**Programming Project**

# Task 1

## Planning

### Language and Technologies

Because the program will likely be used for schools/teaching, then I want to make it as easy as possible for the students to access wherever they are, whether it be at home or at school. This means I have chosen to build my program (known as Arithmetic) on a JavaScript/MongoDB framework known as Meteor. I chose Meteor as is well suited to the quick creation and distribution of browser-based apps, with built in reactive rendering, to allow me to easily port the application to run on mobile devices. The integrated package system makes it quick and easy to import new features and tools available online, and once I have finished writing my app, it can either be distributed as a binary, or published to Meteor’s own backend.

Because the teacher wants to use this system to manage a class of students, a simple account system would be useful to keep track of individual student’s performance over time, as well as letting students submit work in their name from anywhere. Meteor includes a full account management system built in, which I will utilise to create a secure, encrypted system to prevent unauthorised users from accessing the student/teachers data.

For storage, the framework includes a persistent MongoDB database system. This is automatically set up, without any manual actions required, or external databases to be set up. MongoDB is also well suited for quick scalability, so if the system were to be rolled out to a larger user base, it won’t have a problem. This also means that I will be able to store student results for as long as required, as they require very little space, and can be easily searched through.

For the interface, Meteor will be running its own web server, through which Arithmetic can be accessed. This means I will be using HTML and a templating engine called Handlebars, which will allow me to pass data from the JavaScript logic into the webserver, so displayed data will be updated live without the need to refresh.

For more information on the Meteor Stack, see <https://www.meteor.com/>

### Task Overview and Solutions

The task requires the program to be able to generate basic arithmetic questions, and then display them to the student. The student then needs to submit an answer. This will be repeated 10 times until the quiz has been completed, whereupon the results need to be marked and feedback given to the student on their performance.

INSERT FLOWCHART HERE!

## Coding

Due to the nature of the framework I am using, individual functions will not run chronologically – many are called depending on which page of the application is being viewed, and the actions of the user (e.g. pressing buttons or entering text).

To create a working application in Meteor, I need to create a ‘Template’ (HTML file), and then associate some logic (JavaScript) to run when that template is called (Web page is viewed) and interacted with (e.g. buttons pressed or forms submitted).

The template for the quiz is relatively short – I only need to display the question, question number and provide a way for the user to submit their answer. I have also written CSS to improve ease of use, and give feedback as appropriate.



Next, the logic for that template needs to be written.

