Diseño del Software Educativo

Docente

Franklin Eduardo Martínez Ávila

Estudiantes

Laura Marchena Madera

Marianela Medrano Gómez

Eduardo José Robles Carmona

Juan José Vergara Oliveros

Universidad de Córdoba

Facultad de Educación y Ciencias Humanas

Departamento de Informática Educativa

Licenciatura en Informática con Énfasis en Medios Audiovisuales

Asignatura Diseño y Desarrollo de Software Educativo II

Montería – Córdoba

2023-1

Inicialmente, se muestra el proceso para la elaboración de un software como solución a un problema educativo. La necesidad identificada en la Institución Educativa Santa María Goretti es la ausencia de proyectos ambientales, ya que, se evidencia la contaminación ambiental y la falta de sentido ambiental en esta, por ende, los estudiantes carecen de conocimientos de cómo clasificar los residuos sólidos y sobre el cuidado del medio ambiente. Por lo tanto, se busca diseñar, crear e implementar un software educativo donde los estudiantes tengan acceso a los contenidos y puedan interactuar con las diferentes actividades relacionadas con el tema, además, de ser evaluados para contribuir al mejoramiento de dicha problemática en la institución educativa y de esta manera mitigar esta.

La guía a seguir para la implementación del software educativo es el modelo de MODESEC basado en competencias, el cual Caro y Toscano (2009) presentan una visión integral del desarrollo de estas aplicaciones mediante la combinación de componentes pedagógicos, didácticos, multimediales y de ingeniería de software, así mismo, se crearon diagramas de entidad-relación, relacional y casos de usos para establecer la organización y conexión entre los diferentes componentes del sistema, por otro lado, se diseñó un prototipo donde se explicó la señalización y uso de los iconos presentados, adicionalmente se encuentran archivos en JSON, la base de datos de MYSQL y MongoDB como fase final para tener el software educativo terminado y listo para su implementación.

*Formato 1. Ejemplo de Formato de Análisis de la necesidad educativa.*

|  |  |  |
| --- | --- | --- |
| **ANÁLISIS DE LA NECESIDAD EDUCATIVA** | | |
| **Tipo: Sentida** | | La necesidad que se va a trabajar es de tipología sentida, ya que, la problemática a tratar se puede identificar de manera subjetiva en muchas de las Instituciones Educativas de Colombia.  Por lo tanto, se implementará un software educativo sobre el uso y clasificación de los residuos sólidos debido a la observación empírica de la falta de conocimientos y prácticas inadecuadas en varias Instituciones Educativas, la relevancia social y ambiental del tema, el potencial educativo del software y la posibilidad de replicar y adaptar la solución en diferentes contextos educativos. |
| **Identificación del aprendizaje ideal** | | |
| Los estudiantes deben conocer e identificar:   * Uso y clasificación de los residuos orgánicos e inorgánicos. * Datos sobre el impacto Ambiental, causas y consecuencias * Aspectos globales, nacionales y locales relacionados con la manipulación de los residuos. | | |
| **Población** | **Rango de edad:** Entre 11 y 13 años. | |
| **Escolaridad**: Grado 6° | |
| **Conocimiento que posee:** El estudiante conoce los residuos, (cáscaras de fruta o verdura, restos de comida, plástico, papel, cartón y vidrios). | |
| **Intereses y expectativas: (Población)**  Aprender a utilizar el software de manera correcta y que contribuya al mejoramiento de la problemática identificada, generando en los estudiantes un aprendizaje respecto al manejo de los residuos. | |
| **Intereses y expectativas: (Creadores)**  Desarrollar motivación en los estudiantes hacia la utilización de un software educativo para que conozcan cómo se deben clasificar los residuos orgánicos e inorgánicos, y que de esa manera reflejar un aprendizaje de manera dinámica. | |
| **Área de formación** | **Área del saber:** Ciencias Naturales | |
| **Área de contenido:** Clasificación de los residuos sólidos. | |
| **Estado actual** | Diagnóstico:  Teniendo en cuenta la necesidad al interior de las instituciones con respecto a la clasificación de residuos sólidos, se han podido establecer que dentro del plan de clases de Ciencias Naturales del grado 6, haciendo una comparación con los estándares básicos de competencias en esta área y en este grado, que expide el Ministerio de Educación Nacional donde afirma en la Guía No. 7 Formar en Ciencias: ¡el desafío! que el estudiante debe “Diseñar y aplicar estrategias para el manejo de basuras en mi colegio”, “Cuidar el entorno que lo rodea y manejar responsablemente las basuras”. Pero estas competencias no se están llevando a cabo al 100% en las instituciones, teniendo en cuenta lo anterior cabe anotar que esta temática se está dando de una manera muy básica y hace falta profundizar en la cultura de la clasificación de los residuos sólidos, para formar personas competentes en el cuidado del planeta. | |
| **Necesidad** | La necesidad educativa se refleja en la falta de conocimientos relacionados con el impacto ambiental generado por el mal manejo de los residuos orgánicos e inorgánicos producidos por el hombre, y en el cual los estudiantes desconocen diferentes aspectos de esta situación, como lo es, la clasificación de residuos, los contenedores de clasificación, aspectos ambientales, entre otros. | |
| **Causas** | * Ausencia de proyectos que trabajen el buen uso de los residuos y clasificación de estos. * Falta de capacitación a los docentes respecto a la temática y motivación para promover dicho tema. * Falta de interés por parte de los estudiantes para trabajar en dicho problema * Ausencia de avisos y contenedores, los cuales permiten clasificar de manera correcta los residuos. | |
| **Soluciones** | * Crear un proyecto mediado por un Software educativo el cual trabaje el buen uso de los residuos y la clasificación de estos. * Promover espacios en los cuales se capaciten a los docentes respecto a la temática. * Generar interés en los estudiantes para aprender sobre la clasificación de los residuos sólidos, de manera innovadora, creativa, lúdica y didáctica. | |
| **Conocimientos y habilidades que**  **debe tener el**  **estudiante** | **Preconceptos**: Que tengan conocimientos básicos en uso de residuos e incluye ejemplos prácticos en la clasificación de residuos. | |
| **Precondiciones:** Interpretación de residuos, habilidades básicas en la clasificación de residuos, comprensión básica de conceptos y relaciona las necesidades del ambiente en el uso adecuado de los residuos. | |
| **Justificación**  El mal manejo de los residuos es una de las principales causas de impacto ambiental en la actualidad. Cuando los residuos no se manejan adecuadamente, pueden tener varios efectos negativos en el medio ambiente, entre los cuales encontramos; la contaminación de suelo debido al vertido inadecuado de residuos, especialmente de productos químicos y desechos tóxicos. La contaminación del agua que destruye la vida marina. La contaminación del aire ocasionada por la quema de residuos, especialmente la quema de desechos sólidos. De igual forma, el impacto en la vida silvestre debido a la alteración de los ecosistemas por la cantidad de residuos presentes en ellos.  El mal manejo de residuos puede tener varias causas y consecuencias, las cuales varían según el contexto. Entre las causas más comunes encontramos: la falta de infraestructura adecuada que incluye los sistemas de recolección, instalaciones de tratamiento y disposición final de residuos. La Insuficiente educación y conciencia ambiental, ya que la falta de conocimiento sobre el manejo adecuado de los residuos puede llevar a la adopción de métodos inadecuados. Otra de las causas con mayor impacto es el comportamiento cultural y hábitos como la falta de una cultura de reciclaje y la falta de conciencia ambiental. En el ámbito gubernamental, existe la falta de regulaciones y cumplimiento ambientales y normativas relacionadas con el manejo de residuos, sin leyes o regulaciones claras para guiar y controlar el manejo de los residuos, las empresas y los individuos pueden no sentir la necesidad o la obligación de hacerlo correctamente. Todas ellas conllevan al aumento en la generación de residuos, y teniendo en cuenta el potencial crecimiento de la población y el aumento del consumo diario a nivel mundial, pueden tener consecuencias graves como llevar al aumento significativo en la generación de residuos. Si no se implementan medidas adecuadas para gestionar este aumento en la cantidad de desechos generados, puede resultar en una mayor presión sobre los sistemas existentes y un mayor riesgo de mal manejo.  Para mitigar este impacto se propone implementar un software educativo para la clasificación de los residuos, ya que, mediante el uso de este se puede enseñar a los estudiantes la importancia de la clasificación correcta y las consecuencias de una gestión inadecuada de los residuos. Esto fomenta una mayor conciencia ambiental y promueve prácticas sostenibles. De igual forma, el uso de este software proporciona a los alumnos información precisa y detallada sobre cómo clasificar correctamente los diferentes tipos de residuos. Esto ayuda a los estudiantes a comprender las características de cada tipo de residuo y a identificar correctamente los contenedores, y el resultado de esto, es la mejora y eficiencia en la clasificación de residuos y reducción de los errores.  Por otro lado, implementar un software educativo facilita el aprendizaje interactivo, ya que, incluye elementos visuales, juegos y actividades prácticas que mantienen a los estudiantes comprometidos y motivados. Esta forma de aprendizaje activo es más efectiva y memorable que simplemente leer o escuchar información teórica. De la misma manera, el software educativo brinda accesibilidad y disponibilidad en diferentes dispositivos, como computadoras, tabletas o teléfonos móviles, lo que permite que los alumnos aprendan en cualquier momento y en cualquier lugar. Esto es especialmente importante en entornos educativos, donde los estudiantes pueden acceder al software tanto en el aula como en sus hogares.  Finalmente, la implementación del software educativo para la clasificación de los residuos tiene el potencial de mejorar la conciencia ambiental, además de promover la eficiencia en la clasificación, facilitar el aprendizaje interactivo, garantizar la accesibilidad y fomentar la participación de los estudiantes y demás miembros de la comunidad educativa. Todo esto contribuye a una gestión más efectiva de los residuos y a la preservación del medio ambiente para las generaciones futuras. | | |

*Formato 2. Tabla de roles*

|  |  |
| --- | --- |
| **Programador** | **Juan José Vergara Oliveros.** |
| **Experto pedagógico** | **Laura Marchena Madera.** |
| **Diseñador gráfico** | **Eduardo José Robles Carmona.** |
| **Evaluador** | **Marianela Medrano Gómez.** |

*Formato 3. Formulario de desarrollo de actividades*

****

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FORMULARIO DE REGISTROS DE TIEMPO** | | | | | | | | |
| **Fecha** | **Hora** | | **Tiempo de**  **interrupción** | **Tiempo** | **Descripción de**  **la actividad** | **Comentarios** | **Completado** | **Unidades** |
|  | **Inicio** | **Fin** |  |  |  |  |  |  |
|  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

*Formato 4. Ejemplo de Formato Diseño de fines educativos*

|  |  |
| --- | --- |
| **DISEÑO DE FINES EDUCATIVOS** | |
| ***Objetivos de aprendizaje*** | **Objetivo general** |
| Diseñar e implementar un software educativo como ayuda didáctica del área de ciencias naturales en la temática de la clasificación de los residuos sólidos con el fin de promover la conciencia ambiental en estudiantes de sexto grado. |
| **Objetivos específicos** |
| * Conocer el impacto ambiental por el mal manejo de los residuos. * Aprender a clasificar los residuos en los distintos contenedores. * Informar a los estudiantes las causas del buen uso de los residuos |
| ***Dimensiones*** | * Capaz de relacionar, actuar y transformar la realidad de su entorno * Capaz de explicar el impacto que tiene la no clasificación de residuos sólidos. * Capaz de promover una conducta razonable y una cooperación eficaz entre los individuos respecto al tema. |
| ***Valores*** | El presente proyecto puede fortalecer:   1. El trabajo grupal en los estudiantes. 2. Capacidad reflexiva sobre la clasificación de residuos. 3. Creatividad para identificar los tipos de contenedores. 4. Capacidad para reconocer los efectos negativos del mal manejo de los residuos. 5. Concientización de las problemáticas que generan los residuos sólidos. |

*Formato 5. Ejemplo de Formato de competencias.*

|  |  |
| --- | --- |
| **Competencia 1** | **Tipo Cognitiva** |
| **Objetivos** | **Norma** |
| **1: Enunciado** | **1: Contexto** |
| Reconocer la dinámica de un ecosistema teniendo en cuenta las necesidades del ambiente y clasificar los residuos acordes a los códigos de colores. | El estudiante debe comprender y clasificar los residuos teniendo en cuenta los códigos de colores. |
| **2: Recursos** |
| * PC * Software * videos * Diapositivas |
| **2: Elementos** | **3: Evidencias** |
| * Analizó el ambiente y aplicó la utilización del código de colores para los residuos. * Represento los residuos orgánicos e inorgánicos y explico su función. | Evaluación de conocimientos a partir de la interpretación de las temáticas que contiene el software. |
| **Conceptos** | |
| * **Impacto ambiental por el mal manejo de los residuos:** Efectos, causas. * **Residuos:** Tipos de residuos. * **Clasificación de los residuos:** Contenedores, rojo, verde, azul, gris, entre otros**.** * **Buen uso de los residuos:** Ventajas, impacto favorable, tips para el cuidado del ambiente. | |
| **Habilidades y destrezas** | |
| * **Intelectual**: tener manejo de diferentes conceptos relacionados con la temática de la clasificación de los residuos y conocer los códigos de colores, manejo básico y utilización del computador. * **Social**: capacidad de estar en grupos y de explorar e interactuar escuchando opiniones y sugerencias. * **Físicas**: tener unas condiciones físicas y mentales normales. | |

*Formato 6. Vista parcial de la matriz del diseño de contenidos*

|  |  |  |  |
| --- | --- | --- | --- |
| **CONCEPTOS DE LAS COMPETENCIAS** | | | |
| **Conceptos** | **Características** | | **Definición** |
| **Impacto ambiental** | * Desaparición de especies animales y vegetales. * Cambios sobre el modelo de vida humano. * Cambios en la naturaleza de las sustancias que hacen vida en el medio ambiente, tanto en el suelo, el agua o el aire. | | El impacto ambiental abarca los distintos efectos que la actividad humana y el modelo de vida humano desatan sobre el medio ambiente. Sus formas pueden ir  desde la deforestación, la destrucción de los suelos, los derrames petroleros, entre otros. En las instituciones se pueden observar a partir del mal uso de los  recipientes de separación de residuos. |
| **Residuos** | * Residuos domésticos, generados en los hogares. * Residuos comerciales, propios del comercio. * Residuos industriales, por la actividad industrial. * Residuos peligrosos, nocivos para el ser humano. * Residuos inertes, no experimentan transformaciones físico-químicas o biológicas significativas. * Residuos no peligrosos, no presentan características de peligrosidad. * Residuos biodegradables, procedentes de hogares, restaurantes, establecimientos de venta, procedentes de plantas y alimentos. | | Todo material considerado como desecho y que se necesita eliminar. La eliminación tiene como fin evitar problemas sanitarios o medioambientales, pero también y dada la escasez de materias primas y su agotamiento de recuperar todo aquello que se pueda reutilizar. |
| **Mal manejo de los residuos** | * Contaminación del aire. * Contaminación del agua. * Degradación de los suelos. * Alteración de los ecosistemas. | | El mal manejo de los residuos consiste en no clasificar de manera correcta dichos residuos de tal manera que se puedan reutilizar, reciclar y crear materia prima. Una gestión inadecuada de los residuos puede tener consecuencias negativas. Por un lado, puede afectar la salud, también puede afectar negativamente al medio ambiente: la estética urbana, los paisajes naturales y sus especies, y la contaminación del agua, suelo y aire. |
| **Clasificación de los residuos** |  | * Utilización de los contenedores. * Buen uso de los códigos de colores. * Contenedor azul * Contenedor verde * Contenedor rojo * Contenedor gris | La clasificación de residuos orgánicos e inorgánicos se especifica bajo ciertos códigos de colores para determinar que cada cosa esté en su lugar. Es muy importante conocer estos códigos para fomentar la cultura de la limpieza y ayudar al medio ambiente con la separación de los residuos. Los códigos de colores dependen de la zona o área en la cual nos encontremos. En nuestra zona para clasificar nuestros residuos encontramos contenedores verdes, azules, grises y en algunas ocasiones rojos. |
| **Buen uso de los residuos** |  | * Disminución del uso de materia prima, se reduce el consumo de recursos naturales * Ahorro de energía, al clasificar los residuos se disminuye este consumo, y se genera un ahorro de energía. * disminución del riesgo de enfermedades, al clasificarlas evitamos muchos de estos incidentes. * Disminución del riesgo de muerte de animales, el beneficio ambiental que generamos, al clasificar los desechos, como el plástico, papel y cartón evitan que muchos animales mueran. | Cuando se hace un buen uso de los residuos la ciudad, localidad o zona mejora, se evitan enfermedades, respiramos un mejor aire, y la movilidad se agiliza, entre muchos otros beneficios. Por eso se debe saber que separar los residuos en contenedores diferente ayuda a que llegue menos cantidad al relleno sanitario. Por ende, todos debemos ser conscientes de que deben guardarse en los contenedores adecuados para que entre todos ayudemos a cuidar el medio ambiente de nuestra localidad. |

*Formato 7. Ejemplo del diseño de contenidos.*

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***MODELO PEDAGÓGICO CONSTRUCTIVISTA*** | | |
| ***Bases conceptuales*** | | ***Características*** |
| Se concibe el aprendizaje como resultado de un proceso de construcción personal-colectiva de los nuevos conocimientos, actitudes y vida, a partir de los ya existentes y en cooperación con los compañeros y el docente. Este modelo se realiza por medio de diferentes estrategias pedagógicas como el aprendizaje colaborativo, el aprendizaje en contexto, el aprendizaje significativo y la educación basada en competencias, entre otras que han ido surgiendo gracias a la intervención de las tecnologías de la información y la comunicación en la educación. En todas estas estrategias, los roles de los actores son condición necesaria para obtener resultados de aprendizaje creativos, innovadores y útiles.  **DISEÑO PEDAGÓGICO** | | * El objetivo del modelo constructivista, es construir el conocimiento dentro de la reproducción del mismo, para lo cual toma siempre en cuenta el entorno y el contexto. * El aprendizaje debe entenderse como la reconstrucción de saberes culturales, partiendo de los conocimientos previos y permitiendo su reorganización interna. * Apunta a la construcción de individuos autónomos, creativos y dispuestos al aprendizaje, en un entorno que considera sus necesidades, actitudes y creencias para brindarles un entorno idóneo para aprender. |
| ***Enfoques:*** Sociocultural | | |
| Es un modelo que tiene el objetivo de trabajar por el desarrollo integral de las personas y de la comunidad a partir de las reflexiones y búsqueda de caminos para la interacción social, cooperación, vínculos relacionados con la incorporación de saberes culturales implementados a la educación.  De esta manera este enfoque acompañado del constructivismo, se convierten en un proceso por el cual se logra un aprendizaje significativo a través de las experiencias que tenga el docente para enseñar contenidos extensos de forma creativa, incentivando al estudiante a generar nuevos conocimientos. Por lo tanto, la evaluación es guiada por el docente y se convierte en un proceso constante para comprobar la evolución del estudiante. | | |
| ***Principios educativos*** | ***Metáfora educativa*** | |
| **Se debe generar una situación problémica que genere un choque cognitivo y social.**   * Al principio de la interacción con el software el estudiante encontrará diferentes temas que llevan consigo problemáticas sociales, mediante dichas problemáticas se plantean las bases que motivará al estudiante a solucionar problemas del entorno a través del software educativo * Durante la interacción con el software, el docente será un agente cultural, encargado de mediar el saber sociocultural, los procesos y mecanismos que permitan de los estudiantes se apropien en el uso del software educativo. | El estudiante   * **Es un sujeto activo en su proceso de enseñanza-aprendizaje**   El individuo interactúa de manera activa con conocimientos previos a través del software y adquiere las competencias de acuerdo a las actividades que va desarrollando.   * **Aprende cuando interactúa con otros**   El estudiante de manera conjunta se reúne en grupos con otros individuos a explorar e interactuar con el software educativo de la clasificación de los residuos sólidos, en donde los individuos se pueden ayudar mutuamente a resolver las inquietudes que tenga acerca de la temática. | |

*Formato 8. Ejemplo resumido del diseño de la secuencia de aprendizaje basado en competencias.*

|  |  |  |
| --- | --- | --- |
| **COMPETENCIA#1** | | |
| Relaciono la transformación de los recursos naturales con el desarrollo tecnológico y su impacto en el bienestar de la sociedad. | | |
| **Elementos** | **Aplicación modelo pedagógico** | **Indicadores** |
| * Me intereso por las tradiciones y valores de mi comunidad y participó en la gestión de iniciativas en favor del medio ambiente, la salud y la cultura (como jornadas de recolección de materiales reciclables, vacunación, bazares, festivales, etc.). * Indago sobre las posibles acciones que puedo realizar para preservar el ambiente, de acuerdo con normas y regulaciones. | Este modelo constructivista aplicado en el contexto busca trabajar por el desarrollo integral de las personas, es por esto que, a partir de las reflexiones y búsqueda de caminos para la resolución de problemas, cooperación y vínculos relacionados ayuda a la incorporación de saberes del entorno a implementarlos en la educación. | 1.1 Reconoce el impacto que genera la no clasificación de residuos sólidos.  1.2 Demuestra habilidades cuando clasifica correctamente los residuos en los diferentes contenedores  1.3 Identifica las diferentes formas de reciclaje y buen uso de los residuos. |
| **Secuencia de aprendizaje** | | |
| **Objetivos:**   * Conocer el impacto ambiental por el mal manejo de los residuos. * Aprender a clasificar los residuos en los distintos contenedores. * Informar a los estudiantes las causas del buen uso de los residuos. | | **BARRA DE RECURSOS** |
| **Navegación:**  Clasificación de los residuos sólidos.  Efectos y causas.  Tipos de residuos.  Contenedores, rojo, verde, azul, gris, entre otros**.**  Ventajas, impacto favorable, tips para el cuidado del ambiente. |
| **Problema:**  Pedro Pérez un estudiante promedio que cursa el grado 6, un día en el descanso se percató que la cancha donde juegan todos, estaba totalmente llena de basuras y residuos sólidos, no obstante, los contenedores de basura estaban vacíos, Pedro al ver esto, decide implementar una propuesta para mejorar el impacto ambiental que se genera por el mal manejo de los residuos sólidos en la institución educativa donde estudia, la cual está ubicada en una zona urbana del municipio de Montería. Él se percata de que sus compañeros no reconocen los códigos de colores para clasificar los residuos en su respectiva caneca, él sabe que la clasificación inadecuada de los residuos ocasiona consecuencias negativas para el medio ambiente.  Al no tener muchos amigos por ser una persona ambientalista, se le dificulta que sus compañeros comprendan la importancia de clasificar los residuos, y cuidar el medio ambiente.  Por lo tanto, un docente del área de naturales ayuda a Pedro con su propuesta, al ver que Pedro no logra socializar con sus compañeros, el docente ayuda y asesora a los estudiantes de esa institución para que cumplan con la correcta clasificación de los residuos. | |
| **Documentación:**   * Videos. * Diapositivas. * Infografías. * Imágenes. |
| **Comunicación:**   * Correo electrónico. * Chat. * Foro Educativo. |
| **Estrategias:** El docente de ciencias naturales debe conocer la clasificación por colores de los residuos sólidos, además, identificar la clasificación de estos.   * **Impacto ambiental**   Desaparición de especies animales y vegetales.  Cambios sobre el modelo de vida humano.  Cambios en la naturaleza de las sustancias que hacen vida en el medio ambiente, tanto en el suelo, el agua o el aire.   * **Residuos**   Residuos domésticos.  Residuos comerciales.  Residuos industriales.  Residuos peligrosos.  Residuos inertes.  Residuos no peligrosos.  Residuos biodegradables.   * **Mal manejo de los residuos**   Contaminación del aire.  Contaminación del agua.  Degradación de los suelos.  Alteración de los ecosistemas.   * **Clasificación de los residuos**   Utilización de los contenedores.  Buen uso de los códigos de colores.  Contenedor azul.  Contenedor verde.  Contenedor rojo.  Contenedor gris.   * **Buen uso de los residuos**   Disminución del uso de materia prima.  Ahorro de energía.  Disminución del riesgo de enfermedades.  Disminución del riesgo de muerte de animales. | |

***Diseño guía de metáforas***

|  |  |  |
| --- | --- | --- |
| **DISEÑO GUÍA DE METÁFORAS** | | |
| **NOMBRE** | **IMAGEN** | **DESCRIPCIÓN** |
| **Ayuda** |  | Al presionar este botón se despliega la pregunta de ayuda al usuario la cual expresa; ¿en qué podemos ayudarte?, mediante la cual el usuario podrá describir alguna inquietud o problema que tenga al momento de interactuar con el software |
| **Buscar** |  | Al presionar este botón se despliega el menú buscador, para que el usuario busque directamente un determinado contenido o elemento del software. |
| **Menú** |  | Al presionar este icono se despliega el menú del software el cual contiene diferentes elementos con los cuales el usuario puede interactuar. |
| **Idioma** |  | Al presionar este icono se despliega el menú de opciones de configuración de idioma del software para que el usuario determine el idioma con el cual prefiere trabajar en él. El idioma por defecto es español. |
| **Usuario** |  | En esta sección, quién está utilizando el software digita su usuario (el cuál puede ser el correo electrónico o un usuario proporcionado por los desarrolladores del software) para acceder a los diferentes elementos. |
| **Contraseña** |  | En esta sección, quién está utilizando el software digital su contraseña (con anticipación por el usuario) para acceder a los diferentes elementos. |
| **Mostrar Contraseña** |  | Al presionar este icono el usuario puede observar su contraseña (en caso tal que necesite verificar si se ha escrito correctamente). |
| **Configuración** |  | Al presionar este icono se despliegan las opciones de configuración de la cuenta del usuario en el software. |
| **Cuenta** |  | Al presionar este icono se muestra una ventana con la información personal y de la cuenta del usuario que está usando el software. |
| **Volver a cargar** |  | Al presionar este icono se actualizará o volverá cargar la información de la ventana en la cual está navegando el usuario en el software. |
| **Contenidos** |  | Al presionar este icono se direcciona al usuario a la ventana de contenidos del software. |
| **Actividades** |  | Al presionar este icono se direcciona al usuario a la ventana de actividades del software. |
| **Evaluación** |  | Al presionar este icono se direcciona al usuario a la evaluación de los contenidos y actividades del software. |
| **Créditos** |  | Al presionar este icono se direcciona al usuario a la ventana de créditos para los desarrolladores del software, así como de los elementos que utilizaron para el desarrollo de este. |
| **Calificaciones** |  | Al presionar este icono se direcciona al usuario a la ventana de calificaciones la cual contiene las notas cuantitativas de las actividades qué desarrolle dentro del software. |
| **Guardar** |  | Al presionar este icono se descarga un informe con los datos de navegación del usuario dentro del software. |

***Diseño de la interfaz***

****

Los estudiantes deberán ingresar con su usuario y contraseña para ser identificados por el software, y tener acceso a todo el contenido relacionando con la clasificación de residuos sólidos.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ventana 01:** | **Acceso (Login)** | | | | |
| **Texto** | **Imágenes** | **Audio** | **Video** | **Animación** | **Acciones** |
| * Clasificación de los residuos sólidos * Usuario * Contraseña * Acceder | * Icono de clasificación software * Fondo clasificación de residuos * Icono de ayuda * Icono de idioma | Click (al presionar los diferentes elementos) | No | No | * Digitar usuario * Digitar contraseña * Mostrar contraseña * Acceder al software * Cambiar idioma * Solicitar ayuda |

**** En esta ventana se encuentra, el menú de pestañas del software, un carrusel con imágenes alusivas a la clasificación de residuos sólidos y varias herramientas que puede usar el estudiante para navegar a través del software.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ventana 02:** | **Inicio** | | | | |
| **Texto** | **Imágenes** | **Audio** | **Video** | **Animación** | **Acciones** |
| * Clasificación de los residuos sólidos * Contenidos * Actividades * Evaluación * Créditos | * Icono de clasificación software * Fondo clasificación de residuos * Icono de ayuda * Icono de idioma * Icono de Actualizar página * Icono de la cuenta * Icono de Menú * Imagen de clasificación (contenedores) * Icono de contenidos. * Icono de actividades. * Icono de evaluación. * Icono de créditos. | No | No | Carrusel de Imágenes | * Acceder a cuenta * Acceder a menú * Acceder a contenidos * Acceder a actividades * Acceder a evaluación * Acceder a créditos * Cambiar idioma * Solicitar ayuda * Actualizar página |



En esta ventana los estudiantes podrán encontrar los diferentes contenidos respecto al tema que se va a tratar en el software, los contenidos están distribuidos en un video, lectura recomendada e infografía, con el objetivo que independientemente del estilo de aprendizaje que tengan los distintos estudiantes el contenido se adapte a estos y sea fácil de entender, comprender y analizar.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ventana 03:** | **Contenidos** | | | | |
| **Texto** | **Imágenes** | **Audio** | **Video** | **Animación** | **Acciones** |
| * Clasificación de los residuos sólidos. * Contenidos. * Actividades. * Evaluación. * Créditos. * Videos. * Lectura recomendada. * Infografía. | * Icono de clasificación software. * Fondo clasificación de residuos. * Botón de video. * Botón de lectura recomendada. * Botón de Infografía. * Icono de buscar. * Icono de cuenta. * Icono de menú. * Icono de contenidos. * Icono de actividades. * Icono de evaluación. * Icono de créditos. * Icono de volver a cargar. * Icono de ayuda. * Icono de idioma. | Click (al presionar los diferentes elementos) | Si | No | * Ver video. * Leer la lectura recomendada. * Acceder a la infografía. * Buscar información en Software. * Acceder a la cuenta. * Acceder al menú. * Acceder a contenidos. * Acceder a actividades. * Acceder a evaluación. * Acceder a créditos. * Actualizar página. * Solicitar ayuda. * Cambiar idioma. |



En esta ventana los estudiantes pondrán poner en práctica lo aprendido en los contenidos, es una manera de identificar el desempeño de cada uno de los estudiantes y su nivel de aprendizaje. Esta pestaña está compuesta por 4 actividades distintas; cuestionario, preguntas de verdadero o falso, arrastrar y soltar y una actividad de completa los campos.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ventana 04:** | **Actividades** | | | | |
| **Texto** | **Imágenes** | **Audio** | **Video** | **Animación** | **Acciones** |
| * Clasificación de los residuos sólidos. * Contenidos. * Actividades. * Evaluación. * Créditos. * Actividad 1. * Actividad 2. * Actividad 3. * Actividad 4. | * Icono de clasificación software. * Fondo clasificación de residuos. * Botón de actividad 1. * Botón de actividad 2. * Botón de actividad 3. * Botón de actividad 4. * Icono de buscar. * Icono de cuenta. * Icono de menú. * Icono de contenidos. * Icono de actividades. * Icono de evaluación. * Icono de créditos. * Icono de volver a cargar. * Icono de ayuda. * Icono de idioma | Click (al presionar los diferentes elementos) | Si | No | * Acceder a la actividad 1. * Acceder a la actividad 2. * Acceder a la actividad 3. * Acceder a la actividad 4. * Buscar información en Software. * Acceder a la cuenta. * Acceder al menú. * Acceder a contenidos. * Acceder a actividades. * Acceder a evaluación. * Acceder a créditos. * Actualizar página. * Solicitar ayuda. * Cambiar idioma. |



En esta ventana los estudiantes deberán realizar un test de selección múltiple con única respuesta donde pondrán a prueba todos los recursos que tiene el software para el aprendizaje de la temática abordada. La evaluación cuenta con un tiempo determinado para su realización y al finalizar arrojará la nota obtenida.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ventana 05:** | **Evaluación** | | | | |
| **Texto** | **Imágenes** | **Audio** | **Video** | **Animación** | **Acciones** |
| * Clasificación de los residuos sólidos. * Contenidos. * Actividades. * Evaluación. * Créditos. | * Icono de clasificación software. * Fondo clasificación de residuos. * Icono de buscar. * Icono de cuenta. * Icono de menú. * Icono de contenidos. * Icono de actividades. * Icono de evaluación. * Icono de créditos. * Icono de volver a cargar. * Icono de ayuda. * Icono de idioma. | Click (al presionar los diferentes elementos) | No | No | * Realizar la evaluación. * Buscar información en Software. * Acceder a la cuenta. * Acceder al menú. * Acceder a contenidos. * Acceder a actividades. * Acceder a evaluación. * Acceder a créditos. * Actualizar página. * Solicitar ayuda. * Cambiar idioma. |

****

Aquí se encontrará información sobre los desarrolladores del Software como nombres, correos. Además del nombre del programa al que pertenecen, nombre de la facultad y universidad.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ventana 06:** | **Créditos** | | | | |
| **Texto** | **Imágenes** | **Audio** | **Video** | **Animación** | **Acciones** |
| * Clasificación de los residuos sólidos. * Contenidos. * Actividades. * Evaluación. * Créditos. * Desarrolladores * Laura Marchena Madera. * Marianela Medrano Gómez. * Eduardo José Robles Carmona. * Juan José Vergara Oliveros. * Con la colaboración de: * Visual Studio Code * https://daisyui.com/ * https://www.flaticon.es/ * https://www.freepik.es/ * https://es.educaplay.com/ | * Icono de clasificación software. * Fondo clasificación de residuos. * Icono de contenidos. * Icono de actividades. * Icono de evaluación. * Icono de créditos. * Foto de Laura Marchena Madera. * Foto de Marianela Medrano Gómez. * Foto de Eduardo José Robles Carmona. * Foto de Juan José Vergara Oliveros. | No | No | No | * Buscar información en Software. * Ver los creadores del software * Poder acceder a las plataformas que se utilizaron para la creación del software. * Volver a cualquiera de los iconos de: Contenidos, Actividades y Evaluación. |

**Diccionario de datos**

El backend es una parte de un sistema informático que es responsable de procesar, almacenar y administrar información en segundo plano. Consiste en un conjunto de tecnologías, lenguajes y herramientas que le permiten crear aplicaciones y servicios web funcionales. El backend es responsable de interactuar con la base de datos, administrar los servidores, implementar la lógica comercial y proporcionar la información y los recursos necesarios para que la interfaz de usuario del frontend pueda interactuar con los usuarios.

A continuación, estos son los componentes y/o elementos principales del backend:

**Servidor web:** Es responsable de recibir las solicitudes de los usuarios y enviar las respuestas apropiadas.

El primer servidor Web fue puesto en línea en 1991 el autor e inventor de poner el primer servidor en línea fue Tim Berners-Lee como método para compartir día a día la información. El primer servidor web Contaba con un micro de la serie 68040 de Motorola capaz de trabajar a 25 MHz, una memoria de 8 MB ampliables a 64 MB y un monitor de 17''. La máquina, que costaba 6500 dólares, recibió el nombre de NeXT Computer o simplemente "The Cube".

**Base de datos:** Se encarga de almacenar y administrar los datos de la aplicación.

Su origen se remonta a 1884 con Herman Hollerith, que desarrolló el tabulador electromagnético de tarjetas perforadas con el fin de ayudar en el resumen de información y posteriormente a la contabilidad.

**APIs** (interfaz de programación de aplicaciones): permite la comunicación y el intercambio de datos entre diferentes aplicaciones o servicios.

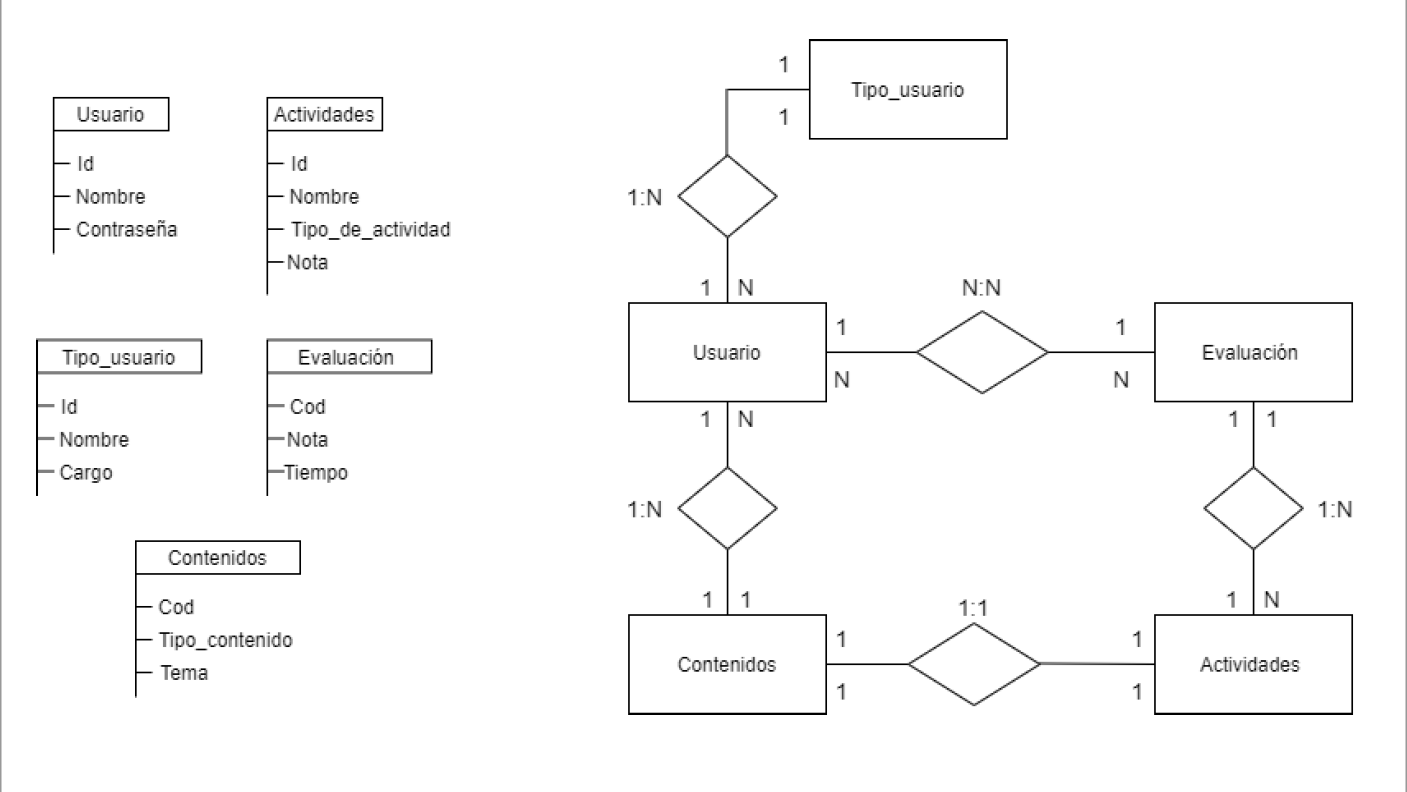
Las API surgieron en los comienzos de la informática, mucho antes que la computadora personal. En esa época, una API normalmente se usaba como biblioteca para los sistemas operativos.

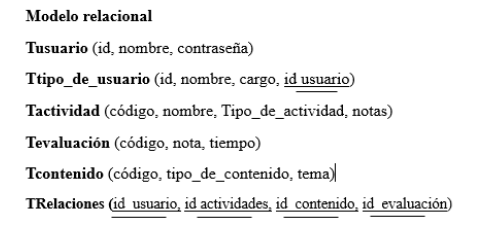
**Sistema de seguridad:** Son responsables de proteger los datos y recursos de las aplicaciones de ataques y vulnerabilidades.

**Frameworks**: Proporcionan una estructura y conjunto de herramientas para simplificar el desarrollo del backend. Estas bibliotecas predefinidas ofrecen funciones comunes, manejo de solicitudes HTTP, manejo de bases de datos y otros componentes esenciales.

Este tipo de Frameworks facilitan el trabajo con Lenguajes de Programación de lado del Servidor como Python, Ruby, PHP, Javascript del lado del servidor (Node JS), Java, etc. Ayudan a gestionar las bases de datos, el envío y procesamiento de datos, cookies, sesiones, etc.

***Modelo de Entidad - Relación del Software educativo para la clasificación de residuos***

******





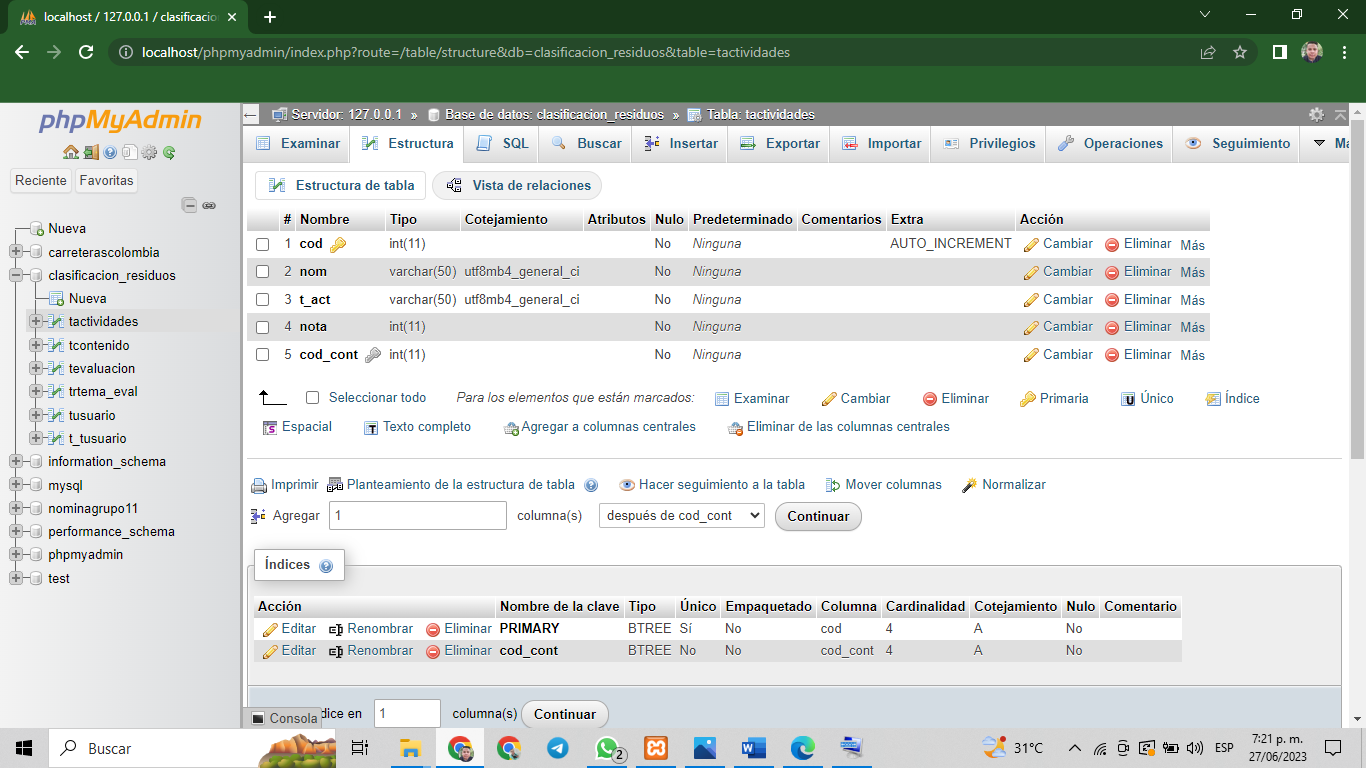
**Modelo NoSql**

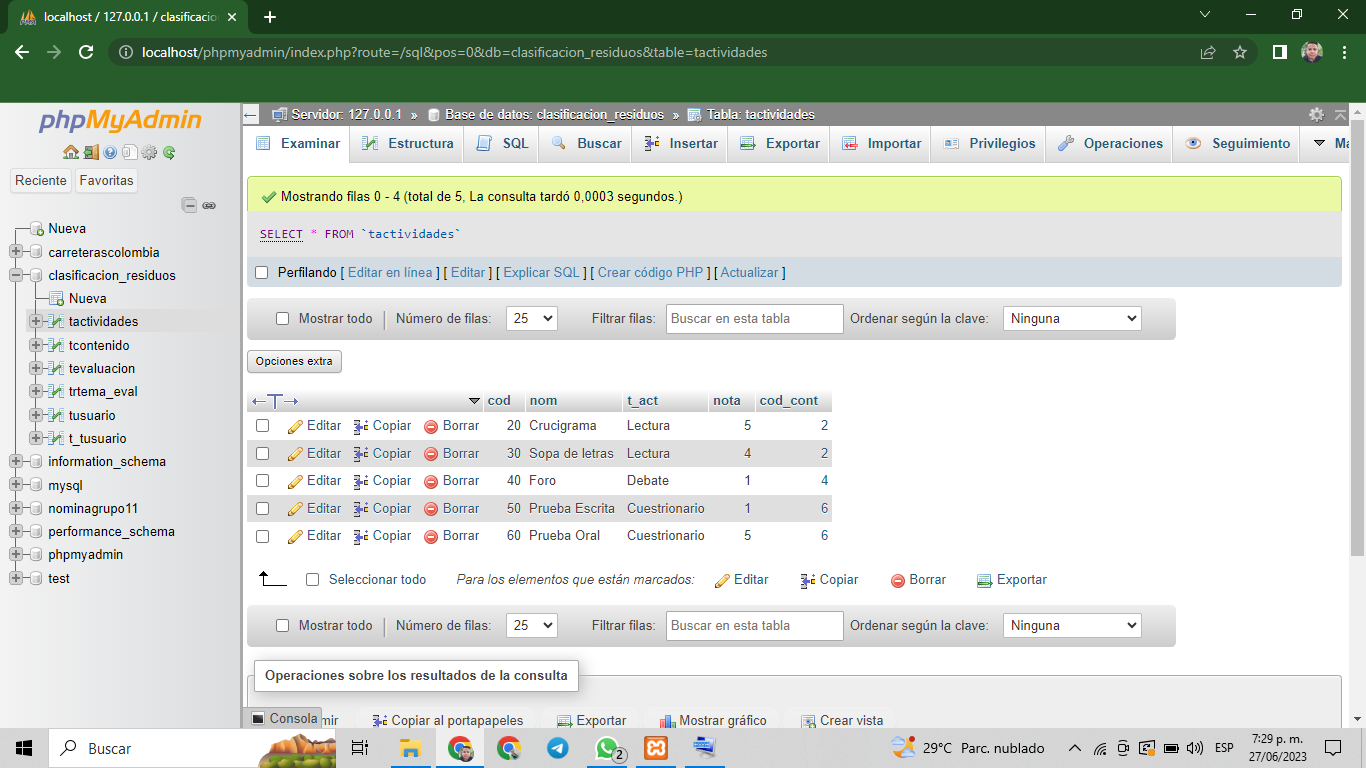
De acuerdo con los requisitos del sistema se identificaron las siguientes entidades de datos y sus atributos;

* Tactividades (código, nombre, tipo\_actividad, código\_contenido)
* Tcontenido (código, tipo\_contenido, tema, Id\_usuario)
* Tevaluacion (código, nota, time, código\_contenido, código\_actividad)
* trtema\_eval (Id, código\_contenido, código\_evaluación, código\_actividad)
* tusuario (Id, nombre, contraseña.)
* t\_tusuario (Id, nombre, cargo, Id\_usuario)

**tactividades**

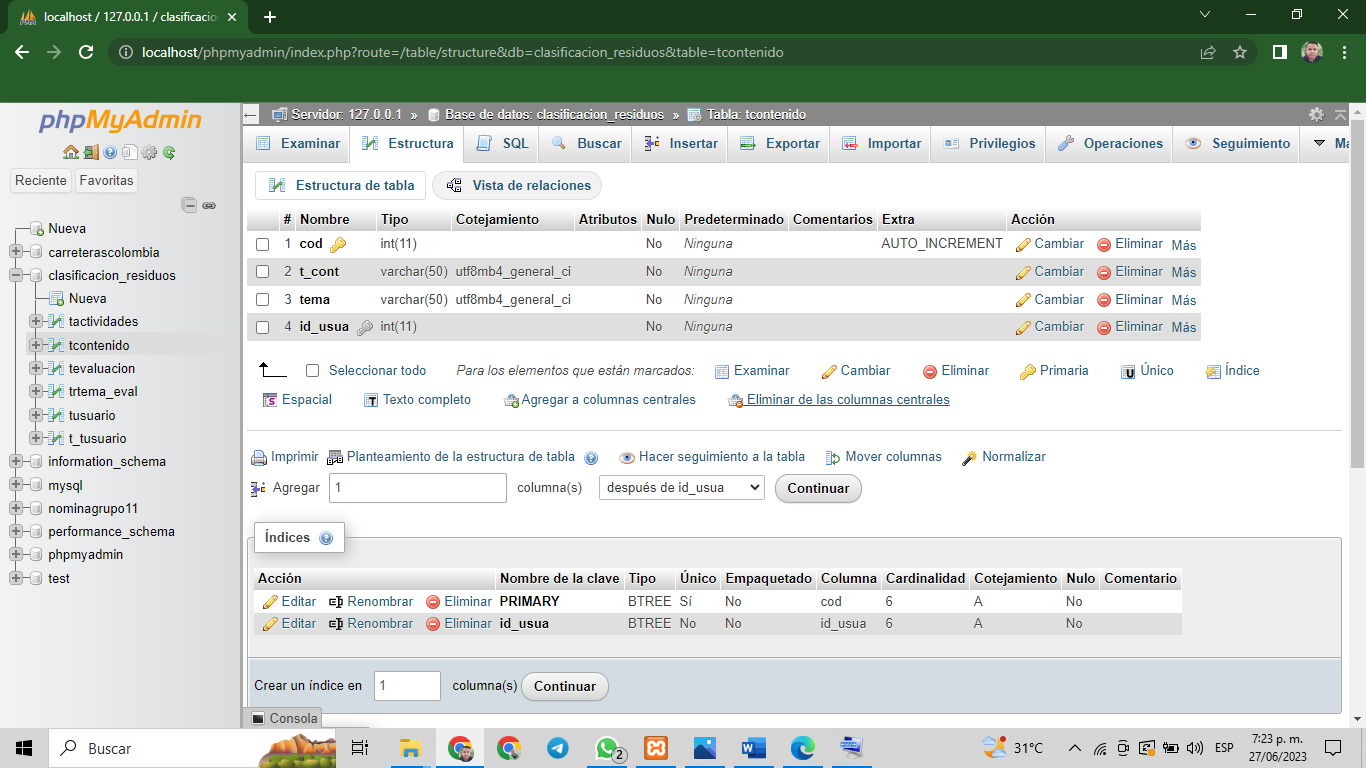
En la primera tabla, denominada *tactividades* se encuentran cinco atributos los cuales son; código, nombre, tipo de actividad, código del contenido. En esta tabla se observa una llave primaria la cual corresponde al atributo *código* y se encuentra una llave foránea que corresponde al atributo *código del contenido*. Los datos que se almacenan en esta tabla corresponden a las actividades del software entre las cuales encontramos crucigramas, sopa de letras, foros, pruebas escritas, entre otros, al igual que los tipos de actividades como lo son lectura, debates, cuestionarios y más. Además de la calificación de las actividades las cuales van en una escala de 0 a 5.

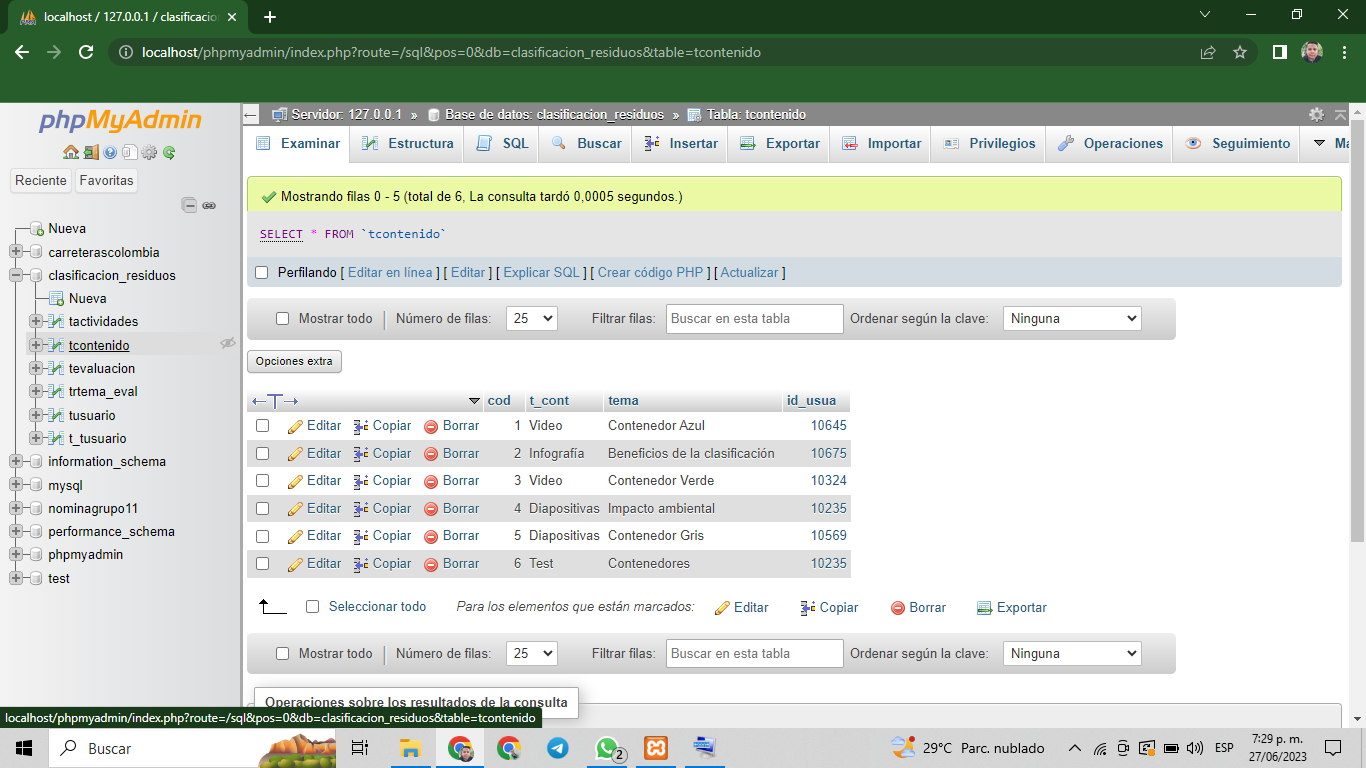




**Tcontenido**

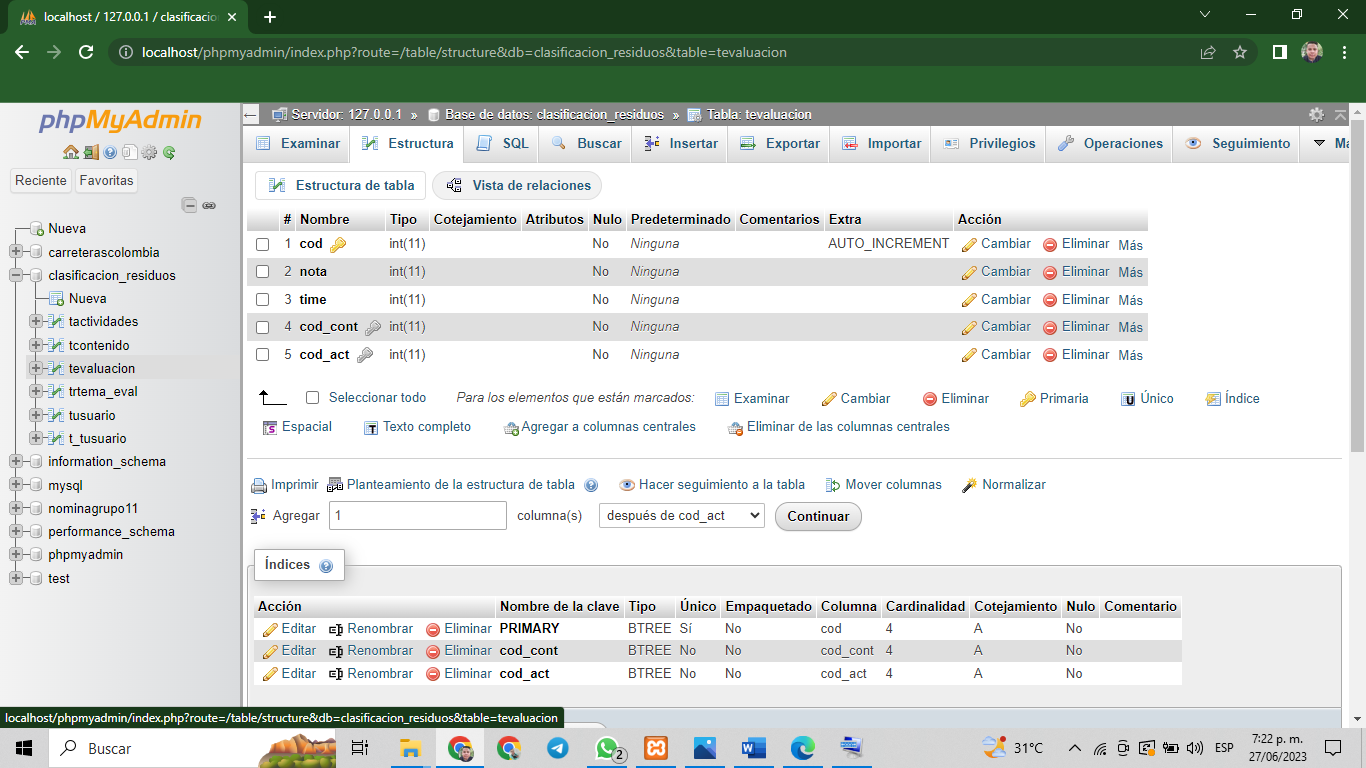
La segunda tabla denominada tcontenido contiene cuatro atributos; código, tipo de contenido, tema e Id del usuario. En esta tablas se encuentra que la llave primaria corresponde al atributo *código* y también se encuentra una llave foránea que corresponde al atributo del *Id del usuario*. En esta tabla se almacenan los tipos de contenido del software de entre los cuales se destacan vídeos, infografías y diapositivas, además de las temáticas como impacto ambiental, beneficios de la clasificación, contenedores, entre otras.

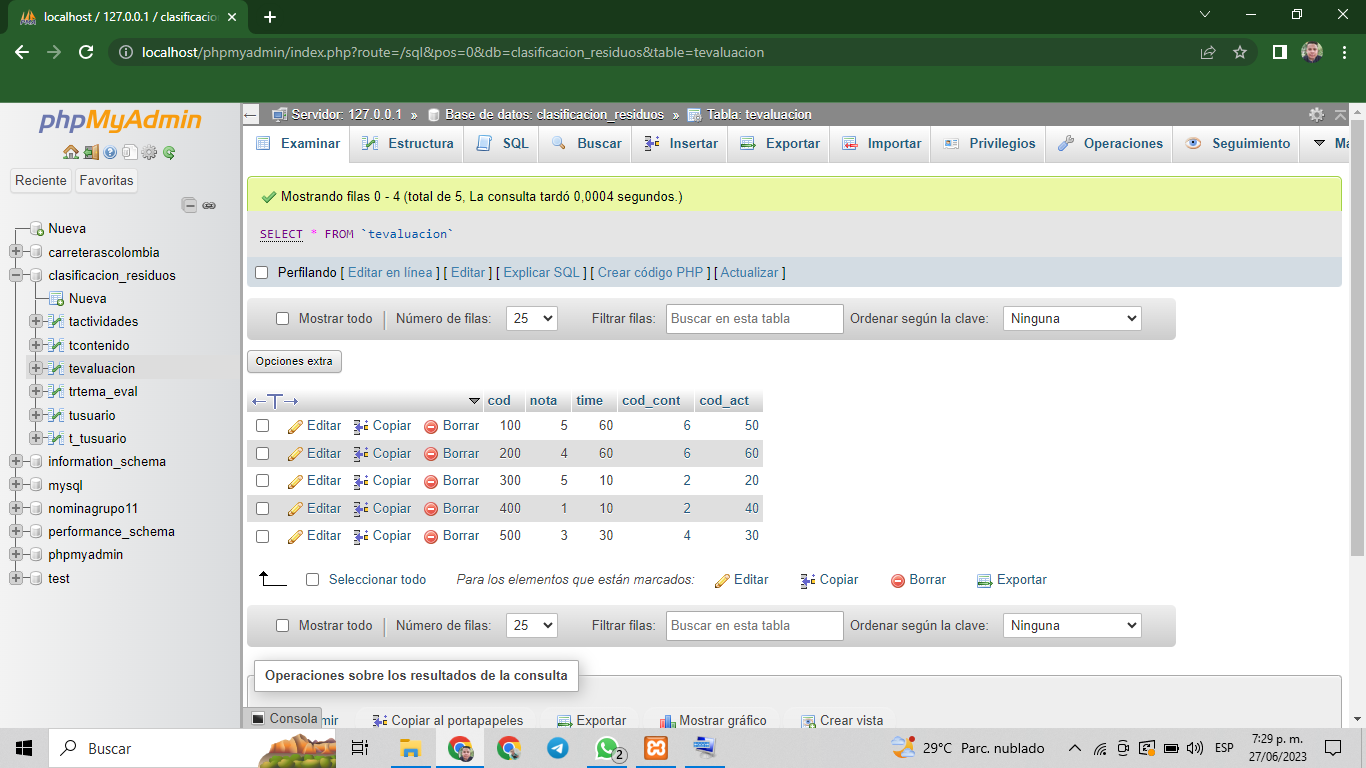




**tevaluacion**

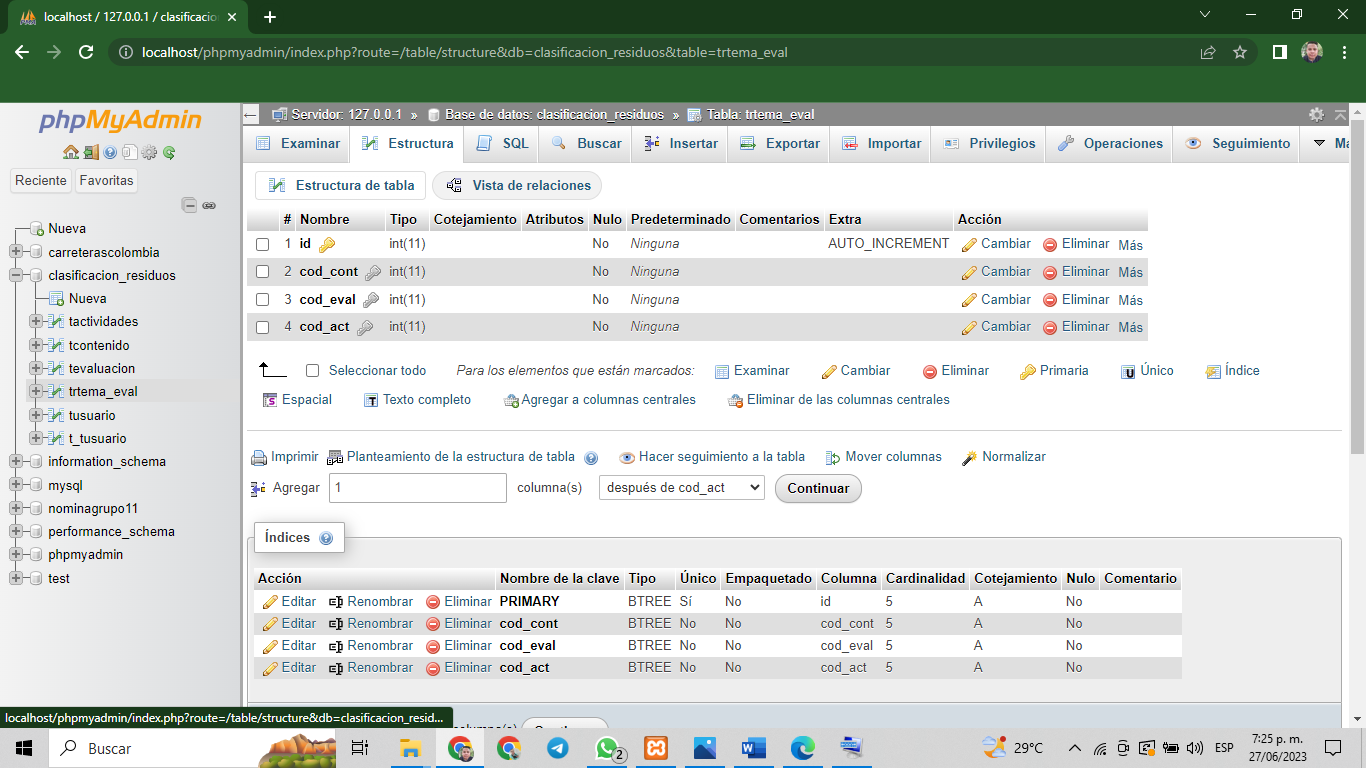
La tercera tabla del modelo lleva por nombre tevaluación y tiene cinco atributos; código, nota, time (tiempo), código del contenido y código de la actividad. Para esta tabla se encuentra que la llave primaria corresponde al *código* y se encuentran dos llaves foráneas las cuales corresponden a los atributos del *código del contenido* y *el código de la actividad*. En esta tabla se almacenan los datos relacionados con la nota obtenida en las evaluaciones, las cuales van en una escala de calificación de 0 a 5, el tiempo para cada evaluación, además de el tipo de contenido y el tipo de actividad.

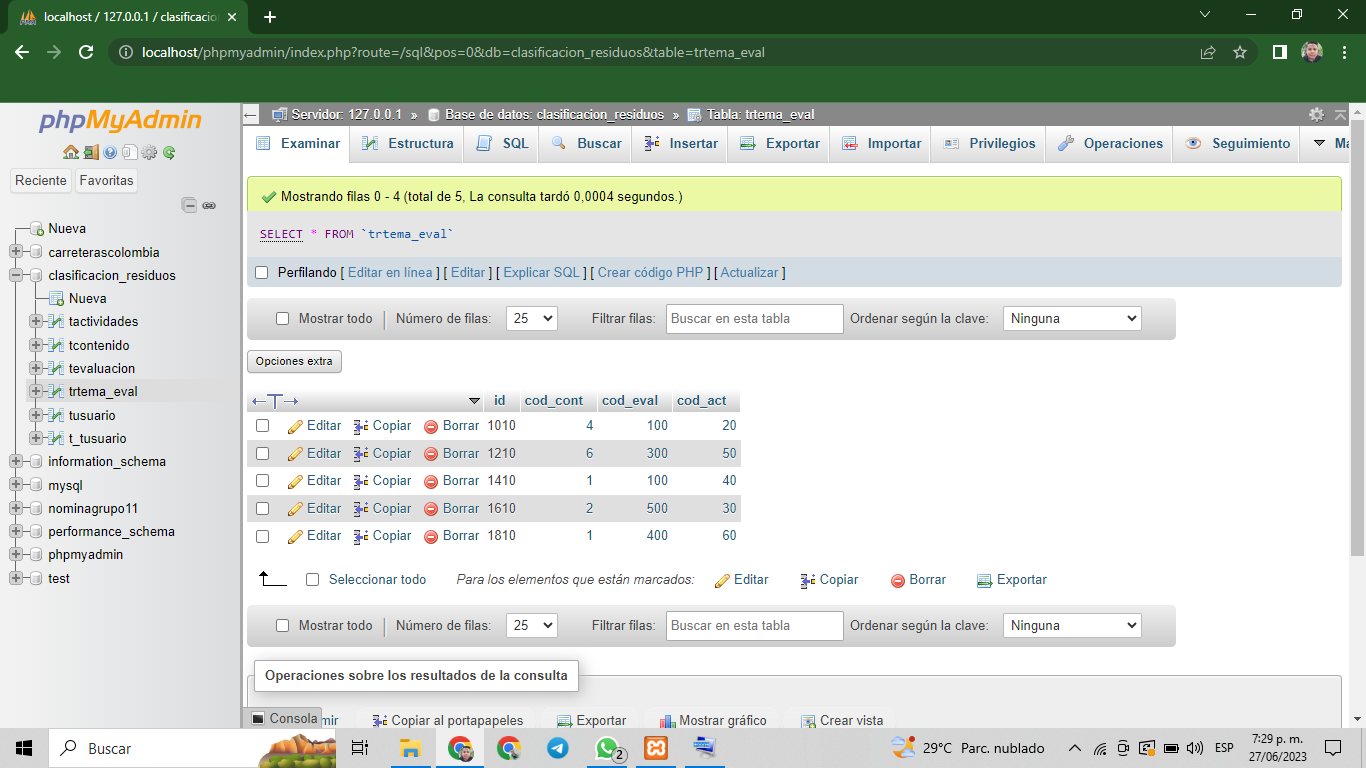




**trtema\_eval**

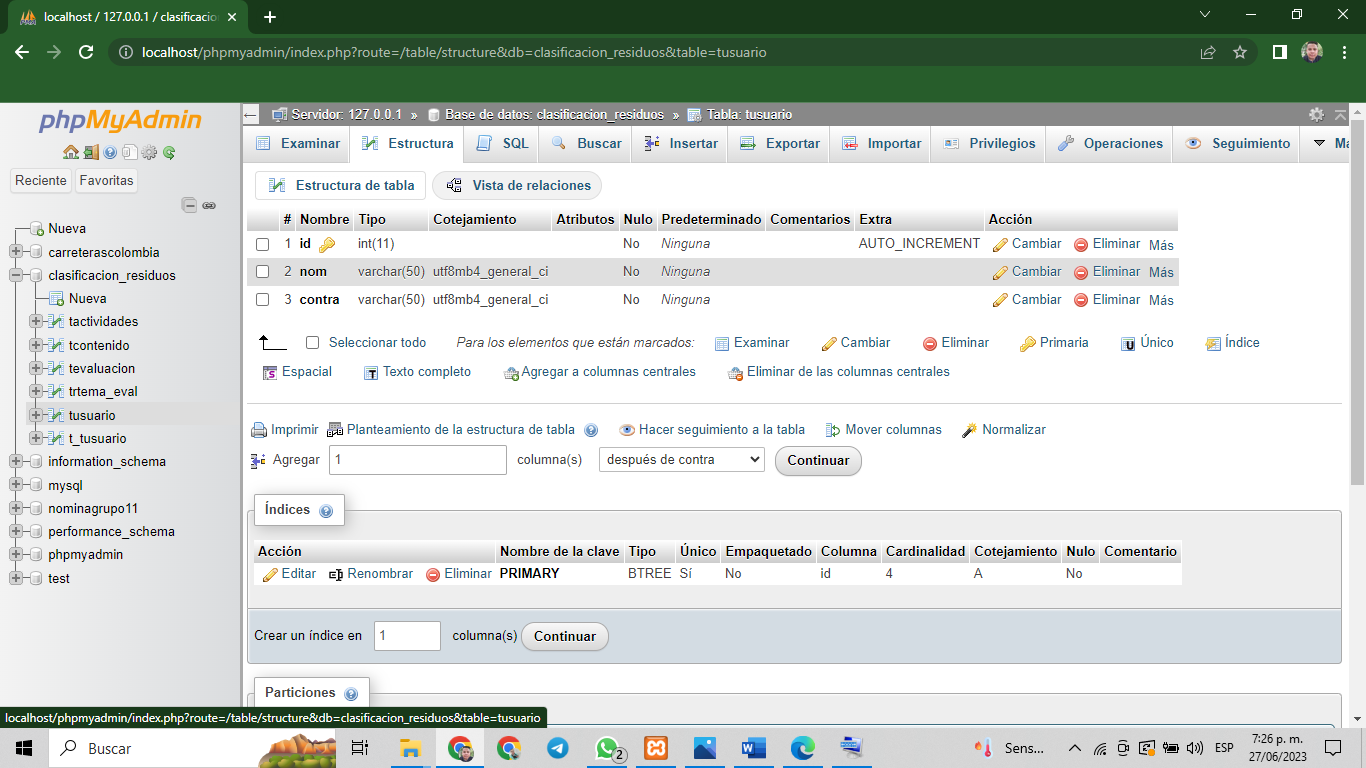
La cuarta tabla del modelo lleva por nombre trtema\_evaluación, es una tabla que contiene cuatro atributos; Id, código del contenido, código de la evaluación y código de la actividad. Para esta tabla se encuentra que que la llave primaria corresponde al *Id* y los 3 atributos restantes que *son código del contenido, código de la evaluación* y *código de la actividad* corresponden a llaves foráneas. En esta tabla se almacenan los datos relacionados con el tipo de contenido en el software, el tipo de evaluación y el tipo de actividad que desarrollarán los usuarios.

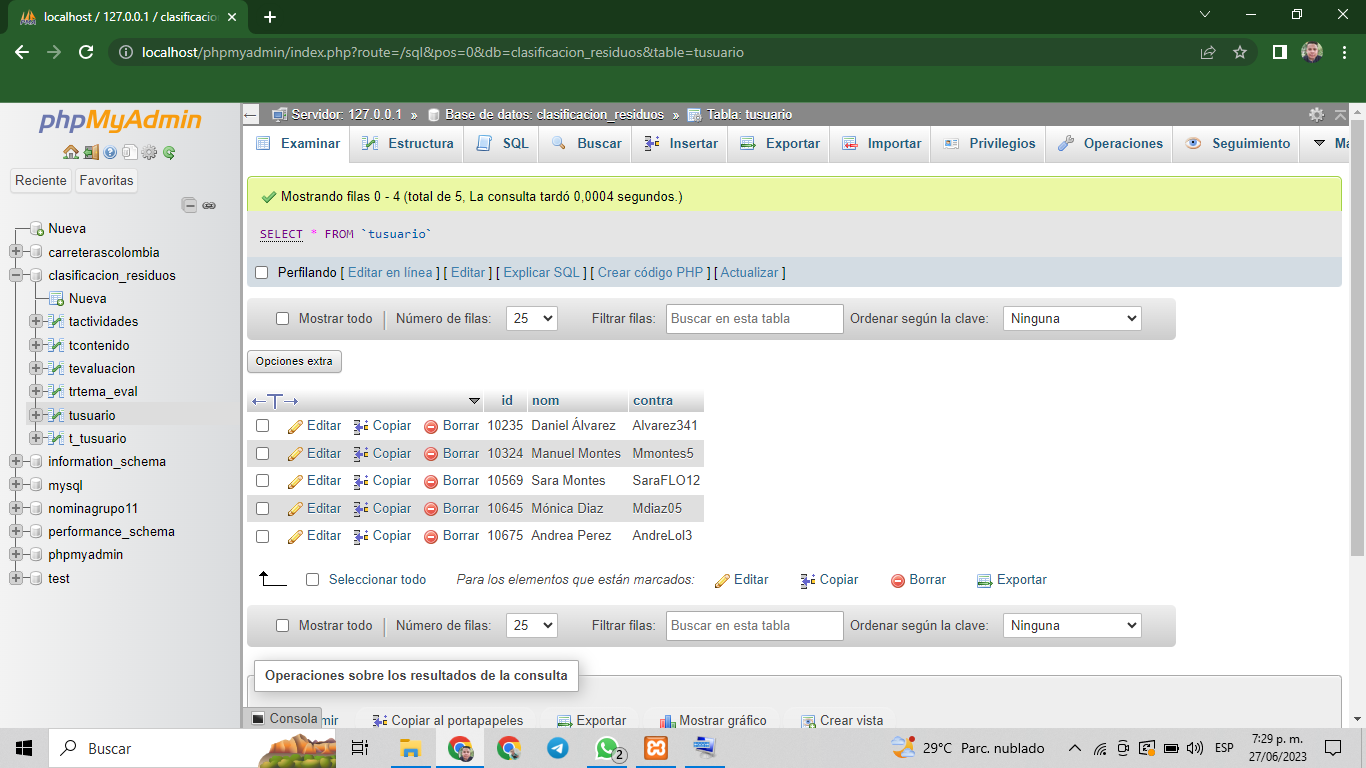




**tusuario**

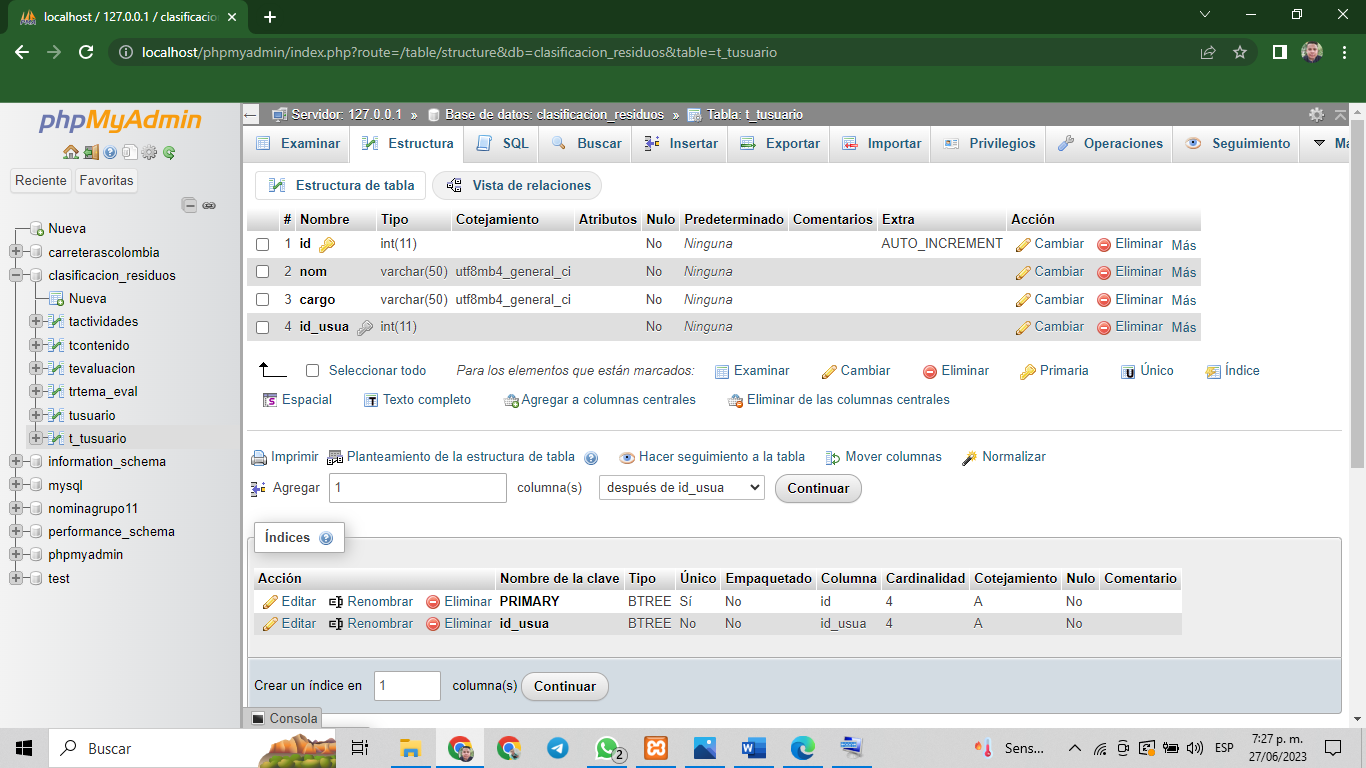
La quinta tabla del modelo lleva por nombre tusuario, esta tabla contiene tres atributos; Id, nombre y contraseña. Para esta entidad la llave primaria corresponde al *Id del usuario* y no se encuentran llaves foráneas. En esta tabla se almacenan los datos que corresponden al Id del usuario, también el nombre del usuario independientemente de su género y rol en el software y por último la contraseña con la cual puede acceder al software.

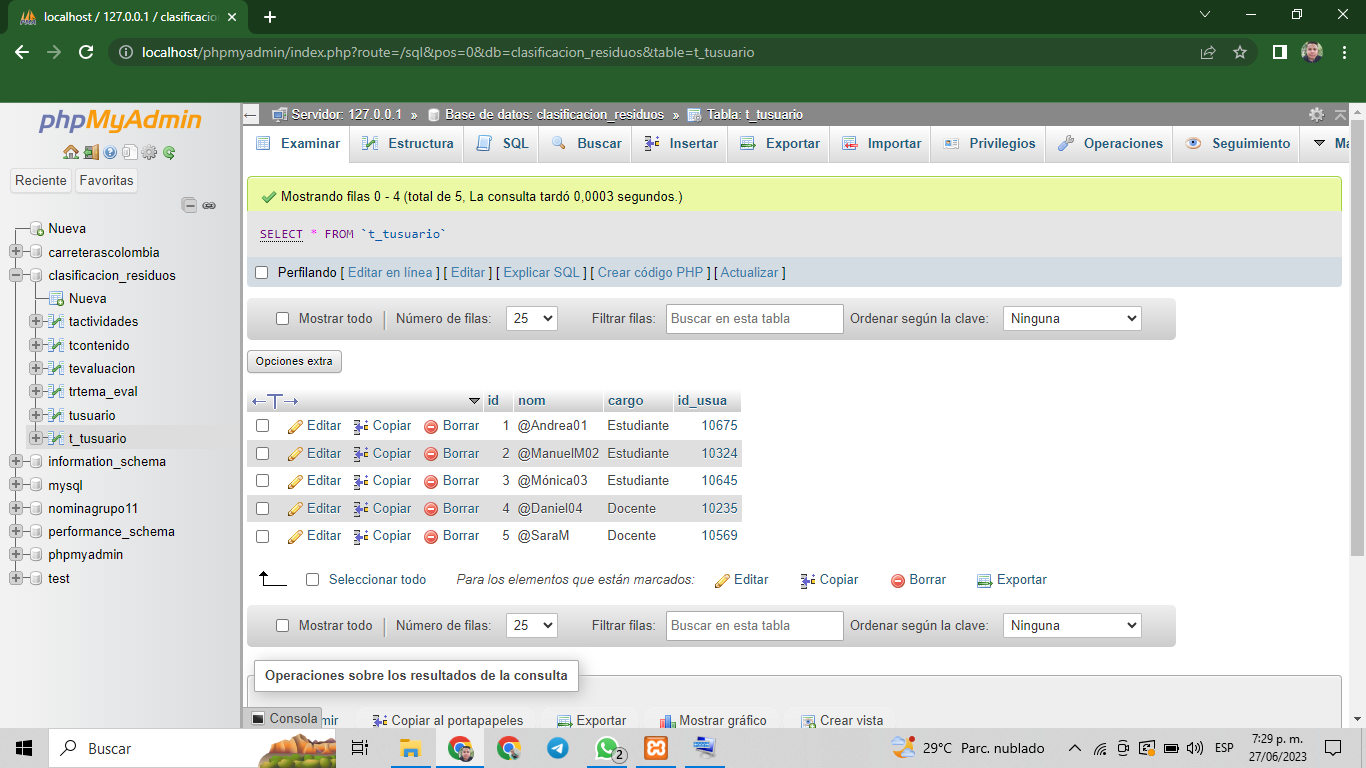




**t\_tusuario**

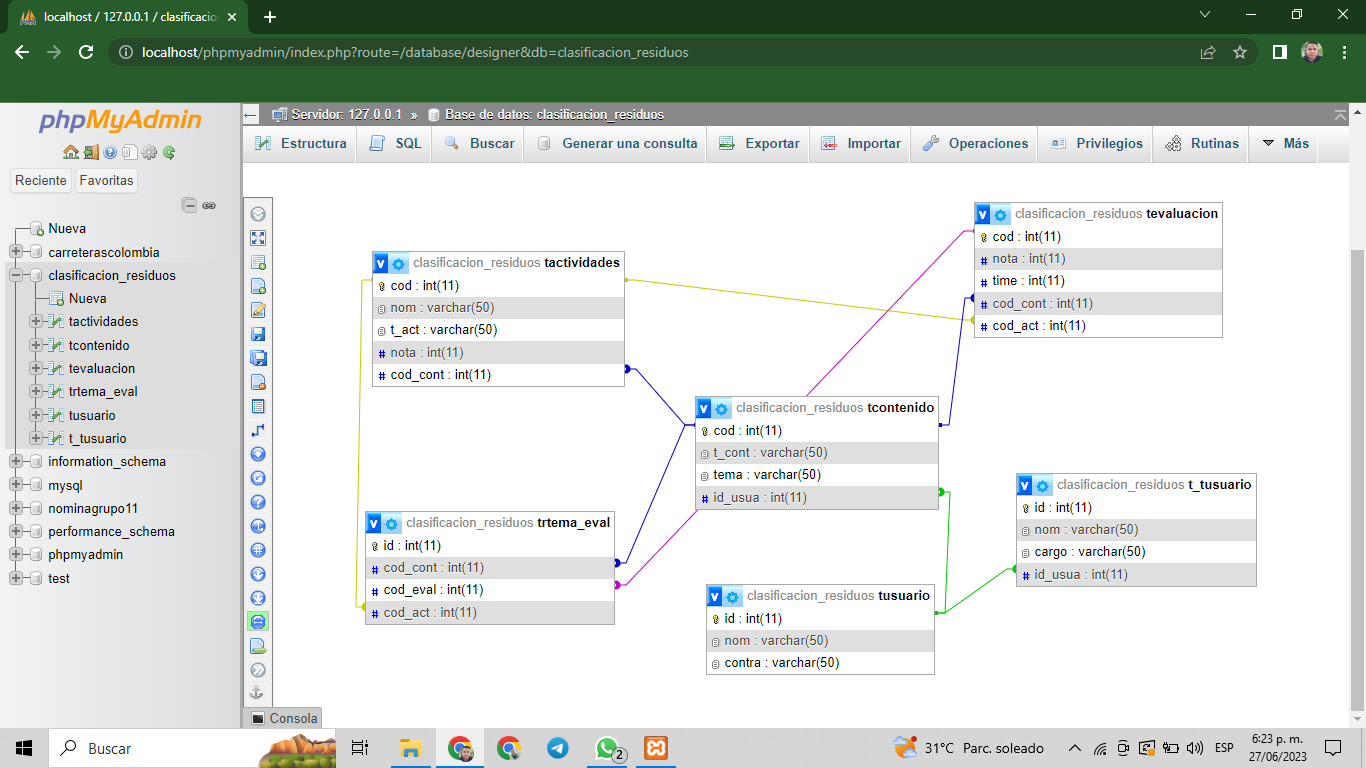
La última tabla del modelo lleva por nombre t\_usuario que corresponde al tipo de usuario, esta tabla contiene cuatro atributos; Id, nombre, cargo e Id del usuario. Para esta tabla se encuentra que la llave primaria corresponde al *Id* y se encuentra también una llave foránea que corresponde al *Id del usuario*. En esta tabla se almacenan los datos relacionados con el usuario que accede al software, el cargo que desempeña en el software y finalmente el Id de dicho usuario.





**Diseñador**

Finalmente se encuentra el diseñador, en el cual se observa el diagrama general de la base de datos con cada una de las tablas (entidades) y sus atributos, para el diseño de la base de datos se tuvo en cuenta principalmente el modelo de entidad relación que también se anexó al final.



**¿Qué es un archivo NoSQL?**

Un archivo NoSQL (Not Only SQL) se refiere a una forma de almacenamiento de datos que difiere del modelo relacional tradicional utilizado en las bases de datos SQL. Mientras que las bases de datos SQL almacenan datos en tablas estructuradas con esquemas fijos, las bases de datos NoSQL ofrecen un enfoque más flexible y escalable para el almacenamiento y la recuperación de datos.

En lugar de utilizar tablas y filas, las bases de datos NoSQL utilizan diferentes modelos de datos, como documentos, grafos, columnas o pares clave-valor, para almacenar información. Esto les permite manejar grandes volúmenes de datos no estructurados o semiestructurados de manera más eficiente.

Los archivos NoSQL son especialmente útiles para aplicaciones web y móviles donde la escalabilidad y el rendimiento son fundamentales. Estas bases de datos son altamente escalables horizontalmente, lo que significa que pueden manejar grandes volúmenes de datos distribuidos en múltiples servidores. Además, suelen ofrecer una mayor velocidad de lectura y escritura en comparación con las bases de datos SQL tradicionales.

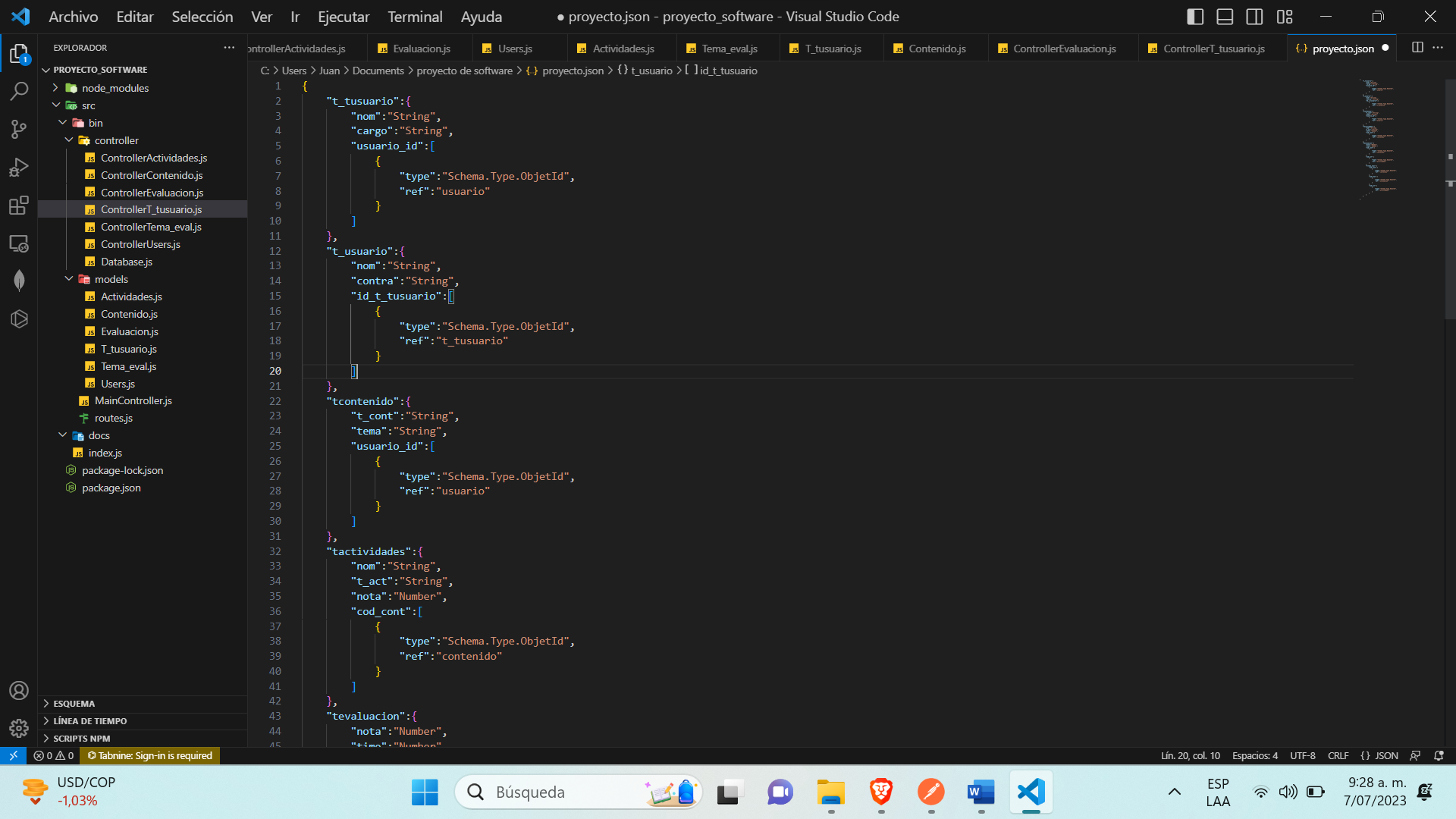
Algunos ejemplos populares de bases de datos NoSQL son MongoDB, Cassandra, Redis, Couchbase y Amazon DynamoDB. Cada una de estas bases de datos NoSQL tiene su propio modelo de datos y características específicas, por lo que es importante elegir la más adecuada para el caso de uso y los requisitos específicos de la aplicación.

**Procedimiento para crear un archivo NoSQL en un ordenador en Visual Studio Code utilizando JSON.**

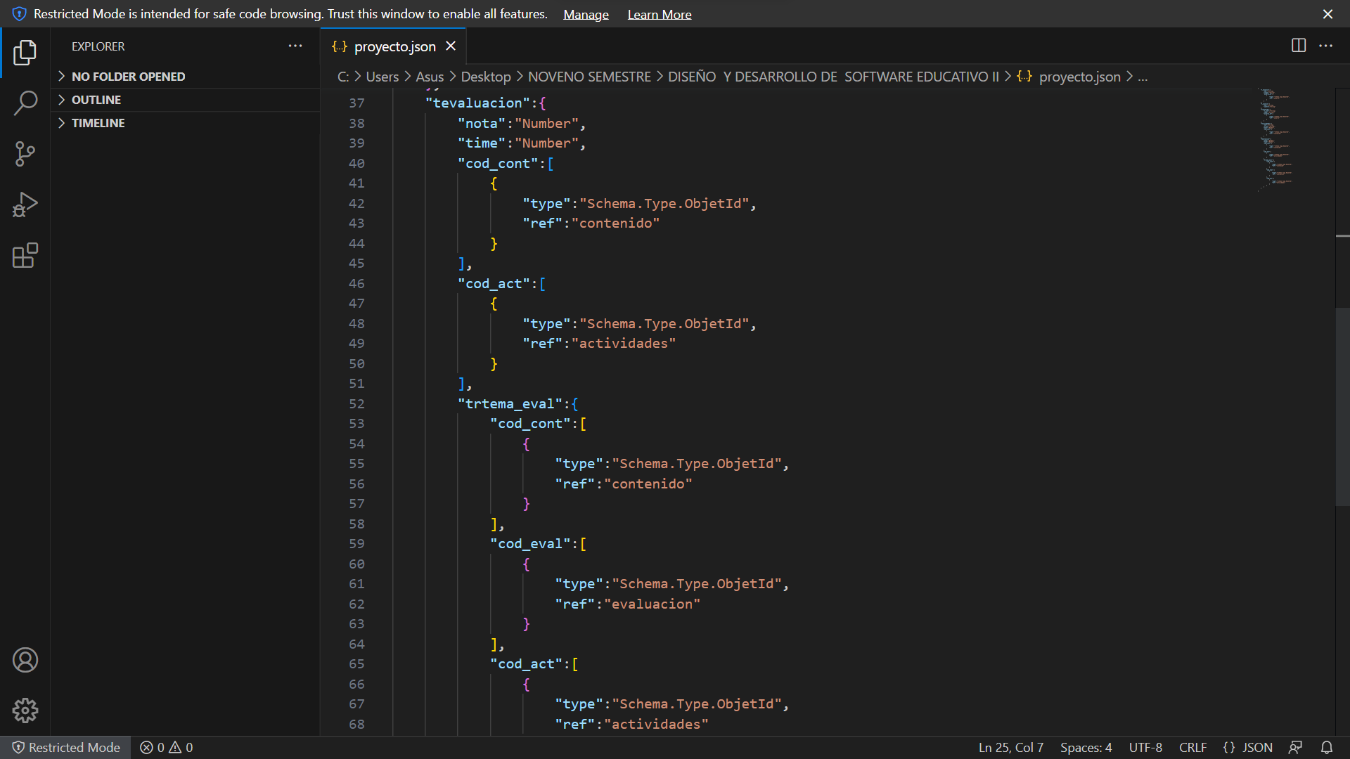
1. Abre Visual Studio: Inicia Visual Studio en tu computadora.
2. Crea un nuevo proyecto: Selecciona la opción "Crear un nuevo proyecto" en la página de inicio de Visual Studio. Elige el tipo de proyecto que deseas crear, como una aplicación de consola o una aplicación web.
3. Agrega una referencia al paquete JSON.NET: JSON.NET es una biblioteca popular para trabajar con JSON en .NET. Para agregarlo a tu proyecto, haz clic con el botón derecho en el proyecto en el Explorador de soluciones y selecciona "Administrar paquetes NuGet". En la ventana de administración de paquetes NuGet, busca "Newtonsoft.Json" y haz clic en "Instalar" para agregarlo a tu proyecto.
4. Crea un archivo JSON: Dentro de tu proyecto, puedes crear un archivo JSON utilizando el lenguaje de programación que estés utilizando. Por ejemplo, si estás desarrollando una aplicación de consola en C#, puedes agregar un archivo JSON haciendo clic derecho en el proyecto en el Explorador de soluciones, selecciona "Agregar" y luego "Nuevo elemento". En la ventana emergente, elige "Archivo de texto" y nómbralo con una extensión ".json", como "datos.json".
5. Edita el archivo JSON: Abre el archivo JSON recién creado y utiliza la sintaxis de JSON para definir la estructura de tus datos. Puedes crear objetos JSON, matrices, y agregar pares clave-valor según tus necesidades.

**Explicación del código NoSQL del proyecto en Visual Studio Code utilizando JSON.**

1. Se creo un archivo con el nombre de *proyecto.json* en Visual Studio Code.
2. El código es una representación de un esquema de base de datos en formato JSON. Es una definición de esquema para una base de datos NoSQL utilizando un framework que utiliza el término "Schema".
3. Se comenzó a escribir el código del proyecto.

****

* **"t\_tusuario":** Es una colección o tabla llamada "t\_tusuario" para identificar el tipo de usuario, ya sea estudiante o docente. Contiene los siguientes campos:
* "nom": Un campo de tipo String que representa el nombre del usuario.
* "cargo": Un campo de tipo String que representa el cargo del usuario.
* "usuario\_id": Un campo que es una referencia a otra colección o tabla llamada "t\_usuario".
* **"t\_usuario":** Es otra colección o tabla llamada "t\_usuario". Contiene los siguientes campos:
* "nom": Un campo de tipo String que representa el nombre del usuario.
* "contra": Un campo de tipo String que representa la contraseña del usuario.
* **"tcontenido":** Es otra colección o tabla llamada "tcontenido". Contiene los siguientes campos:
* "t\_cont": Un campo de tipo String que representa el contenido.
* "tema": Un campo de tipo String que representa el tema.
* "usuario\_id": Un campo que es una referencia a otra colección o tabla llamada "t\_usuario".
* **"tactividades":** Es otra colección o tabla llamada "tactividades". Contiene los siguientes campos:
* "nom": Un campo de tipo String que representa el nombre de la actividad.
* "t\_act": Un campo de tipo String que representa el tipo de actividad.
* "nota": Un campo de tipo Number que representa la nota de la actividad.
* "cod\_cont": Un campo que es una referencia a otra colección o tabla llamada "tcontenido".



* **"tevaluacion":** Es otra colección o tabla llamada "tevaluacion". Contiene los siguientes campos:
* "nota": Un campo de tipo Number que representa la nota de la evaluación.
* "time": Un campo de tipo Number que representa el tiempo de la evaluación.
* "cod\_cont": Un campo que es una referencia a otra colección o tabla llamada "tcontenido".
* "cod\_act": Un campo que es una referencia a otra colección o tabla llamada "tactividades".
* "trtema\_eval": Es un campo de tipo objeto que contiene referencias a otras colecciones o tablas, como "tcontenido", "tevaluacion" y "tactividades".

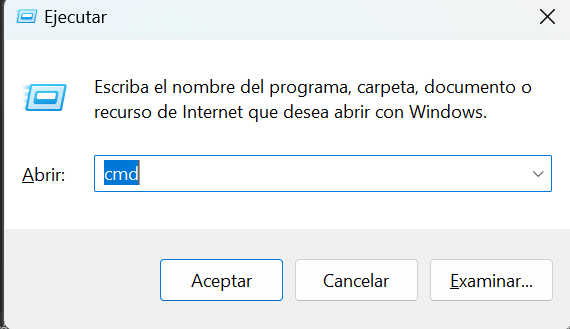
1. La estructura de cada colección se define mediante pares clave-valor, donde la clave representa el nombre del campo y el valor representa el tipo de datos del campo.

**¿Qué es una API?**

Una API es un conjunto de reglas y protocolos que permite que distintas aplicaciones o sistemas se comuniquen y compartan información entre sí. De esta manera una API actúa como un intermediario entre programas informáticos, en la que se intercambian datos realizando tareas de forma coordinada. Además, cualquier API se basan en estándares HTTP, para comunicarse entre sistemas, por lo que las APIs están estrecha mete relacionadas con los métodos GET, PUT, POST Y DELETE, estos métodos HTTP (GET, PUT, POST, DELETE) permiten que los desarrolladores interactúen con una API de manera coherente y realizar operaciones comunes en los recursos que maneja la API. Cada método tiene un propósito específico en relación con la manipulación de los datos y recursos expuestos por la API. Es importante destacar que el uso de estos métodos puede variar según el diseño y la implementación de cada API en particular.

**Procedimiento para crear una API en un ordenador, usando Node.js y Express.**

En este apartado se pretende mostrar el proceso que se realizó para crear una API:

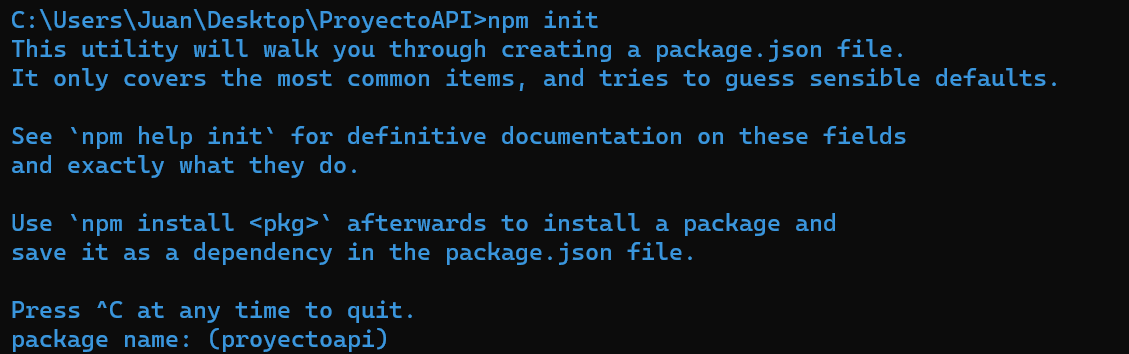
1. Se verifico que Node.js estuviera instalado en el pc, para conocer si Node está instalado, abrimos el CMD de nuestro ordenador a través del comando WIN + R, y a aparecerá un apartado como el que se ve a continuación:
2. Al ejecutar este comando se abrirá la pestaña de CMD, en la cual escribiremos el siguiente comando *node -v,* para verificar que estamos trabajando con node. Texto

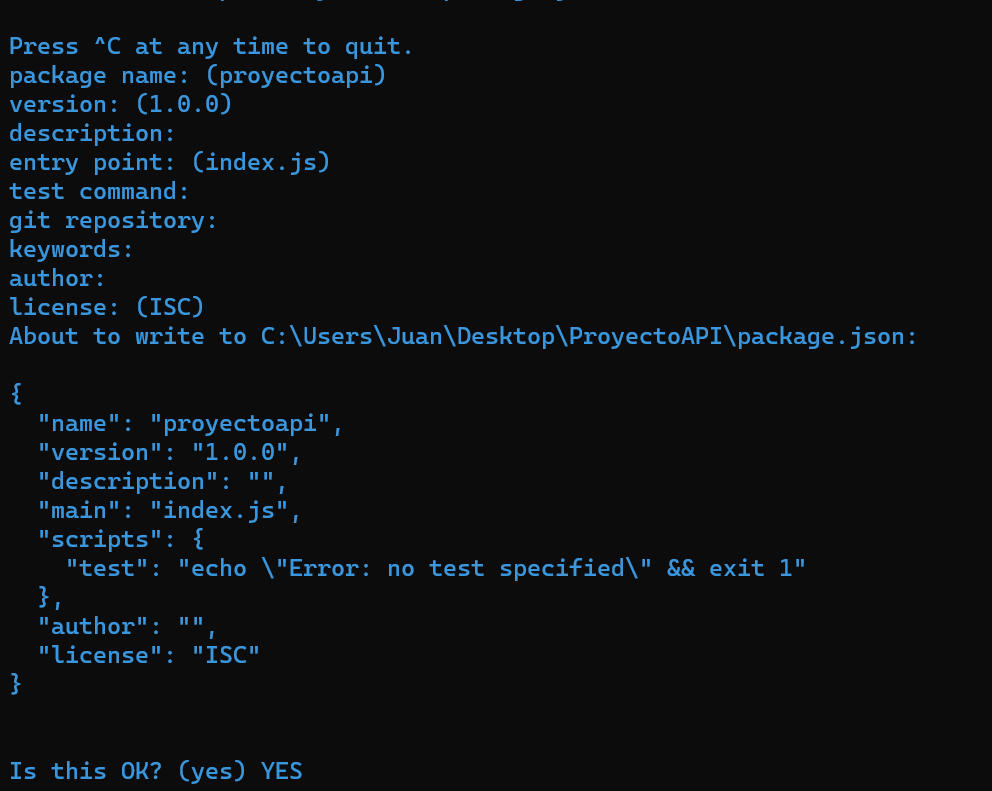
   Descripción generada automáticamente

Cabe resaltar que en este caso me ubique en la carpeta de escritorio, navegando entre carpetas a treves de comandos de CMD, use *cd y el nombre de la carpeta*  para ubicarme en la carpeta deseada. Luego de este paso ejecutamos el comando *node -v,* para conocer la versión con la que estamos trabajando en este caso usamos la versión de v18.16.0.

1. En este paso es necesario crear nuestra carpeta en el escritorio a través del comando *mkdir proyectoAPI,* use ese nombre para crear mi carpeta donde estaría ubicada mi APITexto

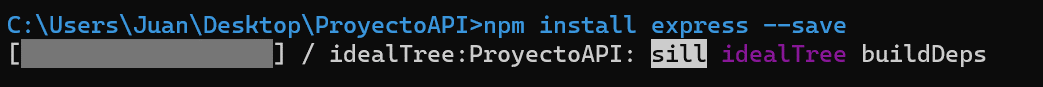
   Descripción generada automáticamente con confianza media
2. Luego de haber creado mi carpeta accedo a mi carpeta por medio del comando *cd proyectoapi,* se vería de esta forma luego de ejecutar el comando
3. Una vez ubicado en la carpeta creamos nuestro primer archivo json, debemos instalar el paquete.json a través del siguiente comando *npm init*, luego de ejecutar ese comando automáticamente instalara el paquete.json a nuestra API.

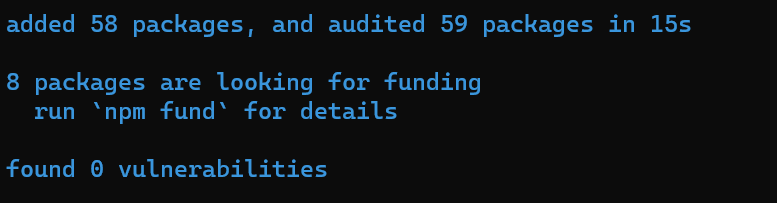




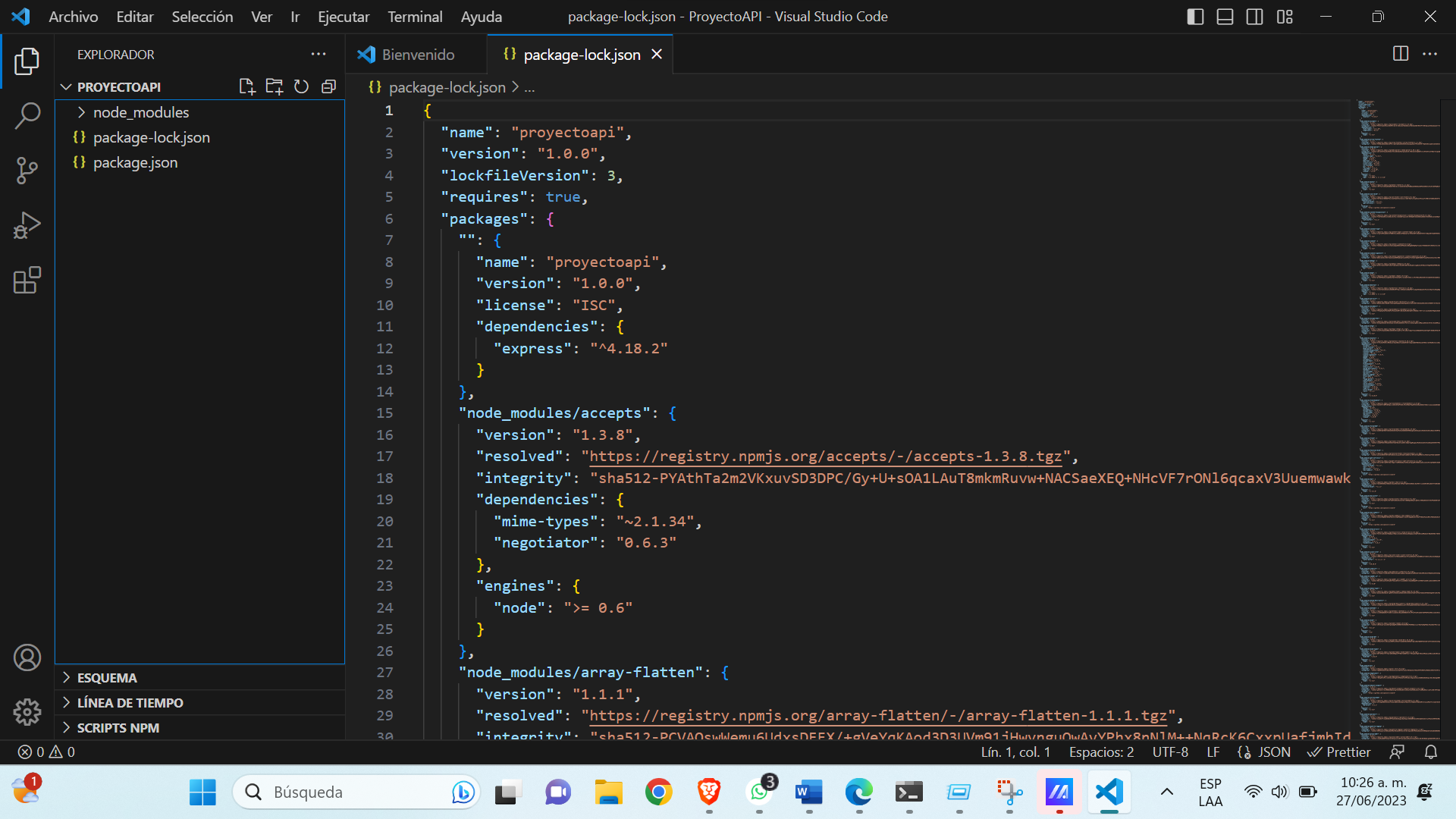
Luego se da enter a cada ítem y estaría instalado nuestro paquete json.

1. En este paso debemos seguir instalando Express para nuestra API, a través del comando *npm install express –save se intalaran los paquetes necesarios para nuestra API.*



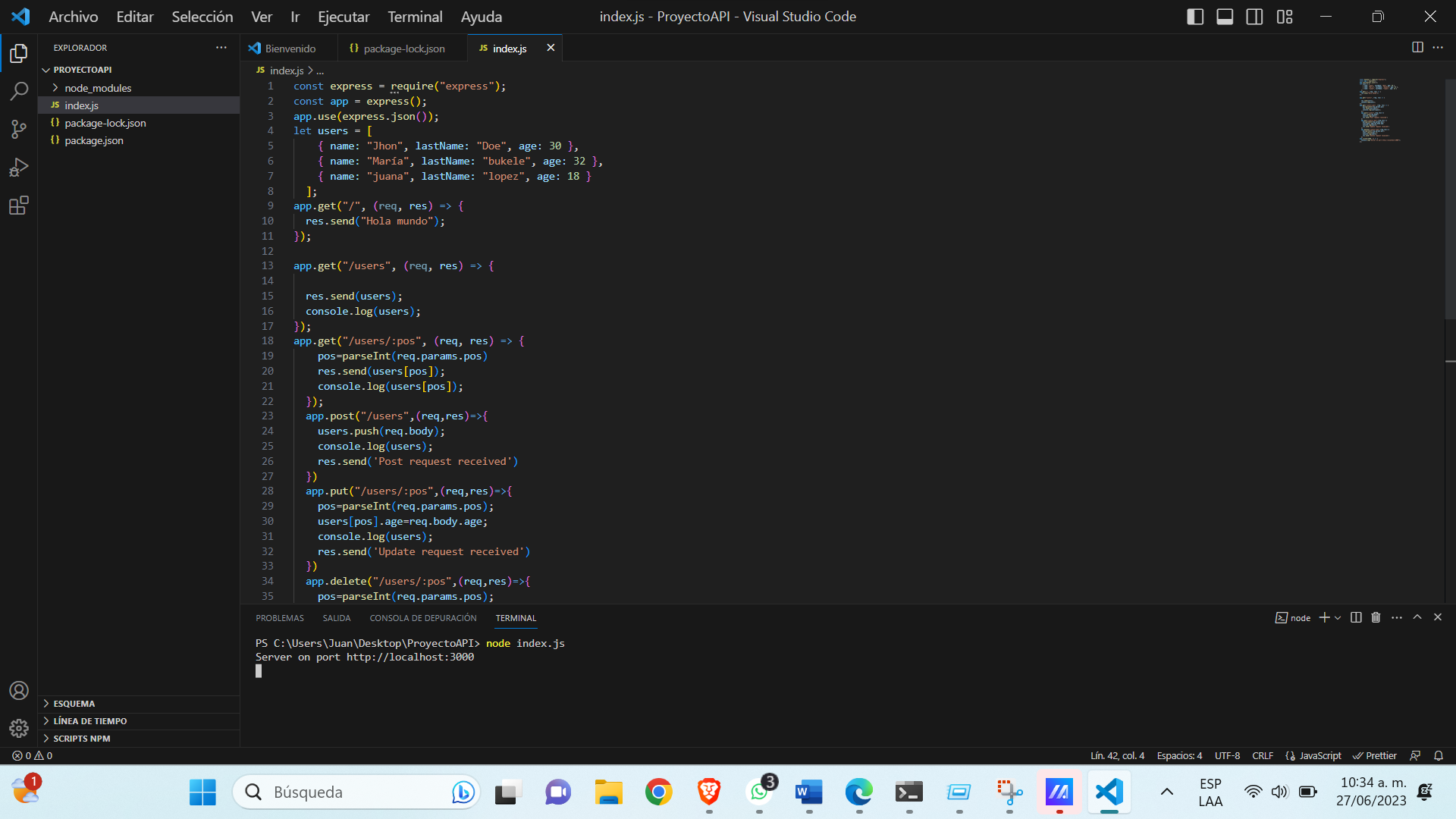


1. Luego abrimos nuestro editor, con el comando *code .* , una vez insertado ese comando se abrirá nuestro editor en una pestaña y se visualizaran los archivos descargados



De esa manera estarían listos los paquetes necesarios para crear nuestra API, según lo anterior ya creada la API con los métodos, GET, POST, PUT, Y DELETE se vería de esta forma,

Si logran notar se creó un archivo index.js



Al ejecutar esta API a través de la terminal se escribe *node index.js* en el navegador de preferencia mostrara en siguiente mensaje*.*



**En el ejercicio de crear una API los métodos quedaron de esta forma**

**Texto

Descripción generada automáticamente**

En este apartado encontramos los métodos necesarios usados para nuestra API REST, en la cual se pueden visualizar, las tablas de “contenido”, y los métodos correspondientes para crear, editar, guardar y borrar cada aspecto necesario para su uso.

Anexamos más imágenes que corresponden a router.js, en la que se aprecia los demás métodos

Texto

Descripción generada automáticamente

En este apartado notamos la tabla de “actividades” y “t\_tusuario”, y sus métodos correspondientes a su funcionamiento.

**Texto

Descripción generada automáticamente**

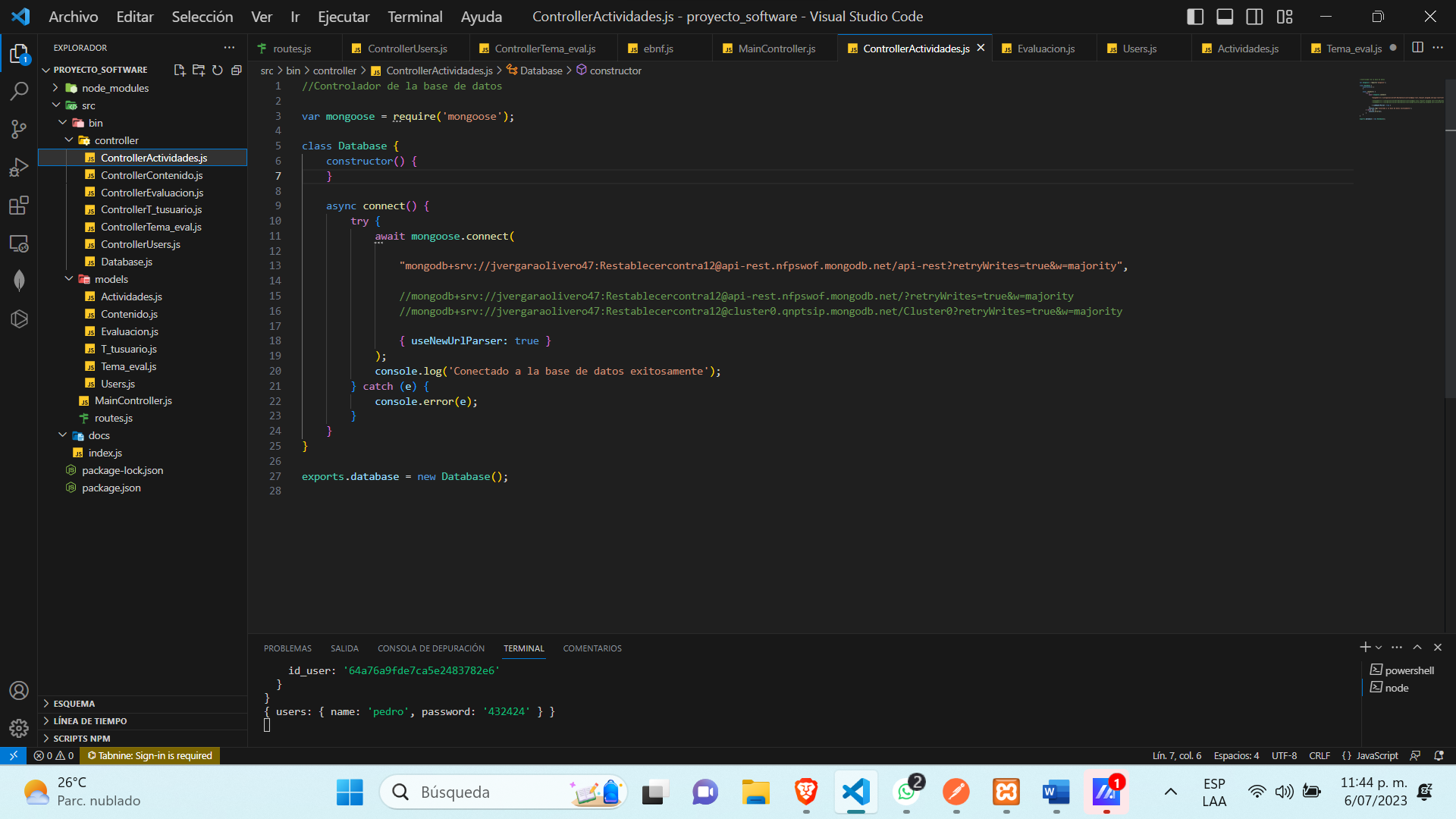
En este apartado encontramos más líneas de código que corresponden a las tablas de “Tema\_eval”

De igual forma anexamos el componente Maincontroler.js, donde se hace referencia a cada método que se quiere implementar en la API REST

**Captura de pantalla de computadora

Descripción generada automáticamente**

A demás en la imagen podemos notar en la parte derecha, que anexamos más carpetas, models, en donde se almacenaron las tablas que queríamos mostrar en moongo db atlas.

Luego de esto también hay que mencionar que la para que la API REST pudiera agregar, eliminar y editar según los métodos, fue necesario crear otra carpeta que lleva por nombre controllers.js, la que almacenaba de forma ordenada los controladores de cada tabla. A continuación, mostramos la imagen de uno de los controladores. 

Cabe resaltar que algunas de las limitaciones que se obtuvieron con esta API, fue la conexión a internet, puesto que es necesario contar con un internet rápido para obtener buenos resultados en la sincronización con moongo atlas.

Por ultimo se muestran las tablas creadas automáticamente gracia a la conexión con moongo atlas;

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Rutas amigables**

**Users**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Ruta | Datos de entrada | Datos de salida | Descripción |
| GET | /users |  | {  "\_id": "64a767a9de7ca5e2483782ce",  "name": "Luiz adrian",  "password": "1234567",  "\_\_v": 0  },  {  "\_id": "64a76994de7ca5e2483782d9",  "name": "Luis antonio",  "password": "765432",  "\_\_v": 0  },  {  "\_id": "64a769fade7ca5e2483782de",  "name": "pedro manuel",  "password": "98765",  "\_\_v": 0  },  {  "\_id": "64a76a9fde7ca5e2483782e6",  "name": "enrique",  "password": "56789",  "\_\_v": 0  },  "\_id": "64a76accde7ca5e2483782eb",  "name": "camilo",  "password": "432422",  "\_\_v": 0  } | El método GET, es usado para mostrar la información que contienen la base de datos referente a los usuarios. |
| GET | /users/:id/ | id | {  "\_id": "64a76a9fde7ca5e2483782e6",  "name": "enrique",  "password": "56789",  "\_\_v": 0  }, | Esta ruta devuelve la información de un usuario a través de un id almacenado en la base de datos |
| Post | /users/ | {      "users":{          "name":"camilo",          "password": "432422"      }  } | El usuario fue creado exitosamente | Este método se usó para agregar datos en usuarios, en la base de datos. |
| Put | /users/:id | id | {      "users":{          "name":"julian",          "password": "432422"      }  } | Este método sirve para editar información de un id especifico de un usuario ya creado. |
| Delete | /users/ |  | Todos los usuarios han sido eliminados | Esta ruta elimina todos los usuarios de la base de datos |
| Delete | /users/:id | id | {      "message": "Usuario ha sido Borrado"  } | Este método sirve para borrar información de los usuarios. A través de su id. |

**Contenido**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Ruta | Datos de entrada | Datos de salida | Descripción |
| GET | /contenido/ |  | {  "t\_cont" : "String",  "tema" : "String",  "id\_users" : "ObjectId"  } | El método GET, es usado para mostrar la información que contienen la base de datos referente a los contenidos. |
| GET | / contenido/:id/ | id | {  "t\_cont" : "String",  "tema" : "String",  "id\_users" : "ObjectId"  } | Esta ruta devuelve la información de un contenido a través de un id almacenado en la base de datos |
| Post | / contenido/ | {  "users":{  "name":"camilo",  "password": "432422"  }  } | El usuario fue creado exitosamente | Este método se usó para agregar datos en contenidos, en la base de datos. |
| Put | / contenido/:id | id | El usuario ha sido actualizado correctamente | Este método sirve para editar información de un id especifico de un contenido ya creado. |
| Delete | / contenido/ |  | Todos los usuarios han sido eliminados | Esta ruta elimina todos los contenidos de la base de datos |
| Delete | / contenido/:id | id | {      "message": "Usuario ha sido Borrado"  } | Este método sirve para borrar información de los contenidos. A través de su id. |

**Actividades**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Ruta | Datos de entrada | Datos de salida | Descripción |
| GET | /actividades/ |  | {  "t\_cont" : "arrastrar y soltar",  "tema" : "Conceptos",  "id\_users" : "64a76a9fde7ca5e2483782e6"  } | El método GET, es usado para mostrar la información que contienen la base de datos referente a las actividades. |
| GET | /actividades/:id/ | id | {  "t\_cont" : "arrastrar y soltar",  "tema" : "Conceptos",  "id\_users" : "64a76a9fde7ca5e2483782e6"  } | Esta ruta devuelve la información de una actividad a través de un id almacenado en la base de datos |
| Post | /actividades/ | {  "t\_cont" : "arrastrar y soltar",  "tema" : "Conceptos",  "id\_users" : "64a76a9fde7ca5e2483782e6"  } | El usuario fue creado exitosamente | Este método se usó para agregar los datos de una actividad, en la base de datos. |
| Put | /actividades/:id | id | La actividad ha sido actualizada correctamente | Este método sirve para editar información de un id especifico de una actividad ya creada. |
| Delete | /actividades/ |  | Todas las actividades han sido eliminadas | Esta ruta elimina todas las actividades de la base de datos |
| Delete | /actividades/:id | id | {      "message": "la actividad ha sido Borrada"  } | Este método sirve para borrar información una actividad, a través de su id. |

**Evaluación**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Ruta | Datos de entrada | Datos de salida | Descripción |
| GET | /evaluación/ |  | {  "nota": Number,  "time": String,  "cod\_contenido": "ObjectId",  "cod\_actividades": "ObjectId"  **}** | El método GET, es usado para mostrar la información que contienen la base de datos referente a las evaluaciones. |
| GET | /evaluacion/:id/ | id | {  "nota": Number,  "time": String,  "cod\_contenido": "ObjectId",  "cod\_actividades": "ObjectId"  **}** | Esta ruta devuelve la información de una evaluación a través de un id almacenado en la base de datos |
| Post | /evaluación/ | {  "nota": "2",  "time": "3h",  "id\_contenido": "64a76aedde7ca5e2483782ee",  "id\_actividades": "64a76accde7ca5e2483782eb"  } | La evaluación fue creada exitosamente | Este método se usó para agregar los datos de una evaluación, en la base de datos. |
| Put | /evaluación/:id | id | La evaluación ha sido actualizada correctamente | Este método sirve para editar información de un id especifico de una evaluación ya creada. |
| Delete | /evaluación/ |  | Todas las evaluaciones han sido eliminadas | Esta ruta elimina todas las evaluaciones de la base de datos |
| Delete | /evaluación/:id | id | {      "message": "la evaluación ha sido Borrada"  } | Este método sirve para borrar información una evaluación, a través de su id. |

**T\_usuario**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Ruta | Datos de entrada | Datos de salida | Descripción |
| GET | /t\_tusuario/ |  | {  "nom": "String",  "cargo": "String",  "id\_user": "ObjectId"  **}** | El método GET, es usado para mostrar la información que contienen la base de datos referente a los tipos de usuarios. |
| GET | /t\_tusuario/:id/ | id | {  "nom": "String",  "cargo": "String",  "id\_user": "ObjectId"  **}** | Esta ruta devuelve la información de un tipo de usuario a través de un id almacenado en la base de datos |
| Post | /t\_tusuario/ | {  "nom": "pedro manuel",  "cargo": "Estudiante",  "id\_user": "64a76a9fde7ca5e2483782e6"  } | El usuario fue creado exitosamente | Este método se usó para agregar los datos de un tipo de usuario, en la base de datos. |
| Put | /t\_tusuario/:id | id | El usuario ha sido actualizado correctamente | Este método sirve para editar información de un id especifico de un usuario ya creado. |
| Delete | /t\_tusuario/ |  | Todos los usuarios han sido eliminados | Esta ruta elimina todos los usuarios de la base de datos |
| Delete | /t\_tusuario/:id | id | {      "message": "El usuario ha sido Borrado"  } | Este método sirve para borrar información un usuario, a través de su id. |

**Tema\_eval**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Método | Ruta | Datos de entrada | Datos de salida | Descripción |
| GET | /tema\_eval/ |  | {  "cod\_contenido": "String",  "cod\_evaluacion": "String",  "cod\_actividades": "String"  **}** | El método GET, es usado para mostrar la información que contiene la base de datos referente a los temas |
| GET | /tema\_eval/:id/ | id | {  "cod\_contenido": "String",  "cod\_evaluacion": "String",  "cod\_actividades": "String"  **}** | Esta ruta devuelve la información de los id de cada tema a través de un id almacenado en la base de datos |
| Post | /tema\_eval/ | {  "cod\_contenido": "String",  "cod\_evaluacion": "String",  "cod\_actividades": "String"  } | El tema relaciones fue creado exitosamente | Este método se usó para agregar los datos de temas, en la base de datos. |
| Put | /tema\_eval/:id | id | El tema ha sido actualizado correctamente | Este método sirve para editar información de un id especifico de un tema ya creado. |
| Delete | /tema\_eval/ |  | Todos los temas han sido eliminados | Esta ruta elimina todos los temas de la base de datos |
| Delete | /tema\_eval/:id | id | {      "message": "El tema ha sido Borrado"  } | Este método sirve para borrar información un tema, a través de su id. |

**Referencias bibliográficas**

Servidor web. ingeniatic. (n.d.). https://www.etsist.upm.es/estaticos/ingeniatic/index.php/tecnologias/item/589-servidor-web%3Ftmpl=component&print=1.html#:~:text=El%20primer%20servidor%20Web%20fue,d%C3%ADa%20a%20d%C3%ADa%20la%20informaci%C3%B3n.

Click-IT. (n.d.). Breve historia del nacimiento de las bases de Datos. Click. https://click-it.es/breve-historia-del-nacimiento-de-las-bases-de-datos/#:~:text=Sus%20or%C3%ADgenes&text=Pero%20si%20hablamos%20de%20bases,y%20posteriormente%20a%20la%20contabilidad.

¿Qué es una api y cómo funciona?. Red Hat - We make open source technologies for the enterprise. (n.d.). https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces#historia-de-las-api

Que Es un framework, Historia y Más Detalles. Blog Nube Colectiva. (n.d.). https://blog.nubecolectiva.com/que-es-un-framework-historia-y-mas-detalles/#

Caro, M., Toscano, R. (2009). MODESEC: Model for educational software development bases on skills. En J. Sánchez (Ed.): Nuevas Ideas en Informática Educativa, Volumen 5, pp. 188 – 200, Santiago de Chile.