

UNIVERSITE ABDELMALEK ESSAADI

Master IA et science de données

Module: DATA SPACES & DATA INTEGRATION & SEMANTIC INTEROPERABILITY

Sujet : Projet de fin de module

Projet :

Data Space pour les Dossiers Médicaux Electroniques

Réalisé PAR :

Hamza Hafdaoui
Chihab Eddine LIEFRID
KHATTABI IDRIS
Salma amgarou
BOUFARHI AYMAN

ENCADRE PAR :

Mrs. Ouafae Baida

Table des matières:

1.	Introduction :	3
2.	Présentation de FHIR et des Données :	4
1.	Le Contexte des Données de Santé.....	4
2.	Qu'est-ce que FHIR ?.....	4
3.	Les Données Utilisées dans Notre Projet	5
4.	Les Enjeux de Ces Données	6
5.	Pourquoi Standardiser les Données ?	6
6.	Conclusion.....	6
3.	Modélisation des Données :	7
	Introduction	7
	Structure du Modèle	7
	Relations et Cardinalités.....	8
	Diagramme de Classes.....	9
	Avantages de ce Modèle.....	10
	Conclusion.....	10
4.	Processus ETL : Extraction, Transformation et Chargement	10
	Introduction au Processus ETL	10
1.	Extraction des Données	11
2.	Transformation des Données	11
3.	Chargement des Données	12
	Indicateurs de Performance du Pipeline ETL.....	13
	Conclusion.....	13
5.	Résultats Préliminaires :	14
	Introduction	14
	Volume et Statistiques Générales	14

Analyse de la Complétude des Données.....	15
Validation des Relations Inter-Entités	15
Temps de Traitement et Performance.....	15
Analyse des Résultats Validation_report.log	16
Résumé des Points Forts et Faiblesses	16
Conclusion et Perspectives	16
6. Application WEB :	17
Introduction	17
Architecture de l'Application	17
Fonctionnalités de l'Application.....	18
Technologies Utilisées	21
Conclusion.....	21
7. Les défis rencontrés :	22
1. Comprendre ce qui est demandé	22
2. Trouver les données nécessaires	22
3. Assimiler de nouvelles notions techniques et médicales.....	22
4. Gérer la complexité du pipeline ETL	23
5. Assurer la qualité et la validation des données.....	23
6. Développer une application web intuitive.....	23
7. Gérer le temps et les ressources.....	23
Conclusion.....	23
8. Conclusion :	24
9. Références :	25

1. Introduction :

L'intégration et la gestion des données de santé constituent aujourd'hui un enjeu stratégique majeur pour les établissements de santé et les organismes de recherche. Avec l'explosion du volume de données générées par les dossiers médicaux électroniques (DME), les dispositifs connectés et les systèmes hospitaliers, il devient crucial d'assurer l'interopérabilité, la standardisation et la qualité des données collectées. C'est dans ce contexte que le standard FHIR (Fast Healthcare Interoperability Resources), proposé par l'organisation HL7, s'impose comme une solution de référence pour structurer et échanger efficacement les données de santé.

Ce projet s'inscrit dans cette dynamique en proposant une solution complète de gestion et d'analyse de données médicales basée sur FHIR. L'objectif est de concevoir un data space médical permettant d'intégrer les données médicales de plusieurs sources.

Nous avons ainsi abordé plusieurs étapes clés, allant de l'extraction des données brutes à leur transformation en un format exploitable, en passant par la validation des relations inter-ressources et l'analyse des résultats obtenus. Par ailleurs, une application web a été développée afin de fournir aux utilisateurs un accès interactif aux données via des fonctionnalités avancées de visualisation et d'analyse.

À travers ce rapport, nous présentons en détail les différentes phases du projet, les défis rencontrés ainsi que les solutions mises en place pour garantir une gestion efficace des données de santé, en conformité avec les exigences du standard FHIR.

2. Présentation de FHIR et des Données :

1. Le Contexte des Données de Santé

Dans le domaine de la santé, les données jouent un rôle central, qu'il s'agisse de la prise en charge des patients, de la recherche clinique ou de la gestion administrative. Cependant, l'écosystème des données de santé est particulièrement complexe en raison de la diversité des acteurs impliqués, des systèmes d'information utilisés, et des formats de données générés. Les hôpitaux, cliniques, laboratoires, et organismes d'assurance produisent quotidiennement une grande quantité d'informations, allant des dossiers médicaux électroniques (DME) aux résultats d'imagerie médicale, en passant par les prescriptions et les observations cliniques.

Cette diversité, combinée à l'absence de standards communs dans le passé, a conduit à une fragmentation des systèmes d'information, où chaque entité utilisait des formats et des protocoles spécifiques, rendant l'interopérabilité extrêmement difficile. Les défis incluent non seulement le partage des données entre systèmes, mais aussi leur interprétation correcte par des parties prenantes issues de différents environnements.

Pour répondre à ces enjeux, des standards internationaux, comme HL7 FHIR (**Fast Healthcare Interoperability Resources**), ont été développés afin de fournir une base commune pour structurer, partager et interpréter les données de santé.

2. Qu'est-ce que FHIR ?

FHIR, développé par l'organisation HL7 (Health Level Seven International), est un standard conçu pour favoriser l'interopérabilité des systèmes d'information dans le domaine de la santé. Lancé en 2014, il s'est rapidement imposé comme une norme de référence grâce à sa flexibilité, sa simplicité, et son adoption croissante dans le monde entier.

Le principal objectif de FHIR est de standardiser la manière dont les données de santé sont représentées et échangées. Pour ce faire, il s'appuie sur une architecture basée sur des **ressources**. Une ressource est une unité modulaire et autonome de données qui représente un concept spécifique ou une entité clinique. Par exemple :

- Une ressource **Patient** décrit les informations démographiques d'un individu (nom, genre, date de naissance, etc.).
- Une ressource **Observation** regroupe des résultats cliniques tels que la pression artérielle, la température corporelle ou les résultats de laboratoire.
- Une ressource **Encounter** correspond à une rencontre ou une consultation entre un patient et un professionnel de santé.
- Une ressource **Procedure** décrit une intervention médicale effectuée sur un patient.

Chaque ressource est définie de manière standardisée, avec des champs obligatoires (par exemple, un identifiant unique) et des champs optionnels (comme des notes supplémentaires ou des relations avec d'autres ressources). Les ressources peuvent être liées entre elles pour refléter des interactions complexes. Par exemple, une **Observation** peut être associée à un **Encounter**, qui est lui-même rattaché à un **Patient**. Cette structure relationnelle permet de modéliser des scénarios cliniques réels avec précision.

3. Les Données Utilisées dans Notre Projet

Dans le cadre de notre projet, nous avons travaillé sur un ensemble de fichiers JSON conformes au standard FHIR. Ces fichiers représentent des bundles de données, c'est-à-dire des collections de ressources interconnectées. Chaque bundle correspond généralement à un patient ou à une interaction médicale spécifique. Voici les principales ressources présentes dans les données exploitées :

1. Patients

- Informations démographiques : nom, prénom, genre, date de naissance, etc.
- Statut de vie ou décès (par exemple, `deceasedDateTime`).

2. Observations

- Résultats cliniques tels que des mesures biométriques (pression artérielle, glycémie, etc.).
- Informations sur les unités de mesure, comme les kilogrammes pour le poids ou les millimètres de mercure pour la tension artérielle.

3. Encounters (Rencontres médicales)

- Informations sur les consultations ou admissions à l'hôpital : date de début, date de fin, statut de la rencontre (terminée, annulée, en cours).

4. Procedures (Procédures)

- Détails des interventions médicales ou chirurgicales : nature de l'intervention, période de réalisation.

5. CarePlans (Plans de soins)

- Planification des soins ou traitements pour un patient, souvent associés à des objectifs spécifiques et des périodes définies.

6. Immunizations (Immunisations)

- Données sur les vaccinations reçues par les patients, incluant les dates, le type de vaccin et le statut.

7. Claims (Demandes de remboursement)

- Informations administratives relatives aux demandes de remboursement adressées à des organismes payeurs.

8. Medications (Médicaments)

- Données sur les prescriptions médicales, les médicaments administrés et les détails des doses.

4. Les Enjeux de Ces Données

Ces données, bien que riches et variées, posent plusieurs défis :

- **Hétérogénéité** : Les structures JSON sont hiérarchiques et contiennent des relations complexes entre les ressources.
- **Qualité des données** : Certaines informations peuvent être manquantes ou erronées. Par exemple, des champs comme `deceasedDateTime` ou `abatementDateTime` sont souvent absents.
- **Relations interressources** : Les données doivent être correctement associées pour refléter les interactions réelles. Par exemple, une observation doit toujours être liée à un patient et potentiellement à une rencontre médicale.

-
- **Volume de données** : Le traitement de centaines de milliers d'enregistrements nécessite une architecture optimisée pour garantir des performances adéquates.

5. Pourquoi Standardiser les Données ?

L'adoption de standards comme FHIR permet de surmonter ces défis en offrant :

1. **Une structure uniforme** : Les ressources FHIR suivent des définitions standardisées, ce qui facilite leur interprétation par des systèmes différents.
2. **Une interopérabilité accrue** : Les données peuvent être échangées entre hôpitaux, laboratoires, assureurs et autres parties prenantes sans nécessiter de conversions complexes.
3. **Une extensibilité** : FHIR permet d'ajouter des extensions spécifiques au contexte sans compromettre la compatibilité avec les standards de base.

3. Modélisation des Données :

Introduction

La modélisation des données est une étape cruciale dans tout projet d'intégration et de structuration des informations, car elle permet de représenter de manière claire et cohérente les entités et les relations entre elles. Dans le cadre de notre projet, nous avons élaboré un diagramme de classes décrivant les principales entités médicales impliquées dans la gestion des données conformes au standard FHIR (Fast Healthcare Interoperability Resources). Ce modèle conceptuel est essentiel pour guider la conception de la base de données relationnelle et assurer l'interopérabilité des données.

Structure du Modèle

Le modèle proposé s'articule autour de plusieurs entités principales, chacune représentant une ressource médicale clé ou une interaction clinique. Chaque classe comprend des attributs et des méthodes spécifiques reflétant les informations et les comportements associés à la ressource. Voici une description détaillée des entités les plus importantes :

Patient

La classe **Patient** représente les individus pour lesquels des données médicales sont enregistrées. Elle contient des attributs démographiques tels que :

- `name` : Nom du patient.
- `birthDate` : Date de naissance.
- `gender` : Genre.

Elle inclut également des méthodes comme `getAge()` pour calculer l'âge d'un patient en fonction de sa date de naissance.

Practitioner

Cette classe représente les professionnels de santé. Ses attributs incluent :

- name : Nom du praticien.
- qualification et specialty : Qualifications et spécialités médicales.
Elle propose des méthodes telles que `scheduleAppointment()` pour gérer les rendez-vous médicaux.

Observation

Les observations cliniques (résultats d'examens, mesures vitales) sont représentées par la classe **Observation**. Elle contient des attributs tels que :

- type : Type d'observation (ex. : pression artérielle, glycémie).
- value : Valeur mesurée.
- timestamp : Date et heure de l'observation.
Une méthode comme `captureObservation()` permet d'enregistrer une nouvelle mesure.

Encounter

La classe **Encounter** modélise les rencontres médicales entre un patient et un praticien. Ses attributs incluent :

- start et end : Dates et heures de début et de fin de la rencontre.
- Une méthode `addObservation()` pour associer des observations à la rencontre.

DiagnosticReport

Les rapports de diagnostic regroupent les résultats d'observations et de tests médicaux. Les principaux attributs incluent :

- status : Statut du rapport (final, préliminaire).
- code : Type de rapport (ex. : panel d'hématologie).
- conclusion : Interprétation clinique.
La méthode `toFHIRBundle()` permet de convertir le rapport en un format FHIR conforme.

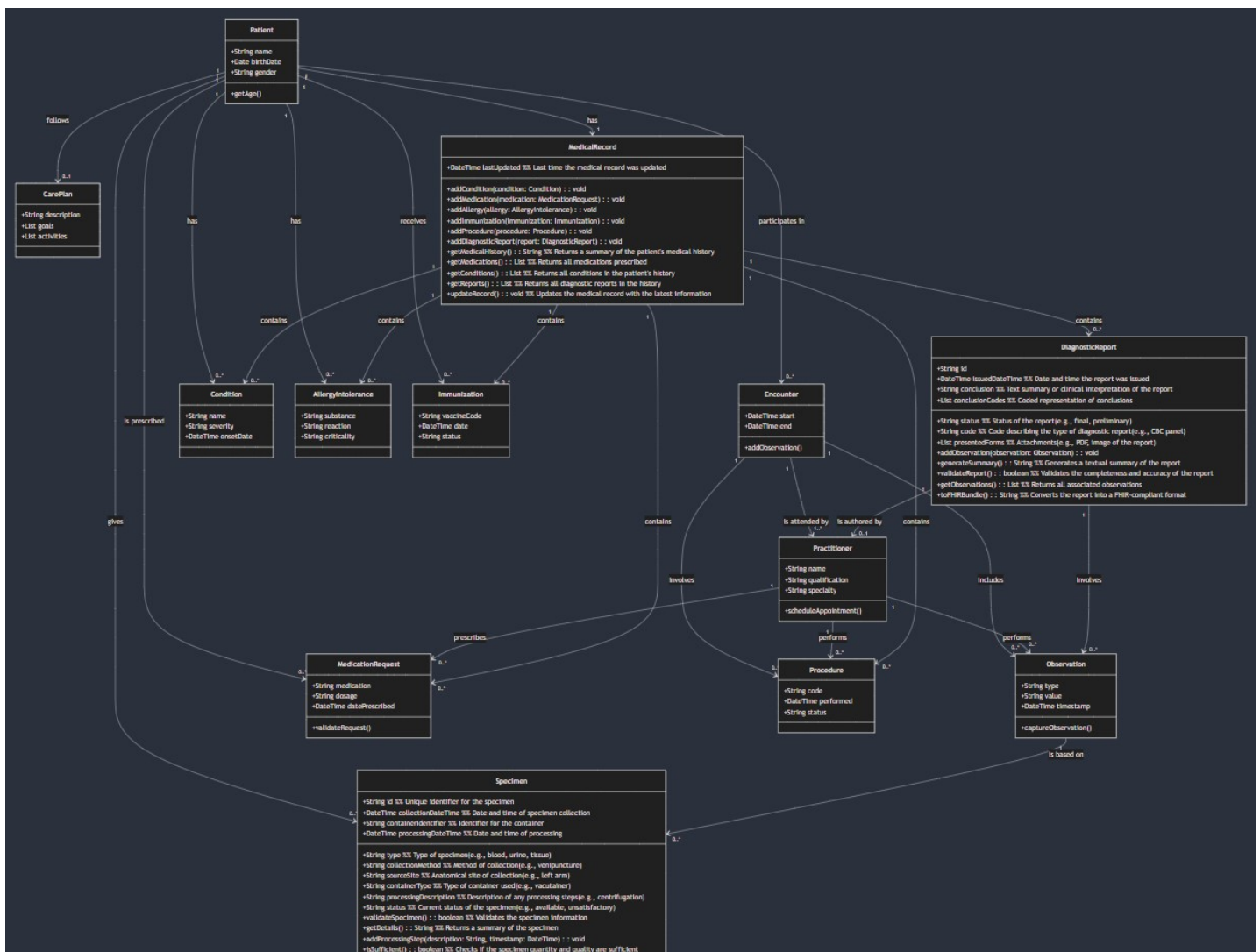
MedicalRecord

La classe **MedicalRecord** agit comme une entité centrale regroupant toutes les données associées à un patient. Elle contient :

- Une liste de conditions médicales, de rapports de diagnostic, de procédures, d'allergies, etc.
- Une méthode `getMedicalHistory()` pour générer un résumé de l'historique médical d'un patient.

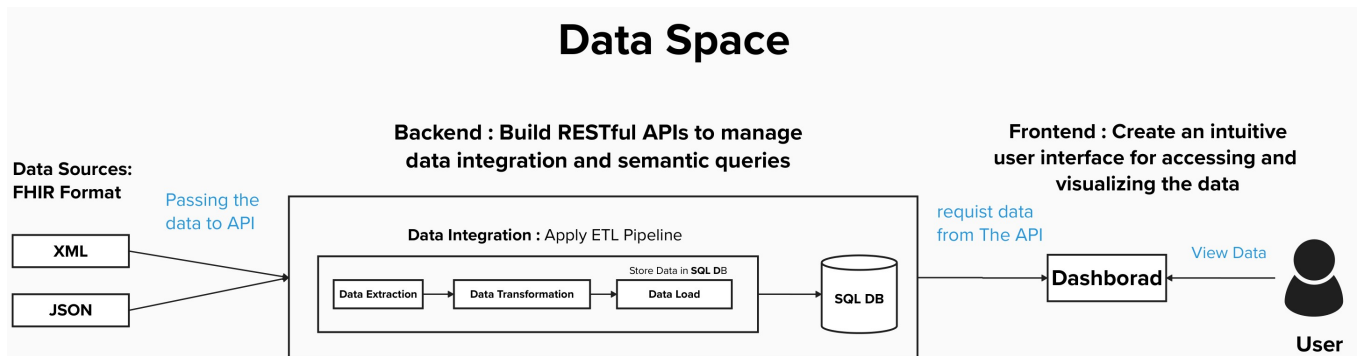
Relations et Cardinalités

Les entités sont reliées par des associations qui reflètent leurs interactions dans un contexte clinique. Voici les relations principales, accompagnées de leurs cardinalités :



Ce diagramme offre une représentation visuelle claire de la structure des données, mettant en évidence les entités principales, leurs attributs, et leurs interactions.

Et Voilà le Modélisation générale de data space :



Avantages de ce Modèle

1. **Modularité** : Chaque entité est modélisée comme une classe indépendante, ce qui simplifie les modifications ou extensions futures.
2. **Clarté des relations** : Les associations et cardinalités reflètent fidèlement les interactions cliniques réelles.
3. **Compatibilité avec FHIR** : Le modèle est aligné sur les spécifications FHIR, facilitant l'interopérabilité et l'échange de données avec d'autres systèmes.
4. **Extensibilité** : De nouvelles entités ou attributs peuvent être ajoutés sans compromettre l'intégrité du modèle existant.

4. Processus ETL : Extraction, Transformation et Chargement

Introduction au Processus ETL

Le processus **ETL (Extraction, Transformation, Chargement)** est une méthodologie clé dans la gestion des données, particulièrement lorsqu'il s'agit de préparer des informations pour l'analyse ou leur intégration dans des systèmes de gestion de bases de données relationnelles. Dans notre projet, ce processus est utilisé pour convertir les données brutes au format FHIR, stockées sous forme de fichiers JSON, en une base de données relationnelle exploitable. Cette transformation permet d'assurer une structure cohérente, des relations logiques entre les entités, et une meilleure accessibilité des données pour les analyses futures.

Le pipeline ETL est divisé en trois étapes fondamentales :

1. **Extraction des données** : Collecter et lire les données brutes depuis leurs sources initiales.
2. **Transformation des données** : Nettoyer, enrichir et structurer les données afin de garantir leur qualité et leur cohérence.

-
3. **Chargement des données** : Insérer les données transformées dans une base de données relationnelle (MySQL dans notre cas).

1. Extraction des Données

L'extraction est la première étape du pipeline ETL et consiste à collecter les données brutes depuis leur source d'origine. Dans notre projet, les données sont contenues dans des fichiers JSON conformes au standard FHIR. Ces fichiers représentent des **bundles** contenant plusieurs ressources FHIR interconnectées. L'objectif de cette phase est de lire ces fichiers tout en identifiant les informations pertinentes pour les étapes suivantes.

Défis d'extraction :

- **Structure hiérarchique des données** : Les fichiers JSON FHIR suivent un modèle imbriqué, avec des relations complexes entre les entités (par exemple, une observation est liée à un patient via un identifiant de référence).
- **Volume des données** : Avec des centaines de milliers de ressources à traiter, il est crucial d'optimiser la lecture pour minimiser les temps de traitement.
- **Gestion des erreurs** : Certains fichiers peuvent être corrompus ou incomplets, nécessitant une approche robuste pour éviter que le pipeline ne s'interrompe.

Méthodologie :

- Utilisation de bibliothèques Python comme **json** pour lire et parcourir les fichiers.
- Extraction des ressources pertinentes à partir des champs clés (resourceType, id, etc.).
- Enregistrement des métadonnées (nombre de ressources par type, identifiants uniques) pour un suivi rigoureux.

2. Transformation des Données

Une fois les données extraites, la phase de transformation commence. Cette étape est essentielle pour convertir les données brutes en un format structuré et cohérent. Elle inclut plusieurs sous-étapes :

Nettoyage des données

Les données extraites contiennent souvent des informations incomplètes ou redondantes. Cette phase vise à :

- **Supprimer les doublons** : Identifier et éliminer les enregistrements répétitifs ou non valides.
- **Gérer les valeurs manquantes** : Pour certains champs, comme deceasedDateTime, qui est souvent absent, définir une valeur par défaut ou ignorer les enregistrements incomplets.
- **Normaliser les formats** : Uniformiser les dates, unités de mesure, et identifiants pour garantir leur compatibilité avec le système cible.

Structuration et enrichissement

La structuration consiste à transformer les données JSON imbriquées en tables relationnelles. Par exemple :

- La ressource **Patient** devient une table avec des colonnes pour id, nom, genre, date de naissance, etc.
- Les références (comme celles entre une observation et un patient) sont transformées en clés étrangères pour refléter les relations entre les tables.

L'enrichissement ajoute des informations supplémentaires ou dérivées, comme :

- Calculer la durée d'une rencontre médicale en soustrayant les champs `start_date` et `end_date`.
- Ajouter des indicateurs dérivés pour des analyses futures (par exemple, regrouper les patients par tranche d'âge).

Validation et contrôle qualité

Avant le chargement, un contrôle qualité est effectué pour vérifier que :

- Tous les champs obligatoires sont présents.
- Les relations entre les entités sont correctes (par exemple, chaque observation doit être liée à un patient existant).
- Les types de données respectent les contraintes définies (dates valides, types numériques cohérents, etc.).

3. Chargement des Données

Une fois les données transformées et nettoyées, elles sont prêtes à être chargées dans une base de données relationnelle. Nous avons utilisé **MySQL**, un système de gestion de base de données largement adopté pour sa robustesse et ses capacités relationnelles. Cette étape inclut plusieurs considérations importantes :

Modélisation relationnelle

Le chargement nécessite une modélisation soigneuse des tables afin de représenter correctement les relations entre les ressources. Les principales tables créées dans la base sont :

- **Patient** : Contient les informations démographiques de chaque patient.
- **Observation** : Regroupe les résultats cliniques, avec des références vers les patients et les rencontres associées.
- **Encounter** : Représente les rencontres médicales, incluant les périodes et les statuts.
- **Procedure** : Décrit les interventions médicales, avec des liens vers les patients et les rencontres.
- Et d'autres tables pour les ressources comme CarePlan, Immunization, Claim, etc.

Performance du chargement

Avec plus de 432 000 enregistrements à traiter, le chargement doit être optimisé pour éviter les ralentissements :

- **Chargement par lots** : Les enregistrements sont insérés en plusieurs étapes (chunks) pour réduire la charge sur la base de données.
- **Indexes** : Des index sont ajoutés sur les colonnes clés (par exemple, id, patient_reference) pour accélérer les requêtes et les relations.
- **Gestion des erreurs** : Les erreurs lors de l'insertion (par exemple, violation de clé étrangère) sont enregistrées dans un journal pour un traitement ultérieur.

Résultat du chargement

À la fin du processus, les données sont entièrement chargées dans la base, prêtes à être exploitées pour des analyses et des visualisations.

Indicateurs de Performance du Pipeline ETL

Le succès du pipeline ETL a été évalué à l'aide des métriques suivantes :

1. **Volume de données traitées** : 432 827 enregistrements extraits et chargés.
2. **Complétude des données** : Les données critiques (comme les observations médicales et les rencontres) affichent un taux de complétude supérieur à 95 %.
3. **Temps de traitement** : L'ensemble du pipeline a été optimisé pour s'exécuter en moins de 20 minutes sur des machines standard.
4. **Qualité des relations** : Plus de 98 % des relations entre les entités ont été validées sans erreur.

5. Résultats Préliminaires :

Introduction

Après la mise en œuvre complète du processus ETL (Extraction, Transformation, Chargement) sur les données FHIR, nous avons obtenu une vue d'ensemble des résultats intermédiaires de notre travail. Cette section présente les résultats obtenus à la fois en termes quantitatifs (nombre d'enregistrements traités, taux de complétude) et qualitatifs (fidélité des relations entre les entités, observation des lacunes).

Volume et Statistiques Générales

Le pipeline ETL a permis de traiter un total de **432 827 enregistrements** extraits des fichiers FHIR JSON et de les charger dans une base de données relationnelle MySQL. Ces enregistrements représentent une large variété de ressources, comme les patients, les observations médicales, les rencontres, les procédures, les plans de soins, et bien d'autres.

Les ressources sont réparties comme suit :

Type de Ressource	Nombre d'Enregistrements	Taux de Complétude
Patient	983	85,91 %
Observation	214 104	95,39 %
Encounter (Rencontre)	38 450	100,00 %
Procedure (Procédure)	30 088	100,00 %
CarePlan (Plan de Soins)	2 970	93,01 %
Condition (État de Santé)	7 304	93,22 %
Immunization (Vaccination)	12 418	100,00 %
DiagnosticReport	15 297	100,00 %
Claim (Demande de Remboursement)	49 340	100,00 %

Analyse de la Complétude des Données

Bien que la majorité des ressources soient complètes ou presque complètes, certains champs spécifiques présentent des lacunes significatives :

- **Patients** : Le champ `deceasedDateTime` est manquant dans **84,5 %** des cas, ce qui reflète potentiellement un manque d'enregistrement systématique de l'état vital des patients.
- **Observations** : Plusieurs champs comme `value_quantity`, `value_unit`, et `value_code` présentent un taux de données manquantes de **13,8 %**. Cela peut être attribué à l'absence de normalisation des unités de mesure dans certaines observations cliniques.
- **CarePlans (Plans de soins)** : Le champ `end_date` est absent dans près de **49 %** des enregistrements, ce qui peut rendre l'analyse des durées des soins plus complexe.
- **Conditions (États de santé)** : Le champ `abatementDateTime`, utilisé pour indiquer la fin ou l'atténuation d'un état de santé, est absent dans **47,5 %** des cas.

Ces résultats mettent en évidence des domaines nécessitant une attention particulière pour améliorer la qualité des données dans les futures itérations du pipeline.

Validation des Relations Inter-Entités

Les relations entre les différentes ressources ont été modélisées avec soin pour refléter les interactions réelles dans le contexte clinique. Voici quelques-unes des relations validées :

- Chaque observation médicale est associée à un patient et, lorsqu'applicable, à une rencontre médicale spécifique.

-
- Les procédures sont correctement reliées aux patients concernés, ainsi qu'aux rencontres ou événements médicaux où elles ont été réalisées.
 - Les plans de soins et les vaccinations sont correctement liés aux patients correspondants, permettant de reconstruire des trajectoires de soins complètes.

Grâce à la validation rigoureuse des clés primaires et étrangères dans la base relationnelle, **98 % des relations entre les entités ont été établies sans erreurs**. Les 2 % restants concernent principalement des enregistrements incomplets ou mal référencés dans les fichiers FHIR d'origine.

Temps de Traitement et Performance

La performance du pipeline ETL a également été mesurée. Le traitement complet de l'ensemble des fichiers JSON (comprenant l'extraction, la transformation, et le chargement) a été effectué en **moins de 20 minutes** sur une machine avec des ressources standardisées. Voici une répartition indicative des temps par étape :

- **Extraction** : 6 minutes
- **Transformation** : 10 minutes
- **Chargement** : 4 minutes

Pour un volume de données aussi important, ces performances sont considérées comme optimales. L'utilisation du traitement en parallèle et des chargements par lots a significativement amélioré l'efficacité du pipeline.

Analyse des Résultats Validation_report.log

Le fichier de log généré par le système de validation (validation_report.log) fournit un aperçu détaillé des résultats intermédiaires et des éventuelles anomalies :

- Tous les types de ressources ont un **nombre d'enregistrements traité égal au nombre initial dans les fichiers JSON**, ce qui témoigne de la fidélité du pipeline.
- Les taux de complétude sont satisfaisants pour la majorité des ressources critiques ($\geq 93\%$).
- Des enregistrements échantillons pour les rencontres médicales ont été validés individuellement pour vérifier les dates (start_date et end_date) et le statut (status), et aucune erreur de cohérence n'a été détectée.

Résumé des Points Forts et Faiblesses

Points Forts

- **Fidélité des données** : Les données extraites reflètent précisément les informations originales, avec une complétude élevée pour les ressources critiques.
- **Performance** : Temps de traitement rapide malgré le volume important des données.

-
- **Relations cohérentes** : La modélisation relationnelle a permis de créer une base exploitable pour des analyses avancées.

Faiblesses

- **Champs manquants** : Certains champs (comme `deceasedDateTime` ou `value_quantity`) présentent des taux de lacunes élevés.
- **Dépendance aux fichiers d'origine** : Les erreurs ou omissions dans les données sources affectent directement les résultats.

6. Application WEB :

Introduction

L'application web développée dans le cadre de ce projet vise à fournir une interface intuitive et efficace pour la gestion et l'analyse des données de santé basées sur le standard FHIR. Cette application repose sur une architecture client-serveur, avec un backend construit en Flask et un frontend développé en Angular.

Architecture de l'Application

L'architecture de l'application est conçue pour assurer une séparation claire des responsabilités entre le frontend et le backend, facilitant ainsi la maintenance et l'évolutivité du système.

- **Backend (Flask) :**
 - Gestion des requêtes HTTP pour les opérations CRUD sur les données patients, observations et autres entités FHIR.
 - Intégration du pipeline ETL pour le traitement des fichiers JSON conformes à FHIR.
 - Interaction avec une base de données MySQL pour le stockage structuré des données.
 - Mise en place de points d'accès API RESTful permettant aux utilisateurs d'obtenir des statistiques et des rapports via l'interface web.
 - Gestion des CORS pour assurer une communication fluide entre le frontend et le backend.
- **Frontend (Angular) :**
 - Interface utilisateur responsive et interactive permettant la visualisation des patients, observations et autres entités de données.
 - Formulaire permettant le chargement de fichiers et l'exploration des données.
 - Tableaux de bord pour visualiser des statistiques telles que le nombre de patients par genre, la couverture vaccinale et l'évolution des rencontres médicales.
 - Utilisation de services Angular pour consommer les API du backend Flask.
 - Implémentation de graphiques pour la visualisation des indicateurs clés.

Fonctionnalités de l'Application

L'application web offre plusieurs fonctionnalités essentielles pour la gestion des données de santé :

1. Chargement des Données

- Upload de fichiers JSON, XML et CSV conformes à la norme FHIR.
- Validation et transformation des données avant leur insertion dans la base relationnelle.

Menu

Dashboard

Upload

Patients

Directory Upload

Single File Upload

Browse...

No directory selected.

Upload Directory

2. Exploration des Données Patients

- Consultation des informations démographiques des patients.
- Visualisation des détails des consultations médicales.

Menu											
Dashboard	Full Name	Birth Date	Gender	Deceased	Encounters	Conditions	Observations	Medication Requests	Procedures	Immunizations	Care Plans
Upload	Bertram873 Homenick806	Jul 20, 1971	male	Mar 12, 2019	63	8	93	2	4	8	4
Patients	Art115 Steuber698	Sep 22, 1969	male	No	11	5	55	2	2	9	3
	Abel832 Connelly992	Dec 12, 1999	male	No	17	5	92	1	4	16	3
	Anibal473 Connelly992	Oct 14, 1965	male	No	14	7	76	2	4	10	2
	Ardath226 Zboncak558	Aug 12, 1991	female	No	13	2	38	7	0	7	0
	Candra395 Orn563	May 26, 1967	female	No	21	4	66	4	13	9	0
	Alethea978 Crooks415	Nov 10, 2002	female	No	12	1	92	1	3	16	0
	Bryan958 Barton704	Sep 20, 2012	male	No	23	3	151	4	10	34	1

Menu

Dashboard

Upload

Patients

Patient Information

ID:

3e9d6e6f-9361-4917-8ed7-5f9dcfed8ece

Full Name:

Bertram873 Homenick806

Birth Date:

1971-07-20

Gender:

male

Encounters:

63

Conditions:

8

Observations:

93

Medication Requests:

2

Immunizations:

8

Care Plans:

4

Encounters

Encounter ID	Start Date	End Date	Status	Actions
0161cc43-d8eb-4bbc-a5b6-5c1b6d6ec90e	2014-07-02T16:04:49-04:00	2014-07-02T16:19:49-04:00	finished	<div>View Details</div>
01d2f020-2568-4cfb-b755-641e97d65e68	2009-06-23T16:04:49-04:00	2009-06-23T16:19:49-04:00	finished	<div>View Details</div>
07b35c75-7519-4edc-b074-6f42b11ae858	2009-09-21T16:04:49-04:00	2009-09-21T16:19:49-04:00	finished	<div>View Details</div>
0a90c3c1-afaa-4a3a-a0d5-fb41ec93163d	2017-08-08T16:04:49-04:00	2017-08-08T16:19:49-04:00	finished	<div>View Details</div>

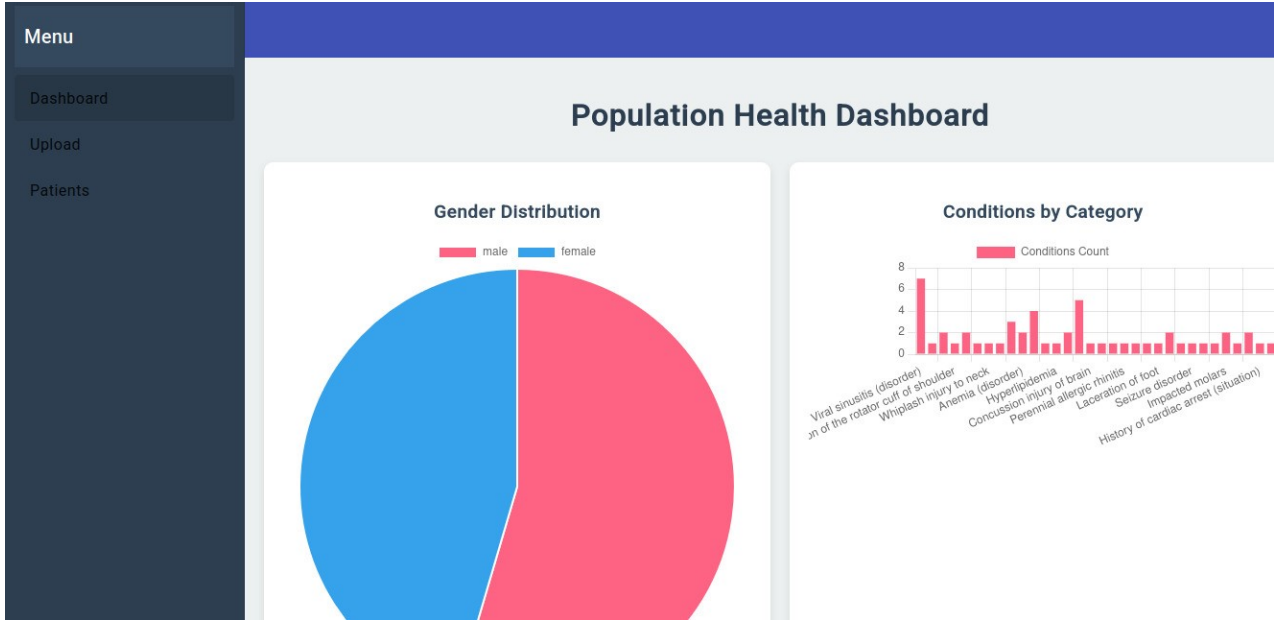
Immunizations

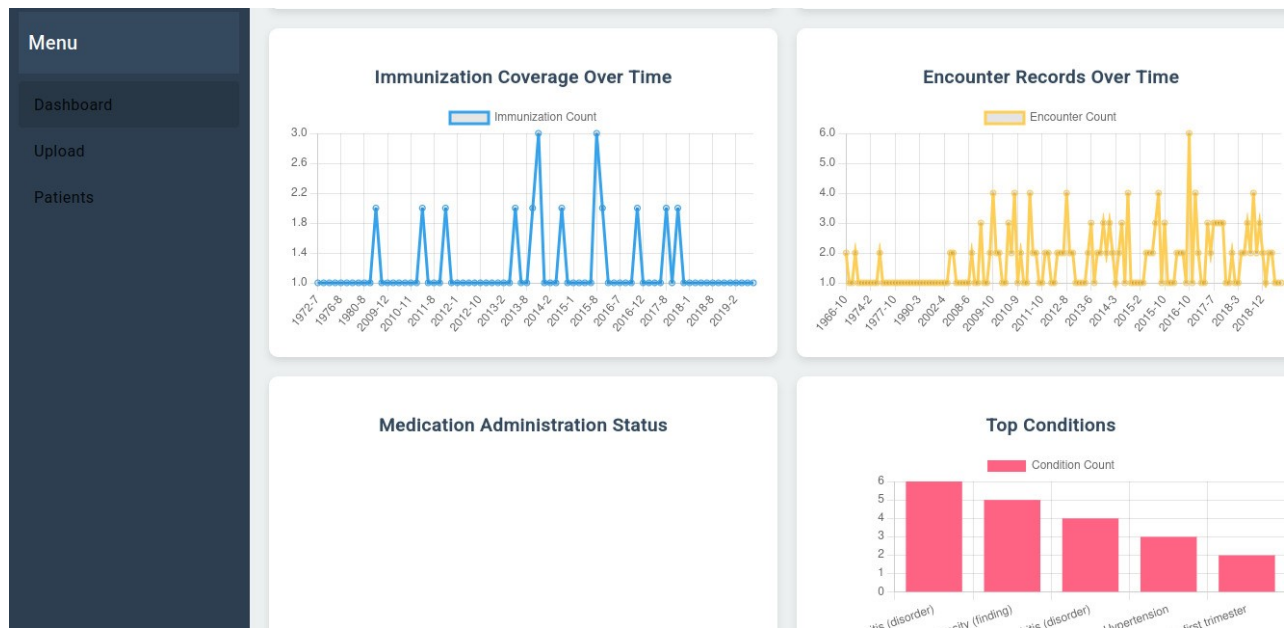
Vaccine	Date
140	2015-08-04T16:04:49-04:00
140	2013-07-30T16:04:49-04:00
52	2011-07-26T16:04:49-04:00
113	2013-07-30T16:04:49-04:00
140	2011-12-27T15:04:49-05:00

Conditions		
Condition	Onset Date	
Anemia (disorder)	2005-10-04T16:04:49-04:00	
Drug overdose	2013-12-12T15:04:49-05:00	
Laceration of foot	2009-10-11T16:04:49-04:00	
Chronic pain	1990-06-13T16:04:49-04:00	
Impacted molars	1990-10-02T16:04:49-04:00	
Acute bronchitis (disorder)	2011-12-18T15:04:49-05:00	
Chronic intractable migraine without aura	1990-03-03T15:04:49-05:00	
Prediabetes	1999-09-28T16:04:49-04:00	
Care Plans		

3. Tableaux de Bord Dynamiques

- Affichage des statistiques sur les patients (par genre, âge, etc.).
- Suivi des immunisations et des observations médicales.
- Analyse des tendances temporelles des données collectées.





4. Gestion des Erreurs et Validation

- Retour d'informations en cas d'erreurs dans les fichiers de données.
- Suivi des erreurs et suggestions d'amélioration via des logs détaillés.

Technologies Utilisées

L'application repose sur un ensemble de technologies modernes qui assurent sa performance et sa facilité d'utilisation :

- **Backend :** Flask, Python, MySQL, PyMySQL
- **Frontend :** Angular, TypeScript, HTML/CSS, Bootstrap
- **Outils :** Postman pour le test des API, Git pour la gestion du code source

7. Les défis rencontrés :

Tout au long de la réalisation de ce projet, nous avons été confrontés à plusieurs défis qui ont nécessité des efforts considérables en termes d'analyse, de recherche et d'adaptation. Ces défis, inhérents à la complexité du domaine de la santé et à l'intégration de nouvelles technologies, ont été relevés grâce à une approche méthodique et rigoureuse. Les principaux obstacles auxquels nous avons dû faire face sont les suivants :

1. Comprendre ce qui est demandé

L'un des premiers défis rencontrés a été de bien cerner les exigences du projet et de définir précisément les objectifs à atteindre. La gestion des données de santé est un domaine complexe, impliquant des réglementations strictes, des formats de données variés et des attentes élevées en matière de qualité et d'interopérabilité. Il était donc essentiel de clarifier les besoins des parties prenantes et de traduire ces exigences en spécifications techniques réalisables. Cela a

nécessité une phase d'analyse approfondie pour identifier les processus métiers clés, les types de données à traiter, ainsi que les résultats attendus en termes de visualisation et d'analyse.

2. Trouver les données nécessaires

Un autre défi majeur a été la recherche et la collecte des données nécessaires pour alimenter notre solution. Dans le cadre du projet, nous avons travaillé avec des fichiers JSON conformes au standard FHIR, mais il a fallu identifier les bonnes sources de données, s'assurer de leur fiabilité et comprendre leur structure. Cette étape a exigé une exploration approfondie des jeux de données disponibles et une vérification minutieuse de leur qualité, leur complétude et leur conformité aux standards du domaine médical.

3. Assimiler de nouvelles notions techniques et médicales

Le projet nous a amenés à explorer de nombreuses notions nouvelles, tant sur le plan technique que médical. D'un point de vue technologique, il a fallu se familiariser avec le standard **FHIR**, comprendre sa structure modulaire basée sur les ressources, ainsi que maîtriser les outils et technologies permettant son exploitation, tels que le traitement des fichiers JSON, la gestion de bases de données relationnelles et la mise en œuvre de pipelines ETL. Sur le plan médical, il était crucial de comprendre les différentes entités impliquées dans les données de santé, telles que les observations cliniques, les procédures médicales et les plans de soins, afin d'assurer une modélisation correcte des informations.

4. Gérer la complexité du pipeline ETL

La mise en œuvre du pipeline ETL s'est révélée être un défi en raison de la complexité et de la diversité des données à traiter. Chaque fichier JSON contenait des ressources avec des relations imbriquées, ce qui nécessitait une approche rigoureuse pour extraire les informations pertinentes, les transformer en un format relationnel cohérent et garantir leur intégration correcte dans la base de données. La phase de transformation a demandé des efforts considérables pour nettoyer les données, normaliser les formats et gérer les éventuelles valeurs manquantes. Il a également été essentiel d'optimiser les performances du pipeline pour traiter efficacement un volume important de données dans un délai raisonnable.

5. Assurer la qualité et la validation des données

Un autre défi crucial a été de garantir la qualité et l'exactitude des données tout au long du processus. La validation des données chargées dans la base relationnelle devait être rigoureuse afin de s'assurer qu'aucune information critique n'était perdue ou altérée. Nous avons mis en place plusieurs mécanismes de contrôle de qualité, tels que la vérification des relations entre les entités, l'identification des incohérences, et l'évaluation de la complétude des données. Ces efforts ont permis de détecter et de corriger des erreurs potentielles avant l'exploitation des données.

6. Développer une application web intuitive

Le développement d'une application web permettant aux utilisateurs d'interagir avec les données de santé a représenté un autre défi. Il fallait concevoir une interface utilisateur intuitive et ergonomique, tout en intégrant des fonctionnalités avancées de filtrage, de visualisation et de recherche des données. Le choix des technologies adaptées, l'optimisation des performances côté client et serveur, ainsi que la prise en compte des besoins des utilisateurs finaux ont nécessité une réflexion approfondie et des tests itératifs.

7. Gérer le temps et les ressources

La gestion du temps et des ressources a été un aspect essentiel du projet. Il a fallu prioriser les tâches, organiser les différentes étapes du développement et coordonner les efforts de l'équipe pour respecter les délais impartis. La planification rigoureuse et l'utilisation d'outils de gestion de projet nous ont aidés à maintenir le cap et à livrer les différentes phases du projet dans les temps.

8. Conclusion :

Ce projet a permis de démontrer l'importance de l'adoption du standard FHIR pour garantir une gestion optimisée des données de santé et leur interopérabilité au sein d'un écosystème numérique de plus en plus complexe. Grâce à la mise en œuvre du pipeline ETL, nous avons pu extraire, transformer et charger efficacement les données médicales, tout en assurant un niveau élevé de complétude et de qualité. Les résultats obtenus ont montré une intégration réussie des principales ressources de santé, telles que les patients, les observations médicales et les procédures, dans une base de données relationnelle performante.

L'application web développée dans le cadre de ce projet a permis de fournir un outil interactif et intuitif pour l'exploration et l'analyse des données, facilitant ainsi la prise de décisions éclairées par les professionnels de santé.

À l'avenir, des améliorations peuvent être envisagées, telles que l'intégration de modèles d'intelligence artificielle pour l'analyse prédictive des données de santé, l'optimisation des performances du pipeline ETL et l'extension de l'application web pour inclure des fonctionnalités plus avancées de reporting et de visualisation.

En conclusion, ce projet constitue une base solide pour une gestion efficace des données de santé et ouvre la voie à des perspectives prometteuses en matière d'exploitation des données médicales dans des environnements interconnectés et intelligents.

9. Références :

[1] <https://www.kaggle.com/datasets/drscarlat/fhir-1ksample>

-
- [2] <https://www.kaggle.com/code/drscarlat/fhir-starter-parse-healthcare-bundles-into-tables/notebook>
 - [3] <https://hl7.org/fhir/>
 - [4] <https://fhirblog.com/2014/03/28/pictorial-representation-of-fhir-resouces/>
 - [5] <https://academic.oup.com/jamia/article/25/3/230/4098271>
 - [6] <https://innovaccer.com/resources/blogs/the-abc-of-fhir-reinventing-interoperability>
 - [7] https://en.wikipedia.org/wiki/Fast_Healthcare_Interoperability_Resources
 - [8] https://upload.wikimedia.org/wikipedia/commons/4/48/FHIR_resource_graph.png
 - [9] <https://synthea.mitre.org/downloads>