**Tarea - TIA-03**

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

**EQUIPO: B**

**MIEMBROS DEL EQUIPO:**

* ESTEFANIA MORA CALDERON
* DIEGO ALEJANDRO VALENCIA PALACIO
* JUAN DAVID ARBOLEDA ARANGO
* SANTIAGO ECHAVARRIA PATIÑO

**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:**

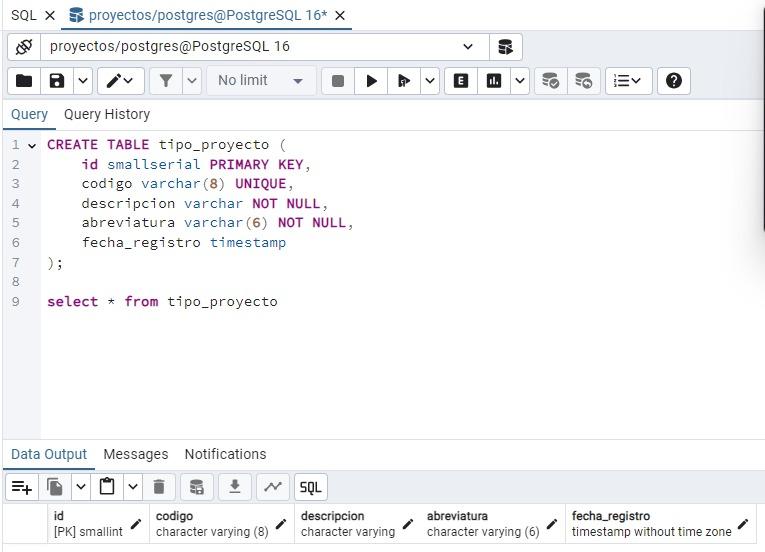
* Los resultados de la Tarea se deben subir al repositorio GIT de tareas de la asignatura en la carpeta Tarea-3 (TIA-3). Nota. Permite el acceso al público y al docente para acceder y ver el contenido.
* En el Classroom solamente colocarán UN SOLO LINK al repositorio. Lo colocará el líder del grupo

**Productos (entregables):**

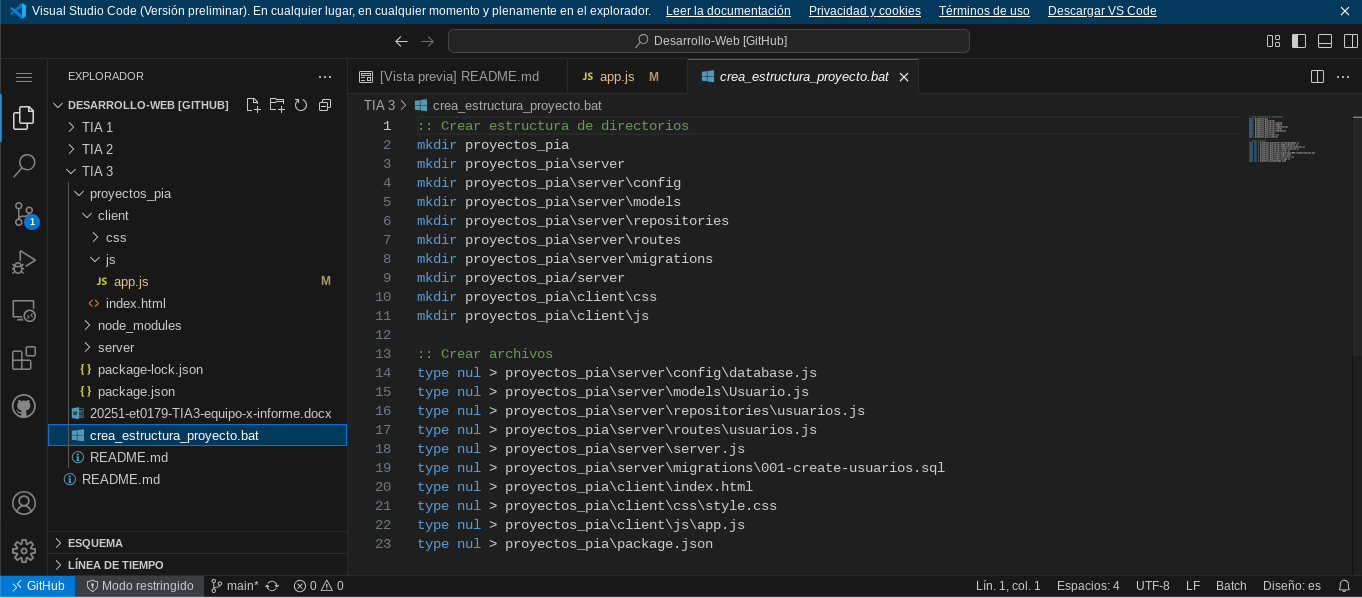
* Plantilla de Informe con los resultados
* Proyecto Node con conexión y actualización de base de datos (carpetas y sub carpetas)
* Repositorio GIT con las Tareas: TIA-1, TIA-2 y TIA-3
* Equipo de miembros en la página principal del GIT
* Breve descripción del Proyecto PIA en el GIT

**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 |  |  |  |  |



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



*Este archivo es un script en formato* ***.bat*** *(Batch) diseñado para usarse en Windows. Su propósito es crear automáticamente la estructura de carpetas y archivos básicos del proyecto proyectos\_pia. Este archivo automatiza la creación de carpetas y archivos para un proyecto de desarrollo web con una arquitectura tipo cliente-servidor.*

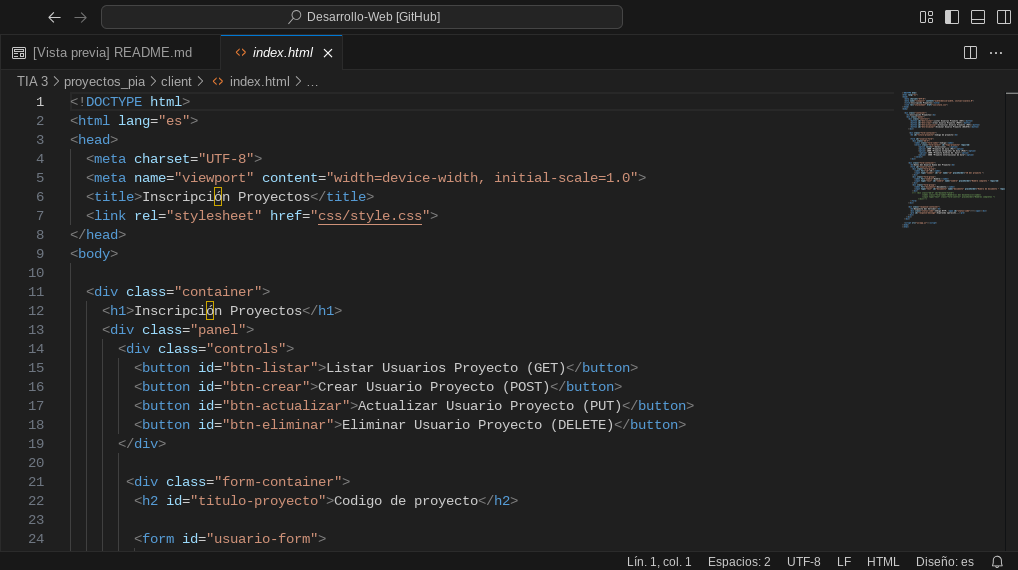
### *Estructura creada:*

1. ***Carpeta principal:***
   * *proyectos\_pia*
2. ***Servidor (server)***
   * *config\database.js: configuración de conexión a la base de datos.*
   * *models\Usuario.js: modelo de datos del usuario.*
   * *repositories\usuarios.js: funciones para acceder a la base de datos.*
   * *routes\usuarios.js: define rutas del API.*
   * *migrations\001-create-usuarios.sql: script SQL para crear la tabla de usuarios.*
   * *server.js: archivo principal del servidor.*
3. ***Cliente (client)***
   * *index.html: interfaz del usuario.*
   * *css\style.css: estilos del proyecto.*
   * *js\app.js: lógica del lado del cliente (JavaScript).*
4. ***Archivo adicional:***
   * *package.json: archivo de configuración para Node.js (dependencias, scripts, etc.).*

**3.- Código fuente del Proyecto**

1. *Escribir el código fuente de todos los archivos del proyecto*

* *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.*

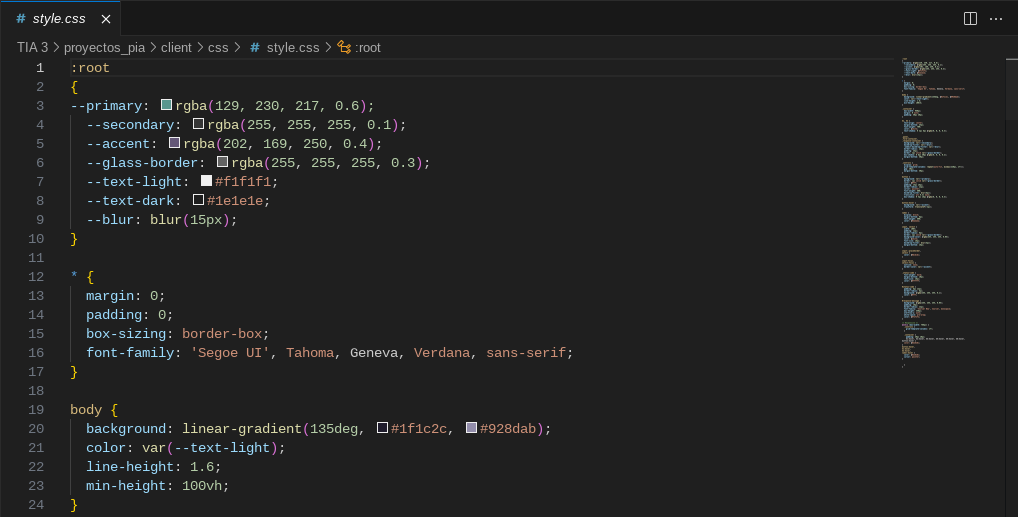


#### *Archivo HTML (index.html)*

*Este archivo contiene la estructura de la interfaz de usuario del proyecto. Incluye:*

* *Un título principal "Inscripción Proyectos".*
* *Botones para realizar operaciones con usuarios (listar, crear, actualizar, eliminar).*
* *Formularios para ingresar los datos del usuario y seleccionar el tipo de proyecto.*
* *Un área para mostrar la respuesta del servidor (código HTTP y mensaje).*

*Su función principal es ofrecer una interfaz sencilla para interactuar con el sistema cliente-servidor.*

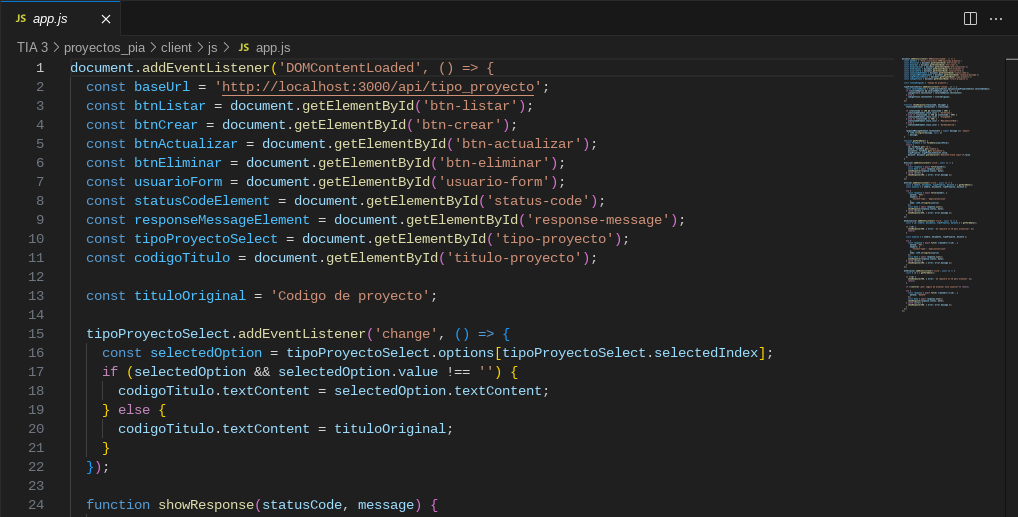


#### *Archivo CSS (style.css)*

*Este archivo contiene los estilos visuales de la interfaz. Algunas de sus funciones son:*

* *Definir márgenes, fuentes, colores y diseño de los elementos.*
* *Cambiar el color de botones o títulos cuando el usuario pasa el cursor por encima (hover).*
* *Mejorar la presentación general para que la aplicación sea más clara y estética.*

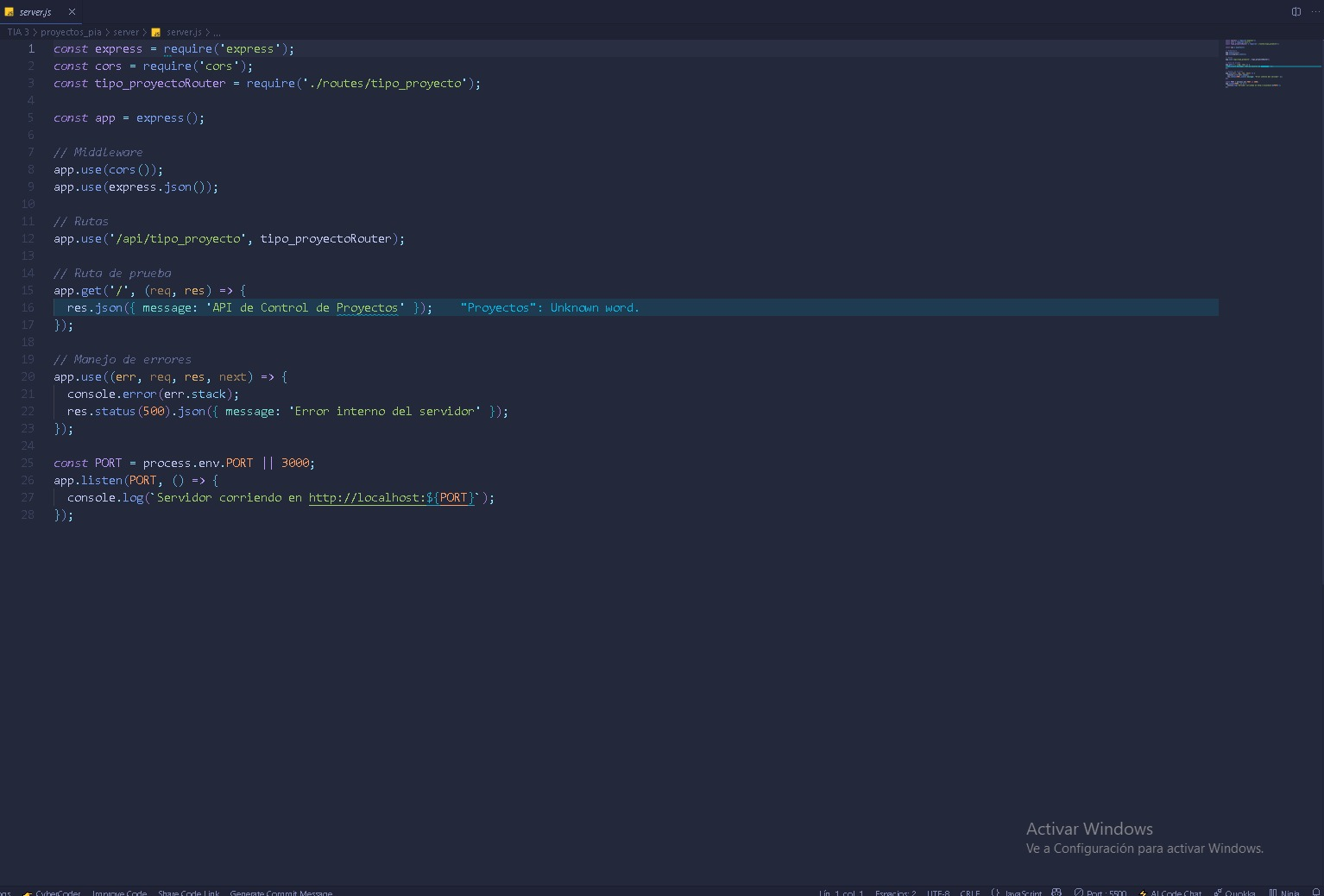
*En resumen, permite que la página web se vea ordenada, amigable y profesional*

***JavaScript (app.js)***

*Este archivo contiene la lógica de la aplicación del lado del cliente. Hace lo siguiente:*

* *Captura eventos cuando el usuario hace clic en los botones (listar, crear, actualizar, eliminar).*
* *Toma los datos del formulario y realiza peticiones HTTP (GET, POST, PUT, DELETE) al servidor.*
* *Muestra la respuesta del servidor (estado y mensaje) en la página.*
* *Cambia dinámicamente el título según el tipo de proyecto seleccionado.*

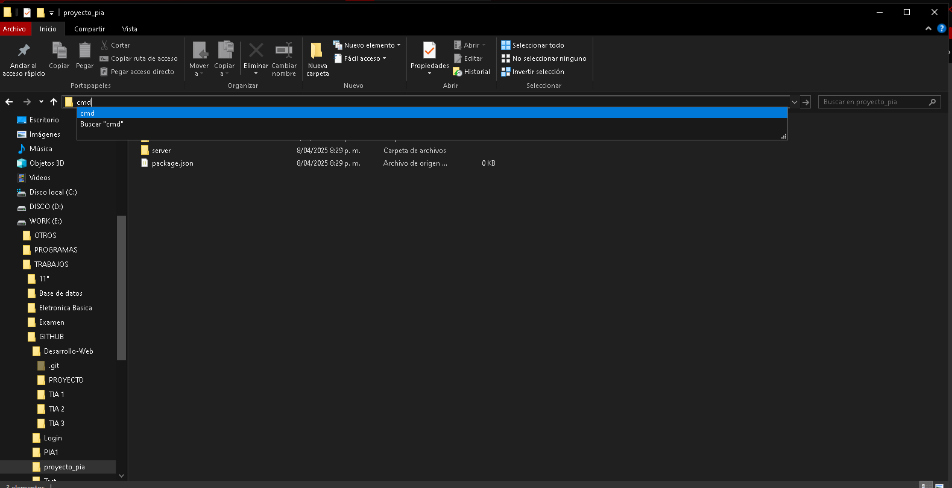
*Esencialmente, conecta la interfaz del usuario con la lógica del servidor para que la aplicación funcione de forma interactiva.*

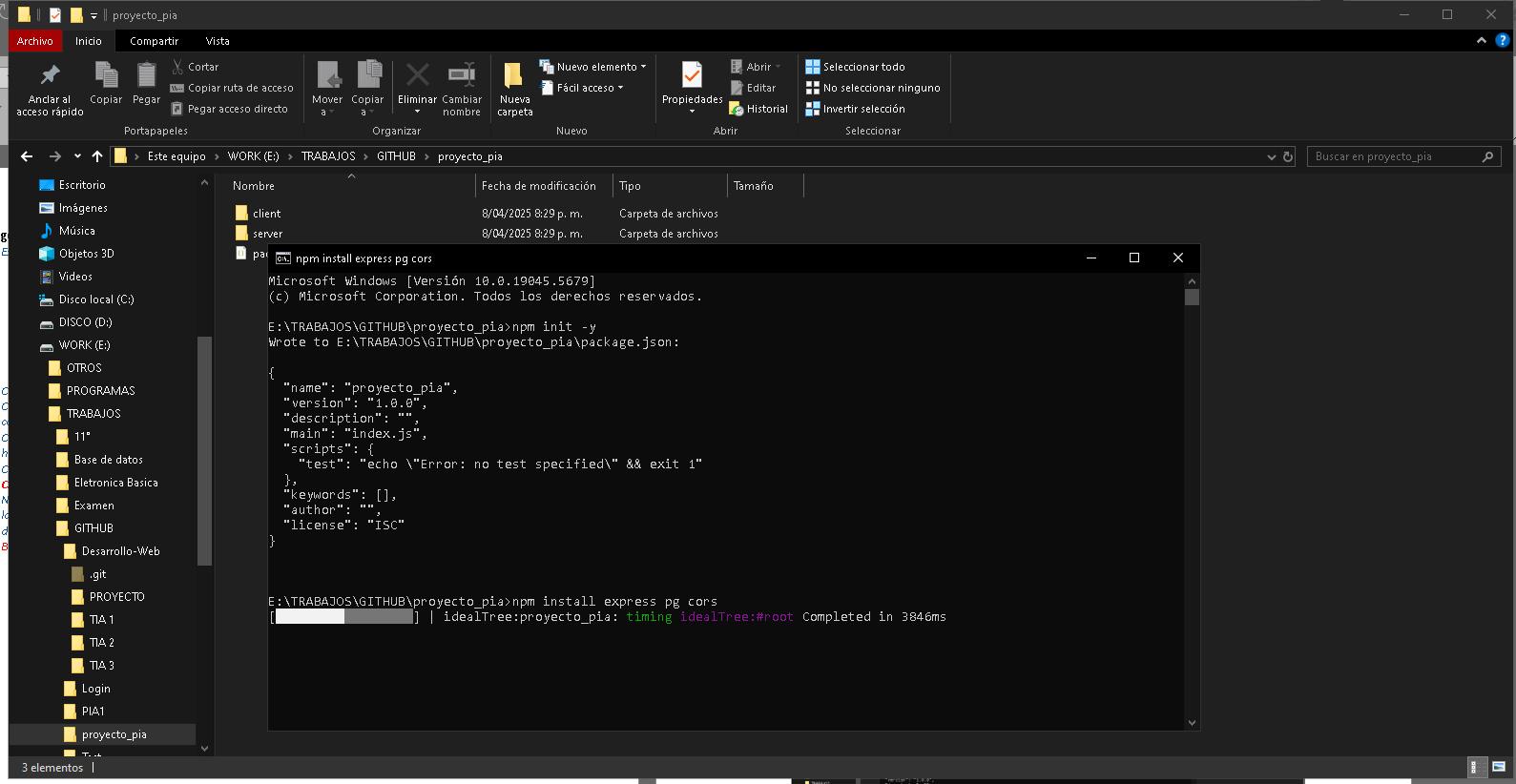


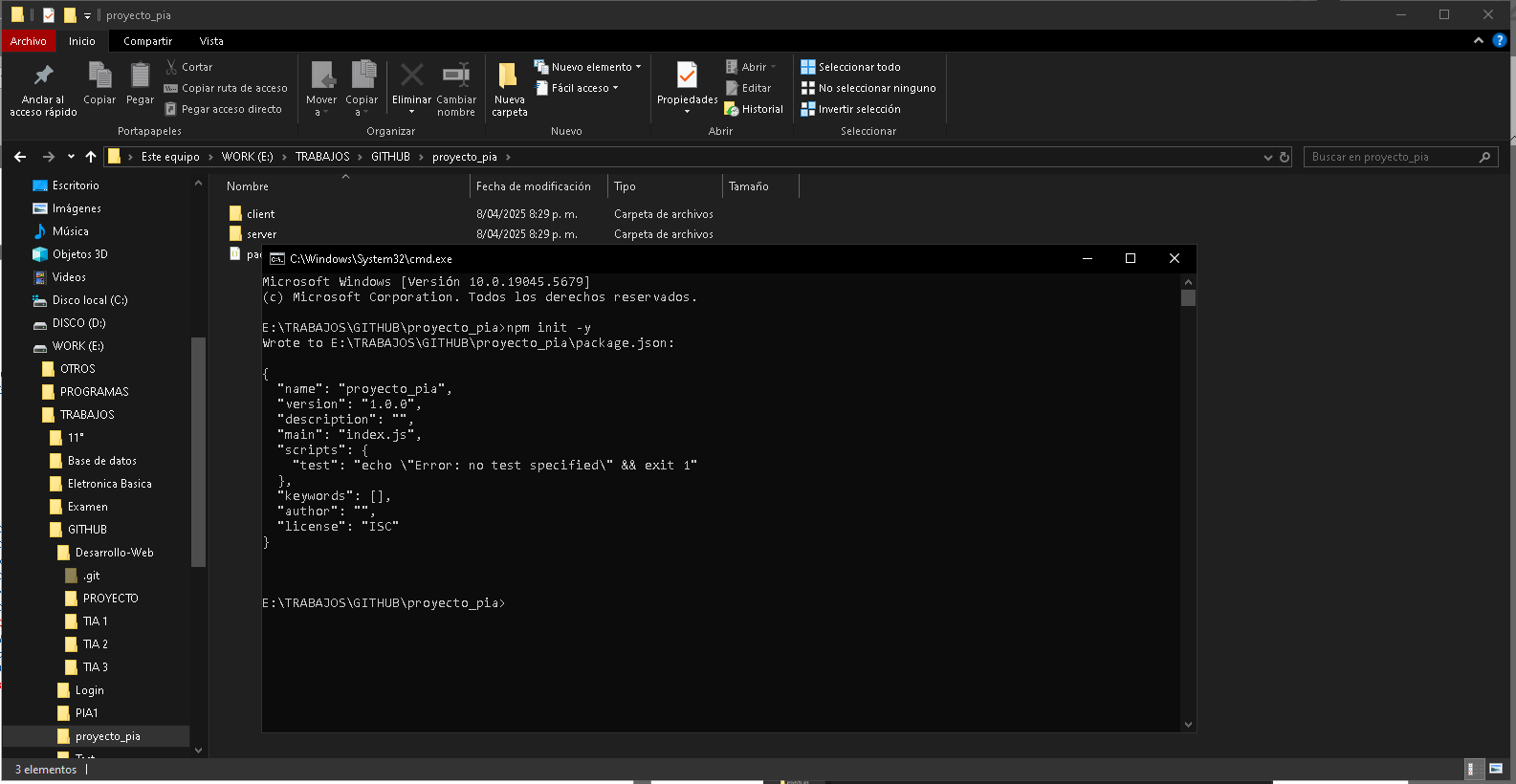
*El archivo server.js se encarga de crear y configurar un servidor web con Express, habilitar CORS, interpretar JSON, definir rutas para la API de "tipo de proyecto", y establecer un manejo básico de errores. Es el punto de entrada principal del backend del proyecto.*

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

1. *Centrar bien los pantallazos y agregar una breve descripción de los hace el código.*
2. *Borrar las indicaciones en azul una vez completado el ítem*





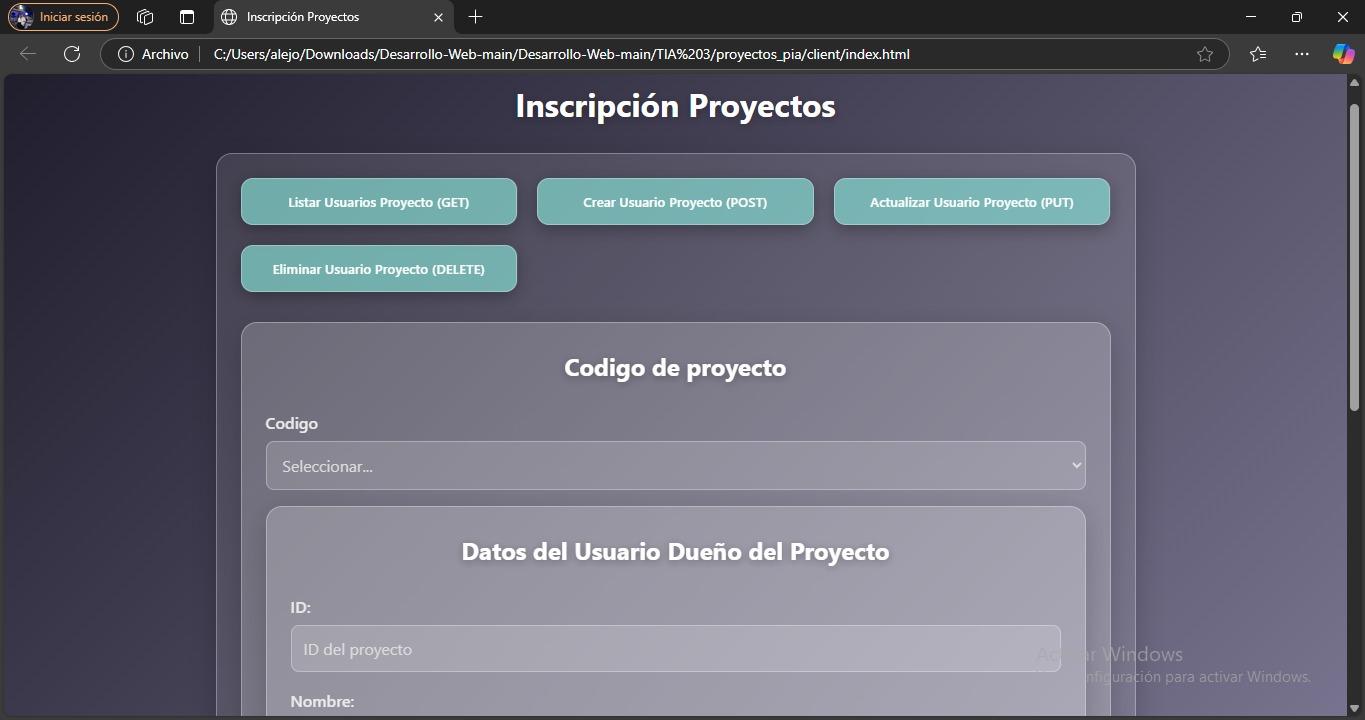


**5.- Configuración de la base de datos**

**6.- Arranque del Servidor Node**

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

**7.- Arrancar la página Web**



se está ejecutando correctamente la página web en el navegador y se ven reflejadas todas las configuraciones del código.

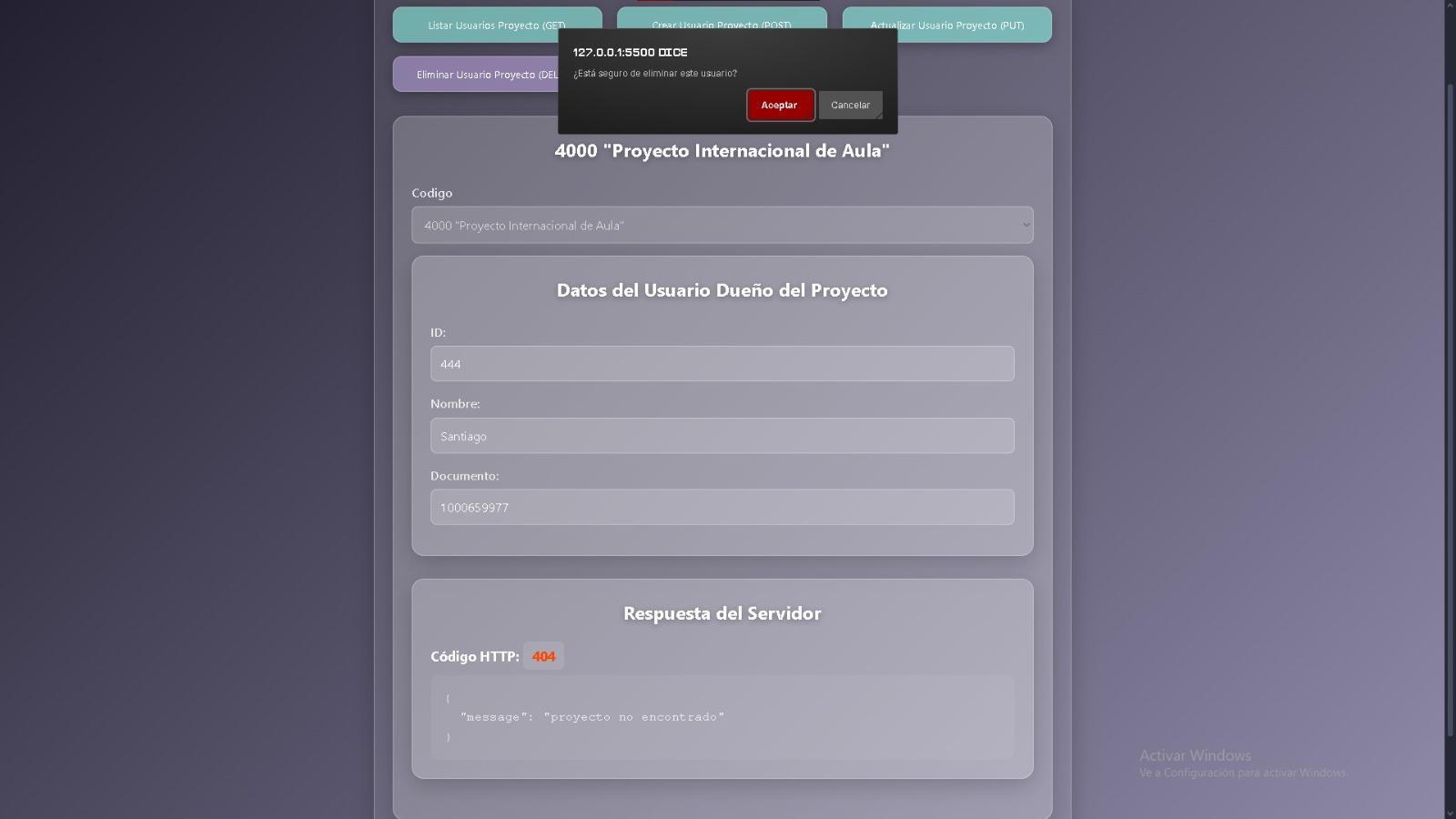
**8.- Inserción de registros (POST/INSERT)**

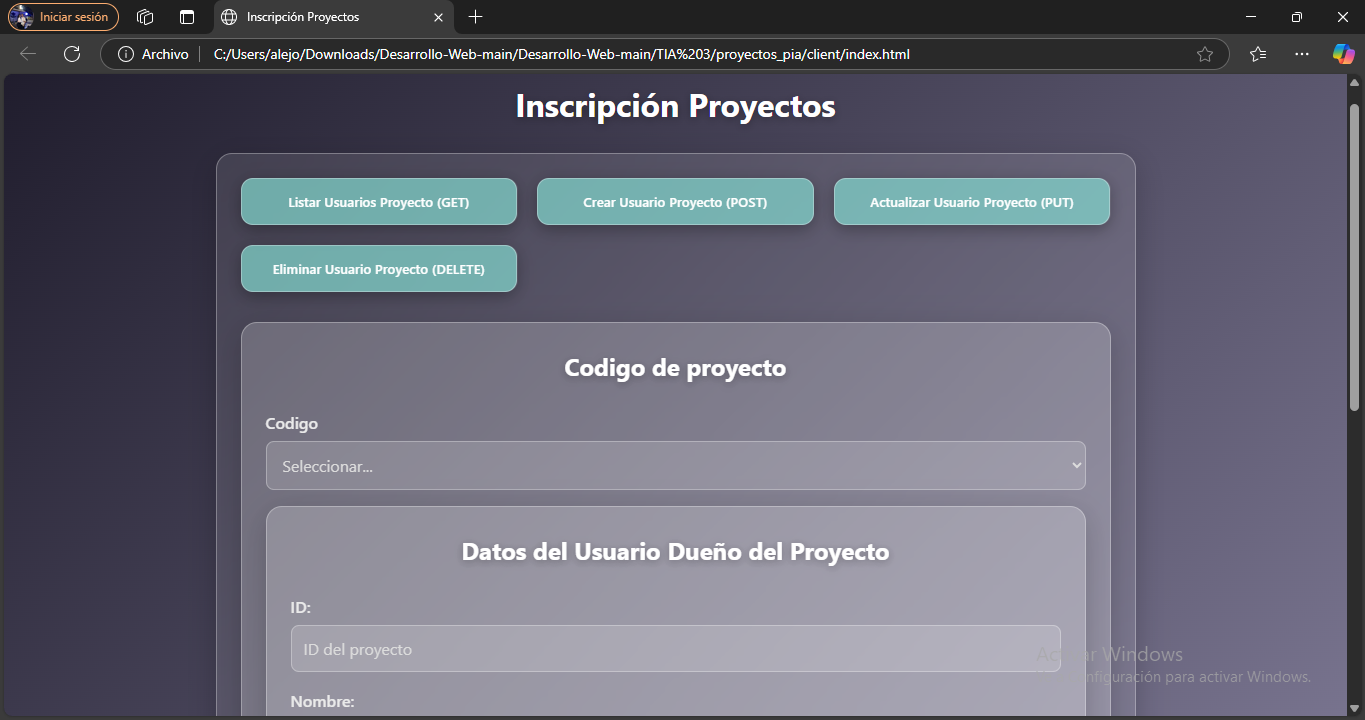


**9.- Modificación de registros (PUT)/UPDATE)**

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

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





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

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

**12.- Conclusiones Individuales**.

**Conclusión Individual - Diego**

Durante el desarrollo del Proyecto Integrador de Aula (PIA), una de mis principales responsabilidades fue la creación de la base de datos y realizar modificaciones en la hoja de estilos (CSS). Esta experiencia me permitió consolidar y aplicar los conocimientos adquiridos en el manejo de bases de datos, especialmente en cuanto a la definición de esquemas, relaciones y migraciones iniciales. Trabajar con la base de datos y estructuras de datos me ayudó a comprender de manera más profunda cómo la información se organiza, almacena y se conecta en un entorno de aplicación cliente-servidor.

La intervención en el CSS también fue una oportunidad enriquecedora. Aunque muchas veces se subestima la importancia del diseño visual en una aplicación, descubrí que una interfaz clara y atractiva mejora significativamente la experiencia del usuario. Las modificaciones en estilos como colores, espaciados y comportamiento al pasar el cursor me ayudaron a interiorizar buenas prácticas de diseño centrado en el usuario.

Esta experiencia me mostró cuán esencial es tener una visión integral del desarrollo web, y cómo el trabajo colaborativo permite construir proyectos funcionales y organizados. Aunque el proyecto no se completó del todo, fue un ejercicio realista que expuso nuestras fortalezas y debilidades. Aprendí que es fundamental complementar la teoría con la práctica y que es necesario dominar cada una de las capas del proyecto para llegar a una solución robusta.

Esta experiencia se vincula directamente con mi futuro ejercicio profesional, ya que me permitió entender que el desarrollo de software requiere precisión técnica, capacidad para trabajar en equipo, y adaptabilidad frente a problemas. La construcción de bases de datos es una habilidad esencial en muchos roles técnicos, y continuaré reforzándola para asumir responsabilidades más complejas.

**Conclusión Individual - Santiago**

Mi participación en el proyecto PIA se centró en la configuración del servidor. Esta tarea incluyó el montaje de un entorno funcional con rutas, controladores y el establecimiento de conexiones con la base de datos. Este rol me permitió comprender mejor la arquitectura del desarrollo backend y su importancia en el funcionamiento fluido de una aplicación web.

A través de la experiencia, reforcé mis conocimientos en Node.js y Express, herramientas fundamentales para el desarrollo de servidores eficientes. Me enfrenté a desafíos como el manejo de errores, el uso adecuado de métodos HTTP y la validación de datos. Aunque nos encontramos con dificultades para cumplir todos los requisitos, fue una oportunidad valiosa para aprender de manera activa y colaborativa.

La conexión con el PIA se evidencia en cómo se integraron nuestras habilidades individuales para construir un producto funcional. Este proyecto reflejó cómo la colaboración y la distribución de responsabilidades permite alcanzar objetivos comunes, a pesar de las limitaciones técnicas que puedan surgir.

Esta experiencia me reafirma que deseo especializarme en desarrollo backend. Aprendí que el conocimiento técnico debe ir acompañado de buenas prácticas, como el control de versiones, la documentación clara y la comunicación efectiva con los compañeros. En el futuro, aplicaré estas competencias para abordar retos más complejos y ser parte de proyectos con impacto real.

**Conclusión Individual - Juan David**

Durante este proyecto, me encargué de construir la estructura principal del archivo HTML, así como de corregir errores de ortografía y gramática que encontré en el código. Esto me permitió afianzar mi comprensión sobre la importancia de la estructura semántica en el desarrollo web y cómo una buena organización del código facilita el trabajo en equipo.

El trabajo en HTML no solo es la base visual y estructural de una aplicación, sino que también influye en la accesibilidad y mantenibilidad del código. Al centrarme en la corrección de detalles gramaticales, también reflexioné sobre la importancia de la comunicación clara y efectiva dentro del código fuente.

Relacionando esto con el PIA, comprendí que cada aporte, por más pequeño que parezca, es valioso para lograr un producto de calidad. Nuestra experiencia demostró que, aunque nos faltaron ciertos conocimientos para lograr todos los objetivos, logramos integrar partes importantes del proyecto trabajando de forma cooperativa.

En mi futuro profesional, esta experiencia me impulsa a seguir perfeccionando mis habilidades de desarrollo front-end y a trabajar en la mejora continua de la presentación y usabilidad de las interfaces. Además, aprendí que el trabajo detallado y ordenado tiene un gran impacto en el desarrollo de software profesional.

**Conclusión Individual - Estefanía**

Durante el desarrollo del proyecto, mi participación se centró en la creación del archivo JavaScript del lado del cliente, la realización de modificaciones necesarias en el HTML, y ajustes en el CSS para mejorar la interacción visual. Adicionalmente, asumí la responsabilidad de redactar el informe final del proyecto, organizando y documentando cada parte del proceso realizado por el equipo. Esta experiencia fue enriquecedora en muchos niveles, especialmente porque me permitió aplicar, afianzar y reconocer la importancia de los conceptos teóricos vistos en clase, así como también descubrir aspectos que debemos reforzar a nivel práctico.

Uno de los mayores aprendizajes fue la vinculación directa entre el HTML y JavaScript a través de los identificadores (ID), lo cual es esencial para que ciertas funciones se ejecuten correctamente. Este detalle, aunque puede parecer menor, fue fundamental para el funcionamiento dinámico del formulario, ya que permitió cambiar de manera automática el título del tipo de proyecto seleccionado. Entender esa conexión me permitió mejorar mis habilidades de integración entre las capas de presentación (HTML), estilo (CSS) y lógica (JavaScript).

En cuanto al CSS, implementé una funcionalidad que permite que al pasar el cursor sobre los títulos o botones, estos cambien de color. Esto le aportó un nivel básico de interactividad a la interfaz, haciendo que fuera más amigable y moderna para el usuario. Aunque son detalles pequeños, generan un impacto visual que puede marcar la diferencia en la experiencia de uso.

También fue valioso trabajar en el JavaScript para mostrar la respuesta del servidor con diferentes colores según el código de estado HTTP, lo que facilita la comprensión de los resultados de cada operación (GET, POST, PUT, DELETE). Fue un reto interesante, especialmente al estructurar las funciones de manejo de eventos y organizar el flujo de datos del formulario hacia el servidor. Estas tareas me ayudaron a mejorar mis competencias de lógica de programación, manipulación del DOM, y uso de fetch para las peticiones asíncronas.

Sin embargo, una de las mayores conclusiones es que aunque logramos desarrollar parte importante del proyecto, no se pudo cumplir a cabalidad con todos los requerimientos planteados por el docente. Esto nos lleva a reflexionar sobre la necesidad de fortalecer tanto nuestros conocimientos teóricos como nuestras habilidades prácticas. En muchos momentos sentimos que nos faltaban bases más sólidas para enfrentar algunos retos técnicos, lo cual es entendible dado que estamos en proceso de formación.

En relación con el Proyecto Integrador de Aula (PIA), considero que este ejercicio fue coherente con su propósito, ya que nos obligó a unir conocimientos de distintas áreas: bases de datos, diseño de interfaces, programación cliente-servidor, y trabajo colaborativo. El hecho de tener que pensar en la funcionalidad completa de una aplicación nos prepara para enfrentar proyectos reales en el futuro, donde no basta con saber solo una parte, sino que es necesario tener una visión integral.

Finalmente, esta experiencia permite combinar la lógica, la creatividad y la organización. Puse en práctica las buenas prácticas de codificación, y diseño centrado en el usuario. En mi futuro ejercicio profesional, estoy segura de que este tipo de desafíos me serán de utilidad, ya que fortalecen competencias clave como la resolución de problemas, el trabajo en equipo, la comunicación técnica y la autogestión del aprendizaje.

**Enlace del vídeo** [**https://youtu.be/NYxnyItlfAc**](https://youtu.be/NYxnyItlfAc)

**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** |  |