

# Proyecto Prueba

*Sesión 2*

*It Soluciones de Software Para Negocios*

**Enviado por:**

Yasser Pushaina Rojas  
Chief Expert Colombia

## Introducción

Este documento contiene las tareas especiales para esta sesión. El tiempo máximo para completar tantas tareas como sea posible es de 210 minutos (3 horas y media). Durante este tiempo, deberá leer y comprender todas las tareas y guardar los resultados, para comprender mejor tu trabajo y evaluarte de una forma óptima deberás crear un readme donde describas todas las indicaciones pertinentes de tu trabajo

## Contenido

Esta propuesta de Test Project consta de la siguiente documentación/archivos:

1. Sesión 2.pdf
2. Estructura de la aplicación Sesión 2.pdf
3. personas.csv
4. profesor.csv
5. personal.csv

## Descripción del proyecto y tareas

La tarea de esta sesión consiste en implementar los datos base para el software de administración de la Escuela de Habilidades TI de WorldSkills.

Todas las personas que tienen relación con la escuela se almacenan en el archivo persons.csv. Los datos que están presentes son:

- nombre, apellido
- fecha de nacimiento y ciudad de nacimiento
- género
- clase
- fecha de inicio
- contraseña

Otra tabla, teacher.csv, contiene los nombres de los profesores de la escuela y las clases en la cual son tutores. El nombre de todos los administradores y su función (por ejemplo, administrador) se presenta en el archivo staff.csv.

Se debe desarrollar una base de datos escolar para fines administrativos y una aplicación con interfaz de usuario. La aplicación debe poder importar datos de otras estructuras en formato "csv" automáticamente y corregirlos automáticamente, si es posible, o manualmente, si es necesario.

Se debe añadir una estructura con diferentes permisos de usuario. Se pueden registrar nuevos estudiantes o profesores, y los usuarios con permisos pueden añadir datos. Se deben implementar funciones de búsqueda. Además, se debe redactar una breve guía del usuario.

## Instrucciones para los Competidores

### 1.1 Estructura de la Aplicación

La aplicación consta de al menos 4 formularios diferentes (formulario de inicio de sesión, formulario de control, formulario de personas, formulario de clase) y además un subformulario/vista.

Cuando se inicia la aplicación, primero debe abrirse un formulario de inicio de sesión. Después de un inicio de sesión exitoso, debería mostrarse el formulario de control.

Aquí el usuario debe encontrar los mecanismos para controlar la aplicación y la conexión con los demás formularios. El acceso a los diferentes formularios depende de los permisos de los distintos tipos de usuario.

Los otros formularios son:

- Formulario de Personas
- Formulario de clase

En el formulario de clase, se muestran los estudiantes de una clase especial seleccionada. En el Formulario Personas, el usuario debe gestionar el trabajo con la base de datos.

Hay cuatro tipos de usuarios con diferentes privilegios en la base de datos, que sólo puede cambiar un administrador:

- administrador
- profesor
- personal
- estudiantes

Cada estudiante puede ver el formulario de su clase y su formulario personal (subformulario/vista), con sus datos personales. No puede editar ningún dato.

El profesor debería tener más privilegios. Todos los profesores pueden ver el Formulario de Clase, el Formulario de Personas con todos los datos, incluido el Formulario Personal de un estudiante seleccionado, y pueden editar sus datos personales.

El personal tiene acceso a todos los formularios. Pueden editar datos personales (excepto los tipos de usuario) y datos de la clase.

Un administrador tiene acceso a todos los formularios y puede cambiar todo tipo de datos y agregar o eliminar campos de la base de datos.

Tenga en cuenta que cada formulario debe tener un mecanismo para volver al formulario de control.

## 1.2 Diseño de la base de datos y carga

- Importar los datos de los archivos csv y crear una estructura de datos útil.
- Los campos necesarios son nombre de usuario, contraseña para los usuarios del programa y para las personas: nombre, apellido, fecha de nacimiento, clase, datos de dirección, habilidad, rama, fecha de inicio, fecha de finalización, tipo de usuario.
- Obtener la información sobre el tipo de usuario de los otros archivos. Todas las personas que no son Incluidos en los archivos teacher.csv o staff.csv son estudiantes.
- Es posible que necesite campos adicionales en el proceso de desarrollo de la base de datos.
- Utilice tipos de datos apropiados
- Cree claves primarias y claves foráneas apropiadas

## 1.3 Formulario de inicio de sesión

- Crear e formulario de inicio de sesión
- Al iniciar la aplicación, se mostrará primero este formulario.
- El usuario debe iniciar sesión con nombre de usuario y contraseña (el nombre de usuario para usuarios normales debe ser “prenom.surname”).
- Un usuario registrado debe poder cambiar su contraseña
- Solo los usuarios registrados pueden abrir la aplicación; no se aceptarán otros. Los nuevos usuarios recibirán un mensaje indicando que deben contactar al administrador para registrarse.
- Se debe implementar un mecanismo para contraseñas olvidadas.

## 1.4 Formulario de control

- Crear el formulario de control
- Crear un mecanismo de control que represente las relaciones con los otros formularios de una manera útil y comprensible
- Se deben implementar las conexiones a todos los demás formularios.
- Tenga en cuenta los privilegios de acceso de los diferentes tipos de usuarios!
- El formulario de control debe tener un mecanismo para salir del programa.

## 1.5 Formulario de Personas

- Crear el formulario Personas
- Tenga en cuenta los diferentes privilegios de usuario!
- Si solo se deben mostrar los datos de una persona (por ejemplo, en el caso de un usuario con privilegios de estudiante), utilice una subformulario o vista especial llamada Formulario personal.
- Implementar las siguientes funciones:
  - importación de datos desde archivos csv en la estructura dada
  - Verificación de errores en los datos importados o añadidos. Claro que no se podrían corregir los errores automáticamente. Sin embargo, se debería crear una forma inteligente de corregir manualmente los datos erróneos (por ejemplo, si un campo de datos está vacío o si la ciudad tiene más de un código postal, etc.). Todos estos errores deberían ser fáciles de detectar y corregir.
- edición de datos o campos especiales
- añadir datos o campos especiales

- buscar datos o campos especiales
- mostrar tabla de personas (los campos mostrados deben ser seleccionables)
- mostrar partes de la tabla de personas utilizando criterios de búsqueda (nombre, apellido, clase)
- al hacer clic en una clase seleccionada se abrirá el formulario de esta clase seleccionada

## 1.6 Formulario de Clase

***Class: Xy***

***Tutor: Mrs. Unknown***

<u>Nr.</u>	<u>ID</u>	<u>Surname</u>	<u>Prename</u>
1	2031-2-a-17	Doe	Jane
2	2019-25-a-18	Doe	John
...	...	...	...

- Crear el formulario de clase
- iTenga en cuenta los diferentes privilegios de usuario!
- Debe aparecer un campo activo llamado “clase” donde se pueda seleccionar una clase de la escuela.
- Después de una selección se deben mostrar todos los demás datos de la clase seleccionada.
- En un campo debe mostrarse el tutor de esta clase.
- Adicionalmente se debe mostrar una tabla con 4 columnas en la que se muestre un número consecutivo, el ID del estudiante, el apellido y el nombre de todos los estudiantes de la clase, ordenados alfabéticamente por su apellido.
- Al hacer clic en un estudiante, sus datos personales se mostrarán en la vista formulario personal.