



# Adding a New Menu Item

Let's begin from UI and create a new page named "**Phone book**".

## Defining a Menu Item

Open `src\app\shared\layout\nav\app-navigation.service.ts` in the client side (**Acme.PhoneBookDemo.AngularUI**) which defines menu items in the application. Create new menu item as shown below (You can add it right after the dashboard menu item).

```
new AppMenuItem("PhoneBook", null, "flaticon-book", "/app/main/phonebook")
```

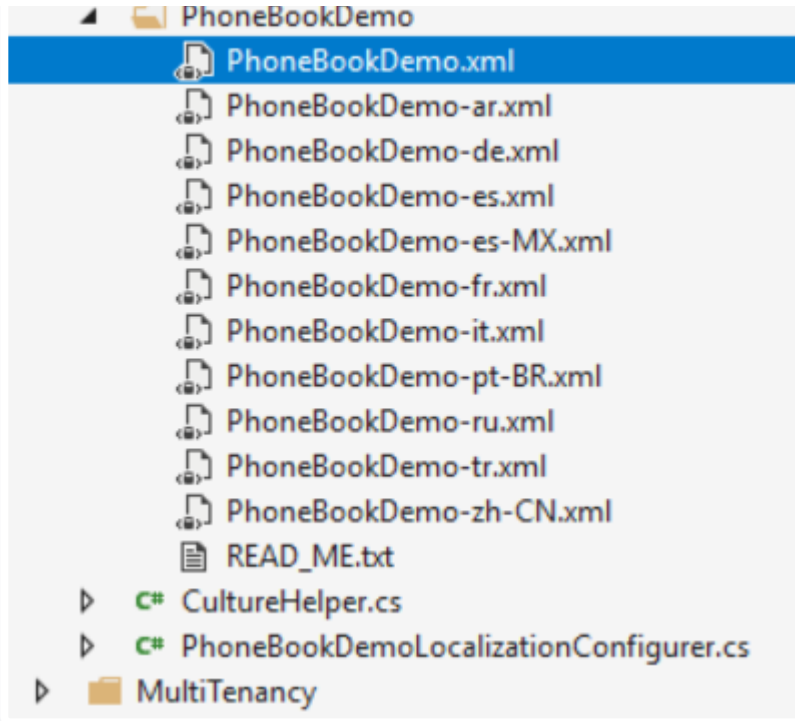
**PhoneBook** is the menu name (will localize below), **null** is for permission name (will set it later), **flaticon-book** is just an arbitrary icon class (from [this set](#)) and **/phonebook** is the Angular route.

If you run the application, you can see a new menu item on the left menu, but it won't work (it redirects to default route) if you click to the menu item, since we haven't defined the Angular route yet.

## Localize Menu Item Display Name

Localization strings are defined in **XML** files in **.Core** project in server side as shown below:





Open PhoneBookDemo.xml (the **default, English** localization dictionary) and add the following line:

```
<text name="PhoneBook">Phone Book</text>
```

If we don't define "PhoneBook"s value for other localization dictionaries, default value is shown in all languages. For example, we can define it also for Turkish in `PhoneBookDmo-tr.xml` file:

```
<text name="PhoneBook">Telefon Rehberi</text>
```

Note: Any change in server side (including change localization texts) requires recycle of the server application. We suggest to use Ctrl+F5 if you don't need to debugging for a faster startup. In that case, it's enough to make a re-build to recycle the application.

## Angular Route

Angular has a powerful URL routing system. ASP.NET Zero has defined routes in a few places (for modularity, see [main menu & layout](#)). We want to add phone book page to the main module. So, open `src\app\main\main-routing.module.ts` in the client side and add a new route just below to the dashboard:



# ASP.NET CORE ANGULAR

## DOCUMENTS

```
loadChildren: () => import( './phonebook/phonebook.module' ).then(m => m.PhonebookModule)
```

We get an error since we haven't defined PhonebookModule yet. Also, we ignored permission for now (will implement later).

## Next

- [Creating the PhoneBook Component](#)

Feedback

