Database Project

1 Project title

Translation on order

2 Problem description

The application should provide information about:

- **Currently implemented document translations (Document_ID, Client_ID, Translator_ID, Percentage_share)**
- **Translated documents (Name, Translation_language, Fee, Document_ID, Client_ID)**
- **Translators (Last name, First name, Address, Known_languages, Translator_ID)**
- **Clients (Name, Address, Client_ID)**
- **Available translation languages (Languages)**

3 Data Model

3.1 Glossary

- **Languages: Available languages for document translation.**
- **Fee: Fee charged for the document translation.**
- **Translation_language: Language to which the document is to be translated, if available.**
- **Name (Documents): Name of the document to be translated.**
- **Name (Client): Name of the client who ordered the translation.**
- **Percentage_share: Percentage share of a given translator in the implemented translation.**
- **Known_languages: Languages that the translator can handle.**
- **Client_ID, Document_ID, Translator_ID: Numerical identifiers for the client, document, or translator, respectively.**

3.2 Business Rules

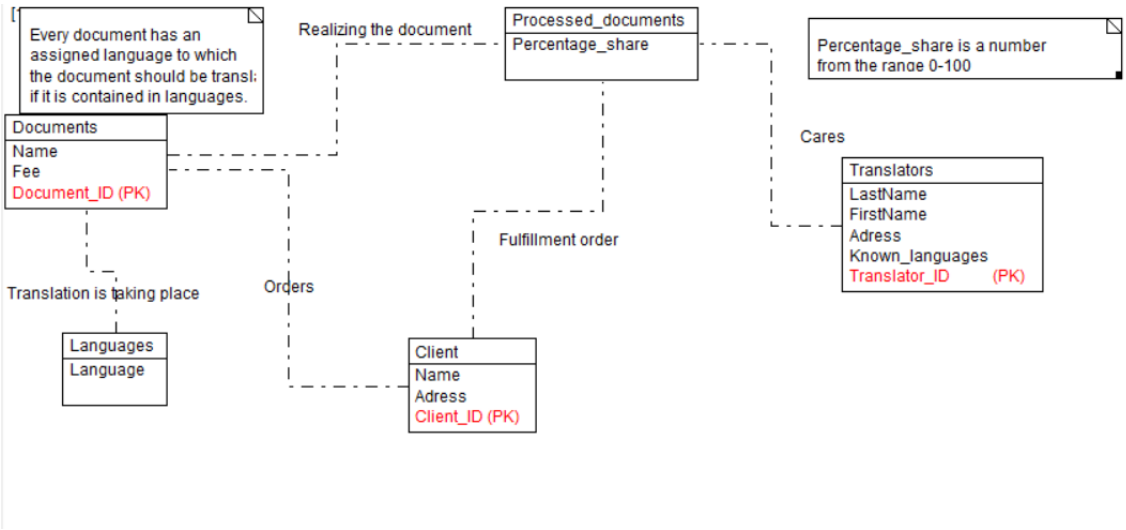
Documents and clients are registered in the database after receiving an order for translation.

The fee for the order is collected at the time of registration and is individually priced by the company.

The company owner assigns translators to documents based on their language skills.

1. **Each document has a designated translation language.**
2. **The translation language must be available in the language table.**
3. **One document can be translated by one or multiple translators.**
4. **A translator can translate multiple documents.**
5. **A client can order the translation of multiple documents.**

3.3 Logic Data Model



3.3.1 Description of Entities, Attributes and their Domains

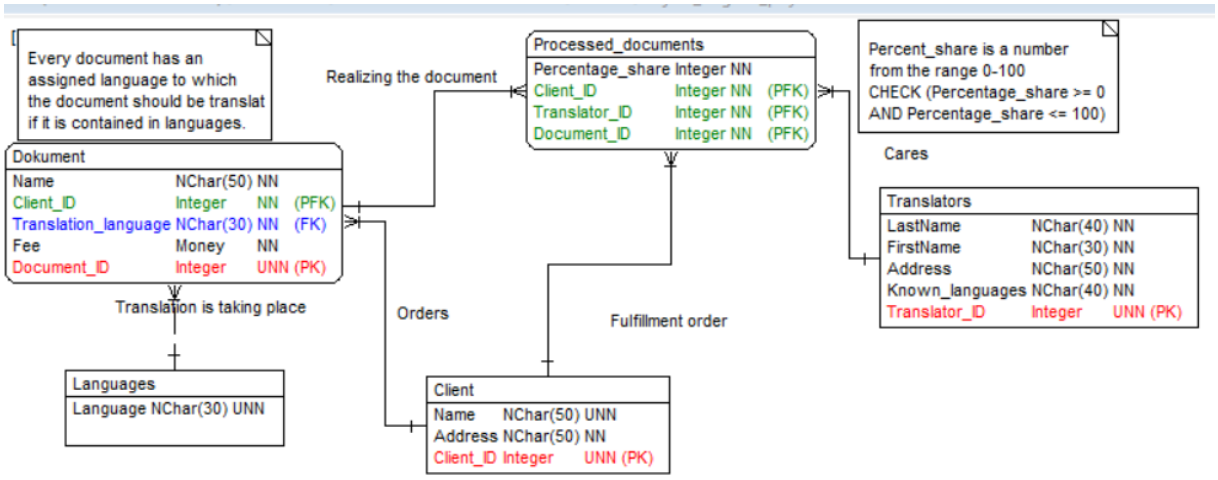
- **Document**
  - **Name - Mandatory, text string no longer than 50 characters**
  - **Fee - Mandatory, number in monetary format (money)**
  - **Document_ID - Mandatory, unique, automatically incrementing number by 1, primary key**
- **Languages**
  - **Language - Mandatory, text string no longer than 30 characters**
- **Client**
  - **Name - Mandatory, text string no longer than 50 characters**
  - **Address - Mandatory, text string no longer than 50 characters**
  - **Client_ID - Mandatory, automatically incrementing number by 1, primary key**
- **Processed_documents**
  - **Percentage_share - Mandatory, number between 0-100**

- **Translator**
  - **Surname - Mandatory, text string no longer than 40 characters**
  - **First Name - Mandatory, text string no longer than 30 characters**
  - **Address - Mandatory, text string no longer than 50 characters**
  - **Known_languages - Mandatory, text string no longer than 40 characters**
  - **Translator_ID - Mandatory, automatically incrementing number by 1, primary key**

3.3.2 Rule Verification

- **Facts 1 and 2 are modeled by a "many-to-many" relationship between the entities "Languages" and "Document"**
  - **The translation language must be present in the languages table**
  - **Each document has its assigned translation language**
- **Facts 3 and 4 are modeled by a "many-to-many" relationship between the entities "Processed_documents" and "Translator"**
  - **A translator can translate multiple documents**
  - **One document can be translated by multiple translators**
- **Fact 5 is modeled by a "many-to-many" relationship between the entities "Client", "Documents" and "Processed_documents"**
  - **A client can commission the translation of multiple documents**

3.4 Physical Data Model (Universal)

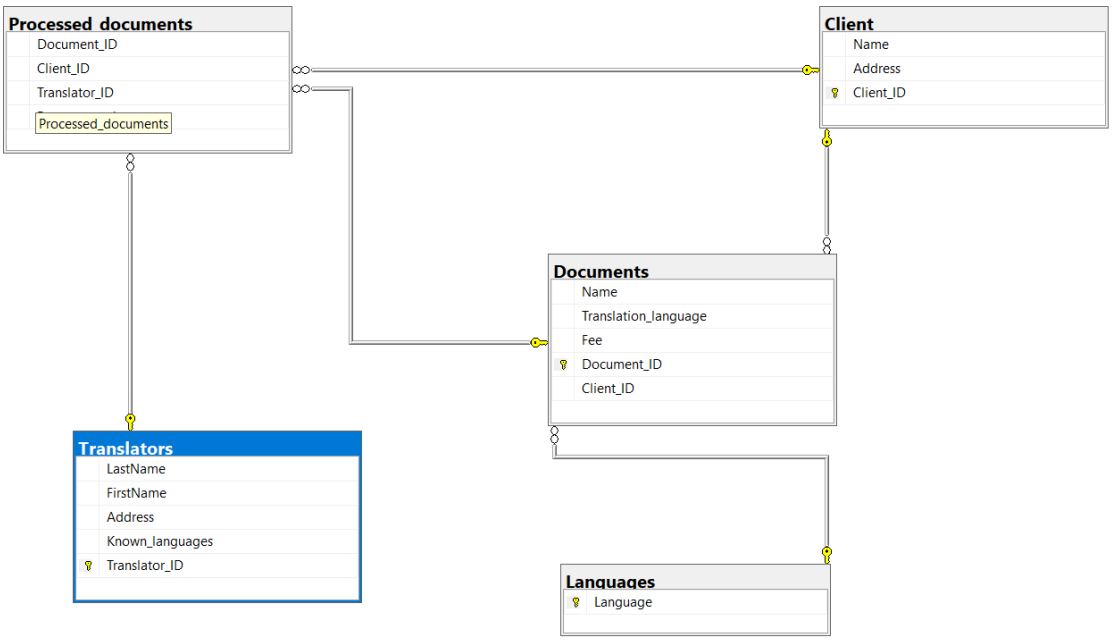


Notes:

- **We introduce CHECK (Percentage_Share >= 0 AND Percentage_Share <= 100) to verify if it is a number between 0-100**
- **A document can be translated if the selected language is included in the languages table.**

4 Database realization

4.1 ER Diagram in MS SQL Server



4.2 Code creating database

```
-- Check if the database "Project" exists. If not, create it.
IF OBJECT_ID('Project') IS NULL
BEGIN
    CREATE DATABASE Project
END
GO

-- Use the "Project" database

USE [Project]
GO

/****** Object:  Table [dbo].[Client]    Script Date: 23.01.2023 18:55:40 ******/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Client](
    [Name] [nchar](50) NOT NULL,
    [Address] [nchar](50) NOT NULL,
    [Client_ID] [int] NOT NULL identity(1, 1) UNIQUE,
    primary key (Client_ID)
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Languages](
    [Language] [nchar](30) NOT NULL UNIQUE,
    primary key (Language)
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Translators](
    [LastName] [nchar](40) NOT NULL,
    [FirstName] [nchar](30) NOT NULL,
```

```
    [Address] [nchar](50) NOT NULL,
    [Known_languages] [nchar](40) NOT NULL,
    [Translator_ID] [int] NOT NULL identity(1, 1) UNIQUE,
    primary key (Translator_ID)
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Documents](
    [Name] [nchar](50) NOT NULL,
    [Translation_language] [nchar](30) NOT NULL,
    [Fee] [money] NOT NULL,
    [Document_ID] [int] NOT NULL identity(1, 1) UNIQUE,
    [Client_ID] [int] NOT NULL,
    primary key (Document_ID),
    foreign key (Client_ID) references Client (Client_ID) on delete no action on
update cascade,
    foreign key (Translation_language) references Languages (Language) on delete no
action
) ON [PRIMARY]
GO

CREATE TABLE [dbo].[Processed_documents](
    [Document_ID] [int] NOT NULL,
    [Client_ID] [int] NOT NULL,
    [Translator_ID] [int] NOT NULL,
    [Percentage_share] [int] NOT NULL CHECK (Percentage_share >= 0 AND
Percentage_share <=100),
    FOREIGN KEY (Document_ID) REFERENCES Documents (Document_ID) ON DELETE NO ACTION
ON UPDATE CASCADE,
    FOREIGN KEY (Translator_ID) REFERENCES Translators (Translator_ID) ON DELETE NO
ACTION,
    FOREIGN KEY (Client_ID) REFERENCES Client (Client_ID) ON DELETE NO ACTION
) ON [PRIMARY]
GO
```

4.3 Code filling the database with test data

```
USE [Project]
GO

-- Inserting data into the Client table
INSERT INTO [dbo].[Client] ([Name], [Address])
VALUES
    ('Orange Solutions', 'Kraków ul. Wesoła 25'),
    ('Americk Team', 'Warszawa ul. Mazowiecka 21'),
    ('Jorick Musisians', 'Poznań ul. Habela 5');
GO

-- Inserting data into the Languages table
INSERT INTO [dbo].[Languages] ([Language])
VALUES
    ('English'),
    ('German'),
    ('Ukrainian');
GO

-- Inserting data into the Translators table
INSERT INTO [dbo].[Translators] ([LastName], [FirstName], [Address],
[Known_languages])
VALUES
    ('Kowalski', 'Jan', 'Kraków ul. Kruka 5', 'English, German, Polish'),
    ('Janicki', 'Kacper', 'Zelczyna ul. Śliczna 12', 'English, Polish'),
```

```
    ('Mrancski', 'Marcin', 'Kraków ul. Daleka 10', 'Ukrainian, Polish');
GO

-- Inserting data into the Documents table
INSERT INTO [dbo].[Documents] ([Name], [Translation_language], [Fee], [Client_ID])
VALUES
    ('List of invoices', 'English', '60', '1'),
    ('List of payments', 'English', '85', '1'),
    ('List of parts of device F-12', 'English', '90', '2'),
    ('Rules of the concert in Poznań', 'German', '110', '3');
GO

-- Inserting data into the Processed_documents table
INSERT INTO [dbo].[Processed_documents] ([Document_ID], [Client_ID],
[Translator_ID], [Percentage_share])
VALUES
    (1, 1, 1, 34),
    (1, 1, 2, 66),
    (2, 2, 2, 100),
    (3, 3, 1, 100);
GO
```

5 Example queries for SQL database

5.1 Query – Translators who know English

Translators who know English

```
SELECT * FROM Translators WHERE Known_languages LIKE '%English%' ORDER BY Lastname;
```

	LastName	FirstName	Address	Known_languages	Translator_ID
1	Janicki	Kacper	Zelczyna ul. Sliczna 12	English, Polish	2
2	Kowalski	Jan	Kraków ul. Kruka 5	English, German, Polish	1

5.2 Query – Documents from the client

All documents commissioned by a given client

```
SELECT D.Name, D.Translation_language, D.Fee, D.Document_ID, D.Client_ID
FROM Documents AS D
JOIN Client AS C ON D.Client_ID = C.Client_ID
WHERE C.Name LIKE '%Americk Team%';
```

	Name	Translation_language	Fee	Document_ID	Client_ID
1	List of parts of device F-12	English	90,00	3	2

5.3 Query – Documents translated by a given translator

All documents translated by a given translator

```
SELECT R.Document_ID, R.Client_ID, R.Translator_ID, R.Percentage_share
FROM Processed_documents AS R
JOIN Translators AS T ON R.Translator_ID = T.Translator_ID
WHERE T.Lastname LIKE '%Kowalski%';
```

	Document_ID	Client_ID	Translator_ID	Percentage_share
1	1	1	1	34
2	3	3	1	100

5.4 Query – Sum of fees from documents of a given client

Sum of all fees collected from documents from a given client

```
SELECT SUM(Documents.Fee) as Total_fee
FROM Documents
JOIN Client as C ON Documents.Client_ID = C.Client_ID
WHERE C.Client_ID = 1;
```

	Total_fee
1	145,00

5.5 Query – All documents of a given language

All documents translated into a given language

```
SELECT * FROM Documents WHERE Translation_language LIKE '%German%';
```

	Name	Translation_language	Fee	Document_ID	Client_ID
1	Rules of the concert in Poznan	German	110,00	4	3

6 Attachments

- **Project documentation (file: documentation.pdf)**
- **SQL Database code with example values (file: insert_values.sql)**
- **SQL Database code that creates databases (file: create_databases.sql)**
- **SQL Database code with example queries (file: select_queries.sql)**
- **SQL Database code for dropping tables to clear all inserted values (file: drop_tables.sql)**