

**MEMORIA FINAL DEL PROYECTO
PROYECTO DE APLICACIÓN WEB RTEA SCHOOL
(RTEA)**

**CICLO FORMATIVO DE GRADO SUPERIOR
DESARROLLO DE APLICACIONES WEB**

AUTOR: CLAUDIA GARCÍA JAÉN

TUTORES:
JESÚS VIVES CÉSPEDES
JOSE ANTONIO NAVARRETE SANCHEZ

COORDINADOR:
JULIO MARTÍNEZ LUCAS

ÍNDICE

1. INTRODUCCIÓN.....	5
2. INTRODUCCIÓN EN INGLÉS.....	5
3. OBJETIVOS.....	5
3.1. Objetivos fase actual.....	5
3.2. Objetivos fases futuras.....	5
4. PLANIFICACIÓN.....	5
4.1. Tabla de hitos.....	6
4.2. Diagrama de Gantt.....	6
5. ANÁLISIS.....	6
5.1. Estado del arte.....	6
5.2. Funcionalidades.....	6
6. DISEÑO.....	6
6.1. Requisitos técnicos.....	6
6.2. Arquitectura web.....	7
6.3. Diseño back-end.....	7
6.3.1. Modelo de datos.....	7
6.3.2. Servicios REST.....	7
6.3.3. Paquetes adicionales.....	7
6.4. Diseño front-end.....	7
6.4.1. Mock-ups.....	7

6.4.2. Guía de estilos.....	8
6.4.3. Paquetes adicionales.....	8
6.4.4. Capturas de la aplicación.....	8
7. IMPLEMENTACIÓN.....	8
7.1. Servidor.....	8
7.2. Cliente.....	8
8. DESPLIEGUE.....	9
8.1. Modelo de despliegue utilizado.....	9
8.2. Datos iniciales y configuración.....	9
8.3. Pasos para el despliegue.....	9
8.4. Proveedores y servicios utilizados.....	9
9. HERRAMIENTAS UTILIZADAS.....	9

1. INTRODUCCIÓN

Hola a todos. Mi nombre es Claudia García. Me gustaría agradecerles a todos por estar aquí hoy. A continuación, comenzaré a explicar el tema de mi Trabajo de Fin de Grado. Mi proyecto se centra en RTEA Web School. En este trabajo me he centrado en el apartado frontend y sobre todo en el tema visual, ya que esta empresa al ser una academia online que vende cursos el tema marketing y publicidad y el apartado visual es el más importante. A continuación, comenzaré a explicar cómo se realiza todo el desarrollo que hay detrás de esta aplicación web. Espero que os guste y os resulte interesante.

2. INTRODUCCIÓN EN INGLÉS

Hello, everyone. My name is Claudia García. I would like to thank you all for being here today. Next, I will begin to explain the topic of my Final Year Project. My project focuses on RTEA Web School. In this work, I have focused on the frontend section and mostly on the visual issue, since this company, being an online academy that sells courses, the marketing and advertising issue and the visual section is the most important. Next, I will begin to explain how all the development behind this web application is made. I hope you like it and find it interesting.

3. OBJETIVOS

El propósito básicamente de este proyecto, es el realizar de manera completa y bien estructurada una aplicación web que supla las necesidades que la empresa de RTEA me planteó al presentarme este proyecto para que yo lo realizara. Esta empresa, es una academia online, que se encarga de la venta de cursos, de impartir mentorías, y la cual consta de un programa de afiliados que está actualmente en construcción aún y que se irá desarrollando poco a poco con el tiempo. Actualmente,

ellos están como colaboradores del grupo Telefónica, participando activamente como empresa seleccionada, dentro del programa de StartUps que han creado sus inversores. Por lo tanto, se debe de tener en cuenta que a lo largo de este proyecto muchas de las cosas siguen todavía en una fase muy temprana de desarrollo y maduración, ya que faltan muchas cosas que la empresa tiene que seguir construyendo y asentando de manera correcta.

3.1. Objetivos fase actual

Los objetivos que acordé con la empresa para que formaran parte y que contarán dentro de este proyecto de final de grado, se dividen en dos partes:

- Parte de frontend: En esta parte, se realizarían todas las páginas web desde cero que actualmente necesita la empresa hasta día de hoy, incluyendo el branding y el diseño creado mediante alguna referencia. Se me especificó que hasta que no fuera necesario más adelante, para mostrar estas páginas al público y publicitarlas de manera visual a través de sus redes sociales, necesitarían estas páginas por ahora únicamente, en formato de ordenador (Full HD) (1920x1080) (1080p). Ya que aunque para el proyecto estas se quedarán finalizadas, más adelante se harán algunos pequeños cambios en las páginas, y sí que se necesitará adaptabilidad para móvil, tablet y formato ordenador de menor resolución. Finalmente, se quedaron cerradas para realizar, 13 páginas web, más las web con toda la información personal de cada uno de los docentes.

- Parte de backend: En esta parte, se realizaría una base de datos que pudiesen gestionar 3 tipos de cargos o roles, los cuales gestionan la empresa (administrador, profesor y estudiante). Cada uno de ellos, entraría mediante su email y contraseña a la BD y accedería a su backoffice, y desde este, podrían gestionar las diferentes acciones que se les permite según su rol. Las acciones que podría realizar cada rol

serían las siguientes:

- Administrador: Tendrá acceso a añadir, listar y buscar por campos de las tablas de usuarios, administradores, profesores, alumnos, cursos, lecciones, roles, grupos de trabajo y al listar y buscar del resto de tablas.
- Profesor: Tendrá acceso al añadir, listar y buscar de sus propios cursos así como de las lecciones de estos, de sus grupos de trabajo, y al listar y buscar de sus alumnos (alumnos los cuales han comprado su/s cursos). Y por último, tendrá la posibilidad de habilitar mediante la pestaña de lecciones de los cursos, de habilitar y desabilitar los enlaces para que sus alumnos accedan a los videos del curso.
- Alumno: Tendrá acceso al listar y buscar de sus cursos (cursos que ha comprado), de sus profesores (profesores creadores de los cursos que ha comprado) con un listado de la información básica del profesor (nombre, email) para si necesita en algún momento ponerse en contacto con alguno de ellos.

3.2. Objetivos fases futuras

En este caso, para este proyecto, se quedan muchos de los objetivos fuera del proyecto de final de grado, ya que son muchos, y algunos forman parte de la propiedad intelectual de la empresa, ya que son datos privados y gestiones internas que no se han considerado oportunas aportar a un proyecto con finalidad educativa. De cualquier manera, si que se pueden decir que algunos de los objetivos que se

quedan para desarrollarlos de manera externa a este proyecto son algunos como: mayor optimización de todo el contenido multimedia que forma parte de las páginas, creación de más cantidad de marketing e inclusión de más fotografías relacionadas con este, no solo de tema corporativo, cambio de gran parte de las estructuras html fijas a contenido dinámico gestionado por el backoffice y reestructuración de la BD para añadir el futuro programa de afiliados, programa de facturación, etc.

4. PLANIFICACIÓN

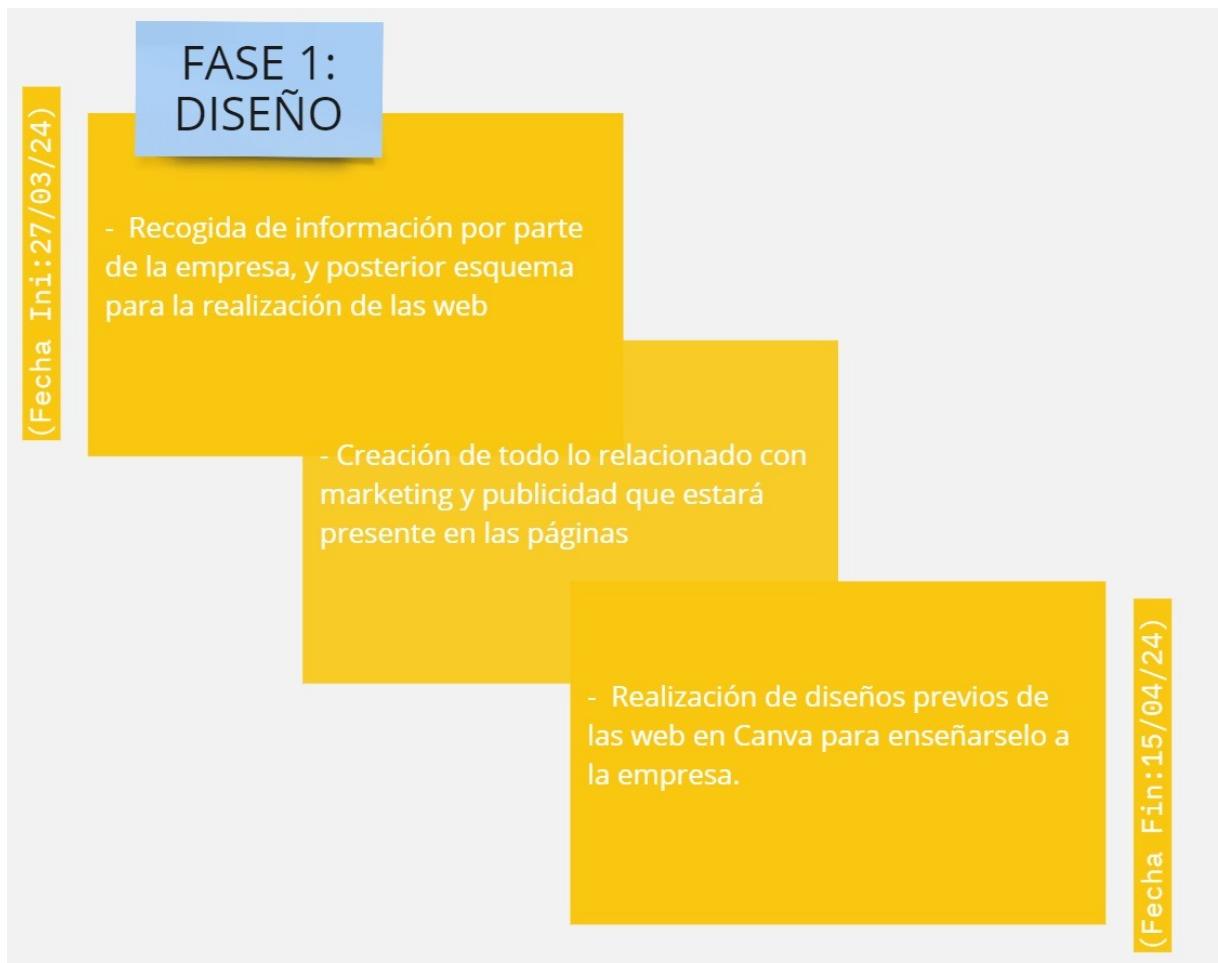
Tareas que ir realizando por orden:

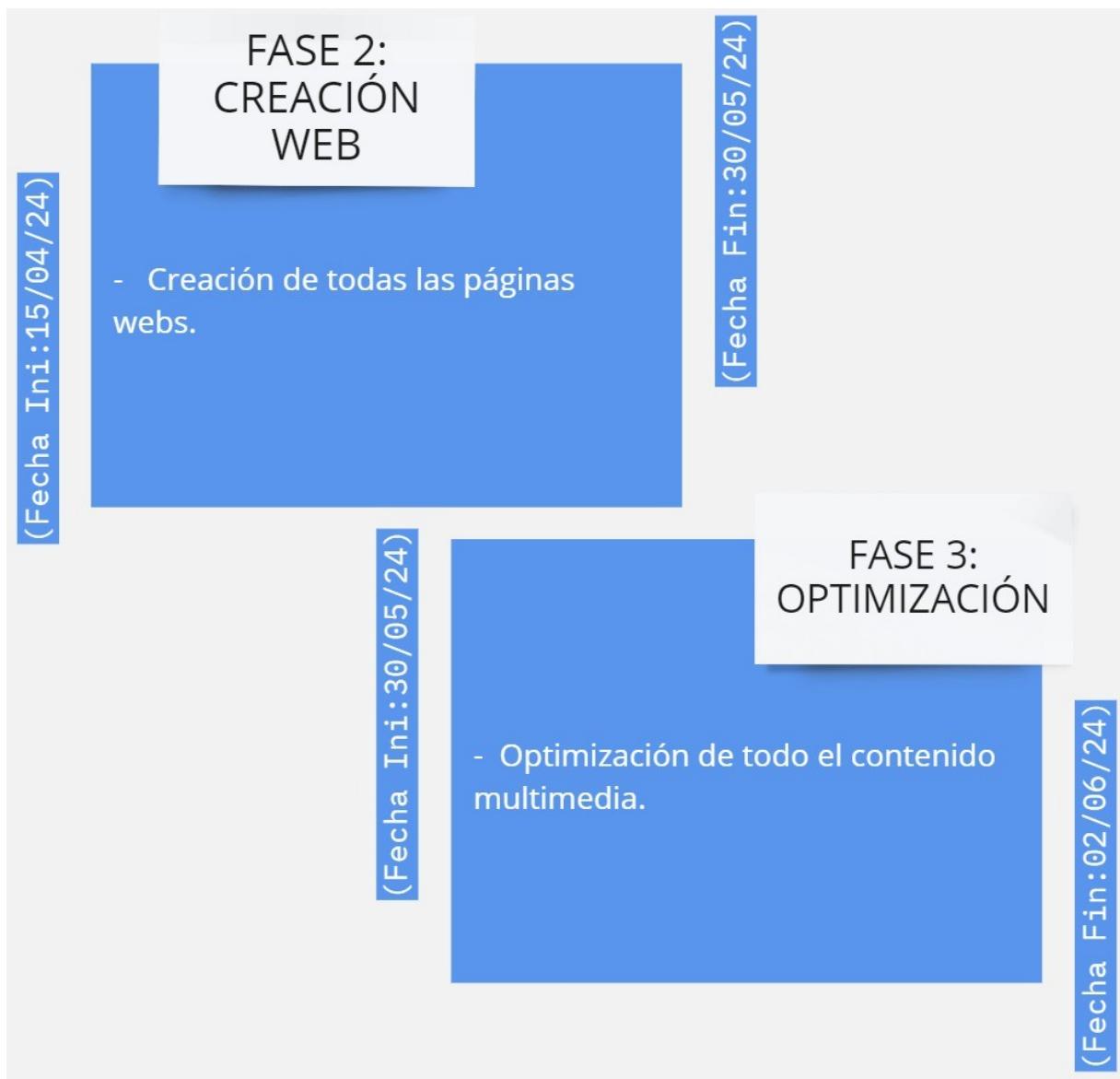
- 1) Recogida de información por parte de la empresa, y posterior esquema para la realización de las web. (Duración aprox: 1 días)
- 2) Creación de todo lo relacionado con marketing y publicidad que estará presente en las páginas. (Duración aprox: 5 días)
- 3) Realización de diseños previos de las web en Canva para enseñarselo a la empresa. (Duración aprox: 12 días)
- 4) Creación de las webs. (Duración aprox: 45 días)
- 5) Optimización de todo el contenido multimedia. (Duración aprox: 3 días)
- 6) Diagrama de flujo, esquema relacional de la BD y posterior creación de esta en un servidor en local. (Duración aprox: 1 días)
- 7) Creación del MVC en PHP. (Duración aprox: 4 días)

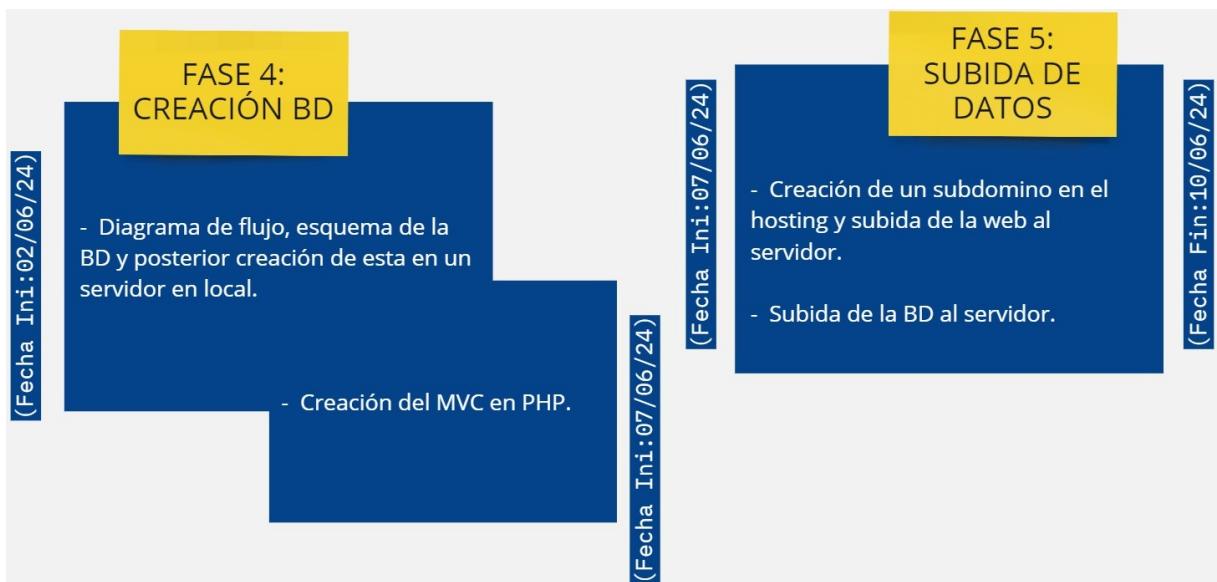
8) Creación de un subdomino en el hosting y subida de la web al servidor. (Duración aprox: 1 días)

9) Creación de la BD dentro del hosting (en mi caso, el hosting ya lo tenía contratado la empresa, por lo que solo tuve que recibir la clave de acceso e investigar un poco para hacer toda la subida de archivos al servidor). (Duración aprox: 1 días)

4.1. Tabla de hitos







4.2 Diagrama de Gantt

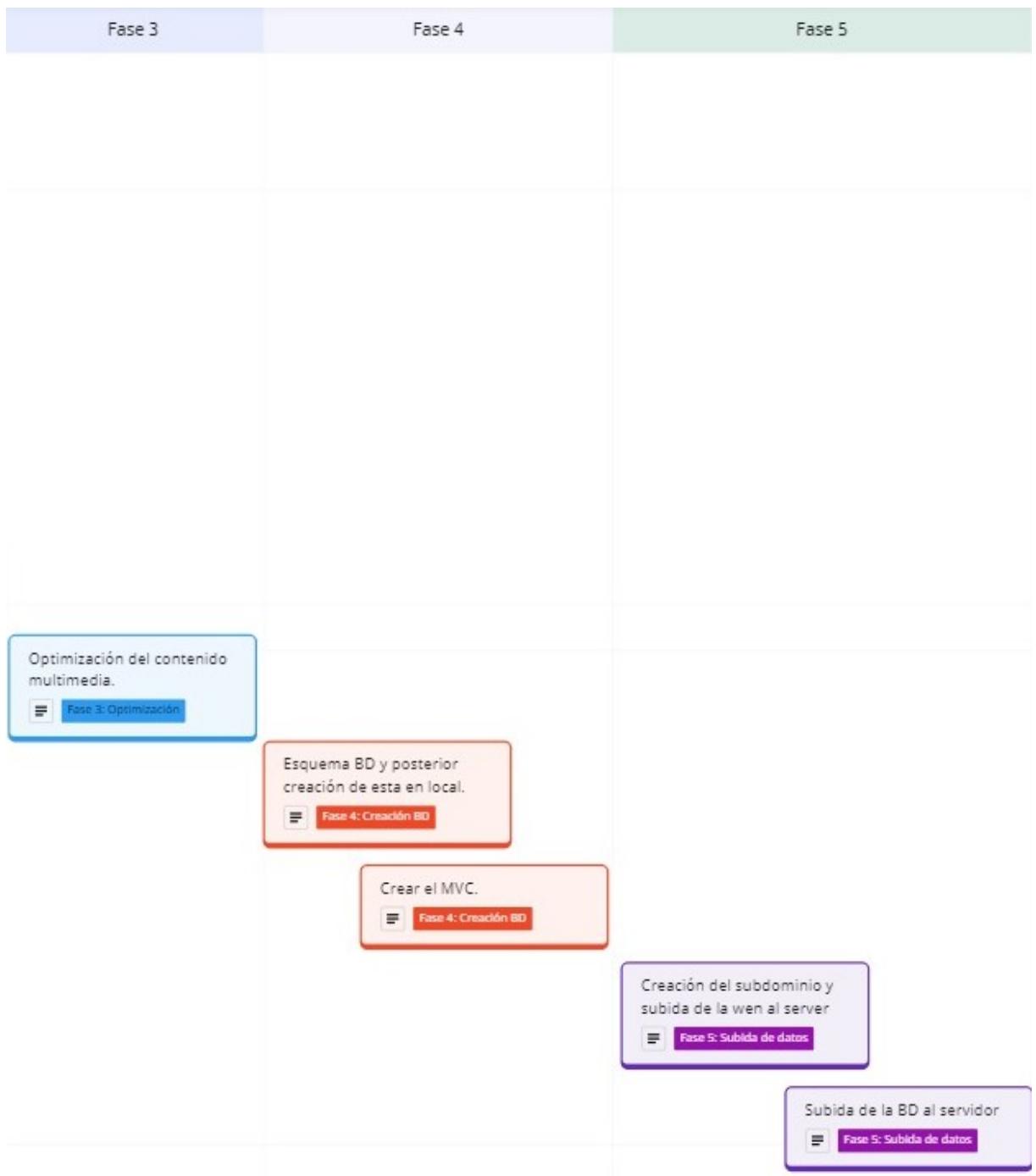
PROYECTO DE APLICACIÓN WEB RTEA SCHOOL

DAW

	Fase 1	Fase 2
Marzo	<p>Recogida de información por parte de la empresa, y posterior esquema para la realización de las web.</p> <p> Fase 1: Diseño</p>	
Abril	<p>Creación de todo lo relacionado con marketing y publicidad que estará presente en las páginas.</p> <p> Fase 1: Diseño</p> <p>Realización de diseños previos de las web en Canva.</p> <p> Fase 1: Diseño</p>	
Mayo		<p>Creación de las webs.</p> <p> Fase 2: Creación</p>

PROYECTO DE APLICACIÓN WEB RTEA SCHOOL

DAW



5. ANÁLISIS

5.1. Estado del arte

En el desarrollo de RTEA Web School, se han analizado varias soluciones existentes para identificar sus fortalezas y debilidades y así poder mejorar nuestra propia implementación. Entre las soluciones evaluadas se encuentran:

- 1) Coursera: Plataforma líder en cursos online que ofrece una experiencia de usuario pulida con funcionalidades avanzadas como seguimiento del progreso, foros de discusión y certificados. Nuestra diferencia principal radica en la personalización del contenido y la inclusión de un programa de afiliados.
- 2) Udemy: Destaca por su enfoque en la diversidad de cursos y el acceso a materiales de por vida. A diferencia de Udemy, nuestra plataforma se centra en la creación de comunidades de aprendizaje con soporte de mentores y un sistema de grupos de trabajo.
- 3) edX: Con un fuerte enfoque académico y asociaciones con universidades de renombre, edX se diferencia en su oferta de cursos acreditados. En contraste, RTEA Web School se orienta hacia el aprendizaje práctico y aplicable directamente en el entorno laboral. Estos análisis revelan que, aunque existen muchas plataformas de e-learning con características similares, RTEA Web School se distingue por su enfoque en la personalización, el apoyo continuo a través de mentores, y la integración futura de un sistema de afiliados para fomentar el crecimiento orgánico de la plataforma.

5.2. Funcionalidades

La aplicación web debe cumplir con los siguientes requisitos funcionales:

1. Gestión de usuarios:

- Registro y autenticación de usuarios (administradores, profesores, estudiantes).
- Gestión de perfiles de usuario con información personal y de contacto.

2. Gestión de cursos:

- Creación, edición y eliminación de cursos por parte de los profesores.
- Inclusión de lecciones y materiales multimedia en los cursos.
- Habilitación y deshabilitación de enlaces de videos por parte de los profesores.

3. Acceso a cursos y lecciones:

- Los estudiantes pueden acceder a los cursos adquiridos y sus respectivas lecciones.
- Funcionalidad de búsqueda y filtrado de cursos y lecciones.

4. Backoffice para diferentes roles:

- Panel de administración para gestionar usuarios, cursos, lecciones, roles, y grupos de trabajo.
- Panel de profesor para gestionar sus propios cursos, lecciones, y grupos de trabajo.
- Panel de estudiante para gestionar sus cursos adquiridos y contactar con profesores.

5. Sistema de marketing y publicidad:

- Páginas web visuales y atractivas para la promoción de la academia y sus cursos.
- Optimización de contenido multimedia para una mejor experiencia de usuario.

6. Programa de afiliados (fase futura):

- Sistema de gestión de afiliados para promover los cursos y generar ingresos adicionales.

6. DISEÑO

6.1. Requisitos técnicos

Basándonos en los requisitos funcionales, los requisitos técnicos incluyen:

1. Gestión de usuarios:

- Implementación de un sistema de autenticación seguro utilizando PHP y bases de datos relacionales.
- Uso de formularios para la gestión de perfiles con validación de datos en frontend y backend.

2. Gestión de cursos:

- Desarrollo de formularios y tablas para la creación, edición y eliminación de cursos en la base de datos.
- Implementación de un sistema de almacenamiento para materiales multimedia.

3. Acceso a cursos y lecciones:

- Desarrollo de interfaces de usuario con HTML5, CSS3 y JavaScript para la navegación y visualización de cursos.
- Implementación de sistemas de búsqueda y filtrado utilizando JavaScript y consultas SQL.

4. Backoffice:

- Creación de paneles administrativos con interfaces específicas para cada rol utilizando PHP y HTML.
- Gestión de permisos y accesos diferenciados para administradores, profesores y estudiantes.

5. Sistema de marketing y publicidad:

- Desarrollo de páginas visualmente atractivas y optimizadas con SASS y Bootstrap.
- Implementación de SEO básico y análisis de rendimiento web.

6. Programa de afiliados:

- Preparación de la base de datos y el sistema de gestión para incluir funcionalidades futuras relacionadas con los afiliados.

6.2. Arquitectura web

1) Modelo: Representa la estructura de la base de datos y gestiona la lógica de negocio.

2) Vista: Se encarga de la presentación de datos al usuario final utilizando HTML, CSS y JavaScript.

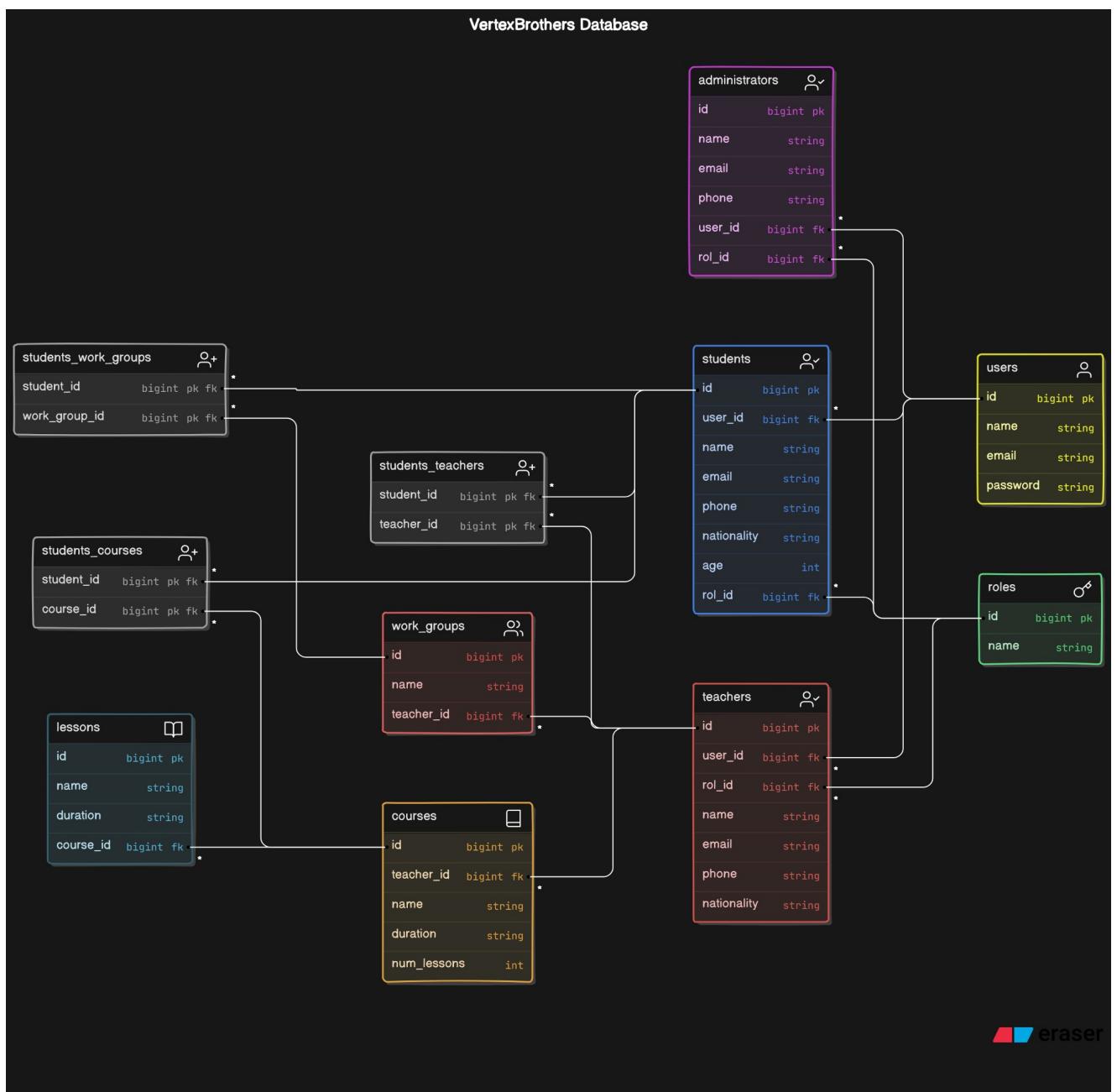
3) Controlador: Gestiona la comunicación entre el modelo y la vista, procesando las entradas del usuario y actualizando las vistas.

La implementación incluye tecnologías de contenedores para facilitar el despliegue y la escalabilidad del sistema.

6.3. Diseño back-end

- Framework utilizado: PHP puro bajo el modelo MVC.
- Configuración específica: PHP configurado para soportar sesiones seguras y operaciones con bases de datos relacionales.

6.3.1. Modelo de datos



6.3.2. Servicios REST

- Endpoints principales:

- `/api/courses` para gestión de cursos.
- `/api/lessons` para gestión de lecciones.
- `/api/users` para gestión de usuarios.

6.3.3. Paquetes adicionales

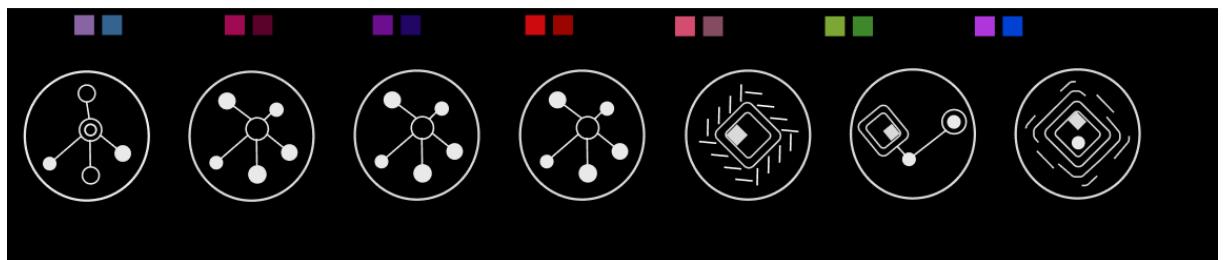
- Utilización de bibliotecas PHP para la gestión de sesiones y la interacción con la base de datos (PDO).
- Uso de bibliotecas JavaScript como jQuery para facilitar las interacciones del usuario.

6.4. Diseño front-end

- Consumo de servicios web: Utilización de AJAX para la comunicación asíncrona con el backend.
- Desarrollo de funcionalidades específicas: Implementación de formularios dinámicos, validaciones y actualizaciones de contenido en tiempo real.

6.4.1. Mock-ups

- Estos son algunos de los diseños, ya sean primeros o diseños finales hechos en Figma:





The screenshot displays a search results page for the 'CURSO AVANZADO DE UNREAL'. The results are organized into three columns, each containing a course card. Each card includes the course title, a 5-star rating, the number of hours (80h), the instructors (RTEA - Alberto García y Javier Cárceles), and a price of 249,99€. A 'Comprar' button is also present in each card.

Categoría	Etiquetas	Softwares
<input type="checkbox"/> Avanzado	<input type="checkbox"/> Modelado 3D	<input type="checkbox"/> Blender
<input type="checkbox"/> Inicial	<input type="checkbox"/> Texturizado	<input type="checkbox"/> Unreal
<input type="checkbox"/> Intermedio	<input type="checkbox"/> Environment	
	<input type="checkbox"/> Shading y Materiales	
	<input type="checkbox"/> Terrenos	
	<input type="checkbox"/> Telas y tejidos	
	<input type="checkbox"/> VFX Game Ready	

PROYECTO DE APLICACIÓN WEB RTEA SCHOOL

DAW

secciónMembresías1

CURSOS

MEMBRESÍAS

MENTORES

RESULTADOS

CREAR Cuenta

INICIAR SESIÓN

secciónMembresías2

BENEFICIOS MEMBRESÍA RTEA

ESTUDIA ONLINE A TU RITMO

Todos nuestros cursos accesibles para aprender a tu ritmo están acompañados por videos en video para que aprendas a tu ritmo.

CERTIFICADO DE CURSOS

Todos nuestros cursos tienen certificado de cumplimiento de horas, siempre y cuando realices sus evaluaciones y sean aprobadas.

MAESTROS ESPECIALISTAS

Todos nuestros profesores son expertos especialistas en cada uno de nuestros cursos, y cada uno de ellos poseen experiencia en su industria.

ESTUDIA 100% DESDE CUALQUIER PARTE

No importa tu ubicación, hoy es posible estudiar desde cualquier punto del mundo.

- Conexión directa con nuestros profesores.
- Resolución de dudas del chat modo via Discord.

- Resolución de dudas del chat modo via Discord.
- Descuentos solo para miembros.

secciónMembresías3

ACCESO A + DE 20 TUTORIALES Y VIDEOS ESPECIALIZADOS

La membresía **RTEA PREMIUM** incluye todos los contenidos regulados a partir de cursos, entrevistas exclusivas y contenido nuevo que se añade mensualmente. Toda lo que necesitas para adentrarte más profundamente dentro del mundo del 3D en una suscripción.

NUESTRA HISTORIA

En la aventura por descubrir el mundo del CGI y llegar a formar parte de un equipo entregado a la creación de escenarios realistas para TV, Cine o Videojuegos, dos artistas deciden dar un gran esfuerzo para conseguir llevar a cabo su objetivo. Desde ese momento, con una fuerte pasión y entrega, van uniendo unas fuertes lazos que decidirán su trayectoria artística y forjarán una amistad que los hará más fuertes y decididos.

De esta gran unión nacen los **VERTEX BROTHERS**.



Frame 2

L. Alberto García - Profesor

Experiencia Laboral:

- Technical VFX Artist - Sober Interactiva
- Más de 20 años de experiencia en calidad de responsable de los Departamentos de Marketing, Publicidad y Comunicación a nivel internacional.

Programas utilizados en la Industria:

- Composición: Nuke, Fusion y After Effects.
- Maquetación: Adobe (Photoshop, Illustrator, Indesign).
- Edición: Premiere y Davinci Resolve.
- Tracking: Match Move y Bonjour.
- Modelado 3D: Cinema 4D, 3D Studio MAX y Blender.
- Texturizado: Substance [Painter, Designer, Alchemist] y Quixel Mixer.

Experiencia como Artista 3D:

- Ha trabajado en cientos de proyectos de Infraestructura usando AutoCAD y 3D Studio MAX.
- Especializado en modelado HardSurface con subdivision, Midpoly, Lowpoly, dirigido 100% a entornos digitales, desarrollo de escenarios para cinematográficos y entornos de Videojuegos, Cine y Televisión, bajo entorno del motor de UNREAL ENGINE.

Javier C.M. - Profesor

Experiencia Laboral:

- Technical VFX Artist - Sober Interactiva
- Más de 20 años de experiencia en calidad de responsable de los Departamentos de Marketing, Publicidad y Comunicación a nivel internacional.

Programas utilizados en la Industria:

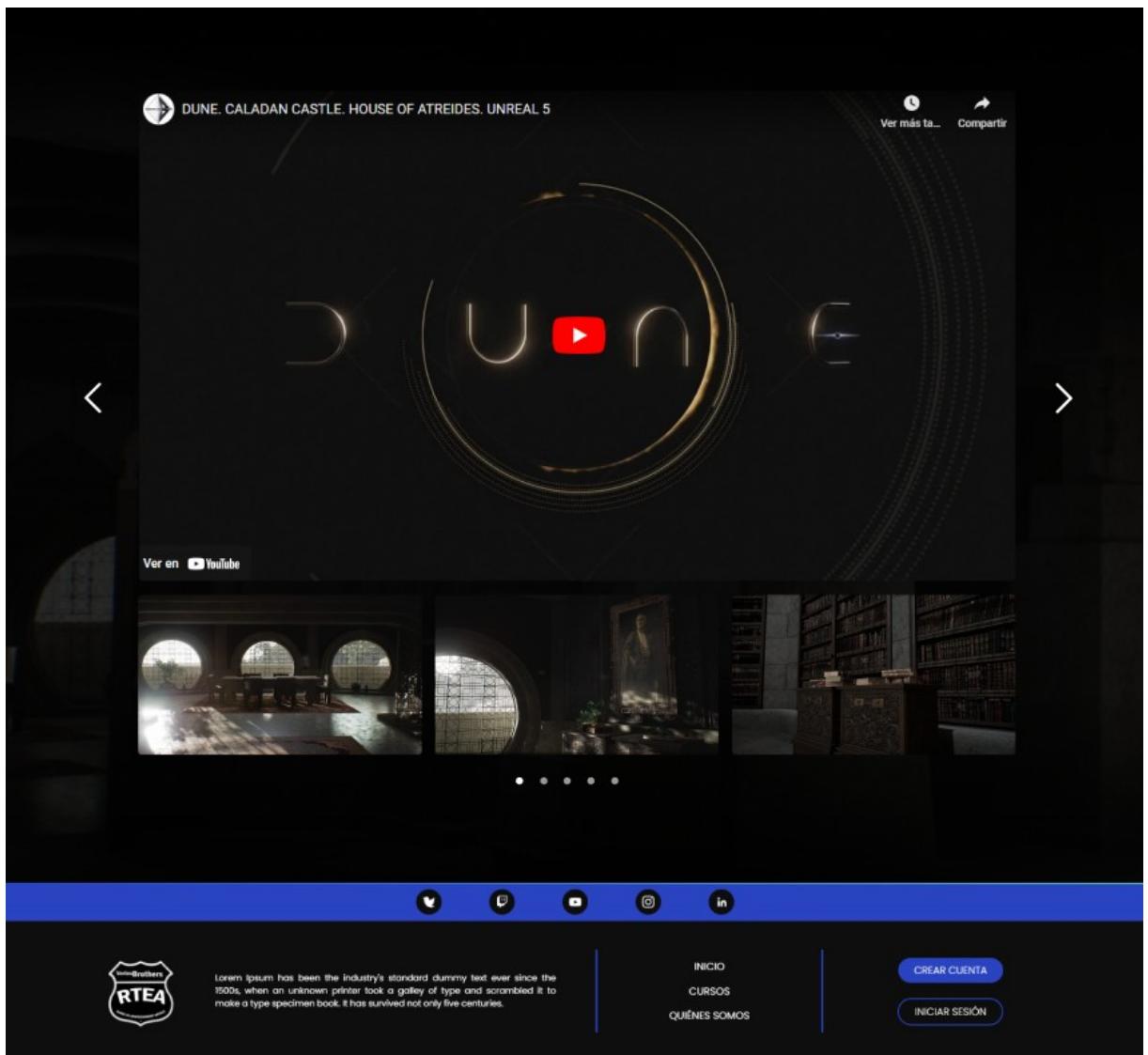
- Composición: Nuke, Fusion y After Effects.
- Maquetación: Adobe (Photoshop, Illustrator, Indesign).
- Edición: Premiere y Davinci Resolve.
- Tracking: Match Move y Bonjour.
- Modelado 3D: Cinema 4D, 3D Studio MAX y Blender.
- Texturizado: Substance [Painter, Designer, Alchemist] y Quixel Mixer.

Experiencia como Artista 3D:

- Ha trabajado en cientos de proyectos de Infraestructura usando AutoCAD y 3D Studio MAX.
- Especializado en modelado HardSurface con subdivision, Midpoly, Lowpoly, dirigido 100% a entornos digitales, desarrollo de escenarios para cinematográficos y entornos de Videojuegos, Cine y Televisión, bajo entorno del motor de UNREAL ENGINE.

Frame 3





6.4.2. Guía de estilos

- Gama de colores utilizada:

```
$color-negro: rgb(13, 13, 13);  
$color-negro2: rgb(8, 8, 8);  
$color-negro-intermedio: #0E0E0E;  
$color-negro-intermedio2: #292929;  
$color-negro-entre-intermedio-y-claro: #181818;  
$color-negro-mas-claro: rgb(32, 32, 32);  
$color-gris-claro: #a1a1a1;  
$color-gris-mas-claro: #cbcbcb;  
$color-gris-normal: #A5A5A5;  
$color-gris-oscuro: #616161;  
$color-gris-mas-oscuro: #383838;  
$color-blanco-roto: #ebebeb;  
$color-blanco-roto: #ebebeb;  
$color-blanco: #fff;  
$color-amarillo-navigationBar: rgb(223, 223, 126);  
  
$color-azul-electrico-oscuro: #16225e;  
$color-azul-electrico: #2a43c1;  
$color-azul-electrico-claro: rgb(130, 151, 255);  
$color-azul-intermedio: #5095C2;  
$color-azul-verdoso: #5BCAAB;  
$color-rojo: #bd3924;  
$color-naranja: #e26b2b;  
$color-amarillo: #faf0cc;  
$color-frozen1: #2e6a7a;  
$color-frozen2: #4895ae;  
$color-frozen3: #c3eaff;  
$color-air1: #26746e;  
$color-air2: #6ca796;  
$color-air3: #bfccc7;  
$color-earth1: #3c7a2d;  
$color-earth2: #49a74d;  
$color-earth3: #7bc77d;  
$color-magic1: #a2129a;  
$color-magic2: #c660bf;  
$color-magic3: #ffa4f9;  
  
$color-modulo1-c1: #8964A3;  
$color-modulo1-c2: #356390;
```

```
$color-modulo2-c1: #A00A52;  
$color-modulo2-c2: #59032A;  
$color-modulo3-c1: #6C0E8D;  
$color-modulo3-c2: #210666;  
$color-modulo4-c1: #CD0A0D;  
$color-modulo4-c2: #9A0500;  
$color-modulo5-c1: #D54D6E;  
$color-modulo5-c2: #834C60;  
$color-modulo6-c1: #7DA734;  
$color-modulo6-c2: #3E882C;  
$color-modulo7-c1: #AE36DA;  
$color-modulo7-c2: #0041D4;
```

- Tipografía usada:

- Poppins-Regular, Poppins-Light, Poppins-ExtraLight, Poppins-Medium, Poppins-Bold, Poppins-ExtraBold, Poppins-SemiBold, Poppins-BoldItalic

6.4.3. Paquetes adicionales

- Bootstrap: Para un diseño responsive y rápido desarrollo de la interfaz de usuario.
- jQuery: Para simplificar las manipulaciones del DOM y las peticiones AJAX.

6.4.4. Capturas de la aplicación

Index:

The screenshot shows the homepage of the RTEA School website. At the top, there is a navigation bar with links for 'Cursos', 'Profesores', 'Quienes Somos', 'Programa de Afiliados', and 'Iniciar sesión'. Below the navigation is a large banner with the text 'RTEA, on the right way with Unreal Engine' in white and blue. The banner features a background image of a jungle scene with a character. Below the banner, there is a call-to-action button 'Comienza a estudiar' and a note '*Accede a nuestros cursos'. A small video thumbnail at the bottom right is labeled 'Video Promocional Unreal Demoreal Unreal Engine 5'. In the center, there is a section for a new course: 'NUEVO LANZAMIENTO Master Creación VFX Unreal 4&5'. It includes a brief description, duration (90h 25'), level (Intermedio), student count (0 estudiantes), and buttons for 'Pronto disponible' and 'Ver detalles'. To the right of this is a slide for 'MASTER DE UNREAL CREACIÓN DE VFX PARA UNREAL 4&5', featuring a red humanoid figure standing next to a stack of books. Below these are sections for 'Cursos más vistos' showing thumbnails for various Unreal Engine courses.

¡Conoce nuestros cursos y todas sus ventajas!

Apuesta por RTEA School para aprender a desarrollar entornos hiperrealistas, cinematográficas para videojuegos, infoarquitectura, cine y televisión, broadcasting, eventos en vivo y mucho más.



Actualizate
Ponte al día con las últimas novedades y aprende nuevas habilidades.



Descuentos
Aprovecha nuestros descuentos exclusivos en algunos de nuestros cursos.



Pago rápido y seguro
Consigue todos nuestros cursos de manera rápida y segura con Stripe.



CURSO DE UNREAL
MAGIC SHADER PACKAGE



CURSO DE UNREAL
CREACIÓN DE MATERIALES UNREAL 4&5



CURSO DE UNREAL
SHADERS UNREAL 4&5 PACKAGE



CURSO DE UNREAL
FIRE SHADER PACKAGE



CURSO DE UNREAL
WATER SHADER PACKAGE



CURSO DE UNREAL
AIR SHADER PACKAGE



CURSO DE UNREAL
WATER SHADER



CURSO DE UNREAL
EARTH SHADER



CURSO DE UNREAL
FIRE SHADER PACKAGE



CURSO DE UNREAL
AIR SHADER PACKAGE



CURSO DE UNREAL
MAGIC SHADER



CURSO DE UNREAL
SHADERS UNREAL 4&5

Lo que ofrecemos en #RTEASchool



Cursos de calidad
Podrás aprender con nuestros cursos, los cuales ofrecen una gran variedad de contenidos a un precio perfecto.

[Conoce nuestros cursos](#)



Afilación
Trae a un amigo para disfrutar de nuestros cursos junto contigo y obtén un descuento en tus próximas compras.

[Programa de afiliados](#)



Nuestros profesores
Disfruta de profesores de calidad especializados en el sector que cuentan con grandes cursos.

[Nuestros profesores](#)



Obten experiencia
Te garantizamos que saldrás habiendo obtenido mucha experiencia con cada uno de nuestros cursos.

[Comienza a estudiar](#)

¡Conoce nuestra metodología!

Conoce la manera que tienen nuestros profesores de impartir el contenido de nuestros cursos. En RTEA contamos con una metodología dinámica, divertida y eficaz, con la cual aprenderás de una manera fácil y rápida una gran cantidad de nuevos conocimientos.



Clases bien explicadas
Contarás de grandes explicaciones en detalle, para que no te pierdas ninguna novedad.

Objetivos realistas

Aprendizaje dinámico

Acumular experiencia

¡Conoce las opiniones de nuestros alumnos!

Conoce de primera mano algunas de las reseñas de nuestros antiguos alumnos.

Simeoni
"El curso te vas para arriba, una verdadera maravilla tanto de los teóricos como prácticos"
Alumno Curso Avanzado

Astrako
"Es un curso muy completo y muy complejo, pero los profesores están ahí constantemente para hacer la curva de aprendizaje más llevadera"
Alumno Curso Avanzado

Karlunix
"El contenido del curso es grande e interesante, obtienes lo complicado y la atención de los profesores es excelente"
Alumno Curso Avanzado

No postergues más tu educación, ¡tu futuro te está esperando!

Un nuevo empleo, mejor salario, mayor calidad de vida, ¡todo lo puedes conseguir con educación! Es el momento de lograr lo que siempre has querido.

[Comienza a estudiar](#)

*Accede a nuestros cursos

¡Conoce nuestros métodos de pago!

En nuestra escuela pensamos en nuestros alumnos y queremos poder brindarles todas las oportunidades posibles para poder acceder a nuestros cursos. Que nada frene tus ganas de seguir aprendiendo, [si necesitas más ayuda contacta con nosotros](#).



PayPal **stripe**

Europa · Latinoamérica

Acerca de la empresa	Nuestros productos	Conecta con RTEA	Áreas de estudio
¿Qué es RTEA?	Nuestros cursos	Política de privacidad	Unreal
Nuestro equipo	Nuestros masters	Términos y condiciones	3D Generalist
Nuestros profesores	Servicio de afiliados	Acceso a las clases	Modelado y texturizado
Nuestros proyectos RTEA		Portal del estudiante	VFX Animation

© VertexBrothers, 2017 – 2024

Símbolos para redes sociales: Instagram, LinkedIn, Twitter, Facebook, YouTube

mysticMaterials: (Una de las páginas de nuestros cursos)

The screenshot displays two course pages from the RTEA School website:

- SHADERS UNREAL 4&5 PACKAGE**:
 - Thumbnail: A landscape scene with a fire effect.
 - Duration: 61h 20'
 - Shaders: 30 shaders
 - Price: 259,99€
 - Action: Añadir al carrito (Add to cart)
- UNREAL SHADERS**:
 - Thumbnail: A landscape scene with a fire effect.
 - Duration: 61h 20'
 - Shaders: 30 shaders
 - Price: 259,99€
 - Action: Añadir al carrito (Add to cart)

FIRE PACKAGE
1º PACKAGE

WATER PACKAGE
2º PACKAGE

AIR PACKAGE
3º PACKAGE

EARTH PACKAGE
4º PACKAGE

MAGIC PACKAGE
5º PACKAGE

FIRE SHADERS PACKAGE

1º PACKAGE

Duración 10h 20'
Software

Fire Shaders Package es un paquete con una duración aproximada de 10h y 20 min, configurado para aprender a desarrollar desde cero los siguientes shaders: burning, flipbook fire, war vision, teleporter, holograms y gameboy.

PULSA AQUÍ Y CONOCE TODOS LOS CONTENIDOS DE ESTE PAQUETE

WATER SHADERS PACKAGE

2º PACKAGE



Water Shaders Package es un paquete con una duración aproximada de 14h y 30 min, configurado para aprender a desarrollar desde cero los siguientes shaders: artic, liquid within glass, deep water, frozen elements I y II, cell shading y refractive dispersion.

PULSA AQUÍ Y CONOCE TODOS LOS CONTENIDOS DE ESTE PAQUETE

Duración 14h 30' 

Software 

AIR SHADERS PACKAGE

3º PACKAGE



Air Shaders Package es un paquete con una duración aproximada de 14h, configurado para aprender a desarrollar desde cero los siguientes shaders: sky custom, magic volumetric clouds, fake volumetric clouds, fog material, iridescent y matcap.

PULSA AQUÍ Y CONOCE TODOS LOS CONTENIDOS DE ESTE PAQUETE

Duración 14h 

Software 

EARTH SHADERS PACKAGE

4º PACKAGE

MAGIC SHADERS PACKAGE

5º PACKAGE

PULSA AQUÍ Y CONOCE TODOS LOS CONTENIDOS DE ESTE PAQUETE

Duración 10h 45'
Software

PULSA AQUÍ Y CONOCE TODOS LOS CONTENIDOS DE ESTE PAQUETE

Duración 11h 45'
Software

7. IMPLEMENTACIÓN

7.1. Servidor

- Desarrollo de código: Implementación en PHP, comenzando con los modelos, luego controladores, configuraciones de administrador y vistas.
- Estructura de ficheros: Organización del proyecto en carpetas como models, views, controllers, templates, assets, etc.

Nombre	Fecha de modificación	Tipo	Tamaño
backend	10/06/2024 18:56	Carpeta de archivos	
fonts	20/04/2024 15:00	Carpeta de archivos	
media	05/06/2024 12:06	Carpeta de archivos	
style	20/04/2024 15:00	Carpeta de archivos	
afiliados.html	10/06/2024 19:16	Opera GX Web Do...	38 KB
airMystic.html	10/06/2024 19:17	Opera GX Web Do...	59 KB
albertoGarcia.html	10/06/2024 19:18	Opera GX Web Do...	18 KB
bd.sql	07/06/2024 12:51	Archivo de origen ...	9 KB
codesCountryPhone.json	31/05/2024 6:28	Archivo de origen ...	31 KB
countries.json	01/06/2024 16:46	Archivo de origen ...	18 KB
cursoAvanzado.html	10/06/2024 19:18	Opera GX Web Do...	95 KB
cursos.html	10/06/2024 19:18	Opera GX Web Do...	114 KB
earthMystic.html	10/06/2024 19:17	Opera GX Web Do...	59 KB
fireMystic.html	10/06/2024 19:17	Opera GX Web Do...	59 KB
index.html	10/06/2024 19:16	Opera GX Web Do...	139 KB
magicMystic.html	10/06/2024 19:17	Opera GX Web Do...	59 KB
mysticMaterials.html	10/06/2024 19:15	Opera GX Web Do...	58 KB
nuestroEquipo.html	10/06/2024 19:17	Opera GX Web Do...	25 KB
obras.html	10/06/2024 19:17	Opera GX Web Do...	18 KB
quienesSomos.html	10/06/2024 19:17	Opera GX Web Do...	36 KB
script.js	22/05/2024 16:17	Archivo JavaScript	10 KB
scriptAfiliados.js	01/06/2024 18:22	Archivo JavaScript	2 KB
scriptIndex.js	13/05/2024 19:27	Archivo JavaScript	5 KB
waterMystic.html	10/06/2024 19:16	Opera GX Web Do...	59 KB

**PROYECTO DE APLICACIÓN WEB RTEA
SCHOOL**

DAW

Nombre	Fecha de modificación	Tipo	Tamaño
assets	10/06/2024 18:56	Carpeta de archivos	
config	10/06/2024 18:56	Carpeta de archivos	
controllers	10/06/2024 18:56	Carpeta de archivos	
models	10/06/2024 18:56	Carpeta de archivos	
router	10/06/2024 18:56	Carpeta de archivos	
views	10/06/2024 18:56	Carpeta de archivos	
backend.php	10/06/2024 17:13	Archivo de origen ...	1 KB
backendStudent.php	10/06/2024 18:17	Archivo de origen ...	1 KB
backendTeacher.php	10/06/2024 17:41	Archivo de origen ...	2 KB
bd.sql	10/06/2024 19:02	Archivo de origen ...	11 KB
login.php	10/06/2024 17:08	Archivo de origen ...	5 KB
logout.php	12/01/2024 8:00	Archivo de origen ...	1 KB

Secundaria (E:) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > assets > php			
Nombre	Fecha de modificación	Tipo	Tamaño
funciones.php	09/06/2024 17:26	Archivo de origen ...	4 KB

Este equipo > Secundaria (E:) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > config			
Nombre	Fecha de modificación	Tipo	Tamaño
db.php	10/06/2024 19:04	Archivo de origen ...	1 KB
sesionControl.php	09/06/2024 16:27	Archivo de origen ...	1 KB

**PROYECTO DE APLICACIÓN WEB RTEA
SCHOOL**

DAW

Este equipo > Secundaria (E) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > controllers				
Nombre	Fecha de modificación	Tipo	Tamaño	
administratorsController.php	10/06/2024 9:12	Archivo de origen ...	4 KB	
coursesController.php	10/06/2024 18:32	Archivo de origen ...	3 KB	
lessonsController.php	10/06/2024 10:42	Archivo de origen ...	3 KB	
rolesController.php	10/06/2024 11:07	Archivo de origen ...	2 KB	
students_coursesController.php	10/06/2024 12:37	Archivo de origen ...	3 KB	
students_teachersController.php	10/06/2024 12:46	Archivo de origen ...	3 KB	
students_workgroupsController.php	10/06/2024 13:18	Archivo de origen ...	3 KB	
studentsController.php	10/06/2024 10:04	Archivo de origen ...	4 KB	
teachersController.php	10/06/2024 9:48	Archivo de origen ...	4 KB	
usersController.php	10/06/2024 10:09	Archivo de origen ...	2 KB	
workgroupsController.php	10/06/2024 11:07	Archivo de origen ...	3 KB	

Este equipo > Secundaria (E) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > router				
Nombre	Fecha de modificación	Tipo	Tamaño	
Este equipo > Secundaria (E) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > models				
Nombre	Fecha de modificación	Tipo	Tamaño	
administratorModel.php	10/06/2024 16:58	Archivo de origen ...	7 KB	
courseModel.php	10/06/2024 18:31	Archivo de origen ...	6 KB	
Este equipo > Secundaria (E) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > views				
Nombre	Fecha de modificación	Tipo	Tamaño	
administrator	10/06/2024 18:56	Carpeta de archivos		
course	10/06/2024 18:56	Carpeta de archivos		
layout	10/06/2024 18:56	Carpeta de archivos		
lesson	10/06/2024 18:56	Carpeta de archivos		
rol	10/06/2024 18:56	Carpeta de archivos		
student	10/06/2024 18:56	Carpeta de archivos		
student_course	10/06/2024 18:56	Carpeta de archivos		
student_teacher	10/06/2024 18:56	Carpeta de archivos		
student_workgroup	10/06/2024 18:56	Carpeta de archivos		
teacher	10/06/2024 18:56	Carpeta de archivos		
user	10/06/2024 18:56	Carpeta de archivos		
workgroup	10/06/2024 18:56	Carpeta de archivos		
404.php	12/01/2024 8:00	Archivo de origen ...	1 KB	
inicio.php	09/06/2024 18:32	Archivo de origen ...	3 KB	

Secundaria (E:) > COPIA DE SEGURIDAD > WEB RTEA-sass > backend > views > administrator				
Nombre	Fecha de modificación	Tipo	Tamaño	
create.php	10/06/2024 9:57	Archivo de origen ...	4 KB	
delete.php	10/06/2024 12:31	Archivo de origen ...	1 KB	
edit.php	10/06/2024 9:07	Archivo de origen ...	5 KB	
list.php	10/06/2024 8:41	Archivo de origen ...	4 KB	
search.php	10/06/2024 9:33	Archivo de origen ...	8 KB	
show.php	10/06/2024 4:15	Archivo de origen ...	2 KB	
store.php	10/06/2024 2:24	Archivo de origen ...	1 KB	

7.2. Cliente

- Framework cliente: Utilización de JavaScript y jQuery.

- Elementos desarrollados: Componentes de interfaz de usuario y scripts para la interacción con el backend.

8. DESPLIEGUE

8.1. Modelo de despliegue utilizado

- Instalación directa en el servidor: Utilización de un hosting con soporte PHP y MySQL. El hosting es Tomahost. El despliegue se basó en crear un subdominio, ya que la empresa tenía ya contratado el hosting, subir los archivos del frontend y luego importar mi base de datos dentro del servidor anclandola al subdominio de: testing.vertexbrothers.club y ya tenía todo en funcionamiento. Lo único a destacar, es que para que el sitio web tuviera la protección de seguridad https y lo encontrara el navegador como un sitio web seguro, tuve que dentro del panel del subdominio en

el hosting, en el apartado de seguridad, generar un certificado gratuito SSL que se instala y protege la página al momento.

8.2. Datos iniciales y configuración

Configuración de conexión con mi base de datos: qwawrodw_vertexBrothers_db

```
backend > config > db.php > db
1  <?php
2  class db
3  [
4      const HOST = "localhost:3306";
5      const DBNAME = "qwawrodw_vertexBrothers_db";
6      const USER = "qwawrodw_clajae";
7      const PASSWORD = "***33486188mM**"; // Evidentemente adapta los valores
8
9      static public function conexion()
10     {
11         $conexion = null;
12         try {
13             $opciones = array(PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION);
14             $conexion = new PDO('mysql:host=localhost; dbname=' .self::DBNAME,self::USER,self::PASSWORD, $opciones);
15         } catch (Exception $e) {
16             echo "Ocurrió algo con la base de datos: " . $e->getMessage();
17         }
18         return $conexion; //Es un objeto de conexión PDO
19     }
20 ]
```

8.3. Pasos para el despliegue

Son los mismos pasos que he mencionado en el apartado anterior.

8.4. Proveedores y servicios utilizados

Utilización de un hosting con soporte PHP y MySQL, en este caso Tomahost, que era el que la empresa ya tenía contratado.

9. HERRAMIENTAS UTILIZADAS

- Frontend:

- HTML5: Estructuración del contenido.
- CSS3 y SASS: Estilización y optimización del CSS.
- JavaScript: Interactividad y dinámicas del cliente.

- Backend:

- PHP: Lógica del servidor y manejo de datos.
- Modelo Vista Controlador (MVC): Organización de la aplicación.

**ANEXO CON GUÍA DE UTILIZACIÓN DE LA APLICACIÓN
PROYECTO DE APLICACIÓN WEB RTEA SCHOOL
(RTEA)**

**CICLO FORMATIVO DE GRADO SUPERIOR
DESARROLLO DE APLICACIONES WEB**

AUTOR: CLAUDIA GARCÍA JAÉN

TUTORES:
JESÚS VIVES CÉSPEDES
JOSE ANTONIO NAVARRETE SANCHEZ

COORDINADOR:
JULIO MARTÍNEZ LUCAS

1. GUÍA DE UTILIZACIÓN:

El acceso al backoffice de esta escuela, se hace mediante el enlace al inicio de sesión que hay arriba a la derecha dentro del botón azul, en todas las cabeceras de las páginas.

Esta aplicación consta de tres roles diferentes mediante los cuales los diferentes integrantes de la escuela podrán hacer uso de su propio backend. A continuación, te explicaré en qué consiste cada uno, cómo se consigue, de qué manera se registran cada uno de ellos, cómo pueden acceder al backoffice y todo lo que pueden hacer dentro de este:

Roles:

1- Administrador: El rol de administrador lo obtendrán todos aquellos que formen parte del equipo técnico de la empresa o bien que desempeñen alguna labor de administración dentro de esta. Este rol tendrá permisos para ver, listar y buscar cualquier tabla que exista dentro de la base de datos. Pero solo tendrá permisos para crear nuevos administradores, profesores, alumnos, cursos y grupos de trabajo. Más adelante, se implantará un rol único y exclusivo para los CEO's de la empresa, en el que si que tendrán acceso al listado, a la búsqueda y al añadir de cualquier tabla existente dentro del backoffice, ya que este será el rol de máxima autoridad dentro de la bd. Mientras tanto, el rol de administrador es el que más privilegios tiene a día de hoy. Este se consigue, siendo directamente asignado por el (futuro CEO) o bien añadiéndolo a la base de datos de la empresa.

2 – Profesor: El rol de profesor lo obtendrán todos aquellos que formen parte del equipo de docentes de la empresa, una vez se formalicé que formas parte de la empresa, en el backoffice se te asignará el rol de profesor. Este rol tiene

permisos para listar y buscar todos sus alumnos (estos serán aquellos que han comprado alguno/s de sus cursos), para añadir, listar y buscar todos sus cursos disponibles, las lecciones de cada curso y sus grupos de trabajo. Si que es verdad, que el profesor, una vez se mete dentro del listado de sus cursos, le aparecen abajo de cada curso, el listado de las lecciones de cada uno de estos, y él podrá mediante un checker que tiene al final de cada lección, puede hacer que el enlace al video de esa lección, lo deje habilitado o no a los alumnos que le han comprado ese curso.

3 – Alumno: El rol de alumno lo obtendrán todas aquellas personas que hayan comprado alguna vez cualquiera de nuestros cursos a través de nuestra plataforma. Estos, después de comprar el curso, recibirán un formulario de registro con su usuario y contraseña con el que podrán acceder al backoffice. El usuario será su email, el cuál habrán usado en su compra y la contraseña será una generada automáticamente que podrán cambiar luego al entrar. (Todo esto, se tiene que terminar de implementar en un futuro, ya que las conexiones con la pasarela de pago todavía no están activas). Este rol de alumno en el backoffice podrá acceder al listar y buscar de: todos sus cursos (los cursos que ha adquirido), todos sus profesores (los profesores de los cursos que ha comprado). (En este listado de profesores aparecerán sus datos personales de contacto (que serán el nombre y el email) por si el alumno necesita trasladarles algo en cualquier momento). Dentro de sus cursos les aparecerán como a los profesores, un listado con todas las lecciones pertenecientes a ese curso y al lado si el profesor lo ha habilitado así desde su backoffice, les aparecerá la url para que ellos puedan visitar el video de la lección. Y por último, también podrán consultar el listado de grupos de trabajo a los que pertenecen.

Al respectivo backoffice de estos diferentes roles, se accederá de la misma manera, mediante el correo electrónico y su respectiva contraseña, y siempre desde el inicio de sesión a través de la página web.

**ANEXO CON EXPLICACIÓN DE LOS PROCESOS PRINCIPALES
PROYECTO DE APLICACIÓN WEB RTEA SCHOOL
(RTEA)**

**CICLO FORMATIVO DE GRADO SUPERIOR
DESARROLLO DE APLICACIONES WEB**

AUTOR: CLAUDIA GARCÍA JAÉN

TUTORES:
JESÚS VIVES CÉSPEDES
JOSE ANTONIO NAVARRETE SANCHEZ

COORDINADOR:
JULIO MARTÍNEZ LUCAS

1. PROCESOS PRINCIPALES DEL PROGRAMA:

Los procesos que se realizan principalmente en este programa son varios. Algunos de los que más cabe destacar son:

- La creación de los diferentes roles: administradores, profesores y estudiantes (que serán los datos de los integrantes de nuestra base de datos):

/models/administratorModel.php:

```
public function insert (array $administrator): ?int //devuelve entero o null
{
    $sql = "INSERT INTO administrators(name, email, password, phone, rol_id) VALUES (:name, :email, :password, :phone, :rol_id);";

    try {
        $sentencia = $this->conexion->prepare($sql);
        $arrayDatos = [
            ":name"=>$administrator["name"],
            ":email"=>$administrator["email"],
            ":password"=>$administrator["password"],
            ":phone"=>$administrator["phone"],
            ":rol_id"=>$administrator["rol_id"],
        ];
        $resultado = $sentencia->execute($arrayDatos);

        /*Pasar en el mismo orden de los ? execute devuelve un booleano.
        True en caso de que todo vaya bien, falso en caso contrario.*/
        //Así podriamos evaluar
        // return ($resultado == true) ? $this->conexion->lastInsertId() : null;

        if ($resultado == true) {

            $administrator = [
                "name" => $administrator["name"],
                "email" => $administrator["email"],
                "password_key" => $administrator["password"],
                "rol_id" => 3,
            ];
            $administratorId = $this->administratorModel->insert($administrator);

            if ($administratorId !== null) {
                return $this->conexion->lastInsertId();

            } else {
                return null;
            }
        } else {
            // Revertir la transacción si hubo un error en la inserción en profesores
            $this->conexion->rollBack();
            return null;
        }
    } catch (Exception $e) {
        echo 'Excepción capturada: ', $e->getMessage(), "<b>";
    }
}
```

/controllers/administratorsController.php:

```

public function crear(array $arrayAdministrators): void
{
    $error = false;
    $errores = [];

    //vaciamos los posibles errores
    $_SESSION["errores"] = [];
    $_SESSION["datos"] = [];

    // ERRORES DE TIPO
    if (!is_valid_email($arrayAdministrators["email"])) {
        $error = true;
        $errores["email"][] = "El email tiene un formato incorrecto";
    }

    if(!is_valid_password($arrayAdministrators["password"])){
        $error = true;
        $errores["password"][] = "La contraseña tiene un formato incorrecto";
    }

    if(!is_valid_phone($arrayAdministrators["phone"])){
        $error = true;
        $errores["phone"][] = "El teléfono tiene un formato incorrecto";
    }

    //campos NO VACIOS
    $arrayNoNulos = ["name", "email", "password", "phone", "rol_id"];
    $nulos = HayNulos($arrayNoNulos, $arrayAdministrators);

    if (count($nulos) > 0) {
        $error = true;

        for ($i = 0; $i < count($nulos); $i++) {
            $errores[$nulos[$i]][] = "El campo {$nulos[$i]} es nulo";
        }
    }
}

//CAMPOS UNICOS
$arrayUnicos = ["email"];
foreach ($arrayUnicos as $CampoUnico) {
    if ($this->model->exists($CampoUnico, $arrayAdministrators[$CampoUnico])) {
        $errores[$CampoUnico][] = "El {$arrayAdministrators[$CampoUnico]} de
        {$CampoUnico} ya existe";
        $error = true;
    }
}

$id = null;
if (!$error) $id = $this->model->insert($arrayAdministrators);

if ($id == null) {
    $_SESSION["errores"] = $errores;
    $_SESSION["datos"] = $arrayAdministrators;
    header("location:backend.php?accion=crear&tabla=administrator&error=true&id={$id}");
    exit ();
}

```

```
$id = null;
if (!$error) $id = $this->model->insert($arrayAdministrators);

if ($id == null) {
    $_SESSION["errores"] = $errores;
    $_SESSION["datos"] = $arrayAdministrators;
    header("location:backend.php?accion=crear&tabla=administrator&error=true&id={$id}");
    exit ();
} else {
    // unset borra los datos de dentro de la variable
    unset($_SESSION["errores"]);
    unset($_SESSION["datos"]);
    header("location:backend.php?accion=ver&tabla=administrator&id=" . $id);
    exit ();
}
}
```

Y para el resto de roles, sería exactamente igual, pero con sus propias variables.

- La forma en la que se muestran los cursos junto con sus lecciones en nuestra bd:

/models/courseModel.php:

```
public function readAll():array
{
    $sentencia = $this->conexion->prepare("SELECT * FROM courses");
    // $arrayDatos = [];
    $resultado = $sentencia->execute();
    if ($resultado == false) return [];
    // usamos método query
    // FETCH_ASSOC te devuelve en un Array Asociativo los datos, se accederá: $c[0]['id'] / $course["id"]
    // FETCH_OBJ nos devolverá un Array de Objetos, y se accederá: $course->id;
    $courses = $sentencia->fetchAll(PDO::FETCH_OBJ);
    return $courses;
}

public function readAllCourse():array
{
    $sentencia = $this->conexion->prepare("SELECT * FROM courses");
    // $arrayDatos = [];
    $resultado = $sentencia->execute();
    if ($resultado == false) return [];
    // usamos método query
    // FETCH_ASSOC te devuelve en un Array Asociativo los datos, se accederá: $c[0]['id'] / $course["id"]
    // FETCH_OBJ nos devolverá un Array de Objetos, y se accederá: $course->id;
    $courses = $sentencia->fetchAll(PDO::FETCH_OBJ);
    return $courses;
}
```

/controllers/coursesController.php:

```
public function ver(int $id): ?stdClass
{
    return $this->model->read($id);
}

public function listar ():array
{
    return $this->model->readAll();
}
```

views/course/list.php:

```
<?php
require_once "controllers/coursesController.php";
require_once "controllers/lessonsController.php";
require_once "controllers/teachersController.php";

$controlador = new CoursesController();
$controladorLessons = new LessonsController();
$controladorTeachers = new TeachersController();

$courses = $controlador->listar();
$lessons = $controladorLessons->listar();
$teachers = $controladorTeachers->listar();

$visibilidad = "hidden";

try {
    if (isset($_REQUEST["evento"]) && $_REQUEST["evento"] == "borrar") {
        $visibilidad = "visibility";
        $clase = "alert alert-success";

        $mensaje = "El curso con id: {$_REQUEST['id']} ;Borrado correctamente!";

        if (isset($_REQUEST["error"])) {
            $clase = "alert alert-danger ";
            throw new Exception("ERROR!!! No se ha podido borrar el curso con id: {$_REQUEST['id']}");
        }
    }
} catch (Exception $deleteError) {
    $mensaje = $deleteError->getMessage();
}

?>
```

```

<main class="col-md-9 ms-sm-auto col-lg-10 px-md-4">
    <div class="d-flex justify-content-between flex-wrap flex-md-nowrap align-items-center pt-3 pb-2 mb-3 border-bottom">
        <h3>Listado de cursos</h3>
    </div>
    <div id="contenido">
        <div class="<?= $clase ?> <?= $visibilidad ?> role="alert">
            <?= $mensaje ?>
        </div>
        <table class="table table-light table-hover">
            <?php
            if (count($courses) <= 0 && count($lessons) <= 0) :
                echo "No hay Datos a Mostrar";
            else : ?>
            <table class="table table-light table-hover" style="text-align: center; border: 1px solid #2f3235;">
                <thead class="table-dark">
                    <tr>
                        <th scope="col">ID</th>
                        <th scope="col">Nombre</th>
                        <th scope="col">Duración</th>
                        <th scope="col">Nº lecciones</th>
                        <th scope="col">Profesor</th>
                        <th scope="col">Eliminar</th>
                        <th scope="col">Editar</th>
                    </tr>
                </thead>
                <tbody>
                    <?php foreach ($courses as $course) :
                        $id = $course->id;
                    ?>
                    <tr>
                        <th scope="row"><?= $course->id ?></th>
                        <th scope="row"><?= $course->name ?></th>
                        <th scope="row"><?= $course->duration ?></th>
                        <th scope="row"><?= $course->num_lessons ?></th>
                        <?php
                            foreach ($teachers as $teacher) :
                                if ($course->teacher_id == $teacher->id) {
                                    ?>
                                    <th scope="row"><?= $teacher->name ?></th>
                                    <?php
                                }
                            endforeach;
                        ?>

                        <td><a class="btn btn-danger" href="backend.php?tabla=course&accion=borrar&id=<?= $id ?>"><i class="fa fa-trash"></i> Borrar</a></td>
                        <td><a class="btn btn-success" href="backend.php?tabla=course&accion=editar&id=<?= $id ?>"><i class="fas fa-pencil-alt"></i> Editar</a></td>
                    </tr>
                    <thead>
                        <tr>
                            <th scope="col" style="background-color: grey;"></th>
                            <th scope="col" style="background-color: grey;">Lección ID</th>
                            <th scope="col" style="background-color: grey;">Nombre de la lección</th>
                            <th scope="col" style="background-color: grey;">Duración</th>
                            <th scope="col" style="background-color: grey;">ID del curso</th>
                            <th scope="col" style="background-color: grey;">Eliminar</th>
                            <th scope="col" style="background-color: grey;">Editar</th>
                        </tr>
                    </thead>
                </tbody>
            </table>
        </div>
    </div>

```

```

<?php
    <table class="table table-light table-hover" style="text-align: center; border: 1px solid #2f3235;">
        <tr>
            <th scope="col" style="background-color: grey;"></th>
            <th scope="col" style="background-color: grey;">Lección ID</th>
            <th scope="col" style="background-color: grey;">Nombre de la lección</th>
            <th scope="col" style="background-color: grey;">Duración</th>
            <th scope="col" style="background-color: grey;">ID del curso</th>
            <th scope="col" style="background-color: grey;">Eliminar</th>
            <th scope="col" style="background-color: grey;">Editar</th>
        </tr>
    </thead>
    <?php foreach ($lessons as $lesson) :>
        $lesson_id = $lesson->id;

        <?php if ($course->id == $lesson->course_id) {>
            <?php ?>
            <tr>
                <th style="background-color: silver;"></th>
                <th scope="row" style="background-color: silver;"><?= $lesson->id ?></th>
                <th scope="row" style="background-color: silver;"><?= $lesson->name ?></th>
                <th scope="row" style="background-color: silver;"><?= $lesson->duration ?></th>
                <th scope="row" style="background-color: silver;"><?= $lesson->course_id ?></th>
                <td style="background-color: silver;"><a class="btn btn-danger" href="backend.php?tabla=lesson&accion=borrar&id=<?= $id ?>"><i class="fa fa-trash"></i> Borrar</a></td>
                <td style="background-color: silver;"><a class="btn btn-success" href="backend.php?tabla=lesson&accion=editar&id=<?= $id ?>"><i class="fas fa-pencil-alt"></i> Editar</a></td>
            </tr>
        <?php ?>
        <?php endforeach; ?>
        <?php ?>
        </tr>
    <?php endforeach; ?>
    <?php ?>
</tbody>
</table>
<?php endif; ?>
</div>
</main>

```

- La forma en la buscamos por campos (ejemplo:usuarios):

/models/userModel.php:

```
public function search(string $campo = "id", string $modoBusqueda = "contiene", string $datoInput = ""):array
{
    $sentencia = $this->conexion->prepare("SELECT * FROM users WHERE $campo LIKE :dato");
    //ojo el si ponemos % siempre en comillas dobles "
    if ($modoBusqueda == "empiezaPor") {
        $arrayDatos=[":dato"=>"$datoInput%"];
    } elseif ($modoBusqueda == "acabaEn") {
        $arrayDatos=[":dato"=>%$datoInput];
    } elseif ($modoBusqueda == "contiene") {
        $arrayDatos=[":dato"=>%$datoInput%];
    } elseif ($modoBusqueda == "iguala") {
        $arrayDatos=[":dato"=>"$datoInput"];
    }

    $resultado = $sentencia->execute($arrayDatos);
    if (!$resultado) return [];
    $users = $sentencia->fetchAll(PDO::FETCH_OBJ);
    return $users;
}
```

/controllers/usersController.php:

```
public function buscar(string $campo = "id", string $metodo = "contiene", string $texto = ""): array
{
    $users = $this->model->search($campo, $metodo, $texto);
    return $users;
}
```

- La forma en la que cualquier usuario puede logearse:

/models/userModel.php:

```
public function login(string $email, string $password_key): ?stdClass
{
    $sentencia = $this->conexion->prepare("SELECT * FROM users WHERE
    email=:email and password_key=:password_key");
    $arrayDatos = [
        ":email" => $email,
        ":password_key"=>$password_key
    ];
    $resultado = $sentencia->execute($arrayDatos);
    if (!$resultado) return null;
    $user = $sentencia->fetch(PDO::FETCH_OBJ);
    //fetch devuelve el objeto standard o false si no hay persona
    return ($user == false) ? null : $user;
}
```

/config/sesionControl.php:

```
sesionControl.php ✘
config > sesionControl.php
1  <?php
2  session_start();
3  if (!isset ($_SESSION["email"])){
4      $_SESSION=[];
5      session_destroy();
6      header ("location:login.php");
7      exit ();
8  }
```

login.php:

```
<?php
session_start();
$email = ($_POST["email"]) ?? "";
$password_key = ($_POST["password_key"]) ?? "";
$rol_id = ($_POST["rol_id"]) ?? "";
$error = false;

if (isset($_POST["email"], $_POST["password_key"], $_POST["rol_id"])) {

    require_once "models/userModel.php";
    require_once "controllers/usersController.php";
    $userModel = new userModel();
    $usersController = new usersController();
    $users = $usersController->listar();

    if ($email != null) {
        $_SESSION["email"] = $email;

        // Verifica si el usuario está en la lista de usuarios
        $userFound = false;
        foreach ($users as $user) {
            if ($user->email == $email && $user->password_key == $password_key) {
                $userFound = true;
                if ($user->rol_id == 1) {
                    header("Location: backendStudent.php");
                    exit;
                } elseif ($user->rol_id == 2) {
                    header("Location: backendTeacher.php");
                    exit;
                } elseif ($user->rol_id == 3) {
                    header("Location: backend.php");
                    exit;
                }
            }
        }

        // Si el usuario no está en la lista o las credenciales son incorrectas
        if (!$userFound) {
            $error = true;
        }
    }
}
```

ANEXO ESTRUCTURA DE DATOS

**PROYECTO DE APLICACIÓN WEB RTEA SCHOOL
(RTEA)**

**CICLO FORMATIVO DE GRADO SUPERIOR
DESARROLLO DE APLICACIONES WEB**

AUTOR: CLAUDIA GARCÍA JAÉN

TUTORES:

JESÚS VIVES CÉSPEDES

JOSE ANTONIO NAVARRETE SANCHEZ

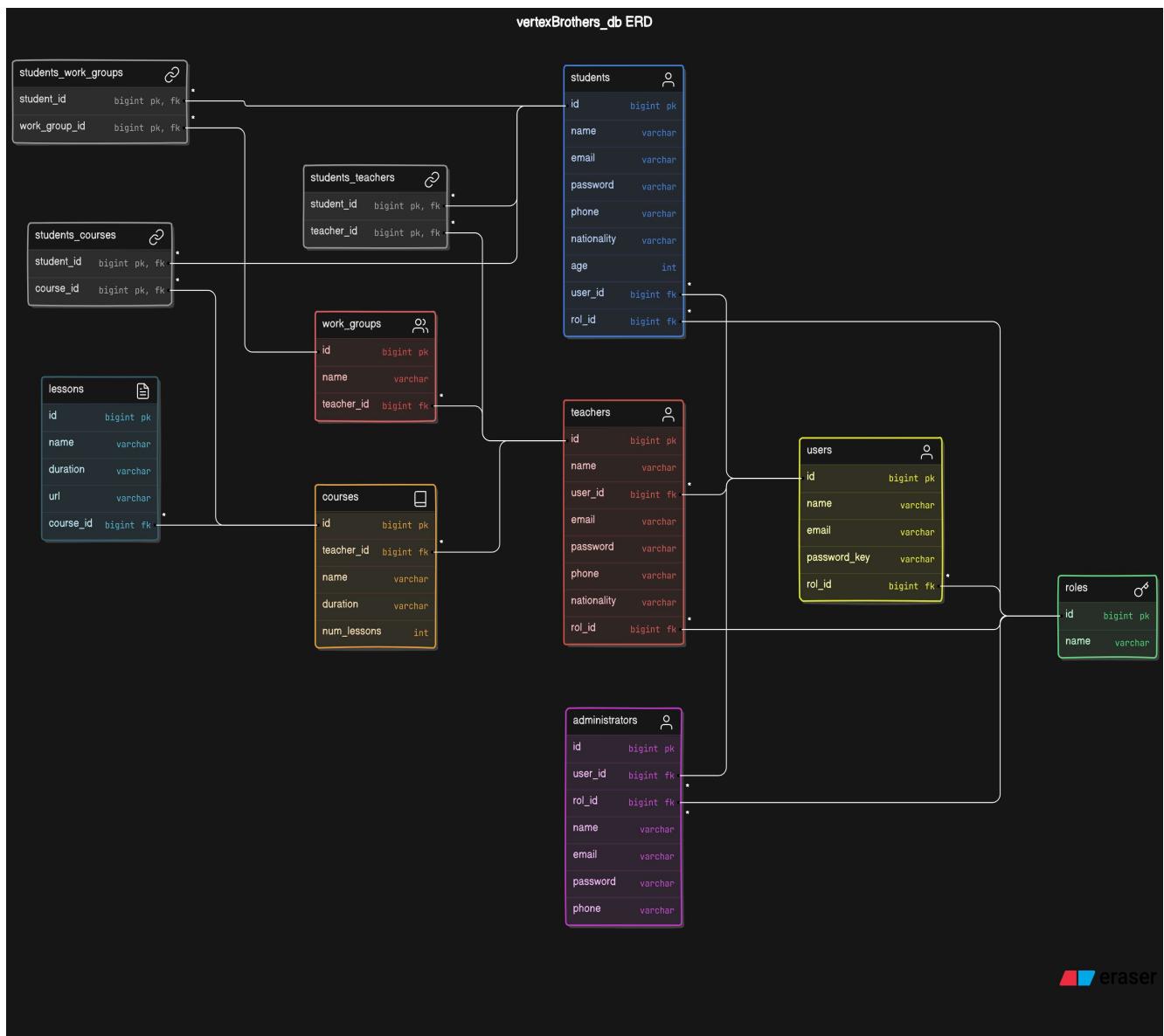
COORDINADOR:

JULIO MARTÍNEZ LUCAS

1. ESTRUCTURA DE DATOS:

Los datos de nuestra BD de vertexBrothers son los siguientes:

- Modelo entidad relación:



- Datos de nuestra BD y sus tipos:

```
CREATE TABLE `users` (
  `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(255) NOT NULL,
  `email` VARCHAR(255) NOT NULL,
  `password` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `users_UN` (`email`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `roles` (
  `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `students` (
  `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(255) NOT NULL,
  `email` VARCHAR(255) NOT NULL,
  `phone` VARCHAR(255) NOT NULL,
  `nationality` VARCHAR(255) NOT NULL,
  `age` INT NOT NULL,
  `user_id` BIGINT(20) UNSIGNED NOT NULL,
  `rol_id` BIGINT(20) UNSIGNED NOT NULL,
  PRIMARY KEY (`id`),
  KEY `students_user_id_foreign` (`user_id`),
```

```
KEY `students_rol_id_foreign` (`rol_id`),
CONSTRAINT `students_FK` FOREIGN KEY (`user_id`) REFERENCES `users`(`id`) ON UPDATE CASCADE ON DELETE CASCADE,
CONSTRAINT `students_FK_1` FOREIGN KEY (`rol_id`) REFERENCES `roles`(`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `teachers` (
`id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
`name` VARCHAR(255) NOT NULL,
`email` VARCHAR(255) NOT NULL,
`phone` VARCHAR(255) NOT NULL,
`nationality` VARCHAR(255) NOT NULL,
`user_id` BIGINT(20) UNSIGNED NOT NULL,
`rol_id` BIGINT(20) UNSIGNED NOT NULL,
PRIMARY KEY (`id`),
KEY `teachers_user_id_foreign` (`user_id`),
KEY `teachers_rol_id_foreign` (`rol_id`),
CONSTRAINT `teachers_FK` FOREIGN KEY (`user_id`) REFERENCES `users`(`id`) ON UPDATE CASCADE ON DELETE CASCADE,
CONSTRAINT `teachers_FK_1` FOREIGN KEY (`rol_id`) REFERENCES `roles`(`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `administrators` (
  `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(255) NOT NULL,
  `email` VARCHAR(255) NOT NULL,
  `phone` VARCHAR(255) NOT NULL,
  `user_id` BIGINT(20) UNSIGNED NOT NULL,
  `rol_id` BIGINT(20) UNSIGNED NOT NULL,
  PRIMARY KEY (`id`),
  KEY `administrators_user_id_foreign` (`user_id`),
  KEY `administrators_rol_id_foreign` (`rol_id`),
  CONSTRAINT `administrators_FK` FOREIGN KEY (`user_id`) REFERENCES `users` (`id`) ON UPDATE CASCADE ON DELETE CASCADE,
  CONSTRAINT `administrators_FK_1` FOREIGN KEY (`rol_id`) REFERENCES `roles` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `courses` (
  `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(255) NOT NULL,
  `duration` VARCHAR(255) NOT NULL,
  `num_lessons` INT NOT NULL,
  `teacher_id` BIGINT(20) UNSIGNED NOT NULL,
  PRIMARY KEY (`id`),
  KEY `courses_teacher_id_foreign` (`teacher_id`),
  CONSTRAINT `courses_FK` FOREIGN KEY (`teacher_id`) REFERENCES `teachers` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `lessons` (
    `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
    `name` VARCHAR(255) NOT NULL,
    `duration` VARCHAR(255) NOT NULL,
    `course_id` BIGINT(20) UNSIGNED NOT NULL,
    PRIMARY KEY (`id`),
    KEY `lessons_course_id_foreign` (`course_id`),
    CONSTRAINT `lessons_FK` FOREIGN KEY (`course_id`) REFERENCES
    `courses` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `work_groups` (
    `id` BIGINT(20) UNSIGNED NOT NULL AUTO_INCREMENT,
    `name` VARCHAR(255) NOT NULL,
    `teacher_id` BIGINT(20) UNSIGNED NOT NULL,
    PRIMARY KEY (`id`),
    KEY `work_groups_teacher_id_foreign` (`teacher_id`),
    CONSTRAINT `work_groups_FK` FOREIGN KEY (`teacher_id`) REFERENCES
    `teachers` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `students_teachers` (
    `student_id` BIGINT(20) UNSIGNED NOT NULL,
    `teacher_id` BIGINT(20) UNSIGNED NOT NULL,
    PRIMARY KEY (`student_id`, `teacher_id`),
    KEY `students_teachers_student_id_foreign` (`student_id`),
    KEY `students_teachers_teacher_id_foreign` (`teacher_id`),
    CONSTRAINT `students_teachers_FK` FOREIGN KEY (`student_id`)
REFERENCES `students` (`id`) ON UPDATE CASCADE ON DELETE CASCADE,
```

```
CONSTRAINT `students_teachers_FK_1` FOREIGN KEY (`teacher_id`)
REFERENCES `teachers` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `students_courses` (
  `student_id` BIGINT(20) UNSIGNED NOT NULL,
  `course_id` BIGINT(20) UNSIGNED NOT NULL,
  PRIMARY KEY (`student_id`, `course_id`),
  KEY `students_courses_student_id_foreign` (`student_id`),
  KEY `students_courses_course_id_foreign` (`course_id`),
  CONSTRAINT `students_courses_FK` FOREIGN KEY (`student_id`)
REFERENCES `students` (`id`) ON UPDATE CASCADE ON DELETE CASCADE,
  CONSTRAINT `students_courses_FK_1` FOREIGN KEY (`course_id`)
REFERENCES `courses` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```
CREATE TABLE `students_work_groups` (
  `student_id` BIGINT(20) UNSIGNED NOT NULL,
  `work_group_id` BIGINT(20) UNSIGNED NOT NULL,
  PRIMARY KEY (`student_id`, `work_group_id`),
  KEY `students_work_groups_student_id_foreign` (`student_id`),
  KEY `students_work_groups_work_group_id_id_foreign` (`work_group_id`),
  CONSTRAINT `students_work_groups_FK` FOREIGN KEY (`student_id`)
REFERENCES `students` (`id`) ON UPDATE CASCADE ON DELETE CASCADE,
  CONSTRAINT `students_work_groups_FK_1` FOREIGN KEY (`work_group_id`)
REFERENCES `work_groups` (`id`) ON UPDATE CASCADE ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```