Database Design

Maryville University

SWDV-691: Capstone Design

Marcelious Willis

For my PC Building Advisor application, I am intending to use a document data store such as MongoDB to persist the data that will be used by the application. One reason I am intending to use a document data store versus a relational database is due to the simplicity of the intended usage. For my application, I would not need complex SQL queries to find/filter data in the database and I believe a relational database such as PostgreSQL would be overkill, at least starting out. As the user selects different components in the application, the application would simply compare a value in the data structure with the value in the data structure of the other component. For example, when the user selects a specific motherboard and specific CPU, the application would then compare the socket value within the data structure; see data structure examples below. Another example is when the user selects a specific graphics card and a specific power supply. The application would compare the wattage values of those components to determine compatibility; see data structure examples below. Another reason I am intending to use a document data store such as MongoDB instead of a relational database such as PostgreSQL is the simplicity to update the database in the future. Since all computer components will not be included in the MVP, the database would need to be updated in the future. Document data stores would allow me to add new data structures or field to existing data structures without the need to update the object-relational mapping or affecting other documents. Below are examples of the JSON format structures for all computers components that will be included in the MVP.

Motherboards:

```
{
    "brand": "ASUS",
    "model": "TUF GAMING X570-PLUS (Wi-Fi)",
    "socket": "AM4"
}
```

Note, only data types for this data structure are strings

CPUs:

```
{
    "brand": "Intel",
    "name": "Core i7-9700K",
    "socket": "Coffee Lake"
}
```

Note, only data types for this data structure are strings

Graphics Cards:

```
{
    "brand": "NVIDIA",
    "model": "GEFORCE RTX 2080 Ti",
    "recommended_watts": 650
}
```

Note, the data types for this data structure are strings and integers.

Power Supplies:

```
{
    "brand": "EVGA",
    "model": "100-BA-0500-K1",
    "max_watts": 500
}
```

Note, the data types for this data structure are strings and integers.

Stretch Feature

The stretch feature I would consider adding per Tunde's peer review after the MVP is a tutorial video tab added to the website. The tutorial video webpage would have numerous videos that provide a walkthrough on how to put a computer together. Since the videos would be too large for the 16MB size limit of MongoDB's BSON documents, I would need to take advantage of MongoDB's GridFS specification to store the videos in file and chunk collections. See example below:

```
// *.files
{
    "filename": "cpu_motherboard.mp4",
    "chunkSize": 112233,
    "uploadDate": "2021-10-18",
    "length": 4373749
}

// *.chunks
{
    "files_id": "534a75d19f54bfec8a2fe44b",
    "n": 0,
    "data": "Mongo Binary Data"
}
```