**Tarea - TIA-03**

* **Modalidad de Evaluación: En grupo**
* **Peso: 10% (de la nota final)**
* **Metodología**: Aprendizaje Basado en Problemas (ABR)

**EQUIPO: \_\_\_\_\_\_**

**MIEMBROS DEL EQUIPO:**

* Estudiante 1
* Estudiante 2

**Caso de Estudio**

El Caso de Estudio está relacionado con los Proyectos PA/PIA. Debe tomar en consideración todo el material que se le ha suministrado como el enunciado que se le entregó en la Tarea 2 (TIA-2), el formato de registro de proyecto, la información que socializó la Profesora Vesna Srdanovic en clases, la revisión de los diferentes diccionarios de datos de los otros grupos y toda la información relacionada con los proyectos que Ud. puede recolectar (puede consultar a otros docentes en relación al tema)

**Material de ayuda:**

* Puede utilizar el contenido del ejercicio del Taller Rest-HTTP del 28-03-2025 relacionado con piscinas y adecuarlo a los datos de esa tarea

**Criterios de desempeño / Indicadores de Resultado de Aprendizaje**

* Utilizar los códigos de respuesta HTTP/HTTPS para comprender el estado de las solicitudes y respuestas en una aplicación web.
* Aplicar el uso de APIs RESTful para la comunicación con un servidor backend y el consumo de dichas APIs para la obtención y manipulación de datos.
* Desarrollar habilidades de trabajo en equipo y comunicación efectiva en la resolución de ejercicios prácticos relacionados con tecnologías web.
* El pensamiento sistémico y complejo
* La visión global
* La multi e interdisciplinariedad
* La vinculación solidaria
* Las emociones y los comportamientos
* La autonomía

**Actividad:**

* Analizar el Caso de Estudio suministrado por el Profesor.
* Diseñar e implementar una página Web responsiva que establezca una interacción con un Servidor
* implementar llamadas APIs RESTful para la comunicación con un servidor Backend
* Implementar respuestas con códigos HTTP.
* Conectar y actualizar una base de datos con las cuatro (4) operaciones básicas de un CRUD: INSERT, UPDATE, DELETE, SELECT

**Propósito:**

* Comprender el contexto de un problema o necesidad real y presentar una propuesta de solución
* Comprender cómo funciona un sistema de interacción FrontEnd / BackEnd
* Profundizar los conocimientos de una arquitectura REST
* Practicar con un sistema de Solicitud/Respuesta con el uso de códigos HTTP
* Aprender a conectar y actualizar bases de datos con sistemas de información Web

**Modalidad de entrega:**

* Los resultados de la Tarea se deben subir al repositorio GIT de tareas de la asignatura en la carpeta Tarea-3 (TIA-3). Nota. Permite el acceso al público y al docente para acceder y ver el contenido.
* En el Classroom solamente colocarán UN SOLO LINK al repositorio. Lo colocará el líder del grupo

**Productos (entregables):**

* Plantilla de Informe con los resultados
* Proyecto Node con conexión y actualización de base de datos (carpetas y sub carpetas)
* Repositorio GIT con las Tareas: TIA-1, TIA-2 y TIA-3
* Equipo de miembros en la página principal del GIT
* Breve descripción del Proyecto PIA en el GIT

**1.- Base de Datos “proyectos” y Tabla de Tipos de Proyecto**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Tabla** | **tipo\_proyecto** |  | | | |
|  | **Descripción** | Tipos de proyecto de Aula que desarrollan los estudiantes durante el semestre | | | | |
| # | **Campo** | **Descripción** | **Tipo Dato** | **Tamaño** | **Restricciones** | **Tablas**  **Relacionada** |
| 1 | **id** | Id de identificación | smallserial |  | Primary Key (PK) |  |
| 2 | **codigo** | Código tipo proyecto | varchar | 8 | Unique Key (UK) |  |
| 3 | **descripcion** | Descripción del tipo de proyecto | varchar |  | Not Null |  |
| 4 | **abreviatura** | Nombre abreviado | varchar | 6 | Not Null |  |
| 5 | **fecha\_registro** | Fecha/hora de inserción |  |  |  |  |

* *Crear la Base de Datos* **“proyectos”** *en PostgreSQL (pgAdmin4)*
* *Elaborar el script “***001-create-tipos-proyectos.sql***” de creación de la Tabla (CREATE).*
* *Correr el script en el “query editor” de pgAdmin4 (PostgreSQL)*
* *Presentar en esta sección un pantallazo de la script de creación ejecutándose en pgAdmin4*
* *El Pantallazo debe estar centrado y que se pueda entender. NO COLOQUE el código fuente en esta sección.*
* *Utilizar la tabla anterior para desarrollar la propuesta de sistema*
* *Colocar este script en la sección de migraciones del proyecto Node*
* *Borrar las indicaciones en azul una vez completado el ítem*

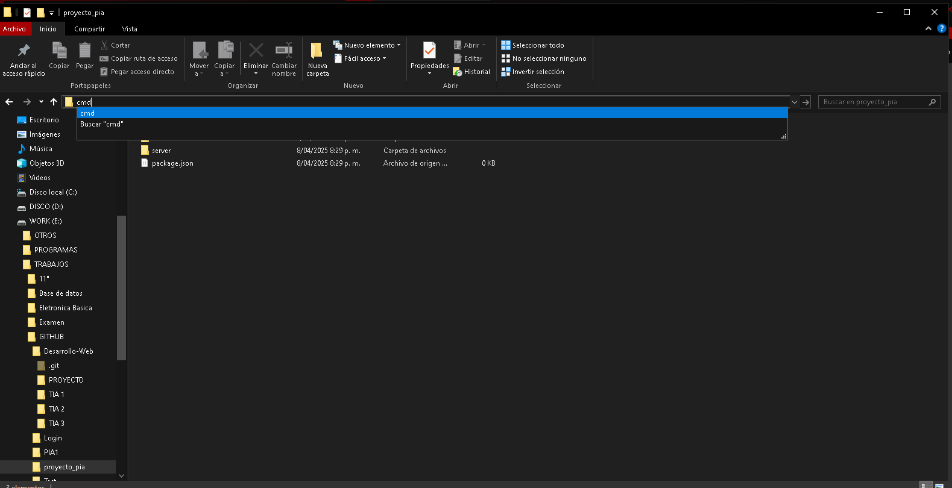
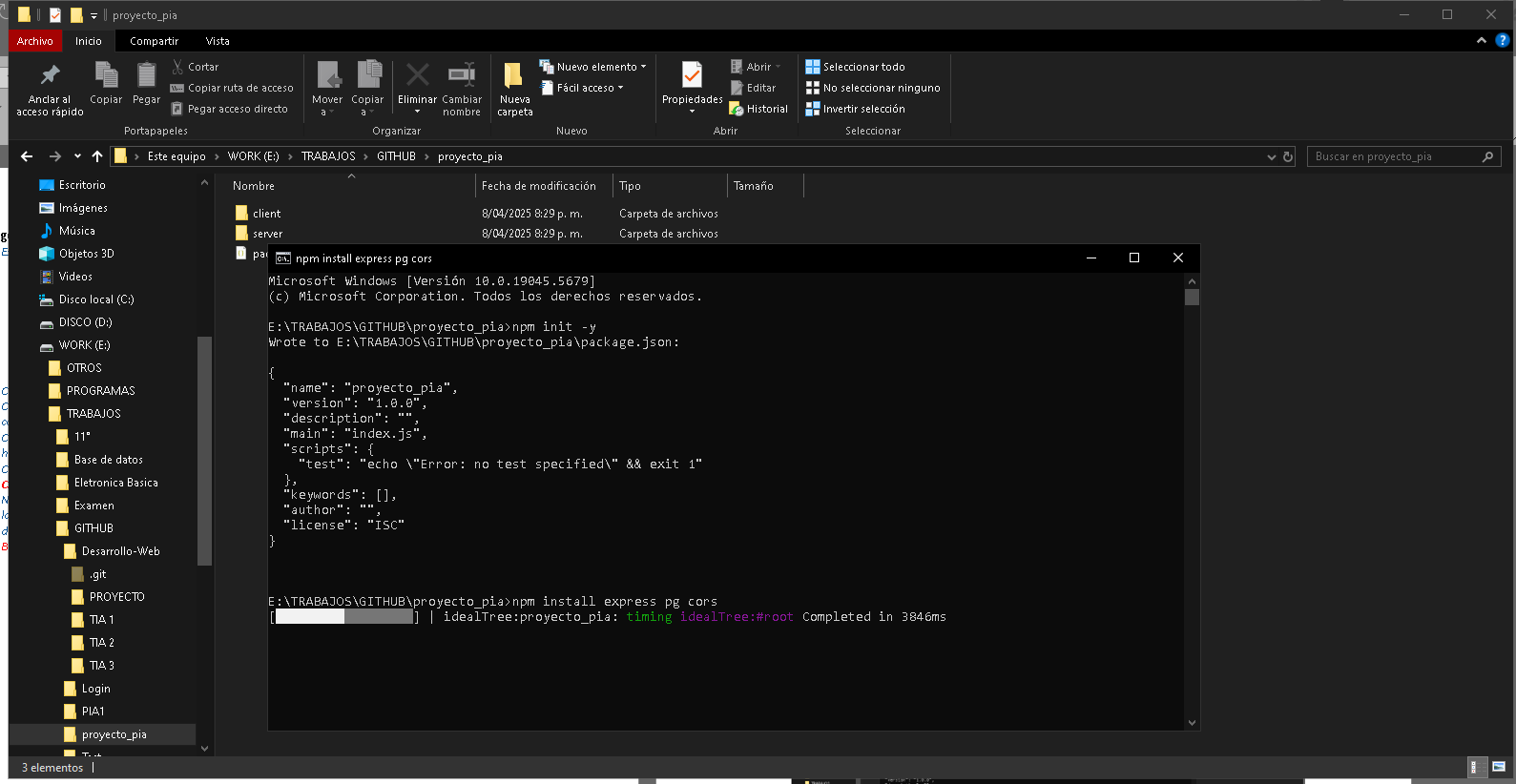
**2.- Estructura del proyecto FrontEnd/BackEnd**

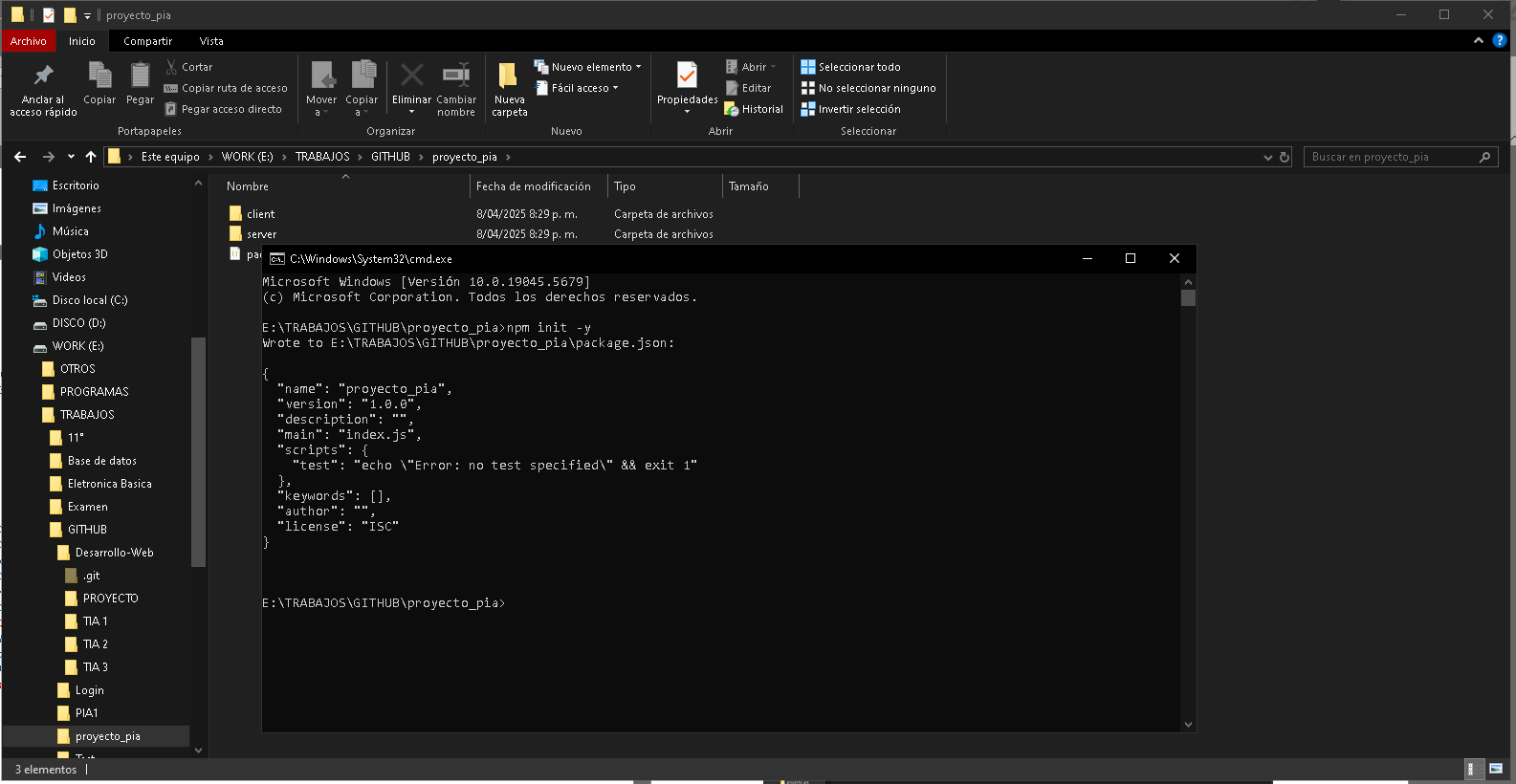
* *Crear la estructura de un proyecto Node con el nombre “***proyectos\_pia***”*
* *Crear un archivo batch con las instrucciones de creación de carpetas y archivos*
* *Este archivo batch lo debe subir al repositorio*
* *Nombre del archivo batch:* **crea\_estructura\_proyecto.bat**
* *Mostrar un pantallazo del contenido del archivo .bat en esta sección. Explicar brevemente de que se trata este archivo*
* *Nota: recuerde que debe tener la misma estructura del proyecto abordado en clase (cliente, servidor)*
* *Borrar las indicaciones en azul una vez completado el ítem*

**3.- Código fuente del Proyecto**

* *Escribir el código fuente de todos los archivos del proyecto*
* *Codificar archivo HTML con los datos del tipo de proyecto y las operaciones de listar, agregar, modificar y borrar tipos de proyecto*
* *Codificar el archivo CSS. Cambiar los colores y la colocación de los objetos en pantalla proponiendo un propio estilo (modifique el estilo del ejemplo abordado en clase).*
* *Codificar el JavaScript (app.js) del lado del cliente*
* *Codificar el JavaScript (server.js) del lado del servidor*
* *Codificar el resto de archivos JavaScript del lado del servidor para gestionar las acciones sobre la tabla de tipos de proyecto.*
* *Colocar un pantallazo del contenido del archivo “index.html”. Breve explicación de lo que hace el código.*
* *Colocar un pantallazo del contenido del archivo “style.css” de lado del cliente. Breve explicación de lo que hace el código.*
* *Colocar un pantallazo del contenido del archivo JavaScript del lado del cliente (app.js). Breve explicación de lo que hace el código.*
* *Colocar un pantallazo del contenido del archivo del servidor (server.js). Breve explicación de lo que hace el código.*
* ***CAMBIAR PRESENTACIÓN DE LA PAGINA WEB (archivo HTML). Cambiar colores y presentación (CSS)***
* *Nota: recuerde que debe tener el mismo tipo de contenido proyecto abordado en clase (cliente, servidor) pero con la tabla del Caso de Estudio de la presente tarea. Es decir, se va a interactuar con una tabla de PostgreSQL a través de un esquema REST y respuestas con código HTTP*
* *Borrar las indicaciones en azul una vez completado el ítem*

**4.- Creación del proyecto Node e instalación de las dependencias iniciales**

* *Ubicarse en la carpeta del proyecto “***proyecto\_pia***”*
* *Correr la consola del sistema operativo*
* *Crear un proyecto Node : Instalar primeras dependencias :* **npm install express pg cors**
* *Colocar un pantallazo de la ejecución de la creación del proyecto en la consola*
* *Colocar un pantallazo de la ejecución de la instalación de las dependencias del proyecto en la consola*
* *Colocar un pantallazo del contenido de la carpeta después de la ejecución de estas instrucciones*
* *Centrar bien los pantallazos y agregar una breve descripción de los hace el código.*
* *Borrar las indicaciones en azul una vez completado el ítem*

****

**5.- Configuración de la base de datos**

* *Ubicar el archivo de configuración de la base de datos en el proyecto Node*
* *Parámetros de la conexión:*
  + *user: 'postgres',*
  + *host: 'localhost',*
  + *database: 'proyectos\_pia',*
  + *password: 'postgres',*
  + *port: 5432,*
* *Colocar un pantallazo del código fuente de la configuración de la Base de Datos*
* *Explicar brevemente este paso y su importancia*
* *Borrar las indicaciones en azul una vez completado el ítem*

**6.- Arranque del Servidor Node**

* *Ubicarse en la carpeta del proyecto “***proyecto\_pia***”*
* *Correr la consola del sistema operativo*
* *Arrancar el servidor Node:* **node server/server.js**
* *Colocar un pantallazo de la ejecución del servidor Node*
* *Centrar bien el pantallazo*
* *Explicar brevemente lo que está ocurriendo*
* *Borrar las indicaciones en azul una vez completado el ítem*

**7.- Arrancar la página Web**

* *Buscar a través del explorador de archivos del computador la carpeta del proyecto “***proyecto\_pia***”*
* *Abrir la carpeta cliente*
* *Abrir la página web (index.html) en el Navegador Web (preferiblemente Google Chrome)*
* *Colocar un pantallazo de la página web en el explorado*
* *Centrar bien el pantallazo*
* *Explicar brevemente lo que está ocurriendo*
* *Nota: utilizando el ratón con botón derecho sobre el archivo “index.html” puede abrirlo en el explorador*
* *Borrar las indicaciones en azul una vez completado el ítem*

**8.- Inserción de registros (POST/INSERT)**

* *Ubicarse en la página Web e intentar agregar tipos de proyecto*
* *Ejecutar una instrucción HTTP (POST)*
* *Recibir un código de operación satisfactoria*
* *Agregar los siguientes tipos de proyecto a la base de datos a través del sistema de información*
* *Código 1000. Descripción: Proyecto de Aula*
* *Código 2000. Descripción: Proyecto Integrador de Aula*
* *Código 3000. Descripción: Proyecto Externo de Aula*
* *Código 4000. Descripción: Proyecto Internacional de Aula*
* *Colocar el pantallazo del resultado del registro del tipo de proyecto con código: 1000*
* *Agregar el siguiente tipo de proyecto a la base de datos a través del sistema de información*
* *Código 1000. Descripción: Proyecto Industrial de Aula*
* *Colocar el pantallazo del resultado del intento de registro del tipo de proyecto “Proyecto Industrial de Aula”*
* *Centrar bien los pantallazos y agregar una breve descripción de los resultados*
* *Borrar las indicaciones en azul una vez completado el ítem*

**9.- Modificación de registros (PUT)/UPDATE)**

* *Ubicarse en la página Web e intentar modificar la información de un tipo de proyecto*
* *Ejecutar una instrucción HTTP (PUT)*
* *Realizar una operación satisfactoria de modificación, mostrar el pantallazo y explicar brevemente los resultados*
* *Colocar el pantallazo del resultado de la actualización satisfactoria de un registro de tipo de proyecto*
* *Realizar una operación incorrecta (a propósito/a drede) de modificación, mostrar el pantallazo y explicar brevemente los resultados*
* *Colocar el pantallazo del resultado de la actualización incorrecta de un registro de tipo de proyecto*
* *Centrar bien los pantallazos y agregar una breve descripción de los resultados*
* *Borrar las indicaciones en azul una vez completado el ítem*

**10.- Eliminación de un registro (DELETE/DELETE)**

* *Ubicarse en la página Web e intentar eliminar un tipo de proyecto*
* *Ejecutar una instrucción HTTP (DELETE)*
* *Realizar una operación satisfactoria de eliminación, mostrar el pantallazo y explicar brevemente los resultados*
* *Colocar el pantallazo del resultado de eliminación satisfactoria de un registro de tipo de proyecto*
* *Realizar una operación incorrecta (a propósito) de eliminación, mostrar el pantallazo y explicar brevemente los resultados*
* *Colocar el pantallazo del resultado de la eliminación incorrecta de un registro de tipo de proyecto*
* *Centrar bien los pantallazos y agregar una breve descripción de los resultados*
* *Borrar las indicaciones en azul una vez completado el ítem*

**11.- Consulta de registros (GET/SELECT)**

* *Ubicarse en la página Web e intentar consultar tipos de proyecto*
* *Ejecutar una instrucción HTTP (GET)*
* *Realizar una operación satisfactoria de consulta, mostrar el pantallazo y explicar brevemente los resultados*
* *Colocar el pantallazo del resultado de consulta satisfactoria de tipos de proyecto*
* *Ejecutar pgAdmin4 y realizar una consulta de los registros de la tabla de tipo de proyecto*
* *Colocar el pantallazo de la consulta SQL y el resultado en pgAdmin4*
* *Centrar bien los pantallazos y agregar una breve descripción de los resultados*
* *Borrar las indicaciones en azul una vez completado el ítem*

**12.- Conclusiones Individuales**.

* *Presentar conclusiones individuales sobre los conocimientos adquiridos, su relación con el PIA, la experiencia y competencias adquiridas; y la vinculación al futuro ejercicio profesional. Cada conclusión debe tener un mínimo de 500 palabras.*
* *Prestar especial atención a la ortografía, gramática y expresión de las ideas.*
* *Describir el grado de participación y las actividades que realizó en la Tarea*
* *Borrar las indicaciones en azul una vez completado el ítem*

**13.- Calidad del Informe de la Tarea. Rúbrica Informe**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Características del Informe** | **Puntos** | **Calificación** |
| 1 | * Presenta informe en la plantilla suministrada de forma correcta * Coloca el informe en la raíz de la carpeta TIA3 en el repositorio GIT * Se presentan todos los miembros del equipo con su nombre * Cubre la totalidad de los ítems de la tarea | **20** |  |
| 2 | * Escribe el nombre del video correctamente (colocando la letra del equipo en la “X”): ***20251-et0179-TIA3-equipo-x-informe.docx*** | **5** |  |
| 3 | * Presenta cuadros centrados, sin “dividirlos” entre saltos de página, sin distorsiones, letra ni tan grande ni tan pequeñita, con colores agradables. * Presenta figuras (pantallazos) centradas y de buena calidad | **50** |  |
| 4 | * Calidad general (tipo de letra, redacción, organización, figuras) * Redacta el informe con buena ortografía, gramática y expresión de ideas | **25** |  |
|  | **Total** | **100** |  |

**14.- Video de Sustentación**. **Rúbrica Video de Sustentación**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Características del Video de Sustentación** | **Puntos** | **Calificación** |
| **1** | * Se presentan todos de manera individual mostrando su cara, diciendo su nombre y describiendo la actividad que van a presentar * Se cubre la totalidad de los ítems de la tarea | **10** |  |
| **3** | Muestra de código en ejecución en PostgreSQL. Script SQL (pgAdmin4) | **5** |  |
| **3** | Muestra de código en ejecución del lado del cliente | **25** |  |
| **4** | Muestra de código en ejecución del lado del servidor | **25** |  |
| **5** | Describe la estructura del proyecto Node FrontEnd/BackEnd | **10** |  |
| **6** | Presenta breve conclusión individual sobre la tarea | **10** |  |
| **7** | Presenta video con calidad tanto de sonido como de visualización. Cada participante muestra su rostro claramente, se escucha bien el audio y se ve lo que presenta. | **10** |  |
| **8** | * Presenta enlace de acceso al video en el informe y en el repositorio * Escribe el nombre del video correctamente (colocando la letra del equipo en la “X”) ***20251-et0179-TIA3-equipo-X-video*** | **5** |  |
|  | Total | **100** |  |
|  | *NOTA: El estudiante que no aparezca, no tendrá evaluación de la tarea. ES OBLIGATORIA LA PARTICIPACIÓN EN EL VIDEO. Si por razones de “fuerza mayor” no puede aparecer en el video en conjunto, presentará un video individual exponiendo parte del trabajo.* | **0** | **0** |

**15.- Repositorio GIT. Rúbrica Repositorio GIT**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Características del Video de Sustentación** | **Puntos** | **Calificación** |
| **1** | Crea y presenta un repositorio con un nombre que identifique fácilmente el equipo de estudiantes | **20** |  |
| **2** | Presenta breve descripción del contenido del repositorio (Proyectos PA/PIA) | **5** |  |
| **3** | Se presentan los miembros del equipo en la primera página del repositorio | **5** |  |
| **4** | Repositorio organizado por carpetas de Tareas de esta manera  Tarea-1 o TIA-1  Tarea-2 o TIA-2  Tarea-3 o TIA-3  Tarea-4 o TIA-4  Tarea-5 o TIA-5  Tarea-5 o TIA-5 | **10** |  |
| **5** | Tarea 1 (TIA1) subida al repositorio | **10** |  |
| **6** | Tarea 2 (TIA2) subida al repositorio | **10** |  |
| **7** | Tarea 3 (TIA3) subida al repositorio (esta tarea) | **20** |  |
| **8** | Repositorio permite el acceso al público, el docente y todos los estudiantes  (modo lectura) | **5** |  |
| **9** | Cada tarea presenta un enlace al video de sustentación. | **5** |  |
| **10** | La Tarea-3 debe estar organizada en subcarpetas y archivos de esta manera:  Archivos en la raíz de la carpeta Tarea-4   * Informe * Enlace video * Subcarpetas “proyecto\_pia” dentro de la Tarea-4   + Cliente   + Servidor | **10** |  |
|  | **Total** | **100** |  |

**Rúbrica de la Tarea: Criterios de Evaluación**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ítem** | **Criterio** | | | **Peso** | **Cal** |
| **1** | **Base de Datos “proyectos” y Tabla de Tipos de Proyecto** | | | **20** |  |
| **2** | **Estructura del proyecto FrontEnd/BackEnd** | | | **30** |  |
| **3** | **Código fuente del Proyecto. Cambiar estilo Página Web** | | | **200** |  |
| **4** | **Creación del proyecto Node e instalación de las dependencias iniciales** | | | **10** |  |
| **5** | **Configuración de la base de datos** | | | **10** |  |
| **6** | **Arranque del Servidor Node** | | | **15** |  |
| **7** | **Arrancar la página Web** | | | **15** |  |
| **8** | **Mostrar ejecución operación Rest - Inserción de registros (POST)** | | | **75** |  |
| **9** | **Mostrar ejecución operación Rest- Modificación de registros (PUT)** | | | **75** |  |
| **10** | **Mostrar ejecución operación Rest - Eliminación de un registro (DELETE)** | | | **75** |  |
| **11** | **Mostrar ejecución operación Rest - Consulta de registros (GET)** | | | **75** |  |
| **12** | **Conclusiones Individuales** | | | **50** |  |
| **13** | **Calidad del Informe** | | | **100** |  |
| **14** | **Video de sustentación** | | | **200** |  |
| **15** | **Repositorio** | | | **50** |  |
|  | **NOTA** |  | **TOTAL** | **1000** |  |