

Repository: <https://github.com/jpagan2/buy-nothing-rest>

Difficulties:

- The first difficulty I encountered was figuring out how to export the asks, gives, and thanks services to the account controller in order to be able to process all the requests made in the “accounts/” endpoint.
- Similar to that initial difficulty, I encountered another roadblock to get around the circular dependency of needing access to the accounts service in order to determine if the accounts are active to process the request or throw an exception if the account is inactive and trying to perform certain tasks restricted to active user accounts.
- In terms of business logic the biggest challenge was handling the note creation to update the lists of notes while simultaneously updating the conversation threads according to the type of the note and depending on the user and issue the note is responding/addressing
- Below I have outlined the biggest difficulty for me, which has been to generate the reports being able to pull from each module and filter the data to display it accordingly.

Status:

- **Complete** with bugs
 - Complete: All endpoints are functional to process data as intended calling on the pertinent service to handle the request and return the appropriate information
 - Bugs: Report Generation (Postman Test 67)
- **The unit testing is complete**
 - Testing all the functionality of the application with the different use cases of each module’s service
 - Unit testing is ignoring the controllers as specified by the Q/A documentation indicating it’s not necessary since I developed the controllers to be *very light* to simply call on the service responsible for handling the request. DTO’s are ignored as well since they are simply imported to the controller in order to pass in the data from the creation requests to the service in with the appropriate data types.

Statistics:

- Lines of code in Project
 - 481
- Unit Test Coverage
 - 92%