Chapitre1 : Étude préalable

I. Introduction :

Dans ce chapitre nous présentons, le domaine d’application en l’analysant selon une démarche rationnelle qui se base sur les aspects fonctionnels et non fonctionnels.

II. Cadre du projet

Ce travail est réalisé dans le cadre d'un projet de fin d'études en vue de l'obtention d'une licence en Informatique Appliquée à la Gestion (E-Business). Le corsise universitaire de l'Institut Supérieur de Gestion de Gabès (ISG) s'achever avec l'obtention de cette licence et à nous initier à la vie professionnelle par la mise en pratique de nos connaissances.

III. Présentation

L'Institut Supérieur de Gestion de Gabès (ISGG) créé le 01/07/1998 par le décret N°468 du 23/02/1998 (J.O.R.T) du 03/03/1998.

IV. Étude de l'existence

1. Description de l’existence

Le service d'impression dans un institut est la charge de procédure processus matière de tirage soit pour les administrateurs, les enseignants et les étudiants.

Les professeurs soumettant les demandes d'impression à la salle d'impression de l'Institut et reçoivent les documents au même endroit dans le bureau.

2. Problématique

Les activités du le processus métiers (l’impression de document a l’ISGG) sont est se semi-automatique.

L’enseignant se déplace pour déposer leur demande d’impression des documents (TD, Cours, TP, Examen).

Les files d'attente peuvent s'accumuler rapidement, ce qui peut entraîner des retards importants pour les enseignants et qui doivent se rendre physiquement au service d'impression pour récupérer leurs documents imprimés, ce qui peut prendre du temps, surtout si le service est éloigné. De plus, c’est ne pas disposer de suffisamment d'espace pour stocker les documents imprimés, ce qui peut entraîner des retards et des problèmes logistiques.

3. Solutions proposées

Une application d’automatisation du service de tirage en ligne peut être un outil important pour aider les services d'impression et les enseignants à gérer plus efficacement les demandes de documents.

Grâce à notre application, les enseignants peuvent lancer directement les fichiers de documents qu'ils doivent imprimer, réduisant ainsi le temps pour gérer les demandes de documents en spécifiant tous les détails.

Le service de tirage en ligne permet aux enseignants de suivre l'état des demandes de documents et de se souvenir de leurs fichiers imprimés.

V. Spécification des aspects

Dans cette sous-section, nous allons identifier les aspects de notre application et nous voyons mettre l’accent sur les aspects fonctionnels ainsi que les aspects non fonctionnels pour éviter le développement d’une application non satisfaisante.

1. les aspects fonctionnels

L’aspect fonctionnel d'un service d'impression en ligne doit permettre aux utilisateurs de sélectionner, personnaliser, commander et recevoir des impressions de qualité, tout en offrant un processus de commande simple ;

- Choix de niveau

- Choix de filière

-  Choix d'auditoire

- Déterminer la matière

-  Type de matière

- Remplir le nombre des étudiants et les pages

- Déterminer la date de besoin du document

-  Type de fichier

- Le privilège peut ou peut ne pas imprimer de cours

Sélection des fichiers

- Suivi de commande.

2. les aspects non fonctionnels

Les aspects non fonctionnels d'un service d'impression en ligne se rapportent à la fiabilité, à la sécurité , à la performance et à la disponibilité du service pour répondre aux besoins des utilisateurs ;

- Fiabilité : les utilisateurs doivent pouvoir compter à tout moment sur la disponibilité et la fonctionnalité du service.

- Sécurité : les utilisateurs doivent être sûrs que leurs données personnelles et les fichiers téléchargés ne seront pas utilisés à mauvais escient ou piratés. Le service doit également garantir la confidentialité des documents et des informations confidentielles.

- Performance : Le service doit être rapide et efficace, capable de traiter les commandes rapidement et d'assurer une livraison dans les délais.

- Disponibilité : le service doit être disponible 24 heures sur 24, 7 jours sur 7, pour répondre aux besoins des utilisateurs en tout temps.

VI. Conclusion

Dans ce chapitre, nous avons introduit le contexte général de notre projet en présent une étude rationnelle d’exitance.

Les fruits de cette étape permettent de spécifier les fonctionnalités du system futur pour se préparer à l’étape suivante qui se résume à l’étude conceptuelle.

Chapitre 2 : Conception

I. Introduction

La phase de l’étude conceptuelle permet d’élaborer une représentation du système d'information. Dans ce chapitre, nous présentons une abstraction lisible et composable des différentes vues du système existant. Nous avons utilisé UML en tant qu'un langage de modélisation orientée objet.

II. Conception

La conception est une étape essentielle du cycle de vie d'une application et son objectif est de développer un modèle détaillé de l'architecture du système à partir du modèle du système obtenu à l'étape d'analyse des exigences. Il vise également à réduire la complexité du système.

La spécification des exigences est la phase initiale de tout projet informatique à développer, où nous identifierons les exigences de l'application. Nous distinguons les besoins fonctionnels, qui démontrent la fonctionnalité cible de notre application, et les besoins non fonctionnels, pour éviter de développer une application insatisfaisante et trouver un accord commun entre experts et utilisateurs pour réussir le projet.

Pour couvrir le cycle de vie de réalisation de ce projet, nous utilisons l’UML

"UML est un langage de modélisation visuelle pour spécifier, construire et documenter les artefacts d'un système logiciel[[1]](#footnote-1)." - Grady Booch

« Le langage de modélisation unifiée (UML) est un langage de modélisation normalisé permettant aux développeurs de spécifier, visualiser, construire et documenter les artefacts d'un système logiciel. Ainsi, UML rend ces artefacts évolutifs, sécurisés et robustes en exécution. UML est un aspect important impliqué dans le développement de logiciels orientés objet. Il utilise la notation graphique pour créer des modèles visuels de systèmes logiciels[[2]](#footnote-2). »

« UML est destiné à faciliter la conception des documents nécessaires au développement d'un logiciel orienté objet, comme standard de modélisation de l'architecture logicielle. Les différents éléments représentables[[3]](#footnote-3) »

  -La conception de logiciels met en œuvre tout un ensemble d'activités, à commencer par la demande d'informatisation des processus, permettant la conception, l'écriture et le développement de logiciels.

-La documentation logicielle est le texte écrit qui explique aux informaticiens comment un logiciel est conçu et explique aux novices comment l'utiliser. Par conséquent, il convient de noter que ce document peut être destiné aux professionnels ou aux utilisateurs finaux du logiciel.

-Communication entre équipes : ‘ La communication assure le fonctionnement du lieu de travail. La qualité de la communication peut affecter les résultats du travail de l'équipe sur la durée. Grâce à une bonne communication, nous renforçons l’efficacité et la productivité entre les collaborateurs[[4]](#footnote-4).’

-L’analyse des besoins évalue la différence entre résultats actuels et résultats souhaités. Bien réalisée, cette analyse vous permet d’obtenir des informations précieuses au sujet des processus de votre équipe et met en lumière des pistes d’amélioration pour gagner en efficacité.

- Le test logiciel « est le processus qui consiste à évaluer et à vérifier qu'un produit ou une application logicielle fait ce qu'il ou elle est censée(e) faire[[5]](#footnote-5). »

-Modélisation de processus métier représentation schématique et graphique de l'organisation optimale de toutes les tâches qui composent les activités d'une entité.

III. Diagrammes de cas d'utilisation

1. Définition

« Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel d'un systeme logiciel.

Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (Human ou Machine) et un system. Il est une entité significative de travail.

Dans un diagramme de cas d'utilisation il existe des acteurs (actors) qui interagissent avec des cas d'utilisation (use case) UC.

Les use case permettent de structurer les besoins des utilisateurs et les objectifs du systeme.

Une fois identifié et structuré ces besoins :

­ Définissent le contour du systeme à modéliser.

­ Permettent d'identifier les fonctionnalités principales ou critiques du systeme[[6]](#footnote-6). »

2. Identification des acteurs :

En UML, un acteur spécifie toute idéalisation d'un rôle joué par une personne.

Externe, processus ou entité qui interagit avec le système. Il existe deux types.

Acteurs, acteurs principaux, ils utilisent directement systèmes et acteurs auxiliaires qui contribuent à la réalisation du cas d'utilisation.

Les acteurs interagissant avec notre système sont :

Admin (responsable impression)

Enseignant

2. Descriptions graphique

La figure ci-dessous, représente le diagramme de cas d’utilisation global pour décrire les fonctionnalités du système.

|  |
| --- |
|  |

**Figure 1 : Diagramme de cas d’utilisation globale de service du tirage en ligne**

La figure suivante illustre toutes les activités des Enseignants

|  |
| --- |
|  |

**Figure 2 : Diagramme de cas d’utilisation : Vue Enseignant**

La figure suivante illustre toutes les activités d’Admin

|  |
| --- |
|  |

**Figure 3 : Diagramme de cas d’utilisation : Administrateur**

Description textuelle

Afin de mieux comprendre notre système et les interactions avec les utilisateurs, nous allons détailler les scénarios des principaux cas d’utilisation.

|  |  |
| --- | --- |
| Cas d'utilisation | S'inscrire |
| Acteur | Enseignant |
| Objective | Pour sauvegarder (entrer) vos données |
| Pré condition | On clique sur bouton « Valider » |
| Post-condition | Affichez la zone d'inscription et poursuivez ces étapes jusqu'à atteindre la zone de formulaire. |
| Scénario | 1-le système affiche l’interface appropriée d’inscription.  2-le client suivre les étapes de choix.  3-l’enseignant doit remplir le formulaire.  4-Valider leur choix.  5-Il peut consulter son archive. |

Tableau 1 : Description textuelle du cas d’utilisation « S'inscrire »

|  |  |
| --- | --- |
| Cas d’utilisation | Gère les impression |
| Acteur | Admin (responsable impression) / Enseignant |
| Pré- condition | Les utilisateurs doivent suivre les étapes. |
| Post-condition | Chaque acteur devenu dans son interface |
| Scénario |  |

Tableau 2 : Description textuelle du cas d’utilisation « Gère les impression »

|  |  |
| --- | --- |
| Cas d'utilisation | Consulter état impression |
| Acteur | Enseignant |
| Objective | Vérification |
| Pré condition | L’utilisateur contrôle l’impression |
| Post-condition | Recevoir un message ou non |
| Scénario | Remplir le formulaire  Valider  Avoir un message email |

Tableau 3 : Description textuelle du cas d’utilisation « Consulter état impression »

IV. Diagramme de classe

« Les diagrammes de classes sont l'un des types de diagrammes UML les plus utiles, car ils décrivent clairement la structure d’un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets[[7]](#footnote-7). »

Dictionnaire apuré des données

Le tableau suivant représente les propriétés de toutes les classes qui composent notre système et leurs descriptifs :

|  |  |  |
| --- | --- | --- |
| Classe Personne | | |
| Attribut | Type | Description |
| Id | Int | L’identifiant d’utilisateur |
| Nom | Chaine | Le nom d’utilisateur |
| Téléphone | Chaine | Le téléphone d’utilisateur |
| Email | Chaine | L’email d’utilisateur |
| Sexe | Chaine | Le sexe du l’utilisateur |
| Matière enseignées | Chaine | Matières enseignées par chaque enseignant |
| Mot de passe | Chaine | Le mot de passe du compte |

|  |  |  |
| --- | --- | --- |
| Classe Enseignant | | |
| Attribut | Type | Description |
| Id | Int | L’identifiant de l’enseignant |
| Spécialité | Chaine | Spécialité de chaque enseignant |

|  |  |  |
| --- | --- | --- |
| Classe Archive \_imp | | |
| Attribut | Type | Description |
| Id\_archive | Int | L’identifiant de l’archive |

|  |  |  |
| --- | --- | --- |
| Classe Serveur\_imp | | |
| Attribut | Type | Description |
| IP serveur | Chaine | L’adresse IP de serveur |
| Dossier\_ partager | Chaine | Le dossier partager par le serveur d’impression |

|  |  |  |
| --- | --- | --- |
| Classe type\_imp | | |
| Attribut | Type | Description |
| Lib\_type | Chaine | Libellé de type d’impression |

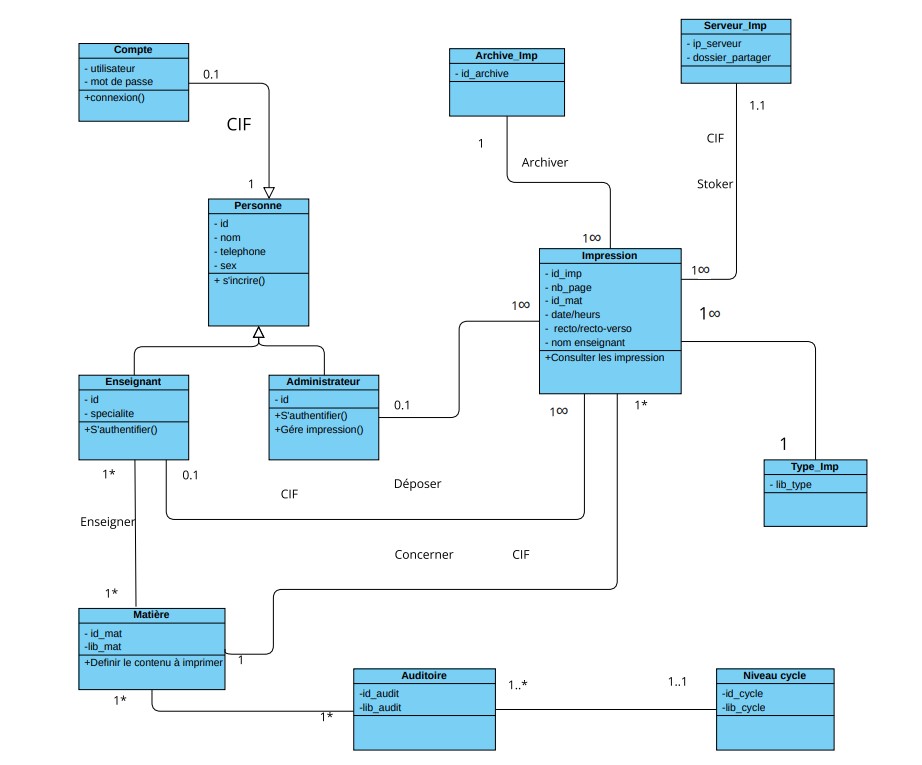
|  |  |  |
| --- | --- | --- |
| Classe Matière | | |
| Attribut | Type | Description |
| Id\_mat | int | L’identifiant de matière |
| Lib\_mat | Chaine | Libellé de  Matière |

|  |  |  |
| --- | --- | --- |
| Classe Auditoire | | |
| Attribut | Type | Description |
| Id\_audit | Int | L’identifiant d’auditoire |
| Lib\_audit | Chaine | Libellé d’auditoire |

|  |  |  |
| --- | --- | --- |
| Classe Niveau cycle | | |
| Attribut | Type | Description |
| Id\_niveau | Int | L’identifiant de niveau |
| Lib\_niveau | Chaine | Libellé  De niveau |

|  |  |  |
| --- | --- | --- |
| Classe Impression | | |
| Attribut | Type | Description |
| Id\_ipm | Int | L’identifiant d’impression |
| Enseignant | Chaine | L’enseignant qui  demande l’impression |
| Matière | Chaine | La matière choisir a imprimer |
| Niveau | Chaine | Le niveau choisir |
| Filière | Chaine | Le filière choisir |
| Auditoire | Chaine | L’auditoire choisir |
| Nb\_étudient | Chaine | Nombre des étudiants choisir |
| Date | Chaine | Date fixer |
| Type\_imp | Chaine | Le type demander |
| Doc | Chaine | Document à imprimer |

Diagramme de classe globale



**Figure 4 : Diagramme de classe globale**

Modèle logique de données liée au diagramme de classe

V. Conclusion

Ce chapitre décrit les différentes étapes de l’étude conceptuelle. Selon l’approche objet et avec un langage unifiait UML.

Nous représentons le système d’information service tirage selon une vue statique, dynamique et comportementale. Cette étape constitutive est un point de départ de la phase suivante qui consiste donnons la réalisation et le test d’intégration.

1. Citation de Grady Booch, l'un des co-créateurs d'UML [↑](#footnote-ref-1)
2. https://fr.theastrologypage.com/unified-modeling-language#menu-1 [↑](#footnote-ref-2)
3. Wikipédia. [↑](#footnote-ref-3)
4. Rédigé par Antoni, Le 17/02/2022 [↑](#footnote-ref-4)
5. https://www.ibm.com/fr-fr/topics/software-testing#:~:text=Le%20test%20logiciel%20est%20le,est%20cens%C3%A9(e)%20faire. [↑](#footnote-ref-5)
6. https://atefsd.weebly.com/uploads/5/0/3/6/503639/csi\_02\_chap02.pdf [↑](#footnote-ref-6)
7. https://www.lucidchart.com/pages/fr/diagramme-de-classes-uml [↑](#footnote-ref-7)