](#)

L'obligation légalisée :

Texte de l'obligation

– d'une part, l'acte juridique entendu comme une expression de la volonté de son auteur destinée à produire des effets juridiques ; l'exemple typique étant le contrat, véritable type exemplifié une donation ou un legs testamentaire (par exemple la vente) ;
 – et d'autre part, le fait juridique compris comme étant également un événement (qui n'a pas été nécessairement volonté par son auteur) tel que la loi accorde des effets de droit (ex. art. 3 du projet de réforme du droit des obligations), par exemple le paiement d'un immovable possédé même par qui on fait sauter le propriétaire, un accident de la circulation, un décès...
 Notamment dès lors que le contrat et le fait juridique sont nécessairement issus du droit privé.¹ Pour s'en convaincre il suffit de constater que nombreux énoncés qui sont bien au-delà de la distinction entre l'acte juridique et le fait juridique².

La jurisprudence n'est pas en reste et y fait référence, aussi bien en ce qui concerne la malice³ que la preuve⁴ ou encore l'application de la loi dans le temps (art. 2 C. civ.).

II. LA CLASSIFICATION DES OBLIGATIONS SELON LEUR OBJET

Objectif de l'obligation désigne la prestation due au créancier par le débiteur. Le Code civil prévoit trois sortes d'obligations : l'obligation de donner (art. 1132 à 1141 C. civ.), l'obligation de faire (art. 1142 à 1144 C. civ.) et l'obligation de ne pas faire (art. 1145 et 1146 C. civ.). On peut y ajouter la distinction entre les obligations en nature et les obligations immobilières.

A. L'obligation de donner

Article 1191 du Code civil dispose que le créancier fait « une concession par laquelle une ou plusieurs personnes « obligent, devant une ou plusieurs autres à donner [...] quelque chose ». Le terme « donner » doit cependant si bien être pris dans le sens donné, mais comme la délivrance du transfert de la propriété ; par exemple d'un bien objet du contrat. Une partie de la doctrine élève aujourd'hui comme une obligation donnante

1. Sur les critères de la classification édictée par le Code civil, d'ici on se reporte de ce qui vient juste plus de bas qu'il considère qui sont le volonté exprimée dans un contrat, ceci telles que la volonté d'un effet d'un énoncé, pour cette des obligations, et qui elles qui constituent déjà dans le contrat soit le volonté dans le contrat pour l'effacement des obligations immobilières d'obligations.

Figure 20: INTERFACE D'UNE DEMANDE

Conclusion :

Au cours de ce sprint, nous avons effectué la conception et la réalisation du module front office. Le sprint a été présenté dans le cadre d'une réunion de fin de sprint et préparation au prochain. Cette réunion était à la présence de l'équipe du projet et le Scrum Master et le product Owner. Après avoir été enrichi par leurs remarques et propositions de perspectives, nous entamons dans le prochain sprint la partie de mise à jour des prévisions des collaborateurs.

Mise en œuvre du Sprint 2 : Back Office

*Ce sprint a pour but de développer
le sprint de gestion des demandes des réservations des équipements culturels.*

1- Le Backlog du sprint :

Le Backlog du sprint contient une liste des tâches identifiées par l'équipe Scrum qui devront être réalisées avant la fin de sprint.

ID du scénario utilisateur	Pour qu'on puisse	Les taches	Estimation par jours
SC_2	Gérer les demandes par les gestionnaires	Développer l'interface de créer un compte Gestionnaire	6
		Développer l'interface d'authentification	5
		Développer l'interface contenant les équipements	3
			3
		Développer l'interface de traiter toutes les demandes des réservations.	6
			6
			6
			6
		Développer l'interface pour visualiser les demandeurs	8
		Développer l'interface du profile du gestionnaire	4
			4
		Développer l'interface des autorisations	5

Figure 21: BACKLOG SCRUM DU SPRINT 2 (GESTIONNAIRE)

ID du scénario utilisateur	Pour qu'on puisse	Les tâches	Estimation par jours
SC_2	Accéder aux statistiques par le décideur	Développer l'interface de créer un compte décideur	6
		Les activités de chaque gestionnaire	10
		Réservations, demandes et autorisations par équipement	
		Le nombre de réservations par mois	
		Réservations, demandes et autorisations par demandeur	
		Réservations, demandes et autorisations par gestionnaire	

Figure 22: BACKLOG SCRUM DU SPRINT 2 (DECIDEUR)

2- Analyse :

Afin de concrétiser les exigences demandées, nous avons réalisé des maquettes, pour qu'on puisse avoir une idée plus précise sur l'ergonomie du module.



Bonjour Gestionnaire

A

Listes des demandes

Id	equipement	Date de demande	status	
1	Agadir	20/05/2020	Initial	details
2	Tasila	21/05/2020	Validée	details
3	Houda	24/05/2020	Rejetée	details
4	Agadir	26/05/2020	Initial	details
5	Khayr dine	28/05/2020	Initial	details
6	Agadir	25/05/2020	Validée	details
7	Agadir	20/05/2020	Initial	details

Visualizeur de diagramme Online Diagrams Express Edition

Figure 23: MAQUETTE DU SPRINT 2

3- Conception :

Nous allons présenter pour ce sprint le diagramme de classe et le diagramme de séquence.

3-1 Diagramme de cas d'utilisation :

Le diagramme suivant montre les différents cas d'utilisation pour ce sprint.

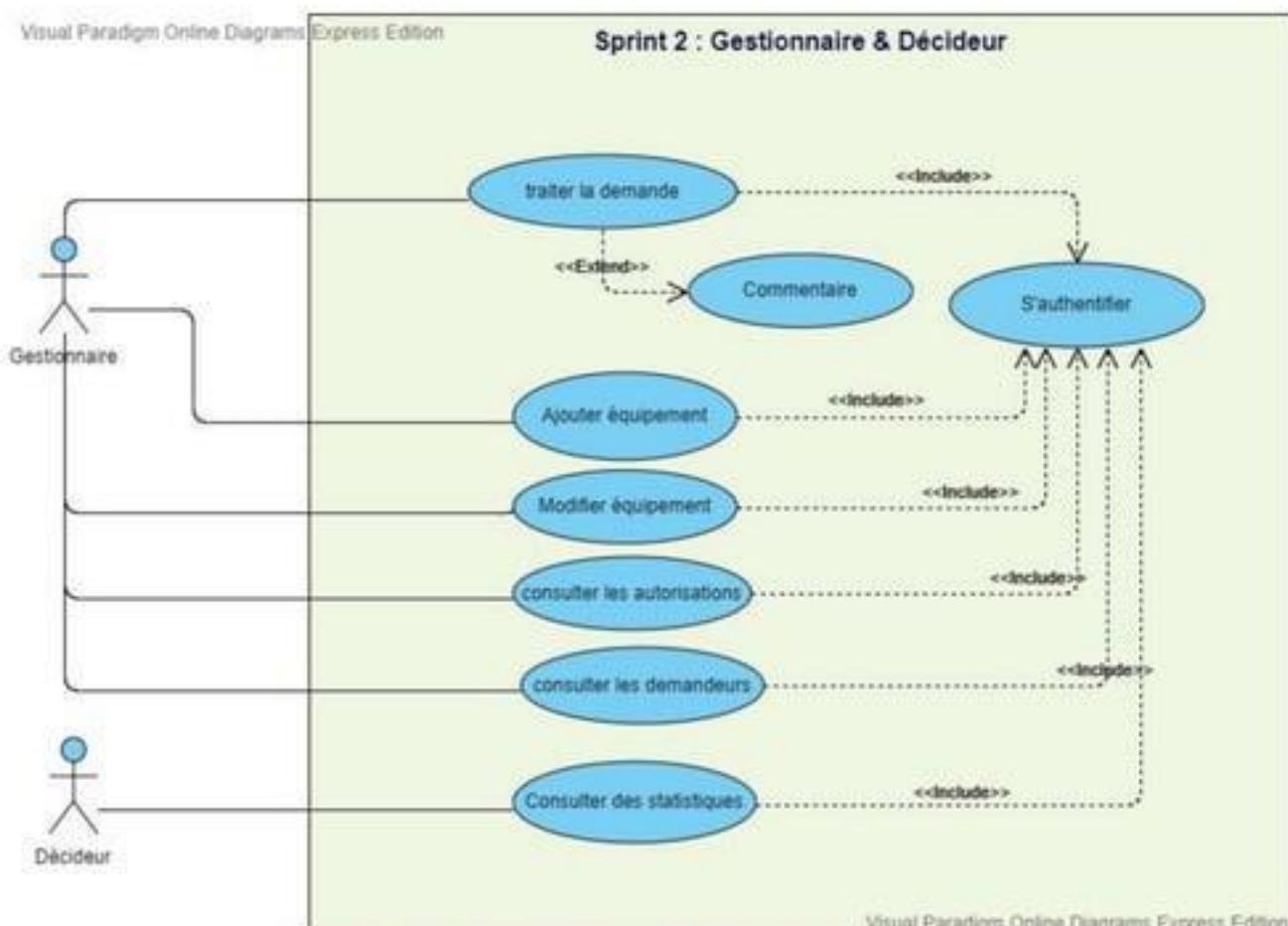


Figure 24: DIAGRAMME DE CAS D'UTILISATION DU SPRINT 2.

3-2 Description Textuelle :

3-2-1 Cas d'utilisation « Traiter une demande » :

UC_001- Traiter une demande	
Type	Core Use Case
Description brève	<ul style="list-style-type: none"> Ce cas d'utilisation décrit les différents cas pour valider ou rejeter une demande de réservation
Acteur principal	<ul style="list-style-type: none"> Gestionnaire
Pré conditions spécifiques	<ul style="list-style-type: none"> L'utilisateur doit être authentifié et son rôle soit « Gestionnaire »

Scénario nominal :	
6. L'utilisateur sélectionne « les demandes » dans le « Navbar » (l'en-tête de l'application).	
7. Le système présente la liste de toutes les demandes de réservations.	
8. L'utilisateur choisit une demande à traiter.	
Post-conditions spécifiques	<p>Une autorisation sera créée au cas où la demande est validée :</p> <ul style="list-style-type: none"> ✓ Sauvegarde des données dans la base de données.

Tableau 7 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « TRAITER UNE DEMANDE »

3-2-2 Cas d'utilisation « Consulter les demandeurs » :

UC_001- Consulter les demandeurs	
Type	Core Use Case
Description brève	Ce cas d'utilisation décrit la façon par laquelle le gestionnaire peut accéder aux données d'un demandeur ainsi que ses demandes.
Acteur principal	<ul style="list-style-type: none"> • Gestionnaire
Pré conditions spécifiques	<ul style="list-style-type: none"> • L'utilisateur doit être authentifié et son rôle soit « Gestionnaire »

Scénario nominal :	
<ol style="list-style-type: none"> 1. L'utilisateur sélectionne « les demandeurs » dans le navbar. 2. Le système présente la liste de tous les demandeurs. 3. L'utilisateur choisit un demandeur. 4. L'utilisateur visualise ses données afin de valider son état juridique. 5. L'utilisateur a la possibilité d'aller vers ses demandes et de savoir l'état de chaque demande. 	
Post-conditions spécifiques	➤ La liste des demandes est affichée

Tableau 8 : DESCRIPTION TEXTUELLE DU CAS D'UTILISATION « CONSULTER LES DEMANDEURS »

3-3- Diagramme de classe :

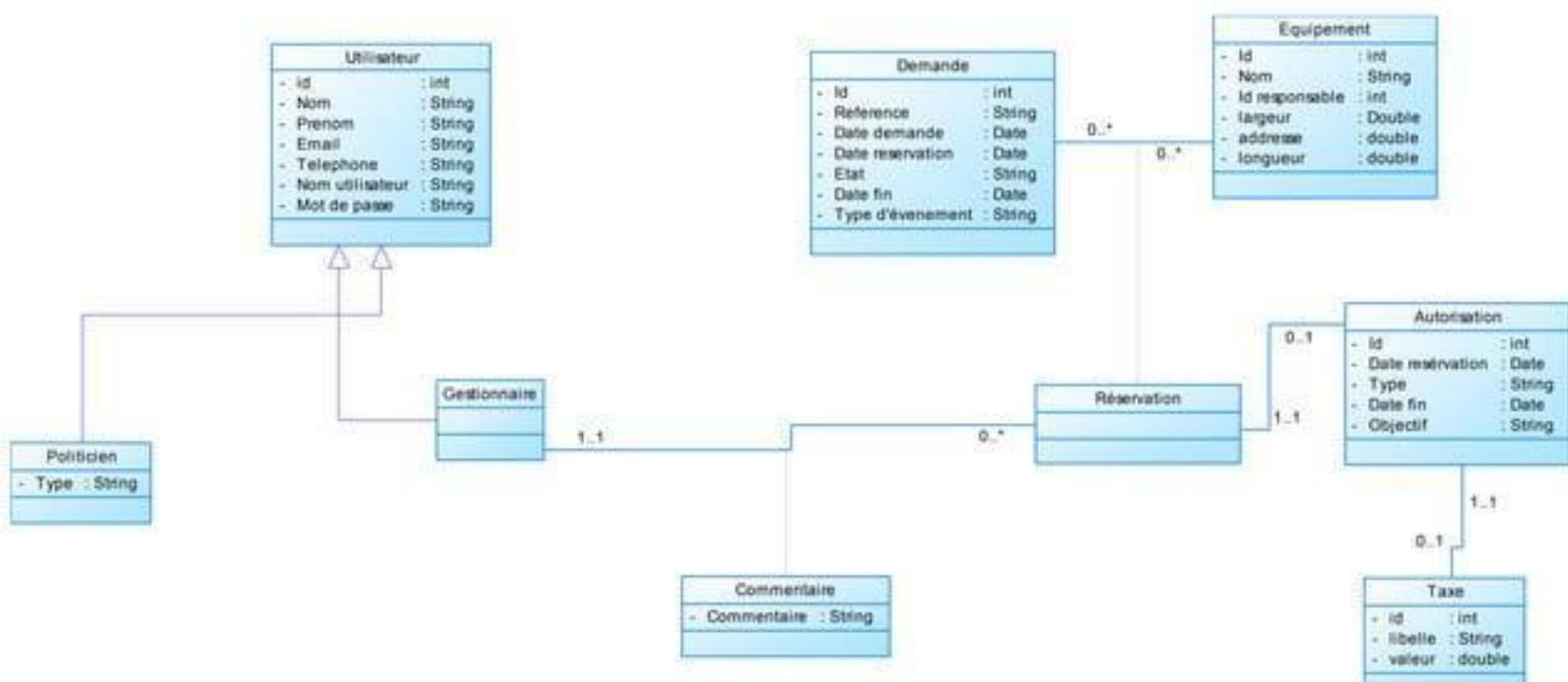


Figure 25: DIAGRAMME DE CLASSE DU SPRINT 2

3-4- Diagramme de séquence :

3-4-1 Boites noires :

a. « Traiter une demande»

Le diagramme ci-dessous décrit les étapes du traitement des demandes de réservations des équipements culturels.

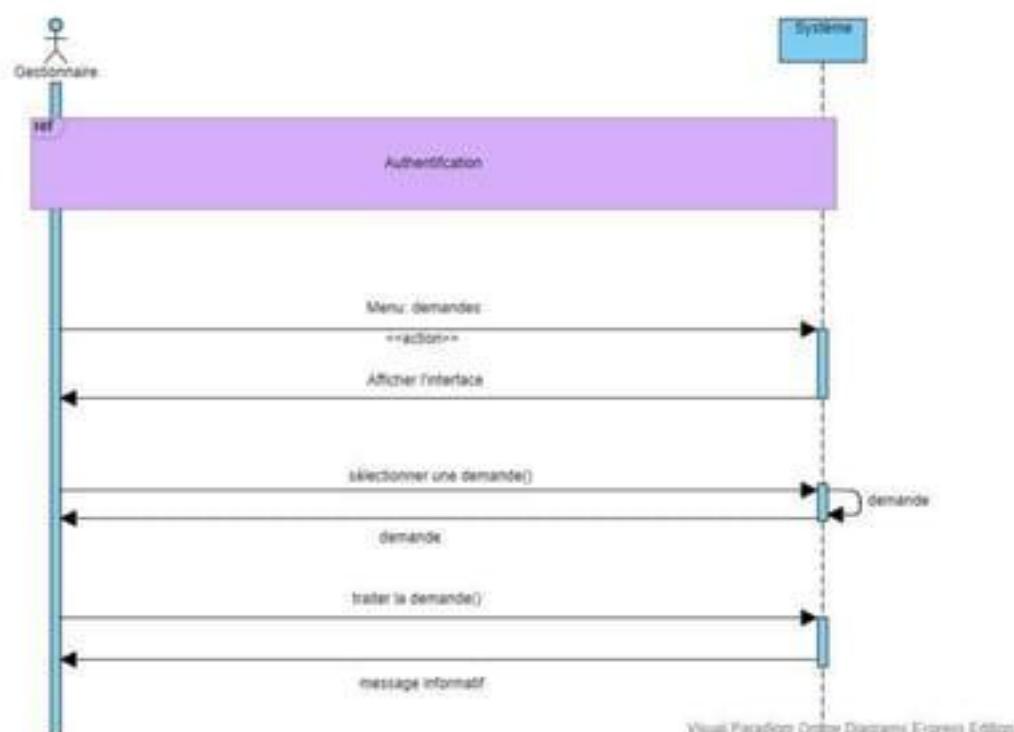
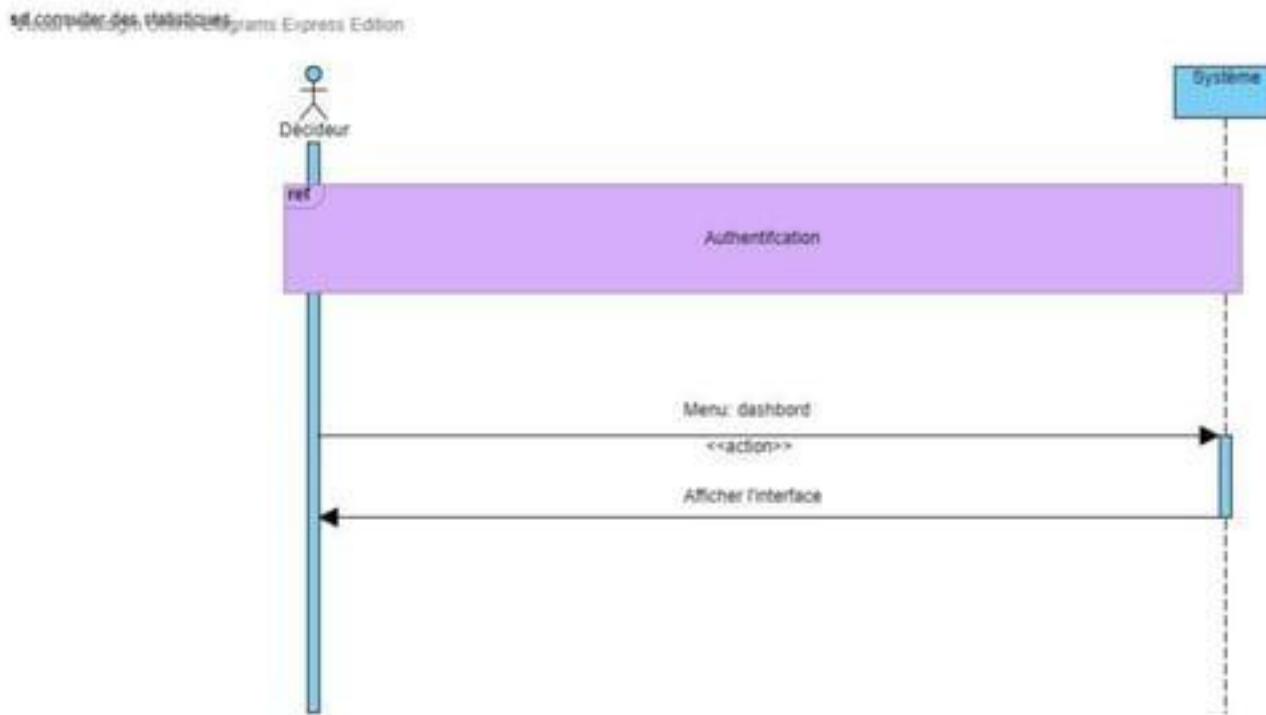


Figure 26 : DIAGRAMME DE SEQUENCE DE « TRAITER UNE DEMANDE »(BOITE NOIRE)

b. « Consulter les statistiques »

Le diagramme ci-dessous décrit les étapes pour afficher les statistiques concernant l'application :



Visual Paradigm Online Diagrams Express Edition

3-4-2 Boîtes blanches :

a. « Traiter une demande »

Le diagramme ci-dessous décrit les étapes détaillées du traitement des demandes de réservations des équipements culturels.

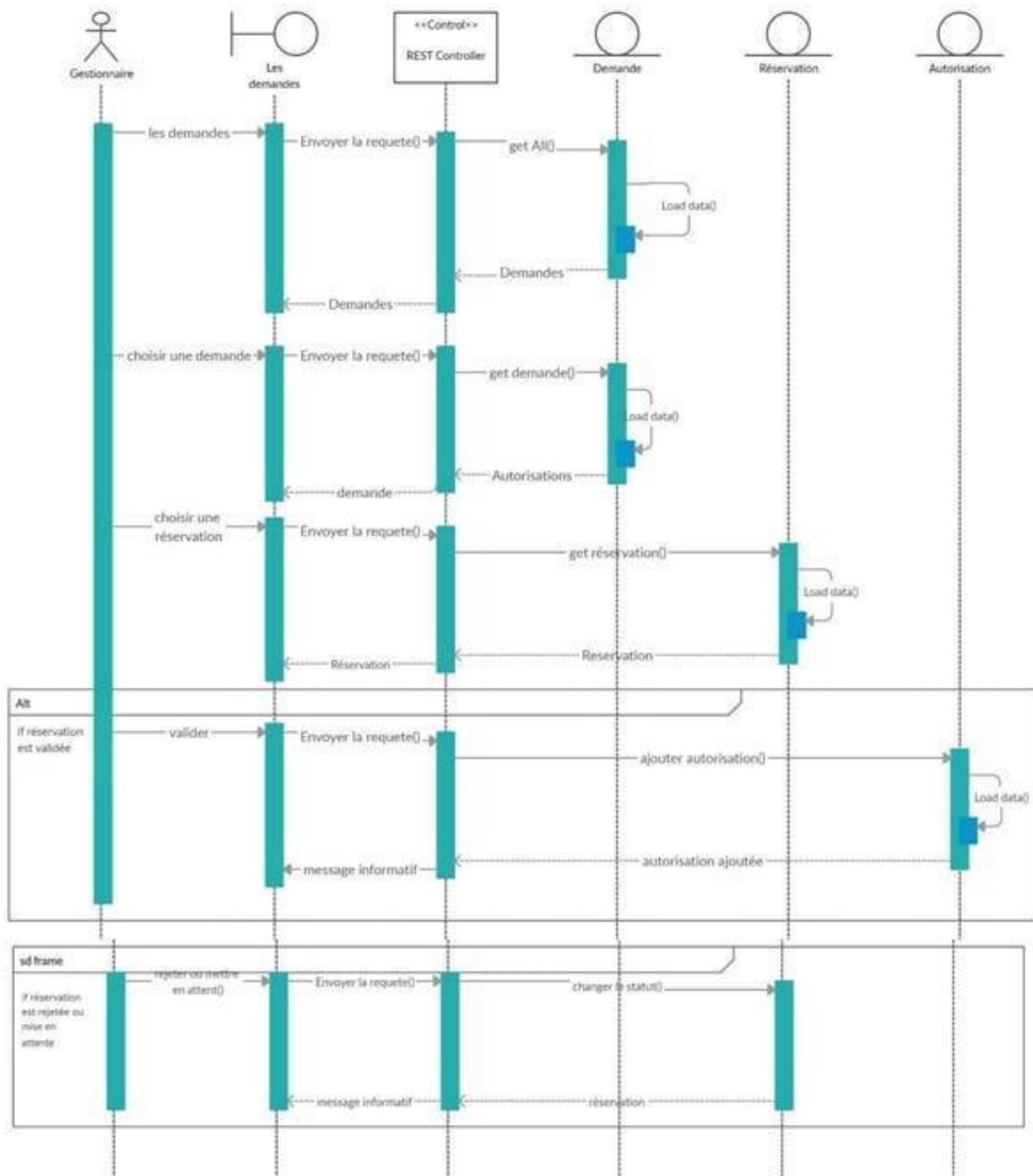


Figure 28: DIAGRAMME DE SEQUENCE DE « TRAITER UNE DEMANDE »(BOITE BLANCHE)

b. « Consulter les demandeurs »

Le diagramme ci-dessous décrit les étapes détaillées pour consulter les demandeurs et leurs demandes :

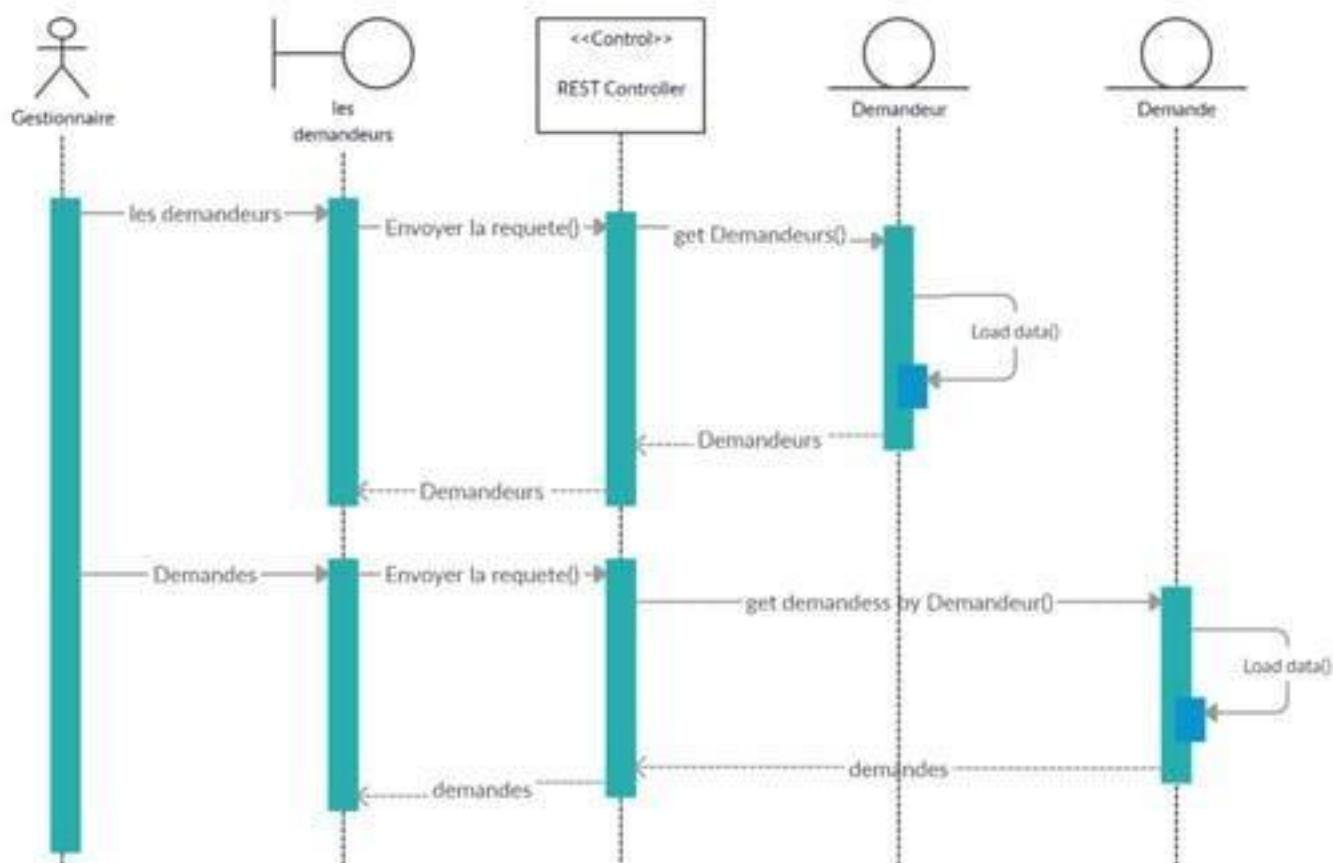


Figure 29: DIAGRAMME DE SEQUENCE DE « CONSULTER LES DEMANDEURS »(BOITE BLANCHE)

c. « Consulter les statistiques »

Le diagramme ci-dessous décrit les étapes détaillées pour afficher les statistiques concernant l'application :

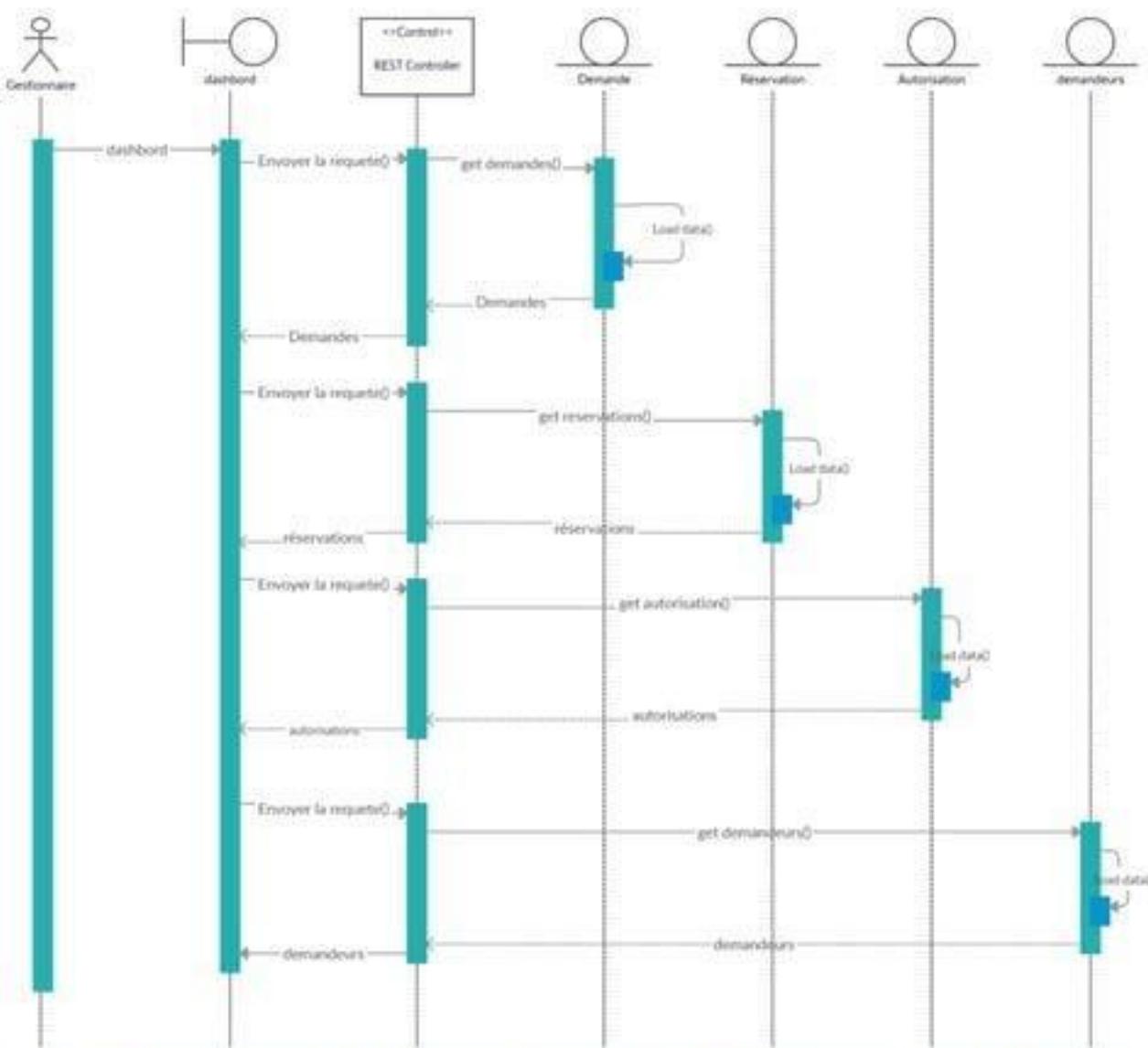


Figure 30: DIAGRAMME DE SEQUENCE DE « CONSULTER LES STATISTIQUES »(BOITE BLANCHE)

5- Réalisation :

5-1 Gestionnaire :

En premier lieu, les interfaces ci-dessous représentent des exemples des parties réalisées au module du gestionnaire :

- Interface des demandeurs :

Ce screen présente la liste des demandeurs :

Identifiant	Username	Nom	Prénom	Email	
4	akhachikh	akhachikh	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
8	amar	mohamed	mohamed	akhachikhouch@gmail.com	<button>DETAILS</button>
9	mohamed	nasser	mohamed	akhachikhouch@gmail.com	<button>DETAILS</button>
11					<button>DETAILS</button>
16	akram	akhachkhouch	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
17	marwa	akhachkhouch	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
18	malak	akhachkhouch	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
19	khalil	akhachkhouch	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>

Figure 31: INTERFACE DES DEMANDEURS

- Interface des réservations :

Identifiant	Username	Nom	Prénom	Email	
4	akhachikh	khalil	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
8	amar	mohamed	mohamed	akhachikhouch@gmail.com	<button>DETAILS</button>
9	mohamed	nasser	mohamed	akhachikhouch@gmail.com	<button>DETAILS</button>
11		mohamed			<button>DETAILS</button>
16	akram	akhachkhouch	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
17	marwa	salami	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
18	malak	tester	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>
19	khalil	akhachkhouch	ahmed	akhachikhouch@gmail.com	<button>DETAILS</button>

Figure 32: INTERFACE DES RESERVATIONS

Listes des équipements

Vous trouvez ici la liste des équipements culturels disponibles.

The screenshot shows a web-based application interface. On the left, there is a vertical list of equipment names in boxes:

- complexe culturel mohamed khair eddine
- complexe culturel mohamed jamal addorra
- complexe culturel de bensergao
- conservatoire de musique
- Théâtre de Vendure
- complexe culturel de Anza

On the right, there is a detailed view of the first item in the list:

complexe culturel mohamed khair eddine

Description: est le complexe où se trouve le service des affaires culturelles

Nom du Responsable: Abdellah ELMOUNTAII

Adresse: Agadir

Coordonnées: (100,100)

A small thumbnail image of an interior space with blue walls and a glass partition is visible above the detailed view.

- Interface des équipements

Figure 33: INTERFACE DES EQUIPEMENTS

- Modifier un équipement :

L'interface ci-dessous décrit les champs spécifiques dont le gestionnaire a l'accès de modifier.

The screenshot shows a web-based application interface for managing equipment. At the top, there's a navigation bar with links: 'Listes des équipements', 'Mon profile', 'Réservations', 'Autorisations', 'Les Demandeurs', and 'Log Out'. Below the navigation is a form titled 'tasse'. The form contains the following fields:

- ID: 1
- Nom: tasse
- id du Responsable: 1212
- adresse: 121
- Section: 1
- Etat: suspendu

At the bottom right of the form are two buttons: 'SAVE' and 'ANNULER'.

Figure 34: INTERFACE DE MODIFICATION D'UN EQUIPEMENT

5-2 Décideur :

En deuxième lieu, les graphes ci-dessous illustrent globalement une vue d'ensemble complète sur les acteurs contribuant dans le processus de la demande d'une réservation d'équipement (gestionnaire, demandeur), ainsi que des statistiques provenant des demandes, réservation et autorisations après les collecter, traiter et finalement les organiser.

- Demandes représentées mensuellement pendant toute l'année :

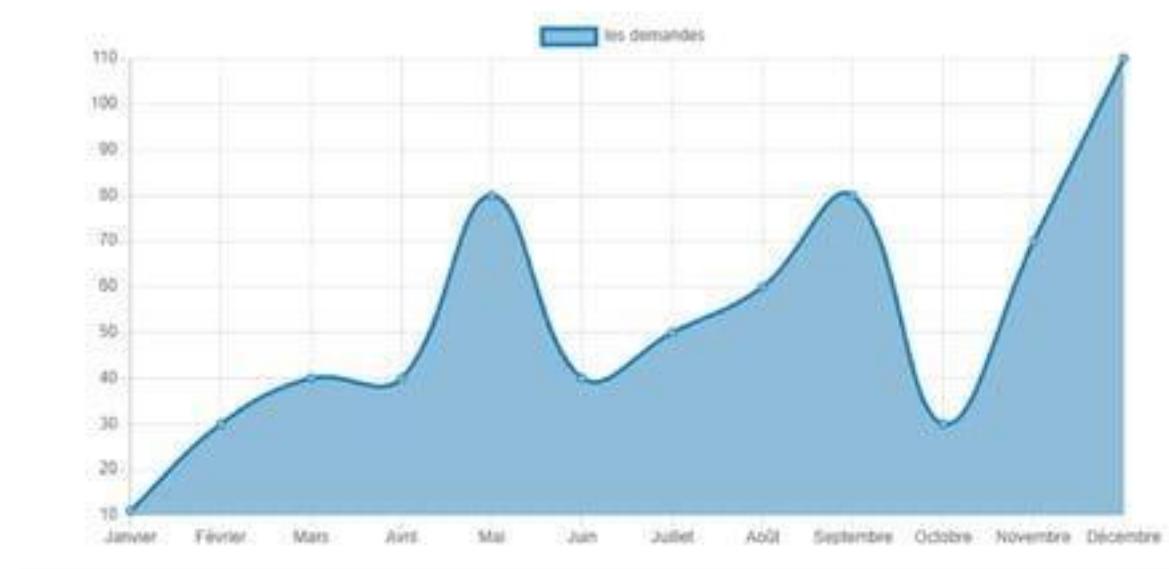


Figure 35: LES DEMANDES PENDANT L'ANNEE

- Demandes définies par leur demandeur :

Ce graphe ci-dessous nous présente d'une manière globale le nombre des demandes déposées par chaque demandeur :



Figure 36: LES DEMANDES PAR DEMANDEURS PENDANT TOUTE L'ANNEE

- Demandes définies par statut :

Ce graphe ci-dessous nous montre clairement la partition des demandes par l'état d'avancement :

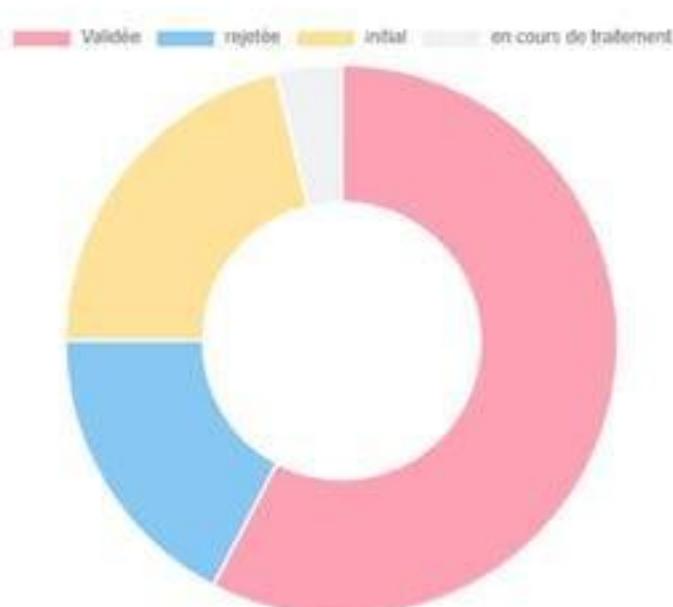
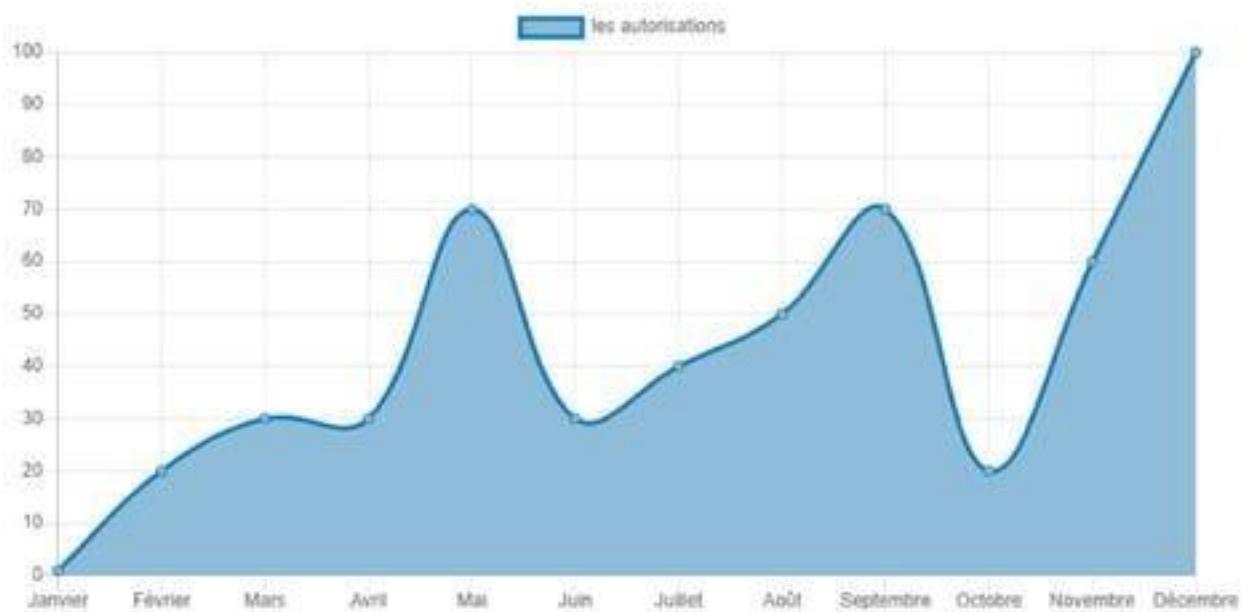


Figure 37: LES DEMANDES PAR ETAT D'AVANCEMENT

- Autorisations affectées par gestionnaire :

Ce graphe ci-dessous nous offre une vue d'ensemble sur le nombre d'autorisations traitées par chaque gestionnaire.

**Figure 38: LES AUTORISATIONS PENDANT TOUTE L'ANNEE**

- Autorisations fournies pour chaque équipement :

Ce graphe ci-dessous nous informe d'une idée générale des autorisations affectées par les gestionnaires aux demandeurs pour exploiter un équipement spécifique.



Figure 39: LES AUTORISATIONS PAR EQUIPEMENT

Conclusion

Au cours de ce sprint, nous avons effectué la conception et la réalisation du module du Back office, Le sprint a été présenté dans le cadre d'une réunion de fin de sprint. Cette réunion était à la présence de l'équipe du projet et le Scrum Master.

Conclusion general :

Le présent document est la présentation du travail réalisé durant notre stage de Fin d'Année au sein de la commune d'Agadir, précisément au service informatique en coordination avec la division des affaires culturelles, sociales et sportives. L'objectif de ce stage est de développer une solution digitale qui optimise la gestion des équipements culturels de la commune d'Agadir.

Ce projet a eu lieu pour combler les lacunes de la procédure manuelle de gestion des équipements culturels utilisé actuellement par la division des affaires culturelles, sociales et sportives de la commune d'Agadir. Comme solution nous avons pensé à digitaliser cette procédure pour générer des économies importantes, de réduire les temps liés aux actions administratives et donc d'optimiser la réactivité et l'efficacité de cette division. En outre, la mise en place du zéro papier permet aussi de garantir l'intégrité et la fiabilité des données, de faciliter leur archivage et leur sauvegarde, de conserver la traçabilité de leur flux mais aussi d'assurer une confidentialité des différents échanges.

L'utilisation de la méthodologie de Scrum a été un bon choix pour nous, puisqu'il nous a vraiment aidé à bien conduire notre projet tout en évitant les inconvénients de la méthodologie classique. D'une façon générale, au premier lieu, nous avons essayé de comprendre le contexte général du projet et les attentes de nos clients. Ensuite, le choix de l'architecture logique et logicielle, et les outils et technologies plus claire après cette analyse initiale. Nous avons préparé la suite en respectant les priorités des besoins déjà fixés avec l'équipe Scrum. Le premier Sprint en étant le "front-office", nous a pris un bon moment, pour réaliser une interface qui sera facile à être utilisé. Et pour le dernier sprint, ça été plus riches des fonctionnalités, puisqu'il est le sprint de gestion "back-office".

Finalement, nous avons appris plusieurs choses non seulement techniques, mais aussi côté comportement et communication (façon d'interagir avec les clients et l'équipe). Cette expérience acquise va nous aider sûrement à réaliser les prochains projets plus correctement. Notre travail ne s'arrête pas dans ce stade. Plusieurs perspectives que nous pouvons envisager pour étendre les fonctionnalités de ce projet. Parmi elles, Nous pouvons proposer un module d'évaluation des événements qui sera destiné aux responsables d'équipements culturels. Il leurs permettra grâce à un système de rating de noter les événements organisés dans les équipements culturels dont ils sont responsables. Les gestionnaires et décideurs pourront se servir aussi de ce module d'évaluation pendant le traitement et la prise de décision pour mieux répondre aux futures demandes de réservation.

Webographie

- ✓ Angular. "Site officiel". Mise à jour le 18 Février 2019. [En ligne] <https://angular.io/> [consulté le 20 Février 2020].

- ✓ Florent Lothon. "Introduction aux méthodes agiles et Scrum". Mise à jour le 26 Février 2019. [En ligne] <https://agiliste.fr/introduction-methodes-agiles/#Fonctionnement-des-m-thodes-agiles> [consulté le 06 Mars 2020].

- ✓ Florent Lothon. " Guide de démarrage Scrum". [En ligne] <https://agiliste.fr/guide-de-demarrage-scrum/> [consulté le 05 mars 2020].

- ✓ Github. "Site officielle ". [En ligne] <https://github.com/> [Consulté le 20 Avril 2020].

- ✓ Maximilian Schwarzmüller. " Angular - The Complete Guide (2020 Edition)". [En ligne] <https://www.udemy.com/course/the-complete-guide-to-angular-2/> [Consulté le 10 février 2020].

- ✓ Mohamed ELYOUSSI. "JWT Spring security Med Youssfi". [En ligne] <https://www.youtube.com/watch?v=1BIJeu-Q4dk&list=PLMgAr-DXdLC1F3p7KbFHEFCinUvvSfgOK> .[Consulté le 10 février 2020].

- ✓ Creatly. "Site officiel". [En ligne] <https://creately.com/app/#> [Consulté le 27 Mars 2020].