

Group Project – Three 2-week Sprints

The group assignment will be to create an application using Service Oriented Design techniques. Each team will be expected to meet independently, create supporting design documentation, and code to produce the desired result. All documentation and source code will be maintained in your teams GitHub repository. Use GitHub's issues facility to keep track of which team member is doing, and to track progress and bug fixing. All team members will be expected to contribute – the instructor will review periodically and check commits to verify every team member is contributing. Service should be modularized into independently executable programs.

Problem:

Create an application that will provide a framework for providing a number of various services. The framework needs to provide continuous integration, therefore allow the software development team to provide new services dynamically.

Requirements:

Create an application to provide various services:

- **Translate English words into Foreign words.** Create services to translate English words into 3 different languages. User will pick one of three languages and type an English word to be translated into the chosen language. The translated English word is then presented to the user in the chosen language.
 - The foreign word will be displayed next to the English word the user provided.
 - The user can provide as many English words as they want in sequence, each time returning the foreign word
 - The user can change the foreign language and then translate English words into that language
 - Make is so that the application can easily add other foreign language translations dynamically to the application.
- **Determine federal taxes.** Determine the tax table for a particular year based upon the user's gross annual income. Create tax tables for the years 2020 and 2021.
 - User provides a tax year
 - User is prompted for gross income
 - Application returns the taxes to be paid
 - User can provide another gross income for that year and get the taxes to be paid for that income.
 - User can change the tax year and a different tax table will be used.
 - Make is so that the application can easily add tax tables for other years.
- Create an orchestration layer using an Enterprise Service Bus and a Service Broker.
- Create a utility layer that will return messages in different languages, base upon a setting in the client application.