Chapitre 4 **Conception**

Après avoir spécifié et analysé les différents besoins à satisfaire, ce chapitre s’intéressera à présenter l’aspect conceptuel, La conception est un processus créatif permettant la description du fonctionnement du système afin d’en faciliter la réalisation. Elle constitue une étape préliminaire préparatoire de la phase d’implémentation de la solution adoptée.

4.1 Conception globale

La planification de l'architecture des applications est une étape importante dans le processus de conception. Elle dépend de divers facteurs tels que les exigences de performance, l'extensibilité, la modularité et les perspectives. Dans cette partie, nous allons donc décrire l'architecture physique et logique sur laquelle nous allons élaborer notre projet.

4.1.3 Architecture physique

La figure ci-dessous montre une représentation de l'architecture physique de notre application, composée principalement d'une Couche de présentation, d'un serveur d'application et d'un serveur de base de données. 

▪ **Couche de présentation** (ou affichage si l'on souhaite) : correspondant à l'[affichage](https://www.techno-science.net/definition/4195.html) qui interagit avec l'utilisateur via un navigateur web.

▪ Le **serveur d’application:** C’est le serveur qui contient l'ensemble des règles de gestion et de la logique applicative.

▪ Le **serveur de base de données:** c’est un serveur de base de données utilisé pour stocker la partie persistance des données de notre application.

4.1.2 Architecture logique

La Figure suivante illustre une représentation de l'architecture logique de notre application qui est l’architecture multicouche, Elle nous donne un aperçu sur la relation entre les différentes parties.

Une image contenant texte

Description générée automatiquement

La structuration de notre application se traduit par une décomposition logique de chaque application en 5 couches :

• Présentation : La couche présentation gère et assure l'affichage de l'interface graphique utilisateur ou les Interfaces Homme-Machine IHM

• Contrôleur : le contrôle de la cinématique des écrans et l’invocation des appels de services

• Services : correspond aux traitements qu’effectue l’application elle représente l’implémentation de la logique des cas d’utilisation use-case fonctionnels.

• Domaine : responsable de la gestion des objets métiers, elle recense les objets métiers manipulées par l’application

• Persistance : fournit des services de stockage des données et de mapping entre les différentes formes des objets métiers du domaine.

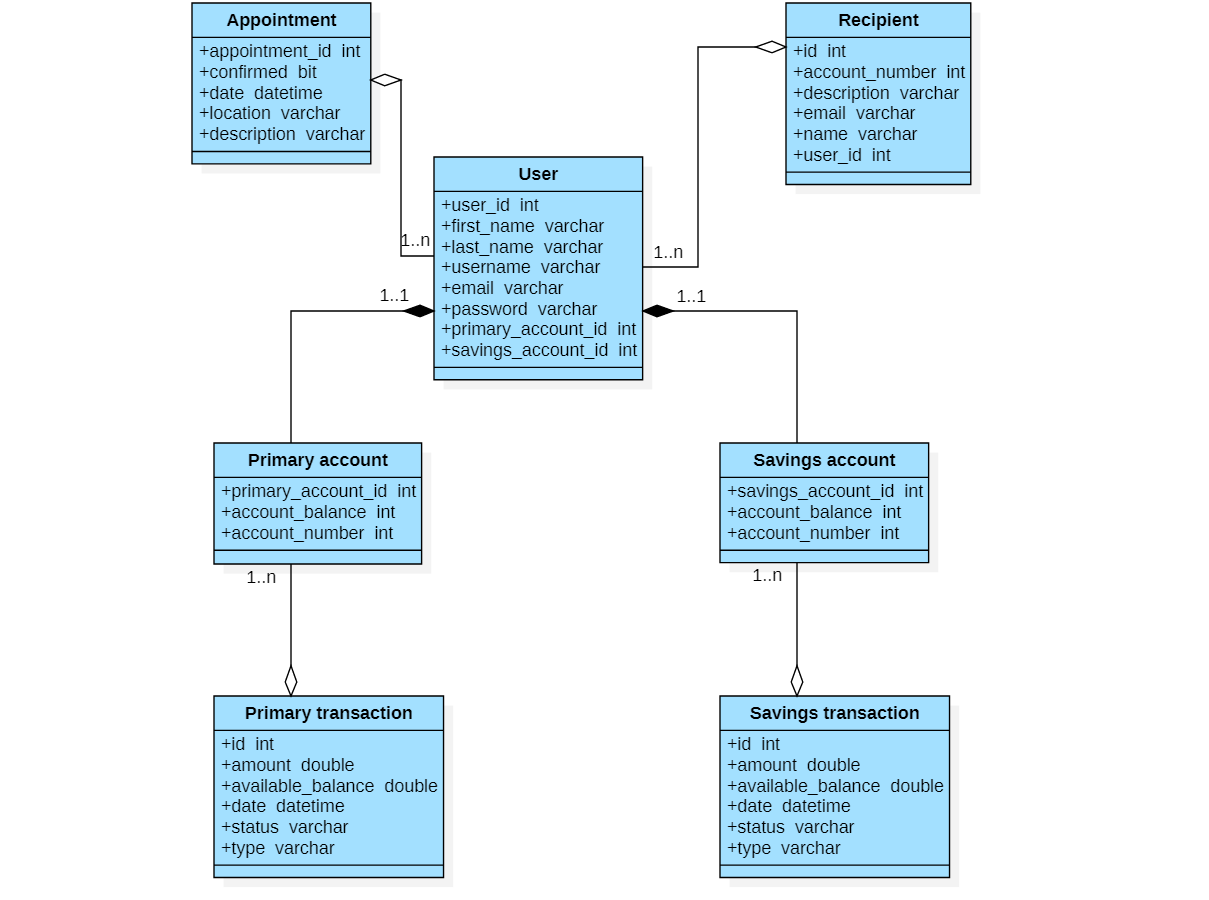
Chaque couche a ses propres responsabilités et utilise la couche située en dessous d’elle

4.2 Conception détaillée

Cette partie a pour objectif de transformer le modèle d’analyse en un modèle concret qui aide le développeur à réaliser son application, nous présentant dans ce qui suit une vue statique et dynamique de notre application moyennant les diagrammes appropriés.

4.2.2 Diagramme de classes

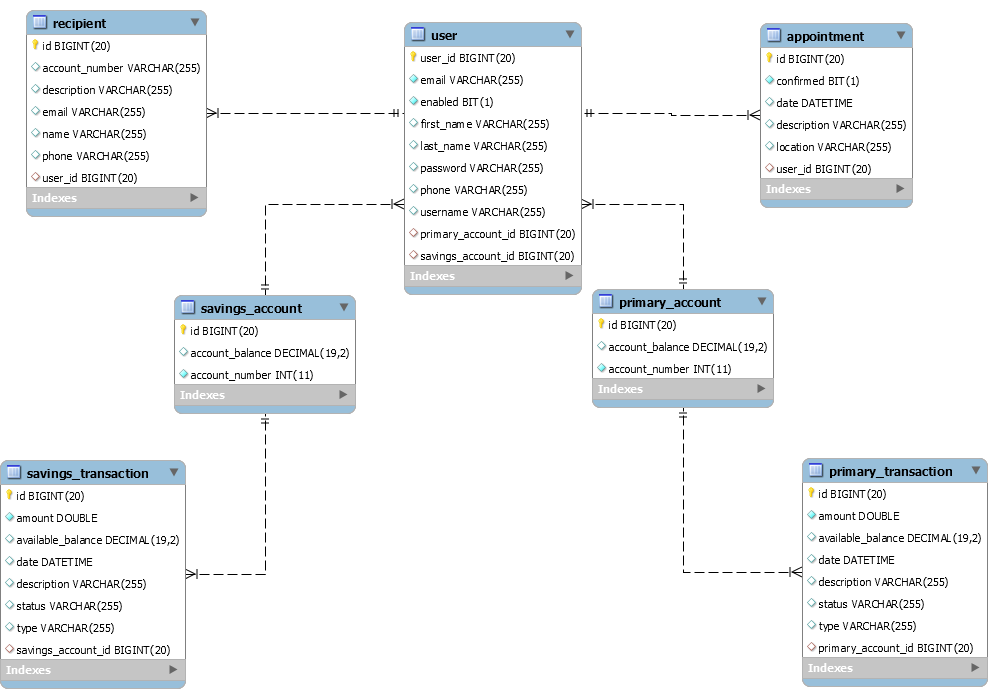
La figure suivante présente le diagramme de classes relatif à notre application.



Digramme de classes

4.2.1 Conception de la base de données

La figure suivante est un [modèle de données](https://fr.wikipedia.org/wiki/Mod%C3%A8le_de_donn%C3%A9es) ou [diagramme](https://fr.wikipedia.org/wiki/Diagramme) pour des descriptions de haut niveau de [modèles conceptuels de données](https://fr.wikipedia.org/wiki/Sch%C3%A9ma_conceptuel)



Digramme entité association

4.3 Conception de l’entrepôt de données

Dans un contexte d’un projet décisionnel, notre objectif c’est d’arriver à restaurer un ensemble de tableaux de bords qui se basent sur des indicateurs de performance (KPI) générés à partir de données instaurées dans l’entrepôt de données. Pour se faire, il faut que nous identifiions le modèle conceptuel de l’entrepôt et les dimensions et les mesures retenus.

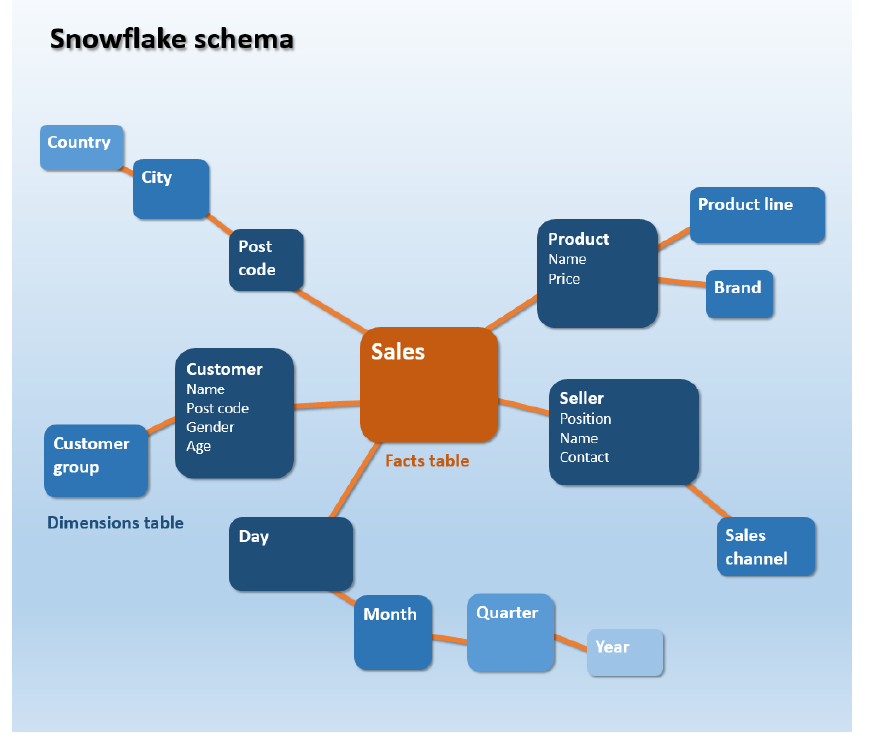
**Étude du modèle conceptuel**

Il existe 3 modèles conceptuel d’entrepôt de données qui sont :

* Modèle en étoile
* Modèle en flocon de neige
* Modèle en constellation

Pour notre solution, nous avons opté pour le modèle en flocon de neige qui permet de formaliser une hiérarchie de dimensions en fonction de la granularité de l’information qui sont reliées entre elles par un ensemble de relations jusqu’à arriver à la granularité la plus fine, ce model permet de s’approfondir dans les détails, ce qui facilite l’analyse et réduit le volume des données.

La suivante représente un exemple de modèle en flocon de neige



Modèle en flocon de neige

Choix des dimensions

On entend par dimensions, tous les axes ou thèmes suivant lesquels les données seront analysées. On attribue une table à chaque dimension ce qui implique qu’il existe autant de tables de dimension que de dimensions.

Une table de dimension comporte des attributs et une clé primaire indépendante des attributs. Notre modèle comportera les dimensions suivantes :

* Dimension date
* Dimension user
* Dimension transactions
* Dimension primary account

Choix des mesures

Pour préparer la table de faits, il faudra bien se servir des éléments recueillis et collectés lors de la phase de préparation et étude des données sources. Il faudra alors préciser les éléments que nous souhaitons mesurer.

Dans notre table de faits, nous avons retenu les mesures suivantes :

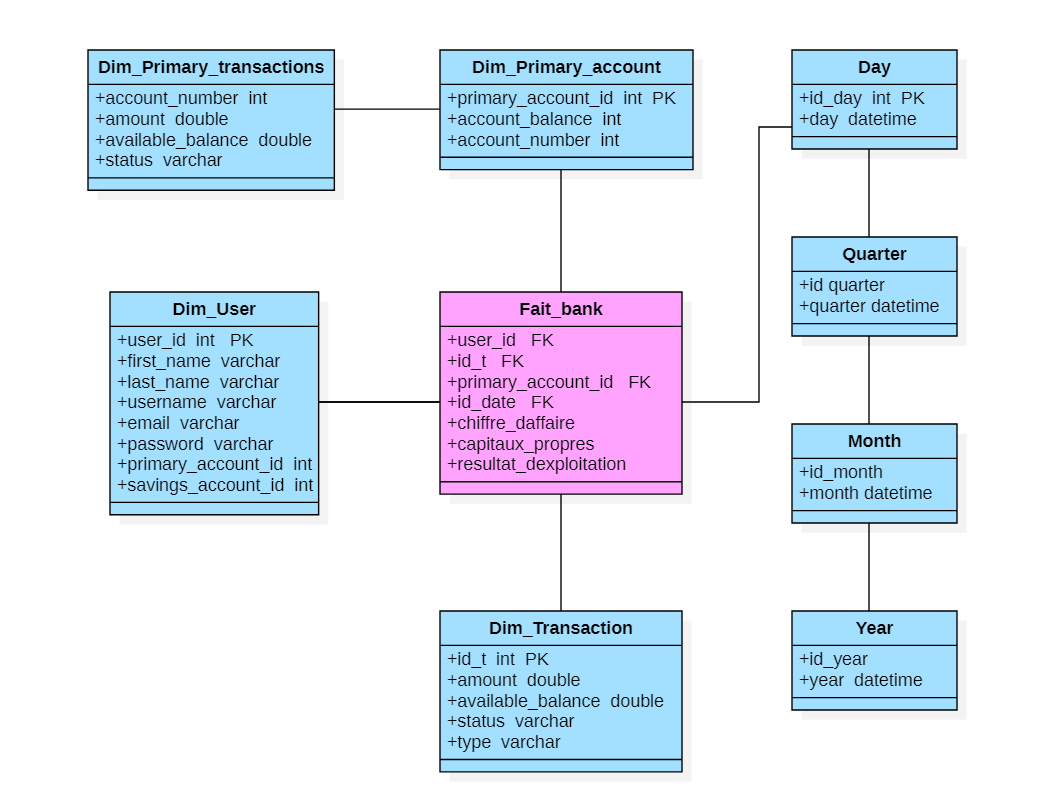
* Chiffre d’affaires
* Capitaux propres
* Résultat d’exploitation

Table de faits

La table de fait contient les données observables (les faits) que l’on possède sur un sujet et que l’on veut étudier, selon divers axes d’analyse (les dimensions), notre table de fait, elle est constituée d’une table ou sont spécifiées les clés des dimensions sous forme de clés étrangères et aussi des mesures.

4.3.2 Modélisation du Data Warehouse

Le schéma conceptuel final de l’entrepôt de données est le suivant :



Le modèle conceptuel en flocon de neige du Data Warehouse

4.2.4 Diagrammes de séquences

Diagrammes d’activité