ENSEMBLE DATA FOUNDATION(EDF)

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING

BY

SHRUTI PATEL
EN18CS302048

Under the Guidance of
Prof. Varsha Sharda

Department of Computer Science & Engineering
Faculty of Engineering
MEDI-CAPS UNIVERSITY, INDORE- 453331

MAY-2022

ENSEMBLE DATA FOUNDATION(EDF)

A Project-II Report

Submitted in partial fulfillment of requirement of the

Degree of

BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE & ENGINEERING

BY

SHRUTI PATEL
EN18CS302048

Under the Guidance of
Prof. Varsha Sharda

Department of Computer Science & Engineering
Faculty of Engineering
MEDI-CAPS UNIVERSITY, INDORE- 453331

MAY 2022

Report Approval

The project work **“Ensemble Data Foundation”** is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approved any statement made, opinion expressed, or conclusion drawn there in; but approve the “Project Report” only for the purpose for which it has been submitted.

Internal Examiner
Name: Ms. Varsha Sharda
Designation: Medi-Caps Faculty
Affiliation

External Examiner
Name: Mr. Ranjit Singh
Designation: Test Architect
Affiliation: Employee at Fifthnote an Ensemble Health Partner’s Company

Declaration

I/We hereby declare that the project entitled **“Ensemble Data Foundation”** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in ‘**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**’ completed under the supervision of **Ms. Varsha Sharda, Medi-Caps Faculty and Department of Computer Science & Engineering,** Faculty of Engineering, Medi-Caps University Indore is an authentic work.

Further, I/we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

Shruti Patel

Date : 30/04/2022

Certificate

I/We, **Ms. VARSHA SHARDA** certify that the project entitled **“Ensemble Data Foundation”** submitted in partial fulfillment for the award of the degree of Bachelor of Technology by **Shruti Patel** is the record carried out by him/them under my/our guidance and that the work has not formed the basis of award of any other degree elsewhere.

___________________________	___________________________
Ms. VARSHA SHARDA	Mr. RANJIT SINGH
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING	Office Mentor
Medi-Caps University, Indore	Fifth Note

____

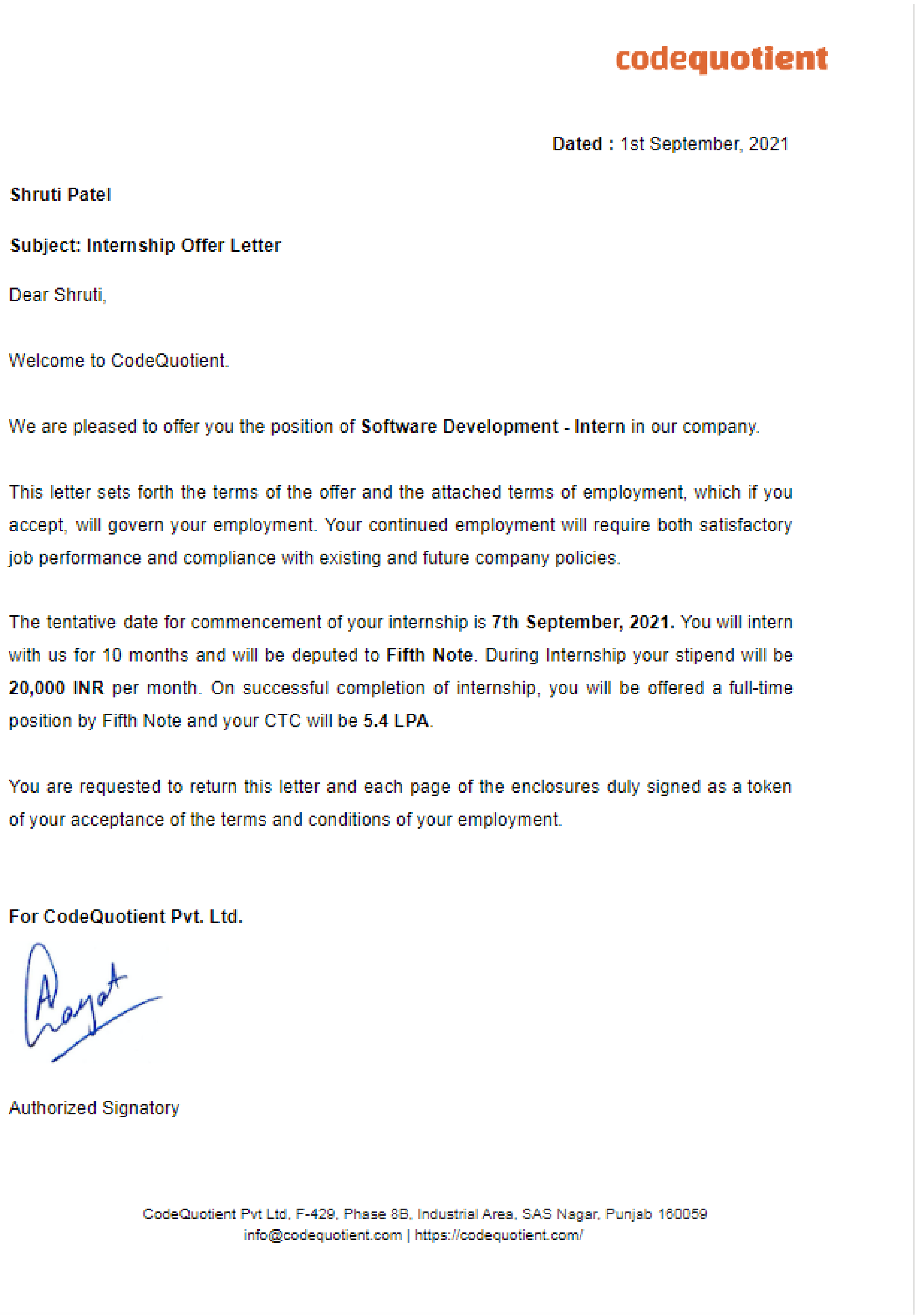
Dr. Pramod S. Nair

Head of the Department

Computer Science & Engineering

Medi-Caps University, Indore

Offer Letter of the Project work-II/Internship



codequotient

Dated : 1st September, 2021

Shruti Patel

Subject: Internship Offer Letter

Dear Shruti,

Welcome to CodeQuotient.

We are pleased to offer you the position of **Software Development - Intern** in our company.

This letter sets forth the terms of the offer and the attached terms of employment, which if you accept, will govern your employment. Your continued employment will require both satisfactory job performance and compliance with existing and future company policies.

The tentative date for commencement of your internship is **7th September, 2021.** You will intern with us for 10 months and will be deputed to **Fifth Note**. During Internship your stipend will be **20,000 INR** per month. On successful completion of internship, you will be offered a full-time position by Fifth Note and your CTC will be **5.4 LPA**.

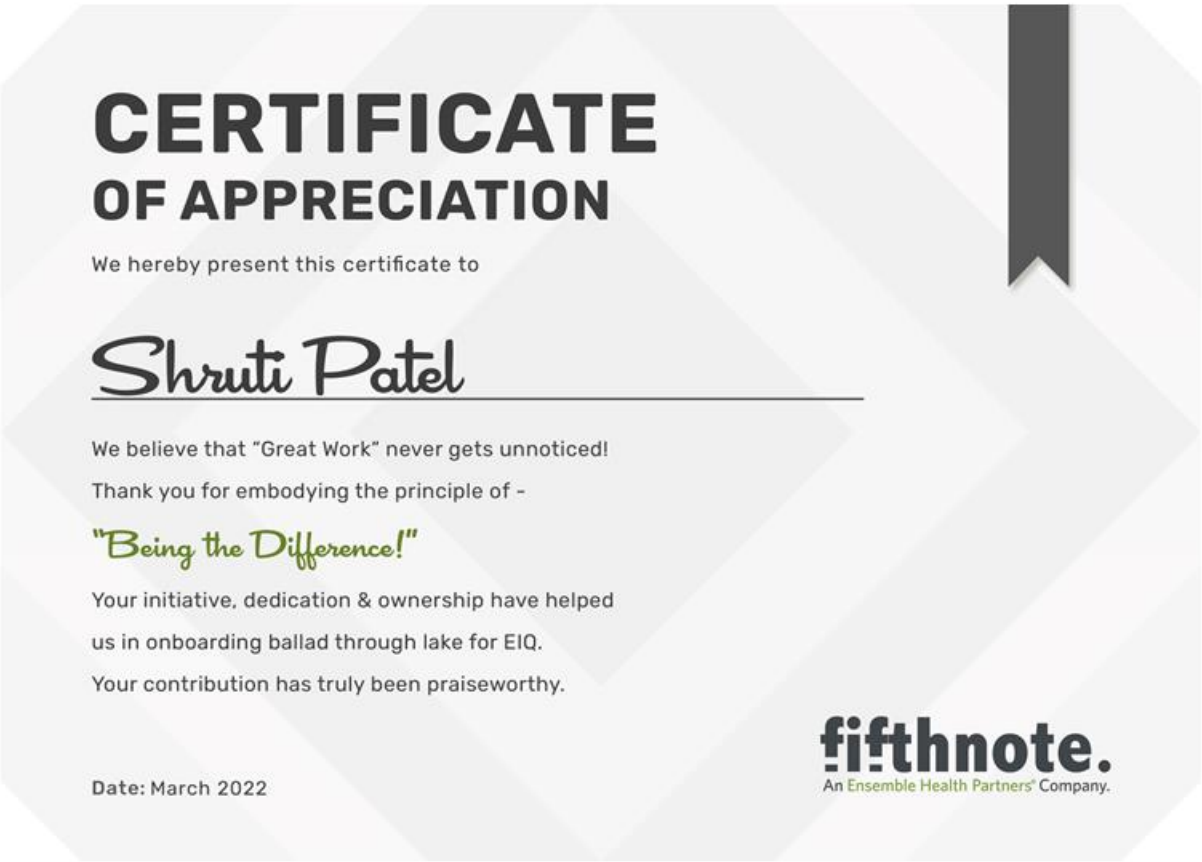
You are requested to return this letter and each page of the enclosures duly signed as a token of your acceptance of the terms and conditions of your employment.

For CodeQuotient Pvt. Ltd.

Authorized Signatory

CodeQuotient Pvt Ltd, F-429, Phase 8B, Industrial Area, SAS Nagar, Punjab 160059
info@codequotient.com | https://codequotient.com/

Appreciation certificate/Letter



CERTIFICATE

OF APPRECIATION

We hereby present this certificate to

Shruti Patel

We believe that "Great Work" never gets unnoticed!

Thank you for embodying the principle of -

"Being the Difference!"

Your initiative, dedication & ownership have helped us in onboarding ballad through lake for EIQ. Your contribution has truly been praiseworthy.

Date: March 2022

fifthnote.

An Ensemble Health Partners® Company.

Acknowledgements

I would like to express my deepest gratitude to Honorable Chancellor, **Shri R C Mittal,** who has provided me with every facility to successfully carry out this project, and my profound indebtedness to **Prof. (Dr.) Dileep K Patnayak,** Vice Chancellor, Medi-Caps University, whose unfailing support and enthusiasm has always boosted up my morale. I also thank **Prof. (Dr.) D K Panda,** Pro Vice Chancellor, **Dr. Suresh Jain,** Dean Faculty of Engineering, Medi-Caps University, for giving me a chance to work on this project. I would also like to thank my Head of the Department **Dr. Pramod S. Nair** for his continuous encouragement for betterment of the project.

I express my heartfelt gratitude to my **External Guide, Mr. Ranjit Singh, Fifthnote an Ensemble Health Partners Company** via **Code Quotient** as well as to my Internal Guide, **Ms. Varsha Sharda,** Medi-Caps faculty, Department of Computer Science & Engineering, without whose continuous help and support, this project would ever have reached to the completion.

I would also like to thank to my team at Fifthnote and my mentor Mr. Ranjit Singh who extended their kind support and help towards the completion of this project. I Appreciate the cooperation that everyone displayed under such strenuous conditions, making the workflow simple and easy. I am glad that I have a team like you.

It is their help and support, due to which we became able to complete the design and technical report. Without their support this report would not have been possible.

Shruti Patel
B.Tech. IV Year
Department of Computer Science & Engineering
Faculty of Engineering
Medi-Caps University, Indore

Abstract

Fifthnote is a leading technology acceleration company which identifies business process optimization by implementing system integration and automation. Founded in 2012 and acquired by Ensemble Health Partners in 2020.

It supports EnsembleIQ (EIQ®), a cloud-based analytics-driven revenue cycle operating platform that enables a highly efficient and intelligent workflow automation. By using extensive domain expertise, automation and advanced acritical intelligence, we are driving efficiency and yield.

An innovative, technology-enabled approach to revenue cycle, paired with fifthnote and Ensemble's people, drives efficiency, productivity and results for healthcare providers, resulting in additional resources for patient care.

Ensemble Data Foundation

Healthcare revenue cycle management is the financial process facilities use to manage the administrative and clinical functions associated with claims processing, payment, and revenue generation. The process consists of identifying, managing, and collecting patient service revenue.

The EDF project basically deals with performing ETL on data from clients for Analytics and BI. The client sends their raw files from different sources in different-different formats to the landing zone which is nothing but a container in azure, from where we focus on structuring and harmonizing data using ADF and FLUX according to the business requirements and standard specified documents. Once the data is harmonized, then it will be consumed by other processes like BI publishing service and EIQ publishing service which is used to extract harmonized data and transform based on requirements and then push that transformed data into the main publishing service database.

List of Figures

List Of Tables

Abbreviations

EDF	Ensemble Data Foundation
EIQ	Ensemble Intelligence Quotient
PS	Publishing Service
ADF	Azure Data Factory
DB	Database
ETL	Extract transform load

Notations & Symbols

Table of Contents

Chapter-1

1.1 INTRODUCTION

Organization Description

FifthNote is a leading technology acceleration company which identifies business process optimization by implementing system integration and automation. We were founded in 2012 and acquired by Ensemble Health Partners in 2020.

We support EnsembleIQ (EIQ®), a cloud-based analytics-driven revenue cycle operating platform that enables a highly efficient and intelligent workflow automation. By using extensive domain expertise, automation and advanced acritical intelligence, we are driving efficiency and yield.

An innovative, technology-enabled approach to revenue cycle, paired with fifthnote and Ensemble's people, drives efficiency, productivity and results for healthcare providers, resulting in additional resources for patient care.

Our goal is to relentlessly seek breakthrough innovation in technology that will shape the healthcare of tomorrow. For, we believe that we are making an impact where it matters the most – we are not your typical software development company, we just don't code, we **#codeforhealth**

REVENUE CYCLE MANAGEMENT

Revenue Cycle is the financial process that healthcare facilities use to manage the administrative and clinical Functions associated with claims processing, payment and revenue generation. The process encompasses the identification, management and the collection of patient service revenue. Now, more than ever, the financial strength of hospitals and health system is imperative. A string revenue cycle enables healthcare providers to deliver exceptional care to their patients and communities, and invest in new equipment, new facilities and expanded community health programs.

Revenue Cycle Components

- Front – Patient Access Services
- Middle – Health Information Management
- Back End – Patient Financial Services
- Support Services

Fig. 1.1 Company's logo

FOLLOWING THE TECH MATURITY CURVE

- RPA (Robotic Process Automation)
  Automating labor-intensive, repetitive activities across multiple systems and interfaces by training and/or programming third-party software to replicate a user's workflow.

- BPA (Business Process Automation)
  Reengineering existing business processes by using software, integrating systems, and restructuring labor to optimize workflows and minimize costs

- IPA (Intelligent Process Automation)
  Combining RPA with artificial intelligence, BPA and statistical analytics technologies to identify patterns, learn over time, and optimize workflows

- Autonomous automation
  Autonomous automation, being developed for the future**,** creates and deploys machines that act on their own

Seamlessly integrate powerful RCM technology into your existing systems.

Improve revenue capture and yield, increase staff productivity and gain real-time visibility and enterprise-wide insights.

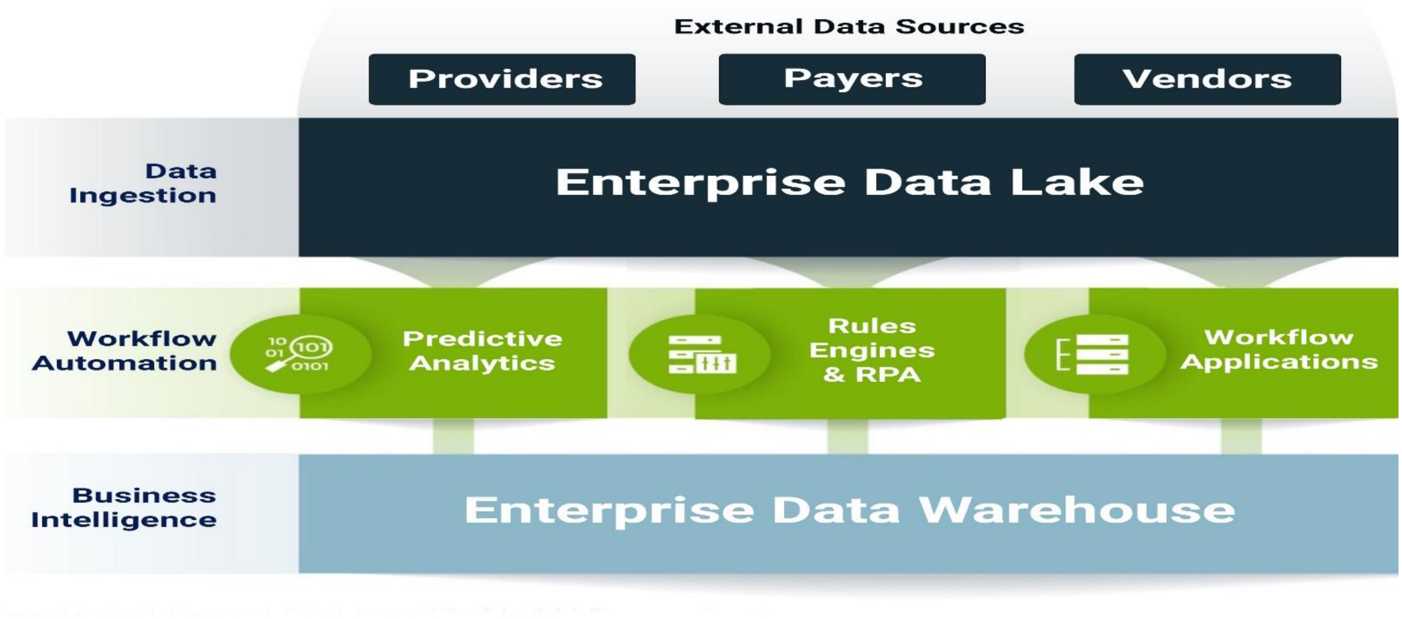


Fig. 1.2 high level Architecture

1.2 Literature Review

The themes explored by academic and healthcare industry journals surround discussions of technology and applications, the benefits delivered through analytical oriented approaches to revenue cycle management, and the barriers to these same innovations. A final set of discussions entailed assessments of likely risk variables and viable risk management approaches to address these challenges. Analyses that explored background themes related to the dissertation's topic focused on three areas of discussion: the concept of artificial intelligence (AI) and machine learning (ML), process definitions of revenue cycle management, and the broader assessment of ML's potential for managing these same processes. A large group of research in the areas of AI and ML that is specific to healthcare finance revolves around the processing of claim requests and payments from third-party payers. The research has indicated that a significant amount of money is lost due to the complexity of claims and inaccurately completed claims. When a claim is inaccurately completed, it must be returned to the institution filing the claim, and this must be rectified. This creates additional time in reworking the claim and extends the time between claim submission and payment, which negatively reflects on the organization's 17 financial health. Numerous studies have used novel approaches combining AI and ML to automatically detect such errors and annotate them with reasons why they are being flagged. Some of these systems boast a 25% improvement over any current claim analysis software or methods. This literature review identified several specific aspects of machine learning and artificial intelligence related to the healthcare revenue cycle. Of importance, the revenue cycle and the processes associated with it often have very repetitive tasks performed by humans. However, many of these tasks would benefit from using machine learning or artificial intelligence to automate them. In implementing these strategies, healthcare organizations could likely reduce costs and improve accuracies related to payments and other similar factors, thus increasing revenue from existing claims by reducing denials.

Despite the continued interest of practitioners (Danielson and Fuller 2007; D’Cruz and Welter 2008; May 2004), hospital revenue cycle management has not received much attention in health care finance research. Instead, the overwhelming majority of publications on revenue cycle management are based on the insights and experiences of practitioners working in the field. These publications develop ad hoc performance measures and standards, discuss anecdotal evidence of potentially important factors of effective revenue cycle management, and offer suggestions for improvements at other hospitals based on the experiences gained at the authors' institutions. The few existing empirical studies of hospital revenue cycle management are mainly exploratory studies that fall in two categories. The first group consists of studies that analyze factors associated with the financial benefits of hospital revenue cycle management performance, such Prince and Ramanan’s (1992) paper on hospitals’ collection performance. The second group represents studies of hospitals’ profitability that include single measures of revenue cycle management performance as explanatory variables, such as average collection periods and mark-up ratios (Cleverley 1990; Cody, Friss, and Hawkinson 1995; Gapenski, Vogel, and Langland-Orban 1993). None of these studies, however, takes into account that effective revenue cycle management may result in not just one but multiple financial benefits. In addition, the empirical methods employed in the above mentioned studies are largely limited to correlation and simple regression analyses and may be improved upon using more advanced econometric techniques.

1.3 Objective

Focus on patients, not payments. It optimize revenue cycle operations for health systems — from patient engagement and intake through revenue collection. Deliver exceptional care, improve patient outcomes and support community wellness. We'll do the rest, so you can focus on what you do best.

From patient intake through revenue collection, revenue cycle management (RCM) spans the entire Lifecycle of a medical claim. It includes the mission-critical processes healthcare providers use to identify, manage and collect revenue from patients, insurance companies and other payors.

1.4 Significance

Sustainable improvements in financial performance driven by increased net patient revenue and operating margins and accelerated cash flow.

Higher patient satisfaction thanks to a streamlined registration and billing experience

Increased physician + staff satisfaction as a result of relief from administrative burdens

Chapter-2

2.1 Experimental Setup

SOFTWARE REQUIREMENTS :

Operating System	Windows 10
IDE/Workbench	SSMS, VS Code, Spyder
Programming Language	Python
Framework	Apache Spark, flux
Azure Subscriptions	ADF, Databricks, Ado, Microsoft 365 E3
Browser	Google Chrome, Edge
Database	MySQL

Table 2.1 Software Requirements

HARDWARE REQUIREMENTS :

Processor	Intel CORE i5
Hard Disk	50GB or more
RAM	64GB or more
System Type	64-bit OS

Table 2.2 Hardware Requirements

2.2 Procedure Adopted

2.2.1 Methodology Used

A software is developed with several different techniques and methodologies. It requires tools, models and other external elements to achieve successful completion. Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like −

- Planning
- Requirements Analysis
- Design
- Coding
- Unit Testing and
- Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

2.2.2 Agile

Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release.

Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

The most popular Agile methods include Rational Unified Process (1994), Scrum (1995), Crystal Clear, Extreme Programming (1996), Adaptive Software Development, Feature Driven Development, and Dynamic Systems Development Method (DSDM) (1995). These are now collectively referred to as **Agile Methodologies**, after the Agile Manifesto was published in 2001.

Following are the Agile Manifesto principles −

- **Individuals and interactions** − In Agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.
- **Working software** − Demo working software is considered the best means of communication with the customers to understand their requirements, instead of just depending on documentation.

- **Customer collaboration** − As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customer interaction is very important to get proper product requirements.
- **Responding to change** − Agile Development is focused on quick responses to change and continuous development.

Here is a graphical illustration of the Agile Model –

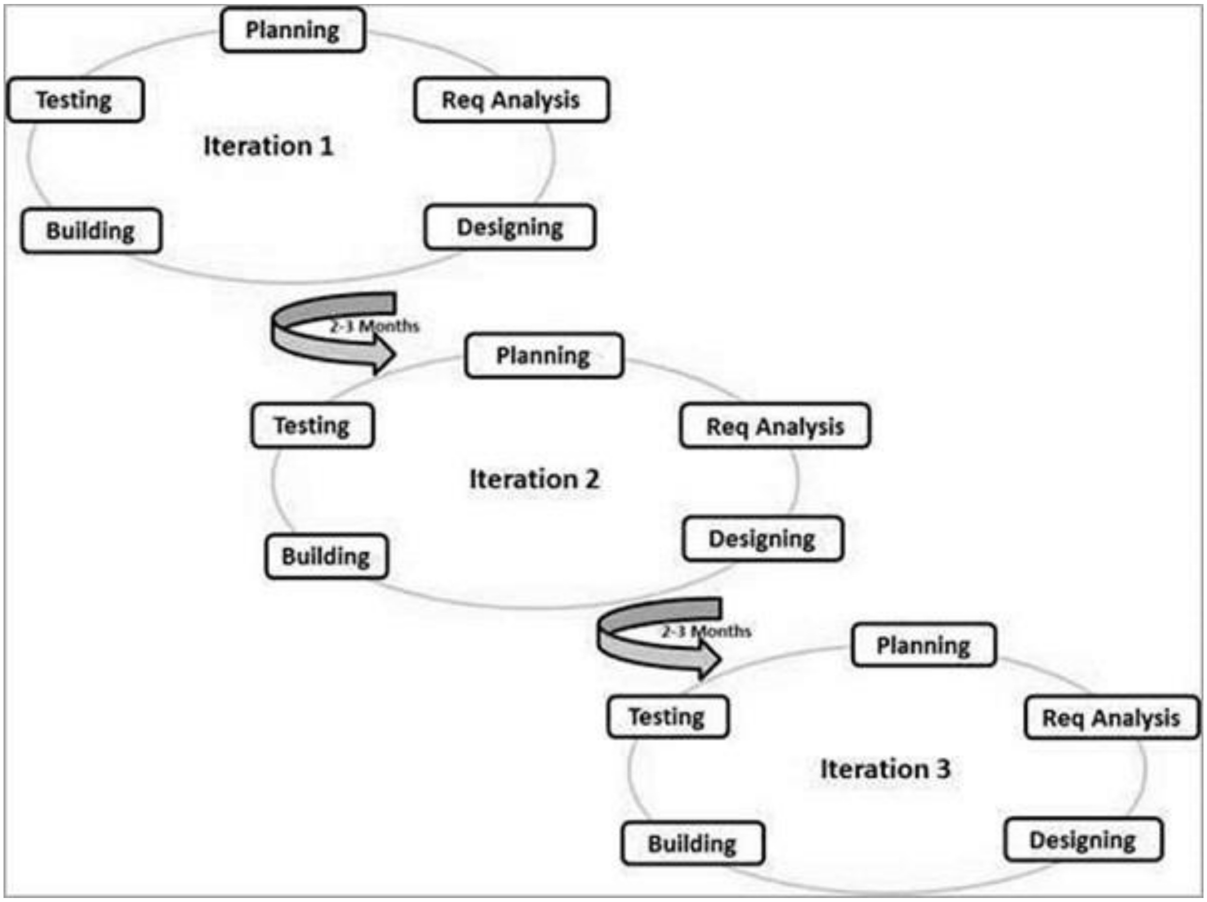


Fig. 2.1 Agile Model

2.2.3 Scrum

It is the most popular agile framework, which concentrates particularly on how to manage tasks within a team-based development environment. Scrum uses iterative and incremental development model, with shorter duration of iterations. Scrum is relatively simple to implement and focuses on quick and frequent deliveries.
The Scrum framework consists of Scrum Teams and their associated roles, events, artifacts, and rules. Each component within the framework serves a specific purpose and is essential to Scrum's success and usage.

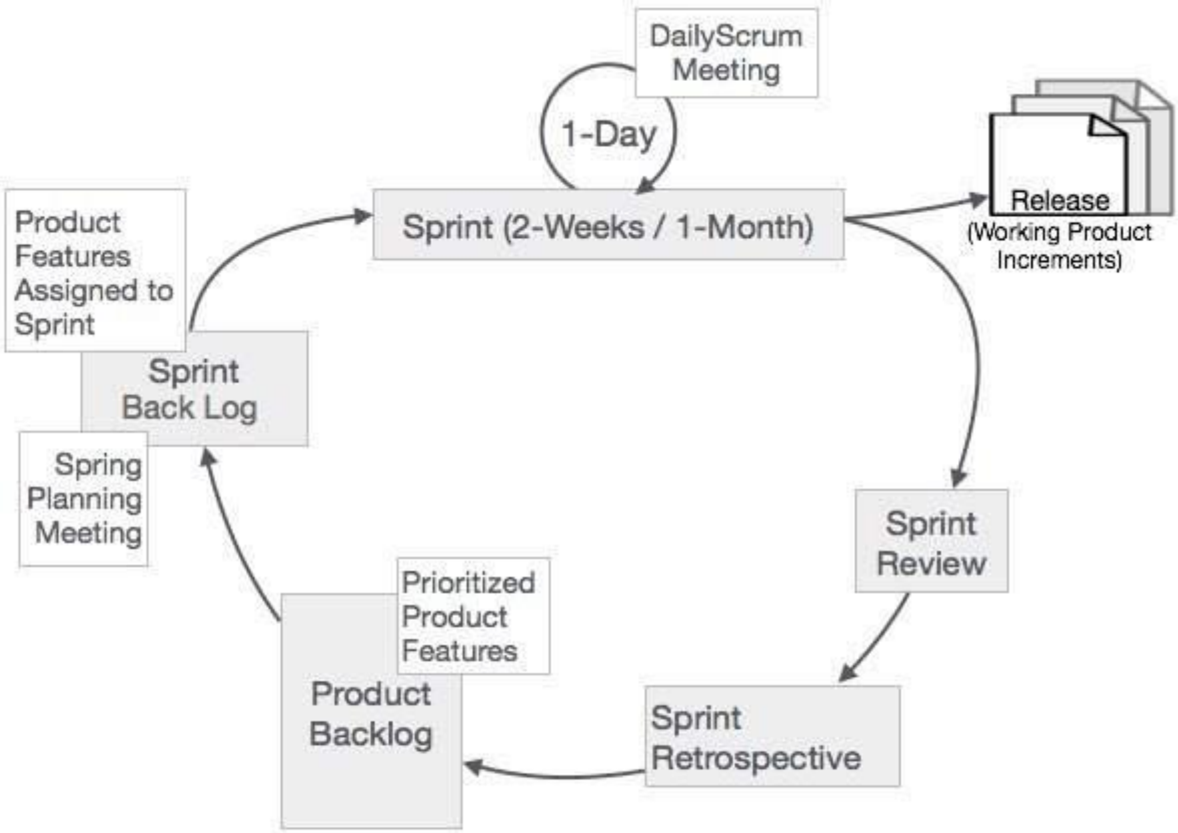

DailyScrum Meeting
1-Day
Sprint (2-Weeks / 1-Month)
Release
(Working Product Increments)
Product Features Assigned to Sprint
Sprint Back Log
Spring Planning Meeting
Sprint Review
Prioritized Product Features
Product Backlog
Sprint Retrospective

Fig. 2.2 Scrum Process Framework

Sprint

The heart of Scrum is a Sprint, a time-box of two weeks or one month during which a potentially releasable product increment is created. A new Sprint starts immediately after the conclusion of the previous Sprint. Sprints consist of the Sprint planning, daily scrums, the development work, the Sprint review, and the Sprint retrospective.

Chapter-3

3.1 Introduction

The EDF project basically deals with performing ETL on data from clients for Analytics and BI. client sends their raw files from different sources in different-different formats to the landing zone which is nothing but a container in azure, from where we focus on structuring and harmonizing data using ADF and FLUX according to the business requirements and standard specified documents. Once the data is harmonized, then it will be consumed by other processes like BI publishing service and EIQ publishing service which is used to extract harmonized data and transform based on requirements and then push that transformed data into the main publishing service database.

The project consists of data flowing through various steps of EDF in order to gain appropriate data EIQ and BI purposes which can be efficiently processed and manipulated.

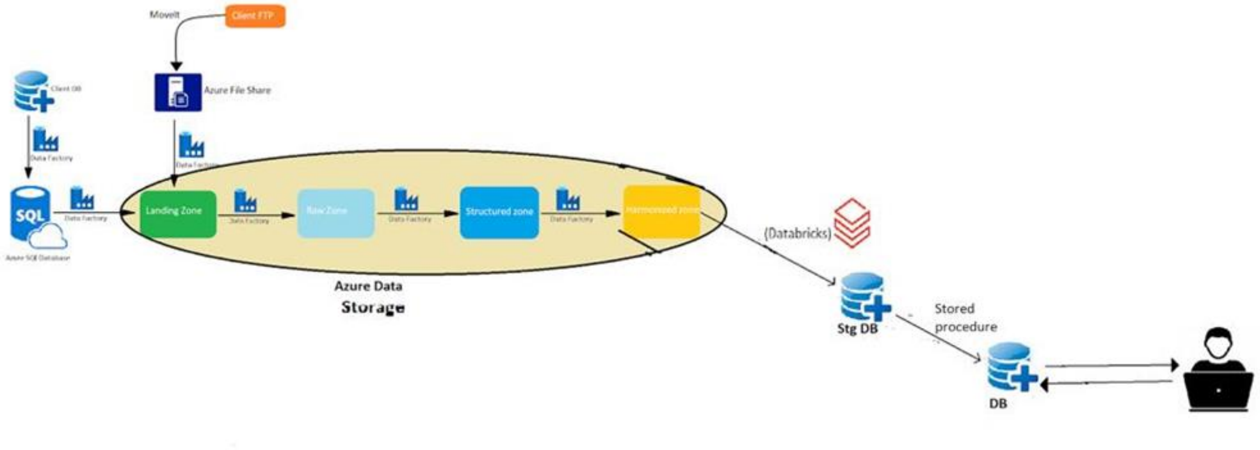


Fig. 3.1 EDF Architecture

EDF is Ensemble project that manages the ETL Task required in the data flow.

It uses Azure to manage Storage and Transformation of data in the organization for EIQ.

EIQ combines – Artificial Intelligence, Robotic Process Automation and Machine Learning to augment and accelerate the best Revenue Cycle Processes managed by Ensemble Health Partners.

3.2 EDF Zone Details

Zones in EDF are simply the names given to the containers in Azure storage to segregate the data Based on the transformations which is done step by step.

3.2.1 Landing Data

The data is received from the client in the following procedure:

- Client does the initial processing of their data.
- Data is partially structured (parquet, text, delimited, etc.)
- Data is Transferred to a collective landing zone
- ADF and SSIS is used to simply move this data
- The landing zone data is segregated for each client.
- Data is received in a predefined format and criteria

3.2.2 Raw Data

The data is received from the client in the following procedure:

- The Source for this is the Client landing data.
- Data is conformed according to specifications.
- Data is initially converted for integrity of further ETL.
- The Zone exists on a data lake managed by azure.
- Data exists in for of semi-structured objects.
- Data is still in raw format.

3.2.3 Structured Data

The data is received from the client in the following procedure:

- The Source for this is the Raw data.
- Data is processes and structured.
- This is first major ETL step.
- In this, data is formed and validated initially.Data is then readied for usability.
- Although data is structured, it still cannot be used in a database.

3.2.4 Harmonized Data

The data is received from the client in the following procedure:

- The Source for this is the Structured data.
- The structured data is extracted.
- The data is manipulated to be harmonized all across.
- This step is important for it to be used in databases.
- Since MSSQL Server is relational database, data must be harmonized according to it.

- Harmonized data is then loaded onto SQL server.

3.3 ETL

ETL stands for Extract Transform and Load. ETL combines all the three database function into one tool to fetch data from one database and place it into another database.

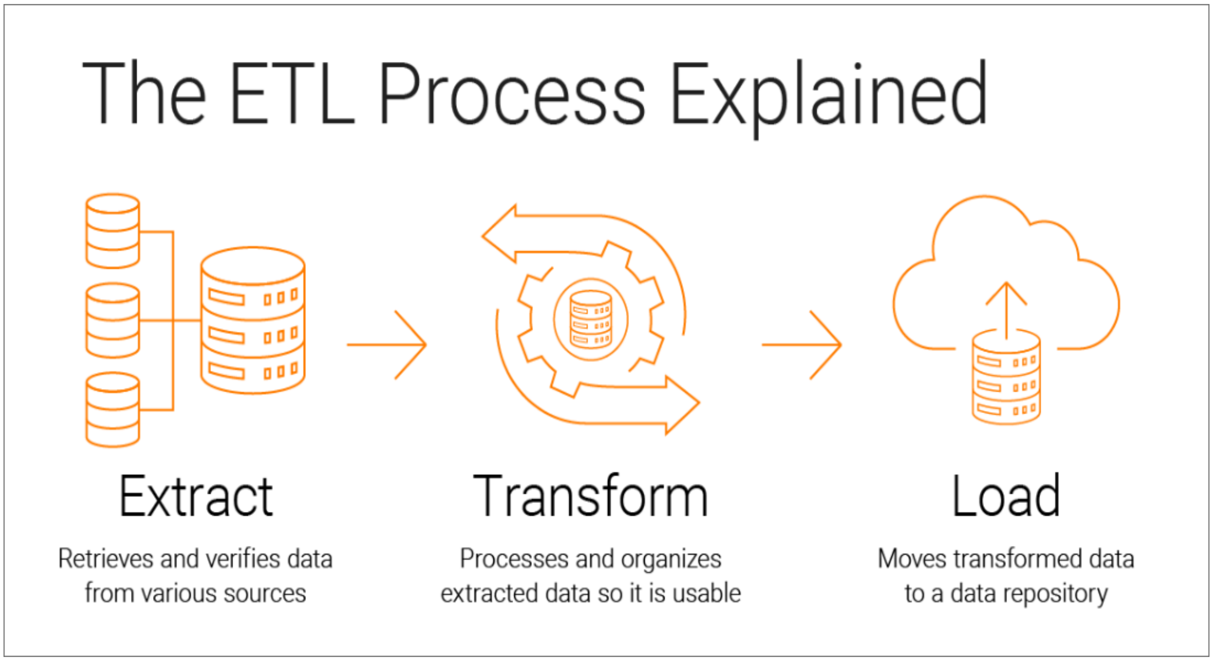

The ETL Process Explained
Extract
Retrieves and verifies data from various sources
Transform
Processes and organizes extracted data so it is usable
Load
Moves transformed data to a data repository

Fig. 3.3.1 ETL process

ETL consists of three parts:

1.Extract : Extract is the process of fetching (reading) the information from the database. At this stage, data is collected from multiple or different types of sources

2.Transform : Transform is the process of converting the extracted data from its previous form into the required form. Data can be placed into another database. Transformation can occur by using rules or lookup tables or by combining the data with other data.

3.Load : Load is the process of writing the data into the target database

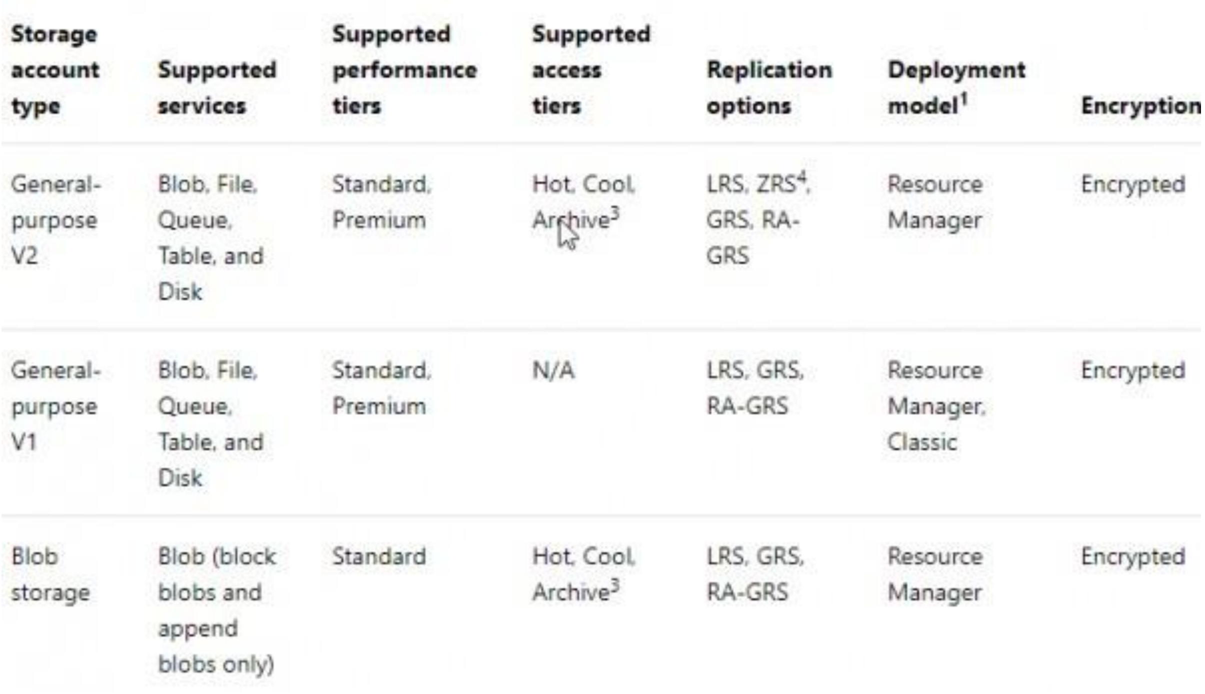
Azure Data Factory and SSIS provide the tools necessary for Extraction, Transaformation and Loading data onto the SQL servers through various file formats which finally deliver the data to the SQL Server.

3.4 Platforms Tools and Languages Used

3.4.1 Azure Storage Account

An Azure Storage Account is a secure account, which provides you access to services in Azure Storage. The storage account is like an administrative container, and within that, we can have several services like blobs, files, queues, tables, disks, etc. And when we create a storage account in Azure, we will get the unique namespace for our storage resources. That unique namespace forms the part of the URL. The storage account name should be unique across all existing storage account name in Azure.

Types of Storage Accounts:



Storage account type	Supported services	Supported performance tiers	Supported access tiers	Replication options	Deployment model[1]	Encryption
General-purpose V2	Blob, File, Queue, Table, and Disk	Standard, Premium	Hot, Cool, Archive[3]	LRS, ZRS[4], GRS, RA-GRS	Resource Manager	Encrypted
General-purpose V1	Blob, File, Queue, Table, and Disk	Standard, Premium	N/A	LRS, GRS, RA-GRS	Resource Manager, Classic	Encrypted
Blob storage	Blob (block blobs and append blobs only)	Standard	Hot, Cool, Archive[3]	LRS, GRS, RA-GRS	Resource Manager	Encrypted

Table 3.4.1 Types of storage account

Storage account endpoints: Whenever we create a storage account, we will get an endpoint to access the data within the storage account. So each object that we stored in Azure storage has an address, which includes y our unique account name and the combination of an account name, and service endpoint, which forms the endpoint for your storage account.

For example:

- Azure Blob storage: http://mystorageaccount.blob.core.windows.net.
- Azure Table storage: http://mystorageaccount.table.core.windows.net
- Azure Queues storage: http://mystorageaccount.queue.core.windows.net

Benefits of using Azure Storage:

- Scalable and Durable to provide space as per your need.
- Automated Backup and Recovery so that you don’t loose your data.
- Data Encryption to make your data secure.
- Support for multiple data types

3.4.2 Azure Data Factory

Azure Data Factory is a Microsoft cloud service offered by the Azure platform that allows data integration from many different sources. Azure Data Factory is a perfect solution when in need of building hybrid extract-transform-load (ETL), extract-load-transform (ELT) and data integration pipelines.
It enables every organization in every industry to use it for a rich variety of use cases: data engineering, migrating their on-premises SSIS packages to Azure, operational data integration, analytics, ingesting data into data warehouses, and more.

3.4.2 Azure Data Factory logo

Components Of Data Factory

Data Factory is composed of four key elements. All these components work together to provide the platform on which you can form a data-driven workflow with the structure to move and transform the data.

Pipeline: A data factory can have one or more pipelines. It is a logical grouping of activities that perform a unit of work. The activities in a pipeline perform the task altogether. For example - a pipeline can contain a group of activities that ingests data from an Azure blob and then runs a Hive query on an HDInsight cluster to partition the data.

Activity: It represents a processing step in a pipeline. For example - we might use a copy activity to copy data from one data store to another data store.

Datasets: It represents data structures within the data stores, which point to or reference the data we want to use in our activities as I/O.

Linked Services: It is like connection strings, which define the connection information needed for Data Factory to connect to external resources. A Linked service can be a data store and compute resource. Linked service can be a link to a data store, or a computer resource also.

Triggers: It represents the unit of processing that determines when a pipeline execution needs to be disabled. We can also schedule these activities to be performed at some point in time, and we can use the trigger to disable an activity.

Control flow: It is an orchestration of pipeline activities that include chaining activities in a sequence, branching, defining parameters at the pipeline level, and passing arguments while invoking the pipeline on-demand or from a trigger. We can use control flow to sequence certain activities and also define what parameters need to be passed for each of the activities.

Advantages Of Azure

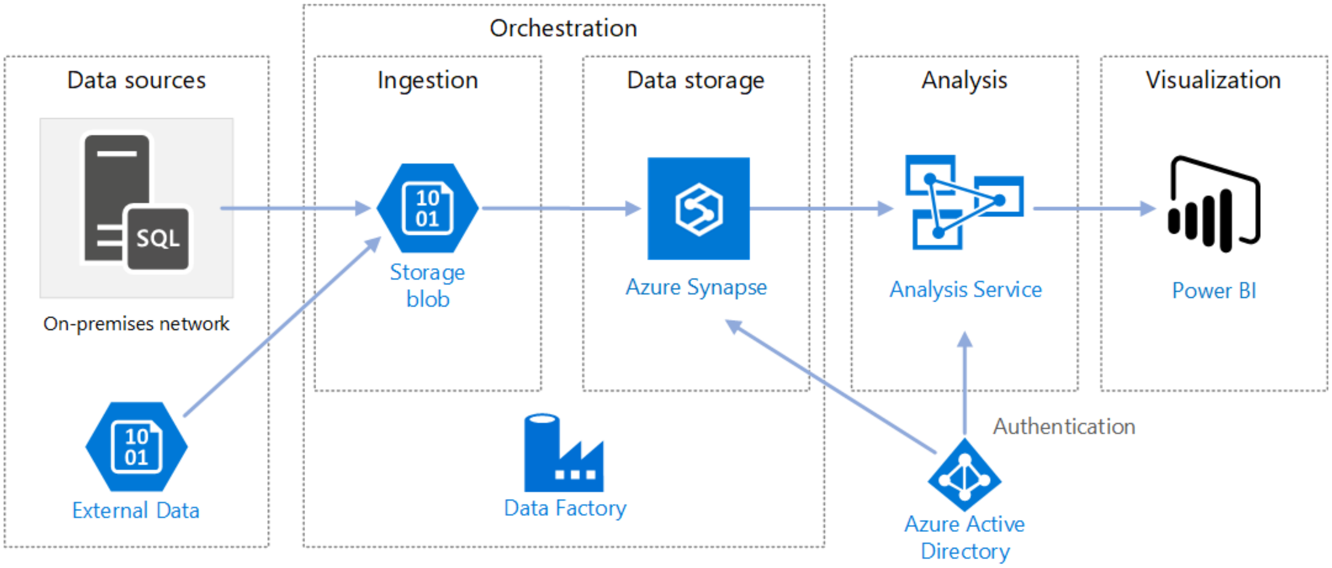
Easy Migration of ETL Workloads to Cloud: You can easily migrate ETL workloads from on-premises EDWs and Data Lakes, to the Azure cloud. ADF use can be used to deploy, run, and manage ETL packages.

Low Learning Curve: Azure Data Factory GUI resembles the other ETL GUIs. Thus, ADF offers a low learning curve for developers familiar with the other ETL interfaces.

Code-free Data Transformation: Traditionally, creating data transformations requires ready code. However, with the mapping data flow capability of Azure Data Factory, you can visually build data transformations without writing code. Essentially, ADF gives an unprecedented opportunity to design and run data transformations in a code-free environment helping companies to focus on business logic.

Better Scalability & Performance: Classic ETL systems were not designed to handle huge volume of data. On the contrary, ADF is a scalable platform that comes inbuilt with parallelism and time-slicing features, allowing users to migrate large amounts (terabytes or petabytes) of data to the cloud in a few hours.

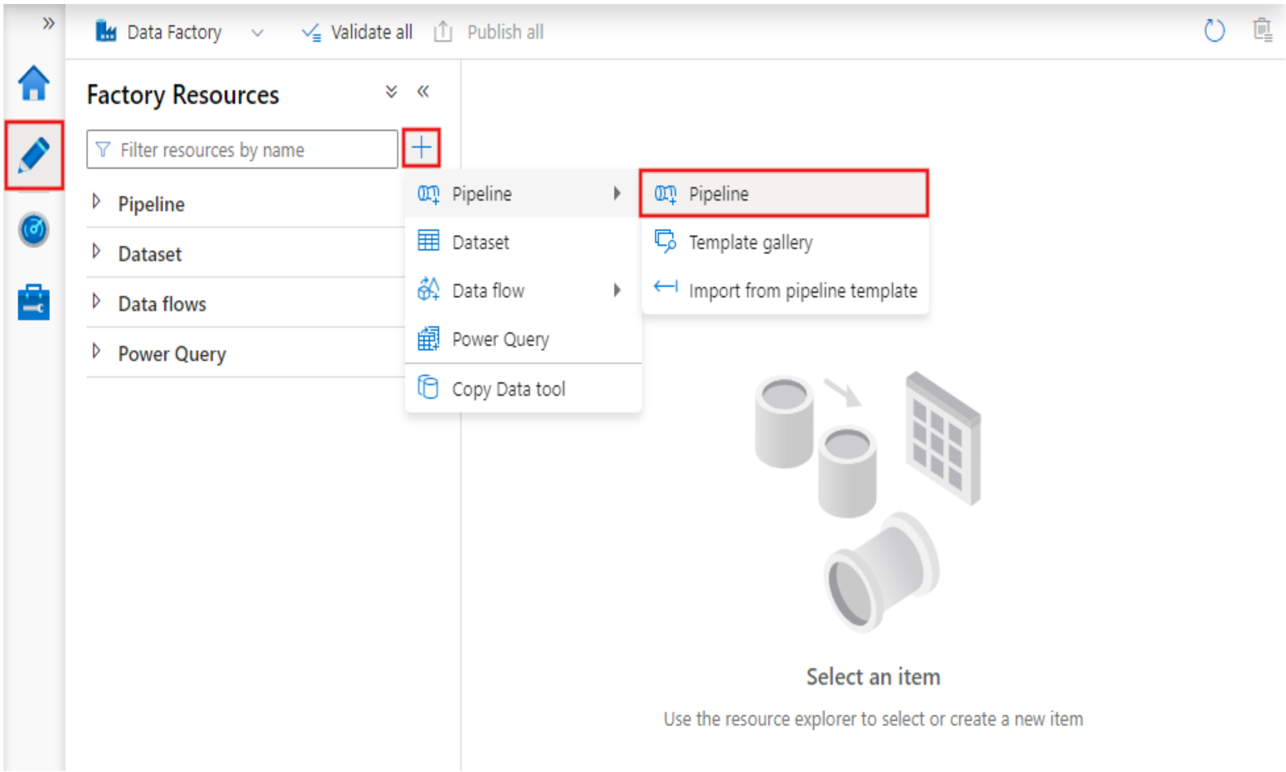
Cost-Efficient Platform: Legacy ETL tools have high licensing fees. Additional cost is required for purchasing hardware and maintaining the tools to cope up with the increasing data volume. With ADF's pay-as-you go service, users won't have to pay upfront costs and are only charged for the services they use.



3.4.3 Azure Data Factory Usage

Creating a Pipeline in ADF

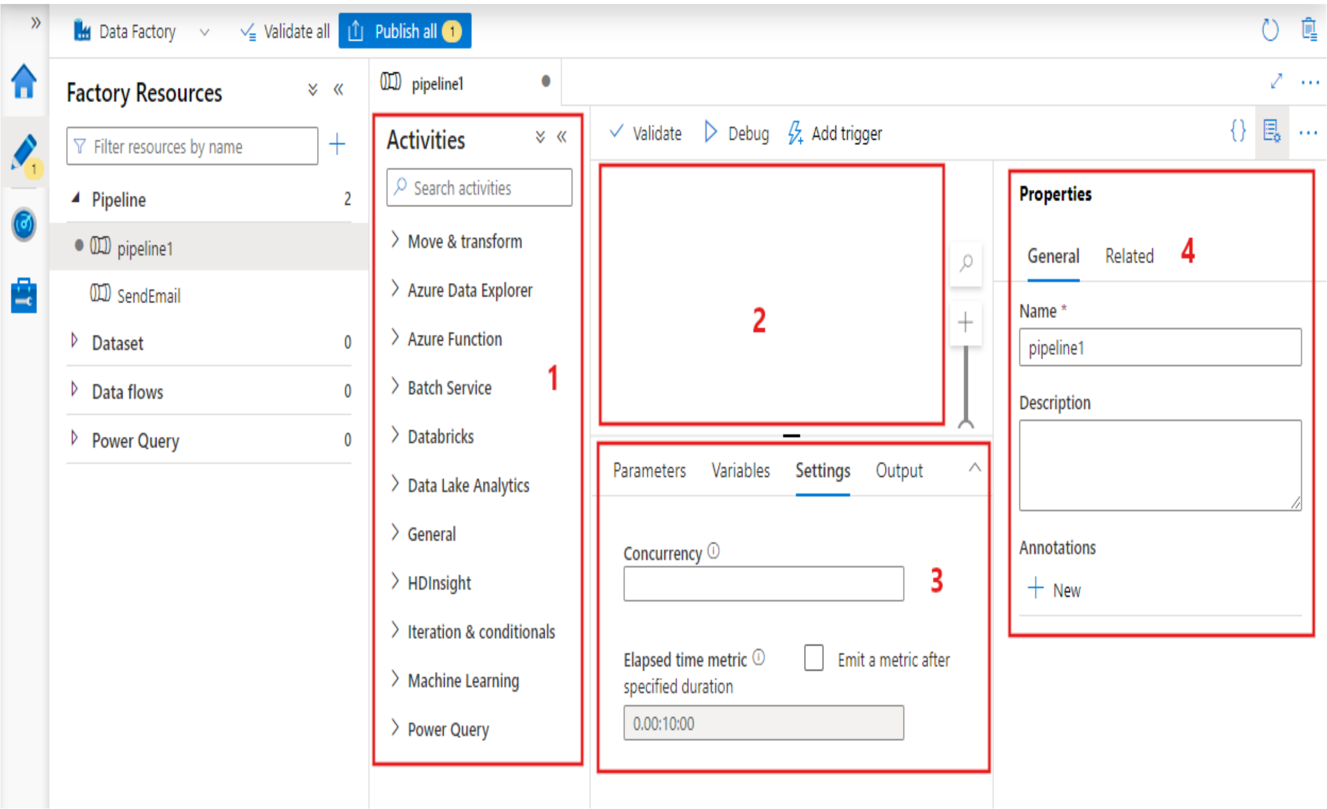
To create a new pipeline, navigate to the Author tab in Data Factory Studio (represented by the pencil icon), then click the plus sign and choose Pipeline from the menu, and Pipeline again from the submenu.



3.4.4 Pipeline Creation

Data factory will display the pipeline editor where you can find:

1. All activities that can be used within the pipeline.
2. The pipeline editor canvas, where activities will appear when added to the pipeline.
3. The pipeline configurations pane, including parameters, variables, general settings, and output.
4. The pipeline properties pane, where the pipeline name, optional description, and annotations can be configured. This pane will also show any related items to the pipeline within the data factory.



Name *: pipeline1

Elapsed time metric: 0.00:10:00

3.4.5 Pipeline UI

3.4.3 SQL Server

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network.

- SQL Server is a database server
- SQL Server is ideal for both small and large applications
- SQL Server supports standard SQL
- SQL Server developed by Microsoft
- SQL Server is a relational database management system.

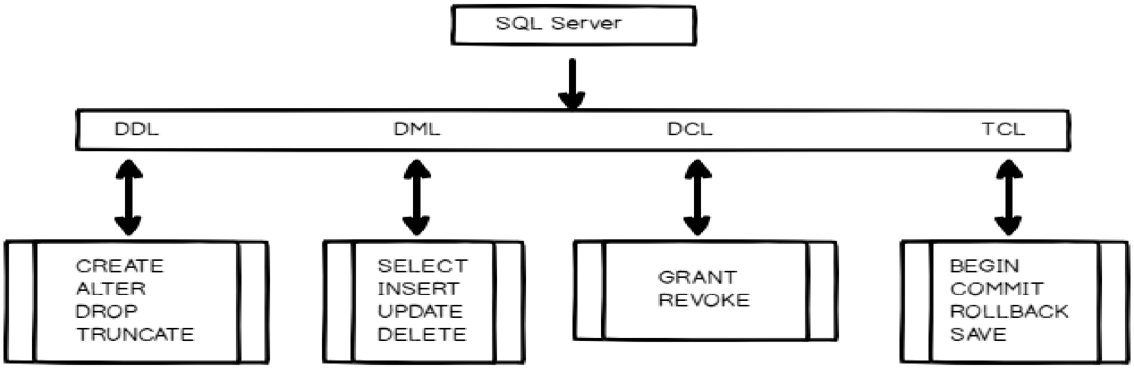
Advantages Of MSSQL Server

It is easy to install Microsoft SQL is easy to use and can be installed via setup wizard. Unlike other database servers requiring extensive command-line configurations, SQL server offers a user-friendly installation interface. Besides the one-click installation process, it comes with a readable

GUI along with lots of instructions.

Enhanced Performance With built-in transparent data compression and encryption features, SQL server offers enhanced performance. To secure and encrypt the data, users need not modify programs. SQL Server provides efficient permission management tools with access controls designed to help users secure sensitive business information.

It is highly secure The SQL Server database is highly secure and uses sophisticated encryption algorithms making it virtually impossible to break the security layers. SQL Server is a commercial relational database with additional security features to reduce the risk of attacks.



3.4.5 SQL Server Diagram

3.4.4 SQL Server Management Studio (SSMS)

SQL Server Management Studio is a free multipurpose integrated tool to access, develop, administer, and manage SQL Server databases, Azure SQL Databases, and Azure Synapse Analytics. SSMS allows you to manage SQL Server using a graphical interface. SSMS can also be used to access, configure, manage & administer Analysis services, Reporting services, & Integration services.

Benefits Of Using SSMS :

- Cost-free
- Advanced user experience
- Various add-in options
- Easy installation

- Manage large data easily
- Faster speed

3.4.5 Azure Databricks

Azure Databricks is a data analytics platform optimized for the Microsoft Azure cloud services platform. Azure Databricks offers three environments for developing data intensive applications: Databricks SQL, Databricks Data Science & Engineering, and Databricks Machine Learning.

- Databricks SQL provides an easy-to-use platform for analysts who want to run SQL queries on their data lake, create multiple visualization types to explore query results from different perspectives, and build and share dashboards.

- Databricks Data Science & Engineering provides an interactive workspace that enables collaboration between data engineers, data scientists, and machine learning engineers. For a big data pipeline, the data (raw or structured) is ingested into Azure through Azure Data Factory in batches, or streamed near real-time using Apache Kafka, Event Hub, or IoT Hub. This data lands in a data lake for long term persisted storage, in Azure Blob Storage or Azure Data Lake Storage. As part of your analytics workflow, use Azure Databricks to read data from multiple data sources and turn it into breakthrough insights using Spark.

- Databricks Machine Learning is an integrated end-to-end machine learning environment incorporating managed services for experiment tracking, model training, feature development and management, and feature and model serving. Simple interface with which users can create a Multi-Cloud Lakehouse structure and perform SQL and BI workloads on a Data Lake. In terms of pricing and performance, this Lakehouse Architecture is 9x better compared to the traditional Cloud Data Warehouses. It provides a SQL-native workspace for users to run performance-optimized SQL queries. Databricks SQL Analytics also enables users to create Dashboards, Advanced Visualizations, and Alerts. Users can connect it to BI tools such as Tableau and Power BI to allow maximum performance and greater collaboration

3.4.6 Python

Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting-edge technology in Software Industry. Python Programming Language is very well suited for Beginners, also for experienced programmers with other programming languages like C++ and Java.
It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

Benefits of using Python:

- Easy to learn
- Availability of support
- Large global community
- In-demand skill in the job market
- Free learning resources
- Extensive libraries
- Powerful frameworks
- Works on any computer
- Versatility
- Productivity and workflow speed
- Fast prototype development
- Free and open-source

3.4.7 Spyder

Spyder is a powerful scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts. It features a unique combination of the advanced editing, analysis, debugging and profiling functionality of a comprehensive development tool with the data exploration, interactive execution, deep inspection and beautiful visualization capabilities of a scientific package.

Furthermore, Spyder offers built-in integration with many popular scientific packages, including NumPy, SciPy, Pandas, IPython, QtConsole, Matplotlib, SymPy, and more.n Beyond its many built-in features, Spyder can be extended even further via third-party plugins.

Spyder can also be used as a PyQt5 extension library, allowing you to build upon its functionality and embed its components, such as the interactive console or advanced editor, in your own software.

Benefits of using Spyder:

- User level install of the version of python you want
- Able to install/update packages completely independent of system libraries or admin privileges
- Conda tool installs binary packages, rather than requiring compile resources like pip - again, handy if you have limited privileges for installing necessary libraries.
- More or less eliminates the headaches of trying to figure out which version/release of package X is compatible with which version/release of package Y, both of which are required for the install of package Z
- Comes either in full-meal-deal version, with numpy, scipy, PyQt, spyder IDE, etc. or in minimal / alacarte version (miniconda) where you can install what you want, when you need it
- No risk of messing up required system libraries

3.4.8 Visual Studio Code (VSCode)

Visual Studio Code is a code editor in layman's terms. Visual Studio Code is "a free-editor that helps the programmer write code, helps in debugging and corrects the code using the intelli-sense method ". In normal terms, it facilitates users to write the code in an easy manner. Many people say that it is half of an IDE and an editor, but the decision is up to to the coders. Any program/software that we see or use works on the code that runs in the background. Traditionally coding was used to do in the traditional editors or even in the basic editors like notepad! These editors used to provide basic support to the coders. Huge Language Support. Not only is VS Code available cross-platform, it aims to be your one-stop code editor with support for 30+ programming languages out-of-the-box Its features let the user modify the editor as per the usage, which means the user is able to download the libraries from the internet and integrate it with the code as per his requirements

3.4.9 Flux

Flux is a standalone data scripting and query language that increases productivity and code reuse. Flux is optimized for ETL, monitoring, and alerting, with an inline planner and optimizer. Flux is the result of the open source community driving innovation with time series data.
The flux cli is customized and modified using Scala and spark to fit the ETL requirements.

These modifications include features for:

- Copying data from azure data lake storage
- Running the data transformation pipelines
- Managing the delta location data
- Efficient testing and error handling
- Performing SQL operations efficiently during ETL
- Managing different formats of data

3.5 Python Scripts

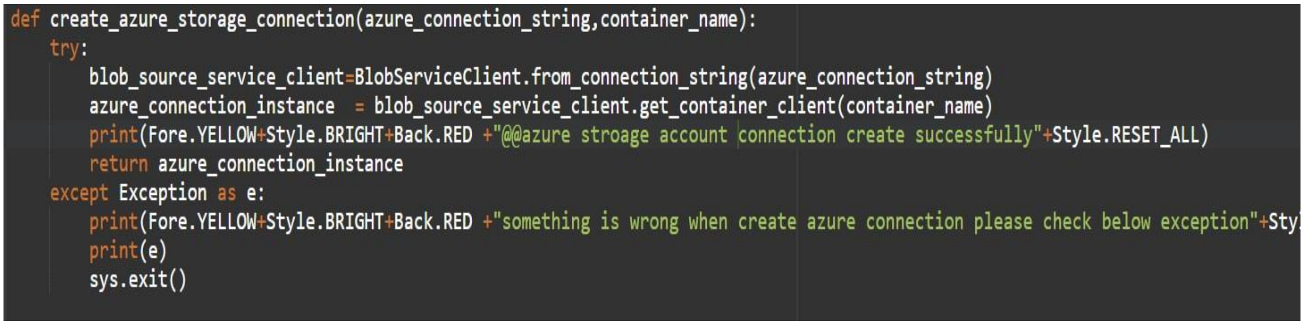
To test the result sets in each stage we created some Python Scripts/Utility which covers following points:

1. Test the csv file data according to spec sheet
2. Test data flow process
3. compare data between csv file and parquet files
4. compare data between two parquet files
5. Read blob file data from azure storage account to perform spec validation
6. compare data between two SQL databases
7. compare data between parquet file and SQL database

3.5.1 Description of some of the functions

The below function help in to create Azure Storage Account Connection.
This function take 2 parameters –

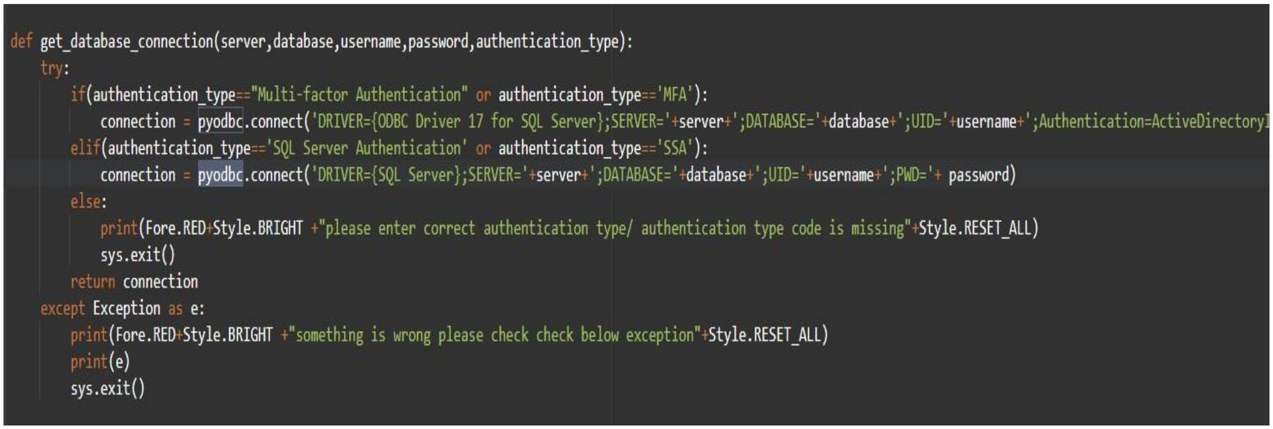
1. Azure storage Account connection string
2. container name



```
def create_azure_storage_connection(azure_connection_string,container_name):
    try:
        blob_source_service_client=BlobServiceClient.from_connection_string(azure_connection_string)
        azure_connection_instance  = blob_source_service_client.get_container_client(container_name)
        print(Fore.YELLOW+Style.BRIGHT+Back.RED +"@@azure stroage account connection create successfully"+Style.RESET_ALL)
        return azure_connection_instance
    except Exception as e:
        print(Fore.YELLOW+Style.BRIGHT+Back.RED +"something is wrong when create azure connection please check below exception"+Sty
        print(e)
        sys.exit()
```

Fig. 3.5.1 Connection function

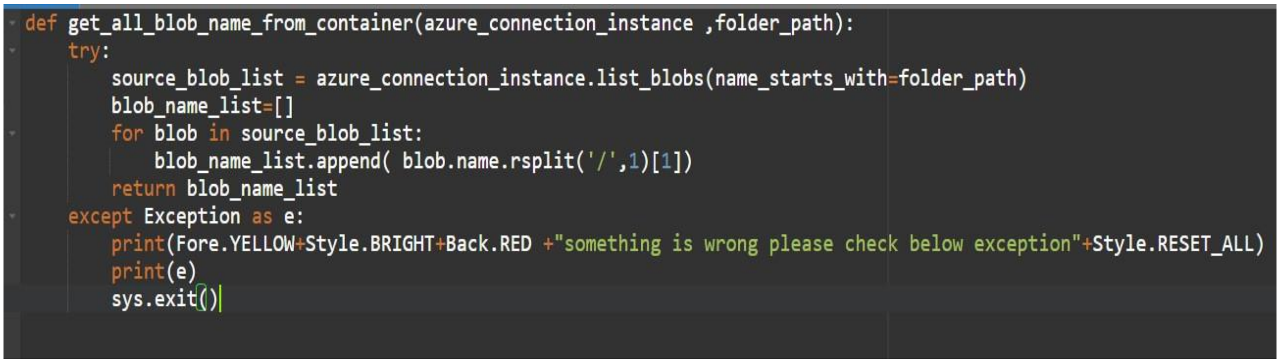
The below function help in to create SQL database connection. It support two Authentication type – MFA(Multi factor Authentication) and SSA(SQL Server Authentication)



```
def get_database_connection(server,database,username,password,authentication_type):
    try:
        if(authentication_type=="Multi-factor Authentication" or authentication_type=='MFA'):
            connection = pyodbc.connect('DRIVER={ODBC Driver 17 for SQL Server};SERVER='+server+';DATABASE='+database+';UID='+username+';Authentication=ActiveDirectoryI
        elif(authentication_type=='SQL Server Authentication' or authentication_type=='SSA'):
            connection = pyodbc.connect('DRIVER={SQL Server};SERVER='+server+';DATABASE='+database+';UID='+username+';PWD='+ password)
        else:
            print(Fore.RED+Style.BRIGHT +"please enter correct authentication type/ authentication type code is missing"+Style.RESET_ALL)
            sys.exit()
        return connection
    except Exception as e:
        print(Fore.RED+Style.BRIGHT +"something is wrong please check check below exception"+Style.RESET_ALL)
        print(e)
        sys.exit()
```

Fig. 3.5.2 SQL Connection function

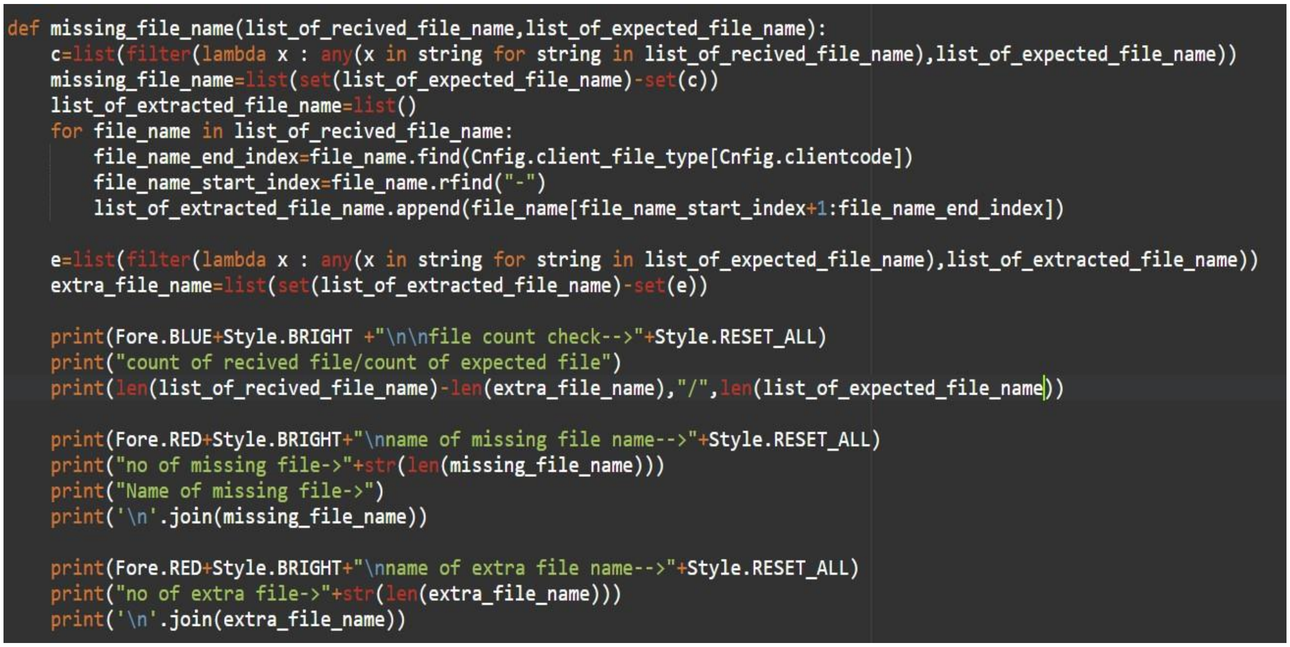
The below function help in get all blob(file) name from azure blob storage folder.



```
def get_all_blob_name_from_container(azure_connection_instance ,folder_path):
    try:
        source_blob_list = azure_connection_instance.list_blobs(name_starts_with=folder_path)
        blob_name_list=[]
        for blob in source_blob_list:
            blob_name_list.append( blob.name.rsplit('/',1)[1])
        return blob_name_list
    except Exception as e:
        print(Fore.YELLOW+Style.BRIGHT+Back.RED +"something is wrong please check below exception"+Style.RESET_ALL)
        print(e)
        sys.exit()
```

Fig. 3.5.3 get file name from container

The below function helps to get the count of received file's name/expected file name, missing file name and extra file name from azure storage account.



```
def missing_file_name(list_of_recived_file_name,list_of_expected_file_name):
    c=list(filter(lambda x : any(x in string for string in list_of_recived_file_name),list_of_expected_file_name))
    missing_file_name=list(set(list_of_expected_file_name)-set(c))
    list_of_extracted_file_name=list()
    for file_name in list_of_recived_file_name:
        file_name_end_index=file_name.find(Cnfig.client_file_type[Cnfig.clientcode])
        file_name_start_index=file_name.rfind("-")
        list_of_extracted_file_name.append(file_name[file_name_start_index+1:file_name_end_index])

    e=list(filter(lambda x : any(x in string for string in list_of_expected_file_name),list_of_extracted_file_name))
    extra_file_name=list(set(list_of_extracted_file_name)-set(e))

    print(Fore.BLUE+Style.BRIGHT +"\n\nfile count check-->"+Style.RESET_ALL)
    print("count of recived file/count of expected file")
    print(len(list_of_recived_file_name)-len(extra_file_name),"/",len(list_of_expected_file_name))

    print(Fore.RED+Style.BRIGHT+"\nname of missing file name-->"+Style.RESET_ALL)
    print("no of missing file->"+str(len(missing_file_name)))
    print("Name of missing file->")
    print('\n'.join(missing_file_name))

    print(Fore.RED+Style.BRIGHT+"\nname of extra file name-->"+Style.RESET_ALL)
    print("no of extra file->"+str(len(extra_file_name)))
    print('\n'.join(extra_file_name))
```

Fig. 3.5.4 fetch file details

And many more such type of functions…..

3.6 Conclusion

In my project MSSQL Server is used as a back end. It became very helpful for me to manage and debug ETL processes and database.
SQL provides the better interaction to the database which is sufficient to this type of project and it is capable to bear the load of the memory storage of any web language information. This project is very helpful for health industries and revenue cycle management.

Chapter-4

TESTING

Before implementing the new system into operations, a test run of the system is doneremoving all the bugs, if any. It is an important phase of a successful system. after codifying the whole programs of the system, attest plan should be developed and run on given set of test data. The output of the test run should match the expected results. This is the most important phase of the system development. The user carries out this testing and test data is also prepared by the user to check for all possible combination of correct data as well as the wrong data i.e. trapped by the system. So, the testing phase consists of the following steps:

Using the test data following test run are carried out:

- Unit testing
- System testing
- Parallel testing

4.1 Unit testing

When the program has been coded and compiled and brought to working condition, they must be individually tested with the prepared test data. Any undesirable happening must benoted and debugged.

It is a software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. This testing methodology is done during the development process by the software developers and sometimes QA staff.

In computer programming, unit testing is a software testing method by which individual units of source code—sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures—are tested to determine whether they are fit for use.

Unit testing comprising the set of tests perfoermed by an individual programmer prior to integration of unit into into a large system. The situation is illustrated as follows:

Coding & debugging → Unit testing →Integration testing

4.2 System testing

After carrying out the unit test for each of the programs of the system and when errors are removed, then system test is done. At this stage the test is done on actual data. The complete system is executed on the actual data. At each stage of the execution, the result or output of the system is analyzed.

When it is ensured that the system is running error free, the user is called with their own actual data so that the system could be shown running as per their requirements.

4.3 Parallel testing

The third in the series of tests before handling over the system to the user is the parallel of the old and new system. This provides the better practical support to the persons using the systemfor the first time who may be uncertain or even nervous using it.

The testing will be performed considering the following points:

- Clerical procedure for collection and disposal of results.
- Flow of data within the organization.
- Accuracy of report output.
- Software testing which involves testing of all the programs together. This involves the testing of system software utilities being used and specifically developes application software.
- Incomplete data formats.
- Halts due to the various reasons and restart procedures.

- Range of items and incorrect formats.
- Invalid combination of data records.

4.4 Test Cases

Here are some of the test cases which we have. Mainly we written the test cases in **BDD format**.

1. Given :

Validate schema of source file

When :

Check file name is valid

Check all required field is present

Check data type of field is valid

Check max length of field is valid

Check not null fields

Validate Primary key Constraint

Then :

File name is valid

All required fields are present

Data type of field is valid

Max length of field is valid

Not null fields are fine

Primary key Constraint Validated

4.5 References

1. https://fifthnote.co

2. https://www.ensemblehp.com/

3. Connecting to SQL Server using SSMS - Part 1 - YouTube – SQL full cource

4. 1. Introduction to Azure Data Factory - YouTube – ADF Full Cource

5. https://docs.microsoft.com

6. https://www.tutorialspoint.com/scrum/index.htm

7. Azure Blob - Read using Python - Stack Overflow python –

8. Run sql query on pandas dataframe - Stack Overflow