**Institut Supérieur de Comptabilité et d’Administration des Entrepris**

*Nouakchott-Mauritanie*



**PROJET DE FIN D’ETUDES**

Pour l’obtention de :

**LICENCE en Développement Informatique**

**Thème :**

**Développement d’une application mobile publication Des résultats nationaux**



***Elaboré par:***

* ***Ebby Cheikhna Sidiboubacar I19112***
* ***Mohamed Sidiya M’Boirik I18939***
* ***Isshagh Alla Menih I18919***

***Encadré par :***

**Dr. Ahmed ould Sejad**

[Année Universitaire : 2024– 2025]

# Dédicaces

Nous dédions ce modeste travail avec une profonde reconnaissance :

À nos chers parents, pour leur dévouement indéfectible, leurs sacrifices inlassables, et leur amour inconditionnel qui ont été notre inspiration et notre soutien tout au long de nos études.

À nos chères sœurs et nos adorables frères, pour leur soutien constant, leurs encouragements sincères et leur présence précieuse qui ont enrichi notre parcours académique.

À tous nos amis, ainsi qu'à toutes les personnes qui ont contribué, de près ou de loin, à notre formation, pour leurs conseils avisés, leur collaboration généreuse, et leur soutien inestimable qui ont joué un rôle crucial dans notre réussite.

À chacun d'entre eux, nous exprimons notre profonde gratitude et notre reconnaissance pour avoir été des piliers essentiels dans notre parcours éducatif et professionnel.

# Remerciements

Tout d'abord, nous exprimons notre profonde gratitude envers Allah le tout puissant et miséricordieux, qui nous a guidés et soutenus tout au long de notre parcours académique, nous accordant la force et la patience nécessaires pour accomplir ce modeste travail.

Ensuite, nous tenons à adresser nos sincères remerciements à notre encadreur, le Dr. Ahmed Ould Sejad. Ses précieux conseils, son soutien constant et son engagement ont été essentiels pour la réalisation de ce travail. Sa disponibilité et son expertise ont grandement enrichi notre expérience de formation à l'Institut Supérieur de Comptabilité et d’Administration des Entreprises (ISCAE).

Nous souhaitons également exprimer notre reconnaissance sincère à l’administration et à l’ensemble du corps enseignant de l'ISCAE. Leur dévouement et leur engagement à fournir une formation de qualité ont été une source d'inspiration et de motivation tout au long de notre cursus.

Enfin, nous adressons nos remerciements chaleureux à toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail. Leur soutien, leurs encouragements et leur collaboration ont été précieux et nous ont permis d'atteindre nos objectifs avec succès.

Nous sommes profondément reconnaissants envers chaque individu au sein de l'institution pour leur soutien précieux et leur contribution essentielle à notre développement académique et professionnel.

**الملخص**

Résumé

Summary

Table des matières

[Dédicaces 1](#_Toc169475486)

[Remerciements 2](#_Toc169475487)

[LISTES DES FIGURES 6](#_Toc169475488)

[Liste des abréviations 7](#_Toc169475489)

[***Introduction*** Générale : 7](#_Toc169475490)

[Chapitre I. Présentation du cadre de projet 9](#_Toc169475491)

[I.1. Introduction : 10](#_Toc169475492)

[I.2. Cadre du projet : 10](#_Toc169475493)

[I.3.Présentation de L’ISCAE 10](#_Toc169475494)

[I.4. Etude de l’existence 12](#_Toc169475495)

[I.5. Problématique 12](#_Toc169475496)

[I.6. Objectif du projet 12](#_Toc169475497)

[I.7. Solutions proposées 12](#_Toc169475498)

[I.8. Équipe de développement. 13](#_Toc169475499)

[Chapitre II. ANALYSE ET SPECIFICATION DES BESOINS 14](#_Toc169475500)

[II. 1. Spécification des besoins : 14](#_Toc169475501)

[II.1.1. Besoins fonctionnels : 14](#_Toc169475502)

[II.1.2. Besoins non fonctionnels : 14](#_Toc169475503)

[Chapitre III. CONCEPTION 16](#_Toc169475504)

[III.1. Présentation du langage UML 16](#_Toc169475505)

[III.2. CONCEPTION DE L’APPLICATION 19](#_Toc169475506)

[III.2.1. Modélisation avec le diagramme des cas d’utilisation : 19](#_Toc169475507)

[III .2.1.1. Diagramme des cas d'utilisation « Administrateur » 20](#_Toc169475508)

[III .2.1.2. Diagramme des cas d'utilisation « Candidat » 20](#_Toc169475509)

[III.2.2. DIAGRAMME DE CLASSE : 20](#_Toc169475510)

[III.3. Conclusion 21](#_Toc169475511)

[Chapitre V. DEVELOPEMENT ET REALISATION 21](#_Toc169475512)

[V.1. INTRODUCTION : 22](#_Toc169475513)

[V.2. Architecture de l’application : 22](#_Toc169475514)

[V.3. Architecture du système : 22](#_Toc169475515)

[V.4. Environnement de Développement : 22](#_Toc169475516)

[V.4. 1. Environnement logiciels : 22](#_Toc169475517)

[V.4. 1.1. Les langages de programmation : 23](#_Toc169475518)

[V.4. 2. Outils et logiciel : 26](#_Toc169475519)

[Chapitre VI. **PRESENTATION DU SYSTḔME** 30](#_Toc169475520)

# 

# LISTES DES FIGURES

[**Figure 1:Logo Iscae** 10](#_Toc169475579)

[**Figure 2:Cadre de filières de l’institut** 11](#_Toc169475580)

[**Figure 3:Diagramme Uml** 18](#_Toc169475581)

[**Figure 4: Diagrammes de cas d’utilisation administrateur** 20](#_Toc169475582)

[Figure 5:Diagrammes de cas d’utilisation candidat 20](#_Toc169475583)

[**Figure 6: Diagrammes de classe** 21](#_Toc169475584)

[Figure 7:logo Dart 23](#_Toc169475585)

[Figure 8: logo Flutter 23](#_Toc169475586)

[Figure 9: Logo Python 23](#_Toc169475587)

[Figure 10: Logo Django 24](#_Toc169475588)

[Figure 11: Logo Bootstrap 24](#_Toc169475589)

[Figure 12: Logo Html 25](#_Toc169475590)

[Figure 13: Logo CSS 25](#_Toc169475591)

[Figure 14: Logo Java Script 25](#_Toc169475592)

[Figure 15: Logo Sql 26](#_Toc169475593)

[Figure 16: Logo MySQL 26](#_Toc169475594)

[Figure 17: Logo Uml 26](#_Toc169475595)

[Figure 18: Logo StartUml 27](#_Toc169475596)

[Figure 19: Logo Visual Studio Code 27](#_Toc169475597)

[Figure 20: Logo Android Studio 28](#_Toc169475598)

[Figure 21: Logo Xampp 28](#_Toc169475599)

[Figure 22: Logo Git 29](#_Toc169475600)

[Figure 23: Logo GitHub 29](#_Toc169475601)

# Liste des abréviations

* **L’OMG: Object Management Group**
* **OMT: Object Modeling Technique**
* **OOSE: Object-Oriented Software Engineering**
* **UML: Unified Modeling Language**

# Introduction Générale :

Dans un paysage où la rapidité et l'accessibilité de l'information sont des impératifs, la création d'une plateforme innovante pour la publication des résultats nationaux des étudiants et des concours revêt une importance cruciale. C'est dans cette perspective que notre projet prend forme : le développement d'une application mobile dédiée à la diffusion des résultats nationaux des étudiants, conjointement à un tableau de bord web permettant l'importation fluide des données et la publication efficace des résultats des concours.

Ce projet s’inscrit dans une volonté de répondre aux besoins éducatifs et administratifs de notre société moderne. La centralisation et la diffusion instantanée des résultats nationaux des étudiants ainsi que des concours constituent un levier majeur pour la transparence, l'équité et la confiance dans les systèmes éducatifs et de sélection. En mettant à disposition une plateforme intuitive et sécurisée pour la publication de ces résultats, notre projet aspire à renforcer la collaboration entre les institutions éducatives, les organisateurs de concours, les autorités gouvernementales et les candidats, contribuant ainsi à une amélioration significative de la gestion des résultats et à une plus grande inclusivité dans l'accès à l'information.

Cette introduction esquisse les objectifs de notre projet, sa pertinence dans le contexte actuel de la publication des résultats nationaux des étudiants et des concours, ainsi que les principales fonctionnalités de l'application mobile et du tableau de bord web qui seront développées pour répondre à ces besoins.

Enfin, nous aborderons les perspectives d'avenir pour cette application, envisageant des améliorations potentielles, une extension vers d'autres domaines administratifs, ainsi que des possibilités de collaboration avec les autorités éducatives et les organisateurs de concours pour une adoption plus large et efficace de la plateforme, dans le but ultime de favoriser l'égalité des chances et l'excellence académique dans notre société.

# Chapitre I. Présentation du cadre de projet

## I.1. Introduction :

Dans ce chapitre je vais commencer par une détermination du cadre du projet puis je vais

présenter mon institut l’ISCAE , ensuite la deuxième partie du chapitre sera une explication

de la situation existante par rapport au sujet de mon projet, les problématiques, les objectifs,

les solutions proposées.

## I.2. Cadre du projet :

Le projet entre dans le cadre de présentation du mémoire de fin d’études pour l’obtention de

la licence appliquée en Développement Informatique (DI), c’est un développement d’une

application mobile pour publication des résultats national

## I.3.Présentation de L’ISCAE

* ISCAE



**Figure 1:Logo Iscae**

L’Institut Supérieur de la Comptabilité et d’Administration des Entreprises (ISCAE) régi par l’ordonnance, n° 2006-007 du 20 février 2006 portant organisation de l’enseignant supérieur, a été créé en 2009 par D2cret, n° 2009-161 dans le but de dispenser un enseignement supérieur à même de répondre aux besoins du marché de l’Emploi. A cette fin, cet établissement public d’enseignement supérieur a été doté de la personnalité morale et de l’autonomie administrative et financière et jouit de l’autonomie pédagogique dans le cadre de l’exercice de sa mission. Ainsi donc, l’ISCAE a pour mission de développer et d’offrir des programmes :

• De formation (initiale et continue) ;

• De recherche ;

• De vulgarisation ;

• De prestation de services, et

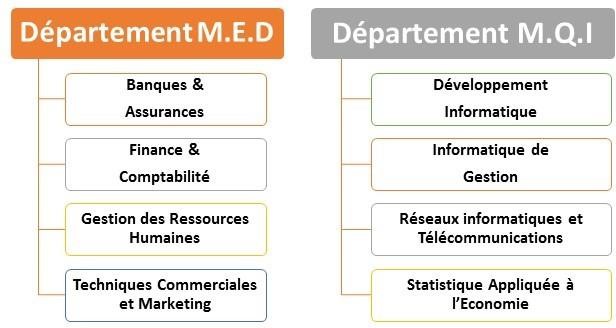
• De promotion de l’entrepreneuriat, dans des domaines aussi divers que :

• La comptabilité et le contrôle (expertise comptable, audit, contrôle, etc.) ;

• Les techniques modernes de gestion (management, ressources humaines, commerce, Marketing et communication, informatique de gestion de développement, etc.).

Ces spécialités sont prises en charge par cet Institut Supérieur dans le cadre de filières Suivantes **:**

* **Filières de l’institut**



**Figure 2:Cadre de filières de l’institut**

## I.4. Etude de l’existence

Analyse de la présence et des alternatives actuelles pour la publication des résultats nationaux et des concours dans les plateformes numériques existantes.

Absence d'applications mobiles : Il n'existe pas d'application mobile dédiée à la centralisation et à la diffusion de ces résultats, ce qui limite l'accessibilité et la commodité pour les utilisateurs.

Sites Web : Actuellement, la publication des résultats nationaux, des concours et des recrutements se fait principalement par des sites Web. Cependant, ces sites sont souvent sujets à des pannes et des surcharges lors des périodes de publication.

## I.5. Problématique

Identification des lacunes dans les systèmes actuels de diffusion des résultats nationaux, notamment la dépendance excessive aux sites Web sujets aux pannes, l'absence de centralisation des informations, et la difficulté d'accès aux résultats par les utilisateurs.

Absence de notification en temps réel : Les utilisateurs doivent constamment vérifier les sites Web pour voir si les résultats ont été publiés, ce qui est peu pratique.

## I.6. Objectif du projet

Développement d'une application mobile innovante pour centraliser, simplifier et sécuriser l'accès aux résultats nationaux, concours et recrutements, tout en offrant une alternative fiable aux sites Web traditionnels et en améliorant l'expérience utilisateur.

Fiabilité et performance : Développer une application robuste et capable de gérer de fortes charges de trafic sans interruption.

Fiabilité et performance : Développer une application robuste et capable de gérer de fortes charges de trafic sans interruption.

## I.7. Solutions proposées

Création d'une application mobile dédiée à la publication rapide et sécurisée des résultats.

Intégration d'une interface utilisateur intuitive pour faciliter la navigation et l'accès aux informations.

Mise en place d'un système de notifications pour informer les utilisateurs des concours et les emplois.

Développement de fonctionnalités statistiques pour analyser les données de résultats et fournir des insights utiles aux utilisateurs et aux administrateurs.

Développement d'une l'application mobile compatible avec les principaux systèmes d'exploitation mobiles (iOS et Android).

## I.8. Équipe de développement.

# Chapitre II. ANALYSE ET SPECIFICATION DES BESOINS

## II. 1. Spécification des besoins :

Dans cette partie, on explique en détail ce que l'application est censée faire et ceci à travers la spécification des besoins fonctionnels et non fonctionnels.

### II.1.1. Besoins fonctionnels :

* Gestion de Tableau de bord
* Gestion des Postes
* Gestion des Résultats
* Gestion des Statistiques

### II.1.2. Besoins non fonctionnels :

* Performance

L’application répond à tous les exigences de l’utilisateur d’une manière optimale

* Fiabilité

Bon fonctionnement de l’application sans détection de défaillance

* Rapidité

Déconnexion après un temps d’inactivité

* Convivialité

Un design clair, souple et interactif

Une bonne interface qui donne l’envie à l’utilisateur d’utiliser l’application

Positionnement du contenu dans les pages de la manière la plus accessible

* Portabilité

L’application est multiplateforme : Elle fonctionne sur tout système d’exploitation o Elle fonctionne sur tout type de terminal

# Chapitre III. CONCEPTION

La démarche de conception est une étape fondamentale dans le processus de développement puisqu’elle fait correspondre la vision applicative (le modèle d’analyse) à la vision technique (l’environnement de développement et d’exécution). Ce chapitre vise à illustrer la phase de conception et les modèles UML associés. J’ai commencé par une présentation du lagunage UML, et par la suite établir les diagrammes de séquences, des cas d’utilisation, ensuite et le diagramme de classe, et je termine avec une petite conclusion.

## III.1. Présentation du langage UML

Pour faire face à la complexité des systèmes d'information, de nouvelles méthodes et outils ont été créées. La principale avancée des quinze dernières années réside dans la programmation orientée objet.

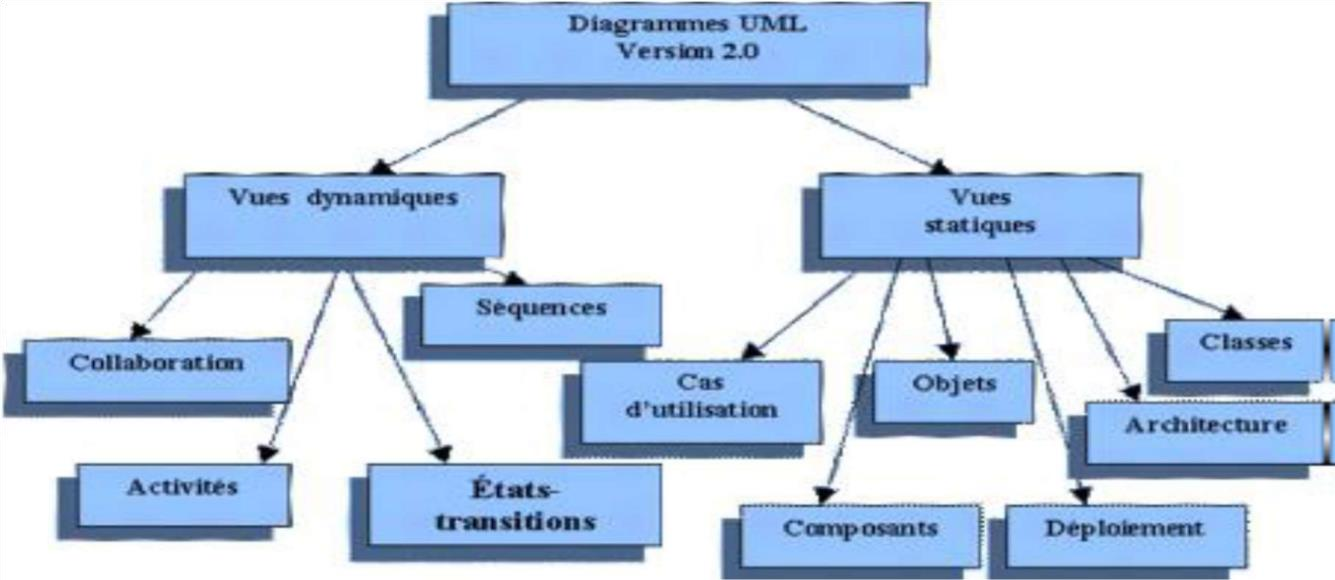
Face à ce nouveau mode de programmation, les méthodes de modélisation classique (telle que **MERISE**) ont rapidement montré certaines limites et ont dû s'adapter.

De très nombreuses méthodes de modélisation ont également vu le jour comme **Boch**, **OMT**

...

Dans ce contexte et devant le foisonnement de nouvelles méthodes de conception orientée objet, **l'OMG** (Object Management Group) a eu comme objectif de définir une notation standard utilisables dans les développements informatiques basés sur l'objet. C'est ainsi qu'est apparu **UML** (qui signifie en français langage de modélisation unifiée), qui est issu de la fusion des méthodes de Booch, **OMT** et **OOSE**. C'est un langage de modélisation qui permet de représenter graphiquement les besoins des utilisateurs à l'aide de diagramme.

Un diagramme **UML** est une représentation graphique, qui s'intéresse à un aspect précis du modèle. C'est une perspective du modèle, pas « le modèle ». Chaque type de diagramme **UML** possède une structure.



**Figure 3:Diagramme Uml**

Il existe deux types de vues du système constituant chacune des diagrammes qui sont répartis selon leurs aspects statiques ou dynamiques.

Selon les vues statiques nous avons :

* Le diagramme des cas d'utilisation

Le diagramme de cas d'utilisation permet de structurer les besoins des utilisateurs et les objectifs correspondants d'un système. Il essaiera de répondre à des questions du genre : Qui devra pouvoir faire quoi grâce au logiciel. Les acteurs principaux qui sont liés à un package auront besoin de cette partie du logiciel pour réaliser une plusieurs lots d'actions.

* Le diagramme d'objet

Le diagramme d'objet sert à illustrer les classes complexes en utilisant des exemples d'instances. A l'exception de la multiplicité, qui est explicitement indiquée, le diagramme d'objets utilise les mêmes concepts que le diagramme de classes. Ils sont essentiellement utilisés pour comprendre ou illustrer des parties complexes d'un diagramme de classes.

* Le diagramme des classes

Le diagramme des classes représente les entités manipulées par les utilisateurs. L’intérêt du diagramme des classes est de modéliser les entités du système d'information. Le diagramme met en évidence d'éventuelles relations entre les classes ou les entités.

* Le diagramme de déploiement

Les diagrammes de déploiement correspond à la description de l'environnement d'exécution du système (matériel, réseau...) et de la façon dont les composants y sont installés. Les diagrammes de déploiement sont donc très utiles pour modéliser l'architecture physique d'un système.

Selon les vues dynamiques nous avons :

Le diagramme de collaboration Le diagramme de collaboration (appelé également diagramme de communication) permet de mettre en évidence les échanges de messages entre objets. Cela nous aide à voir clair dans les actions qui sont nécessaires pour produire ces échanges de messages. Et donc de compléter, si besoin, les diagrammes de séquence et de classes.

* Le diagramme de séquence

Le diagramme de séquence permet de décrire les différents scénarios d'utilisation du système. C'est une variante du diagramme de collaboration sauf qu'il possède intrinsèquement une dimension temporelle mais ne représente pas explicitement les liens entre les objets.

* Le diagramme d'états-transitions

Le diagramme d'état-transition permet de décrire le cycle de vie des objets d'une classe. Il définit l'enchaînement des états de classe et font donc apparaitre l'ordonnancement des travaux.

* Le diagramme d'activités

Le diagramme d'activité représente le déroulement des actions, sans utiliser les objets. En phase d'analyse, il est utilisé pour consolider les spécifications d'un cas d'utilisation.

Conscient de la complémentarité qui existe entre ces différents diagrammes, et sachant qu'UML ne préconise aucune démarche, on a jugé nécessaire de ne prendre que certains de ces diagrammes qui répondent bien à nos besoins afin de concevoir notre système.

Cependant, les diagrammes qui vont être utilisés dans la suite de mon exposé sont les Suivants:

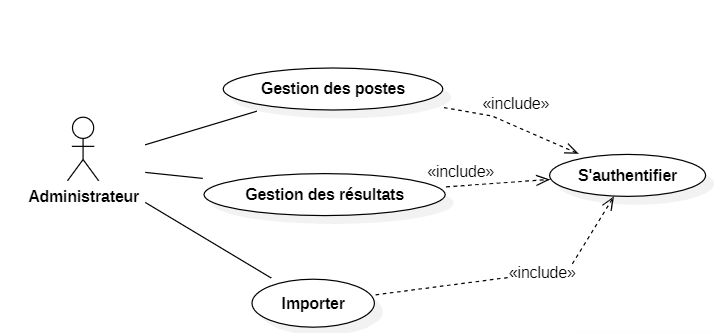
Le diagramme des cas d'utilisation.

Et Le diagramme des classes.

## III.2. CONCEPTION DE L’APPLICATION

### III.2.1. Modélisation avec le diagramme des cas d’utilisation :

#### III .2.1.1. Diagramme des cas d'utilisation « Administrateur »



**Figure 4: Diagrammes de cas d’utilisation administrateur**

#### III .2.1.2. Diagramme des cas d'utilisation « Candidat »

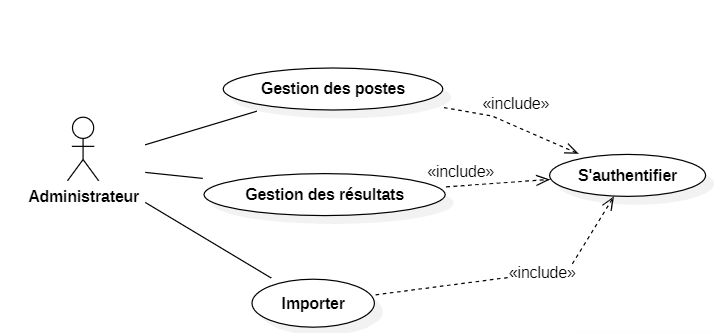


Figure 5:Diagrammes de cas d’utilisation candidat

### III.2.2. DIAGRAMME DE CLASSE :

Le diagramme de classes exprime la structure statique du système en termes de classes et de relations entre eux, Son intérêt est de modéliser les entités du système d’information.

Le diagramme de classe permet de représenter l’ensemble des informations finalisées qu Le diagramme de classe permet de représenter l’ensemble des informations finalisées qui sont gérées par le domaine.

Ces informations sont structurées, c’est-à-dire quelles sont regroupées dans des classes. Le diagramme met en évidence d’éventuelles relations entre ces classes Le diagramme met en évidence d’éventuelles relations entre ces classes Le diagramme met en évidence d’éventuelles relations entre ces classes Le diagramme De classes de mon application est le suivant**:**

**Figure 6: Diagrammes de classe**

## III.3. Conclusion

# Chapitre V. DEVELOPEMENT ET REALISATION

## V.1. INTRODUCTION :

Après l’étape de conception de l’application, nous allons dans ce chapitre, d’écrire la phase de réalisation et de développement. Nous allons présenter, en premiers lieu, L’architecture de l’application et du Système ensuite parlerons de l’environnement du travail utilisé pour le développement de l’application.

## V.2. Architecture de l’application :

## V.3. Architecture du système :

## V.4. Environnement de Développement :

Pour interagir avec le serveur et la base de données, nous sommes appelés à faire recours au moins à un langage de programmation. Dans cette partie, nous avons identifié les différentes caractéristiques de l’environnement matériel et logiciel que nous avons servi à l’implémentation de notre application.

### V.4. 1. Environnement logiciels :

#### V.4. 1.1. Les langages de programmation :

#### 



Figure 7:logo Dart

Est unlangage de programmation optimisé pour les applications sur plusieurs plateformes. Il est développé par Google et est utilisé pour créer des applications mobiles, bureau, de serveur et web. Dart est un langage orienté objet à ramasse-miettes avec une syntaxe de type C++.



Figure 8: logo Flutter

Flutter est un framework open-source de Google permettant de créer des applications nativement compilées pour mobile, web et desktop avec une seule base de code. Utilisant le langage Dart, Flutter offre des widgets intégrés et personnalisables pour des interfaces utilisateur performantes et attrayantes. Il propose des performances élevées grâce à la compilation en code natif et facilite le développement avec la fonctionnalité Hot Reload. Soutenu par Google et une communauté active, Flutter est idéal pour des applications multiplateformes cohérentes et efficaces.



Figure 9: Logo Python

Python est un langage de programmation qui peut s'utiliser dans de nombreux contextes et s'adapter à tout type d'utilisation grâce à des bibliothèques spécialisées. Il est cependant particulièrement utilisé comme langage de script pour automatiser des tâches simples mais fastidieuses, comme un script qui récupérerait la météo sur Internet ou qui s'intégrerait dans un logiciel de conception assistée par ordinateur afin d'automatiser certains enchaînements d'actions répétitives (voir la section Adoption). On l'utilise également comme langage de développement de prototype lorsqu'on a besoin d'une application fonctionnelle avant de l'optimiser avec un langage de plus bas niveau. Il est particulièrement répandu dans le monde scientifique, et possède de nombreuses bibliothèques optimisées des estinées au calcul numérique



Figure 10: Logo Django

Django est un framework Python de haut niveau pour développer rapidement des sites web sécurisés et maintenables. Gratuit et open source, il offre une large gamme de fonctionnalités intégrées et bénéficie d'une communauté active et d'une documentation complète.  
  
**Django vous aide à écrire des logiciels qui sont :**

 **Complète** : Fournit tout ce dont les développeurs ont besoin, garantissant compatibilité et cohérence.

 **Polyvalent** : Utilisable pour divers types de sites (gestionnaires de données, wikis, réseaux sociaux, sites d'actualités) et supporte différents formats de données (HTML, RSS, JSON, XML).

 **Sécurisé** : Protège contre les vulnérabilités courantes et gère les utilisateurs et mots de passe de manière sécurisée.

 **Scalable** : Architecture "shared-nothing" permettant d'ajouter du matériel pour gérer l'augmentation du trafic. Utilisé par des sites comme Instagram et Disqus.

 **Maintenable** : Encourage un code réutilisable et non redondant (philosophie DRY), suivant le motif d'architecture Modèle-Vue-Template (MVT).



Figure 11: Logo Bootstrap

Bootstrap est un framework open-source développé par Twitter pour la création de sites web responsives. Il utilise HTML, CSS et JavaScript, offrant une variété de modèles, composants et styles prédéfinis. Le système de grille responsive permet d'organiser le contenu adaptativement. Bootstrap inclut des composants comme des boutons, formulaires et carrousels, facilitant l'intégration d'éléments interactifs. La personnalisation est possible via des variables Sass ou CSS et des thèmes prédéfinis. Avec une documentation détaillée et une large compatibilité navigateur, Bootstrap est apprécié pour sa facilité d'utilisation et sa robustesse dans le développement web.



Figure 12: Logo Html

L’HyperText Mark up Langage, généralement abrégé HTML, est le langage de balisage conçu pour représenter les pages web. C’est un langage permettant d’écrire de l’hypertexte, d’où son nom. HTML permet également de structurer sémantiquement et logiquement et de mettre en forme le contenu des pages, d’inclure des ressources multimédias dont des images, des formulaires de saisie et des programmes informatiques.



Figure 13: Logo CSS

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



Figure 14: Logo Java Script

JavaScript est un langage de programmation principalement utilisé pour les pages web interactives et les serveurs via Node.js. C'est un langage orienté objet à prototype, où les objets ne sont pas des instances de classes mais possèdent des constructeurs pour créer des propriétés et des objets héritiers personnalisés. Les fonctions y sont des objets de première classe. JavaScript supporte les paradigmes objet, impératif et fonctionnel, et possède un large écosystème grâce à son gestionnaire de dépendances NPM. Créé en 1995 par Brendan Eich, JavaScript a été standardisé sous le nom d'ECMAScript en 1997 par Ecma International. Il existe différentes implémentations d'ECMAScript, comme JScript de Microsoft et ActionScript d'Adobe. Avec HTML et CSS, JavaScript est une des technologies fondamentales du Web, permettant des pages interactives et étant largement utilisé par la majorité des sites web.



Figure 15: Logo Sql

SQL est un langage informatique utilisé pour effectuer des requêtes sur des bases de données ou systèmes d'information. Il permet d'obtenir les données vérifiant certaines Conditions (on Parle de critères de sélection). Les données peuvent être triées, elles peuvent également être regroupées suivant les valeurs d'une donnée particulière.

SQL est Le langage de requête le plus connu et le plus utilisé.

### V.4. 2. Outils et logiciel :



Figure 16: Logo MySQL

MySQL est un Système de Gestion de Base de Données (SGBD) largement utilisé, disponible sous GNU GPL et licence propriétaire selon l'utilisation. Lancé en 1995, il est intégré aux stacks de développement comme WAMP, LAMP et MAMP, notamment pour ses performances avec PHP sur les serveurs web.

Compatible avec Linux, macOS, Windows, Solaris et FreeBSD, MySQL supporte divers langages dont PHP, Java, Ruby, C, C++, .NET et Python. Sa flexibilité réside dans la capacité à utiliser différents moteurs de bases de données, tels que MyISAM et InnoDB, optimisant ainsi les performances des applications.

MySQL offre également la réplication, permettant de répartir la charge, améliorer les performances et assurer la sauvegarde des données. Globalement, MySQL est reconnu pour sa fiabilité et sa capacité à répondre aux besoins variés des applications nécessitant une gestion efficace des données relationnelles.



Figure 17: Logo Uml

L'Unified Modeling Language (UML), ou langage de modélisation unifié, est conçu comme un langage visuel et sémantiquement riche pour la modélisation des architectures, conceptions et implémentations de systèmes logiciels complexes. Utilisé non seulement dans le développement logiciel mais aussi dans l'industrie pour les flux de processus, l'UML se compose de divers types de diagrammes qui décrivent la structure, les limites et le comportement des systèmes et des objets qu'ils contiennent.

Bien que l'UML ne soit pas un langage de programmation, il est souvent utilisé avec des outils capables de générer du code dans plusieurs langages à partir des diagrammes UML. Il est étroitement lié à l'analyse et à la conception orientées objet, fournissant un cadre standardisé pour la visualisation et la communication des concepts complexes de développement de logiciels et de systèmes.



Figure 18: Logo StartUml

StarUML est un logiciel de modélisation UML qui, après avoir été commercial, est désormais disponible sous une licence modifiée de la GNU GPL, bien que la version récente, StarUML V3, soit uniquement proposée en tant que produit commercial. Il supporte la majorité des diagrammes spécifiés dans la norme UML 2.0 et est développé en utilisant Delphi, reposant sur des composants propriétaires qui ne sont pas open source.).



Visual Studio Code



Figure 19: Logo Visual Studio Code

Visual Studio Code est un éditeur de code multiplateforme édité par Microsoft. Cet outil destiné aux développeurs supporte plusieurs dizaines de langages de programmation comme le HTML, C++, PHP, JavaScript, Markdown, CSS, etc.

Visual Studio Code intègre plusieurs outils facilitant la saisie de code par les développeurs comme la coloration syntaxique ou encore le système d’auto-complétion IntelliSense. En outre, l'outil permet aux développeurs de corriger leur code et de gérer Les différentes versions de leurs fichiers de travail puisqu'un module de débogage est aussi de la partie.



Figure 20: Logo Android Studio

Android Studio est un environnement de développement intégré (IDE) conçu pour créer des applications mobiles Android. Basé sur IntelliJ IDEA et soutenu par Google, il utilise le moteur Gradle pour la gestion des projets et des dépendances. Disponible sur Windows, macOS, Chrome OS et Linux, Android Studio offre des outils complets :

* Émulateurs Android pour tester sur différents appareils virtuels.
* Assistant de création de projet pour configurer automatiquement les fichiers nécessaires.
* Débogueur avancé pour détecter les erreurs et surveiller les performances.
* Éditeur de mise en page graphique avec aperçu en temps réel pour concevoir l'interface utilisateur.
* Intégration Firebase pour des fonctionnalités comme l'analyse et le stockage de données.

Android Studio est essentiel pour les développeurs Android, bénéficiant du soutien continu de Google et de la communauté open-source.



XAMPP



Figure 21: Logo Xampp

XAMPP est une suite logicielle complète conçue pour faciliter la mise en place d'un serveur web local, d'un serveur FTP et d'un serveur de messagerie électronique. Cette distribution, basée sur les logiciels libres (X pour cross-platform, Apache, MariaDB, Perl, PHP), est appréciée pour sa simplicité d'installation et sa rapidité d'exécution.

Elle permet à un large éventail d'utilisateurs, même sans connaissances avancées, de créer un environnement de développement ou de test sur leurs systèmes d'exploitation courants. XAMPP offre ainsi une flexibilité notable pour le déploiement de sites web et d'applications locales, tout en simplifiant la gestion des services essentiels comme les bases de données et les services web.



GIT



Figure 22: Logo Git

Git est un logiciel de gestion de versions décentralisé, créé par Linus Torvalds, célèbre pour avoir développé le noyau Linux. Distribué sous les termes de la licence publique générale GNU version 2, Git est un logiciel libre largement utilisé dans le développement de logiciels.



Figure 23: Logo GitHub

GitHub est une plateforme de gestion de projets web essentielle, intégrant Git pour la gestion avancée des versions de code. Elle permet l'hébergement sécurisé de projets et facilite le travail collaboratif grâce à des fonctionnalités comme les pull requests, les revues de code et les discussions sur les modifications. GitHub est largement adopté dans la communauté open source pour sa capacité à encourager la contribution et la collaboration à grande échelle.

La plateforme offre une version de base gratuite avec des outils robustes pour la gestion complète des cycles de développement, y compris GitHub Actions pour l'automatisation des workflows. Pour les entreprises, GitHub Enterprise propose des solutions personnalisées avec des options de sécurité avancées et un support premium, répondant ainsi aux besoins des grandes organisations pour gérer leurs projets de manière efficace et sécurisée.

# Chapitre VI. **PRESENTATION DU SYSTḔME**