**Database Design [MongoDB]**

-Although it would be fairly easy to set up a relational database for our data store, as the most important data being the perks and a concise description of each perk by fans. I am more comfortable and proficient with non-relational databases, hence, my decision to go with MongoDB.

-Upon reviewing the article and analyzing the structure of the article, the following are my thoughts on the design of a data store to capture all the important data from the article:

▪ The data we need to store to capture the essence of the article would be the *perks* and the short *description* to go with each perk. We will store each fans username, password, email address, the fans gender and whether a fan is admin or not for authorization/authentication purposes.

▪ This can be done in a pretty simple and efficient manner. We will store each *perk* and the *description* as strings. We will make each perk required and unique to keep the data store clean and efficient.

▪ The following will be the schema of our MongoDB store:

const mongoose = require('mongoose');

const Schema = mongoose.Schema;

// create a schema

const fanSchema = new Schema({

// Simple declaration of datatype that will be used:

name: String,

// You can add specifics to each one too that help with validation, like making something required, or unique

username: {

type: String,

required: true,

unique: true

},

password: {

type: String,

required: true

},

emailAddress: [String],

// Add 'enum' of an array of options to force selection between a given number of options.

gender: {

type: String,

enum: ["male", "female"]

},

admin: Boolean,

perk: {

type: String,

required: true,

unique: true

},

perkDescription: {

type: String,

required: true,

},

});

module.exports = mongoose.model(“Fan”, fanSchema)