PRACTICO O ENTREGA N1 DE PRACTICAS PROFESIONALIZANTE 4

**Etapas del análisis de requerimientos**

1. **Identificar a los usuarios y partes interesadas.**

Se identifica en este caso a los usuarios como profesores de educación nivel secundaria común y técnica. La parte interesada seria los profesores ya que encontrarían una manera de organizarse y los alumnos del sistema educativo ya que encontrarían un nivel de organización superior**.**

1. **Definir los objetivos del proyecto.**

Se espera diseñar un software que ayude a las docentes o profesores a cargar datos personales, datos de alumnos en cada institución, que se generen reportes con estados de calificaciones.

* + Los objetivos serian sistematizar el ingreso de datos.
  + Se espera formar bases de datos para cada variable.
  + Se desea poder, generar reportes que sean mandados a las distintas instituciones con información estado de calificaciones.

1. **Capturar los requisitos de los usuarios y partes interesadas**.
   * Disminuir tiempos en escritura a mano alzada de libretas.
   * Modernizar el sistema educativo público.
   * Absorber parte del trabajo de cada institución, logrando pasar informes o reportes finalizados.
   * El software debería poder garantizar el ingreso de datos a una base de datos en un proveedor de la nube o en una PC portátil sin acceso a internet (dependiendo del cliente).
2. **Categorizar los requisitos del proyecto**

Requerimientos funcionales

El software debe:

* + Poder registrar datos personales de alumnos, carga de notas, carga de docente, modificaciones de notas, asignar periodos de cursados (trimestres o cuatrimestres).
  + Poder relacionar cada alumno cargado con el curso o institución, su turno, el docente implicado, etc.
  + Generar reportes de estado de libreta.
  + Generar listados de alumnos, docente, cursos y materias.

Requerimientos no funcionales:

El software de comportarse de la siguiente manera:

* + Debe ser una herramienta eficiente.
  + Debe tener seguridad.
  + Debe ser fácil de utilizar.

1. **Analizar los requisitos**
   * Se decide construir un ingreso con distintos niveles de usuarios.
   * Se diseña un con bootstrap un diseño de Interfaz de Usuario (UI) apto para el profesor.
   * Luego de trabaja en el diseño del código, donde se debe establecer la funcionalidad del sistema (UX).
   * Seguido a esto se crean las bases de datos según las clase y definición de funciones.
2. **Modelar los requisitos**

El modelado UML para un sistema educativo de profesores con entrada de datos, listado y reportes varios podría incluir varios diagramas para describir diferentes aspectos del sistema.

**1. Diagrama de Casos de Uso**

Describe las interacciones entre los actores y el sistema. Actores principales:

* **alumno:** Solicita información sobre calificaciones.
* **docentes:** Ejecuta ingreso de datos. Solicita listado de docentes.
* **cursos:** El docente solicita listado cursos, en ingreso de cursos.
* **Instituciones:** El docente realiza ingreso de instituciones.
* **Calificaciones**: El docente realiza ingreso de calificaciones.
* **Libreta:** El docente solicita datos de calificaciones.
* **Materias:** El docente registra y consulta materias.
* **Usuarios:** El administrador el usuario con distintos permisos.

Casos de uso:

* **Registro de alumnos, cursos, calificaciones, docentes, materias.**
* **Validación de usuarios y claves.**
* **Validar entrada de datos como ingresos de calificaciones, ingresos por DNI,**
* **Generar listados de alumnos, docentes, materias, cursos y calificaciones y reportes.**
* **Generar un aula virtual para descargar PDF.**

**2. Diagrama de Actividad**

Representa el flujo de actividades:

1. El docente/alumno/administrador inician sesión en el sistema.
2. El sistema genera la verificación de usuario, sí verifica al usuario abre el portal de entrada y otorga permisos.
3. El usuario:

**usuario/administrador:**

* + Consultar listado materias, luego agregar /buscar materias. Luego volver.
  + Consultar listado alumnos buscar/agregar/eliminar/abrir calificaciones e alumnos.
  + Cargar calificaciones nuevas.
  + Listar cursos/buscar cursos.
  + Consultar docente/modificar docente.
  + Listar /editar y guardar calificaciones.
  + Cargar datos de alumnos.

**alumno:**

* + Consultar nota individual.
  + Ingresar al aula virtual.
  + Poder descargar información.

**administrador:**

* + Consultar listado materias, luego agregar /buscar materias. Luego volver.
  + Consultar listado alumnos buscar/agregar/eliminar/abrir calificaciones e alumnos.
  + Cargar calificaciones nuevas.
  + Listar cursos/buscar cursos/guardar cursos nuevos.
  + Consultar docente/modificar docente/agregar docente nuevo/eliminar.
  + Listar /editar y guardar calificaciones.
  + Cargar datos de por formulario de alumnos/docentes/cursos/materias.

1. Generar reportes
2. Salir de la sesión.

**3. Diagrama de Clases**

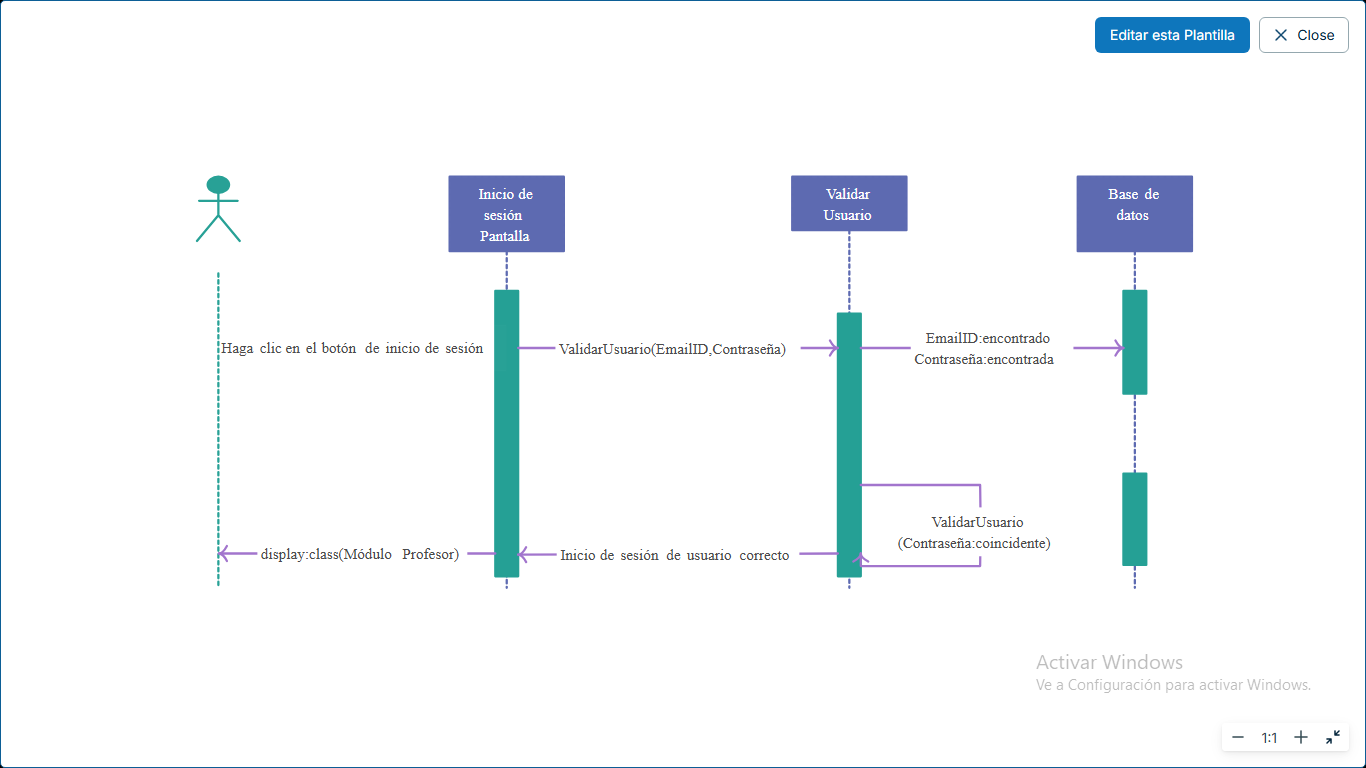
Modela las principales clases del sistema:

* **Alumnos**
  + Datos personales ingreso y modificación.
* **Calificaciones**
  + Ingreso y modificación de calificaciones.
  + Algoritmos de validación
* **Curso**
  + Ingreso y modificación de cursos.
  + Algoritmos de validación
* **Docente**
  + Ingreso y modificación de docentes.
  + Algoritmos de validación
* **Materia**
  + Ingreso y modificación de materias.
* **Planilla**
  + Vista de calificaciones con datos adicionales.
* **Reporte**
  + Convertidor de audio a texto
* **Aula virtual**
  + Ingreso de información en pdf para descargar.

**4. Diagrama de Secuencia**

Describe cómo interactúan los objetos, para eso podemos analizar un ejemplo de loggin:

1. **Usuario realiza inicio de sesión**.
2. **Se valida el usuario (verificando nombre de usuario y contraseña).**
3. **Se consulta a la base de datos de datos.**



1. Validar los requisitos con las partes interesadas

En la validación de datos se destacan:

* + Validación de usuarios en base de datos con claves encriptadas.
  + Validación de creación de calificaciones con idalumno en nivel medio y nivel técnico.
  + Validación entrada de correo electrónico.
  + Validación de usuario para creación de cursos, docentes, materias.

1. Aprobar el Proyecto

En la aprobación de proyecto se realiza un contrato de licencia de software con cada cliente.

**CONTRATO DE LICENCIA DE SOFTWARE**

**Entre:** [Nombre del Licenciante], con domicilio en [Dirección], en adelante denominado "EL LICENCIANTE".

**Y**

[Nombre del Licenciatario], con domicilio en [Dirección], en adelante denominado "EL LICENCIATARIO".

**1. OBJETO** EL LICENCIANTE concede a EL LICENCIATARIO una licencia de uso del software [Nombre del Software], bajo los términos y condiciones que se detallan a continuación.

**2. ALCANCE DE LA LICENCIA** La licencia otorgada es [exclusiva/no exclusiva], [transferible/no transferible] y [limitada/ilimitada] en el tiempo.

**3. RESTRICCIONES DE USO** EL LICENCIATARIO se compromete a:

* No modificar, copiar, distribuir o sublicenciar el software sin autorización expresa.
* No utilizar el software para actividades ilícitas o que vulneren derechos de terceros.
* No realizar ingeniería inversa, descompilación ni cualquier acción para obtener el código fuente del software.

**4. RESPONSABILIDADES** EL LICENCIANTE no será responsable por:

* Daños derivados del uso indebido del software.
* Fallos de compatibilidad con sistemas no especificados en los requisitos.

EL LICENCIATARIO será responsable de:

* Cumplir con los términos de uso establecidos en el contrato.
* Mantener la confidencialidad de las claves de acceso y configuraciones del software.

**5. DURACIÓN Y TERMINACIÓN** El presente contrato tendrá una duración de [Plazo] y podrá ser rescindido en caso de incumplimiento de las cláusulas establecidas.

**6. LEGISLACIÓN APLICABLE** Este contrato se rige por las leyes de [Jurisdicción].

En señal de conformidad, ambas partes firman el presente contrato en [Lugar] a [Fecha].

[Nombre y Firma del Licenciante] [Nombre y Firma del Licenciatario]

**Técnicas de análisis de requerimientos**

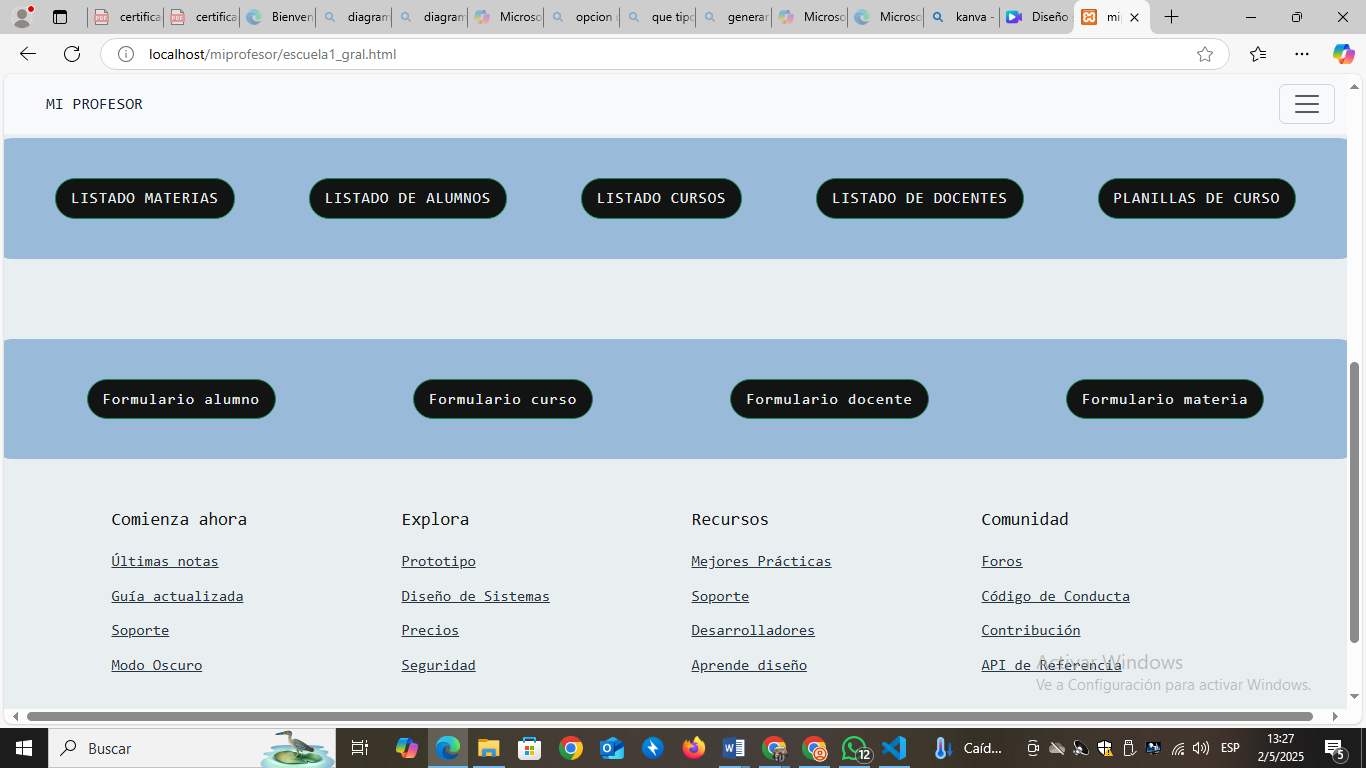
* Entrevistas

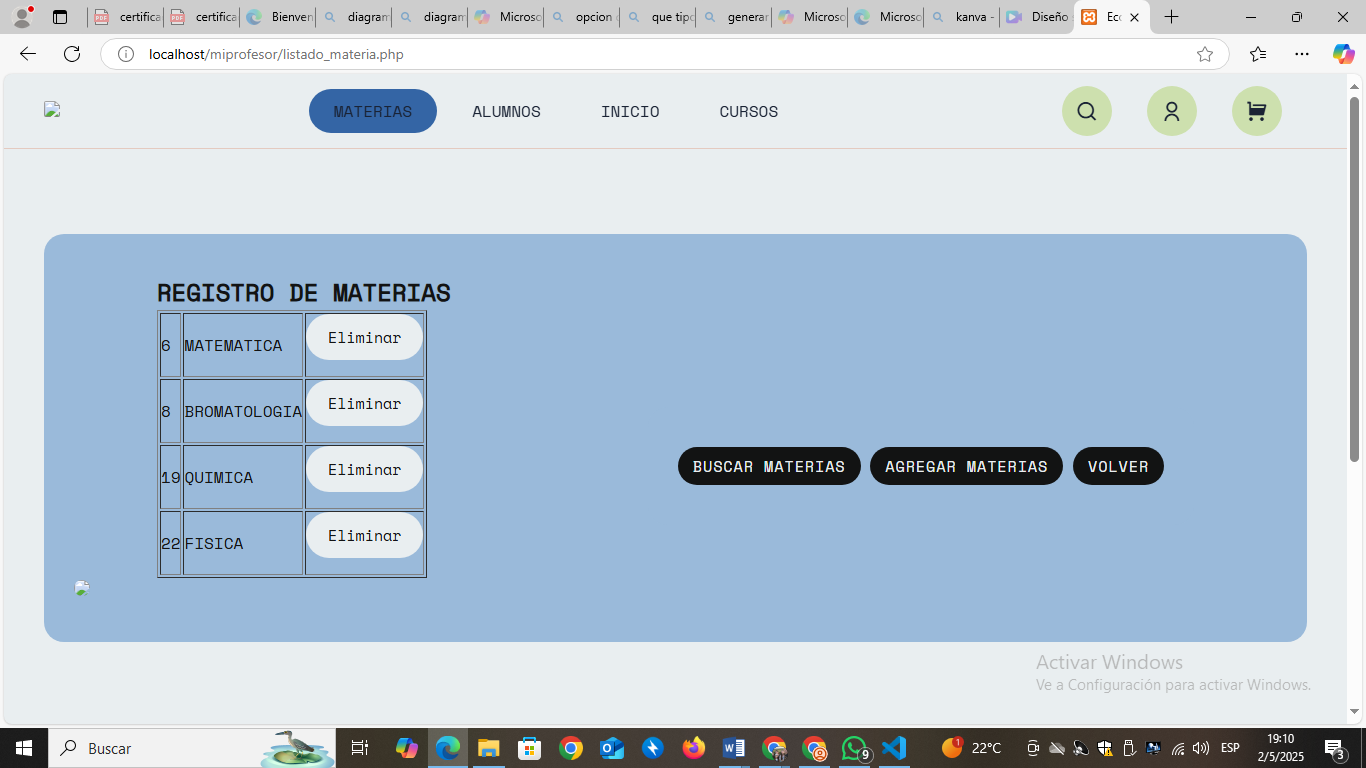
Para poder identificar los requisitos se entrevistó a varios profesores de distintas instituciones. Estos concordaron que sería necesario, un sistema que cuente con los siguientes requerimientos:

1. Ingreso de alumnos (datos como nombre, apellido, y datos de contactos de padres).
2. El sistema cuente con la información de todos los cursos, materias, calificaciones que posea dicho docente.
3. Que se puedan emitir reportes.
4. Que la libreta pueda ser exportada al sistema escolar de la institución.
5. Que los alumnos posean un usuario, para acceder a sus notas y al aula virtual. Esta debe contener material de lectura, y trabajos prácticos.

* Creación de prototipos

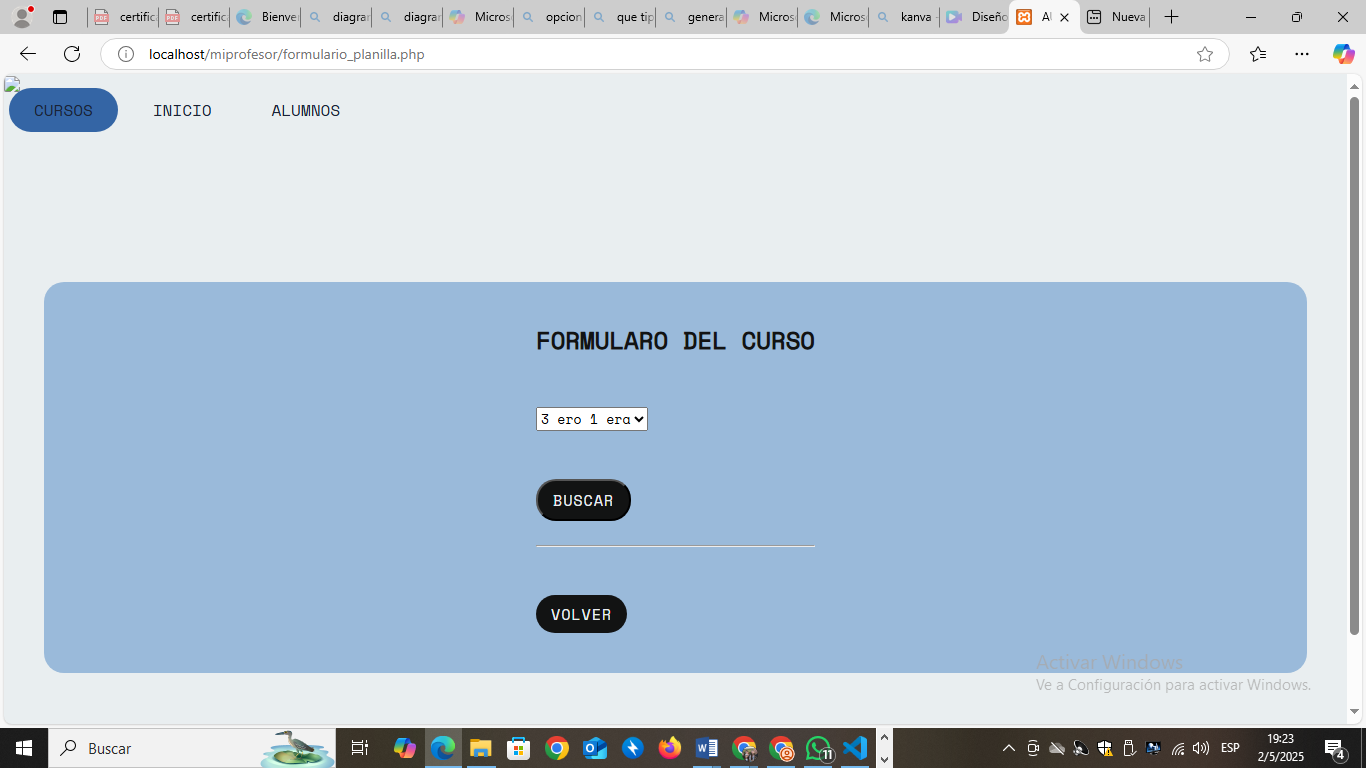




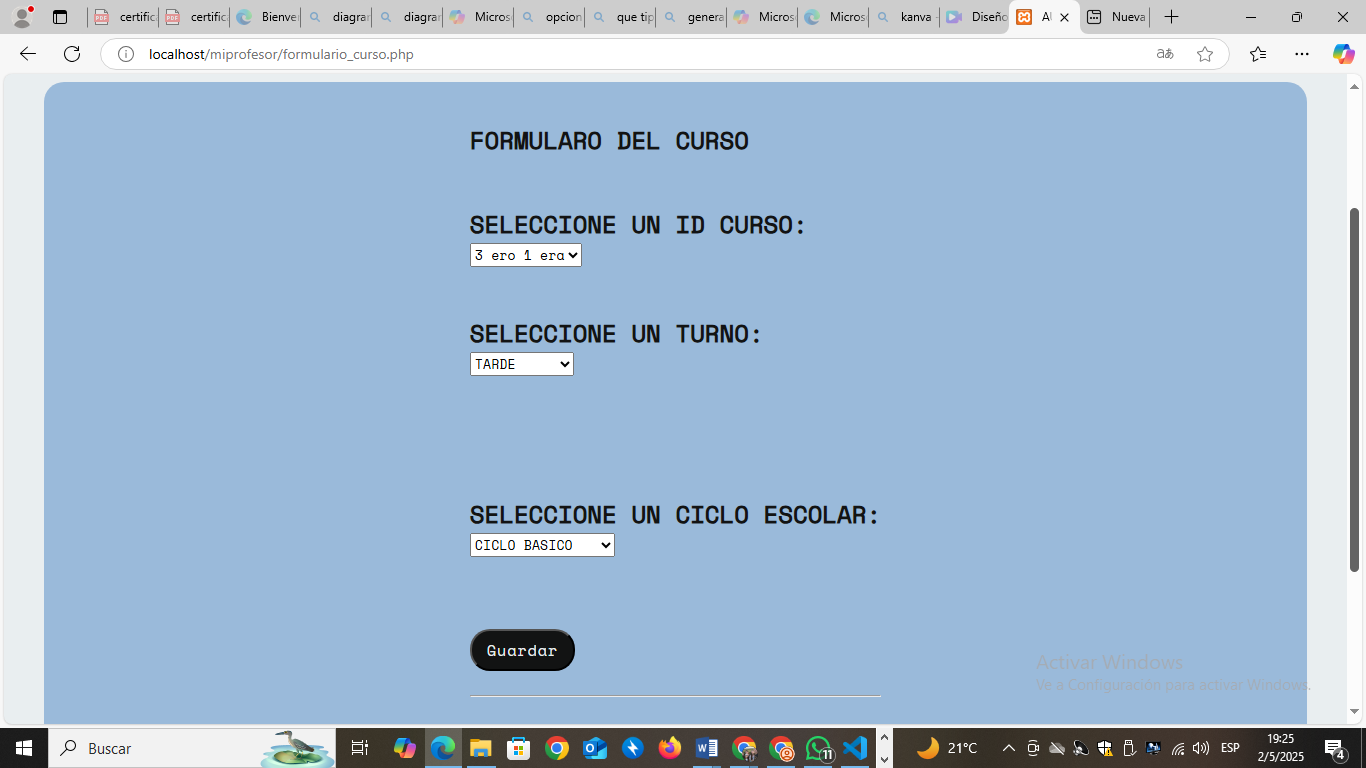


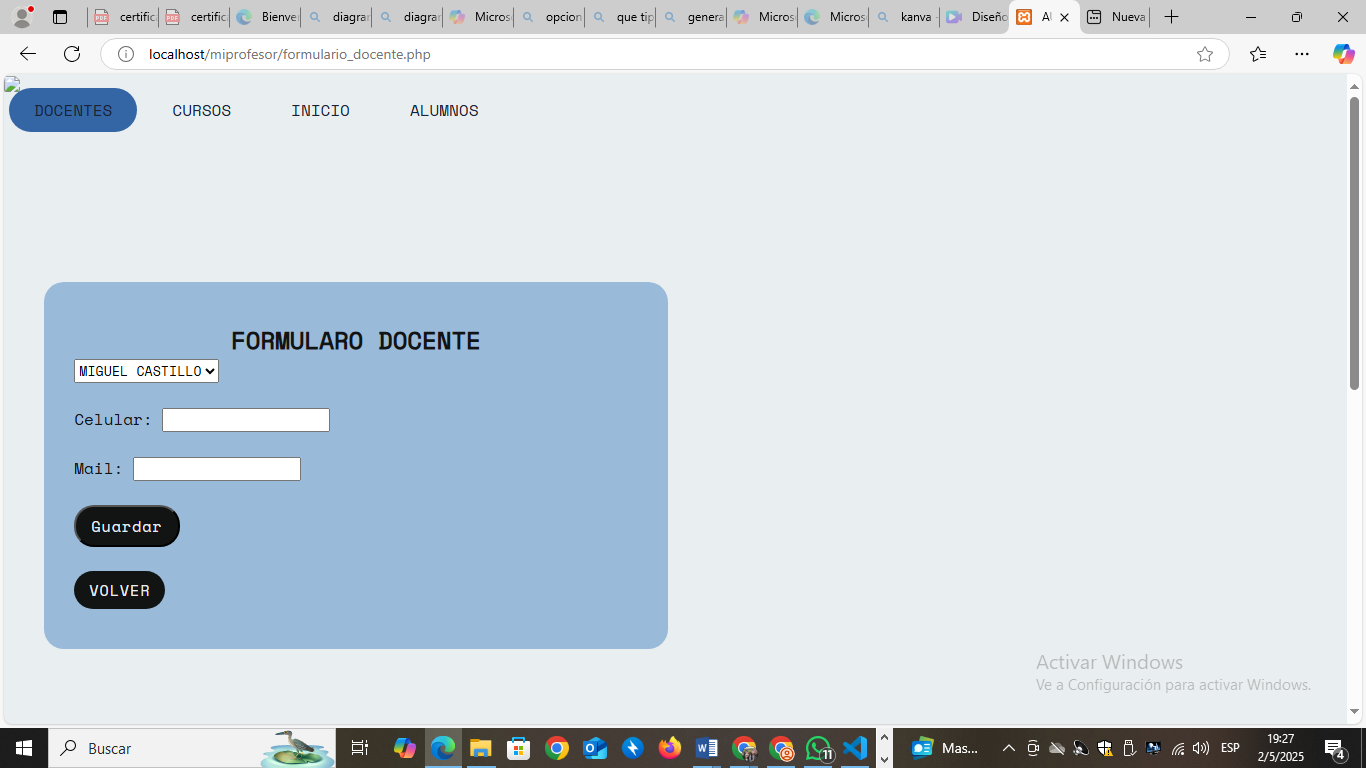


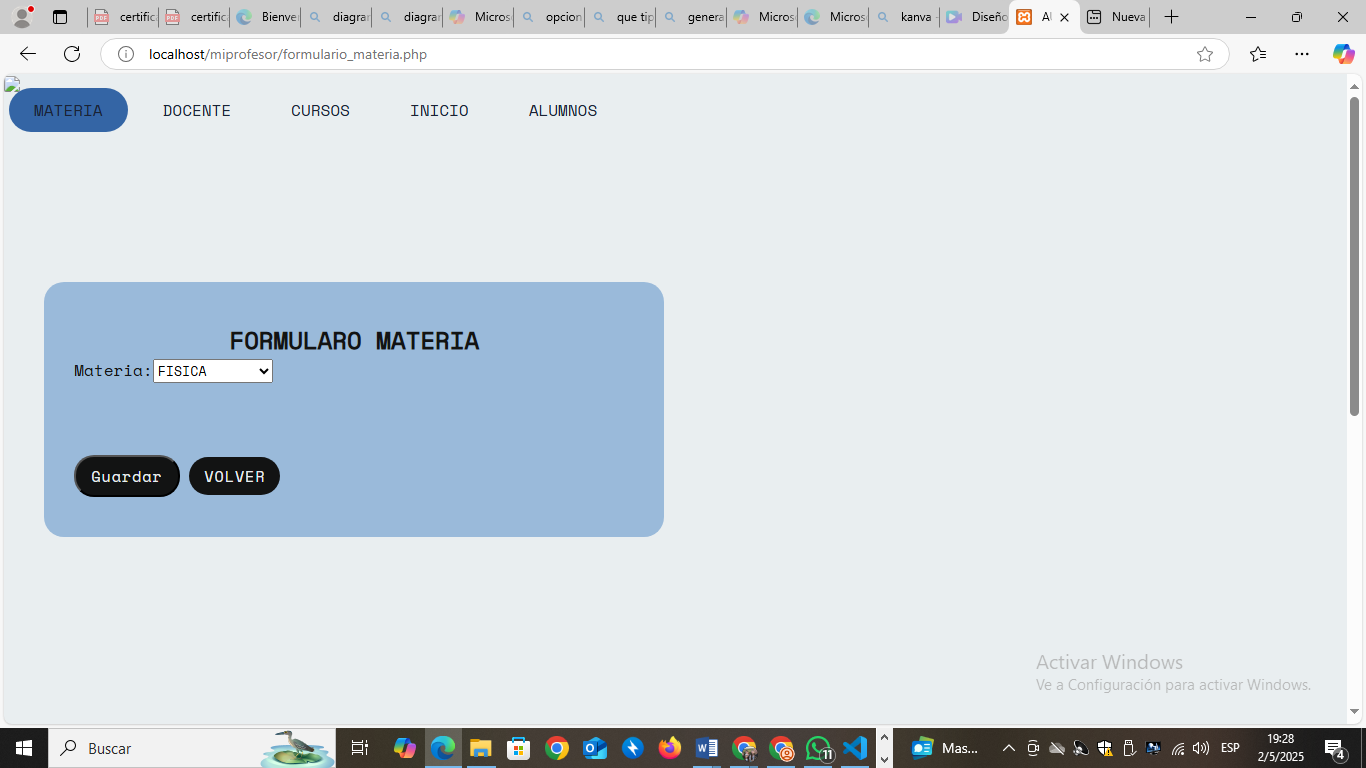




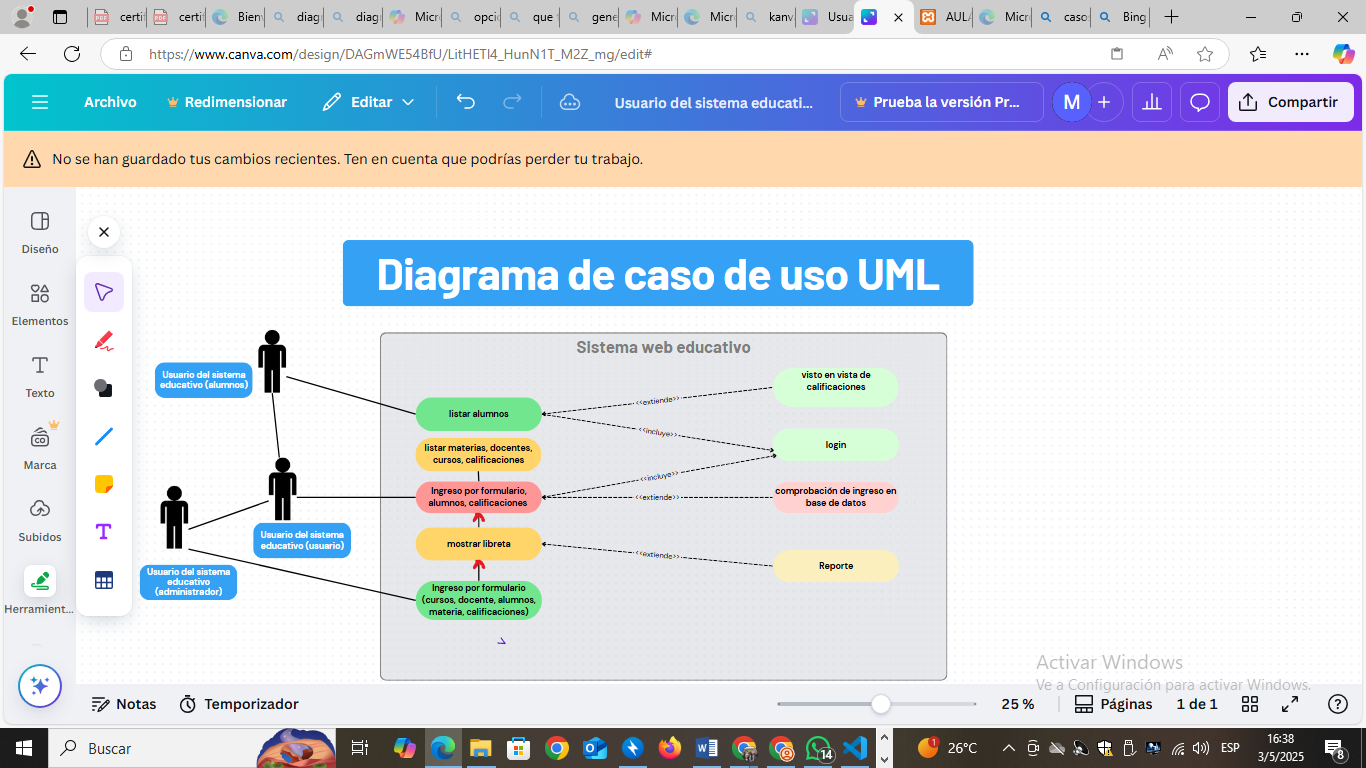




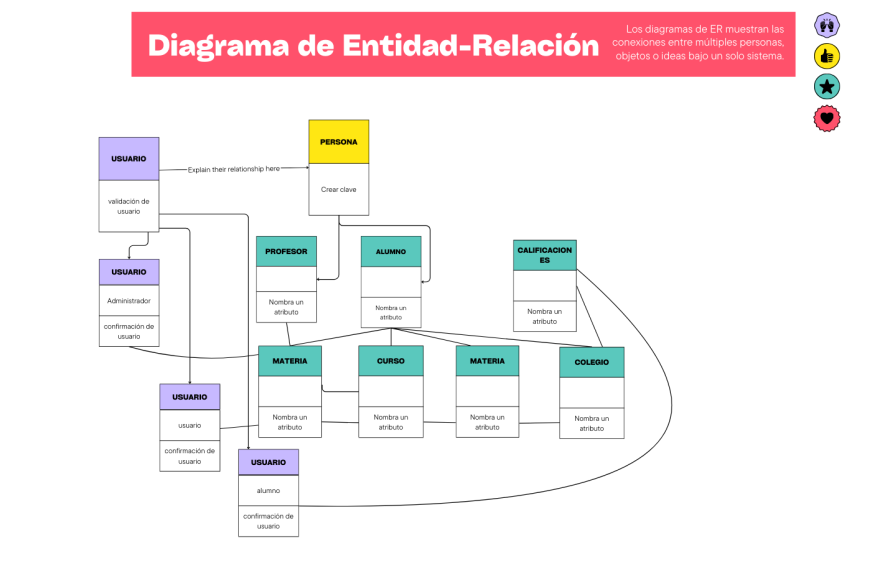




* Análisis de casos de uso



* Lenguaje Unificado de Modelado (UML)



**Importancia del análisis de requerimientos**

**El análisis de requerimientos permite:**

* Conocer y especificar las características del producto
* Definir el alcance del desarrollo
* Comprender la funcionalidad del sistema
* Garantizar la alineación entre los equipos comerciales y técnicos
* Identificar las necesidades completas del cliente
* Especificar las características operacionales del software

PRACTICO O ENTREGA N2 DE PRACTICAS PROFESIONALIZANTE 4

NOMBRE: MIGUEL CASTILLO

De esta manera podemos realizar este cuestionario para entender más datos sobre **nuestro proyecto** y **los problemas que podamos defini**r.

* 1. What -> Se quiere lograr un software educativo. Este debería mantener sus funciones básicas, pero si se produce un problema, nuestro cliente debería infórmanos vía WhatsApp o por llamado. Las definiciones del problema podrían ser:
  + Corte de luz.
  + Ruptura del disco rígido. En el caso que se maneje la información del software in situ.
  + Que el software colapse, por falta de internet, caso de servidor externo.
  + Por alteración de archivos existentes in situ.

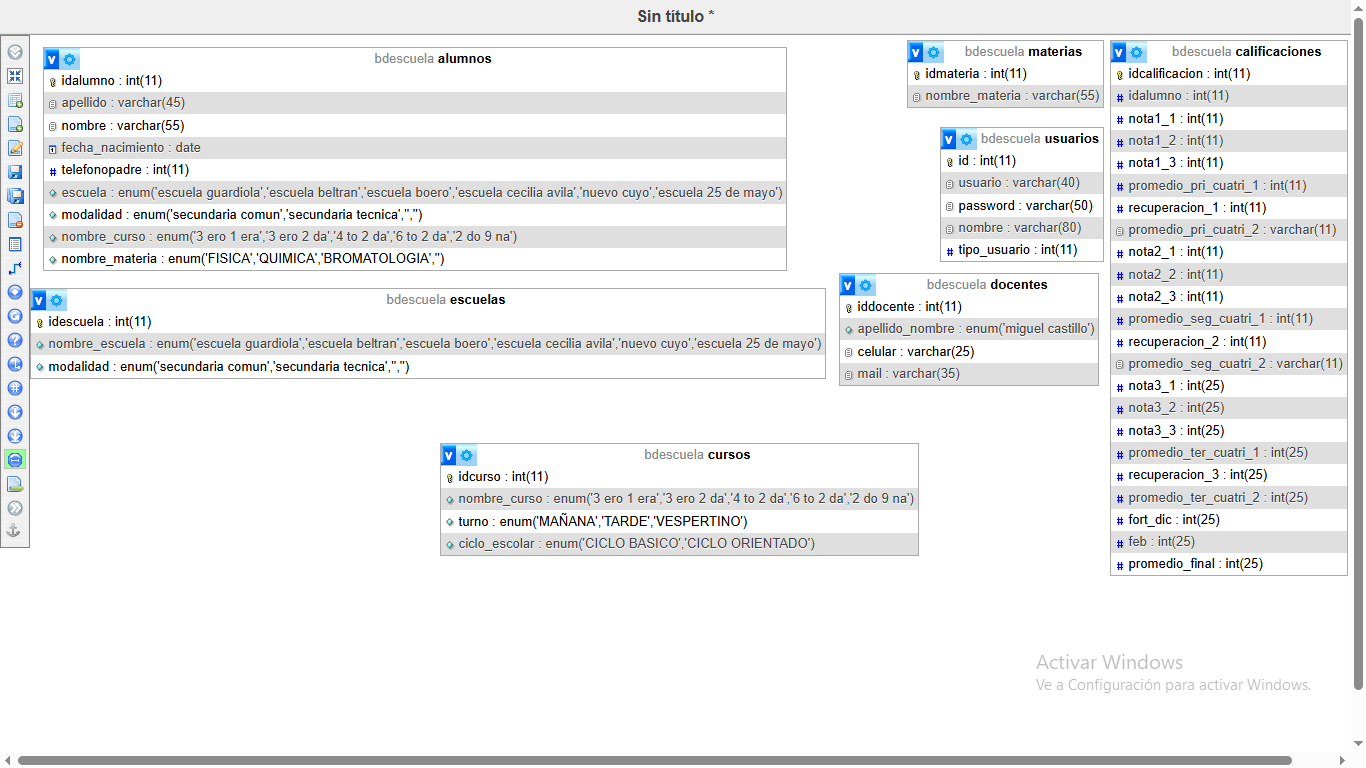
* 1. Why -> Se va a realizar para poder:
  + Aumentar nivel de organización.
  + Para poder aumentar el nivel de tecnología.
  + Para tener un plan de contingencia, en caso de mal funcionamiento, bases de datos sin conexión o problemas de actualización.
  1. Where -> Se va a realizar un sistema educativo, con conexión a internet, proveedor con servidor o en las computadoras personales. Para esto podemos tener un sistema con reseñas donde el consumidor final manifieste disconformidad o aceptación. Por otro lado, podemos colocar distintas vías de atención al cliente, con solución de problemas.
  2. Who -> Se realizará por el técnico desarrollador de software Miguel Castillo.
  3. When-> Se estima iniciar en el mes de mayo en el proyecto.
  4. How-> Se puede realizar haciendo el ciclo de vida de un producto o software. En este caso podemos definir como son las vías para resolver el problema. Algunos ejemplos son:
  + TeamViewer, AnyDesk, Zoom, Microsoft Remote Desktop y Google Workspace Assistan.
  + Asistencia personalizada.
  1. Wow many-> En este caso tenemos que definir cuáles son los niveles de urgencia de asistencia.

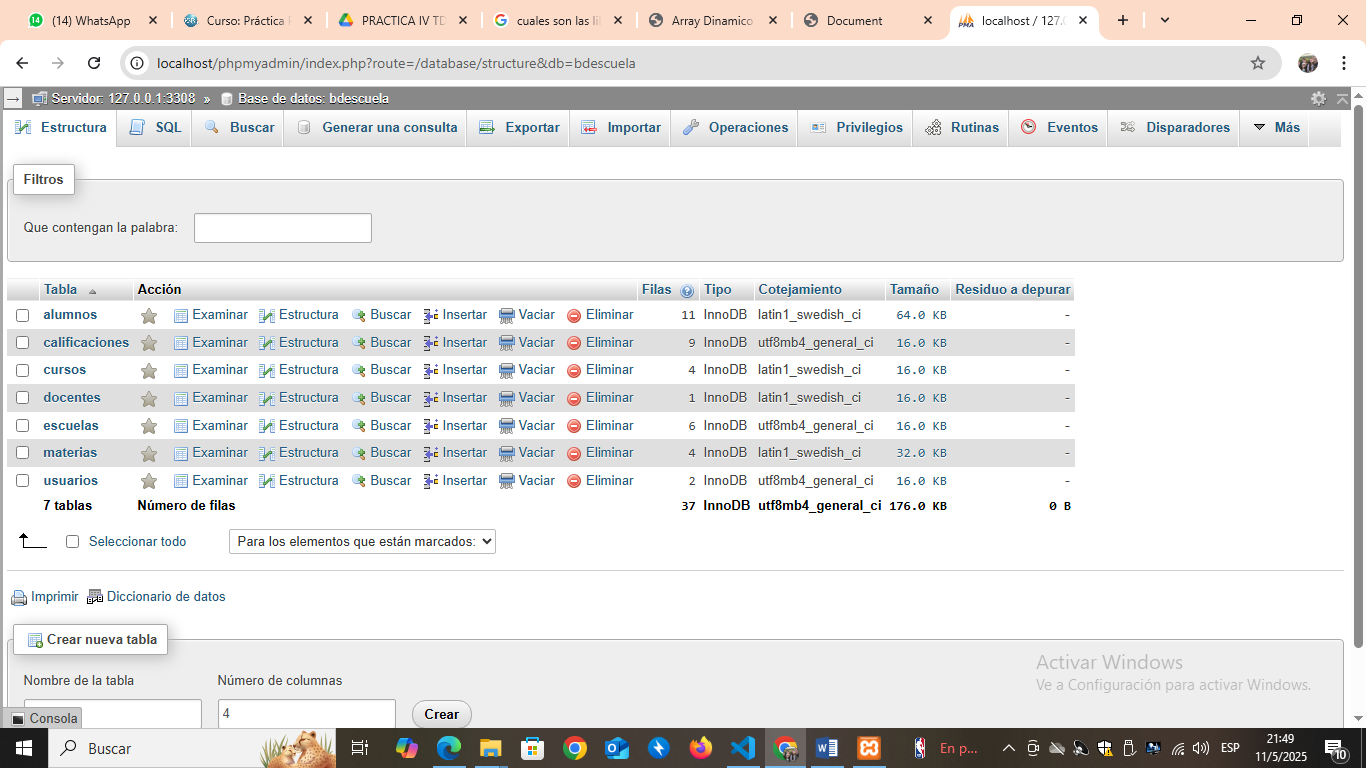
Alto (mucha urgencia o asistencia inmediata), medio (solucionar al cabo del día) y ocasional (es un problema menor).

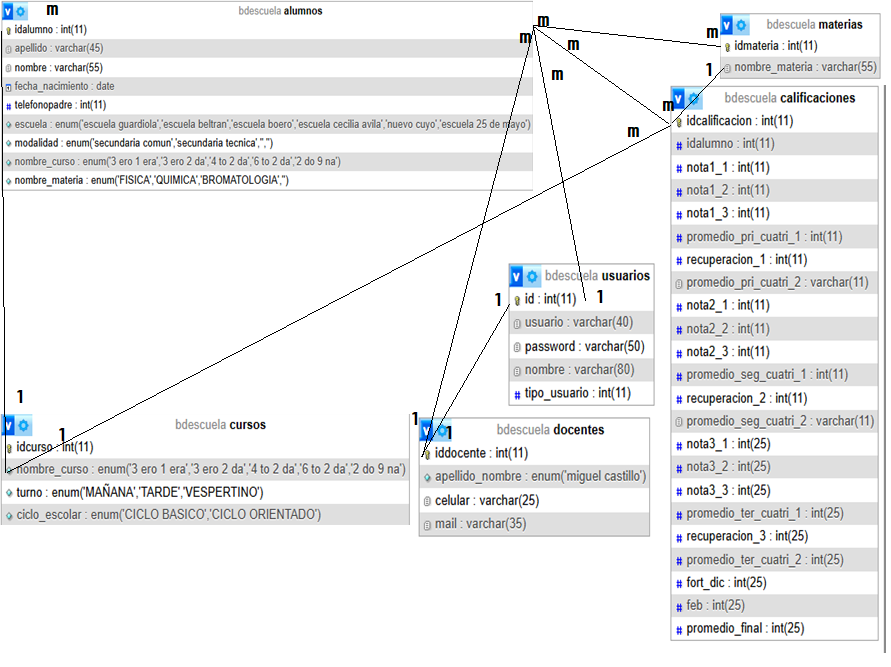
Licencia: Privada con licencia de uso sin replica.

PRACTICO O ENTREGA N3 DE PRACTICAS PROFESIONALIZANTE 4

**SE REALIZA UN DIAGRAMA DE BASES DE DATOS EN MYSQL**







PRACTICO O ENTREGA N4 DE PRACTICAS PROFESIONALIZANTE 4

Realización de un bot para el proyecto

Usuario: [miguelarturocastillo@hotmail.com](mailto:miguelarturocastillo@hotmail.com)

Contraseña: Mcastillo23!