# Design For EquationCalculation

## Goal:

This application aims at providing a RESTful service solving 2 linear equations with 2 variables. And the application should be able to be deployed to Azure platform.

## Structure & Detail:

Though this is a small project, it still has front end and back end.

First for front end, this project is single paged which means it uses AJAX to send requests to make the page not rendered again for new calculation.

Next for back end, it is written in Golang. There are 2 source files as main.go and equationSolver.go. More specifically, main.go implements the RESTful interface. Codes that really solve the equations are in equationSolver.go which first parses the equation string, and then does the calculation. If any error happens during the calculation, response with error message will be sent. In detail, there can be 3 kinds of error. First one is “Invalid String” which means the input string cannot be parsed correctly and the second one is “Equations Not Match” which means the variable name of the 2 equations are not matched. And the third one is trickier: if 2 given strings are relative like “x + 2y = 1”and “2x + 4y = 1”, the error should be “Infinite Number of Solution” if they are exactly the same (after adjusting the coefficient) or “No Solution” if they are not.

Finally for RESTful part, by calling the RESTful interface “/solution”, the request contains 2 equation strings, and the response contains the solution for both variables or the error message. Besides, the status code is 200(OK) for successful response and 400(Bad Request) for failure response.

## Docker & Deployment:

As mentioned before, this application should be deployed to Azure platform, so I use Docker to wrap the application into a container which can be deployed into Azure app service.