# Calculator Requirement Understanding and Prototypes

# Purpose:

Purpose of this document is to share the requirement understanding, approaches, prototype and tech stack to make sure, we are on same page.

# Calculator Requirement:

For your exercise, we need you to build a calculator. The backend should be a REST API using [ASP.NET](http://asp.net/) Web API 2 and C#. Create a MVC application using Angular 2 that consumes your API. Your calculator should exhibit craftsmanship as a developer, and the quality of code should be at the level you would feel comfortable submitting to a client. You should spend no more than 4 hours, if possible, to build your calculator. Please record, but do not include, environment setup within the 4-hour time limit. For example, environment setup for this project could be getting your server up and running. This is a complex problem and we do not expect you to create a perfect calculator in the given amount of time; however, we do expect it to show your working knowledge of .NET, C#, Angular 2 and other technologies. We would appreciate it if you could submit your exercise no later than next Thursday, November 10th.

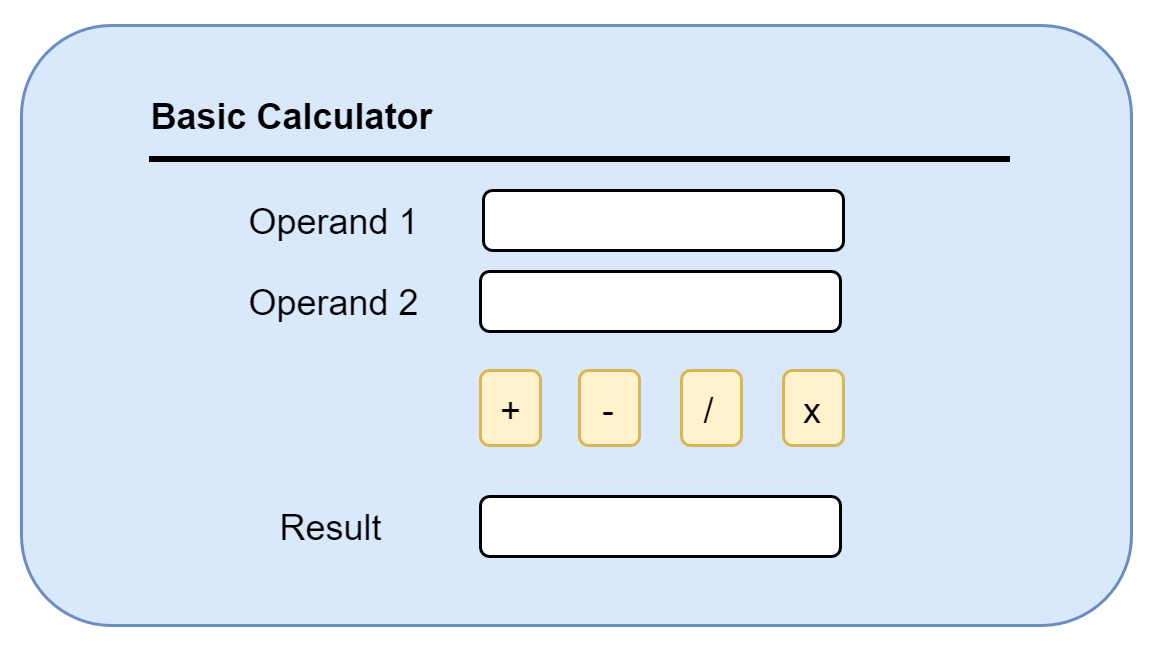
Once completed, please submit the solution via Github (or equivalent) and we will determine the next steps of the interview process.

Approaches:

I see two approaches to implement this requirement.

**Approach1:**

Implement basic operations like Add, Subtract, Multiply and Divide as per below Prototype.



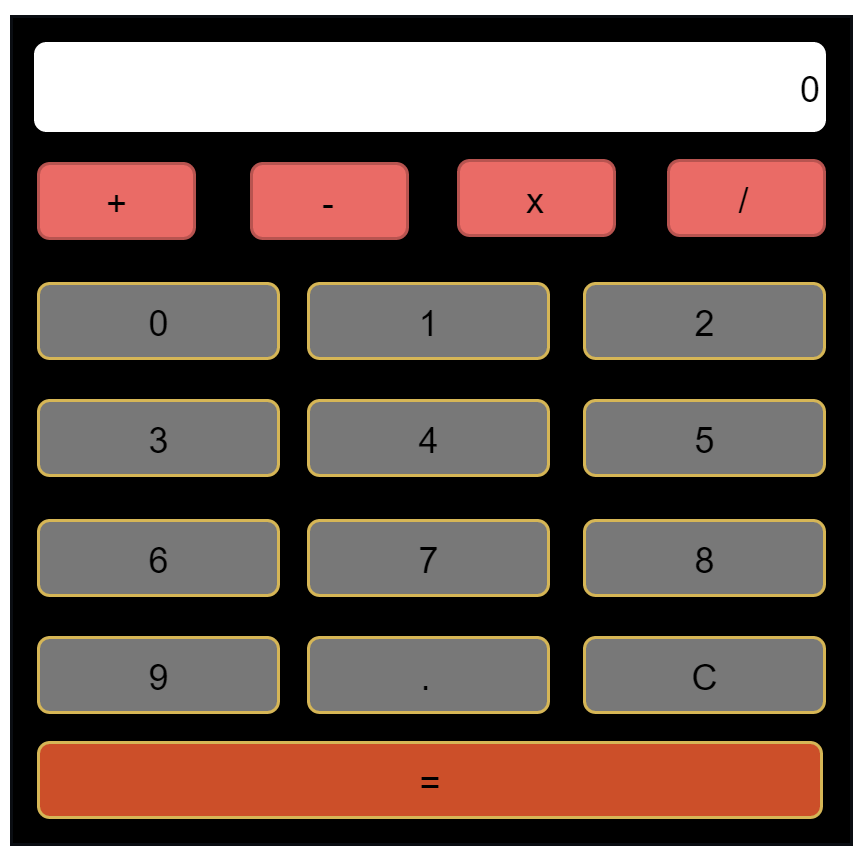
Implement REST API for operations like add, subtract, divide and multiply. e.g. –

URI: [http://[address]/api/[Operation]/[operand1]/[operand2](http://[address]/api/%5bOperation%5d/%5boperand1%5d/%5boperand2)]

Number input will be handled UI side using angular 2.

**Approach2:**

Implement Calculator as per below prototype.



URI: [http://[address]/api/math/[expression]](http://[address]/api/math/%5bexpression%5d)

Expression Example: 2+3x3-4

Expression will be passed as an input to math api and expression result will be the returned

As JSON output. Calculator keypad UI and client side operation will be handled using angular 2.

Technology Stack:

This requirement is to be implemented using below technologies.

**Backend:**

C# (for Library/Services), REST using .NET Web API(for API), JSON(for Data transfer)

**Front end (Client App):**

Angular2 MVC application using Typescript, Html, CSS, Bootstrap (Not using .Net MVC like Controller for routing and RAZOR view for UI)

Please confirm approach and technology stack.