# Fujitsu Die-Tech Corporation of the Philippines

# Manufacturing Information System

# Assessment Test for IT-STAFF

|  |  |
| --- | --- |
| **Date of Exam :** | **August 16, 2022** |
| **Name of Applicant :** | **Rico Estribo Guinanao** |

**Exam # 1 : Code Debugging (Easy Part)**

1. **Parts by Section**

Try to Complete this ***PartsBySection***  function that:

* Accepts an Associative array containing the Machine Parts to designated Section Area.
* Returns an Associative array containing an array of Machine Parts for Section Area, in any order.

//Codes

<?php

class SectionArea

{

public static function partsBySection ($data)

{

return $data;

}

}

$data = array

(

"Battery" => "Assembly",

"Tablet" => "ITS",

"Magnet" => "Assembly"

);

print\_r(SectionArea:: partsBySection ($data));

Example Output: ["Assembly " => ["Battery ", " Magnet "], " ITS " => ["Tablet"]].

1. **Merge EmployeeNames**

Complete the **unique\_names** function. There will be two array parameters need to pass inside the function, it will return an array containing the names that appear in **either or both** arrays. The returned array should have **no duplicates**.

//Codes

<?php

class EmployeeNames

{

public static function unique\_names ($array1, $array2)

{

return array\_unique(array\_merge($array1, $array2));

}

}

//Do not change this part

$empnames = EmployeeNames::unique\_names([“jack”, “marwan”, “balato”], [“balato”, “Arniel”, “Ivana”]);

echo join(', ', $empnames); // should print jack, marwan, balato, Arniel,Ivana

Example Output: jack, marwan, balato, Arniel,Ivana in any order.

1. **Palindrome**

Write a function that checks if a given word is a palindrome. Character case should be ignored.

//Codes

<?php

class Palindrome

{

public static function checkPalindrome($word)

{

echo strrev($word) === $word ? "True" : "False";

}

}

//Do not change this part

echo Palindrome:: checkPalindrome ('Deleveled');

Expected Output: True or False

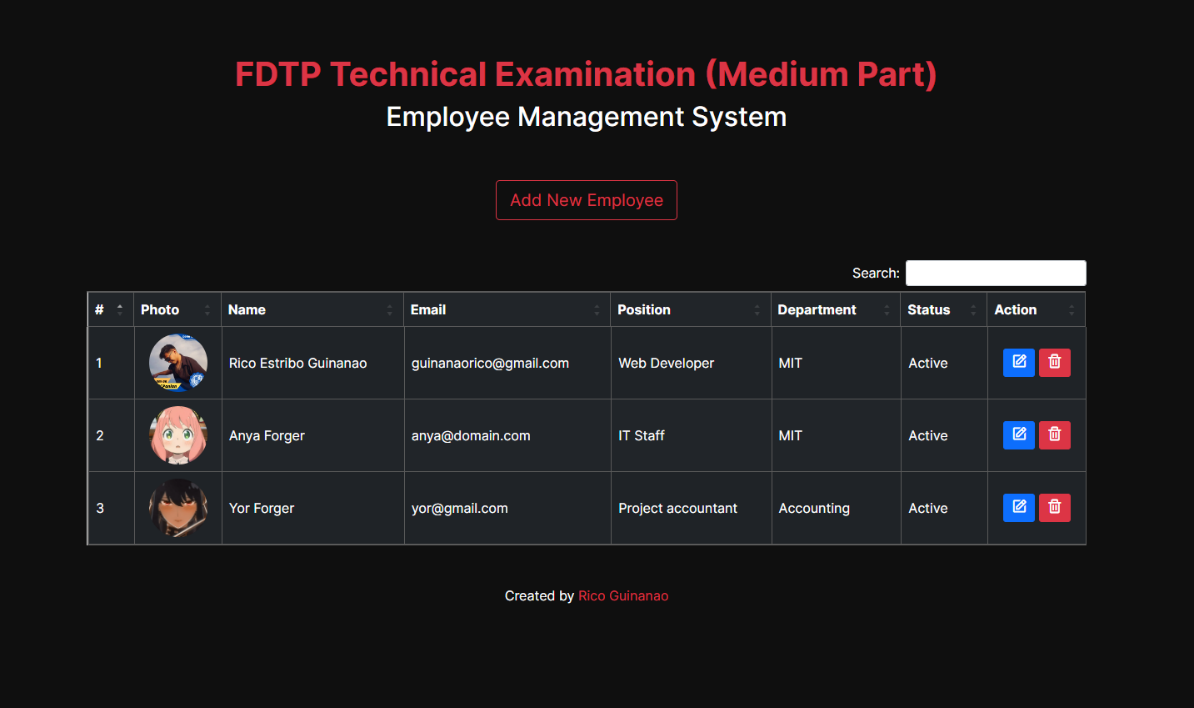
**Exam # 2 : PHP: CRUD (Medium Part)**

Create a Program using your desire PL ,which focus on Employees Module using **CRUD** (create, read, update, and delete) and Database using **MySQL/Postgre (or other Database**). Must have :

* Adding/Register New Employees
* Edit/Update Employees Information
* Deleting Resigned Employees
* Load all registered Employee on Data table

**Bonus Points** (Optional) - Having an additional feature like:

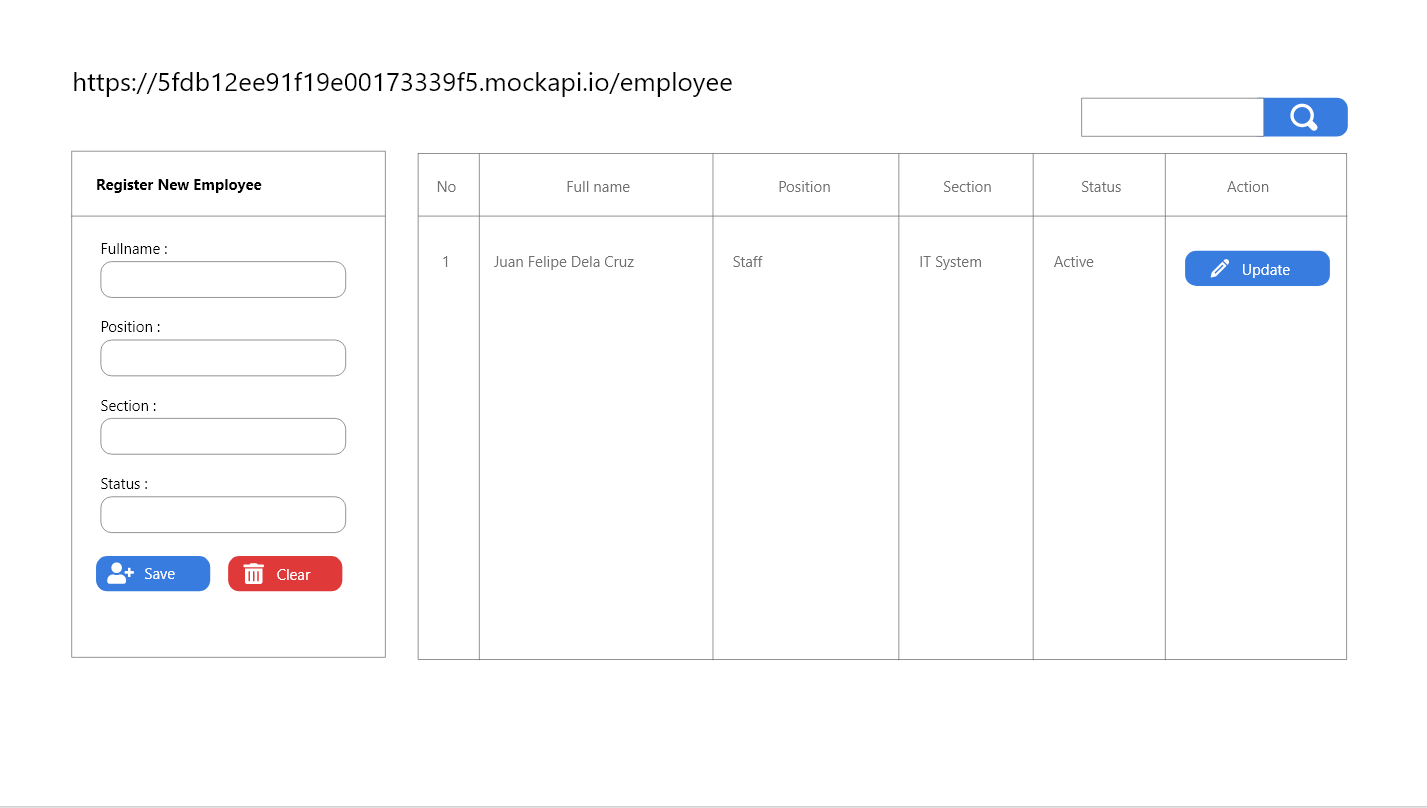
* uploading file function to register a new employee is a plus.
* A Good User-Friendly Interface
* Using **OOP (Object Oriented Programming)** techniqueis a Plus
* Using PHP Language is a Plus
* Database: Table Relationship

**Github link:** <https://github.com/coricss/php_crud_fdtp.git>

**Exam # 3 : Design the API (Average Part)**

Create a program that will fetch all the data from API and view in HTML Table using JavaScript.

API Sample UI :



To load all data : [GET] method <https://5fdb12ee91f19e00173339f5.mockapi.io/employee>

**Bonus Points** (Optional) - Having an additional feature like:

* A Good User-Friendly Interface
* Adding / Edit / Delete employee Function
* UI using CSS or SASS
* Using Framework (Vue,Angular,React, Etc.)

**Github link:** <https://github.com/coricss/fdtp_api_exam.git>