eCommerce Application

Objective: Build a multi-tenant eCommerce platform with Laravel and Vue.js, leveraging multi-database tenancy for strict data isolation, and publish the finished project on a public GitHub repository.

Requirements:

Multi-Tenancy Setup with Multi-Database:
- Implement a multi-tenancy architecture where each tenant (store) has its dedicated database.
- Set up automatic or manual tenant onboarding, creating separate databases per tenant dynamically.

User Authentication and Authorization:
- Implement user authentication and authorization within the multi-tenant, multi-database environment.
- Ensure proper segmentation of users and their associated databases to maintain data isolation.

Product Management:
- Develop CRUD functionality for managing products within each store (tenant), ensuring data stored in the respective tenant-specific database.

Shopping Cart:
- Implement a shopping cart functionality within the multi-tenant, multi-database environment.

Frontend Development using Vue.js:
- Design and implement the user interface for the eCommerce platform using Vue.js.
- Integrate with the Laravel backend to fetch and display tenant-specific data stored in their respective databases.

Database Configuration and Connection:
- Configure Laravel to dynamically connect to different databases based on the current tenant's context.
- Ensure proper database switching and isolation when serving requests from different tenants.

Upload to Public GitHub Repository:
- Upload the completed project to a public GitHub repository.

Testing:
- Write unit tests and/or feature tests to validate data isolation and functionality across multiple databases and tenants.

Bonus/Optional:
- Implement database-level caching mechanisms for improved performance per tenant.
- Apply security measures such as input validation, data encryption, and protection against SQL injection for each tenant's database.
- Document the multi-tenant, multi-database architecture and key components for reference.

Deliverables:
- A functional multi-tenant eCommerce platform demonstrating data isolation among tenants using separate databases.
- Codebase with clear comments, documentation, and adherence to Laravel and Vue.js best practices.
- Project uploaded to a public GitHub repository..

Evaluation Criteria:
- Successful implementation and validation of multi-tenancy with multi-database setup.
- Code quality, readability, and adherence to best practices in Laravel and Vue.js.
- Effective management of data isolation and operations across different databases for each tenant.
- Timely completion and upload of the project to the public GitHub repository.

Thank you for your work on this task. Your efforts are greatly appreciated!
Eng Mohab Mamdouh