# **Angular 9 Tutorial on Internationalization (i18n)**

When it comes to the global consumption of modern, multilingual web experiences, Angular has been on quite a journey. We have previously walked you through the [best Angular libraries for internationalization](https://phrase.com/blog/posts/best-libraries-for-angular-i18n/) and explored the use of [ngx-translate for localization in Angular](https://phrase.com/blog/posts/angular-localization-i18n/). We have also had a look at the [built-in Angular i18n module](https://phrase.com/blog/posts/angular-translation-in-angular-8/) for the previous versions of the TypeScript-based open-source web app framework.

Building on that knowledge, this Angular 9 tutorial will walk you through the process of [Angular i18n](https://angular.io/guide/i18n) by using their very much improved built-in I18n module. For the purposes of this tutorial, we will create a sample feedback form for Phrase and serve it in three different languages – English, French, and German.

The working demo is accessible via [Google Firebase](https://angular-localization.firebaseapp.com/en/) to help you get a better grasp of how i18n works in the production environment. We will use the @angular/localize package available as part of the [Angular 9 release](https://blog.angular.io/version-9-1-of-angular-now-available-typescript-3-8-faster-builds-and-more-eb292f989428). To get the source code for the demo app, make sure you stop by at [GitHub](https://github.com/PhraseApp-Blog/phrase-angular-9-i18n-tutorial).

Contents

* [Getting Ready for Angular i18n](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Getting_Ready_for_Angular_i18n)
* [Creating an Angular Application](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Creating_an_Angular_Application)
* [Setting Up Localization Using the Angular CLI](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Setting_Up_Localization_Using_the_Angular_CLI)
* [Understanding the i18n Attribute](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Understanding_the_i18n_Attribute)
  + [Custom id](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Custom_id)
  + [Translate HTML attributes](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Translate_HTML_attributes)
* [Install Angular Material](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Install_Angular_Material)
* [Update the AppModule](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Update_the_AppModule)
* [Create the nav-bar Component](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Create_the_nav-bar_Component)
* [Create a Model](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Create_a_Model)
* [Add Forms Module](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Add_Forms_Module)
* [Create the Feedback Component](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Create_the_Feedback_Component)
* [Creating a Translation Source File](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Creating_a_Translation_Source_File)
* [Translating the Content](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Translating_the_Content)
* [Configure the App to Serve in Multiple Languages](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Configure_the_App_to_Serve_in_Multiple_Languages)
* [Set up the app component](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Set_up_the_app_component)
* [Executing the App in the Local Environment](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Executing_the_App_in_the_Local_Environment)
* [Create the Production Build](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Create_the_Production_Build)
* [Deploy the App to Firebase](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Deploy_the_App_to_Firebase)
* [Wrapping Up the Angular 9 Tutorial on i18n](https://phrase.com/blog/posts/angular-9-tutorial-on-i18n/#Wrapping_Up_the_Angular_9_Tutorial_on_i18n)

Getting Ready for Angular i18n

Make sure that you have an Angular development environment set up on your machine. Should this not be the case, make sure you install the following software before proceeding:

* [LTS version of Node.js](https://nodejs.org/en/download/) as per your operating system
* [Angular CLI](https://cli.angular.io/)
* [Visual Studio Code](https://code.visualstudio.com/)

Creating an Angular Application

Navigate to the directory where you want to create the new project. Open the command prompt, and run the command shown below to create a new Angular app named as angular-localization.

ng **new** angular-localization --routing=**false** --style=css

Run the following set of commands:

cd angular-localization

code .

These will change the directory to the root project folder, i.e. angular-localization, and open the project in VS Code.

I18n Manager for Global Success

Learn how to find the best i18n manager and follow our best practices for making your business a global success.

[Check out the guide](https://phrase.com/lp/i18n-manager/?utm_campaign=i18n-manager&utm_medium=blog&utm_source=linkbox)

Setting Up Localization Using the Angular CLI

Run the following command in the root directory of the project:

ng add @angular/localize

This command will install the package in your angular app. It will also update the polyfills.ts file which allows the project to take advantage of Angular’s localization features.

Understanding the i18n Attribute

The Angular framework provides us with the i18n attribute, which can be used to mark an HTML element for translation:

<**h1** i18n>Welcome to Phrase</**h1**>

Custom id

When we create the translation file, each element marked with the i18n attribute will be created as a separate translation unit. Each translation unit will have a unique id associated with it.

<**trans-unit** id="d8f6108d0a0dad756b4a38c17436b9dfda296c2d" datatype="html">

<**source**>Welcome to Phrase</**source**>

</**trans-unit**>

If you change the text for any tag in your HTML file, you need to regenerate the translation file. Regenerating the file will override the default id of <trans-unit> tags. Hence, it is advisable to provide custom ids to each translatable tag to maintain consistency.

You can provide the custom id in the i18n attribute by using the prefix @@:

<**h1** i18n="@@pageHeader">Welcome to Phrase</**h1**>

If you are using custom ids, then make sure to use unique ids. If the custom id is duplicated, then only the first tag will be extracted, and its translation will be used in place of all the subsequent occurrences of that id.

Translate HTML attributes

If you want to translate an attribute of an HTML tag, then you can use the i18n attribute as i18n-x, where x is the name of the HTML attribute you wish to translate:

<**input** i18n-placeholder placeholder="Name" name="name">

Here, we have marked the placeholder attribute for translation by adding the i18n-placeholder attribute to the <input> tag.

Let us proceed with the app creation to get a better understanding of these concepts.

Install Angular Material

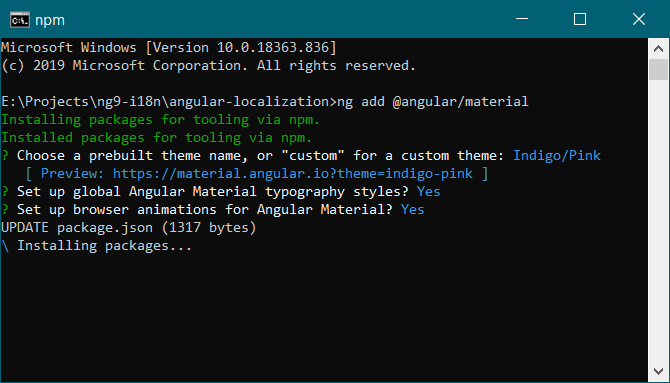
We will use the Angular Material component library for designing our app. Run the following command to install the library:

ng add @angular/material

As soon as you execute the command, it will ask you the following three questions:

1. Choose a prebuilt theme name, or “custom” for a custom theme: You can choose from a set of pre-built themes available or provide your custom theme. We will select the Indigo/Pink theme here,
2. Set up global Angular Material typography styles? (y/N) – y,
3. Set up browser animations for Angular Material? (Y/n) – y.

Refer to the image shown below:



You can learn more about the Angular Material in the [official guide](https://material.angular.io/).

Update the AppModule

We will import all the required modules of Angular material components into the AppModule. Open the src/app/app.module.ts file, add the import statements, and update the imports array as shown below:

**import** { MatButtonModule } from '@angular/material/button';

**import** { MatToolbarModule } from '@angular/material/toolbar';

**import** { MatSelectModule } from '@angular/material/select';

**import** { MatCardModule } from '@angular/material/card';

**import** { MatInputModule } from '@angular/material/input';

**import** { MatRadioModule } from '@angular/material/radio';

**import** { MatMenuModule } from '@angular/material/menu';

**import** { MatIconModule } from '@angular/material/icon';

@NgModule({

...

imports: [

...

MatCardModule,

MatButtonModule,

MatToolbarModule,

MatSelectModule,

MatInputModule,

MatRadioModule,

MatMenuModule,

MatIconModule

],

...

})

Create the nav-bar Component

Run the following command to create the nav-bar component for our application.

ng g c nav-bar

Open src\app\nav-bar\nav-bar.component.html and replace what you see there with the following code.

<**mat-toolbar** color="primary" class="nav-bar mat-elevation-z2">

<**mat-label**>Phrase</**mat-label**>

<**span** class="spacer"></**span**>

<**ng-container**>

<**mat-icon**>language</**mat-icon**>

<**button** mat-button [matMenuTriggerFor]="menu">

{{siteLanguage}}

<**mat-icon**>arrow\_drop\_down</**mat-icon**>

</**button**>

<**mat-menu** #menu="matMenu">

<**a** mat-menu-item href="/en">English</**a**>

<**a** mat-menu-item href="/fr">Français</**a**>

<**a** mat-menu-item href="/de">Deutsch</**a**>

</**mat-menu**>

</**ng-container**>

</**mat-toolbar**>

We have created a mat-toolbar, which will act as the nav-bar for our app. The nav-bar contains a menu having three options to set the language of our app. When we click on the menu item, it will append the language locale to the base URL. This will allow us to serve the app in different languages at runtime. We will also display the siteLanguage in the nav-bar and it will update as we change the language of our app.

Open src\app\nav-bar\nav-bar.component.ts and add the following code inside the NavBarComponent class.

siteLanguage: **string** = 'English';

siteLocale: **string**;

languageList = [

{ code: 'en', label: 'English' },

{ code: 'fr', label: 'Français' },

{ code: 'de', label: 'Deutsch' }

];

**constructor**() { }

ngOnInit() {

**this**.siteLocale = **window**.location.pathname.split('/')[1];

**this**.siteLanguage = **this**.languageList.find(f => f.code === **this**.siteLocale).label;

}

Here, we have defined a list of languages and their standard [locale codes](https://help.phrase.com/help/language-codes). Inside the ngOnInit lifecycle hook, we will read the siteLocale value from the URL. We will then set the siteLanguage by fetching the language name from the list of languages corresponding to the siteLocale.

Add the following style to src\app\nav-bar\nav-bar.component.css file

**.nav-bar** {

position: fixed;

top: 0;

z-index: 99;

}

**.spacer** {

flex: 1 1 auto;

}

Create a Model

We will use the template-driven form for creating the feedback form for our application. Therefore, we need to create a model. Set up a new folder called models inside the src/app folder. Create a file called feedback.ts inside the models folder and insert the following code:

**export** **class** Feedback {

name: **string**;

gender: **string**;

rating: **string**;

comment: **string**;

}

If you are not familiar with the Angular forms, then please refer to the [Angular Forms guide.](https://angular.io/guide/forms-overview)

Add Forms Module

We will import the FormsModule in src/app/app.module.ts as shown below. This will allow us to use template-driven forms in our application.

**import** { FormsModule } from '@angular/forms';

@NgModule({

...

imports: [

...

FormsModule,

],

})

Create the Feedback Component

Run the following command to create the feedback component.

ng g c feedback

Open src\app\feedback\feedback.component.ts and put the following code inside the FeedbackComponent class:

customerFeedback = new Feedback();

**constructor**() { }

ngOnInit(): **void** {

}

saveFeedback() {

alert('Thanks for your valuable feedback!!!**\n**The feedback has been submitted succesfully.');

**console**.table(**this**.customerFeedback);

}

We will create an object of type Feedback class which will bind to the form. The saveFeedback method will be invoked upon the successful submission of the form. It will show a success alert message and log the form output in the browser console.

Open src\app\feedback\feedback.component.html and insert the following code:

<**div** class="feedback-form-container">

<**mat-card** class="feedback-form mat-elevation-z2">

<**mat-card-title** i18n>

Phrase Service Feedback

</**mat-card-title**>

<**mat-card-content**>

<**form** #feedbackForm="ngForm" (ngSubmit)="feedbackForm.form.valid && saveFeedback()" novalidate>

<**mat-form-field** class="full-width">

<**input** i18n-placeholder matInput placeholder="Name" name="name" [(ngModel)]="customerFeedback.name"

#name="ngModel" required>

<**mat-error** i18n \*ngIf="feedbackForm.submitted && name.errors?.required">Name is required</**mat-error**>

</**mat-form-field**>

<**mat-form-field** class="full-width">

<**mat-select** i18n i18n-placeholder placeholder="Gender" name="gender"

[(ngModel)]="customerFeedback.gender" #gender="ngModel" required>

<**mat-option** value="Male">Male</**mat-option**>

<**mat-option** value="Female">Female</**mat-option**>

<**mat-option** value="Others">Don't Want to disclose</**mat-option**>

</**mat-select**>

<**mat-error** i18n \*ngIf="feedbackForm.submitted && gender.errors?.required">Gender is required

</**mat-error**>

</**mat-form-field**>

<**mat-form-field** class="full-width">

<**textarea** i18n-placeholder matInput placeholder="Comment" name="comment"

[(ngModel)]="customerFeedback.comment" #comment="ngModel" required></**textarea**>

<**mat-error** i18n \*ngIf="feedbackForm.submitted && comment.errors?.required">Comment is required

</**mat-error**>

</**mat-form-field**>

<**label** i18n class="example-margin">Rate our customer service: </**label**>

<**mat-radio-group** i18n class="example-radio-group" name="rating" [(ngModel)]="customerFeedback.rating"

#rating="ngModel" required>

<**mat-radio-button** class="example-radio-button" value="Excellent"> Excellent </**mat-radio-button**>

<**mat-radio-button** class="example-radio-button" value="Good"> Good </**mat-radio-button**>

<**mat-radio-button** class="example-radio-button" value="Bad"> Bad </**mat-radio-button**>

</**mat-radio-group**>

<**mat-error** i18n \*ngIf="feedbackForm.submitted && rating.errors?.required">Rating is required</**mat-error**>

<**mat-card-actions** align="right">

<**button** i18n type="submit" mat-raised-button color="primary">Submit</**button**>

</**mat-card-actions**>

</**form**>

</**mat-card-content**>

</**mat-card**>

</**div**>

We now have a template-driven form created. The form will have the following fields:

* Name (an input field used to store the name of the user),
* Gender (a select field displaying three different options to choose from),
* Comment (a text area field used to store comments provided by the user),
* Rating (a radio group field that asks the user to rate the service of Phrase).

We have made all the form fields required fields. Together with the validation error fields, we have also [marked them with the i18n attribute](https://angular.io/guide/i18n#mark-text-with-the-i18n-attribute).

Open src\app\feedback\feedback.component.css and insert the following style:

**.feedback-form-container** {

margin-top: 50px;

display: flex;

justify-content: center;

}

**.feedback-form** {

width: 40%;

}

**.full-width** {

width: 100%;

}

**.mat-radio-button** {

margin: 10px;

}

Creating a Translation Source File

Run the following command in the CLI to create a translation source file:

ng xi18n --output-path src/translate

It will create a folder called translate inside the src folder and create a messages.xlf file inside it. Open the file and you can observe the following XML code inside it.

**<?xml version="1.0" encoding="UTF-8" ?>**

<**xliff** version="1.2" xmlns="urn:oasis:names:tc:xliff:document:1.2">

<**file** source-language="en-US" datatype="plaintext" original="ng2.template">

<**body**>

<**trans-unit** id="d8f6108d0a0dad756b4a38c17436b9dfda296c2d" datatype="html">

<**source**>

Phrase Service Feedback

</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">3</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cff1428d10d59d14e45edec3c735a27b5482db59" datatype="html">

<**source**>Name</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">9</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5b752f143ab30faa45a2679b5e7a28bbf3fb246d" datatype="html">

<**source**>Name is required</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">11</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="b764af9a75dd6519882ca31bdaa1332f36e5d3a5" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Male<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Female<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Don&apos;t Want to disclose<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">15</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="003d4f5e4909ccf01b6a204c688fd6dab15c20bd" datatype="html">

<**source**>Gender</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">14</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cdfeb9057d2051759a9bd5e040b2604b897eac40" datatype="html">

<**source**>Gender is required</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">20</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5a5d7ee2acbfa9c91ab7f41d26bda9ff0cafe42f" datatype="html">

<**source**>Comment</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">24</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="ebee5d3534087e8de4eedc2b02fcde41a9e6f94f" datatype="html">

<**source**>Comment is required</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">26</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="9a689f29ccb25c2e6e861ae4de5dff9c7e723237" datatype="html">

<**source**>Rate our customer service: </**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">29</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="c92fe165f8c6a8b29a32c3ea7a0f732f73b21570" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Excellent <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Good <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Bad <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">31</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="8488059f99539c207978ae09a0313aeeea6df161" datatype="html">

<**source**>Rating is required</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">36</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="71c77bb8cecdf11ec3eead24dd1ba506573fa9cd" datatype="html">

<**source**>Submit</**source**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">38</**context**>

</**context-group**>

</**trans-unit**>

</**body**>

</**file**>

</**xliff**>

This file contains a list of <trans-unit> tags. They will have all the content that was marked for translation using the i18n attribute. You can also observe that each <trans-unit> tag has an id property associated with it. This unique id will be generated by default for each tag that was marked with the i18n attribute.

Translating the Content

We will launch our app in three different languages – English, German, and French. Therefore, we will create three copies of the messages.xlf file and rename them to messages.en.xlf, messages.fr.xlf, and messages.de.xlf. These file names can be customized as per your choice, but the extension should be .xlf.

Open messages.de.xlf and put in the following content in it.

**<?xml version="1.0" encoding="UTF-8" ?>**

<**xliff** version="1.2" xmlns="urn:oasis:names:tc:xliff:document:1.2">

<**file** source-language="en-US" datatype="plaintext" original="ng2.template">

<**body**>

<**trans-unit** id="d8f6108d0a0dad756b4a38c17436b9dfda296c2d" datatype="html">

<**source**>Phrase Service Feedback</**source**>

<**target**>Phrase Service Feedback</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">3</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cff1428d10d59d14e45edec3c735a27b5482db59" datatype="html">

<**source**>Name</**source**>

<**target**>Name</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">9</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5b752f143ab30faa45a2679b5e7a28bbf3fb246d" datatype="html">

<**source**>Name is required</**source**>

<**target**>Name ist erforderlich</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">11</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="b764af9a75dd6519882ca31bdaa1332f36e5d3a5" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Male<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Female<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Don&apos;t Want to disclose<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**source**>

<**target**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Männlich<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Weiblich<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Ich möchte nicht offenlegen<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">15</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="003d4f5e4909ccf01b6a204c688fd6dab15c20bd" datatype="html">

<**source**>Gender</**source**>

<**target**>Geschlecht</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">14</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cdfeb9057d2051759a9bd5e040b2604b897eac40" datatype="html">

<**source**>Gender is required</**source**>

<**target**>Geschlecht ist erforderlich</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">20</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5a5d7ee2acbfa9c91ab7f41d26bda9ff0cafe42f" datatype="html">

<**source**>Comment</**source**>

<**target**>Kommentar</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">24</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="ebee5d3534087e8de4eedc2b02fcde41a9e6f94f" datatype="html">

<**source**>Comment is required</**source**>

<**target**>Kommentar ist erforderlich</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">26</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="9a689f29ccb25c2e6e861ae4de5dff9c7e723237" datatype="html">

<**source**>Rate our customer service: </**source**>

<**target**>Bewerten Sie unseren Kundenservice: </**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">29</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="c92fe165f8c6a8b29a32c3ea7a0f732f73b21570" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Excellent <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Good <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Bad <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**source**>

<**target**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Ausgezeichnet <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Gut <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Schlecht <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">31</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="8488059f99539c207978ae09a0313aeeea6df161" datatype="html">

<**source**>Rating is required</**source**>

<**target**>Bewertung ist erforderlich</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">36</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="71c77bb8cecdf11ec3eead24dd1ba506573fa9cd" datatype="html">

<**source**>Submit</**source**>

<**target**>Einreichen</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">38</**context**>

</**context-group**>

</**trans-unit**>

</**body**>

</**file**>

</**xliff**>

This file has the same content as that of the messages.xlf file. However, we have added a <target> tag corresponding to each <source> tag. The <target> tag contains the translated text for the content inside the <source> tag. The <source> tag has the original content in English whereas the <target> tag contains the translated text in German. Here I have used Google translate for the translation. However, in a real-world application, a language expert should translate the contents from the messages.xlf file.

Similarly, open the messages.fr.xlf and put in the following content in it.

**<?xml version="1.0" encoding="UTF-8" ?>**

<**xliff** version="1.2" xmlns="urn:oasis:names:tc:xliff:document:1.2">

<**file** source-language="en-US" datatype="plaintext" original="ng2.template">

<**body**>

<**trans-unit** id="d8f6108d0a0dad756b4a38c17436b9dfda296c2d" datatype="html">

<**source**>Phrase Service Feedback</**source**>

<**target**>Commentaires sur le service de phrase</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">3</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cff1428d10d59d14e45edec3c735a27b5482db59" datatype="html">

<**source**>Name</**source**>

<**target**>Nom</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">9</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5b752f143ab30faa45a2679b5e7a28bbf3fb246d" datatype="html">

<**source**>Name is required</**source**>

<**target**>Le nom est requis</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">11</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="b764af9a75dd6519882ca31bdaa1332f36e5d3a5" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Male<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Female<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Don&apos;t Want to disclose<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**source**>

<**target**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Mâle<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Femelle<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Je ne veux pas divulguer<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">15</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="003d4f5e4909ccf01b6a204c688fd6dab15c20bd" datatype="html">

<**source**>Gender</**source**>

<**target**>Le genre</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">14</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cdfeb9057d2051759a9bd5e040b2604b897eac40" datatype="html">

<**source**>Gender is required</**source**>

<**target**>Le sexe est requis</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">20</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5a5d7ee2acbfa9c91ab7f41d26bda9ff0cafe42f" datatype="html">

<**source**>Comment</**source**>

<**target**>Commentaire</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">24</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="ebee5d3534087e8de4eedc2b02fcde41a9e6f94f" datatype="html">

<**source**>Comment is required</**source**>

<**target**>Un commentaire est requis</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">26</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="9a689f29ccb25c2e6e861ae4de5dff9c7e723237" datatype="html">

<**source**>Rate our customer service: </**source**>

<**target**>Évaluez notre service client: </**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">29</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="c92fe165f8c6a8b29a32c3ea7a0f732f73b21570" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Excellent <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Good <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Bad <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**source**>

<**target**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Excellent <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Bien <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Mauvais <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">31</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="8488059f99539c207978ae09a0313aeeea6df161" datatype="html">

<**source**>Rating is required</**source**>

<**target**>Une évaluation est requise</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">36</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="71c77bb8cecdf11ec3eead24dd1ba506573fa9cd" datatype="html">

<**source**>Submit</**source**>

<**target**>Soumettre</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">38</**context**>

</**context-group**>

</**trans-unit**>

</**body**>

</**file**>

</**xliff**>

The <source> tag has the original content in English whereas the <target> tag contains the translated text in French.

Open messages.en.xlf and put in the following content in it.

**<?xml version="1.0" encoding="UTF-8" ?>**

<**xliff** version="1.2" xmlns="urn:oasis:names:tc:xliff:document:1.2">

<**file** source-language="en-US" datatype="plaintext" original="ng2.template">

<**body**>

<**trans-unit** id="d8f6108d0a0dad756b4a38c17436b9dfda296c2d" datatype="html">

<**source**>Phrase Service Feedback</**source**>

<**target**>Phrase Service Feedback</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">3</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cff1428d10d59d14e45edec3c735a27b5482db59" datatype="html">

<**source**>Name</**source**>

<**target**>Name</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">9</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5b752f143ab30faa45a2679b5e7a28bbf3fb246d" datatype="html">

<**source**>Name is required</**source**>

<**target**>Name is required</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">11</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="b764af9a75dd6519882ca31bdaa1332f36e5d3a5" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Male<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Female<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Don&apos;t Want to disclose<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**source**>

<**target**>

<**x** id="START\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Male<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_1" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Female<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

<**x** id="START\_TAG\_MAT-OPTION\_2" ctype="x-mat-option" equiv-text="&lt;mat-option&gt;"/>Don&apos;t Want to disclose<**x** id="CLOSE\_TAG\_MAT-OPTION" ctype="x-mat-option" equiv-text="&lt;/mat-option&gt;"/>

</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">15</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="003d4f5e4909ccf01b6a204c688fd6dab15c20bd" datatype="html">

<**source**>Gender</**source**>

<**target**>Gender</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">14</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="cdfeb9057d2051759a9bd5e040b2604b897eac40" datatype="html">

<**source**>Gender is required</**source**>

<**target**>Gender is required</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">20</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="5a5d7ee2acbfa9c91ab7f41d26bda9ff0cafe42f" datatype="html">

<**source**>Comment</**source**>

<**target**>Comment</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">24</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="ebee5d3534087e8de4eedc2b02fcde41a9e6f94f" datatype="html">

<**source**>Comment is required</**source**>

<**target**>Comment is required</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">26</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="9a689f29ccb25c2e6e861ae4de5dff9c7e723237" datatype="html">

<**source**>Rate our customer service: </**source**>

<**target**>Rate our customer service: </**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">29</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="c92fe165f8c6a8b29a32c3ea7a0f732f73b21570" datatype="html">

<**source**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Excellent <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Good <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Bad <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**source**>

<**target**>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Excellent <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_1" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Good <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

<**x** id="START\_TAG\_MAT-RADIO-BUTTON\_2" ctype="x-mat-radio-button" equiv-text="&lt;mat-radio-button&gt;"/> Bad <**x** id="CLOSE\_TAG\_MAT-RADIO-BUTTON" ctype="x-mat-radio-button" equiv-text="&lt;/mat-radio-button&gt;"/>

</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">31</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="8488059f99539c207978ae09a0313aeeea6df161" datatype="html">

<**source**>Rating is required</**source**>

<**target**>Rating is required</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">36</**context**>

</**context-group**>

</**trans-unit**>

<**trans-unit** id="71c77bb8cecdf11ec3eead24dd1ba506573fa9cd" datatype="html">

<**source**>Submit</**source**>

<**target**>Submit</**target**>

<**context-group** purpose="location">

<**context** context-type="sourcefile">src/app/feedback/feedback.component.html</**context**>

<**context** context-type="linenumber">38</**context**>

</**context-group**>

</**trans-unit**>

</**body**>

</**file**>

</**xliff**>

In this case, both the <source> tag and the <target> tag contains text in English language.

Your Guide to Continuous Localization

Find out how continuous localization saves your agile development to grow your business on a global scale.

[Check out the guide](https://phrase.com/lp/continuous-localization/?utm_campaign=continuous-localization&utm_medium=blog&utm_source=linkbox)

Configure the App to Serve in Multiple Languages

We need to define the supported locales, as well as the path of the corresponding translated files for our app. This will help the compiler to understand how to serve the translated content for different languages.

Open the angular.json file and add the following configuration.

**"projects":** {

...

**"angular-localization":** {

...

**"i18n":** {

**"locales":** {

**"en":** "src/translate/messages.en.xlf",

**"fr":** "src/translate/messages.fr.xlf",

**"de":** "src/translate/messages.de.xlf"

}

}

...

}

}

The i18n option is used to define the locales for the app. It has two sub-options as described below:

* sourceLocale: It defines the locale used within the source code of the app. This field is optional and the default value is set to “en-US”.
* locales: It defines a key-value pair. The key denotes the locale in which you want to serve the app. The value represents the location of the translation file to serve the content.

We will set the localize build configuration option. This will instruct the AOT compiler to use the translation configuration.

Add the following lines in the angular.json file.

**"build":** {

...

**"configurations":** {

...

**"en":** {

**"localize":** ["en"]

},

**"fr":** {

**"localize":** ["fr"]

},

**"de":** {

**"localize":** ["de"]

}

}

},

Here, we have defined the localize configuration for the three languages we want to serve our app in. The localize options allow the Angular CLI to place the app build output in a locale-specific folder. These folders will be created in the path defined in the outputPath property. Therefore, in this case, the locale-specific folders will be created inside the dist\angular-localization folder. We will see this in action in the latter section when we will create the production build of our app.

Finally, we will add the custom locale-specific configurations. This will allow us to apply specific build options to a particular locale.

Add the following lines in the angular.json file.

**"serve":** {

...

**"configurations":** {

**"production":** {

**"browserTarget":** "angular-localization:build:production"

},

**"en":** {

**"browserTarget":** "angular-localization:build:en"

},

**"fr":** {

**"browserTarget":** "angular-localization:build:fr"

},

**"de":** {

**"browserTarget":** "angular-localization:build:de"

}

}

}

To get an even better understanding, please have a look at the complete [angular.json file on GitHub](https://github.com/PhraseApp-Blog/Angular9i18nTutorial/blob/master/angular.json" \t "_blank).

Set up the app component

Open app.component.html file. Replace the already existing text with the following code:

<**app-nav-bar**></**app-nav-bar**>

<**div** class="container">

<**app-feedback**></**app-feedback**>

</**div**>

Open styles.css file and add the following styles definitions to it:

**.container** {

padding-top: 60px;

}

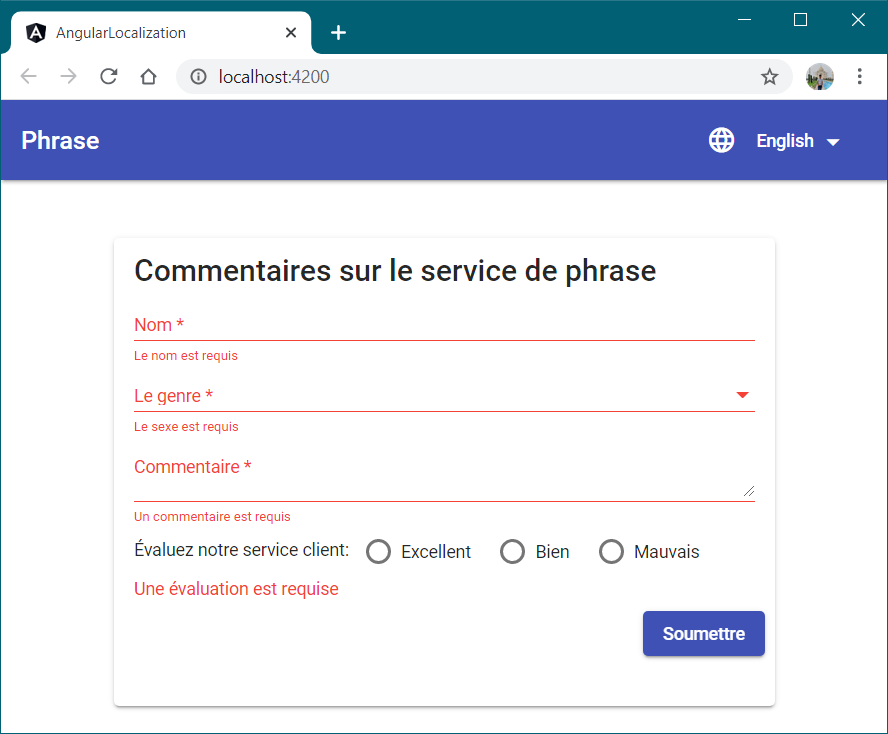
Executing the App in the Local Environment

The time has finally come to see our Angular 9 tutorial demo app in action. Run the following command to execute the app in local environment:

ng serve --configuration=fr

This will launch the application in the “fr” configuration and our app will be displayed in French.

Refer to the output screen as shown below:



Note: We can use only one locale at a time while executing the app in the local environment.

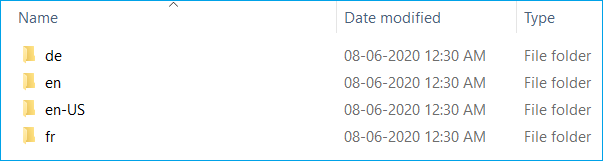
Create the Production Build

Run the following command to create the production build of the app:

ng build --prod --localize

Upon successful execution, this command will create a ‘dist’ folder in the application’s root folder. Inside the ‘dist’ folder we will have another folder named ‘angular-localization’. The ‘angular-localization’ folder contains four sub-folders to serve our application in different languages.

Refer to the image shown below:



The three sub-folders – en, fr, and de are for the language configuration which we have defined for our app i.e. English, French, and German respectively. The “en-US” folder is for the default locale used by the Angular app.

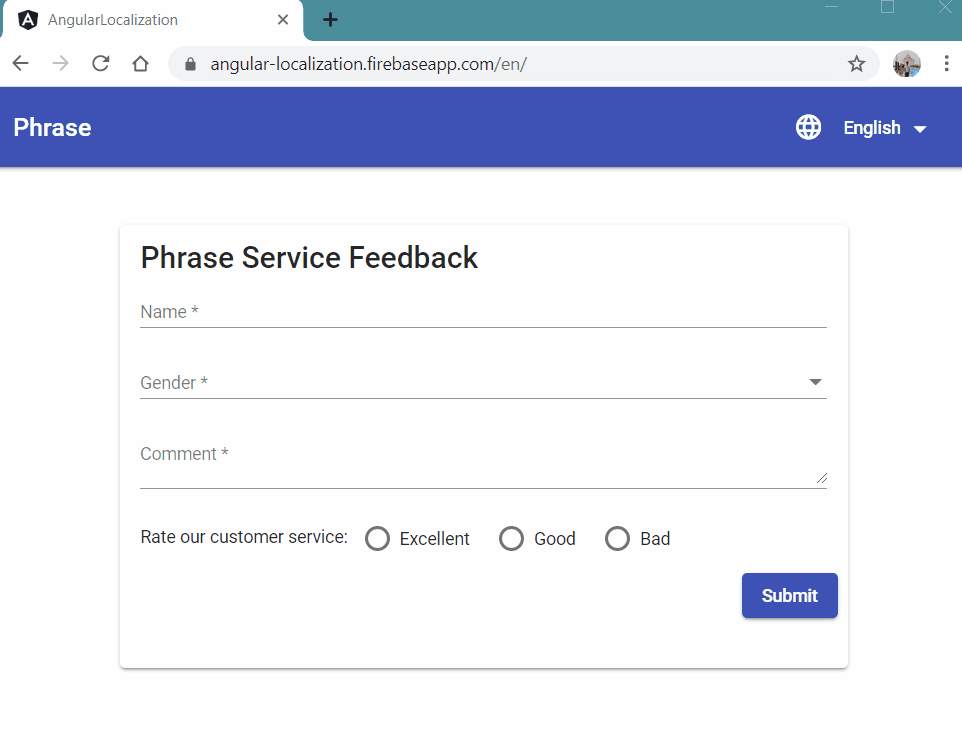
When we deploy the app to a prod environment, the app will be served from these different folders based on the locale value set in the URL parameter. The URL parameter can be changed by setting the site language from the drop-down menu in the nav-bar.

Deploy the App to Firebase

We will deploy this demo app on Firebase to see how i18n works in the production environment.

Check out all the [steps to deploying to Firebase.](https://phrase.com/blog/posts/angular-translation-in-angular-8/#Deploying_to_Firebase) follow the direction to deploy this Angular app on Firebase.

Open the URL and append the language locale to it. Hence, the URL will be yoursite.com/fr/ for the French language, and so on. The application, which we built here, is hosted at [angular-localization.firebaseapp.com/en/](https://angular-localization.firebaseapp.com/en/). If you open this URL, you will see the output as shown below:



Wrapping Up the Angular 9 Tutorial on i18n

In this end-to-end Angular 9 tutorial focused on the new and quite improved built-in i18n module in Angular, we learned how to internationalize an Angular app and make it available in multiple languages by using i18n features. We created a sample Phrase feedback form and deployed it to Firebase to see its core functions. The app is served in three different languages – English, French, and German. We used the latest @angular/localize package to set up the localization for our app.