**Technical Test**

Objective:

The main objective of this technical test is to assess your proficiency in Laravel's latest features, your code quality, along with your ability to create a user-friendly UI using Vue.js to validate your JS knowledge. We also aim to understand how well you convert user requirements into functional software and try to use the industry best practices to write better code and deliver values.

Duration:

Approximately 3 hours

Tools Required:

* Laravel (latest version)
* Vue.js
* MySQL or SQLite for database
* Any text editor of your choice (PHPStorm, Visual Studio Code, Sublime Text, Atom, etc.)
* Git (for version control)

Task:

You are required to create a simple task management system. The system should include user registration, login/logout functionality, task creation, and task assignment.

Details:

* User Registration: A new user should be able to register themselves providing essential details like name, email, and password.
* User Login/Logout: Users should be able to log in and log out of the system.
* Task Creation: Logged-in users should be able to create tasks. Each task should include a title, status(open, in-progress, done), description, and deadline.
* Task Assignment: Users should be able to assign tasks to other registered users. An email notification should be sent to the assignee when a task is assigned.

Guidelines:

* Laravel: Use Laravel's latest features such as Events, Queue management, Listeners, Notifications, Mailable, etc., where applicable. It is important that you figure out where to implement which advanced laravel feature to make the application better.
* Vue.js: Implement the front-end using Vue.js to create an SPA (Single Page Application). If you prefer other JS frameworks, no problem. Your ability to create components, manage state, and create responsive and user-friendly interfaces will be evaluated, how good you are in JavaScript.
* Database: Use Eloquent ORM for database interaction.
* Coding Style: Your code should be clean, well-organized, and easy to read.
* Version Control: Use Git for version control. Proper usage of commits and branches would be considered a plus.
* Commenting: Ensure your code is properly commented, explaining why certain actions are performed. Especially in complex parts of the code.
* Error Handling: Your application should gracefully handle possible errors and edge cases.

Deliverables:

* A link to the Git repository containing the source code.
* A brief documentation in the README.md file explaining your design choices and any instructions necessary to run the project.
* Submit your submission by email (just send a reply to the initial mail)

Evaluation:

Your test will be evaluated based on the following:

* Completion of the given task.
* Understanding and usage of Laravel's latest features.
* UI design and application structure.
* Database design and Eloquent ORM usage.
* Coding standards and style.
* Usage of version control (Git).
* Ability to meet user requirements and deliver a functional product.

Good luck, and we look forward to reviewing your submission!