**Asistencia automática (BD)**

Objetivos

Desarrollar una aplicación de escritorio en C# que permita gestionar y controlar la asistencia de los alumnos de manera eficiente y segura brindando acceso según el rango de usuario.

Descripción

ES una aplicación de asistencia automática para facilitar el trabajo de los preceptores y llevar un mejor control del mismo, este sistema va a contar con la creación de la base de datos en MYSQL SERVER en el cual se van a crear y almacenar los datos suficientes para su funcionamiento,

Requerimientos

Se requiere un sistema el cual contabilice y controle las inasistencias a través de un lector de huella y una aplicación de escritorio la cual gestione la alta, baja y modificación de los alumnos; considerando aspectos visuales y de comodidades para el operador.

Solución funcional

Desarrollar una aplicación de escritorio en C# que permita gestionar la asistencia de los alumnos de manera eficiente y segura brindando acceso según el rango del usuario.

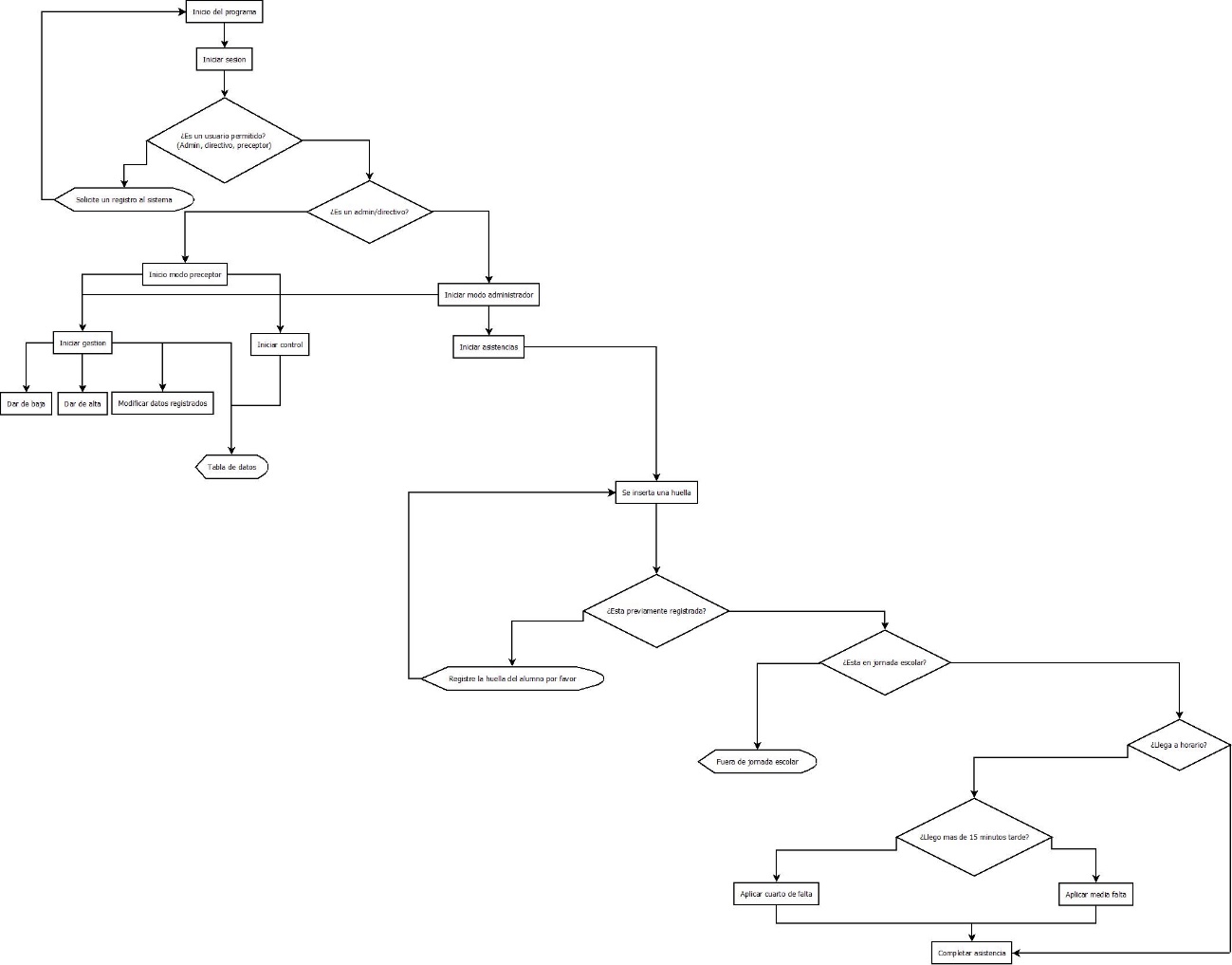
**Tablas**

* + **Tabla: alumnos**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_curso (fk) Numérico (int) NOT NULL**
  + **dni Varchar (255) NOT NULL**
  + **huella Varchar (255) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **apellido Varchar (255) NOT NULL**
  + **direccion Varchar (255) NOT NULL**
  + **sexo Varchar (255) NOT NULL**
  + **país Varchar (255) NOT NULL**
  + **provincia Varchar (255) NOT NULL**
  + **ciudad Varchar (255) NOT NULL**
  + **nacionalidad Varchar (255) NOT NULL**
  + **fecha\_nac Varchar (255) NOT NULL**
  + **obra\_social Varchar (255) NOT NULL**
  + **correo Varchar (255) NOT NULL**
  + **inacistencias Numérico (Float) NOT NULL**
  + **tel\_adulto Varchar (255) NOT NULL**
  + **Índice:**
  + **ialumnos\_nombre\_apellido nombre ASC + apellido ASC**
  + **ialumnoLdni dni ASC ialumnos\_curso curso ASC**
  + **Tabla: ingreso**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_alumno (fk) Numérico (int) NOT NULL**
  + **entrada Time (7) NOT NULL**
  + **Índice:**
  + **iingreso\_entrada entrada ASC**
  + **iingreso\_id\_alumno (fk) id\_alumno ASC**
  + **Tabla: egreso**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_alumno (fk) Numérico (int) NOT NULL**
  + **salida Time (7) NOT NULL**
  + **Índice:**
  + **iegreso\_salida salida ASC**
  + **iegreso\_id\_alumno id\_alumno ASC**
  + **Tabla: division**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **Índice:**
  + **idivision\_nombre nombre ASC**
  + **idivision\_id (pk) id ASC**
  + **Tabla: curso**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_administrador (fk) Numérico (int) NOT NULL**
  + **id\_division (fk) Numérico (int) NOT NULL**
  + **id\_grupo (fk) Numérico (int) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **Índice:**
  + **icurso\_nombre nombre ASC**
  + **icurso\_id (pk) id ASC**
  + **Tabla: materia**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_profesor (fk) Numérico (int) NOT NULL**
  + **id\_horario (fk) Numérico (int) NOT NULL**
  + **id\_curso (fk) Numérico (int) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **Índice:**
  + **Imateria\_id (pk) id ASC**
  + **Imateria\_nombre nombre ASC**
  + **Imateria\_profesor profesor ASC**
  + **Tabla: profesor**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **apellido Varchar (255) NOT NULL**
  + **dni Varchar (255) NOT NULL**
  + **Índice:**
  + **Iprofesor\_id (pk) id ASC**
  + **Iprofesor\_nombre\_apellido nombre ASC + apellido ASC**
  + **Iprofesor\_dni dni ASC**
  + **Tabla: horarios**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **hora Time (7) NOT NULL**
  + **dia Time Day (7) NOT NULL**
  + **Índice:**
  + **Ihorarios\_dia\_hora dia DESC + hora DESC**
  + **Ihorarios\_id (pk) id ASC**
  + **Tabla: grupo**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_administrador (fk) Numérico (int) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **Índice:**
  + **Igrupo\_id (pk) id ASC**
  + **Igrupo\_nombre nombre ASC**
  + **Tabla: administradores**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_permisos (fk) Numérico (int) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **apellido Varchar (255) NOT NULL**
  + **dni Varchar (255) NOT NULL**
  + **Índice:**
  + **Iadministradores\_nombre\_apellido nombre ASC + apellido ASC**
  + **Iadministradores\_dni dni ASC**
  + **Iadministradores\_id (pk) id ASC**
  + **Tabla: alumnostemporales**
  + **Campos:**
  + **id (pk) Numérico (int) NOT NULL**
  + **id\_curso (fk) Numérico (int) NOT NULL**
  + **dni Varchar (255) NOT NULL**
  + **nombre Varchar (255) NOT NULL**
  + **apellido Varchar (255) NOT NULL**
  + **sexo Varchar (255) NOT NULL**
  + **pais Varchar (255) NOT NULL**
  + **provincia Varchar (255) NOT NULL**
  + **ciudad Varchar (255) NOT NULL**
  + **nacionalidad Varchar (255) NOT NULL**
  + **fecha\_nac Date NOT NULL**
  + **obra\_social Varchar (255) NOT NULL**
  + **correo Varchar (255) NOT NULL**
  + **inasistencias Numérico (Float) NOT NULL**
  + **tel\_adulto Varchar (255) NOT NULL**
  + **Índice:**
  + **ialumnostemporales\_nombre\_apellido nombre ASC + apellido ASC**
  + **ialumnostemporales\_dni dni ASC ialumnostemporales \_curso curso ASC**

**Back-End**

* **Funciones:**

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre Funcion | Descripción | Parámetros Entrada | Parámetros Salida |
| Conexion | Conecta a la base de datos. Devuelve un dato booleano dependiente del éxito de conexión |  | Conexión: Bool |
| GetAsistencias | Muestra las asistencias  Del alumno |  | Asistencias: Int |
| GetHorarioEntrada | Muestra el horario que ingreso la huella al entrar |  | HorarioEntrada: Date |
| GetHorarioSalida | Muestra el horario que ingreso la huella al salir |  | HorarioSalida: Date |
| CalcularFaltaAcumulada | Calcula el total de falta acumulada durante el día (entre 0 y 1) |  | FaltaAcumulada: Float |
| AplicarFaltaAcumulada | Aplica la falta acumulada a las faltas en la base de datos |  |  |
| GetProximaClase | Muestra la próxima clase que tiene un alumno o profesor |  |  |
| FiltrarAlumno | Filtra a todos los alumnos por el dato que se precise | Dato de búsqueda (puede ser: DNI, ID, nombre, etc) | Lista alumnos que coinciden con el filtro |
| GuardarAlumno | Guardar a un alumno en la base de datos |  |  |
| GuardarProfesor | Guardar a un profesor en la base de datos |  |  |
| GuardarPreceptor | Guardar a un preceptor en la base de datos |  |  |
| BajaAlumno | Da de baja a un alumno seleccionado |  |  |
| BajaProfesor | Da de baja a un profesor seleccionado |  |  |
| BajaPreceptor | Da de baja a un preceptor seleccionado |  |  |
| CerrarConexionBD | Cierra la conexión a la base de datos |  |  |

**Diagrama****Objetivo:** Mostrar el funcionamiento de la aplicación

**Front-End**

¿Qué información se necesita para dar de alta a un alumno? ¿qué datos se pueden modificar?

GESTION alta-baja-modificaciones

Datos para dar de ALTA:

* Alumno: Apellidos, nombres, DNI, dirección, Cuil, obra social, certificado de egreso escuela primaria, vacunas completas, sexo, fecha de nacimiento, país, provincia, localidad, nacionalidad, escuela origen.
* Adulto responsable: Nombre completo, DNI, Cuil, dirección, teléfono, localidad, nacionalidad, email de adulto responsable, profesión, Tel madre, padre.
* Adultos autorizados a retirar (máximo 4): Fotocopia DNI, firma en original, número de teléfono y la relación que tiene con el alumno.

Datos que se MODIFICAN del alumno (datos que cambian con el tiempo):

* Alumno: Domicilio, obra social
* Adulto responsable: Domicilio, teléfono, email, profesión
* Adultos autorizados

BAJA de un alumno: Nombre, apellido, DNI, curso, división, grupo.

¿En dónde quedan registradas las altas y bajas?

Cuando un alumno es dado de baja se borra del sistema y queda registrado en una carpeta formato papel, las altas quedan registradas en el sistema y en una carpeta.

Pantallas

1)Pantalla INICIO SESION

* Pantalla login: dos tipos de usuario (preceptores y directivos).
* Botones: Usuario, contraseña, iniciar y olvidaste tu contraseña.

2)Pantalla CONTROL/GESTIÓN

* Gestión: altas, bajas, modificaciones.
* Control: lista de alumnos, filtrar por: [Nombre, Apellido, DNI, Curso]

¿Quién realiza las altas-bajas?

Permisos:

* Preceptores:

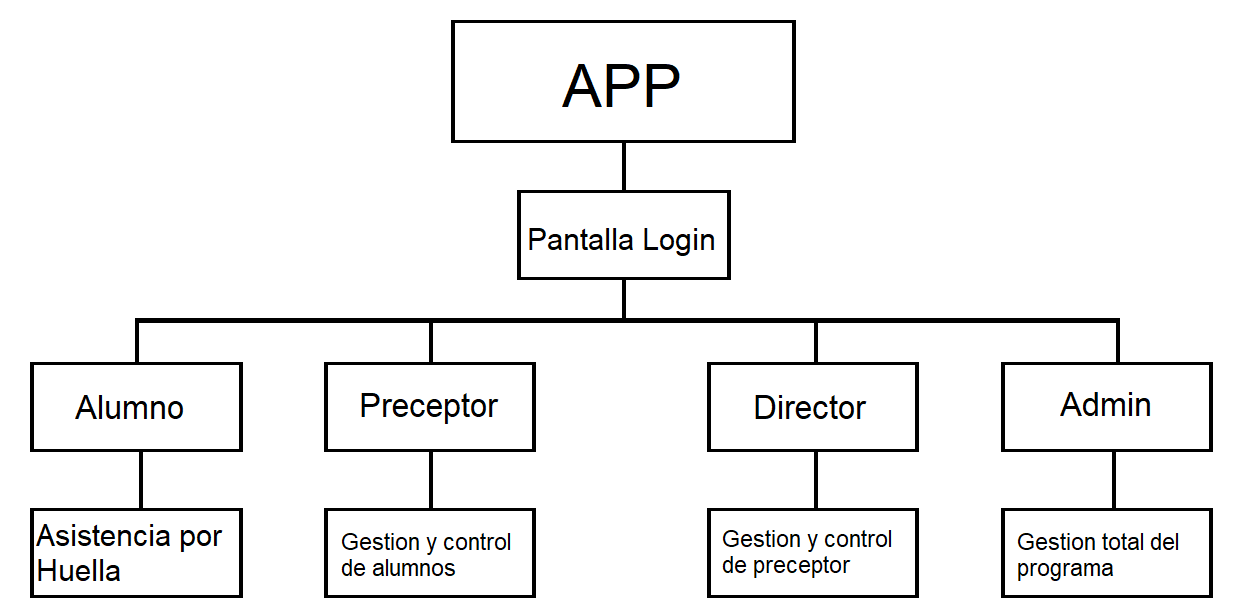
CONTROL: controlar asistencias

GESTION: altas y modificaciones

* Directivos:

CONTROL: ver asistencias

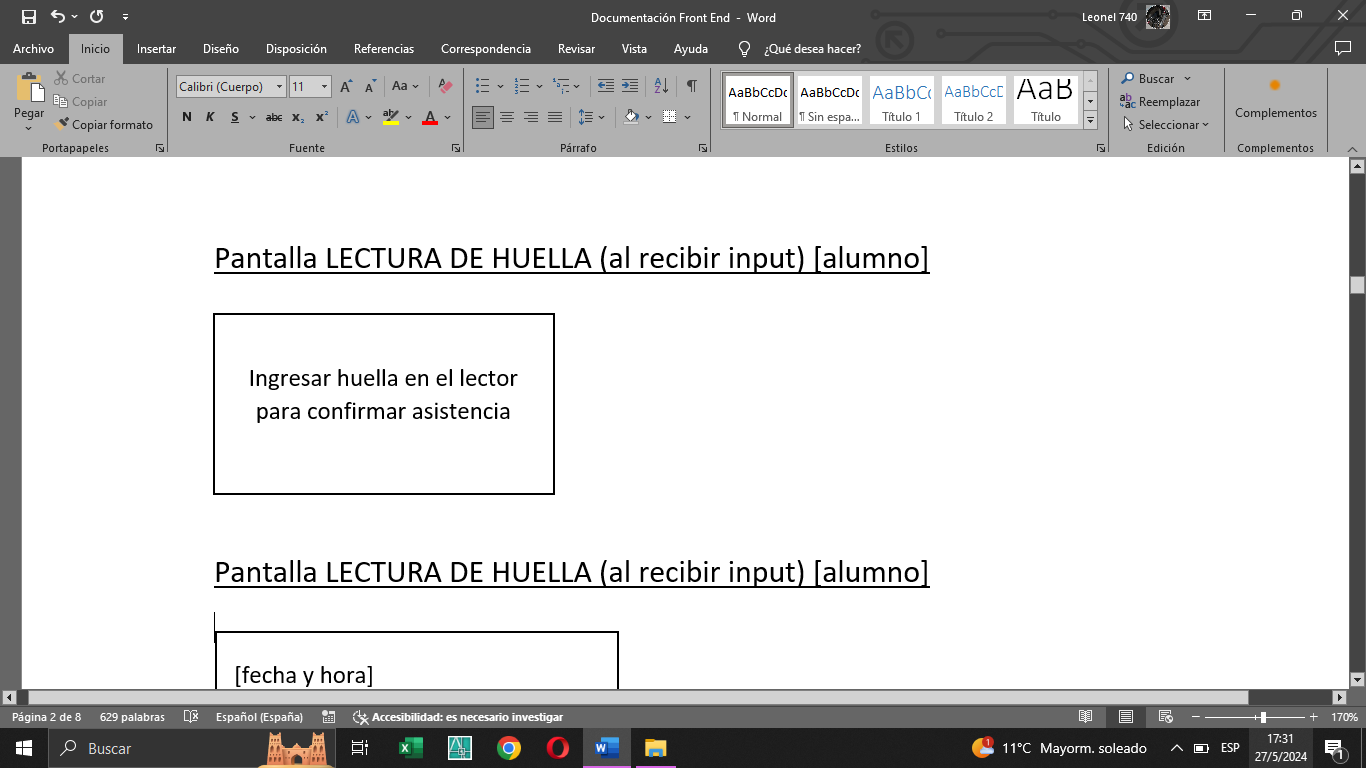
GESTION: altas y bajas



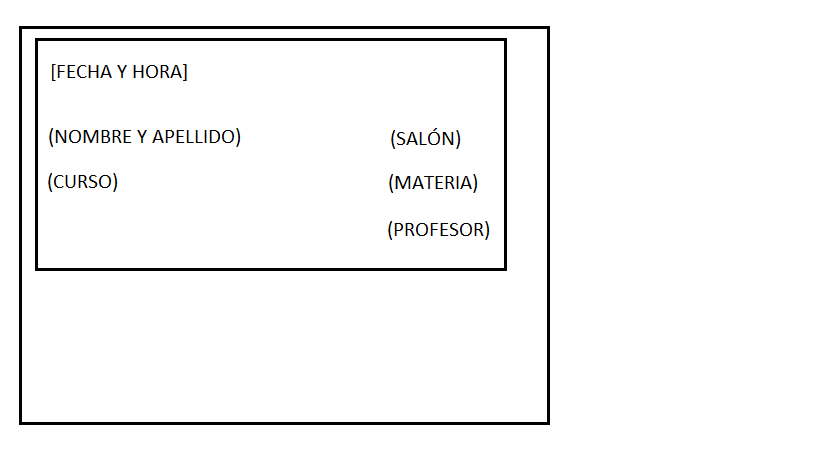
¿Quién forma parte del equipo de conducción? ¿Qué rol cumpliría cada uno en el sistema?

El equipo de conducción está conformado por el director, los vicedirectores, el secretario y el jefe de área. Los directivos están encargados de lo pedagógico, lo administrativo y lo social, la secretaria esta encargada de lo administrativo y de las altas de los ingresantes a primer año.

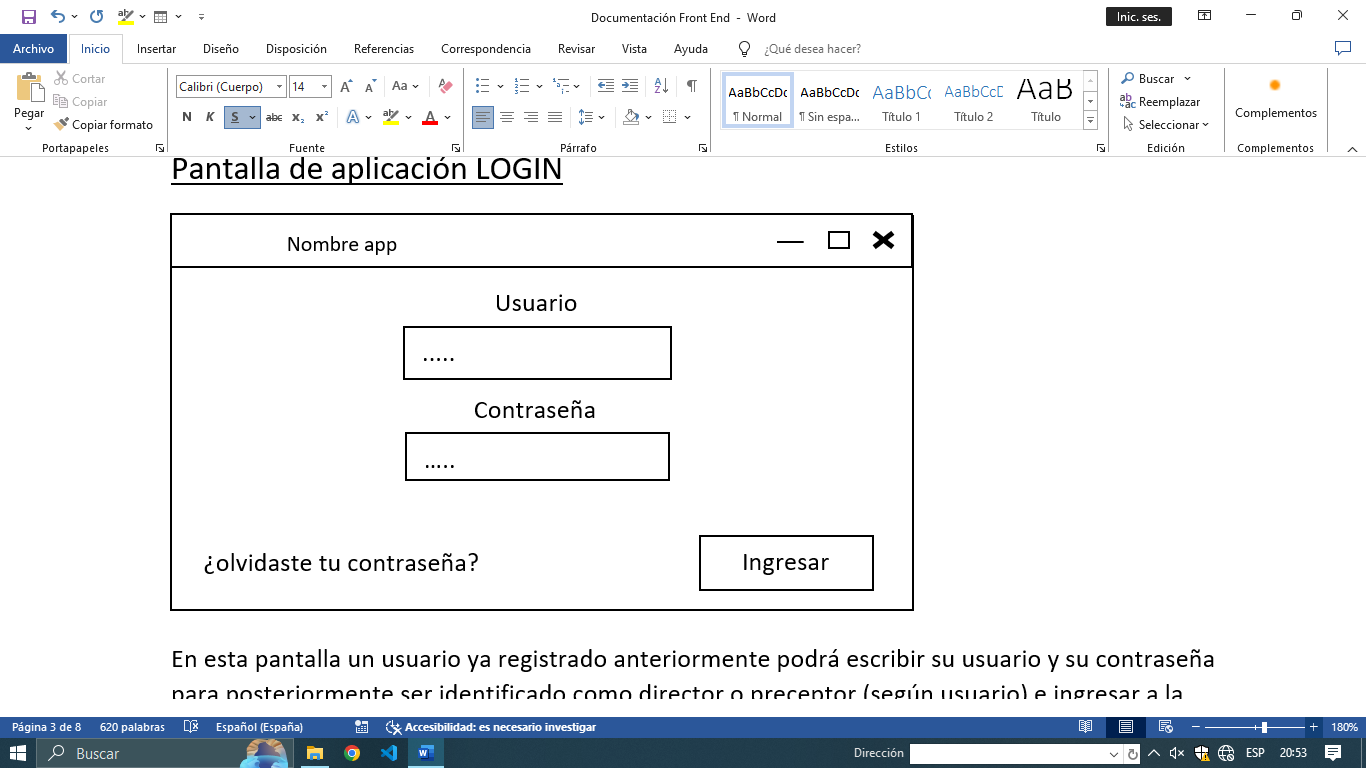
Pantalla LECTURA DE HUELLA (al recibir input) [alumno]



Pantalla LECTURA DE HUELLA (al recibir input) [alumno]



Pantalla de aplicación LOGIN:



(LOGO)

En esta pantalla un usuario ya registrado anteriormente podrá escribir su usuario y su contraseña para luego ser identificado como director o preceptor (según usuario) e ingresar a la siguiente pantalla (control/gestión) con el botón “Ingresar”. También habrá un botón “¿olvidaste tu contraseña?” donde se podrá recuperar la contraseña en caso de olvidarla.

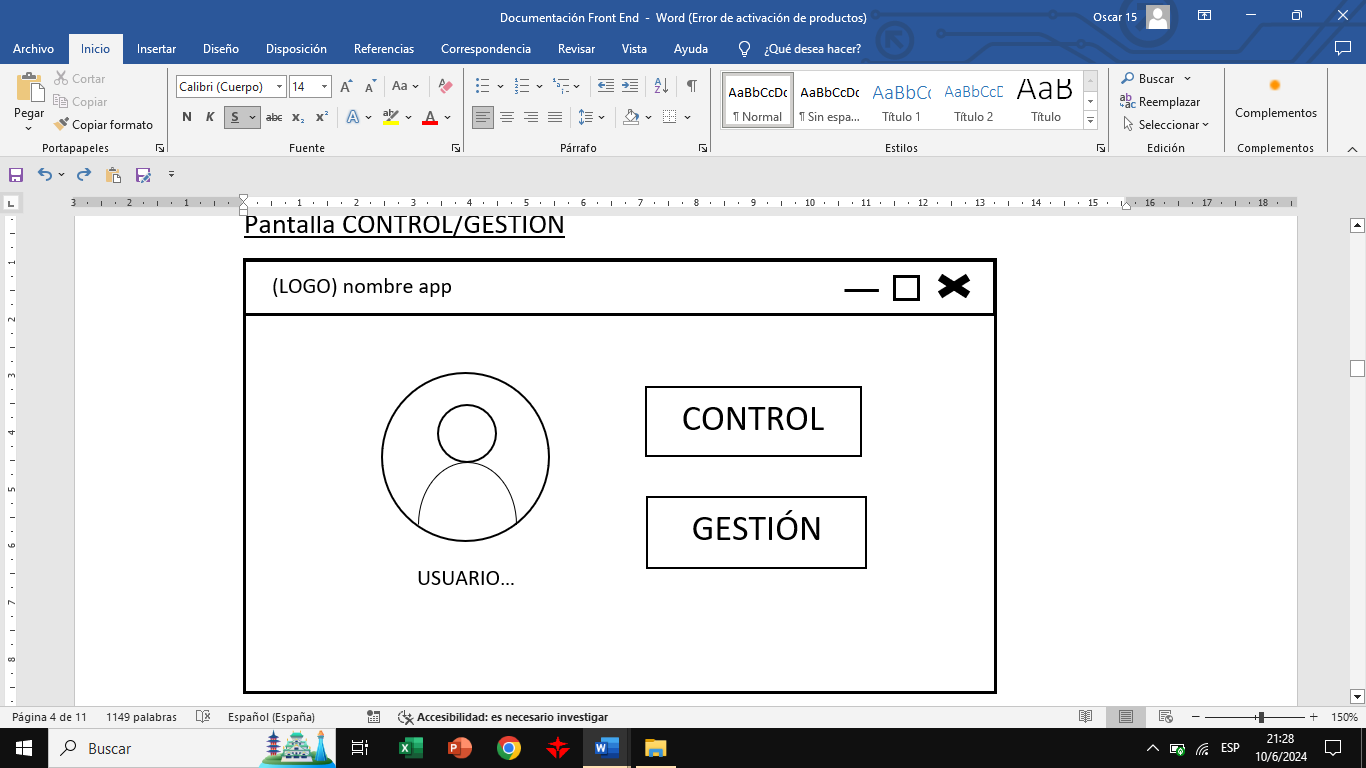
CONTROL

(LOGO) nombre app



(LOGO) nombre app

Pantalla CONTROL/GESTION



Pantalla GESTION

(LOGO) nombre app

USUARIO

GESTIÓN

CONTROL



ALTA

BAJA

MODIFICACIONES

Pantalla GESTION (ALTA)

(LOGO) nombre app



GESTIÓN

CONTROL

Datos del alumno a ingresar



Nombre…

Apellido…

ALTA

BAJA

Código Huella

DNI…

Fecha de nacimiento:

Edad.

Día - Mes - Año



MODIFICACIONES

Cursoad.

Sexo

(LOGO) nombre app



GESTIÓN

CONTROL

Datos del alumno a ingresar



Dirección

Nacionalidad

ALTA

Obra social

Correo

Teléfono adulto

BAJA

País

Provincia….

Ciudad…

Ingresar

MODIFICACIONES

Pantalla GESTIÓN (BAJA)

(LOGO) nombre app



GESTIÓN

CONTROL



Nombre Nombre..

..

DNI..

Curso..

Apellido..

|  |  |  |  |
| --- | --- | --- | --- |
| …. | …. | …. | …. |
| … | …. | …. | …. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |



ALTA



BAJA

MODIFICACIONES

Pantalla GESTIÓN (MODIFICACIONES)

(LOGO) nombre app



GESTIÓN

CONTROL



DNI

Domicilio

..

Teléfono

Correo

..

Curso

..

ALTA

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |



BAJA

MODIFICACIONES

(LOGO) nombre app



GESTIÓN

CONTROL



DNI



Obra social

..

ALTA

|  |
| --- |
|  |

Modificar

BAJA

MODIFICACIONES

Pantalla CONTROL

(LOGO) nombre app



GESTIÓN

CONTROL



Nombre Nombre..

..

Apellido..

Curso..

DNI..

. . .

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Codhuella | Nombre | Apellido | DNI | Curso | Edad | Correo |
| … | … | … | … | … | … | ... |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |



(LOGO) nombre app



GESTIÓN

CONTROL



. . .

Curso..

Nombre Nombre..

..

Apellido..

DNI..

|  |  |  |
| --- | --- | --- |
| Asistencias | Faltas | Justificadas |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |



**Plan de Trabajo**

* Finalizar la base de datos, con sus tablas, atributos, relaciones.
* Rellenar tabla con registros e información.
* Validaciones de base de datos (creación, modificación, eliminación).
* Crear una aplicación en C#.
* Dividir aplicación 4 capas.
* Primera capa: Capa base de datos, crear conexión y comprobarla.
* Pantallas/funciones.
* Declarar funciones.
* Diseñar estructura del código.
* Desarrollo de las pantallas.
* Conexión con lector de huella.
* Validación del lector.
* Recorrido de la aplicación (testear)
* Implementación.

**Estrategia Validación**

**Primera etapa**

1. Prueba 1:

Se testeo como trabaja cada uno en el ambiente de su grupo, y luego se reorganizaron estos hasta que cada integrante este cómodo y tenga su función dentro del proyecto

**Segunda etapa**

1. Prueba 1:

Se testea si todos los botones y los inputs en el programa sirven y si la interacción con ellos es la deseada.

1. Prueba 2:

Se llevarán a cabo pruebas iniciando sesión en la app como usuarios de distintos tipos de nivel para comprobar que los permisos que tenga cada persona sean los correctos, y que nadie mas que el equipo de trabajo del proyecto que este a cargo pueda entrar y modificar aspectos importantes del programa

1. Prueba 3:

Se revisa que la resolución del programa sea la adecuada y que el contenido cambie según el tamaño de la ventana a elegir.

**APP**

1. Prueba 1:

Se testea si al ingresar una huella, esta se registra y crea un nuevo campo en la tabla de la base de datos. Esto es una prueba tanto para saber si esta correctamente conectado el programa a la base de datos, como para comprobar si el lector detecta, registra y reconoce correctamente las huellas dactilares.

1. Prueba 2:

Se toman pruebas para ver si el programa puede reconocer a qué hora se ingresa la huella, y si las faltas que aplica este se acumulan para tener un registro para el fin de año