

Review Group: 5

Name: Breno Kenji Ogata Bianchi

NUSP: 10260372

Name: Mariana Aya Suzuki Uchida

NUSP: 11236205

Name: Pedro Fernando Christofolletti dos Santos

NUSP: 11218560

Group Being Revised: 6

Name: Guilherme Holanda Sanches

NUSP: 10734370

Name: Ygor Pontelo

NUSP: 10295631

1. Project Quality: ★★☆☆ (3.7)

a) Requirements:

Requirements	Situation	Comments
Client	Ok	<ul style="list-style-type: none">- I liked the layout, very cool to have the book collection and wish list, however, it makes more sense to me that the user's collection is the first thing to be shown rather than the people the user follows.- I liked the minimalist design and the addition of categories to separate the books; I think it would be nice to name each category.- I liked the design, it's simple and clean. However, in the client profile, I think it would be good to implement the friend list as a 'slider' similar to Figma's profile.
Admin	Missing	Not implemented yet
Products	Ok	<ul style="list-style-type: none">- I liked the layout, it was nice and clean, but it could have more categories and filters, and show the rating of the book.- The layout is clean and objective; I think it would look prettier if the full book cover fit the product block.- The products are well organized.
Product Sale	Ok	<ul style="list-style-type: none">- I liked the simplicity, it would be perfect if I had comments from users who have already read the book.- The layout makes information easy to be found.- It is simple and easy to understand.
Product Management	Missing	Not implemented yet
The group Functionality	Ok	<ul style="list-style-type: none">- The new features chosen were well thought out and are consistent with the character of the project, but not all were presented in the client like "Review" and "Reader".

		<ul style="list-style-type: none"> - Some functionalities such as the 'book review' weren't implemented; and the 'local best-sellers will be highlighted' part is a little confusing since the store sells e-books and the phrasing makes it look like they also sell physical books. - Most functionalities are implemented, but they have some minor problems.
--	--	--

b) Project Description:

The wishlist was the only functionality present in the project description that was implemented, other functionalities such as the promotional banners the administrators can edit, the reading feature, or the product reviews weren't implemented yet. Project-specific features have been placed in the description but should be in the requirements, the description could be more complete and go deeper into how the features will be implemented and why. Also, talk more about how users can interact, given that there is the possibility of following people.

c) Comments About Code:

Since the group didn't use many external libraries, what was commented is enough and easy to understand. However, it would be interesting if the group explained how routes are being implemented and how Vue store states are being handled.

The functionalities that are implemented have some minor problems like it is not possible to remove an item from the wishlist/ cart.

Some other functionalities, such as the login with a password will be added in the next milestone.

d) Plans and Test Results:

The group has not formulated a test plan yet, just said it will use postman to test the requests, but it has not been explained what a possible route for a user would be, ranging from logging into the site to buying and reading a book.

e) Build Procedures:

The steps to create the build are clear, we were able to download and run the application without any problems.

g) Problems, General Work Quality, etc.

*The group has not yet reported any issues or comments in the GitHub readme.

- The store's logo is cropped from the page.
- The "npm" package manager found 37 vulnerabilities, being 25 moderate and 12 high. There are also several warnings from functions that have been deprecated in some modules.
- It's not possible to perform a logout;
- It's not possible to review a product;
- After purchasing a book, it does not show up in the user's profile;

- You can't remove items from the wishlist or the cart;
- You can't use the "Reader" functionality;

*Don't forget to add the node_modules folder in gitignore.

~~f) Group Interactions:~~

2. Code Quality: ★★★★★ (4.5)

a) Code Design and frameworks' use:

The code design is well done, the use of Google Fonts and the bootstrap was a good choice, the code is well indented, good use of names for elements, classes, ids, filenames, it's easy to understand.

The code works with the VueJs framework, an excellent choice.

b) Code Organization:

The code organization is good, separating the different functionalities into different files and folders. But it could present some comments to explain how some functions work.

c) How the code works:

The group is using the Vue.js framework, so the basic functioning of the code consists of a single HTML page called index that calls in an "app" div the entire application, the framework uses the "npm" package manager to install modules that help the implementation and organization of application states and the formulation of components.

The package.json is the file where all the scripts and dependencies used are, as well as the project name and version, as well as the package-lock. json is automatically generated for all operations where npm modifies the node_modules tree or package. json. It describes the exact tree that was generated so that subsequent installations are able to generate identical trees regardless of intermediate dependency updates.

The main.js file, located at the root of the source folder, is responsible for starting the application already inserting the routes and the Vue store, which contains the functions that change the application's state and the state variables, in addition, there too vuetify starts. In the main file "App.vue" is where the magic happens, it is the first component that will call all the others that are in the "views" and "components" folder with the help of VueRouter that contains all the possible routes of the application and if finds in the "router" folder.

Finally, we have the "assets" folder that contains the logo images and the "data" folder that contains the .json files used temporarily to simulate the data until the api is complete.

d) Code Documentation:

It does not have in-line comments, but most class and id names are self-explanatory. For now, it is good and easy to understand.

3. What you would make differently / Suggestions and ideas:

- I would add an option to perform a logout.
- I would add the possibility to see friend's profiles.
- I would add an option to remove a book from the wishlist.
- I would add the 'recently read' tab in the profile page so the focus of the main page can be the new books the client could buy; the bookmarks could show the book's title to make navigation easier.
- The reading tool must be well implemented and allow zooming in and out, allowing page and chapter change at any time and the possibility to mark the place where you stopped reading to go back there later.
- You could implement the functionality to add public comments for each chapter of a book that can be viewed at the bottom of the reading page.
- If you allow the profile of friends to be viewable, it is important that you add the functionality of viewing permission, that is, the user who decides whether he wants to make his profile viewable or not.
- You could separate books by genre (i.e mystery, romance, etc) and add a search functionality for it, with search exclude included (possibility to not show one or more specific genre).
- You could add your favorite books and the ones you're currently reading in your profile.
- Add a "recommend books to friends" feature. You could even turn this feature into a way to increase sales, for example, if a friend buys a book you recommended, both get a 5% coupon on the next purchase.
- For this type of store, especially with the possibility of evaluating the content and public profile, an area of interaction between users is very good, in the future you could implement a discussion forum or blog.
- Don't forget to add the node_modules folder in gitignore.
- Remember to use "npm audit fix" to automatically fix them or correct them manually.