A Mon Père

Merci pour tout

DEDICACE

REMERCIEMENT

La réalisation de ce rapport s’est faite grâce au soutien de l’**Eternel.**

Au terme de ce travail, nous tenons à exprimer notre reconnaissance et notre profonde gratitude à :

* *Dr.* ***OUAFO Blaise,*** *Directeur de IME, pour la formation de qualité qu’il met à notre disposition ;*
* *M.* ***KENFACK Bienvenu,*** *notre encadreur académique, pour sa disponibilité, son soutien et son attention à notre égard ;*
* *M.* ***FOKENG Silva,*** *responsable du département recherche et développement, qui nous a permis d’effectuer un stage chez SOFINEZ pour son soutien, pour le temps qu’il nous a consacrés, ainsi que les conseils utiles qu’il nous a donnés tout au long de notre stage.*
* *Nos vifs remerciements s’adressent également aux membres du jury qui ont accepté d’évaluer notre travail*
* *Tous les enseignants d’IME à qui nous devons du respect. Et le personnel administratif pour leur sympathie et aux services qu’ils nous ont rendus ;*
* ***Mon père****,* ***maman Germaine*** *pour votre patience, vos sacrifices et votre soutien tout au long de mon parcours ;*
* *A l’ensemble de ma famille, à mes grands-parents pour leur accompagnement tout au long de ma vie ;*
* *A mes chers frères, sœurs, cousins et cousines avec qui j’ai passés des instants inoubliables. Je ne saurais exprimer ma reconnaissance la plus sincère et mon respect le plus profond pour tout votre amour, ainsi qu’à vos encouragements qui ont été pour moi un grand soutien moral tout au long de mes études ;*
* *A mes camarades de promotions ;*

A tous ceux qui m’ont soutenu, **Merci** infiniment.

PRESENTATION DE L’ENTREPRISE

1. ENVIRONNEMENT INTERNE DE L’ENTREPRISE
2. HISTORIQUE

Créée en 1999, par un groupe d’ingénieurs polytechniciens, Sécurité Electronique (SECEL) est une SARL camerounaise dont la vocation première est d’être à l’écoute de ses clients et leur apporter la solution idoine à leurs problèmes liés aux Technologies de l’Information et de la Communication (TIC). Au cours des années, cette vocation s’est consolidée. C’est ainsi que la fourniture des solutions matérielles, logicielles et managériales, a connu un fort développement. Toujours à cette période, la fourniture des services comme la maintenance informatique, l’infogérance, et la formation n’ont pas été en reste. Aujourd’hui, SECEL continue son expansion et a ainsi innové en mettant sur pied un Centre de compétences dédié non seulement au transfert de compétences tant dans le domaine des TIC, que du management mais aussi aux formations professionnelles.

1. SECTEURS D’ACTIVITES

SECEL demeure l’entreprise dont la vocation est d’être à l’écoute des entreprises pour leur apporter la solution idoine aux problèmes liés aux TIC. La persévérance dont SECEL fait preuve est le signe de la satisfaction que procure chacun des produits et services ; En effet, SECEL est. « **HP** Gold Partner », « **DELL** Business Partner », «**CITRIX** Solution Advisor », **« MICROSOFT** Gold Certified Partner », « **MICROSOFT** Certified Learning Solution », **«** Accredited **EXIN** Center» et Distributeur officiels des produits et solutions **EATON** au Cameroun. A ce titre, SECEL peut mettre à profit la bonne relation qu’elle entretient depuis des années avec ces différents partenaires fournisseurs pour fournir des produits et services dans les conditions les plus avantageuses.

SECEL aujourd’hui, c’est une couverture nationale et internationale, c’est un ensemble de carrières dans plusieurs corps de métier, c’est un point d’honneur mis sur un système de valeur partagées et un engagement pour la satisfaction totale de leurs clients et des autres parties prenantes à travers une qualité croissante de leur prestation de service. Tous ces éléments traduisent une seule ambition :

**Faire de SECEL le premier centre de compétences multidisciplinaires d’envergure internationale en Afrique Centrale**.

Offrir ainsi aux entreprises des solutions innovantes et personnalisées aux problèmes de système d’information et de management, qui répondent aux exigences légitimes de performance, dans la réalisation de leurs ambitions.

Les métiers de SECEL s’articulent autour de trois principaux pôles :

* Protection électrique des onduleurs de grande puissance ;
* ITIL, COBIT et autres bonnes pratiques informatiques ;
* Virtualisation (Citrix, VMware) ;
* Infrastructure d’accès et réseau WAN (Citrix, Packeteer) ;
* Interconnexion et VSAT ;
* Stockage des données et IT Business Continuity ;
* Gestion des projets selon les recommandations du PMI ;
* Les Solutions matérielles et logicielles ;
* La formation et certification ;
* Le conseil et l’assistance **;**
* Son expertise est exprimée à travers 10 domaines de compétences;
* Maintenance du parc informatique ;
* Infogérance matérielle et logicielle ;
* Gestion des projets selon les recommandations

1. LES DEPARTEMENTS

SECEL est une équipe de près de 53 personnes (principalement jeunes) constituée en particulier d’ingénieurs et commerciaux tous diplômés des universités et grandes écoles du Cameroun. Les talents de cette équipe sont organisés en Service. Enumérés comme suit :

1. Un Service Marketing et Commercial

Constitué des hommes et des femmes à votre écoute pour répondre à toutes les sollicitations, mais surtout aider à accroître l’efficacité des actions d’optimisation des systèmes de travail, de renforcement des capacités de vos collaborateurs, à travers leurs conseils.

1. Un service Logistique et Achats

Qui anticipe sur les besoins, et recherche au quotidien la meilleure solution pour répondre à votre objectif de performance au moindre coût.

1. Département Soft and Training

En tant que centre de formation, ce département est spécialisé dans la formation informatique tels que ITIL Foundation, PMP, administration Windows server, Sharepoint, le développement des applications et bien d’autres.

1. Un centre de Service

Doté d’un personnel jeune, dynamique et courtois, qui met à votre disposition son assistance 7j/7et 24H/24 conformément aux normes internationales ITIL.

Ces ingénieurs Techniques et commerciaux sont tous certifiés et spécialisés sur les produits et services proposés aux clients de SECEL. On note ainsi :

* 04 certifiés CITRIX ;
* 07 certifiés HP ;
* 04 certifiés DELL ;
* 02 certifiés McAfee ;
* 01 certifié Packeteer ;
* 03 certifiés CISCO ;
* 02 certifiés mikrotik.
* 06 certifiés MICROSOFT tous MCT (Microsoft Certified Trainer);
* 02 MCPD (Microsoft Certified Professional Developper) ;
* 18 certifiés sur la bonne pratique ITIL ;
* 03 PMP (Project Management Professional) du PMI ;

La jeunesse des équipes confère entre autre souplesse, réactivité, pro activité et disponibilité, autant de qualités requises pour combler les attentes d’une prestigieuse clientèle.

1. LES SOLUTIONS ET LES SERVICES OFFERTS

* **SOLUTIONS MATERIELLES & LOGICIELLES**

**SECEL** offre une diversité de produits, services et formations informatiques et managériales relevant des TIC en partenariat avec de nombreux éditeurs mondiaux pour garantir une qualité absolue. Elle met également à la disposition des clients des systèmes d’exploitation, des outils bureautiques, des outils de développement, des applications spécifiques pour une exploitation optimale de leur matériel.

* **FOURNITURES DES SERVICES INFORMATIQUES**

Elledispose d’une équipe de techniciens qui réalisent l’étude des systèmes informatiques. Cette équipe conseille également dans le choix et l’implémentation des solutions les mieux adaptées à votre entreprise.

* **SECURITE ELECTRONIQUE**

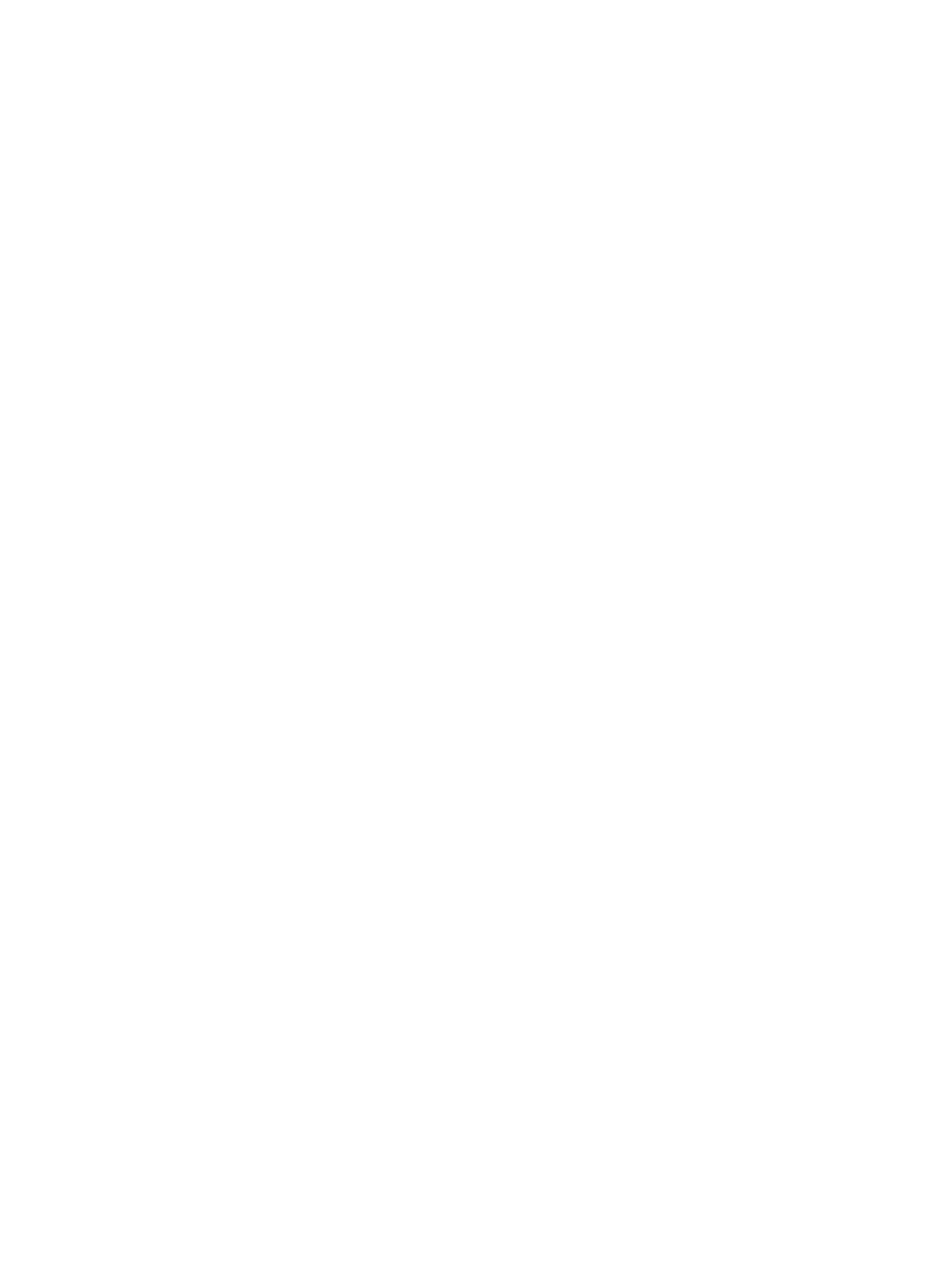
Spécialisée dans la fourniture des solutions de sécurité basées sur les systèmes de vidéo-surveillance, de câblage réseau et d’étude d’installation de satellite VSAT, SECELconçoit, étudie et réalise, tout un ensemble de systèmes électroniques permettant de sécuriser l’accès à des bâtiments et bureaux.

1. ORGANIGRAMME DE SECEL

Figure1 : Organigramme de SECEL Sarl

1. SITUATION GEOGRAPHIQUE

Figure2 : Plan de situation de l’entreprise SECEL



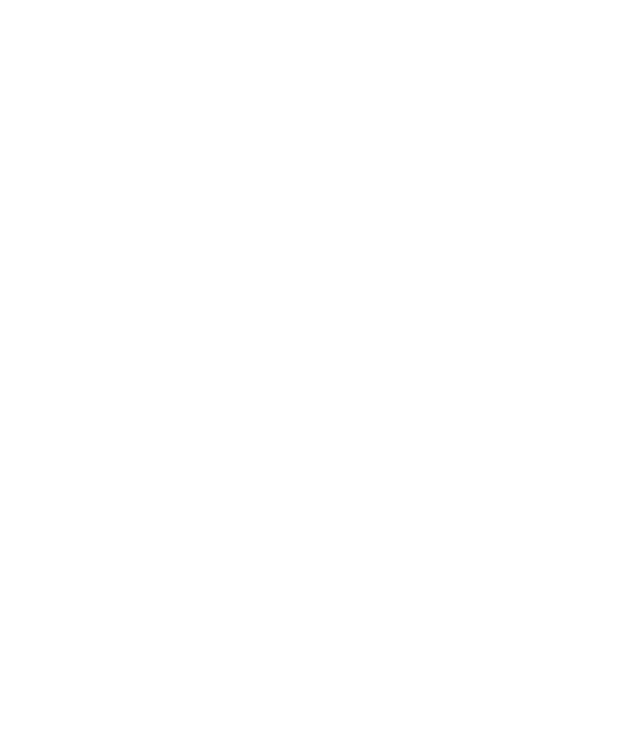
**Douala**



**Yaoundé**



**Garoua**



**SECEL Douala**

**Direction Générale:**

Immeuble I.M.E Situé à 50m de

La mairie de Douala 5

ème

A Bonamoussadi

Tel

:

33 47 88 01

–

Fax

:

99

33 47 87

**Agence de Bali**

606

, boulevard de la République

Carrefour «

Nouvelle route Bonadibong

», Immeuble

R+1, 1

er

étage

BP 15685 Douala

-

Cameroun

Tél

-

fax

(237) 33 43 74 93

:

–

33

64

88

43

E

-

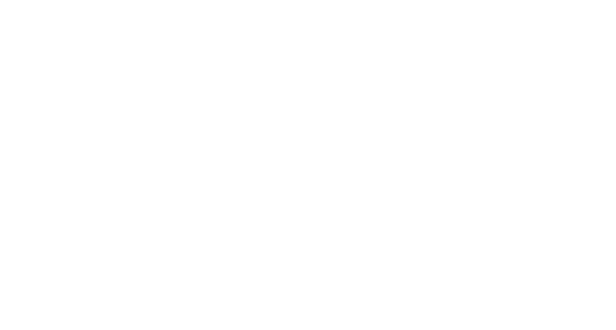
mail

:

INFODLA@SECELGROUP.C

OM

SCC@SECELGROUP.COM



**SECEL Yaoundé**

Omnisport, Derrière TEXACO BP 14828 Yaoundé

Tél

-

fax

:

(237) 22 21 40 08GSM

(

:

237) 99 10

77 56

CT phone

:

(237) 22 01 12 84

–

01 12 82

22

E

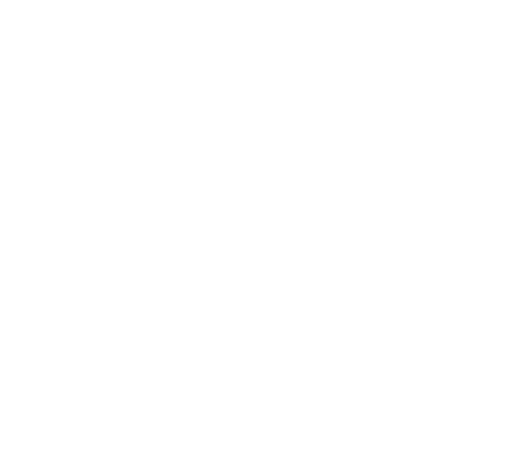
-

mail

:

INFOSYDE@SECELGROUP.

COM



**SECEL Garoua**

**Bidemire**

**Centre Commercial**

Face Direction PMUC

Tél: 22 27 13 83

GSM: 699 68 62 67 6

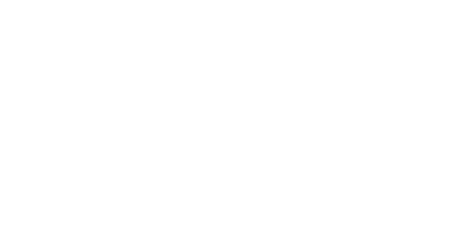
–

19 56 92

77

Email

INFOSGAR@SECELGROUP.COM



**Représentant commercial SECEL**

**Bertoua**

Nkol

-

bikon, Route Nationale N°1

CT Phone

32

22 03 96

:



SECEL Ndjamena



Ndjamena

**Bertoua**

1. FICHE D’IDENTIFICATION ET ORGANISATION

* FICHE D’IDENTIFICATION

Tableau 1 : fiche d’identification de SECEL

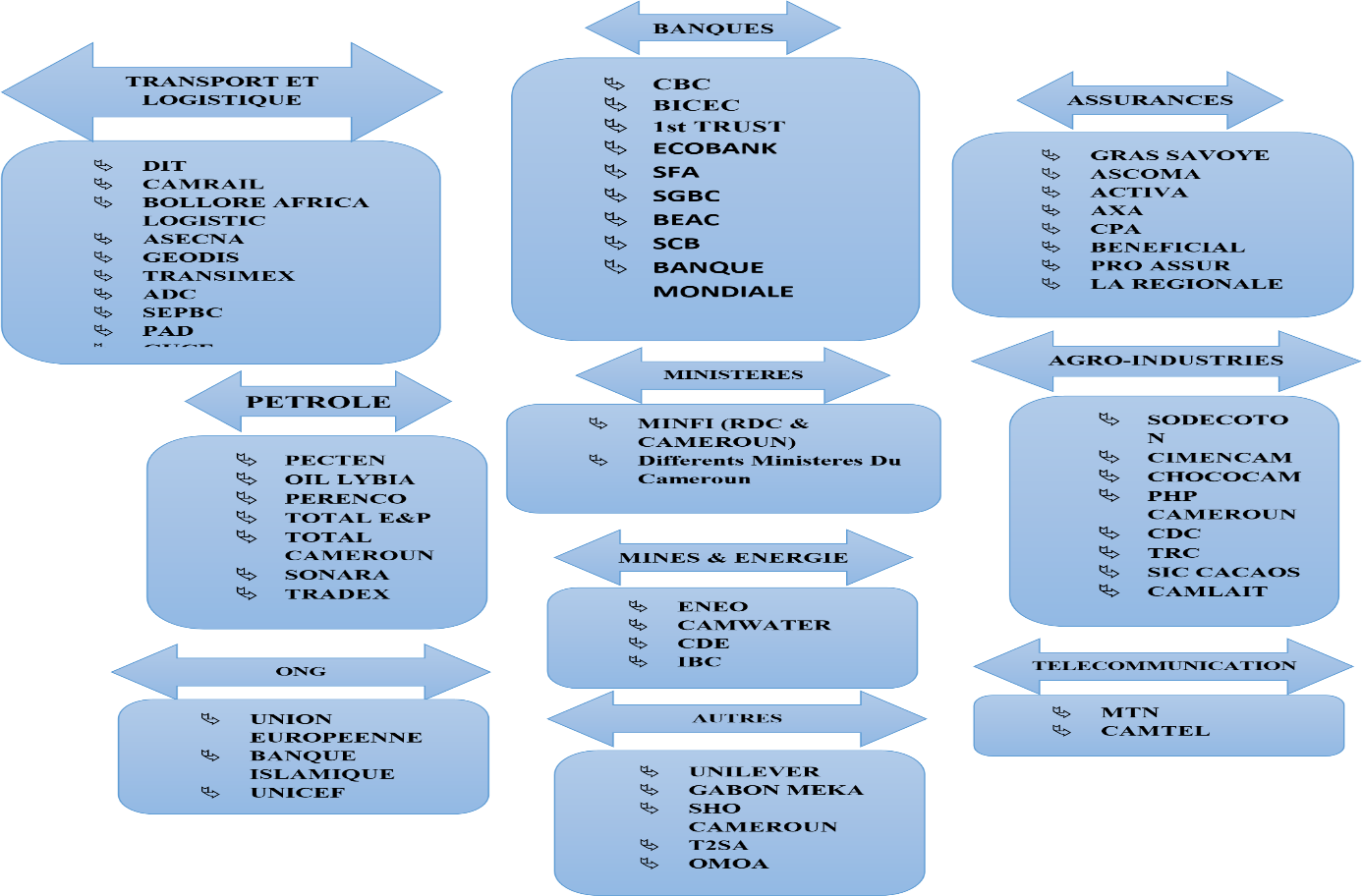
|  |  |
| --- | --- |
| Désignation | Informations |
| Nom de l’entreprise : | SECEL |
| Raison sociale | SARL |
| Capital de la société | 5 000 000FCFA |
| Numéro d’immatriculation | M100100013211L |
| Date et lieu d’immatriculation | 03 OCTOBRE 2001 DOUALA |
| Directeur Général | Blaise OUAFO |
| Secteur d’activité | Informatique & Management |
| Effectifs | Global : 53  Personnel d’encadrement : 08  Personnel technique : 15 |
| Chiffres d’Affaires | 2014 : 1 255 243 307 F CFA  2013 : 740 788 489 FCFA  2012 : 1 225 687 359 FCFA |
| Domaines de compétence | Vente de matériel et logiciels informatiques  Maintenance et sous-traitance informatique  Management informatique  Sécurité informatique et électronique  Certifications internationales  Formations en informatique et management  Développement d’applications |
| Adresse : | **Douala** : 606, Blvd de la République B.P : 15685 Douala-Cameroun ; BP 15 685  **Yaoundé** : Face centre des impôts Biyemassi ; BP 14 828  **Garoua** : Bidemire, Centre commercial face Direction PMUC  **Bertoua** : Nkol-Bikon, Route Nationale N°1  **N’Djamena** : 2ème étage, immeuble ASIA HOTEL, Avenue Behagle Djambalbar |

1. **Vue Externe de SECEL**

On parlera principalement ici des clients, des concurrents et des fournisseurs de SECEL.

1. LES CLIENTS

SECEL dispose d’une multitude de clients qu’on peut regrouper en secteur :



1. LES PARTENAIRES DE SECEL

Suivant les produits et services qu’elle commercialise, SECEL entretient des relations privilégiées avec des partenaires éditeurs, chacun leaders dans leurs domaines dont les principaux sont représentés sur la figure suivante :

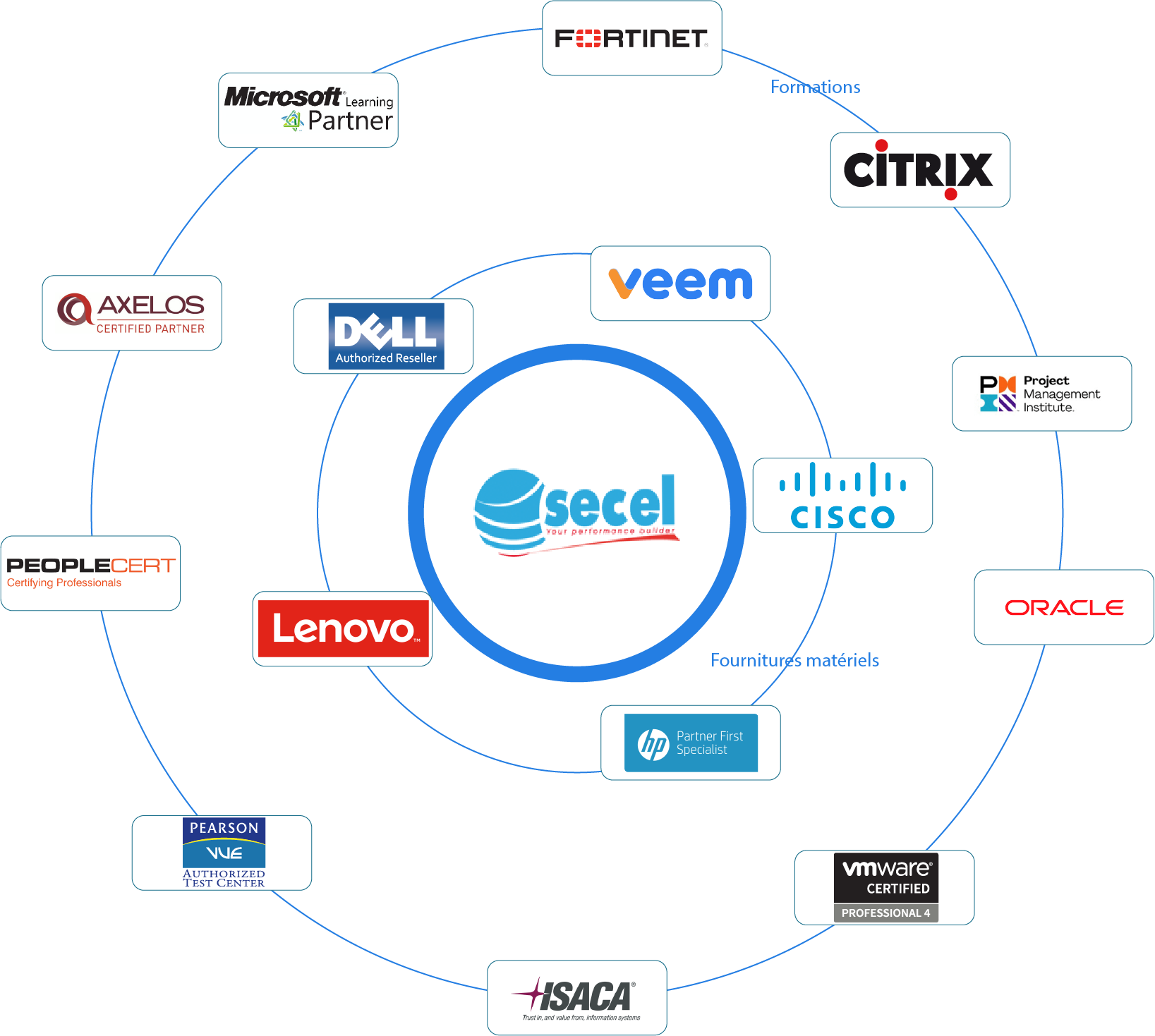


Figure 3 : Liste des Partenaires de SECEL

1. LES CONCURENTS

L’entreprise SECEL a de nombreux concurrents, principalement dans le domaine de la vente du matériel informatique. Parmi ces concurrents les plus influents, on a: WISE COMPUTERS, MIRACLE INFORMATIQUE, JR COMPUTER, COGEICOM, HARDIS, PANESS, France INFORMATIQUE, CFAO.

Maintenant qu’on a parlé de l’entreprise SECEL il sera important pour nous de présenter la manière dont le stage a été effectué.

RESUME

Le présent document expose de manière systématique le déroulement de notre stage au sein de Sofinez ou il nous a été pour travail la mise en place d’un système de réservations hôtelières pour les plateformes mobiles : cas d’ARENA PLAZA.

ARENA PLAZA est une société de service hôtelier qui dispose à son actif de plusieurs agences pour diversifier son activité. Consciente de l’apport des Technologies de l’information et de la Communication dans le domaine professionnel, elle souscrit à un service externe pour vulgariser son activité tant sur le plan national qu’internationale.

Cette expérience motivante et riche en découverte, nous a permis de nous initier aux nouvelles technologies du web et du mobile, et nous a offert l’opportunité de faire un nouveau pas en avant vers l’entrée dans la vie active et dans les métiers IT, que nous retrouverons à la fin de notre formation.

ABSTRACT

This document systematically explains the course of our internship at Sofinez where we worked on the implementation of a hotel reservation system for mobile platforms : case of ARENA PLAZA.

ARENA PLAZA is a hotel service company which has to its credit several agencies to diversify its activity. Aware of the contribution of Information and Communication Technologies in the professional field, it subscribes to an external service to popularize its activity both nationally and internationally.

This motivating experience, rich in discovery, allowed us to be introduced to the new web and mobile technologies, and gave us the opportunity to take a new step towards entering the working life and IT professions, which we will find again at the end of our training.

TABLE DES MATIERES

[DEDICACE I](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839599)

[REMERCIEMENT II](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839600)

[PRESENTATION DE L’ENTREPRISE III](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839601)

[RESUME V](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839602)

[ABSTRACT VII](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839603)

[SOMMAIRE VIII](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839604)

[LISTES DES TABLEAUX XII](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839605)

[LISTES DES FIGURES XIII](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839606)

[LISTES DES ABREVIATIONS XIV](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839607)

[AVANT-PROPOS XV](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839608)

[INTRODUCTION GENERALE 1](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839609)

[PREMIERE PARTIE : ETUDE DE L’ENVIRONNEMENT DE SOFINEZ 2](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839610)

[CHAPITRE 1 : PRESENTATION DE L’ENTREPRISE 3](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839611)

[SECTION 1 : ENVIRONNEMENT INTERNE 4](#_Toc108839612)

[I. HISTORIQUE 4](#_Toc108839613)

[II. MISSIONS DE SOFINEZ 4](#_Toc108839614)

[III. ACTIVITES ET PERSPECTIVES 5](#_Toc108839615)

[A. Activités 5](#_Toc108839616)

[B. Perspectives 5](#_Toc108839617)

[IV. IV. ORGANISATION, FICHE D’IDENTITE ET ORGANIGRAMME 5](#_Toc108839618)

[A. Organisation 5](#_Toc108839619)

[a. Direction générale 5](#_Toc108839620)

[b. Direction des Ressources Humaines 5](#_Toc108839621)

[c. Département Recherche et développement 5](#_Toc108839622)

[d. Département marketing 6](#_Toc108839623)

[B. Fiche d’identité 6](#_Toc108839624)

[C. Organigramme 6](#_Toc108839625)

[SECTION 2 : ENVIRONNEMENT EXTERNE 7](#_Toc108839626)

[V. I. FOURNISSEURS 7](#_Toc108839627)

[VI. II. LES CLIENTS 7](#_Toc108839628)

[VII. III. LA CONCURRENCE 7](#_Toc108839629)

[VIII. 7](#_Toc108839630)

[CHAPITRE2 : ACTIVITE DE STAGE ET PROBLEMATIQUE 8](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839631)

[IX. 9](#_Toc108839632)

[SECTION 1 : ETUDE DE L’EXISTANT 9](#_Toc108839633)

[I. ACTIVITES EN STAGE 9](#_Toc108839634)

[X. II. ETUDE DE L’EXISTANT 9](#_Toc108839635)

[A. Etude d’opportunité 9](#_Toc108839636)

[B. Etude de faisabilité 9](#_Toc108839637)

[C. Description de l’existant 9](#_Toc108839638)

[D. Critique de l’existant 10](#_Toc108839639)

[1. Sur le plan organisationnel 10](#_Toc108839640)

[2. Sur le plan technique 10](#_Toc108839641)

[XI. SECTION 2 : DESCRIPTION DU BESOIN 11](#_Toc108839642)

[XII. 11](#_Toc108839643)

[I. EXTRAIT DU CAHIER DES CHARGES 11](#_Toc108839644)

[A. Présentation générale du projet 11](#_Toc108839645)

[B. Objectifs du projet 11](#_Toc108839646)

[C. Rôle des parties prenantes 12](#_Toc108839647)

[D. Expression des besoins 13](#_Toc108839648)

[1. Les besoins fonctionnels 13](#_Toc108839649)

[2. Besoins non fonctionnelles 14](#_Toc108839650)

[a. Une navigation intuitive et fluide 14](#_Toc108839651)

[b. Le design 14](#_Toc108839652)

[c. La richesse et la qualité de l'information disponible 15](#_Toc108839653)

[d. La fraîcheur de l'information disponible 15](#_Toc108839654)

[e. La réactivité de l’application 15](#_Toc108839655)

[f. L'interactivité de l’application 15](#_Toc108839656)

[g. La visibilité de l’application sur les Stores 16](#_Toc108839657)

[E. Les contraintes du projet 16](#_Toc108839658)

[1. Contraintes d’exploitation 16](#_Toc108839659)

[2. Contrainte de sécurité 16](#_Toc108839660)

[3. Contrainte technique 16](#_Toc108839661)

[F. Déroulement du projet 17](#_Toc108839662)

[a. Plan d'assurance qualité 19](#_Toc108839663)

[b. Documentation 19](#_Toc108839664)

[c. Responsabilités 19](#_Toc108839665)

[DEUXIEME PARTIE : MODELISATION DU SYSTEME FUTUR 20](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839666)

[CHAPITRE 3 : ANALYSE ET CONCEPTION 21](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839667)

[XIII. SECTION 1 : ANALYSE FONCTIONNELLE 22](#_Toc108839668)

[I. Les Acteurs 22](#_Toc108839669)

[II. LES CAS D’UTILISATION 23](#_Toc108839670)

[A. Objectif d’un cas d’utilisation 23](#_Toc108839671)

[B. Identification des Cas d’utilisation 24](#_Toc108839672)

[SECTION 2 : ANALYSE DYNAMIQUE : DIAGRAMME DE SEQUENCE 27](#_Toc108839673)

[XIV. I. LES CONCEPTS DU DIAGRAMME DE SEQUENCE 27](#_Toc108839674)

[XV. II. QUELQUES DIAGRAMMES DE SEQUENCE 28](#_Toc108839675)

[i. Créer un compte 28](#_Toc108839676)

[ii. Se connecter 28](#_Toc108839677)

[iii. Réserver un logement 29](#_Toc108839678)

[SECTION 3 : ANALYSE STATIQUE : DIAGRAMME DE CLASSE 30](#_Toc108839679)

[I. CONCEPTS 30](#_Toc108839680)

[II. DIAGRAMME DE CLASSE DU SYSTEME 32](#_Toc108839681)

[SECTION 4 : CONCEPTION DES DONNEES 33](#_Toc108839682)

[I. MODELE CONCEPTUELLE DES DONNEES 33](#_Toc108839683)

[II. MODELE LOGIQUE DES DONNEES RELATIONNELLES 34](#_Toc108839684)

[III. MODELE PHYSIQUE DES DONNEES 36](#_Toc108839685)

[IV. ARCHITECTURE DE L’APPLICATION 39](#_Toc108839686)

[CHAPITRE 4 : IMPLEMENTATION ET TEST 41](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839687)

[SECTION 1 : IMPLEMENTATION 42](#_Toc108839688)

[I. OUTILS UTILISES 42](#_Toc108839689)

[II. LANGAGES ET TECHNOLOGIES 43](#_Toc108839690)

[III. BASE DE DONNEES 43](#_Toc108839691)

[SECTION 2 : TEST 44](#_Toc108839692)

[CONCLUSION 45](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839693)

[REFERENCES BIBLIOGRAPHIQUES 46](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839694)

[WEBOGRAPHIES 47](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839695)

[TABLE DES MATIERES 48](file:///E:\me\Cours%20Master%202\Projet%20de%20Soutenance\OK\Memoire%20de%20fin%20d'étude.docx#_Toc108839696)

LISTES DES TABLEAUX

[Tableau 1:: Fiche d'identité de Sofinez 5](#_Toc51659437)

[**Tableau 2:Découpage du projet** 15](#_Toc51659438)

[Tableau 3:Tableau descriptif de quelques cas d'utilisation 21](#_Toc51659439)

[Tableau 4:Quelques concepts du diagramme de classes 24](#_Toc51659440)

LISTES DES FIGURES

[Figure 1:Organigramme de Sofinez 5](#_Toc51659412)

[Figure 2: formalisme d'un utilisateur 19](file:///C:\Users\ngami_4vnxadk\Documents\PROJET_STAGE\debut.docx#_Toc51659413)

[Figure 3:formalisme d'un cas d'utilisation 19](file:///C:\Users\ngami_4vnxadk\Documents\PROJET_STAGE\debut.docx#_Toc51659414)

[Figure 4: Diagramme de Cas d'utilisation du système 21](#_Toc51659415)

[Figure 5:Diagramme de séquence :créer un compte 22](#_Toc51659416)

[Figure 6: Diagramme séquence: se connecter 23](#_Toc51659417)

[Figure 7:Diagramme de séquence: Réserver un logement 23](#_Toc51659418)

[Figure 8:Diagramme de Classes du système 25](#_Toc51659419)

[Figure 9: Modèle conceptuelle des Données 26](#_Toc51659420)

[Figure 10:Modèle Physique des Données 29](#_Toc51659421)

[Figure 11: Logos des outils utilisés pour l’implémentation de la solutions 31](#_Toc51659422)

[Figure 12: logos des technologies utilisés pour developper notre solution 32](#_Toc51659423)

LISTES DES ABREVIATIONS

**BLL** : Business Logic Layer

**CIF** : Contrainte d’Intégrité Fonctionnelle

**CIT** : Contrainte d’Intégrité Temporelle

**DAL** : Data Access Layer

**iOS** : iPhone Operating System

**IoT** :

**MCD** : Modèle Conceptuelle des Données

**MLDR**: Modèle Logique des Données Relationnelles

**No SQL** : Base de Données Non Relationnelles

**Sd** : Sequence Diagramm

**UML** : Unified Modeling Language

**UIL** : User Interface Layer

**SDK** : Software Developpnment Kit

INTRODUCTION GENERALE

L'évolution rapide de l'automatisation et le développement rapide de la numérisation sont des statuts auxquels toute entreprise qui se dit sereine doit s'habituer, car il est nécessaire d’entretenir l’information, d’assurer son acheminement et son traitement afin de pouvoir cerner les besoins des utilisateurs. De nos jours, de nombreux hôtels proposent des services de réservation en ligne, que ce soit en passant par Booking.com, ou via leurs propres sites web. Il devient utile de disposer d’un service de réservation en ligne, limitant les opérations entre le client et la structure hôtelière, de même que les opérations manuelles liées à l’enregistrement des réservations.

C'est dans ce vent de modernité que PREMIUM HOTEL, une société de services hôteliers, souhaite automatiser la gestion des réservations sur son site internet.

**Objectif du projet** :

***Premium Hôtel*** est présent sur internet via son site Web <https://premiumhotelgroup.com/> et ***Booking.com,*** dispose aussi d’un système de gestion en locale (**HOTIS**), souhaite automatiser les réservations en ligne sur via son site web.

L’objectif de ce projet est de « **mettre en place un système de réservation en ligne intégrée au système d'information de l'entreprise : cas de Premium Hôtel** » afin d’avoir en temps réel au sein d’un même système les informations liées aux réservations. Cette solution aura pour but de permettre à **PH** d’offrir un service de réservation et de paiement en ligne pour sa clientèle, mais aussi le report en temps réel des données au sein de son système d’information (**HOTIS**).

Le problème qui se pose ici est de savoir **comment faire communiquer Les différents systèmes (site web, système en locale) de Premium Hôtel, de manière sécurisée et d’avoir des données centralisées et en temps réel** ?

**Approche de solution :**

La réalisation de cette solution débute par la compréhension des besoins et attentes utilisateurs en ce qui concerne les réservations. Également la compréhension de l’utilisation du système de réservation existant jusqu’à l’implémentation et le déploiement d’une solution de réservation en ligne. Les principales activités de notre approche sont :

* Le recueil des besoins et attentes des gestionnaires de PH ;
* L’analyse de la proposition de solution à l’aide de diagrammes UML ;
* La conception de l’architecture technique et logicielle, des modèles, de la base de données et des fonctionnalités de notre solution ;
* L’implémentation de cette solution en 03 composants (un service de réservation via le site web, une application web pour gérer les réservations et la liaison avec le site web, une API pour régir la communication entre l’application web et le système en local pour la visualisation des données en temps réel).

Afin de gérer notre projet, nous avons utilisé la méthodologie SCRUM, qui est un standard de gestion de projet de développement logiciel. Nous avons également utilisé les meilleures pratiques de l’industrie pour l’ingénierie logicielle et les recommandations de sécurité.

Notre travail sera subdivisé en trois grandes parties :

* **Partie 1 :** dans laquelle nous allons présenter le contexte du projet et la revue de littérature.
* **Partie 2** **:** qui s’articulera autour du matériel et des méthodes utilisées pour l’étude et l’analyse de la solution envisagée.
* **Partie 3 :** il sera question pour nous de présenter d’une part la conception, l’implémentation et test de la solution mise au point et d’autre part faire une conclusion générale sur le travail effectué.

Dans cette partie, nous allons faire une bref description des généralités des système d’information et l’intégration des SI.

PREMIERE PARTIE : REVUE DE LITTERATURE

Pour parvenir à offrir à **PH** une solution de réservation en ligne en temps réel et répondant à ses besoins, il est capital pour nous de bien comprendre les concepts théoriques liés au projet. Ainsi, nous aborderons les notions comme système d’information, intégration de solution, API, paiement en ligne, Bases de données relationnelles….

1. **Généralités sur les Systèmes d’information**
2. **Système d’information : Définitions**

Si la notion de **système d’information** (SI) semble moderne dans sa formulation, elle n’en est pas moins ancienne dans sa conception. La notion de systèmes d’information, au sens de ***Reix*** (2004) combine de nombreux systèmes, de la simple collection d’informations aux traitements et autres croissements les plus élaborés. Des systèmes comptables de l’Antiquité (sumériens, phéniciens, codex étrusques, liber…) aux systèmes de *data mining* les plus performants et contemporains, la nécessité de collecter et d’agencer des données nait, comme le rappelle ***Degos*** (1998, p.53) à propos des systèmes comptables, de « la complexité de la langue et la rusticité des moyens d’écriture, la socialisation des comportements et la pertinence des règles juridiques, la rigueur de l’encadrement des fonctionnaires aux formateurs ». ***Reix*** et ***Rowe*** (2002) définissent les SI comme « un ensemble d’acteurs sociaux qui mémorisent et transforment des représentations via des technologies de l’information et des modes opératoires ». ***Reix*** (2004) offre une définition encore plus large et intégratrice des systèmes d’information à la fois technique et organisationnelle. Il les intègre comme « un ensemble organisé de ressources (matériel, humaine, financière, etc…) permettant d’acquérir, de traiter, de stocker des informations (données) dans et entre des organisations ».

D’un autre côté, ***Noy*** et ***Ruiz*** (2007, p.90) insistent sur la difficulté à définir le concept même de SI. Ils l’assimilent à un système de communication. « Le système d’information, même pour les spécialistes du domaine est un concept difficile à appréhender et à décrire. Les uns parlent plutôt du système des informations c’est-à-dire de l’ensemble des informations qu’il faut posséder pour fonctionner et décider, les autres se focalisent plus sur les moyens matériels (machines, connexions, logiciels) à mettre en œuvre pour traiter, stocker…, les informations, d’autres encore privilèges les aspects organisationnels ou humains dans ces systèmes. Le caractère à la fois diffus au sein de l’organisation et englobant tout ou partie des activités de celle-ci, rend le concept de SI difficile à cerner. »

Ces différentes définitions des systèmes d’information contribuent donc à décrire essentiellement des systèmes permettant de collecter, de traiter, de stocker de l’information, afin de pouvoir utiliser cette agrégation à des fins utiles.

1. **Finalité et rôle d’un SI**

Le système d’information tient une importance capitale dans le processus stratégique de toute entreprise. C’est en effet lui qui permet de réaliser toutes les actions liées à la gestion des données : collecte, stockage, traitement et partage… autant d’étapes qu’il rend possible à travers un ensemble de ressources à la fois matérielles, humaines et technologiques. Détenant à la fois un rôle opérationnel et stratégique, le système d’information, au-delà de sa fonction « opérationnelle », dispose d’une véritable dimension sociale en permettant aux différents acteurs d’une entreprise de disposer de l’information dont ils disposent en temps réel, au bon format et diffusable auprès de tous. Lorsqu’il est correctement exploité, le système d’information représente à la fois un gain de temps et d’argent, en rendant automatique des processus jusque-là effectués manuellement, mais il optimise également la performance commerciale en améliorant la relation client.

Le système d’information d’une entreprise est souvent considéré comme sa « colonne vertébrale », à la fois performante et évolutive. C’est pourquoi tout ce qui touche de près ou de loin à sa conception doit faire l’objet d’une réflexion stratégique en amont du lancement du projet.

1. **Les fonctions d’un SI**
   * + 1. **Collecter l’information**

Pour toute organisation, l’information est précieuse et indispensable pour sa **pérennité. Colleter l’information consiste à l’enregistrer afin de procéder à son traitement.**

* + - 1. **Mémoriser l’information**

Une fois collectée et saisie, l'information doit être **stockée** de manière **durable, stable** et **sécurisée** afin de pouvoir être ultérieurement utilisée ou tout simplement pour répondre à des obligations légales.

Les informations sont stockées dans des fichiers, ou soit dans des bases de données.

* + - 1. **Traiter l’information**

Le traitement de l’information consiste à produire de nouvelles informations à partir d’informations existantes grâce à des programmes informatiques ou des opérations manuelles.

Le traitement de l’information peut prendre 4 formes différentes :

* **Consulter l’information** : il s’agit du traitement le plus simple puisqu’il consiste à accéder à l’information telle qu’elle a été enregistrée ;
* **Organiser l’information** : ce traitement consiste à structurer l’information selon des critères spécifiques. Cela peut-être par exemple regrouper l’information par client, par zones géographiques, par activités et bien d’autres encore ;
* **Mettre à jour l’information** : ce traitement va consister à reprendre une information précédemment enregistrée et à l’actualiser ;
* **Produire de nouvelles informations** : à partir d’information(s) existante(s), ce traitement va permettre la création de nouvelles informations.
  + - 1. **Diffuser l’information**

Quel que soit son origine ou ce qu’elle représente, une information n’a de valeur que si elle est communiquée au bon destinataires, au bon moment et sous une forme directement exploitable.

1. **Principaux types de système d’information**

**1. Système de traitement des transactions**

Une transaction englobe tous les achats et ventes de produits et de services, ainsi que toutes les transactions commerciales quotidiennes ou les activités nécessaires à l'exploitation d'une entreprise. Un système de traitement des transactions garantit que toutes les données contractuelles, transactionnelles et relatives à la relation client sont stockées dans un endroit sûr et accessibles à tous ceux qui en ont besoin. Il aide également au traitement des écritures de commande client, de la paie, de l'expédition, de la gestion des ventes ou d'autres transactions courantes nécessaires a la maintenance des opérations.

**2. Système de bureautique**

Un système de bureautique est un réseau de divers outils, technologies et personnes necessaires pour effectuer des taches de bureau et de gestion. Il aide à améliorer la communication entre les différents services afin que chacun puisse collaborer pour accomplir une tâche.

**3. Système de gestion des connaissances.**

Un système de gestion des connaissances stocke et extrait de l'information pour aider les utilisateurs a améliorer leurs connaissances et à optimiser les efforts de collaboration pour accomplir les tâches.  Ce système d’information fournit également un accès intuitif a l’information externe requise par les travailleurs qui ont besoin de connaissances externes pour remplir leur rôle.

4. **Système d’information de gestion**

Un système d'information de gestion utilise diverses données sur les transactions provenant d'un TPS pour aider les cadres intermédiaires à optimiser la planification et la prise de décisions. La plupart des formats de rapport comprennent des résumes des données annuelles sur les ventes, des données de performance ou des enregistrements historiques. Cela fournit aux gestionnaires un moyen sur et systemise d'atteindre leurs objectifs et de superviser les unités opérationnelles.

**5. Système d’aide à la décision**

Un systeme d'aide a la decision traite les donnees pour faciliter la prise de decisions par la direction. Il stocke et rassemble l'information requise pour que la direction prenne les mesures appropriees au bon moment. Les modeles decisionnels sont programmes dans le SI afin d'analyser et de resumer de grandes quantites d'information et de les placer dans un visuel qui les rend comprehensibles.

**6. Système de soutien à la direction**

**Ils sont** utilisés par les dirigeants et les propriétaires pour optimiser la prise de décisions.

1. **Composant d’un SI**

Le domaine des systèmes d'information et de communication a certes une forte composante technologique et [informatique](https://fr.wikipedia.org/wiki/Informatique). Mais c'est seulement un aspect de ce domaine qui est en fait beaucoup plus vaste. Il s'agit de concevoir comment circule et est stockée l'information de façon efficace et cohérente pour toutes les activités d'une entreprise, d'un réseau d'entreprises, d'une administration publique, des relations entre entreprises, des citoyens, des gouvernements, etc.

Ainsi peuvent être inclus comme composants dans un système d’information en vue d’offrir des caractéristiques techniques ou fonctionnalités spécifiques :

* Applications métier ;
* [Bases de données](https://fr.wikipedia.org/wiki/Base_de_donn%C3%A9es) de l'entreprise,
* Contrôle d'accès,
* Dispositifs de [sécurité](https://fr.wikipedia.org/wiki/S%C3%A9curit%C3%A9_des_syst%C3%A8mes_d%27information),
* [Infrastructure réseau](https://fr.wikipedia.org/wiki/R%C3%A9seau_(informatique)),
* [Postes de travail informatique](https://fr.wikipedia.org/wiki/Poste_de_travail),
* Accès aux réseaux [Internet](https://fr.wikipedia.org/wiki/Internet), [intranet](https://fr.wikipedia.org/wiki/Intranet) ou [extranet](https://fr.wikipedia.org/wiki/Extranet),
* [Serveurs d'application](https://fr.wikipedia.org/wiki/Serveur_d%27applications),
* Serveurs de [données](https://fr.wikipedia.org/wiki/Donn%C3%A9e) et systèmes de [stockage](https://fr.wikipedia.org/wiki/Sauvegarde),
* Système de paiement électronique,
* Système de sécurité (protection et [chiffrement](https://fr.wikipedia.org/wiki/Chiffrement)),
* Outils de [groupware](https://fr.wikipedia.org/wiki/Groupware), agendas, espace de partage de documents, échange d'informations (forums électroniques), gestion de contacts, conférence électronique (chat, vidéoconférence).

1. **Généralités sur l’intégration des système d’information**
2. **Définition :**

L’**intégration de système** consiste à réunir au sein d’un même système d’information, des parties développées de façon séparées

**II. Présentation de l’intégration des Système d’information**

L'***intégration informatique***, ou ***intégration des systèmes***, désigne le fait de connecter les données, les applications, les API et les appareils au sein d’un service informatique, dans le but d'augmenter l'efficacité, la productivité et l'agilité d’une entreprise. Elle permet à tous les éléments d'un environnement informatique de fonctionner ensemble. Ainsi, elle représente un aspect clé de la transformation d'une entreprise, c'est-à-dire de son adaptation face à l'évolution du marché.

L'intégration ne se limite pas à connecter des éléments, elle ajoute également de la valeur. En effet, en connectant les différentes fonctions de plusieurs systèmes, elle peut offrir de nouvelles fonctionnalités.

**NB : Intégration système différent d’intégration continue**

***L’intégration continue***est unepratique employée par les développeurs, consistant à fusionner des versions de codes qui fonctionnent sur un référentiel central partagé plusieurs fois par jour.

L'objectif de l'intégration continue est d'automatiser la création de versions et la vérification afin de détecter rapidement les éventuelles erreurs et d'accélérer le développement.

1. **Historique de l’intégration des SI**

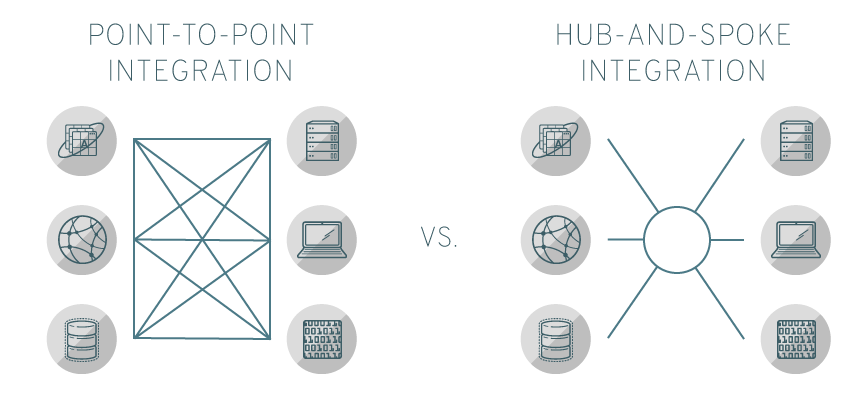
Au fil du temps, les systèmes informatiques grossissaient et se développaient, ils s’éloignaient les uns des autres. Les solutions provenant de fournisseurs différents ne pouvaient plus communiquer entre elles. Au final, les éléments d’une pile informatique n'étaient plus connectés que par le simple fait qu'ils appartenaient à une organisation. Il fallait donc trouver une solution pour organiser l’environnement et éviter la répétition des tâches, surtout lorsqu'il s'agit de mettre en œuvre et de gérer une logique métier.

**1. les solutions d’intégration des SI**

**a. L’intégration d’application d’entreprise**

Il existe une solution pour rassembler et organiser tous ces éléments disparates **: l'intégration d'applications d'entreprise** (EAI), c'est-à-dire un ensemble de technologies, des outils et un Framework qui permettent une intégration en temps réel et basée sur les messages entre les différentes applications. L'envoi de ces messages est déclenché par des modifications ou des paramètres au sein de chaque application.

Il existe deux modèles d'EAI : l'intégration point à point (« ***point-to-point*** ») et l'intégration en étoile (« ***hub-and-spoke*** »).



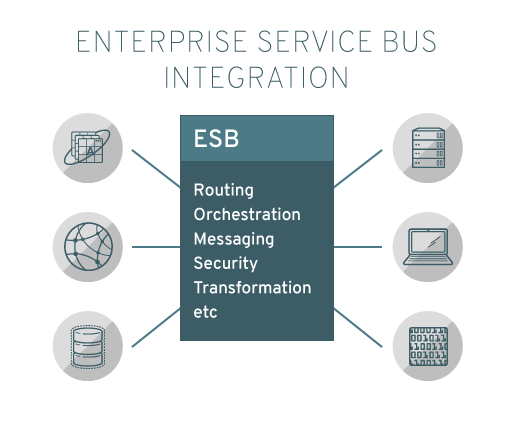
*Point-to-point vs Hub-and-spoke integration*

* **Le modèle point à point :** ici, chaque application (ou ressource) est personnalisée pour pouvoir communiquer (fonctionner) avec les autres applications et éléments de l’environnement informatique. Il s'agit d'une tâche fastidieuse et le modèle est par conséquent fortement sujet aux erreurs. Pour ne rien simplifier, la maintenance du modèle se complexifie à chaque mise à jour de l'infrastructure et des applications.
* **Le modèle étoile :** Il résout le problème du modèle point à point avec un point de connexion central (le cœur) qui relie toutes les applications et les services. Les rayons qui relient le cœur aux applications et aux services peuvent faire l'objet d'une maintenance individuelle. Ainsi, il est possible de concevoir des applications plus spécialisées et réserver les tâches d'intégration au cœur et aux rayons. Le principal désavantage de cette approche réside dans la centralisation du cœur, car il devient le point unique de défaillance du système et des communications au sein de votre infrastructure. Dans un modèle en étoile, toutes les intégrations dépendent, par définition, du bon fonctionnement du cœur.

**b. Entreprise Service Bus**

**ESB** pour *Enterprise Service Bus* est un outil d'abstraction orienté messages qui fournit des modules de service entre les applications.

L'ESB tient également le rôle de point central où tous les modules de service sont partagés, redirigés et organisés afin de connecter les applications et données entre elles. Cette solution remplace avantageusement l'intégration en étoile, mais ce n'est pas non plus la panacée, surtout lorsque l’entreprise croît et acquiert de nouvelles ressources, car elle requiert alors une plus grande rapidité au niveau des propriétés et des ressources logicielles.



*ESB intégration*

De manière structurelle, un ESB ressemble fort au modèle d'intégration en étoile. Mais l'ESB propose tout de même quelques fonctionnalités très spécifiques.

* Un ESB est un service qui utilise des normes ouvertes. Ainsi, pas besoin d'écrire des interfaces uniques pour chaque application.
* Les services d'intégration peuvent être déployés sans modifier fondamentalement les applications.
* Les ESB s'appuient sur des interfaces et des protocoles ouverts et standard, afin de simplifier les nouveaux déploiements.

Toutefois, un déploiement d'ESB implique souvent une architecture centralisée, et ce, pour les mêmes raisons évidentes déjà avancées avec le modèle en étoile : il faut un emplacement unique pour héberger et contrôler tous les services d'intégration. Malheureusement, les déploiements et architectures d'ESB s'accompagnent d'une gouvernance centrale rigide qui ne permet pas de fournir les solutions plus rapides et capables de s'adapter qui sont à la base des initiatives de [transformation numérique](https://www.redhat.com/fr/topics/digital-transformation/what-is-digital-transformation). De plus, les ESB finissent souvent par se changer en applications monolithiques.

**c. L’intégration agile**

Aujourd'hui, les architectures d'applications **cloud-native**, faiblement couplées et développées selon des méthodes [agiles](https://www.redhat.com/fr/devops/what-is-agile-methodology) et des principes DevOps, exigent une approche évolutive et tout aussi agile de l'intégration. L'intégration agile (Selon *Red Hat*) est une stratégie de connexion des ressources qui allie des technologies d'intégration, des techniques de distribution agile et des plateformes natives pour le cloud dans le but d'accélérer la distribution des logiciels tout en renforçant la sécurité. En pratique, l'approche consiste à déployer des technologies d'intégration (des API, par exemple) dans des conteneurs Linux, et à répartir la gestion de l'intégration entre des équipes pluridisciplinaires. Une architecture d'intégration agile peut être divisée en trois capacités clés : **l'intégration distribuée**, les [**conteneurs**](https://www.redhat.com/fr/topics/containers)et les **API**.

|  |  |  |
| --- | --- | --- |
|  | **Caractéristiques** | **Avantages** |
| **Intégration distribuée** | * Encombrement réduit * Basée sur des modèles * Orientée événements * Communautaire | **FLEXIBILITÉ** |
| **Les conteneurs** | * Cloud-native * Simples et déployables individuellement * Évolutifs et hautement disponibles | **ÉVOLUTIVITÉ** |
| **Les API (Application Programming Interface)** | Points de terminaison bien définis, réutilisables et bien gérésInfluence et utilisation de l'écosystème | **RÉUTILISATION** |

*Tableau : capacités d’une Architecture agile*

En outre ses différents types d’intégration de SI, on distingue des architectures logicielles permettant de faire communiquer plusieurs sous-systèmes ensemble. Notamment :

* L’architecture Orienté évènement (**EDA**) ;
* L’architecture Orienté services (**SOA**) ;

**2. Les Architectures logicielles d’intégration de SI**

**a. Architecture orienté évènements**

**L'architecture orientée événements** (de l'anglais *event driven architecture*, ou **EDA**) est une forme d'[architecture de médiation](https://fr.wikipedia.org/wiki/Architecture_de_m%C3%A9diation) qui est un modèle d'interaction applicative mettant en œuvre des services (composants [logiciels](https://fr.wikipedia.org/wiki/Logiciels)) répondant à des sollicitations externes :

* avec une forte cohérence interne (par l'utilisation d'un format d'échange pivot, le plus souvent [XML](https://fr.wikipedia.org/wiki/XML)),
* Et des couplages externes lâches (par l'utilisation d'événements)

Par opposition à l'[architecture orientée services](https://fr.wikipedia.org/wiki/Architecture_orient%C3%A9e_services) (SOA) où un « fournisseur » rend un service à la demande d'un consommateur; en architecture EDA, un [composant](https://fr.wikipedia.org/wiki/Composant_logiciel) prévient par émission d'un événement qu'il a réalisé une opération donnée. C'est aux Clients potentiels de traiter cet événement.

Les architectures EDA ont été popularisées avec l'apparition de standards pour les places de marchés et les systèmes de vente aux enchères. Elles mettent en application une partie des principes d'[urbanisation](https://fr.wikipedia.org/wiki/Urbanisation_(informatique)). Une architecture orientée événements repose principalement sur un bus disposant de fonctionnalités d'abonnement et de publication (*publish and Subscribe*).

**i. les concepts de l’architecture orienté évènements**

Le couplage entre composants est un couplage lâche et les communications sont en général [asynchrones](https://fr.wikipedia.org/wiki/Asynchronisme).

*Le service peut :*

* Être codé dans n'importe quel langage,
* S’exécuter sur n'importe quelle plateforme (matérielle et logicielle).

*Le service doit :*

* S’abonner aux événements qu'il souhaite traiter
* Traiter les événements auxquels il est abonné sans préjuger d'un quelconque ordre et émettre un événement compte rendu de l'action qu'il vient de réaliser
* Fournir les événements qu'il est susceptible d'émettre dont les structures sont publiées,
* Être autonome (disposer de toutes les informations nécessaires à son exécution : pas de notion d'état)
* Respecter un ensemble de contrats (règles de fonctionnement).

**ii. Un évènement**

Un **événement** désigne tout phénomène ou changement d'état significatif au niveau du matériel ou d'un logiciel système. Il ne faut pas confondre un événement et une notification d'événement, c'est-à-dire une notification ou un message envoyé par le système pour signaler à une autre partie du système qu'un événement s'est produit.

Les événements peuvent être causés par des actions internes ou externes. Ils peuvent être provoqués par des utilisateurs (clics de souris ou frappe sur le clavier, par exemple), provenir d'une source externe (un capteur) ou être générés par le système (lors du chargement d'un programme, par exemple).

Les consommateurs d'un événement doivent s'abonner (*subscribe*) à un intermédiaire de gestion d'événement (le bus) et le producteur de l'événement doit le publier auprès de ce gestionnaire (le bus). Quand le gestionnaire d'événement reçoit un événement d'un producteur, il le diffuse (*forward*) aux consommateurs concernés. Si le consommateur est injoignable, le gestionnaire peut conserver le message et le diffuser ultérieurement. Ce moyen de transmission d'événements repose sur un bus de message *store and forward*.

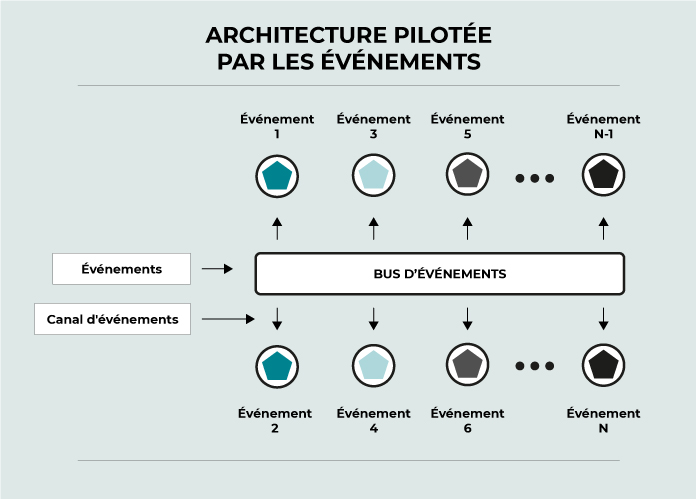
**iii.** ***Fonctionnement de l’architecture orienté évènements***

Ce type d'architecture implique des **producteurs** et des **consommateurs** d'événements. Un producteur d'événements détecte ou reconnaît un événement et le représente sous forme de message. Il ignore quels seront les consommateurs et les conséquences de chaque événement.

Lorsqu'un événement a été détecté, il est transmis du producteur d'événements au consommateur via des canaux d'événement, où une plateforme de traitement les prend en charge de façon asynchrone. Les consommateurs d'événements doivent être informés lorsqu'un événement se produit. Ils peuvent traiter l'événement ou être seulement affectés par ce dernier.

La plateforme de traitement des événements exécute la réponse adaptée à chaque événement et envoie l'activité en aval aux consommateurs concernés. Cette activité permet de visualiser le résultat d'un événement.

Il existe d'autres gestionnaires d'événements de type middleware qui peuvent servir de plateforme de traitement des événements.



*Architecture Orientée évènements*

*iv.*  ***Modèles d'architectures orientées événements***

Une architecture orientée événements peut être basée sur un modèle de publication/abonnement ou sur un modèle de flux d'événements.

#### **Modèle de publication/abonnement**

Ce modèle est une infrastructure de messagerie basée sur des abonnements à des flux d'événements. Lorsqu'il est utilisé, chaque fois qu'un événement se produit ou est publié, il est envoyé aux abonnés qui doivent en être informés.

#### **Modèle de flux d'événements**

Avec un modèle de flux d'événements, les événements sont enregistrés dans un journal. Au lieu d'être abonnés à un flux d'événements, les consommateurs d'événements peuvent accéder à n'importe quelle partie du flux et le rejoindre à tout moment.

Il existe différents types de flux d'événements :

|  |  |
| --- | --- |
| **Flux d’évènements** | **Description** |
| ***Le traitement des flux d’évènements*** | S’appuie sur une plateforme de diffusion de données en continu, telle qu'Apache Kafka, pour ingérer des événements et des processus, voire transformer le flux d'événements. Le traitement des flux d'événements peut permettre d'y détecter des tendances importantes. |
| ***Le traitement des évènements simples*** | correspond aux cas où un événement déclenche immédiatement une action au niveau du consommateur d'événements. |
| ***Le traitement des évènements complexes*** | oblige un consommateur d'événements à traiter une série d'événements afin de détecter des schémas récurrents. |

*Les flux d’évènements*

*v.* **Avantage de l’architecture orienté évènements**

La programmation orientée événements présente plusieurs avantages, notamment :

* **Des services faiblement connectés**: les créateurs et les consommateurs d’événements ne sont pas connectés. Le créateur ne sait pas quels consommateurs attendent les événements qu’il produit. Il envoie simplement l’événement à un routeur, où les consommateurs pourront trouver les types d’événements auxquels ils sont abonnés.
* **Évolutivité**: comme les créateurs et les consommateurs ne se connaissent pas, il n’y a pas d’intégration directe. Il est beaucoup plus facile d’ajouter de nouveaux services dans le système et d’éviter les pannes au cours de cette opération.
* **Développement agile** : comme il est facile d’ajouter de nouveaux services aux systèmes EDA, le développement s’intègre bien dans un [environnement agile](https://www.lucidchart.com/blog/fr/release-planning-agile). Le routeur d’événements est le médiateur entre les services qui filtre et envoie les événements aux consommateurs.
* **Réduction des coûts** : les systèmes EDA sont basés sur le principe du « push », à la demande. Le système n’interroge pas continuellement les serveurs. La consultation et le traitement à la demande requièrent moins de ressources réseau, ce qui réduit les coûts globaux.
* **Réponse et analyses en temps réel** : les applications réagissent aux changements en temps réel à l’aide des données actuelles et historiques pour prendre des décisions. Ce type de réponse aide le système à prévoir et à éliminer les menaces de sécurité à la volée, en automatisant et en optimisant les chaînes d’approvisionnement.

**vi. Pourquoi utiliser l’architecture orienté évènements**

Lorsqu’on souhaite que divers systèmes ou sous-système soient interopérables, l’utilisation d’un EDA est utile pour répondre aux besoins suivants :

|  |  |
| --- | --- |
| **Besoins** | **Explications** |
| **Réplication des données entre comptes et entre régions** | Les routeurs d’événements peuvent transférer des données vers différents systèmes situés dans différentes régions et différents comptes. Cette approche permet de garantir la synchronisation des systèmes tout en laissant vos équipes développer et déployer de nouvelles applications et de nouveaux services indépendamment des autres équipes. |
| **Répartition et traitement parallèle** | Utilise l’architecture EDA pour « répartir » un événement unique entre plusieurs systèmes qui doivent y répondre, ce qui évite à vos équipes de devoir développer un code personnalisé pour transmettre l’événement à chaque consommateur. Chaque système peut traiter l’événement en même temps à des fins différentes. |
| **Surveillance et avertissements de l’état des ressources** | Les architectures EDA peuvent surveiller les ressources et envoyer des alertes en cas de changement d’état, d’anomalies et de mise à jour des composants. |
| **Intégration de systèmes hétérogènes** | Utilise des routeurs d’événements pour partager des informations entre des systèmes utilisant différentes piles technologiques. |

*Tableau : Besoins d’utilisation d’un EDA*

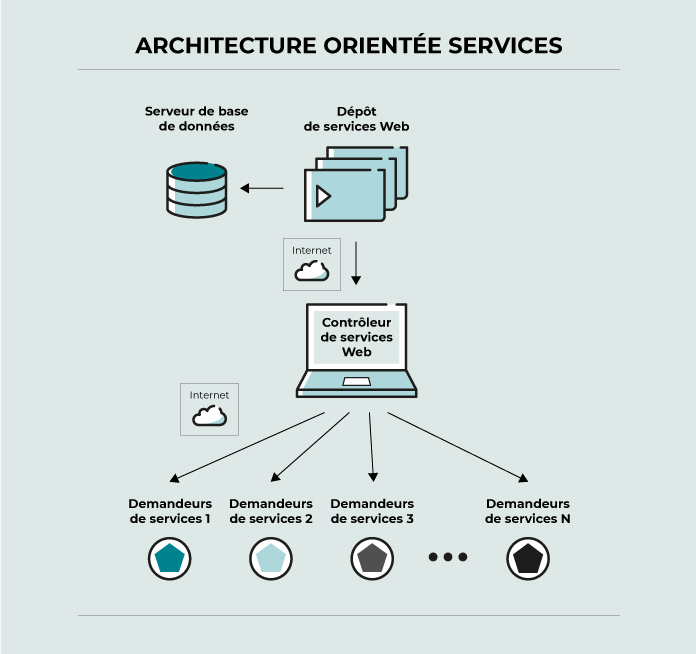
**b. Architecture orienté Services**

**i. Définition**

L'architecture orientée services (ou SOA, *Service-Oriented Architecture*) est un modèle de conception qui rend des composants logiciels réutilisables, grâce à des interfaces de services qui utilisent un langage commun pour communiquer via un réseau.

Un **service** est une unité autonome de fonctionnalité logicielle, ou d'un ensemble de fonctionnalités, conçue pour réaliser une tâche précise comme récupérer des informations ou exécuter une opération. Il contient les intégrations de code et de données nécessaires pour exécuter une fonction métier distincte et complète. On peut y accéder à distance, et interagir avec lui ou le mettre à jour de manière indépendante.

**ii. Comment fonctionne SOA ?**



*Architecture SOA*

Au sein d'une architecture orientée services, les services communiquent via un système de « couplage faible ». Les composants (aussi appelés « éléments ») sont interconnectés dans un système ou réseau, ce qui leur permet de transmettre des informations ou de coordonner un processus métier, en réduisant la dépendance entre eux.

L’architecture Orienté services est structuré en quatre parties :

* **Le dépôt de services web (Web service repository) :**

Bibliothèque de services web conçue pour répondre à des demandes d'informations externes. L'information fournie est généralement un petit élément, comme un numéro, un mot, quelques variables, … Elle est généralement documentée de manière très détaillée, car des applications externes font appel aux fonctions qu'elle contient.

* **Le contrôleur de services web (Web service controller) :**

Ce module communique les informations contenues dans le dépôt de services web aux demandeurs de services. Lorsqu'un demandeur de service externe appelle une certaine fonction du dépôt de services web, le contrôleur de services web interprète la demande et recherche la fonction dans le dépôt de services web. Il exécute ensuite cette fonction et renvoie une valeur au demandeur.

* **Le serveur de base de données (Database Server) :**

Il contient les tables, les index et les données gérés par l'application. Les recherches et les opérations d'insertion/suppression/mise à jour y sont exécutées.

* **Les demandeurs de services (Service Requester) :**

Ceux sont des applications externes qui demandent des services au dépôt de services web par l'intermédiaire d'Internet, comme une organisation demandant des informations sur les vols à une compagnie aérienne, ou une autre entreprise demandant à un transporteur la localisation d'un colis à un moment donné.

**iii. Avantages par rapport à une approche monolithique**

* **Mise sur le marché accélérée et plus grande flexibilité :**le caractère réutilisable des services facilite et accélère le regroupement des applications. Les développeurs ne doivent plus repartir de zéro à chaque fois, comme c'est le cas pour les applications monolithiques.
* **Utilisation de l'infrastructure existante sur les nouveaux marchés :**grâce à l'architecture SOA, les développeurs peuvent plus facilement étendre et mettre à l'échelle les fonctionnalités d'une plateforme ou d'un environnement.
* **Coûts réduits grâce à une meilleure agilité et à un développement plus efficace**
* **Maintenance facilitée :**les services étant autonomes et indépendants, ils peuvent être modifiés et mis à jour autant que nécessaire, sans affecter les autres services.
* **Évolutivité :**comme l'architecture SOA s'adapte à plusieurs services, plateformes et langages de programmation, l'évolutivité est considérablement accrue.
* **Fiabilité améliorée :**il est plus facile de déboguer des petits services plutôt qu'un long code, ce qui permet de créer des applications plus fiables.
* **Grande disponibilité :**les fonctions de l'architecture SOA sont à la portée de tous.

**iv. Inconvenients de SOA**

* Les services web peuvent représenter une faiblesse au niveau du site pour les pirates informatiques qui veulent obstruer le système. Certaines formes d'attaques sont des « dénis de service ». Elles consistent à demander le même service web des millions de fois par seconde, jusqu'à ce que le serveur tombe en panne.
* Le propriétaire du service web aide d'autres sites, mais reçoit une petite rémunération pour ce faire.

**v. Quand utiliser SOA**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom de l'exemple** | **Suivi de colis** | **Informations sur les vols** | **Convertisseur de devises** |
| **Définition** | En fonction d’un numéro de suivi, on décrit l'itinéraire et la localisation actuelle d'un colis commandé par un client. | Un service web accepte en entrée un numéro de vol. Il renvoie les données relatives au vol : heure de départ, heure d'arrivée prévue, lieu de départ, lieu d'arrivée, altitude actuelle, etc. | Un service accepte une devise source et donne sa valeur dans une devise de destination (par exemple, « combien de dollars américains font un franc »). |
| **Exemple concret** | * FedEx shipping Integration (fedex.com). | Flight Explorer Professional Edition (flightexplorer.com) | Toutes les grandes banques en disposent. |
| **Avantages** | Les sites de vente au détail qui utilisent ces services web n’accèdent pas aux systèmes internes de FedEx ou de DHL. | Les sites de voyage, les hôtels et les entreprises de logistique qui utilisent ce produit peuvent disposer d'informations immédiates sur un vol par Internet sans avoir à accéder aux systèmes internes. | Les transactions monétaires nécessitent les taux de change officiels à un instant donné. Ce service web de conversion de devises publie toutes les conversions de devises fournies par la Banque centrale. |

*Tableau : Exemples d’utilisation de l’architecture SOA*

vi. **Architecture SOA ou microservices ?**

Le concept de services introduit par l'architecture orientée services est aujourd'hui un élément essentiel du [cloud computing](https://www.redhat.com/fr/topics/cloud) et de la [virtualisation](https://www.redhat.com/fr/topics/virtualization) dans les solutions de type middleware et microservices.

On confond souvent l'architecture SOA et microservices en raison de leurs similitudes. Toutefois, la principale caractéristique qui les distingue est leur portée : l'architecture SOA est *une approche à l'échelle de l'entreprise*, tandis que les microservices sont *une stratégie de mise en œuvre au sein des équipes de développement des applications*.

Ils ne communiquent pas non plus de la même façon avec leurs composants. L'architecture SOA utilise un ESB, alors que les microservices peuvent communiquer entre eux sans état, à l'aide d'[API](https://www.redhat.com/fr/topics/api) indépendantes de tout langage, ce qui permet aux équipes de développement de choisir leurs outils. Ainsi, les microservices offrent ***plus de tolérance et de flexibilité***.

Il arrive également que l'architecture SOA soit confondue avec le modèle [SaaS](https://www.redhat.com/fr/topics/cloud-computing/what-is-saas) (Software-as-a-Service). Le SaaS est une forme de cloud computing qui permet de fournir une application cloud aux utilisateurs, avec ses plateformes et son infrastructure sous-jacentes. Dans une architecture SOA, les services web sont parfois distribués sous forme d'applications SaaS. En général, un fournisseur de services cloud (par exemple AWS, Azure ou IBM Cloud) s'occupe de la gestion de l'environnement cloud sur lequel l'application SaaS est hébergée. Les utilisateurs interagissent avec le logiciel par le biais d'un navigateur web sur leur ordinateur ou leurs appareils mobiles. Ils peuvent également le connecter à d'autres fonctions à l'aide d'API telles que **REST** ou **SOAP**.

1. **Les enjeux liés à l’intégration des Systèmes d’information**

L’intégration des systèmes en informatique figure aujourd’hui parmi les principaux enjeux des entreprises. En effet, dans un contexte où les données, les processus et les acteurs sont en perpétuelle évolution, il est parfois bien compliqué de parvenir à « tout faire marcher » ensemble.

Cette difficulté à réunir des éléments très hétérogènes et ne partageant bien souvent que très peu de standards, (en termes de langage de programmation, de format ou encore de [système d’exploitation](https://www.lesjeudis.com/technologies/syst%C3%A8me-d'exploitation) par exemple) met en évidence des problématiques de type technologique, économique et organisationnel auxquelles les entreprises doivent faire face. Le volume de données de plus en plus important ainsi que les [exigences en matière de confidentialité et de sécurité](https://blog.lesjeudis.com/rgpd-la-protection-des-donnees-evolue-en-2018-etes-vous-prets) transforment un peu plus la question de l’intégration des systèmes en un véritable défi.

L’intégration des systèmes permet, à terme, de répondre à différents besoins formulés par les entreprises :

* **Obtenir des processus automatiques** et donc réduire les erreurs pouvant être effectuées lors des processus « manuels » (actions de la part des utilisateurs pour faire converger les différentes applications entre elles) ;
* **Consolider les données**en étant capable de croiser différents types de données entre elles : en liant les données entre elles et de ce fait, l’entreprise a plus de visibilité en obtenant facilement des statistiques globales ;
* **Gagner du temps et une valeur ajoutée** en regroupant l’ensemble des informations sur un seul et même support (au lieu d’avoir à saisir ces informations sur chacune des applications).

C’est pourquoi, davantage que des enjeux techniques, ce sont bien des [enjeux stratégiques qui se cachent derrière tout projet d’intégration](https://www.supralog.com/technologie-integration-et-enjeux).

Pour l’entreprise, il s’agit de trouver la meilleure équation possible entre les besoins liés à son métier et les capacités logicielles qui sont mises à son service. La réussite d’un tel projet réside en effet en la capacité d’adopter une vision globale tout en faisant intervenir des leviers complémentaires.

Dans cette partie, nous présenterons la démarche que nous avons adoptée dans le cadre de notre travail afin d’arriver à ressortir la solution proposée, ainsi que le matériel utilisé.

Tout d’abord, nous débuterons par l’étude stratégique et fonctionnelle, au cours de laquelle nous avons effectué les activités de compréhension des besoins et attentes des parties prenantes, d’analyse des spécifications du projet, de conception et d’implémentation de la solution à mettre en œuvre.

Et enfin, nous ferons une bref la présentation de la méthodologie agile de gestion des projets informatiques qui a orienté notre projet.

DEUXIEME PARTIE : MATERIEL ET METHODES

1. **Etude Stratégique et Fonctionnelle de l’environnement existant**

Dans le but de proposé une solution logicielle utile pour PH, il est convient tout d’abord de s’attarder sur le contexte et le fonctionnement actuel du processus de gestions des réservations en ligne, sur la vision stratégique de l’entreprise, sur les difficultés et les besoins des utilisateurs dans le but d’avoir une compréhension plus détaillée du système existant et du système futur, afin de pouvoir proposer une solution qui sera alignée sur les objectifs de l’entreprise et qui permettra de créer une valeur ajoutée aux utilisateurs. En outre, il s’agit pour nous ici de ressortir les objectifs SMART recherchés par l’entreprise vis-à-vis de ce projet, et de ressortir sous forme de spécification fonctionnelle ou non fonctionnelle l’ensemble des besoins et exigences des parties prenantes.

1. **Etude de l’existant**

**Etude d’opportunité**

Elle consiste à recenser les influences liées au problème et à mettre en place une solution plus efficace. Ainsi les principaux problèmes rencontrés sont :

* **L’indisponibilité** de certains clients à se rendre dans une agence hôtelière et d’effectué une réservation, il doit parfois faire recours à un tiers et son besoin n’est pas satisfait ;
* Le nombre infinie d’opérations manuelles que doit effectuer un gestionnaire de l’hôtel pour enregistrer une réservation effectuée en ligne.

**Etude de faisabilité**

La mise en place de cette solution informatique aura pour objectif de permettre aux clients de réserver le logement qu’il souhaite, d’avoir des offres promotionnelles et surtout de prévoir, programmer à l’avance un séjour au sein de PREMIUM Hôtel, mais aussi de facilité l’accès en temps réel des données liées au réservations.

**Description de l’existant**

*Premium Hôtel*, dispose :

* D’un système de gestion en locale ;
* D’un **site internet** de présentation de ces services et permettant d’effectuer une réservation : Suite à une reservation faite par un client, une notification par mail est envoyée au gestionnaire d’hôtel pour l’informer qu’une nouvelle réservation a été émise. Suite à cela, le gestionnaire renseigne au niveau du système de gestion en locale la nouvelle reservation et relance le client pour le paiement de la réservation.
* D’un espace sur **Booking.com** pour être présent à l’international et faciliter les réservations et les paiements en ligne : Ici, le client peut effectuer une réservation et la payer directement (via carte de crédit). Lorsqu’une réservation est faite, une notification par mail est envoyée au gestionnaire de l’hôtel, ce dernier se rend sur l’espace admin de PH sur Booking.com pour avoir les détails de la réservation, il renseigne ces informations au niveau du système de gestion en local, il contacte le client pour lui confirmer que sa réservation a été bien prise en compte et enregistrée.

Nous avons remarqué, que ces différents systèmes ne communiquent pas entre elle, permettant ainsi l’accès aux données en temps réel. Il n’existe donc pas de software permettant de gérer de manière centralisée et chronophage les réservations effectuées en ligne.

**Critique de l’existant**

La gestion des réservations effectué en ligne se fait de façon manuelle, ce qui ne permet pas d’avoir en temps réel au niveau du système d’information, des informations sur les réservations effectuées en ligne.

L’analyse du système actuelle nous a permis de relever les limites suivantes :

**i. Sur le plan organisationnel**

* La perte de temps ;
* Un nombre infinie d’opération dans le processus de collecte des réservations effectué en ligne.
* La lente dans la recherche d’une information précise ;
* Non prise en compte des spécificités du besoin des clients ;
* La mauvaise gestion des réservations effectués en ligne ;

**ii.** **Sur le plan technique**

* La lenteur dans les traitements des informations des réservations effectuées en ligne ;
* La quasi-impossibilité d’établir des statistiques fiables sur les réservations effectuées par un client en ligne, du fait de la proportion importante des données à traiter ;

1. **Description du besoin**

**Extrait du cahier de charges**

**a. présentation générale du projet**

***Premium Hôtel*** est un établissement commercial qui offre des services hôteliers payant à une clientèle de passage, notamment :

* Service d’hébergement ;
* Service traiteur ;
* Service évènementiel ;
* Service de transport ;

***Premium Hôtel*** étant déjà présent sur internet via son site Web <https://premiumhotelgroup.com/> et ***Booking.com*** souhaite faciliter les réservations en ligne sur via son site web. C’est dans cette optique que se situe notre projet : **mise en place d’un service de réservation en ligne intégrée au système d’information de l’entreprise : cas de *Premium Hôtel.***

* 1. **Objectif du projet**

L’objectif de ce projet est de mettre en place un service de reservation en ligne, qui pourra faciliter la reservation d’une chambre en ligne et le paiement. Cette application devra :

* Gérer les réservations en ligne ;
* Faciliter les paiements en ligne ;
* Assurer le report des réservations faites en ligne au niveau du système de gestion en locale ;
  1. **Motivation du projet**

Les principales motivations de la mise au point de ce projet sont :

* Automatiser le recueil des réservations faites en ligne ;
* Faciliter le paiement en ligne via une monnaie électronique (Orange Money) ;
* Avoir un récapitulatif des évaluations faites en ligne ;
  1. **Organisation du projet**
     1. **Comité de pilotage**

Le projet est piloté par le directeur général de Secel qui en est le commanditaire.

* + 1. **La maitrise d’œuvre**

**La maitrise d’œuvre est constituée de :**

* Ousmanou, Chef de projet ;
* Ngami Michelle, Software developper
  + 1. **La maitrise d’ouvrage**

La Maitrise d’ouvrage est constituée des parties prenantes suivante :

* Le directeur général de Secel ;
* Le(s) gestionnaire(s) de PH (La réceptionniste, Directeur administratif, …) ;
* Clients ou clients potentiel de PH ;

**Rôle de l’équipe de développement** :

L’équipe de développement devra :

* Proposer et respecter un échéancier de réalisation du projet en accord avec le client ;
* Concevoir, réaliser et mettre en place ledit système ;
* Assurer la maintenance et l’évolution de l’outil applicatif ;
* Assurer un support technique ;
* Respecter les délais de réalisation et de la mise en place du service ;

**2. Expressions des besoins**

* + - * 1. **Exigences et spécifications d’exigences : définition et présentation**

D’après le standard **IEE 610.12**; une *exigence* est :

* Une capacité ou une condition nécessaire à un utilisateur pour résoudre un problème ou atteindre un objectif ;
* Une condition ou une capacité que doit posséder un système afin de satisfaire aux termes d’un contrat, d’une norme, ou d’une spécification formellement imposée ;
* La représentation documentée de cette condition ou capacité.

L’élément important lorsqu’on aborde le concept d’exigence est la garantie que le système aide le client ou l’utilisateur à atteindre son objectif.

En outre, selon **IEEE 830**, une spécification d’exigences comprend :

* **Les fonctionnalités** : services et fonctions que le système doit fournir.
* **Les interfaces externes**: identification des interactions avec les utilisateurs et les autres systèmes avec lesquels le nouveau système doit s’intégrer.
* **La performance**: contrainte d’opération du système en disponibilité et en temps réponse.
* **Les attributs du système** : caractéristiques intrinsèques telles que la portabilité, l’exactitude, la maintenabilité, la sécurité, etc.
* **Les contraintes de conceptions** : les exigences sur la façon de developper le système.
  + - * 1. **Les types d’exigences**

On distingue deux types d’exigence :

* **Les exigences fonctionnelles** : Qui renseignent sur ce qui doit être fait, c’est-à-dire les fonctionnalités ;
* **Les exigences non fonctionnelles** : elles identifient les contraintes internes et externes du système basées sur des valeurs quantitatives. Il s’agit entre autres de :
* *L’utilisabilité* : Capacité pour un utilisateur d’’exécuter une tâche dans un temps donné après une formation d’une durée déterminée.
* *La performance*: Temps d’attente < 10s ;
* *La fiabilité*;
* *La disponibilité* : 24/7- sécurité : accès personnalisés, connexions sécurisées ;
* *La maintenabilité* : Modularité, commentaires, complexité, interfaces ;
* *La Portabilité* : utilisable sur plusieurs systèmes d’exploitation.
  + - * 1. **Livrables des activités de la phase approche de solution**

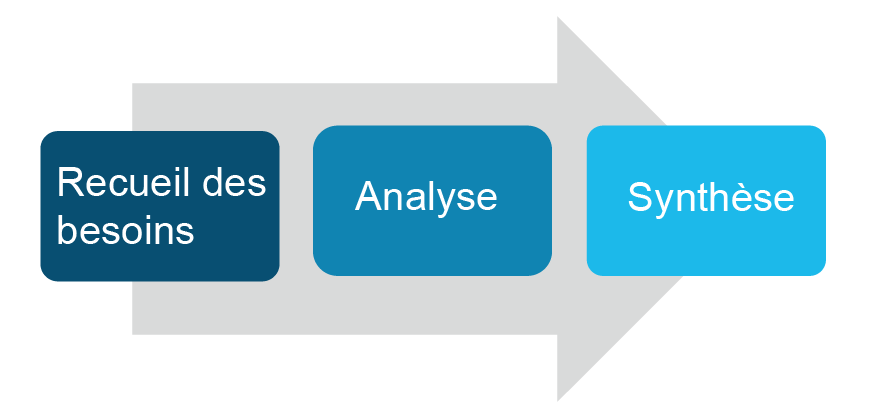
Les livrables attendus de cette phase de travail sont les suivantes :

* Cahier de charges validé par les parties prenantes du projet ;
* Document de spécification des besoins Clients ;
* Liste des fonctionnalités à implémenter ;
* Diagrammes UML de cas d’utilisation ainsi que les diagrammes de séquence ;
* Extraits des maquettes qui serviront d’IHM pour la solution logicielle ;
* Extrait de l’architecture applicative**;**
  + - * 1. **Approches d’analyse de l’existant**

Il est question pour nous ici d’effectuer une analyse de l’existant afin de ressortir les points clés de notre travail. Mais aussi, effectuer une analyse du système d’information existant afin d’assurer que le système qui va être développé et proposé corresponde bien aux attentes et exigences du demandeur.

Notre analyse se présente comme suit :

* **Le recueil des besoins** : collecte des informations et recherches dans le but de mieux comprendre le contexte et le problème qui se pose ;
* **L’analyse des besoins** : analyser les besoins, attentes et exigences des parties prenantes du projet et également les éléments pertinents résultants du Benchmarking sur le système en place
* **La phase de synthèse** : spécifier de façon bien détaillée chacun des besoins de manière plus formelle et utile pour le client, mais aussi pour l’équipe de développement.



*Démarche globale des étape d’analyse de la solution*

* 1. **Phase de recueil des besoins**

Tout au long de cette phase, la démarche consistait à :

* Recenser les fonctionnalités du système à mettre en place ;
* Recenser les solutions de réservation en ligne existantes ;
* Recenser les éléments d’Interface Homme Machine ;
* Recenser les besoins, attentes et exigences des utilisateurs ;
* Recenser les tâches utilisateur.

En ce qui concernent les types d’utilisateurs qui interagissent avec les sous-système suscités ; nous avons recensé :

* Les utilisateurs (les gestionnaires du l’hôtel) ;
* Les administrateurs

En ce qui concerne les besoins visés par chaque utilisateur, nous avons noté au travers des échanges que :

Les utilisateurs (gestionnaires) : ils souhaitent avoir une solution logicielle au leur permettra à partir d’un seul clic de renseigner une réservation effectuée en ligne ;

* Pour les administrateurs : la solution à mettre en place devra prendre en compte : la gestion des comptes utilisateurs, la gestion des droits d’accès.
  1. **La phase d’analyse**

Il s’agitici de ressortir les critiques positives et négative de la solution à mettre en place :

* L’absence de solution logicielle permettant de faciliter le processus de réservation en ligne ;
* Impossibilité d’effectuer un suivi des réservations effectué en ligne du fait de leur absence dans la base de données ;
* Comme ***critique positive :*** nous avons noté l’adhésion au projet et une volonté d’accompagner l’équipe de développement pour la mise en œuvre de la solution logicielle.
  1. **Phase de synthèse**

Le but de la phase de synthèse était de ressortir sous deux axes par rapport au système existant :

* **Les éléments à conserver :**  suite à l’analyse menée ci-dessus, le système de gestion en local actuellement disponible est celui à laquelle la solution développée sera intégrée
* **Les éléments à concevoir**: il sera question pour nous de concevoir les architectures logicielle et technique, la base de donnée sur laquelle devra s’appuyer la solution et implémenter une solution logicielle pour l’interaction entre les différents sous-systèmes**.**
  1. **Caractéristiques fonctionnelles du projet**
  2. **Au niveau du site web** :

Le site web de PH est présentement à caractère vitrine, il sera donc question pour nous d’ajouter les fonctionnalités suivantes :

* **Réservation d’une chambre ;**

Cette fonctionnalité permettra à un internaute de réserver une chambre pour une durée bien définie et en fonction de la disponibilité des chambres. Cette réservation sera disponible en temps réel dans le système interne de gestion hôtelier afin de rendre ladite chambre indisponible dans cette plage horaire.

* **Gestion des comptes clients ;**

Cette rubrique est consacrée à l’internaute qui se rend sur le site de l’hôtel pour effectuer une réservation. Ses informations de base à l’instar de son nom, numéro de téléphone et son email son pris pour des relances et notifications suite à une réservation. De plus un système de newsletter sera mis en place pour la notification des clients suite à une campagne publicitaire ou un évènement spécial.

* **Gestion des paiements :**

Cette fonctionnalité optionnelle permettra à un client de payer une réservation faite en ligne par un service de paiement (Orange Money, MOMO, PayPal, …). Ce n’est qu’après ce paiement que la chambre de la réservation sera Considérée comme occupée ;

ii. **L’application de gestion des réservations en ligne**

* **Le module de gestion des réservations en ligne devra permettre de :**
  + **Créer une réservation**, en outre modifier, annuler une réservation ;
  + Notifier l’administration lorsqu’une nouvelle réservation est effectuée via le site web ;
  + Assurer la liaison avec le système en local ;
  + Gérer les statistiques des réservations effectuées via le site web ;
* **Fonctionnalités de statistiques**
* Génération des statistiques globales paramétrables ;
* Journal de vente ;
* Audits ;
* Génération des rapports ;

**iii. Caractéristiques non fonctionnelles du projet**

* **Interface utilisateur**

L’interface de l’application (*web*) aura une ergonomique très moderne et un fonctionnement de type ‘’**cloud**’’. L’application sera accessible depuis un navigateur web et respectera une charte graphique préalablement établi.

* **Contraintes d’exploitation**

**L’application devra être accessible depuis n’importe quel navigateur web.**

Les administrateurs ou les gestionnaires doivent pouvoir avoir un accès à l’historique des opérations réalisées sur l’application. Une routine d’archivage de données devra être mise en place et une politique de sauvegarde devra être établie.

La gestion de l’application ne doit pas générer des coûts de fonctionnement supplémentaires.

* **Contraintes d’organisation**

La procédure de gestion de l’application se base sur la politique et les règlements établis par les gestionnaire ou le directeur de l’établissement. Cependant, cette application vise à réduire les enregistrements manuels des réservations faites en ligne et apporte une nouvelle solution de gestion qui prône l’automatisation tout en réduisant le temps de traitements de certaines tâches.

Il sera prévu la formation d’un agent de l’hôtel qui aura un rôle d’administrateur et devra suivre le cycle de vie de la gestion de l’application dans son intégralité *(accès à toutes les fonctionnalités).* Il aura aussi pour tâche de former à son tour le personnel de l’hôtel dans l’utilisation de l’application, définir et attribuer les rôles et les accès de ce personnel.

* **Contraintes de développement**

Le système qui sera mis en place disposera d’une architecture modulaire et de type client-serveur ; et présentera les caractéristiques suivantes :

* Faible usage du trafic réseaux
* Temps de latence réduit pour toutes les opérations, afin de fournir un niveau de performance acceptable
* Niveau de sécurité optimal : une sécurité à deux niveaux à savoir l’accès à l’application et l’accès aux données.
* Fournir la documentation utilisateur, d’installation et de maintenance
* **Qualités de la solution mise en place**
* Gérer le temps de réponse (5 seconde au plus pour une requête dans les conditions normales) ;
* Gérer la sécurité du logiciel (crypter les données transitant via l’url, crypter les données sensibles aux utilisateurs en base de données, protéger les formulaires des attaques par injection SQL et autres, le canal de communication entre la base de données locale et celle en ligne devra également être sécurisé) ;
* Gérer la robustesse (personnaliser les pages d’affichage des bugs) ;
* Gérer les utilisateurs et leurs droits d’accès à l’application (authentification et autorisation) ;
* Respecter la charte graphique de Premium Hôtel dans les différentes interfaces.
* L’application devra disposer d’une API pour récupérer les données du site web et d’une autre API pour fournir des données utiles au système de gestion en local.
* L’application devra être accessible via internet, via le smart phone Android de certains membres de Premium Hôtel.
* L’administrateur doit avoir accès à l’historique des opérations réalisées sur l’application via une interface d’administration.
* Une routine d’archivage des données devra être mise en place.
* Une routine de synchronisation devra être mise en place (Web services, réplication).
* Les technologies utilisées devront être optimales pour répondre aux besoins de temps de réponse et de rapidité d’exécution.

**iv. Critères d’acceptabilité du produit**

Après tous les tests effectués, la validation du présent document ainsi que tous les autres livrables se fera par un consensus avec le directeur de Secel.

L’application sera hébergée en ligne. Une documentation sur les fonctions de la solution est à prévoir.

* 1. **Les contraintes du projet** 
     1. **Contrainte d’exploitation**

L’application web sevra être déployée sur le cloud et administrée par SECEL. Les différents utilisateurs auront besoins d’avoir une connexion internet stable et fiable, pour bénéficier de la meilleure qualité de service.

* + 1. **Contrainte de sécurité**

La solution devra assurer la confidentialité, l’’intégrité et la disponibilité des données et informations qu’elle collecte, stocke, traite et diffuse. Le contrôle d’accès doit être réalisé suivant un système d’authentification basé sur les rôles.

* + 1. **Contraintes technique**

Les caractéristiques matérielles nécessaire aux supports et à la solution.

* **Serveur pour un hébergement cloud, VPS MTN**

|  |  |
| --- | --- |
| **Critères** | **Caractéristiques** |
| |  |  | | --- | --- | | Taille de la mémoire vive |  | | 8 giga |
| Capacité du disque dur | 100 giga |
| Traffic | 200 giga |

* **Poste Client**

|  |  |  |
| --- | --- | --- |
| **Critère** | **Caractéristique minimum** | **Caractéristiques conseillés** |
| Type de processeur | Dual Core | Quad Core |
| Taille de la mémoire vive | 2Go | 4Go |
| Capacité du Disque Dur | 50Go | 100Go |

* **Réseau**

|  |  |  |
| --- | --- | --- |
| **Critère** | **Caractéristique minimum** | **Caractéristiques conseillés** |
| Connexion Ethernet | Câbles à paires torsadés | Câbles à paires torsadés |
| Connexion Internet | Connexion ADSL | Connexion ADSL |

* + 1. **Contraintes logicielles**

Les caractéristiques logicielles nécessaires au support de la solution proposée sont les suivantes :

* **Poste Serveur pour un hébergement cloud :**

Tout poste client peut être utilisé pour jouer le rôle de poste serveur afin de se connecter à l’application.

* **Poste Serveur pour un hébergement local**

|  |  |  |
| --- | --- | --- |
| **Critère** | **Caractéristique minimum** | **Caractéristiques conseillés** |
| Système d’exploitation | Windows | Windows |
| Base de données | SQL Server / MySQL | SQL Server |
| Navigateur | Firefox ou Chrome, etc… | Firefox ou Chrome, etc… |

* **Poste client**

|  |  |  |
| --- | --- | --- |
| **Critère** | **Caractéristique minimum** | **Caractéristiques conseillés** |
| Système d’exploitation | Windows | Windows |
| Navigateur | Firefox ou Chrome, etc… | Firefox ou Chrome, etc… |

1. **Phase d’analyse**
2. **Outil : UML**
   * + 1. **Définition**

**UML :** Unified Modeling Language. Normalisé par l'OMG (Object Management Group) : <http://www.omg.org/spec/UML/>.

**Dernière version** : 2.4.1 (Août 2011)

**UML** est une notation standard pour la modélisation d'applications à base d'objets (et de composants depuis la version 2), Mais utilisable dans de nombreux autres contextes de conception ou spécification.

* + - 1. **Présentation**

UML hérite principalement des méthodes objets de Booch(Booch), OMT (Rumbaugh) et OOSE (Jacobson). Mais intègre également d'autres approches, comme les machines à états de Harel

**But initial** : Définir un processus/méthode de développement complet (de l'analyse à l'implémentation) orienté objet

* + - 1. **Principe**

**UML** favorise les processus :

* Itératif et incrémental ;
* Centré sur l’architecture ;
* Et piloté par les cas d’utilisation.
  + - 1. **Avantages et Inconvenients**
         1. **Avantages**
* Standardisation des concepts objet
* Indépendance vis-à-vis de tout langage de programmation
* Mise en oeuvre de toute la richesse de l’approche objet
* Couvre tous les aspects liés au développement logiciel
* Extensible vers la modélisation des processus
* Configurable pour des domaines autre que le logiciel
* Formalisme décrit avec un méta-modèle : MOF (Méta Model Facility)
  + - * 1. **Inconvenients**
* UML ne fournit aucun guide méthodologique. Il est nécessaire de définir une démarche rigoureuse d’utilisation des différents diagrammes ;
* La mise en pratique d’UML n’est pas intuitive. Elle suppose d’acquérir un véritable savoir-faire ;
* Les relations entre les différents diagrammes ne sont pas formalisées

1. **Méthodes**

**Analyse fonctionnelle**

Elle désigne une méthode d’organisation du bon sens, consistant à rechercher et à caractériser les fonctions offertes par un système pour satisfaire les besoins d’un utilisateur.

On distingue 2 types d’analyse fonctionnelle :

* L’analyse fonctionnelle externe en vue d’exprimer le besoin de l’utilisateur mettant en évidences les fonctions de services ou d’estime.
* L’analyse fonctionnelle interne qui exprime le point de vue du concepteur du système, mettant en évidences les fonctions techniques.

L’objectif de l’analyse est de définir de manière objective les besoins de l’utilisateur, de recenser toutes les fonctions que doit avoir un produit pour répondre à un besoin et elle s’utilise soit pour concevoir un produit soit pour reconcevoir des produits. Elle s’articule autour de 3 concepts :

* Les Acteurs ;
* Les Cas d’utilisation ;
* Le Diagramme de Cas d’utilisation ;

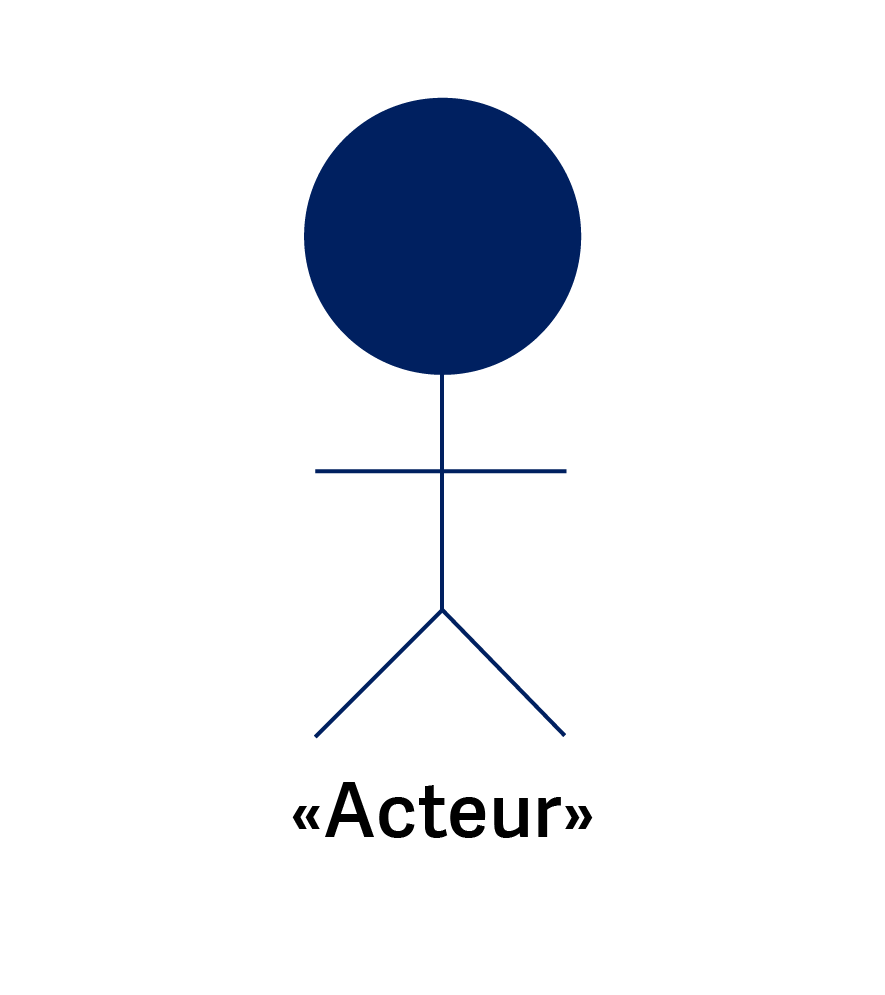
1. **Les Acteurs**

Un ***Acteur*** est une entité externe qui agit sur le système et jouant un rôle très important dans la délimitation du système du sujet modéliser. Un acteur n’est pas forcément une personne physique et est considéré comme externe aux périmètre d’étude.

21 Il représente un rôle joué par une entité qui interagir avec le système. On distingue deux types d’acteur :

* **Acteur principale** : participant externe interagissant directement avec le système
* **Acteur secondaire** : participant offrant un service qui contribue à la réalisation d’une action ;

***Formalisme d’un acteur en UML***



« Acteur »

**Dépendance entre Acteurs**

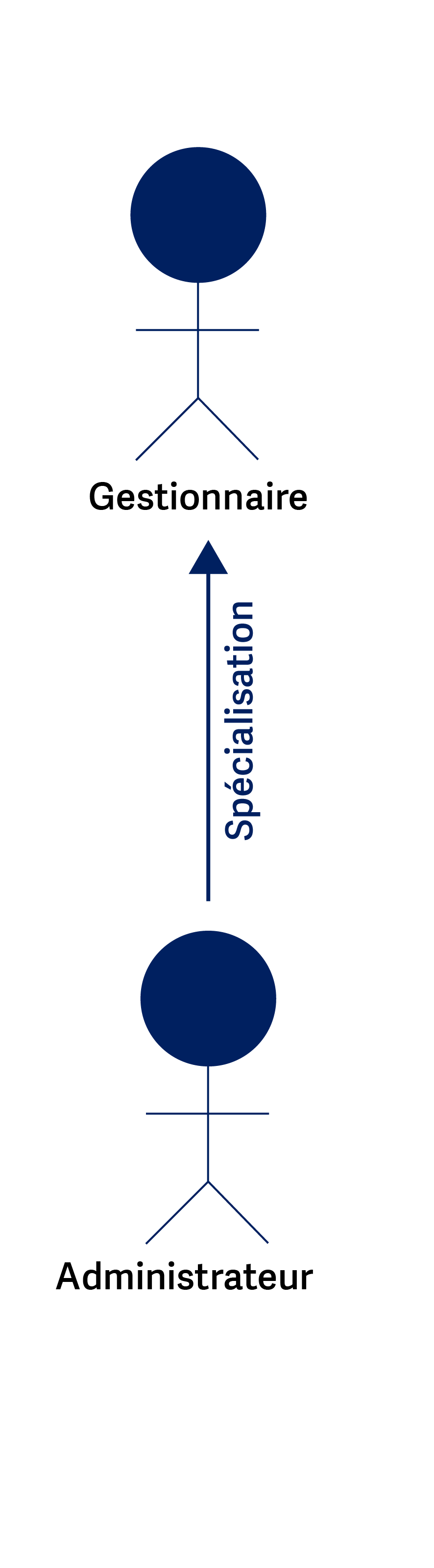
****

Figure 2: formalisme d'un utilisateur

Dans notre système à modéliser les acteurs sont :

* **Acteur secondaire** :
  + L’utilisateur (le client) ;
  + API de paiement ;
* **Acteurs Principaux** :
  + Le(s) gestionnaire(s) de l’hôtel : Gestionnaire ;
  + Administrateur (via l’application web) ;
  + Service web ;

1. **Les cas d’utilisation**

Les Cas d’Utilisation (USE CASE en anglais) est une représentation d’UML du modèle conceptuel, c’est l’image d’une fonctionnalité du système en réponse à une simulation de l’extérieur.

***Formalisme d’un cas d’utilisation***

Figure 3:formalisme d'un cas d'utilisation

**i. Objectif d’un cas d’utilisation**

Il a pour objectif de comprendre les besoins du client pour rédiger un cahier des charges fonctionnelle.

**Identification des cas d’utilisation**

Les différents cas d’utilisation gérer au sein de notre système sont :

* Un client peut :
  + Rechercher chambre ;
  + Consulter la liste des chambres ;
  + Réserver chambre ;
  + Payer réservation ;
* Un utilisateur (gestionnaire) peut :
  + Se connecter / déconnecter ;
  + Consulter notification ;
  + Effectuer réservation ;
  + Enregistrer une réservation ;
  + Etablir facture ;
  + Consulter les notifications ;
* Administrateur de l’application web doit pouvoir :
  + Gérer les comptes utilisateurs ;
  + Consulter l’historiques des opérations
  + Gérer les rôles et privilèges utilisateurs ;
  + Effectuer les différentes opérations qu’un gestionnaire peut faire.

Il s’assure de la mise à jour en temps réel des informations du système (disponibilité des chambres, notifications, système de paiement) ;

Autre cas d’utilisation :

* Emettre une notification (service web) ;
* Payer réservation (service de paiement) ;

**Description textuelle de quelque cas d’utilisation**

* **Effectuer une réservation**

|  |  |
| --- | --- |
| **Cas d’utilisation** | ***Effectuer une réservation*** |
| **Acteur principal** | Client |
| **Acteur secondaire** | Système (site web) |
| **Evènement déclencheur** | Rechercher une chambre |
| **Intérêt** | Avoir plus d’information sur une chambre et effectuer une réservation |
| **Préconditions** | Etre sur le site web et cliquer sur réserver |
| **Post conditions** | Chambre réserver |
| **Scenario nominal** | Le client se rend sur le site, il consulte les différentes catégories de chambre disponible, sélectionne une catégorie de chambre |
| **Extensions** | Un mail est envoyé au client pour le notifier de sa réservation, dont il souhaite avoir une facture proforma de sa réservation |
| **contraintes** | Il ne parvient pas à effectuer la réservation ;  Le paiement n’est pas possible |

* **Enregistrer une réservation**

|  |  |
| --- | --- |
| **Cas d’utilisation** | ***Enregistrer une réservation*** |
| **Acteur principal** | Gestionnaire |
| **Acteur secondaire** | Système (Application web) |
| **Evènement déclencheur** | Notification reçu |
| **Intérêt** | Enregistrer une réservation en base de données |
| **Préconditions** | Etre connecté à l’application web |
| **Post conditions** | Base de données mis à jour ( *réservation enregistrer*) |
| **Scenario nominal** | 1. le gestionnaire est connecté ;  2. il accède aux notifications ;  3. il sélectionne une notification, la consulte ;  4. Enregistre la réservation et met à jour la BD; |
| **Extensions** |  |
| **contraintes** | 1.a. Impossible de se connecter aller à 1 ;  2.a. Utilisateur non reconnu aller à 1 ;  2.b. aucune notification ;  4.a. impossible d’enregistrer la réservation, contacter le support technique ; |

1. **Diagrammes de cas d’utilisation**

Le diagramme de cas d’utilisation représente un ensemble de cas d’utilisation et d’acteurs et leurs relations. Il est utilisé pour modéliser la vue statique des cas d’utilisation d’un système, il est particulièrement important pour l’organisation et la modélisation des comportements d’un système.

Formalisme d’un diagramme de cas d’utilisation peut contenir des :

* Cas d’utilisation
* Acteurs
* Relations (de dépendance, généralisation, association).

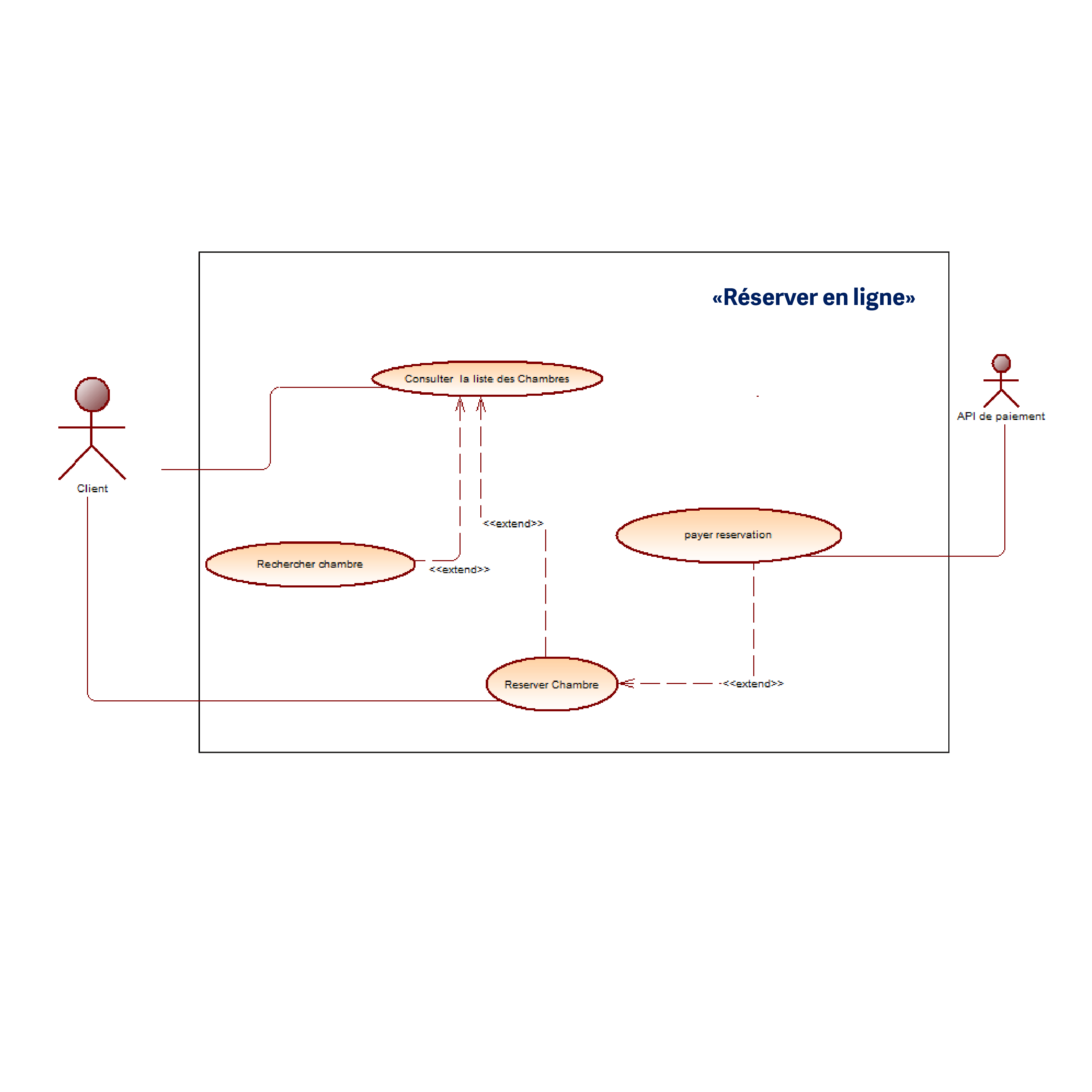
**Objectifs du DCU**

Il a pour objectif de :

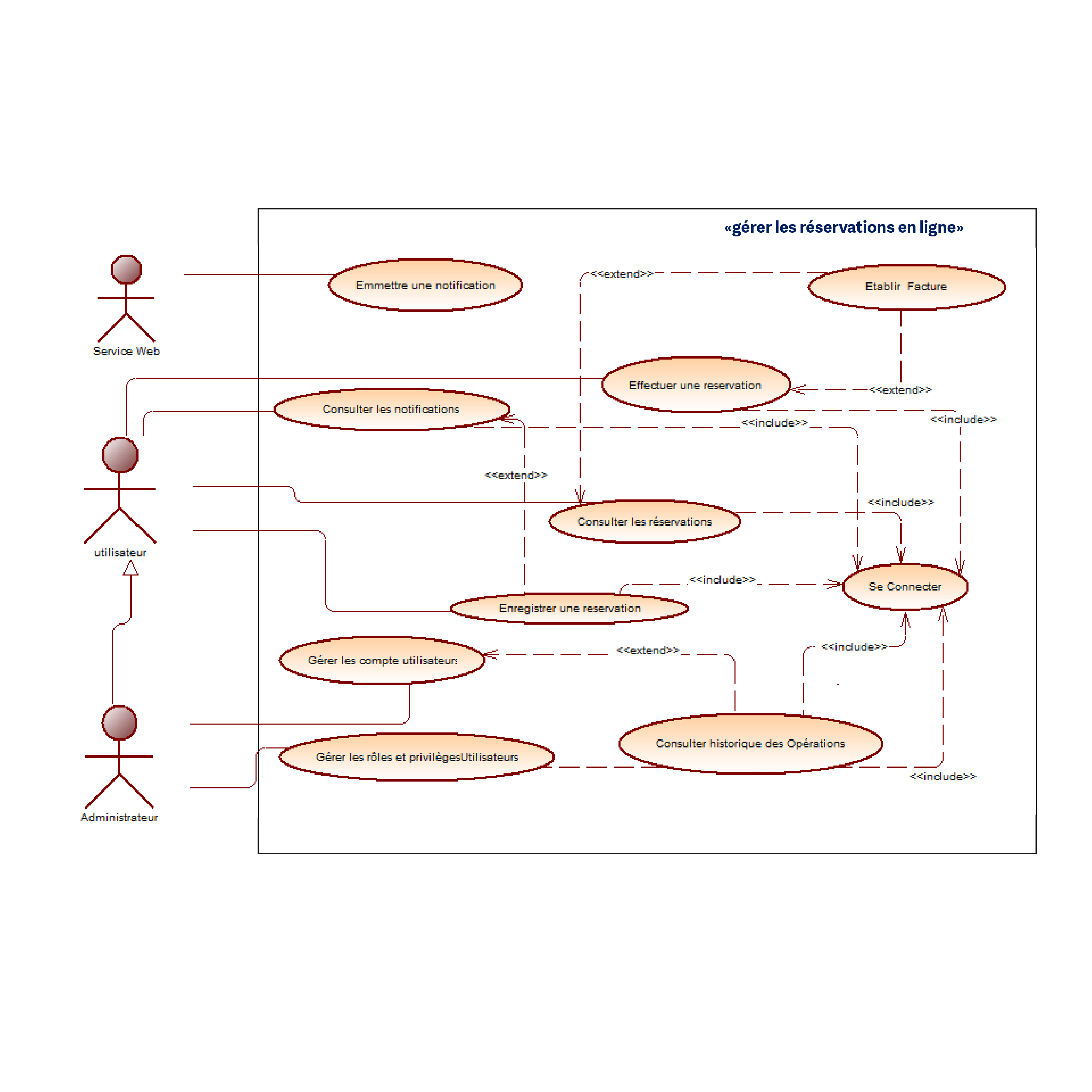
* Décrire la manière dont une organisation ou un système externe doivent interagir avec le système ;
* Décrit ce que doit faire le système ;
* Mettre en évidence les services rendu par le système.

**Représentation du DCU**

* **DCU au niveau du site web**

****

* **DCU au niveau de l’application web**

****

**Analyse dynamique : Diagramme de séquence**

Le diagramme de séquence fait partie des diagrammes comportementaux, plus précisément des diagramme d’interactions. Il permet de :

* Représenter les échanges entre les différents objets et acteurs du système en fonction du temps ;
* D’illustrer, de compléter le Diagramme de Cas d’Utilisation et pour un système donné on peut avoir plusieurs diagrammes de Séquence chacun correspondant à un cas d’utilisation ;

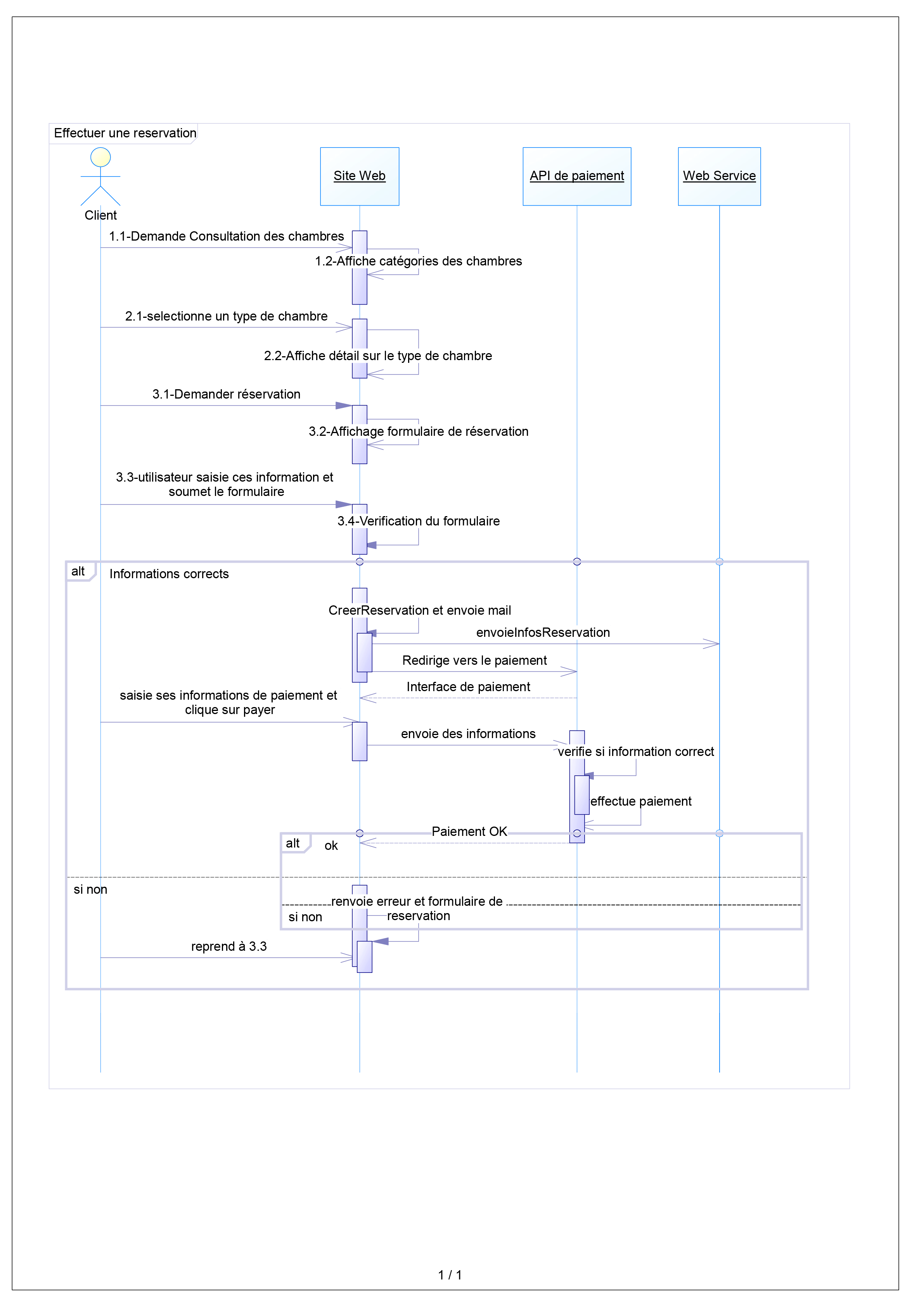
**a. Concepts du diagramme de séquence**

Les principaux concepts du diagramme de Séquence sont :

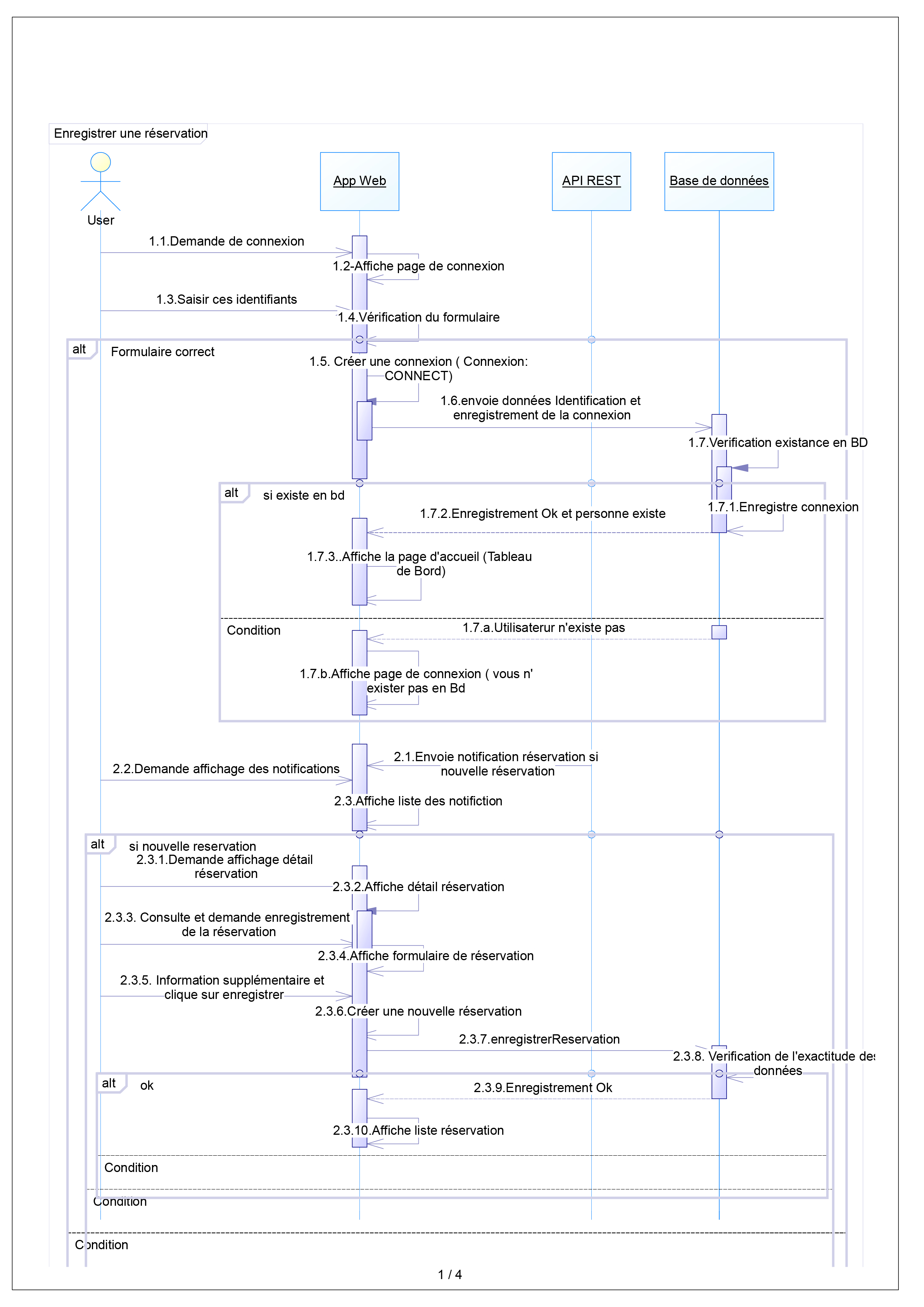
* Le nom du diagramme précédé de l’étiquette **sd** (sequence diagramm) ;
* L’objet ;
* La ligne de vie : période d’utilisation de l’objet, le moment ou l’objet exécute une méthode ;
* Les messages : communication particulière entre ligne de vie, communication entre objet. Ils peuvent signifier :
* L’innovation d’une opération : appel d’une méthode (message synchrone) :
* L’envoi d’un signal : gestion des évènements, signal (message asynchrone) ;
* Création ou destruction d’une instance de classe ;

**b. Quelques diagramme de séquence**

**i. Effectuer une réservation**

****

**ii. Enregistrer une réservation**

****

**Analyse statique**

**a. concepts**

Le **diagramme de classes** représente l’architecture conceptuelle du système : il décrit les classes (ou le type des objets) que le système utilise, ainsi que les relations qui les relient entre eux (association, héritage, agrégation, composition,).

Une **classe** est une description d’un groupe d’objets partageant un ensemble commun de propriété (attributs), de comportement (opérations) et de relations avec d’autres objets (associations ou agrégations).

**b. formalisation du diagramme de classe**

Les principaux symboles utilisés dans le diagramme de classes sont les suivants :

|  |  |
| --- | --- |
|  | Une note est un commentaire placé sur un diagramme. |
| Classes | Modèle de représentation des objets qui possède des données et des |
| Associations | comportements |
| Attributs | Propriété d'une classe qui décrit un domaine de valeurs possibles partagé par tous les objets de la classe. |
| Opérations | Traitement qui peut être demandé à n'importe quel objet de la classe et partagée par tous les objets de la classe. |
|  | Relations qui pourraient exister entre classe |

Tableau 4:Quelques concepts du diagramme de classes

**c. Diagramme de classe du système**

Les différentes classes du système sont :

1. **Conception de la solution**
2. **La phase de conception**

**But et activités**

Le but de la conception est de fixer les choix techniques et de préparation de l’implémentation.

Le modèle de la conception décrit la solution (comment le problème est résolu). Il est en général destiné aux développeurs et non pas aux utilisateurs.

Ainsi, les trois préoccupations suivantes sont spécifiques à la conception :

* Organiser le développement du système informatique et adresser des questions comme les dépendances entre modules, la configuration, la gestion des versions,
* Distribuer physiquement les différentes parties logicielles du système et
* Faire des choix techniques : définir les langages de programmation, les schémas de bases de données pour la persistance des données, etc.

**Conception globale et conception détaillée**

On peut organiser la phase de conception en deux étapes complémentaires : la conception globale et la conception détaillée.

La conception globale consiste à identifier les principaux composants du système et leurs dépendances (les interfaces et les protocoles de communication), les exigences de chaque composant, les caractéristiques de l’infrastructure.

Pendant la conception détaillée on s􀂶intéresse à chaque composant individuel et on décrit plus précisément comment il sera réalisé. Des équipes différentes peuvent travailler en parallèle pour le développement des différents composants.

UML met à la disposition des développeurs les diagrammes de composants et de déploiement, pour la construction de l’architecture du logiciel.

1. **Diagramme de composants**
2. **Définition et intérêt**

Le diagramme de composants fait partie des diagrammes structuraux (statiques) d’UML. Il décrit l'organisation du système du point de vue des éléments logiciels comme :

* Les modules (paquetages, fichiers sources, bibliothèque, exécutables) ;
* Des données (fichiers, bases de données) ;
* Ou encore d'éléments de configuration (paramètres, scripts, fichiers de commandes).

Ce diagramme permet de mettre en évidence les dépendances entre les composants. Ces dépendances peuvent être :

Des relations de compositions (boîte blanche)

Des relations d’assemblages et de connexion (via des interfaces)

1. **Notion de composant et d’interface**

En UML, un composant est un élément logiciel remplaçable et réutilisable qui fournit ou reçoit un service bien précis. Il peut être vu comme une pièce détachée du logiciel.

Un composant est un conteneur d'éléments logiques et représente des choses qui participent à l'exécution d'un système. Les composants utilisent également les services d'autres composants via certaines de leurs interfaces.

En principe, les composants sont utilisés pour visualiser les packages logiques de code source (composants produit de travail), le code binaire (composants de déploiement) ou les fichiers exécutables (composants d'exécution).

Les composants fournissent des services via des interfaces.

* Un composant peut-être remplacé par n’importe quel autre composant compatible c'est-à-dire ayant les mêmes interfaces.
* Un composant peut évoluer indépendamment des applications ou des autres composants qui l’utilise à partir du moment où les interfaces sont respectées.

Il existe deux types d’interface :

* Les interfaces requises : Ce sont des interfaces qui fournissent un service au composant et dont il a besoin pour fonctionner.
* Les interfaces fournies : Ce sont des interfaces par lesquels le composant fourni lui-même un service.

1. **Diagramme de déploiement**
2. **Méthodologie de gestion de projet : SCRUM**