DB DESIGN

INTRO

This project will utilize MongoDB for the database solution. MongoDB was the preferred database because of the performance it offers, as well as the format of the data storage. There will not be a need for relationships between the collections to create the application. Each collection should contain the data necessary to operate efficiently throughout the project. MongoDB Atlas will be the specific tool used in the project.

MongoDB will only be accessed via the Express API that uses REST principles. Actions taken on the front-end of the application will trigger API calls that will in turn run queries in the database.

DATA

Users:

```
{
  "_id": string,
  "email": string,
  "salt": string,
  "hash": string,
  "name": string,
  "membership": string,
  "isAdmin": boolean
}
```

Articles:

```
{
  "_id": string,
  "title": string,
  "author": string,
  "date": date,
  "image": string,
  "teaser": string,
  "body": string,
  "tags": array
}
```

EXPLANATIONS

Articles:
This is the fundamental portion of the application, as it is a blog. Articles must exist and be a robust collection with great performance. I've set up the structure of that collection to allow the documents to

exist on multiple pages throughout the website. For instance, on the homepage, we will display the title, image, and teaser. On the article page, most of the items are displayed with some exceptions like the teaser and the tags. Having the tags field will allow for an easy implementation of a search bar during future iterations of the project.

Users:
Users, like articles, is set up for future iterations of the project. Without this collection, the project would not meet the business criteria that is required for the Capstone course. The membership field is a string type, which will allow for paid subscribers to receive a special role with access to more articles compared to your user in the free tier (future iteration). The “Create Account” page will create users in the backend, meanwhile “Login” will log the users in. The isAdmin field will be useful when the “Create Article” page is created in a future iteration of the project.