



## TRABAJO FINAL INTEGRADOR SEGUNDO CUATRIMESTRE

En el trabajo integrador de este segundo cuatrimestre, se debe una situación problemática del TFI de Sistemas y Procesos de Negocios.

### CONFORMACIÓN DE GRUPOS

El trabajo es grupal y de hasta 4 personas (**de la misma comisión**).

### CONSULTAS

Las consultas se atenderán por el medio que el docente y/o auxiliar lo dispongan.

### PAUTAS DE ENTREGA

Las entregas asociadas se realizarán **exclusivamente** por medio del campus virtual de la materia; el archivo a subir debe tener el nombre:

***Comisión\_apellido1\_apellido2\_apellido3\_apellido4.pdf***

Comisión, seguida por los apellidos de los integrantes separados por guiones.

### DOCUMENTACIÓN POR ENTREGAR

Deberán **entregar a través de la tarea** el archivo pdf con el análisis completo de la situación problemática a resolver.



## Composición del Archivo PDF

### Situación Problemática:

Este módulo se debe explicar la situación problemática a resolver y los métodos de recolección de la información (entrevista, documentos, etc.) que se utilizaron en el TFI de la asignatura Sistemas y Proceso de Negocios.

### Estructura de la Información:

Determinar las **entidades** que se encuentran en la situación problemática elegida y realizar la estructura de la información de estas indicando solo:

- Nombre del archivo
- Estructura del registro

## I. INTERFAZ DE USUARIO

El diseño visual de la interfaz es libre y se pueden utilizar elementos gráficos (logos, colores) para hacer las pantallas de la aplicación más atractivas.

Al ingresar a la aplicación, se presenta el Menú Principal, donde el usuario debe tener una cuenta creada para poder iniciar una sesión. A continuación, se muestra el esquema de opciones que debe respetar el menú principal.

### MENÚ DE OPCIONES

- Crear archivo
- Grabar información
- Baja Lógica
- Baja Física
- Modificación de campos
- Listado de la información
- Consulta por alguno de los campos
- Crear una estructura dinámica a partir de los datos del archivo.
- Mostrar datos de la estructura dinámica
- Salir

## II. GESTIÓN Y AUTENTICACIÓN DE USUARIOS

En este ejemplo, el usuario tiene la posibilidad de crear un nuevo usuario y la opción de crear un nuevo asistente. Para esto el programa le solicitará nombre de usuario y contraseña, condiciones que deberán tener en cuenta al diseñar su interfaz de usuario.



**Nombre de usuario:** Quedará definido por una cantidad mínima de 6 caracteres y máxima de 10, los cuales podrán ser letras, números y/o símbolos del conjunto {+, -, /, \*, ?, !, ,}. Deberá cumplir con los siguientes requisitos:

- Ser único para cada usuario registrado.
- Comenzar con una letra minúscula.
- Tener al menos 2 letras mayúsculas.
- Tener como máximo 3 dígitos.

Ejemplos de nombres de usuario incorrectos: AbCl123 (no cumple con b), pTS!1234 (no cumple con d), gl78Mci (no cumple con c), mARtin123gomez (tiene más de 10 caracteres).

Ejemplos de nombres de usuario correctos: mARtin12, jo97!AR

**Contraseña:** Su conformación no podrá darse al azar, sino que deberá respetar lo siguiente:

- Deberá contener al menos una letra mayúscula, una letra minúscula y un número.
- No podrá contener ningún carácter de puntuación, ni acentos, ni espacios. Sólo caracteres alfanuméricos.
- Deberá tener entre 6 y 32 caracteres.
- No debe tener más de 3 caracteres numéricos consecutivos.
- No debe tener 2 caracteres consecutivos que refieran a letras alfabéticamente consecutivas (ascendentemente). Este criterio es válido tanto para letras mayúsculas, minúsculas o combinación de ambas.

Ejemplos de contraseñas mal formadas: Ach32 (no cumple con c), dorit1234 (no cumple la regla a ni la d), sA;gotAP.10 (no cumple con b), aBuel123 (no cumple con e).

Ejemplo de contraseñas bien formadas: Achus32, 125Af89, aLejo123, DORItO45, 4aC2sa.

**Nota:** Tanto en el nombre de usuario como en la contraseña deben distinguirse mayúsculas y minúsculas.

### III. PERSISTENCIA

Para no perder la información entre ejecuciones, es necesario mantener los datos de las cuentas de usuario registradas. Para cumplir con este objetivo deberá implementarse la persistencia de la información manejando archivos que se describirán más adelante en este apartado. La estrategia sugerida para cumplir con este objetivo es:

- Cuando se inicia la aplicación, se cargan los datos en estructuras de datos en memoria.
- Durante la ejecución de la aplicación se actualizan estas estructuras.



- Al terminar cada sesión se agrega el contenido de esta al archivo correspondiente.

Los datos de las cuentas de usuario se almacenarán en el archivo **Usuarios.dat**

Este archivo será de formato binario y guardarán variables diseñadas en una estructura de datos a fin de representar la información asociada a las cuentas de usuarios (es decir, nombre y contraseña). Cada vez que se inicie la aplicación, se utilizará la información almacenada en este archivo para validar el inicio de sesión y/o la creación de un nuevo usuario.

La estructura de la información para los Usuarios debe ser la siguiente:

<b>Usuarios</b>	
Usuario	Char [10]
Contraseña	Char [10]
Apellido y Nombre	Char [60]