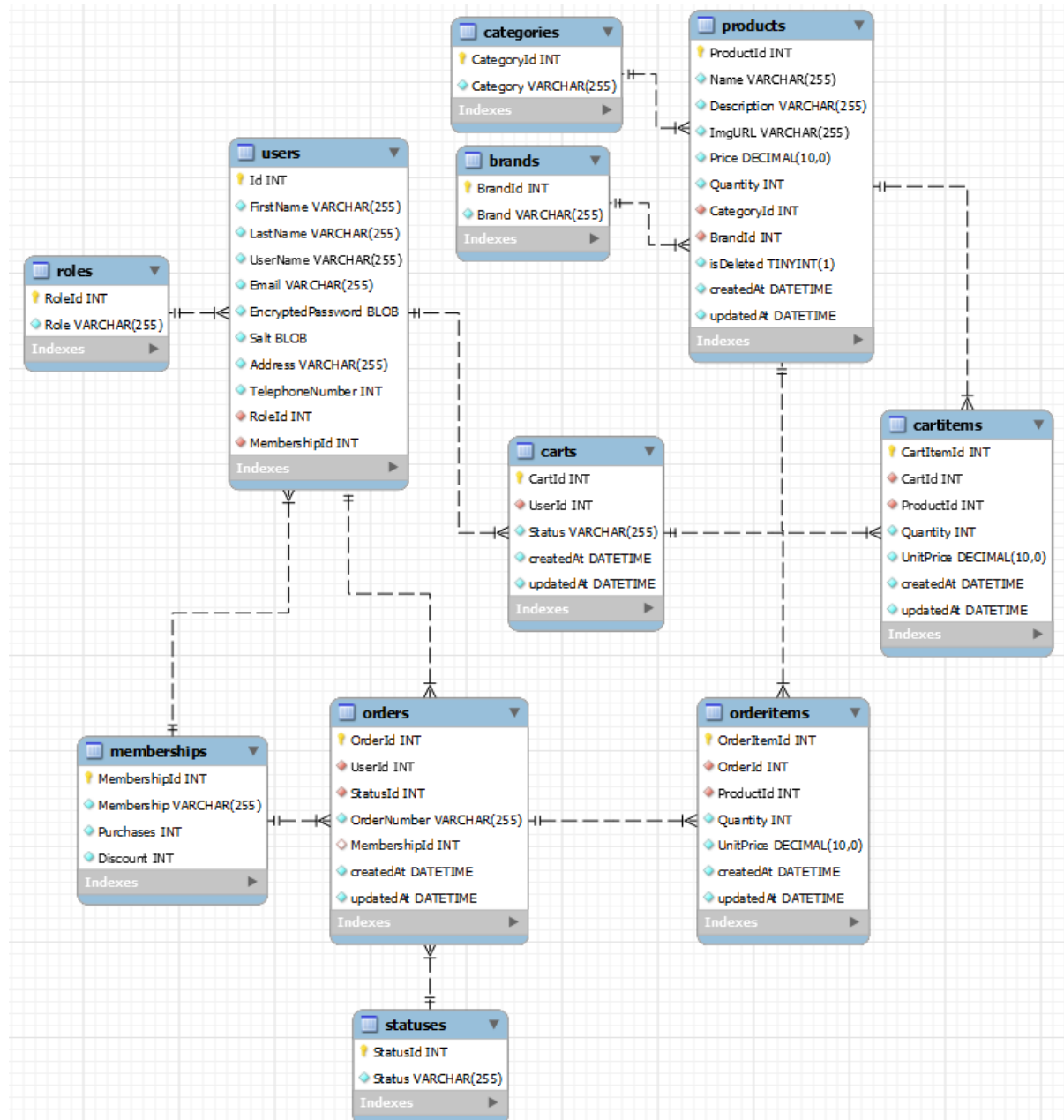


# Reflection Report

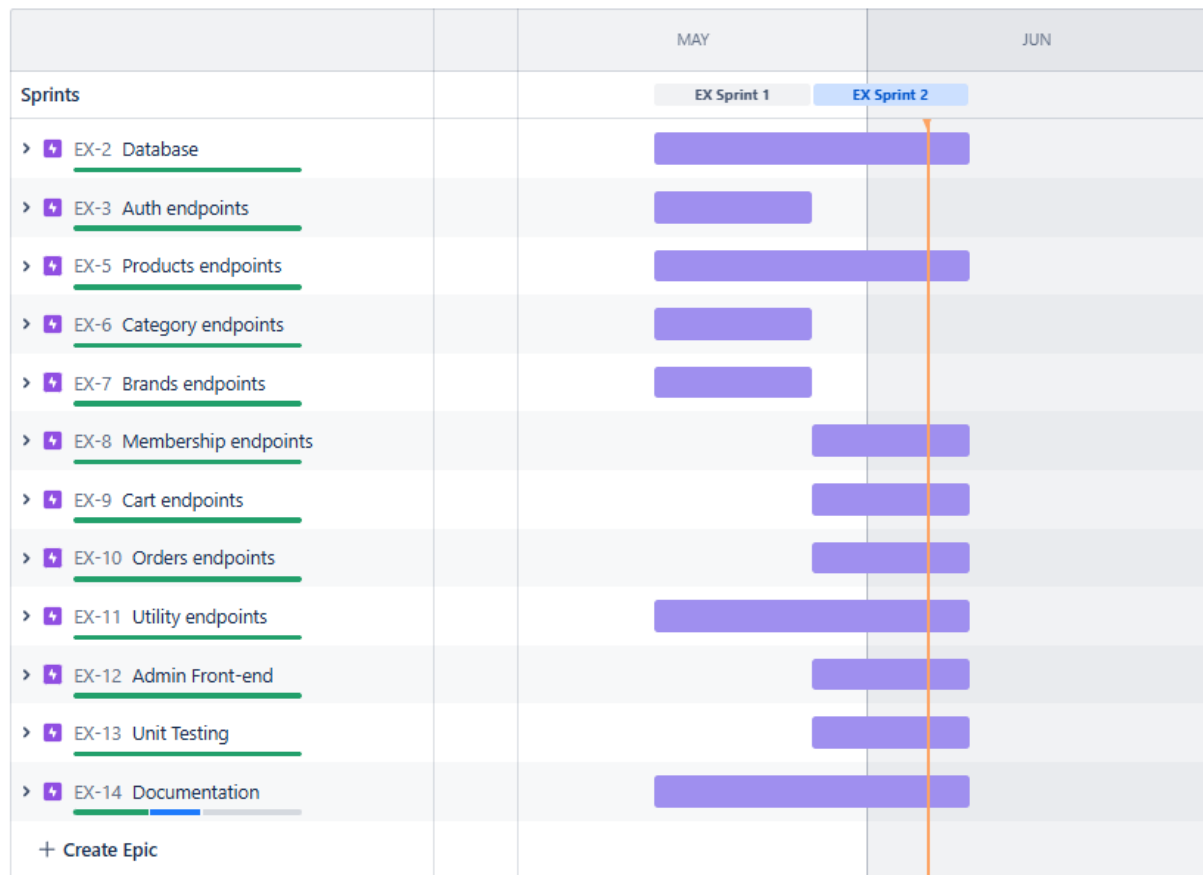
## Database ERD



- Roles and memberships can have many users, but users can only have one role and one membership, so therefore roles and membership have a one-to-many relationship with users.
- Categories and brands can have many products, but products can only have one category and one brand, so therefore categories and brands have a one-to-many relationship with products.
- A product can be added to many different cartItems and orderItems, but cartItems and orderItems can only have one product, hence the one-to-many relationship from products to cartItems and orderItems.
- Users can have multiple carts and orders, whilst one cart and one order can only have one user. So, users have a one-to-many relationship with carts and orders.

- Carts and orders can have multiple cartitems and orderitems, but cartItems and orderItems can only have one cart and order. So, carts have a one-to-many relationship with cartItems, and orders have a one-to-many relationship with orderItems.
- Orders can only have one status, but statuses can have multiple orders, therefore orders have a many-to-one relationship to status.

## Jira Roadmap



## Discussion

### Progression of the project

The beginning of the project was a slow one. The first week was spent mostly trying to understand exactly what the task asked us to do as well as plan sprints, as I often find that the hardest part of a project; the planning. Once I got into it, I started on the first parts of the task that I based the first sprint on; databases, authentication, products, categories and brands. Databases and products were pushed over to the second sprint, where the reason is explained down below. Utility endpoints were split between both sprints as I did the /init in the first sprint, and /search on the second sprint. Documentation was placed on both sprints as I worked on swagger documentation and references during both sprints. The second sprint was the most productive one for me, as I got more into the flow. The first sprint was also slower because there were a lot of different things that I felt a bit rusty on, so when I got into the second sprint, it was more productive as I better knew how to do the different tasks.

## Challenges faced during development

I faced a lot of challenges with the admin user interface, as it had been a while since I had done any front-end development, so it took some time to understand how to do that again. I also struggled a lot with the JWT token part, making me spend a lot of time figuring out how to lock the front-end part to admins only. I ended up using cookies, which worked well for that problem.

I procrastinated a lot on products in the beginning, mostly due to how complicated it all looked with the REST API link and the example of the JSON response. However, as I progressed with those endpoints, I realized it wasn't all that complicated after all. I was also intimidated by the cart and orders endpoints, not quite sure how I was going to proceed in the beginning. But once I looked at it more simplistically and tried to break the end result into smaller tasks, I finally understood how to proceed, and it became a lot less intimidating.

The databases were created during both sprints as I found it difficult in the beginning to figure out all the relationships, what tables were needed and how to follow 3NF, as I wasn't completely sure how to proceed with the earlier mentioned endpoints.

I also struggled a lot with errors from the middleware validations. I used a lot of try and catch, as well as next functions, and learned that those were often the cause for the "cannot set headers" errors. I also learned the importance of using `console.error` and `error.message`, as it made things a whole lot easier to understand and fix.