PRACTICO O ENTREGA N1 DE PRACTICAS PROFESIONALIZANTE 4

**Etapas del análisis de requerimientos**

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

Se destina al uso por parte de ***profesores y alumnos*** de educación nivel secundaria común y técnica. La parte interesada es:

* + Profesores.
  + Preceptores.
  + Directivos del establecimiento.
  + Autoridades del sistema educativo.
  + Alumnos.

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

Diseñar un software que ayude a docentes o profesores, con carga de datos personales, de alumnos en cada institución. Que genere reportes con estados de calificaciones. Los objetivos son:

* + Controlar y proteger el ingreso de datos.
  + Formar bases de datos para cada variable.
  + Generar reportes a las distintas instituciones, con información del 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, obtiene informes o reportes finalizados.
   * El software garantiza el ingreso y protección de datos. Se utiliza tanto en un entorno de computación en la nube, a través de un proveedor de servicios, como en una computadora portátil, ofreciendo flexibilidad y adaptabilidad según las necesidades del usuario.
2. **Categorizar los requisitos del proyecto**

Requerimientos funcionales

El software cumple con:

* + Registrar datos personales de alumnos, ingreso calificaciones, carga docente, actualización de calificaciones, asignar periodo de cursado (trimestres o cuatrimestres).
  + Relacionar cada alumno con el curso e institución, su turno y docente implicado, etc.
  + Generar reportes de estado de libreta.
  + Generar listados de alumnos, docente, cursos y materias.

Requerimientos no funcionales:

El software cumple con ser:

* + Herramienta eficiente.
  + Posee seguridad de datos, copias diarias.
  + Intuitivo y de uso sencillo.

1. **Analizar los requisitos**
   * Ingreso con distintos niveles de usuarios.
   * Diseño Interfaz de Usuario (UI) con BOOTSTRAP apto para el profesor.
   * Diseño del código en lenguaje PHP. Se establece la funcionalidad del sistema (UX).
   * Creación de bases de datos según 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 incluye 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.
  + Descargar información.

**administrador:**

* + Consultar listado materias, agregar /buscar materias. 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**

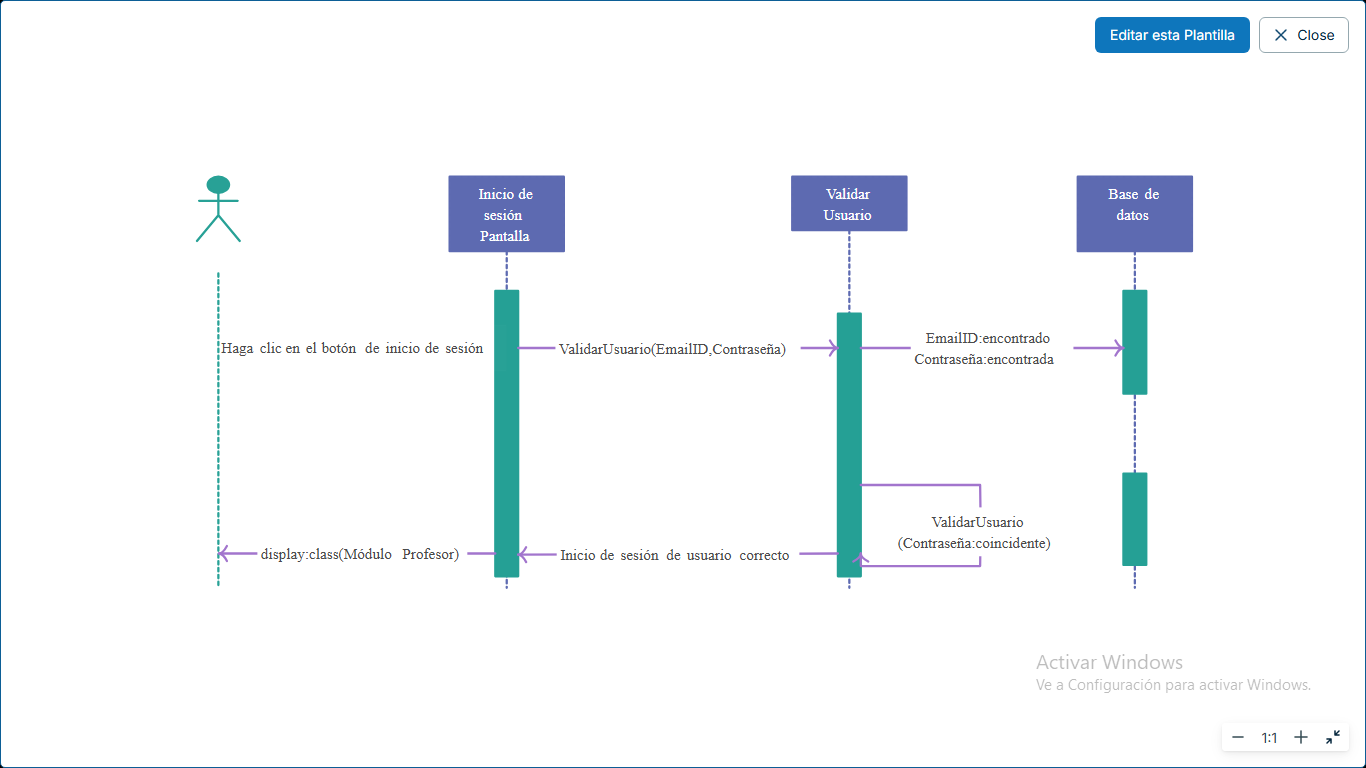
Modela las principales clases del sistema:

* **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**
  + Emisión de reporte en PDF.
* **Aula virtual**
  + Ingreso de información en PDF para descargar.

**4. Diagrama de Secuencia**

Describe cómo interactúan los objetos, se analiza 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]