**Developer EXAM**

* **Rules:**
  + You have 5 days to complete both the frontend and backend exams once you receive this email.
  + Please ensure that your submission includes the source code, documentation, project support files (such as make files and test code), or any other related materials necessary for building your code and comprehending the approach taken in your solution. Additionally, kindly specify the total time required to create your solution.
  + Upon completion of the test, please provide us with a public **GitHub repository** with your solutions.
  + Ensure that your work predominantly showcases your personal effort. While exercising your own judgment, bear in mind that evaluators will assess whether your submission upholds the intended spirit of the exercise. Failure to comply with this guideline may result in a negative evaluation.
  + Avoid copying or plagiarizing the work of others. If you utilize open source or third-party code, clearly distinguish it and provide proper citation for the authors. Maintain a clear separation between any incorporated third-party code and your own work.
  + The code should include documentation to facilitate its execution and testing. Provide detailed information regarding the platform, versions, and any other relevant details used during the development and testing of the code.
  + Implement your solution in such a way how would you implement it on a production system(Having best practices enforced, design patterns, SOLID principles).
* **Frontend Exam:**
  + You can use **Vanilla JS or Framework(React, Angular, Vue)**. Using **Typescript** is a plus point.
  + Use the [JSON Placeholder API](https://jsonplaceholder.typicode.com/) to display a nice UI and only display the **first 20 records**.
    - **endpoint:** <https://jsonplaceholder.typicode.com/todos>
    - UI can be a **table, a collection of cards** or any other representation of data that demonstrates your **frontend skill**.
    - Ensure that the user interface is **optimized for mobile devices**.
    - Include an **Add** button that will call a form and create a new record and add it to the list (there's no need to create it on the backend, just add it to the list).
    - Include an **Edit** button for each record that will transfer the data to a form (there's no need to submit, just transfer the data into a form).
    - Include a **Delete** button for each record which will delete the record on the list.
    - Include a **search functionality** that will scan the list and return the searched record.
    - **DO NOT copy** another developer’s work. The purpose of this scenario is to demonstrate your effort and knowledge.
    - Provide your source code, test code, and any documentation or supporting file to execute the project.
* **Backend Exam:**
  + Kindly develop a REST API utilizing **NodeJS and TypeScript**. Note that no user interface is required for this test, and the use of a database is not necessary for this solution.
  + Has a resource route which **outputs** **a random employee** object with property **id**, **name** and **position.**
    - Example: {”id”: 1, “name”: “John Doe”, “position”: “Frontend Developer”}
  + A resource route that **adds** a list of new employee object with property **id**, **name** and **position**.
  + A resource route that **deletes** a specific employee object.
  + A resource route that **updates** a specific employee object.
  + A resource route that **outputs the full list of employee**.

Example: [

{

“id”: 1,

“name”:”John Doe”,

“position”:”Frontend Developer”

},

{

“id”: 2,

“name”:”Jane Doe”,

“position”:”Marketer”

},

{

“id”: 3,

“name”:”Juan Dela Cruz”,

“position”:”Clinician”

}

]

* **Note:** **All outputs and inputs are formatted in JSON**
  + **Summary for both Exam (Please document honestly in the project readme file)**
    - How long it took you to complete the work
    - What you had to research to complete the work
    - Question: If you did use a database for the backend project, what kind would you use and why?
      * Can you give an example of how would you delete an employee from the database?