OTTO VON GUERICKE
UNIVERSITÄT
MAGDEBURG

GLEIF PARSER and Visualizer

Functional Design Document

20 April 2019

Version 1.0

AUTHORS

Name	Role	Department
Bhuvanesh Leelakrishnan	Developer & Team Lead	Data and Knowledge Engineering
Aarthi Ilangovan	Developer & Documentation	Data and Knowledge Engineering
Sowmya Prakash	Developer & Documentation	Data and Knowledge Engineering
Yamuna Nagasandra Rajaiah	Developer & Documentation	Data and Knowledge Engineering

APPROVALS

Approval Date	Approver Role	Approver
21.04.2019	Project Supervisor	Stefan Willi Hart

Table of Contents

1. Introduction 4

1.1 Purpose of the document 4

1.1 Project Scope 5

1.2 Related documents 5

1.4 Risks and Assumptions 6

2. System/ Solution Overview 7

2.1 Implementation workflow 8

2.1.1 Data flow Diagram 8

2.1.2 Application Process-flow 8

2.2 Description of the Process flow 9

2.2.1 Application screen flow 11

2.2.3 Use-case Diagram 13

2.3 Dependencies and Change Impacts 14

2.3.1 System Dependencies 14

3. Functional Design Specifications 15

3.1 General Functional Requirements 15

3.2 Field Level Specification 16

3.3 Use cases 16

3.4 Test Cases 17

4 System Configurations 17

4.1 System Availability and Fault Tolerance 17

4.2 Other System Requirements/ Non-Functional Requirements 18

5. Integration Requirements 19

5.1 Application process flow and Exception handling 19

5.2 Exception Handling/ Error Reporting 20

6. Project Planning and Execution 21

6.1. Gantt Chart 21

1. Introduction

The document is intended to provide high-level description of the Functional Design Description, Functional Specification, System Specification, System Performance, Non-Functional Requirements, Use-case Diagram, Process flow, Application work flow for GLEIF Parser and Visualizer. Detailed explanation of the application processing cycles and process flows enables the audience of this document to understand the usability and functionality of GLEIF Parser and Visualizer. Document also provides the evaluation of the model with an appropriate test case. The document is subjected for the audience with non-technical background of an organization. The direct users of the document are involved directly with the business with little technical background. In order to understand the business use-cases one can leverage the document for better understanding.

The aim of this project is to build an application solution that is capable of Parsing GLEIF XML files, efficient data transformation, data storage into SAP HANA Database, and efficient retrieval of data based on suitable query from the user.

Initially, the GLEIF Parser and Visualizer application takes the input from the user in .zip format. The recent data available on the GLIEF official website is ideal. The LEI (Legal Entity Identifier) data available on the GLEIF server is usually LEVEL 1 data and LEVEL 2 data. The user is provided with an interface to feed the golden copy of the data (XML files) into the application. The Level1 and Level2 data is then subjected to be parsed into flat file format. Further steps are to integrate into the SAP HANA database. User can also search LEI record details by entering one or few search fields. The output is the report of the GLEIF data stored in the database based on search criteria. The search filtering criteria from the user are based on Legal Name, Country Name.

1.1 Purpose of the document

The Functional Design Document for the specific project provides detailed description of the developed GLIEF application and defines the specific functionalities and desired behavior of the software. Document is intended to provide high-level function requirements for the software application built. The audience for this document is directly involved with the business processes who has little knowledge about the technology involved in the development processes. The documents include process flows, System Functionality, System Availability, Software use - cases, system processing details, data flow diagrams and application workflow.

1.1 Project Scope

The project focuses on automatic parsing of the XML data files of the GLIEF server into SAP-HANA database server which then can be used for data search and data view. The input for the GLIEF Parser and Visualizer application software is the golden copy of the Level1 and Level2 copies of GLIEF data. Goal of the project is providing a time-efficient parsing model. GLEIF data consists of large volume of Level 1 (4 GB) data and Level 2 (20 GB) data. The data consumes enormous amount of processing time in case of legacy system while GLEIF Parser and Visualizer is capable to reduce the processing time by executing the muti-threading process. The project aims to reduce the amount of time required to parse the data. The challenge here is to build a model in order to automatically parse the data given from the user and store the resulted records into SAP-HANA for potentially faster access. Additionally, the user is provided with the ability to search the required contents based on the searching criteria of Legal Name and Country Name. The final result of the framework is to produce reports for data visualization based on query search.

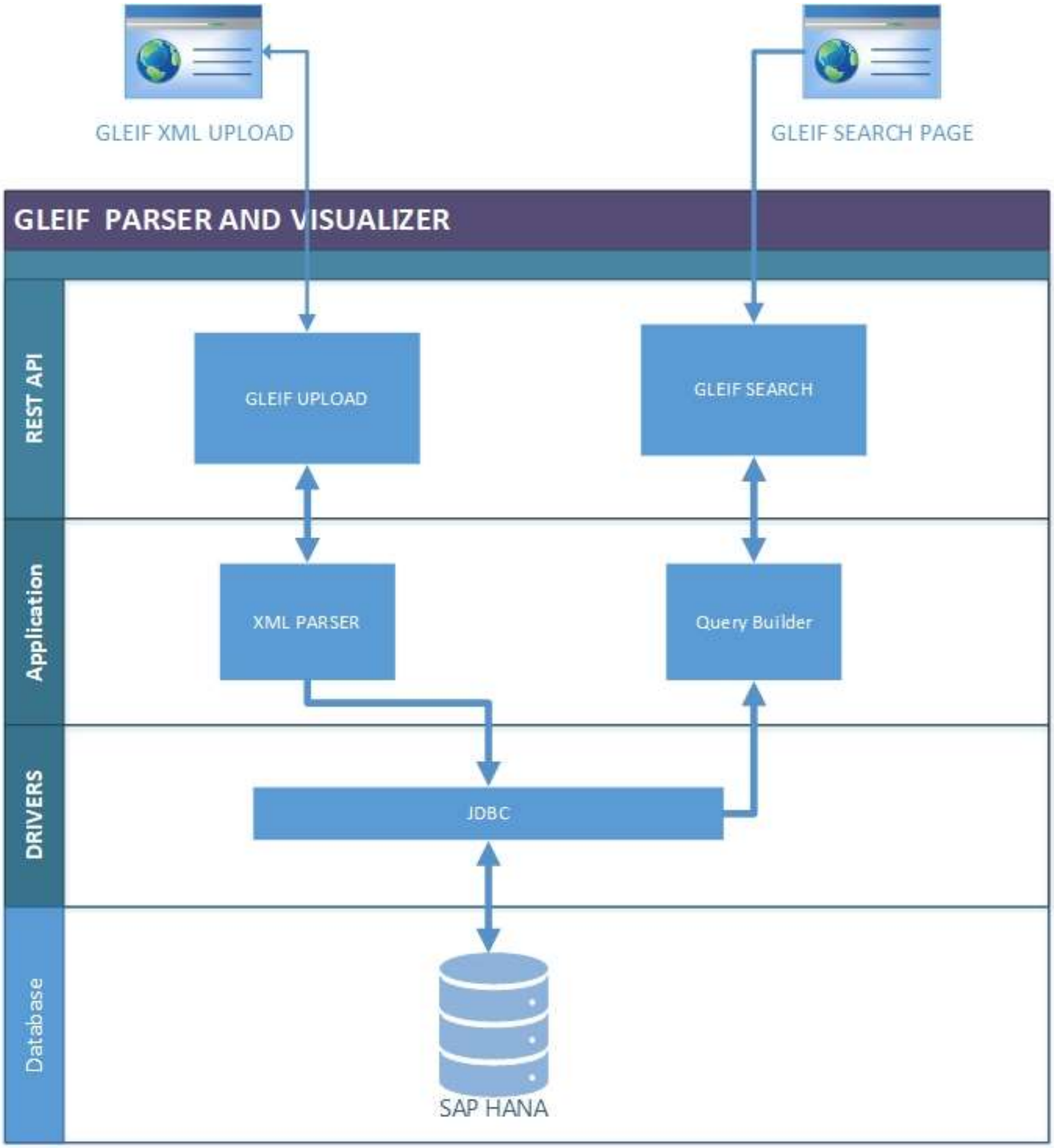
1.2 Related documents

Component	Name	Description
Technical Design Document	High-Level-Technical Design Document (HLD)	Describes the technical specification of the project and aims at the audience such as Development team and Project Management Team
User Guide	User Manual	User document provided to the registered user to be able to use the software application with minimizing the errors.
Implementation Details	Deployment Manual	Technical Document describing the step by step procedure involved in the Implementation of the solution. Helps in successful execution of the software application with ability of Fault Tolerance.
Test Results	Test Results Document	Describes various testing scenarios and application test cases results.

1.4 Risks and Assumptions

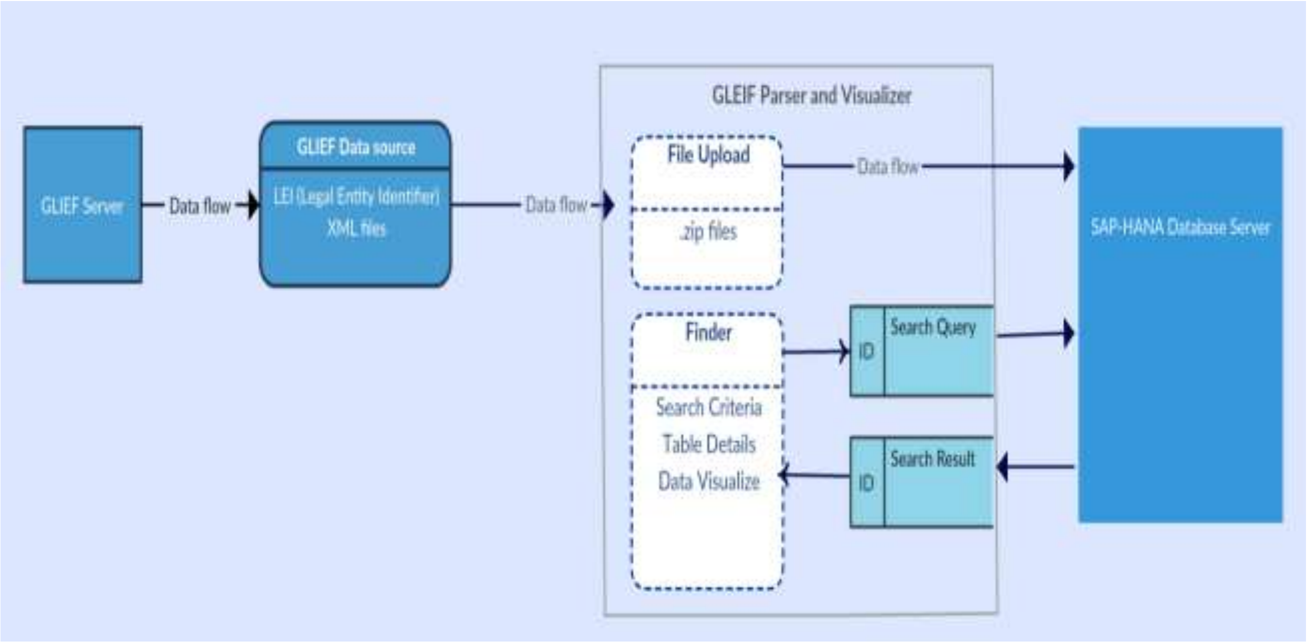
RISKS	ASSUMPTIONS
Business Constraints	Satisfy the budget and resources assigned on the project. Meet the SLA’s (Service Level Agreement)
Technical Constraints	Implementation limited based on programming languages and their dependencies, hardware infrastructure assumed.
Application Server Failure	Code backup in case of server failure or version conflict.
Database Failure	Disaster Recovery by maintaining table scripts

2. System/ Solution Overview

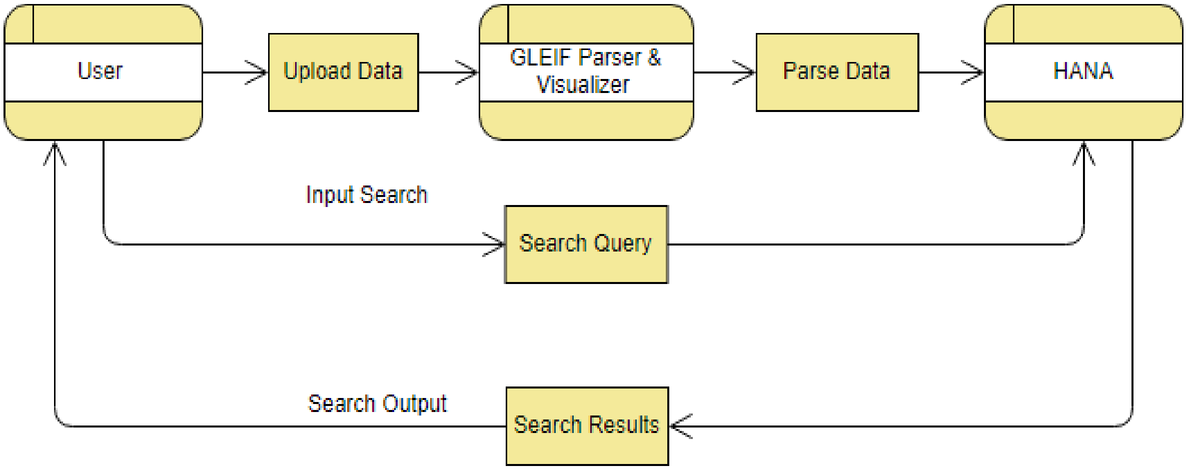


2.1 Implementation workflow

2.1.1 Data flow Diagram

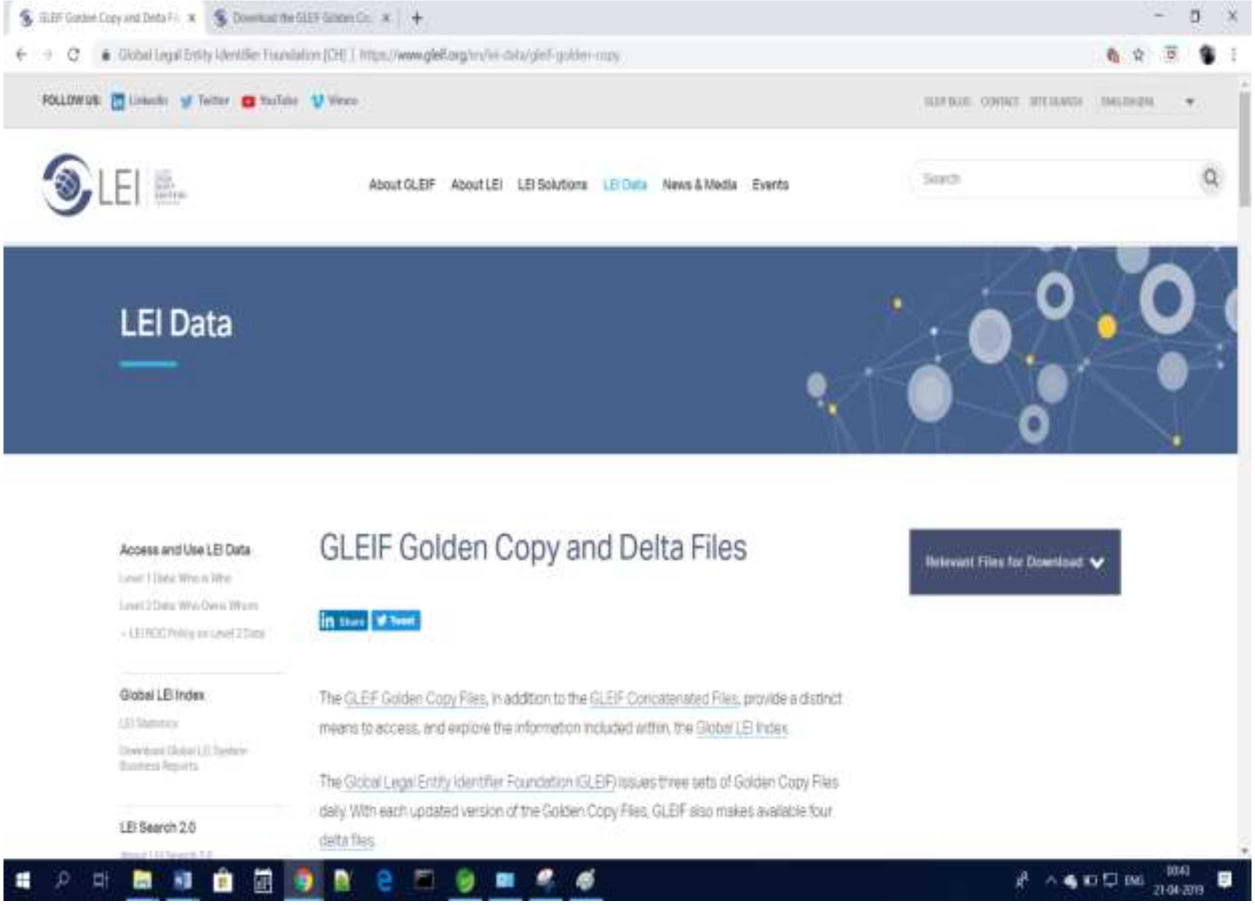


2.1.2 Application Process-flow



2.2 Description of the Process flow

- **Step 1:** user access the GLIEF data server over the website: https://www.gleif.org/en/lei-data/access-and-use-lei-data/

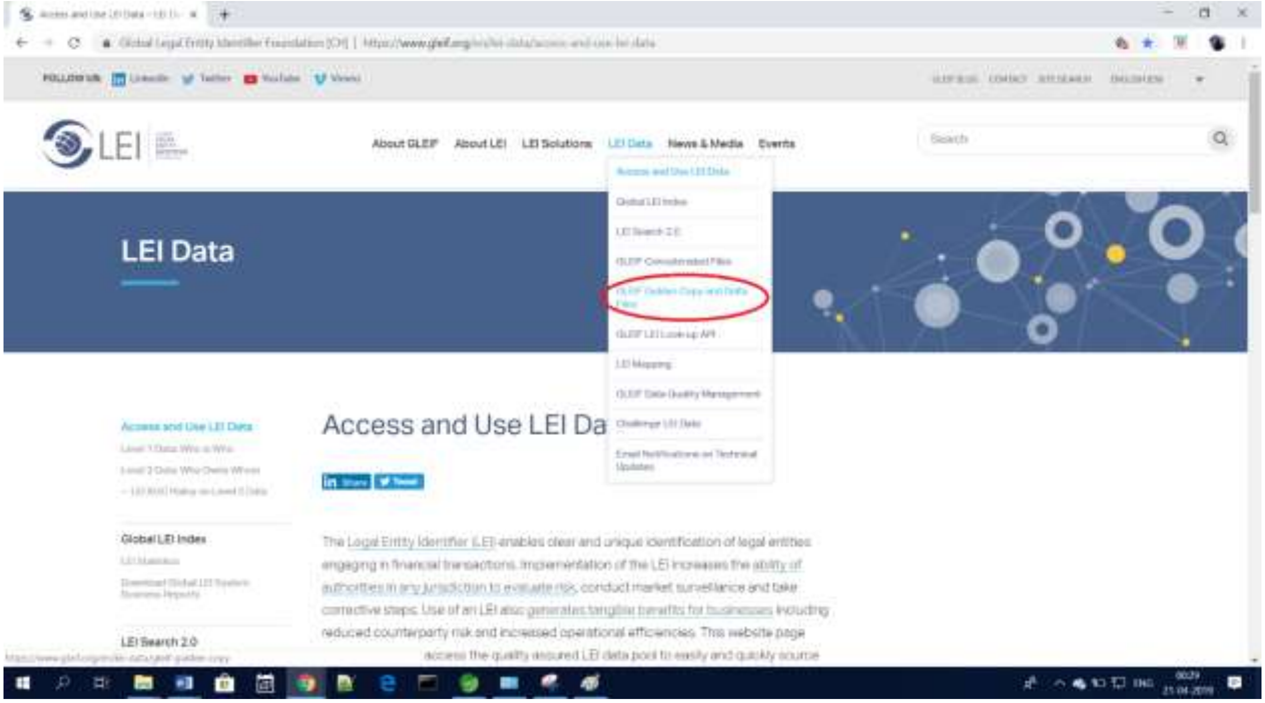

GLEIF Golden Copy and Delta Fi
Download the GLEIF Golden Co
Global Legal Entity Identifier Foundation (CH) | https://www.gleif.org/en/lei-data/gleif-golden-copy
FOLLOW US: LinkedIn Twitter YouTube Vimeo
LEI
About GLEIF
About LEI
LEI Solutions
LEI Data
News & Media
Events
Search
LEI Data
Access and Use LEI Data
Level 1 Data: Who is Who
Level 2 Data: Who Owns Whom
+ LEI ROC Policy on Level 2 Data
Global LEI Index
LEI Statistics
Download Global LEI System Business Reports
LEI Search 2.0
Share
Tweet
Relevant Files for Download

GLEIF Golden Copy and Delta Files

The GLEIF Golden Copy Files, in addition to the GLEIF Concatenated Files, provide a distinct means to access, and explore the information included within, the Global LEI Index.

The Global Legal Entity Identifier Foundation (GLEIF) issues three sets of Golden Copy Files daily. With each updated version of the Golden Copy Files, GLEIF also makes available four delta files.

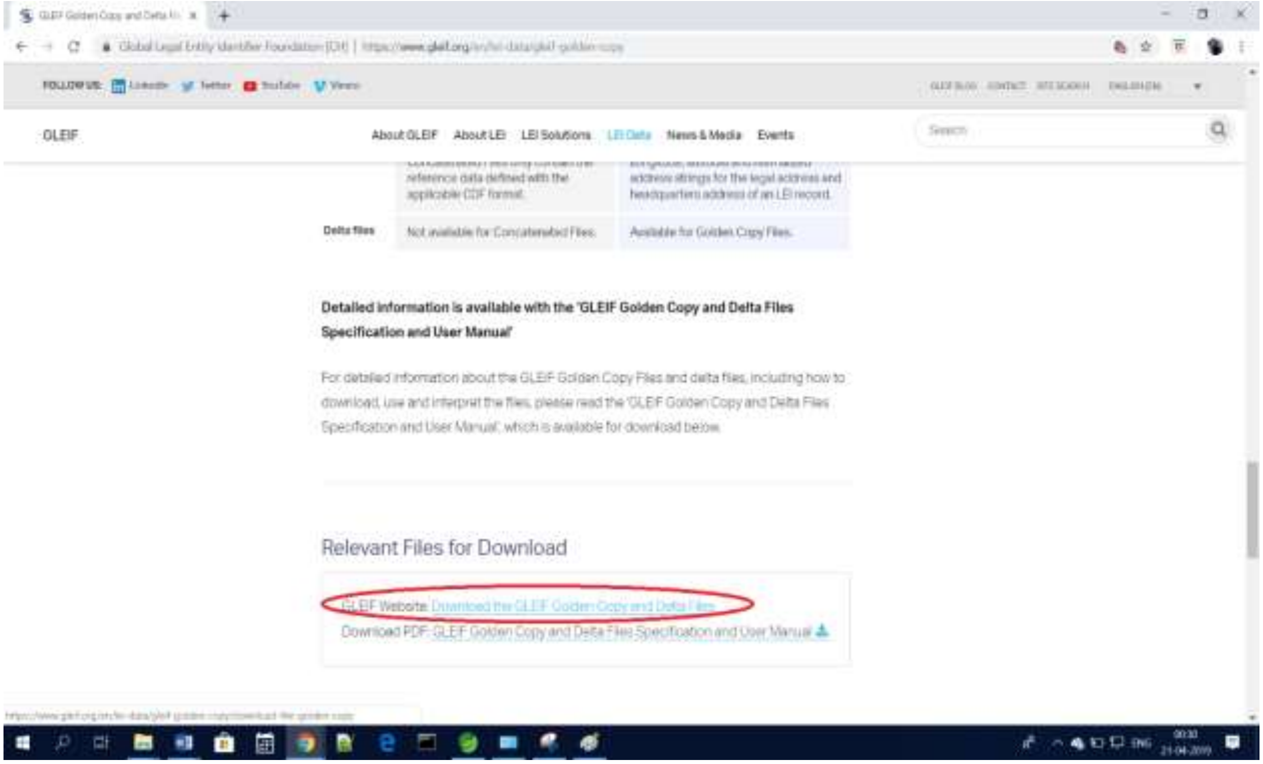
- **Step 2:** LEI (Legal Entity Identifier) golden copies data collection from the user from the defined GLIEF data source.



Access and Use LEI Da

The Legal Entity Identifier (LEI) enables clear and unique identification of legal entities engaging in financial transactions. Implementation of the LEI increases the ability of authorities in any jurisdiction to evaluate risk, conduct market surveillance and take corrective steps. Use of an LEI also generates tangible benefits for businesses including reduced counterparty risk and increased operational efficiencies. This website page access the quality assured LEI data pool to easily and quickly source

- **Step 3:** The LEI data available on GLIEF server are XML, JSON and CSV data format.



	reference data defined with the applicable CDF format.	address strings for the legal address and headquarters address of an LEI record.
Delta files	Not available for Concatenated Files.	Available for Golden Copy Files.

Detailed information is available with the 'GLEIF Golden Copy and Delta Files Specification and User Manual'

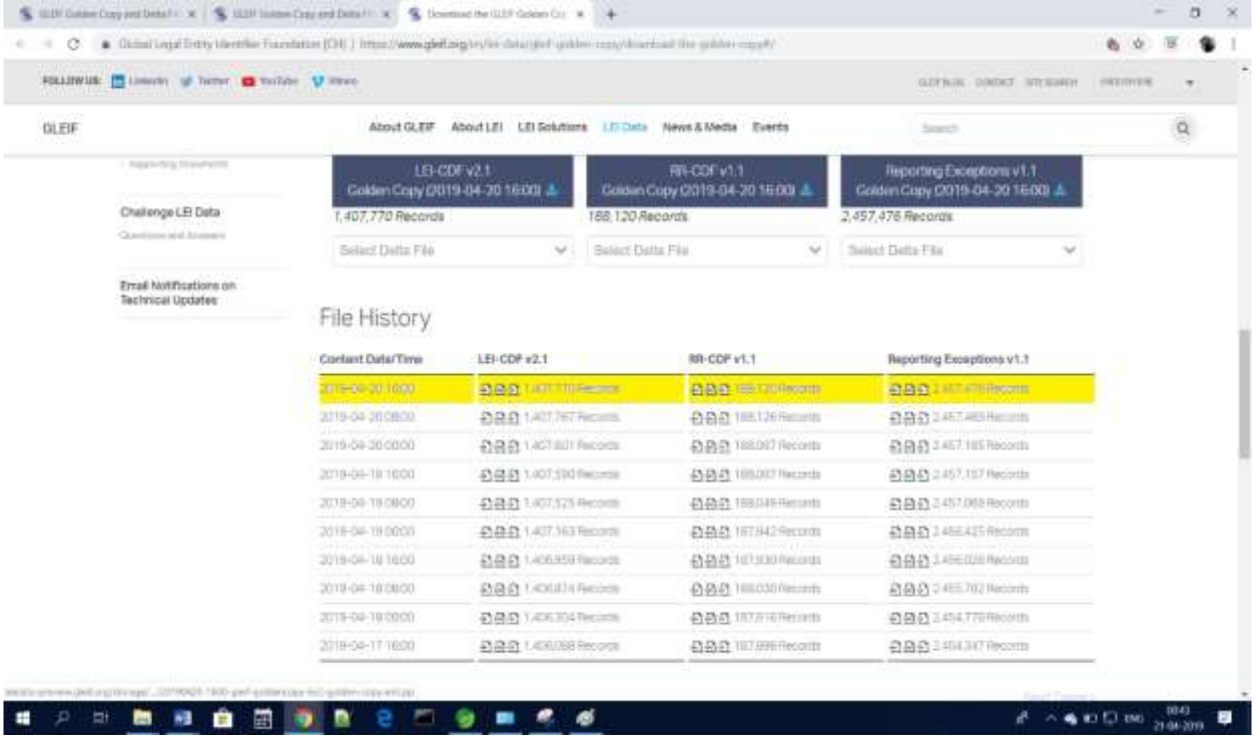
For detailed information about the GLEIF Golden Copy Files and delta files, including how to download, use and interpret the files, please read the 'GLEIF Golden Copy and Delta Files Specification and User Manual', which is available for download below.

Relevant Files for Download

GLEIF Website: Download the GLEIF Golden Copy and Delta Files

Download PDF: GLEIF Golden Copy and Delta Files Specification and User Manual

- **Step 4:** The LEI data available on GLIEF server are XML, JSON and CSV data format. Downloadable XML files are visible at the File History tab.


FOLLOW US: LinkedIn Twitter YouTube Vimeo
GLEIF
About GLEIF About LEI LEI Solutions LEI Data News & Media Events
Search
Challenge LEI Data
Questions and Answers
Email Notifications on Technical Updates

LEI-CDF v2.1 Golden Copy (2019-04-20 16:00) — 1,407,770 Records

RR-CDF v1.1 Golden Copy (2019-04-20 16:00) — 188,120 Records

Reporting Exceptions v1.1 Golden Copy (2019-04-20 16:00) — 2,457,476 Records

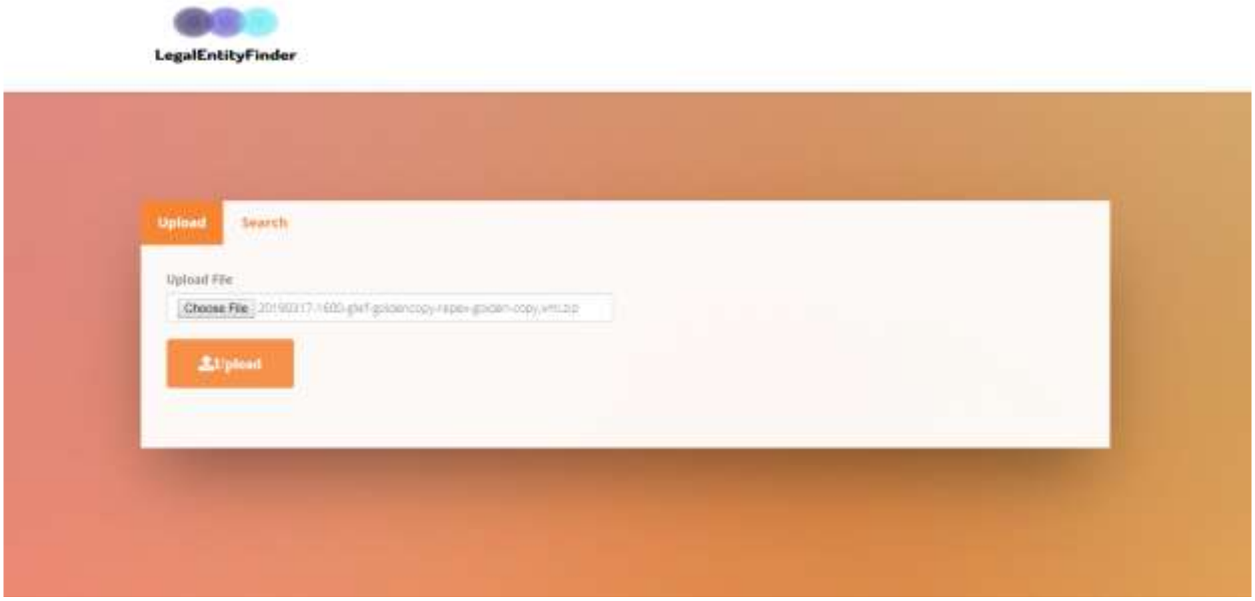
Select Delta File

File History

Content Date/Time	LEI-CDF v2.1	RR-CDF v1.1	Reporting Exceptions v1.1
2019-04-20 16:00	1,407,770 Records	188,120 Records	2,457,476 Records
2019-04-20 08:00	1,407,767 Records	188,126 Records	2,457,465 Records
2019-04-20 00:00	1,407,601 Records	188,097 Records	2,457,185 Records
2019-04-19 16:00	1,407,590 Records	188,007 Records	2,457,157 Records
2019-04-19 08:00	1,407,525 Records	188,046 Records	2,457,069 Records
2019-04-19 00:00	1,407,163 Records	187,942 Records	2,456,425 Records
2019-04-18 16:00	1,406,959 Records	187,930 Records	2,456,026 Records
2019-04-18 08:00	1,406,874 Records	188,030 Records	2,455,782 Records
2019-04-18 00:00	1,406,304 Records	187,816 Records	2,454,779 Records
2019-04-17 16:00	1,406,088 Records	187,898 Records	2,454,347 Records

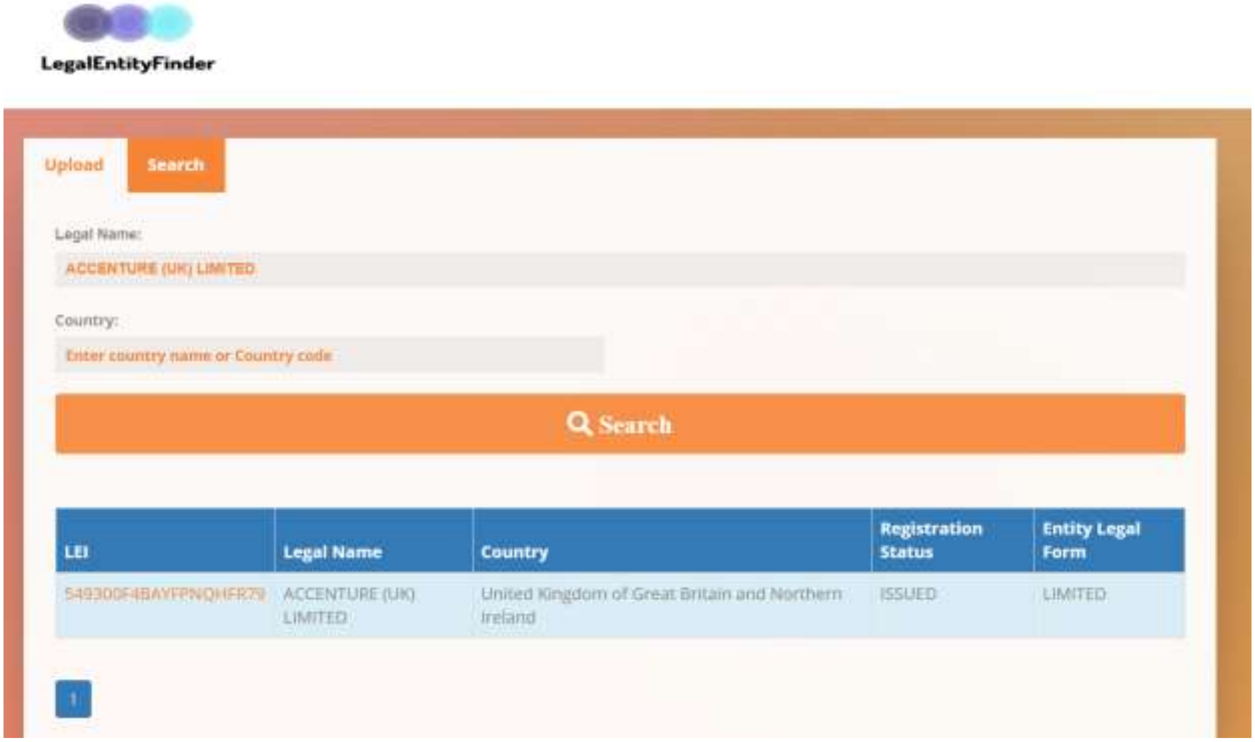
2.2.1 Application screen flow

- **Step 5:** Upload the available downloaded XML data copy into the GLEIF Parser and Visualizer



Upload File: 20190317-1600-gleif-goldencopy-repex-golden-copy.xml.zip

- **Step 6:** User ability to retrieve the required search fields out of available Content filters such as Legal Name, Country Name

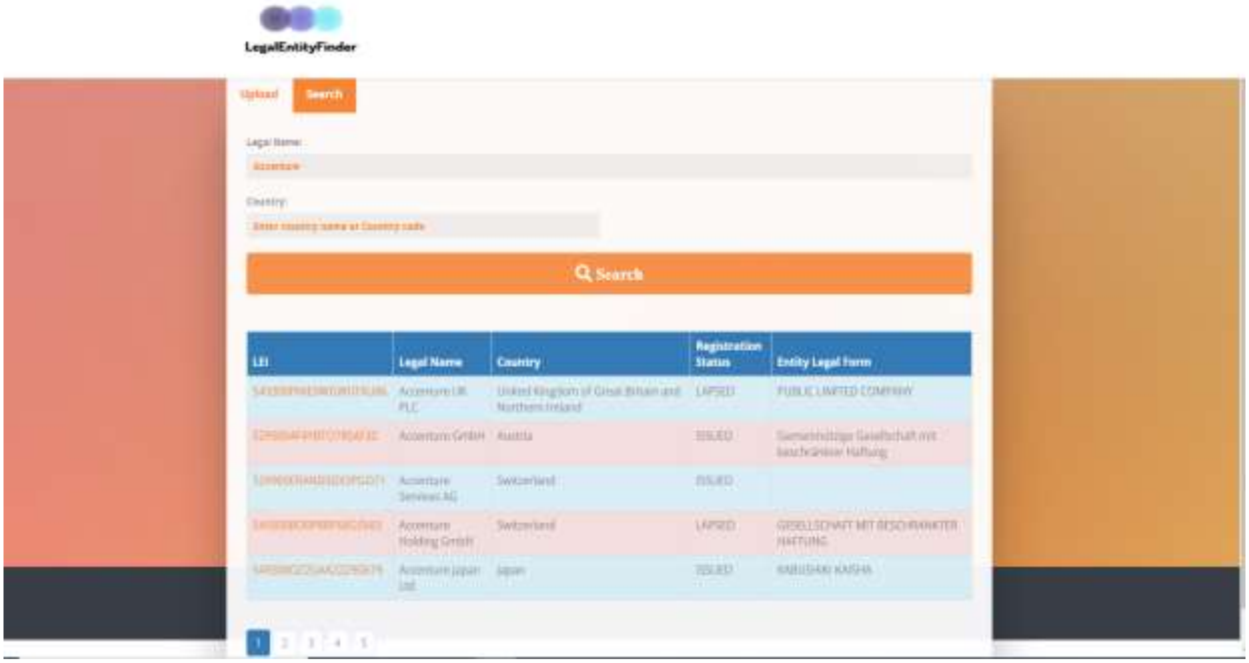


Legal Name: ACCENTURE (UK) LIMITED

Country: Enter country name or Country code

LEI	Legal Name	Country	Registration Status	Entity Legal Form
549300F4BAYFPNQHFR79	ACCENTURE (UK) LIMITED	United Kingdom of Great Britain and Northern Ireland	ISSUED	LIMITED

- **Step 7:** User is able to retrieve search results the data based.

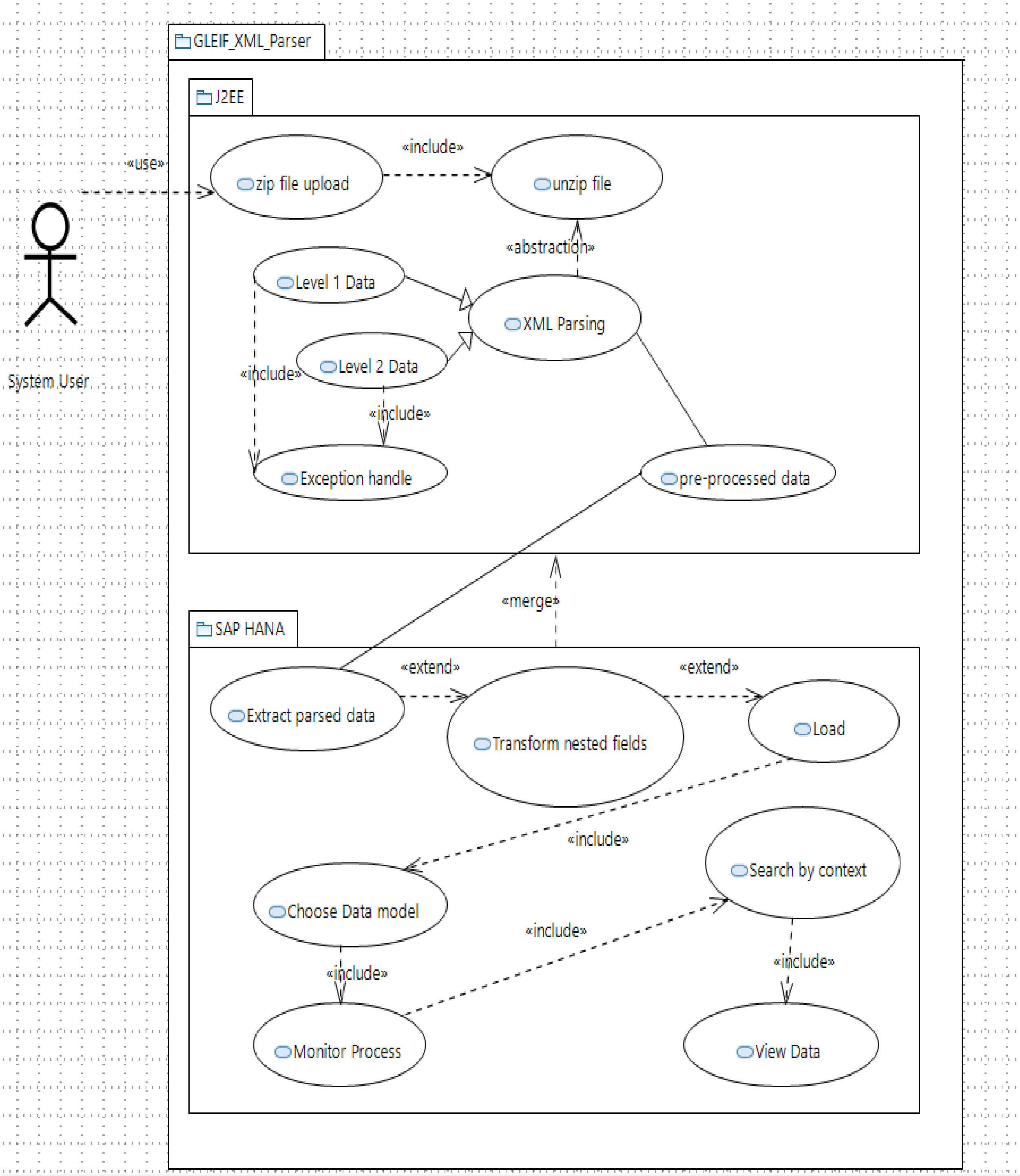


Legal Name: Accenture

Country: Enter country name or Country code

LEI	Legal Name	Country	Registration Status	Entity Legal Form
[illegible]	Accenture UK PLC	United Kingdom of Great Britain and Northern Ireland	LAPSED	PUBLIC LIMITED COMPANY
[illegible]	Accenture GmbH	Austria	ISSUED	Gemeinnützige Gesellschaft mit beschränkter Haftung
[illegible]	Accenture Services AG	Switzerland	ISSUED	
[illegible]	Accenture Holding GmbH	Switzerland	LAPSED	GESELLSCHAFT MIT BESCHRANKTER HAFTUNG
[illegible]	Accenture Japan Ltd	Japan	ISSUED	KABUSHIKI KAISHA

2.2.3 Use-case Diagram



2.3 Dependencies and Change Impacts

2.3.1 System Dependencies

ID	Components	Description
1	Hardware Dependency	Operating System, Windows 7/8/10/LINUX, Network Bandwidth
2	Memory Requirements	RAM: 2GB
3	System Requirements	SAP-HANA Database server, JDK8 and above
4	Language and Frameworks	Language: JAVA JDK Version: 8 and above / Spring 5.0, Spring Boot version: 2.1+, JPA (Java Persistent API), Gradle 5+ (Build Management), Web socket Programming
5	Others dependency	Parallel processing, Optimization techniques using multi-threading

3. Functional Design Specifications

3.1 General Functional Requirements

SECTION ID	FUNCTIONAL REQUIREMENT	EXPECTED OUTPUT	ACTUAL OUTPUT
1	Data collection	Load files from different GLIEF server daily	Data collection form GLEIF server from user
2	File upload	Time-efficient file uploads into SAP HANA database.	Time efficient data upload into SAP HANA database. Parallel execution to load the data to avoid latency.
3	Parsing of XML files	Efficient parsing of XML file into HANA database	Automatic efficient parsing of XML into SAP-HANA database
4	Data transformation	Structured table format	Transform nested fields into structured data table format
5	Data storage	Store data into suitable data model	Columnar data storage into SAP-HANA database for faster access and retrieval
6	Data search and retrieval	Ability to search based on the fields like: Legal Name, Country Name	Performance effective data search based on the fields like: Legal Name, Country Name
7	Data Visualization	Display Required fields	Display Required fields

3.2 Field Level Specification

- Refer to the Technical Design document for the Field Level Specification requirement and implementation details. The technical document describes the overall application software specification of Navigation Menu, Actions, transaction status, verification and validation requirements etc.
- The document also describes the Form Elements, Business rules and Dependencies, Suitable links, “onClick” Events, buttons and links and Navigation.

3.3 Use cases

Use-cases	Description of the use-case
Primary Actor(s)	Application Users,
Business Stakeholders	Business Analysts, Sales operation,
Trigger	File upload, View Search Results
Pre-conditions	Application up and running in the user’s system
Post-conditions	User can view the uploaded results in the Search tab in application
Main Success Scenario	Uploaded LEI (Legal Entity Identifier) XML Files parsed successfully and stored in SAP HANA Database
Extensions	Mentioned stored results can be viewed through the Search tab present in the application
Special Requirements	System requirements necessary to be satisfied for running the application successfully due to large file size

3.4 Test Cases

- Refer to the Test Results document for the Test cases implementation along with the Test Results and success scenarios.

4 System Configurations

4.1 System Availability and Fault Tolerance

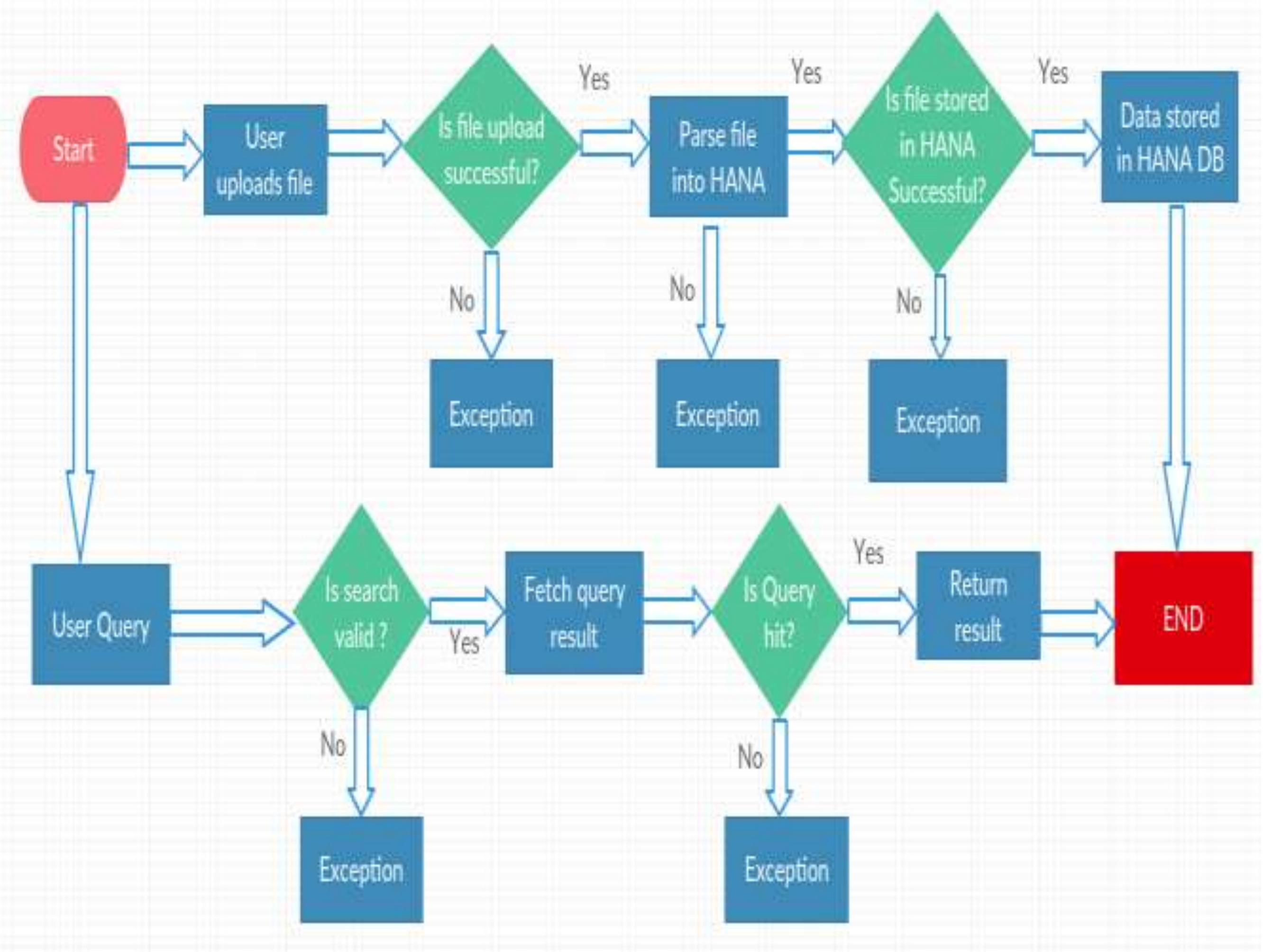
STEPS	DESCRIPTION	SYSTEM AVAILABILITY
Step 1	Collect XML documents from GLEIF server	GLEIF Interface, GLEIF Server, Server access, Network Bandwidth, User availability, System Memory (2GB)
Step 2	Upload the XML files into GLEIF Parser and Visualizer	GLEIF Parser and Visualizer Application Server Engine, Thymeleaf, SAP-HANA Database, JPA/Hibernate, Tomcat server, JAVA version: 8
Step 3	Parsing the XML files into SAP-HANA Database	SAP-HANA Database, JAXB/ Stax, Network Bandwidth
Step 4	Search the required results	GLEIF Parser and Visualizer Application Server Engine, Thymeleaf, Integrated Table access to SAP-HANA Database
Step 5	Search and retrieval	Ability to search based on the fields like: Legal Name, Country Name
Step 6	Data storage	HANA Database, JPA/Hibernate
Step 7	Data Visualization	GLEIF Parser and Visualizer Application Server

4.2 Other System Requirements/ Non-Functional Requirements

Business Requirements	Description	Application Agreements with Stakeholders
Performance (optimum response time)	Application Software with good response time	SLA (Service Level Agreement) between the Clients and the Application provider on Performance impacts of the System
Fault tolerance	Application should be robust, Recover from failure	SLA (Service Level Agreement) to overcome Disaster Recovery
Availability	Uninterrupted availability of system resources and application services.	SLA (Service Level Agreement) for continuous availability/ High availability or Redundancy during component failure
Parallel Processing	Optimizing the components to avoid Latency	SLA (Service Level Agreement) alternatives methods for multiple background tasks
Usability	Availability to multiple users during the required Business running scenarios	SLA (Service Level Agreement) multiple users accessing same system
Maintainability	System Maintenance and Enhancements	SLA (Service Level Agreement) to provide prior awareness about the Maintenance window during system upgrades to avoid Business risks/ Business critical scenarios

5. Integration Requirements

5.1 Application process flow and Exception handling

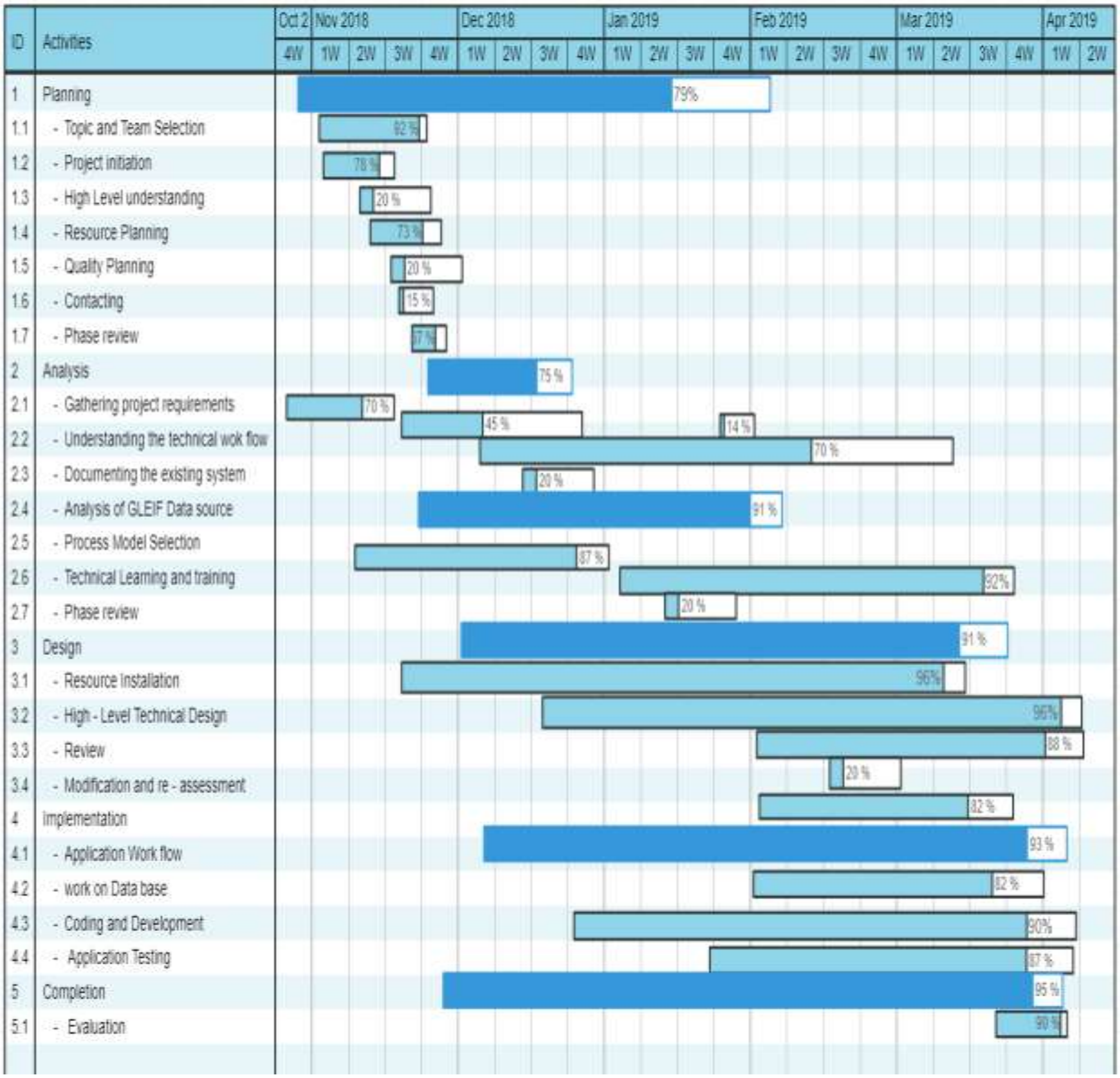


5.2 Exception Handling/ Error Reporting

Exception / Error ID	Error	Cause	Solution Strategy
1	Upload not success Exception	Unable to upload files of another format	Load files that are .zip format. XML files
2	Parse no success Exception	Delta Files upload cause	Acceptance of only fully published XML files
3	File not found Exception	Unable to find the files of specific type	Upload golden copy XML files
4	Null Pointer Exception	Uploaded .zip file is corrupted or empty	Upload correct version of files recently downloaded from GLEIF server
5	Memory Exception	Exception due to lack of system memory	Follow System pre-requisites as mentioned in user / Deployment document

6. Project Planning and Execution

6.1. Gantt Chart


ID
Activities
Oct 2
Nov 2018
Dec 2018
Jan 2019
Feb 2019
Mar 2019
Apr 2019
4W
1W
2W
3W
4W
1 Planning 79%
1.1 - Topic and Team Selection 92 %
1.2 - Project initiation 78 %
1.3 - High Level understanding 20 %
1.4 - Resource Planning 73 %
1.5 - Quality Planning 20 %
1.6 - Contacting 15 %
1.7 - Phase review 67 %
2 Analysis 75 %
2.1 - Gathering project requirements 70 %
2.2 - Understanding the technical wok flow 45 % 14 % 70 %
2.3 - Documenting the existing system 20 %
2.4 - Analysis of GLEIF Data source 91 %
2.5 - Process Model Selection 87 %
2.6 - Technical Learning and training 92%
2.7 - Phase review 20 %
3 Design 91 %
3.1 - Resource Installation 96%
3.2 - High - Level Technical Design 96%
3.3 - Review 88 %
3.4 - Modification and re - assessment 20 % 82 %
4 Implementation 93 %
4.1 - Application Work flow 82 %
4.2 - work on Data base
4.3 - Coding and Development 90%
4.4 - Application Testing 87 %
5 Completion 95 %
5.1 - Evaluation 90 %