Data Overview:

I will be using MongoDB to hold data and keep it persistent while running the web application. This database was chosen because I can use Mongoose to create relationships between the tables, which is necessary for various tables. MongoDB also provides free cloud hosting, which keeps costs down and allows me to easily scale based on my needs throughout the project and after the class if I decide to publish the site. The database will interact with the web application through the REST API layer built by Ionic and Angular to change or retrieve things as necessary. This prevents the database layer from being on the front end.

Users

This collection will simply hold the user's username and password. This will be used for a very basic login interaction for MVP purposes. I need the usernames to exist so that I can allow a singular user to edit their posts otherwise, it would not be possible to edit posts, or every user would be able to edit them, both options being unfavorable for my project. The username also allows me to have users save specific posts and link them to their accounts.

Document Structure:

```
{
"username": string (primary key)
“password”: string
}
```

Bracelet Designs

This collection will simply hold the properties of a bracelet design, marked by an auto-generated id. The unique id allows users to name the bracelet whatever they want without having to worry about a unique name. This allows me to link the bracelets to users' accounts and not worry about retrieving the wrong bracelet with the same name. All of the data will be supplied by the form to create designs, with the username and id being automatically supplied.

Document Structure:

```
{
“BraceletId”: int (generated by mongo db primary key),
"username": string (foreign key),
“name”: string,
“description”: long string,
“beadtype”: string,
“colors”: string,
“letters”: string,
“strands”: int,
“tags”: string
}
```

Saved posts

This collection will hold posts a user has selected to save. This uses two foreign keys to link the posts to the users. The id value is created as the primary key so that a user can save more than one post otherwise, with the username as the primary key, this wouldn’t be possible. The username and bracelet id are then linked so I can pull the appropriate saved designs based on the user.

Document Structure:

```
{
“Id”:  int (generated by mongo db primary key)

"username": string (foreign key)
```

“BraceletId”: string (foreign key)

}

Gallery Images

This collection is not part of a database but instead interacts with Cloudinary to retrieve images. This will allow me to store and retrieve images that can not be stored in Mongodb. Images will be optional for users to upload and will be retrieved in the gallery.

Document Structure:

{

“braceletimage”: string (url)

“uniquename”: string

}