APPLICATION TECHNICAL SCOPE

DATA MODEL

For this application to work was necessary to use two custom objects, a new created one called "Translation", and the "Translation Request" from the Unbabel Connector managed package.

The Translation object was created to storage all translations made by users in the "Unbabel Translator" page, its sharing model is private so that users can only see their own translations.

Object Singular Label	**Translation**	
Object Plural Label	**Translations**	
Object Name	**Translation**	
Object API Name	**Translation__c**	
Field Label	**API Name**	**Data Type**
ID	Name	Auto Number
External ID	ExternalId__c	Auto Number (External ID)
Message	Message__c	Long Text Area(1000)
Status	Status__c	Picklist
Text	Text__c	Text(255)
Translate from	TranslateFrom__c	Picklist
Translate to	TranslateTo__c	Picklist
Translation	Translation__c	Text(255)

TECHNICAL SCOPE

The Unbabel Translator application can be found as a tab of the "Translations" app, this tab is a lightning app page used to host the translator lightning component called "textTranslation".

The page is composed by two sections, the section for the user to request translations and the section to show the translations already made. Every time a new translation is requested, the Translation record will be shown in the below section.

Every time a translation is requested in the page, besides creating a Translation record, the Unbabel Connector is used in the back end to send the request to the Unbabel API, for this request is used the "Translation Request" custom object to set the language and some other parameters of the API request. Once Salesforce receives the response from the Unbabel Connector, the apex class called "GenericTranslationHandler" will process the information returned, and save it to the Translation record created by the user in the Unbabel Translator page.

Meanwhile the Unbabel Translator page will keep its data refresh in real time, using javascript to update the translations table every 15 seconds, so when the API response is processed by Salesforce and the Translation record is updated, the table will also update its informations.

USE CASES

For this application, it's possible to identify two use cases:

Name of Use Case	**Translate any text from english**
Description	The user inputs some word or sentence in the text box and then submits for translation, the translation is made from english to the language chosen by the user.
Actors	System administrator and any other user with the required profile permissions.
Preconditions	1. Access to the Translations app. 2. At least read and create permissions to the Translation custom object.
Postconditions	Once submited for translation, the page should be refreshed and the new translation be shown in the below section, still without the translation result.
Flow	1. Access the Translations app in Salesforce; 2. Go to the Unbabel Translator tab; 3. Input any text with a maximum of 255 characters; 4. Choose the language to which the text will be translated; 5. Submit for translation.
Exceptions	1. If a text longer than 255 characters is entered, the user should get an error. 2. If any problems occur while sending the translation request to the Unbabel API, an error message is thrown to the user and the Translation record is not created.

Name of Use Case	Consult translations
Description	The user access the page to consult translations in real time.
Actors	System administrator and any other user with the required profile permissions.
Preconditions	Access to the Translations app.
Postconditions	The translations already requested should be shown in the data table, and if any new translation to which the user has access is created, the table is updated automatically.
Flow	1. Access the Translations app in Salesforce; 2. Go to the Unbabel Translator tab; 3. Check on the translations data table.
Exceptions	N/A