PROG 3017 Assignment 1

**FULL STACK DEVELOPMENT**

# 10% of overall course mark

# Prerequisites

### Required Labs: Lab 1 (Learning MongoDB)

# Summary

* As the first step in developing what will be your own Full Stack application, you are required to do the following in this assignment:
  1. Conceive and model the data of your choosing (ie. something that interests you) upon which you will build your simple application. Examples of entities might include:
     1. People
     2. Recipes
     3. Business Cards
     4. Online Technical articles
     5. Automobiles
     6. Books
     7. Pets

etc…there is really no limit to what you can choose.

* 1. Choose which properties/fields your entity will contain.
  2. Generate sample data based on your document.
  3. Create your database in MongoDB (local and cloud instances) and import your data.

Note: You don’t have to make your models too big. Keep it simple! Keep it to about half a dozen root-level fields so that you can work with your models easily when coding your API and creating your front-end user interface.

# Application Requirements (35 points)

1. **Create an ERD for your Data Design (10 pts.)**

The concept of an ERD does not go away when modeling data even for a Document Database such as MongoDB as you will still want to track data and relationships. The big difference is how you will implement the data relationships. The related data will either be nested in your main document (as a sub document) or referenced from a second document collection. It is highly recommended for this simple application that you make every effort to keep all of your data in one collection. (ie. nested data).

You are free to choose what data you wish to work with, however as a basic starting point, your document design must at the very least meet the following criteria.

* One of your fields must be of type *array*.
* One of your fields must be of type document (ie. a nested document).
* One field must be an image (usually an address or path to an image).

Create and demonstrate your ERD using an available tool. Suggested: <http://www.lucidchart.com>

1. **Generate Mock Data for your Design (10 pts.)**

Generate twenty(20) documents for your collection. You can do this manually or using the available online JSON Generator utility (<https://next.json-generator.com/>

). Shape a collection of documents that will reflect your chosen design. You may have to modify or tweak the data manually if the generator does not produce exactly the type of data you’re looking to create. If you wish, you can research other mechanisms for generating your document collection, or you can create the whole collection manually.

1. **Import Your Data into MongoDB (10 pts.)**

To begin development of your application’s API, you’ll need to import your generated data into your local instance of MongoDB (in your Docker container). You will demonstrate your data as part of this requirement.

1. **Create and store an Initial Backup of your generated data (5 pts.)**

There may come a time when you’d like to reset your data back to its initial state. In order to facilitate this, create a backup of your data.

Resources:

JSON Generator

<https://stoplight.io/blog/mock-json-generator/>

FullStack Academy – Modeling with MongoDB (<https://www.youtube.com/watch?v=4rhKKFbbYT4>

Rules of thumb for MongoDB Schema Design

<https://keon.github.io/mongodb-schema-design/>

# Upon review of this assignment, you may be instructed to add or modify your initial document model before proceeding with further work on your application.

# Instructions

1. Don’t forget that a code review is a **necessary** part of this assignment. You will need to show your code to the instructor in class on the due date while going through an evaluation of the site’s functionality. You will need to explain how the code works and complete the code review part of the rubric. You will need to do this to at least a developing level (see the Note in the rubric below).
2. ***Late submissions will be subject to the late penalties laid out in the course outline.***