



# GESTIÓN DE DONACIONES DE ALIMENTOS MEDIANTE BASE DE DATOS

Informe Final – Proyecto final de Carrera AIA

Facultad de Ingeniería y Ciencias Hídricas

Universidad Nacional del Litoral

20/06/2020

Lucas Ríos

## Contenido

I.	Justificación .....	0
II.	Alcance del sistema .....	1
III.	Modelo del sistema .....	2
IV.	Diseño de Funcionalidades.....	3
V.	Modelado de la BD .....	4
VI.	Modelado de Interfaz de usuario.....	7
VII.	Prueba de BD.....	10
VIII.	Pruebas de Interfaz .....	11
IX.	Pruebas Finales.....	12
A.	Ventana Inserción de Negocio .....	12
B.	Ventana Modificación de un Negocio .....	14
C.	Ventana de Crear nuevo Usuario .....	15
D.	Ventana Ingresar Nueva Donación.....	18

**Palabras clave—Base de Datos, Aplicación, Automatización, Optimización, Seguridad.**

#### I. JUSTIFICACIÓN

**E**l Banco de Alimentos Santa Fe, ubicado en la capital de la provincia, es una organización sin fines de lucro que se dedica al rescate de alimentos, de manera trazable y segura, aptos para el consumo que por distintos motivos no poseen valor comercial, con el fin de facilitarlos a diferentes ONG, comedores, y asociaciones encargadas de brindar alimentos a personas de bajos recursos económicos. Este rescate de alimentos se realiza mediante visitas programadas al Mercado de Alimentos de la ciudad, donde se visitan los distintos puestos en busca de donaciones de aquellos alimentos que por distintos motivos ya no se consideran a la venta por parte de los comerciantes.

Hoy en día, estas donaciones son registradas en un formulario impreso por distintos grupos (de hasta 4 integrantes cada grupo) de voluntarios del BASFE que visitan el mercado de manera y presencial y recorren cada uno de los puestos en busca de alimentos que cumplan las condiciones anteriormente dichas. Una vez finalizada la visita, estas donaciones son cargadas a una planilla de cálculo por uno de estos voluntarios al cual se le asigna dicha responsabilidad, el cual cambia constantemente. Cada una de estas donaciones es registrada en la planilla junto con la información necesaria (tipo producto, nombre de negocio, etc.). Sin embargo, debido a que el responsable de realizar la carga no siempre es el mismo, pueden presentar distintas costumbres al momento de escribir, o puede considerar que no es necesario completar un campo o utilizar distintas nomenclaturas. Cada una de estas variables genera mayores inconsistencias en la planilla cuando esta va creciendo a lo largo del tiempo.

A partir de esta, se elabora manualmente un resumen de datos. Este resumen consiste en un conjunto de tablas de valores obtenidos mediante la implementación de funciones estadísticas, las cuales buscan, representar el comportamiento y características de las distintas donaciones en distintos periodos y bajo distintas condiciones. Estas funciones abarcan:

- Total de donaciones diarias por producto
- Peso total diario obtenido
- Peso total de donaciones por Negocio semestral y anual
- Promedio de peso total por mes

Estos resultados son utilizados por los directivos para conocer cómo se comportan los distintos negocios con respecto a las donaciones en distintos periodos del año, donde, por ejemplo, pueden saber en qué mes del año se suele obtener una mayor cantidad de donaciones con respecto al resto, o cual es el negocio que más donaciones registra, entre otros.

A su vez se realizan distintas gráficas comparativas que sirven como resúmenes periódicos de los datos registrados (total por visita, por mes, etc.) así como informes en épocas puntuales del año que sirven como resumen general (mayores donantes, mayores productos obtenidos, etc.).

Todas estas tareas demandan mucho tiempo y trabajo humano para llevarse a cabo cuando podrían realizarse fácilmente de manera automática y minimizando el riesgo de error. Al momento de ingresar los datos, estos no son verificados de ninguna forma, tanto como para asegurarse que no se ingrese una letra donde iría un número u otra letra como para asegurar que una entrada esté dentro de un conjunto de valores permitidos.

Mediante este sistema propuesto se busca llevar a cabo una automatización de las tareas de manera que estas ya no tengan

que ser llevadas a cabo en su mayor parte por el operador, disminuyendo así las responsabilidades y, a su vez, las posibilidades de cualquier clase de error humano cometido dentro de los datos manejados.

Los usuarios de este sistema serán, por un lado, aquellos voluntarios responsables de realizar la carga de donaciones en la planilla en cada momento. Al momento de ingresar una nueva donación, esta será validada por la base de datos de manera que se cumplan todos los requisitos necesarios. Por el otro lado, los miembros de la Comisión Directiva son los encargados de la obtención de los resúmenes para su posterior análisis. Estos resúmenes son presentados como un conjunto de tablas que muestran el resultado de distintas funciones:

- Cantidad de donaciones en un día
- Peso de cada producto en un día
- Peso semestral donado por un negocio
- Peso anual donado por un negocio
- Promedio de donaciones por mes

Estas tablas son analizadas por la comisión para llevar a cabo las decisiones pertinentes.

Esto no solo permitirá que no haya errores en la realización de estos, sino que también liberaría de responsabilidad de muchas tareas a los operarios encargados, que podrán dedicar ese tiempo a otras responsabilidades. A su vez, la utilización de este serviría de manera indirecta a la comisión directiva, ya que estos dispondrán de un conjunto de resultados más rápido, más confiables, automatizado y con posibilidad de ampliación sin mucha complejidad debido a que la modularidad con la que se desarrollaran las tareas en la base permite que cualquier función añadida futura requerida por la institución pueda añadirse de una manera sencilla.

## II. ALCANCE DEL SISTEMA

El funcionamiento esencial del sistema se basa en la creación de una base de datos representativa de la planilla utilizada actualmente. Esta se utilizará tanto para el almacenamiento de la información como para el tratamiento de esta. Aquí se almacenará información representativa de las donaciones de alimentos a la institución para cada día de manera individual, permitiendo además la posterior edición de la misma:

- Donaciones
  - Fecha
  - Peso

Debido a que cada donación involucra mucha información relacionada (producto, donado, voluntarios, etc.), Cada uno de estos datos pueden ingresarse (y modificarse) de manera individual para luego ser utilizado al ingresar la nueva donación:

- Voluntarios
  - Nombre
  - DNI
- Dueños
  - Nombre
- Ubicaciones
  - Número
  - Tipo de Ubicación
  - Dueño
- Donantes
  - Nombre del negocio
  - Información de Contacto
- Productos
  - Nombre

Todas las donaciones, junto con su respectiva información será mostrada en una tabla, la cual podrá ser exportada a una planilla de cálculo tradicional.

Para llevar a cabo tanto las operaciones de administración como las de comprobación dentro de la base de datos se hará uso de funciones en lenguaje PostgreSQL. Estas funciones también serán utilizadas para

llevar a cabo las operaciones para obtener el resumen:

- Peso total de donaciones de un mismo producto
- Kilogramos totales obtenidos por visita
- Kilogramos totales obtenidos por cada donante
- Promedio de Kilogramos mensuales obtenidos

De ser necesario, se pueden agregar fácilmente nuevas funciones que se encarguen de realizar nuevas operaciones o informes.

Parte de los objetivos de la implementación de este sistema es la disminución de la interacción directa del voluntario responsable con la BD. Para lograr esto se hará uso del desarrollo de una aplicación que actuará como interfaz entre ella y el usuario. La comunicación entre ambos se llevará mediante llamados a las funciones declaradas previamente. De esta manera la aplicación llevará a cabo la verificación de sintaxis de la información ingresada, mientras que la base de datos se encargará de llevar a cabo la validación de las inserciones en cada tabla mediante el uso de *checks*. Debido a que la aplicación será operada por distintos tipos de personas, se hará que la carga de datos sea lo más simple posible, buscando reducir las acciones necesarias:

- La fecha de la visita se generará automáticamente al momento de comenzar la carga, permitiendo editarla en caso de ser necesario
- Los campos que admitan únicamente una opción entre ciertos valores fijos se mostrarán en una lista para seleccionarse
- Un puesto siempre tiene el mismo ocupante, por lo que los campos se autocompletarán

Además, se dispondrá del acceso a una sección donde se mostrará, de manera gráfica, los resultados de los resúmenes previamente declarados que serán generados de manera automática.

Por último, para considerar los distintos usuarios del sistema, se complementarán los voluntarios con una contraseña, lo cual nos permitirá ingresarlos y posteriormente modificarlos, a los cuales se les permitirá distintas funcionalidades:

- Administradores: serán aquellos usuarios que tendrán acceso a todas las funcionalidades del sistema, pudiendo administrar el ingreso y modificación de cada tipo de dato (Productos, Negocios, Etc.), inclusive el ingreso y modificación de donaciones, administrar el ingreso de nuevos usuarios o modificar los datos de los actuales y por último, visualizar los resúmenes generados.
- Visitantes: son usuarios con accesos restringidos que únicamente tienen permitido realizar la inserción de donaciones dentro de la BD.

Debido a que toda la información pertinente al programa será almacenada dentro de la BD, permitirá únicamente el ingreso controlado de datos, de manera que la estructura de esta no podrá verse modificada de ninguna manera desde la aplicación.

### III. MODELO DEL SISTEMA

Para llevar a cabo una representación del proceso de desarrollo se hará uso del modelo cascada. Mediante este modelo representaremos el ciclo de vida completo del sistema, organizando las actividades a realizar y como se hará uso de estas para las sucesivas etapas. Se eligió este modelo debido a que las pruebas serán llevadas a cabo una vez que el sistema este diseñado y

en un estado avanzado de desarrollo. Habrá dos etapas de prueba:

- Verificar la correcta creación de la Base de datos y las funciones que se encargan del manejo de esta.
- Una segunda prueba que verificará el funcionamiento del sistema integrado a la interfaz de usuario, que corresponderá a la prueba final.

Además, se agregaron conexiones de retorno de una etapa a la anterior para representar la vuelta a una actividad anterior en caso de ser necesario (debido a un posible requerimiento agregado o cambio en el proyecto):

visualizar a los distintos usuarios y cada una de las funcionalidades ofrecidas por el sistema.

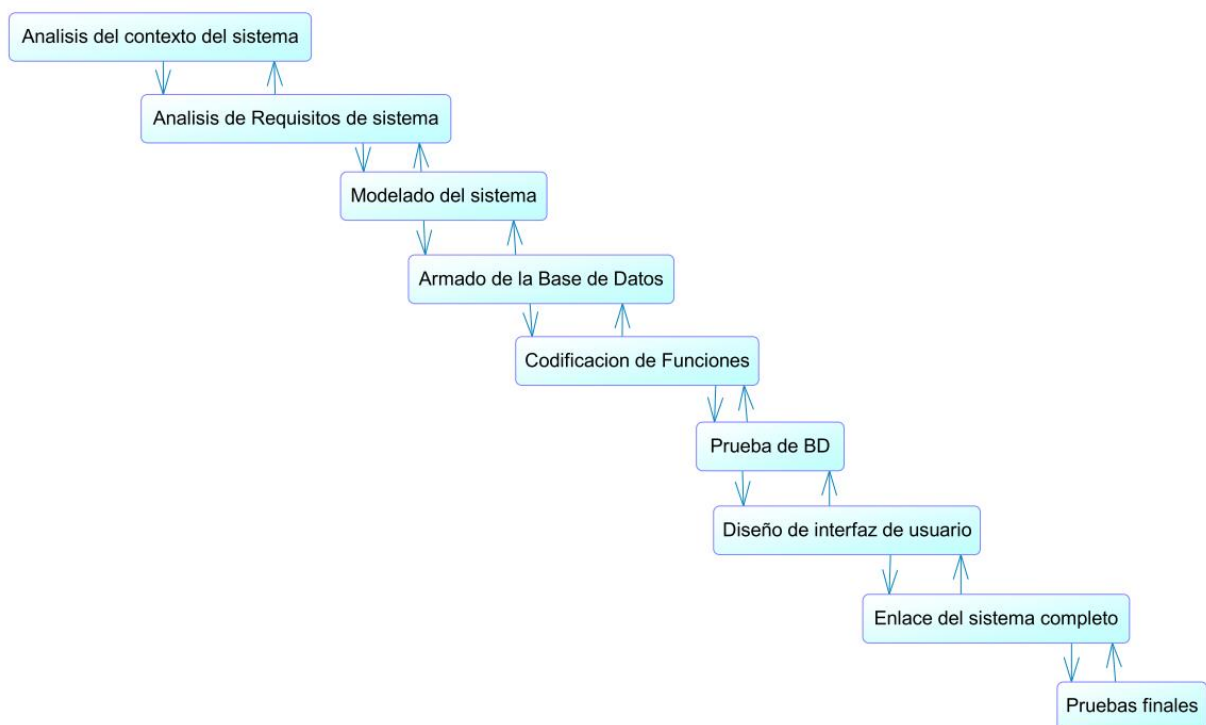
En nuestro sistema encontramos dos tipos de roles (o actores), los cuales se repartirán la disposición de funcionalidades provistas por este:

- Comisión Directiva
- Voluntarios Responsables

En el siguiente diagrama podemos visualizar cada una de las funciones llevadas a cabo por cada rol, donde la Comisión (CD) utiliza el programa para dos funciones principales:

- Por un lado, obtener información acerca del rendimiento de las

#### Modelo de Proceso de Actividades



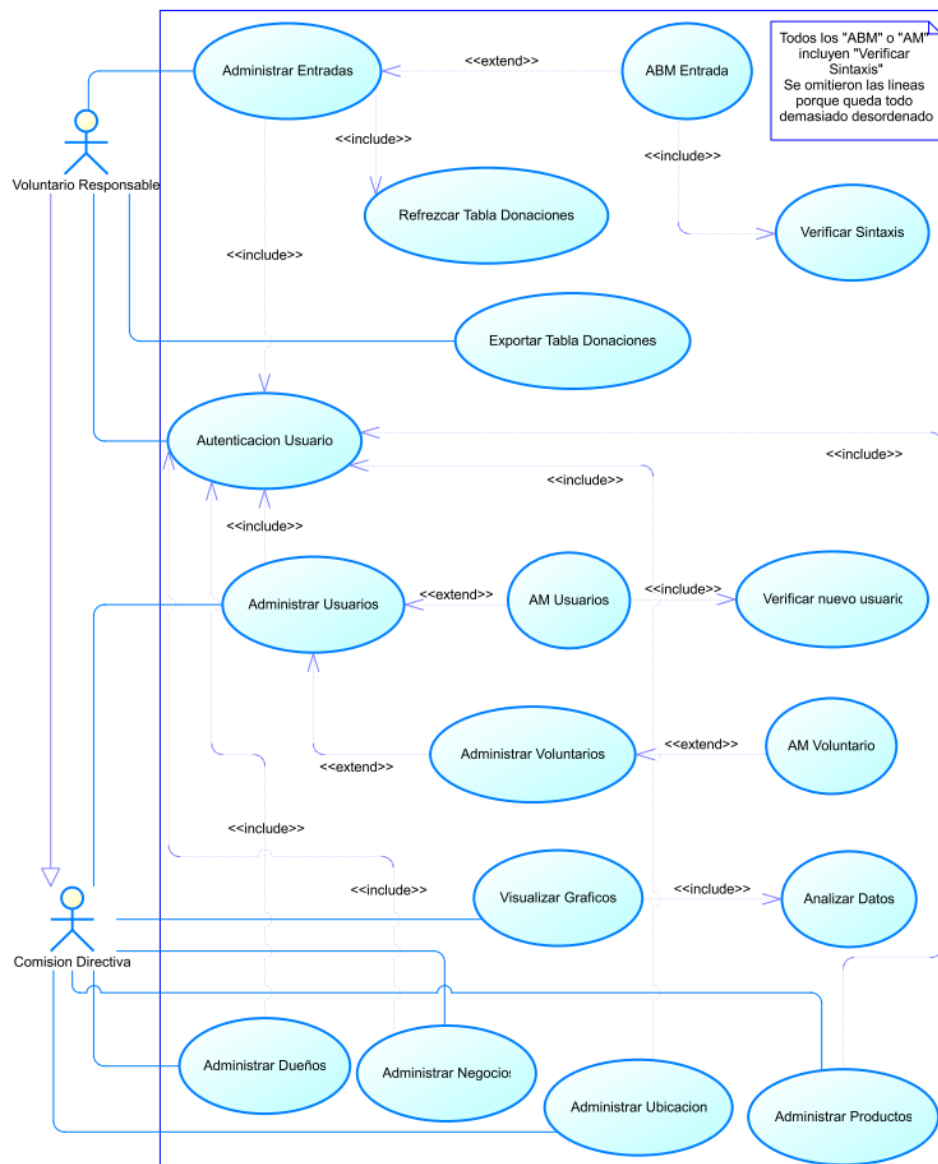
#### IV. DISEÑO DE FUNCIONALIDADES

Antes de comenzar con el desarrollo del sistema, debemos modelar cuáles serán las funcionalidades propias de este y quien estará a cargo de cada una de ellas. Para ello se hará uso del siguiente Modelo de Casos de Uso mediante el cual podremos

donaciones mediante el uso de distintos resúmenes que muestran sus resultados de manera grafica

- Eliminar cualquier donación ingresada
- Por ultimo, dispondrá de la opción de dar de alta o modificar:
  - Dueños

Diagrama de  
Casos de uso



- Productos
- Negocios
- Ubicaciones
- Usuarios

Estos datos, estarán disponibles para seleccionarse al momento de registrar una nueva donación mediante listas desplegables.

Por otro lado, los responsables de las donaciones únicamente se encargan de llevar a cabo la inserción de donaciones, proceso que es completado mediante la selección de aquellos datos que fueron cargados previamente por la CD, pudiendo además modificar cualquiera de estas

entradas. Todos estos cambios se ven plasmados constantemente en la tabla principal de donaciones, por lo que esta va siendo actualizada a medida que se lleven a cabo.

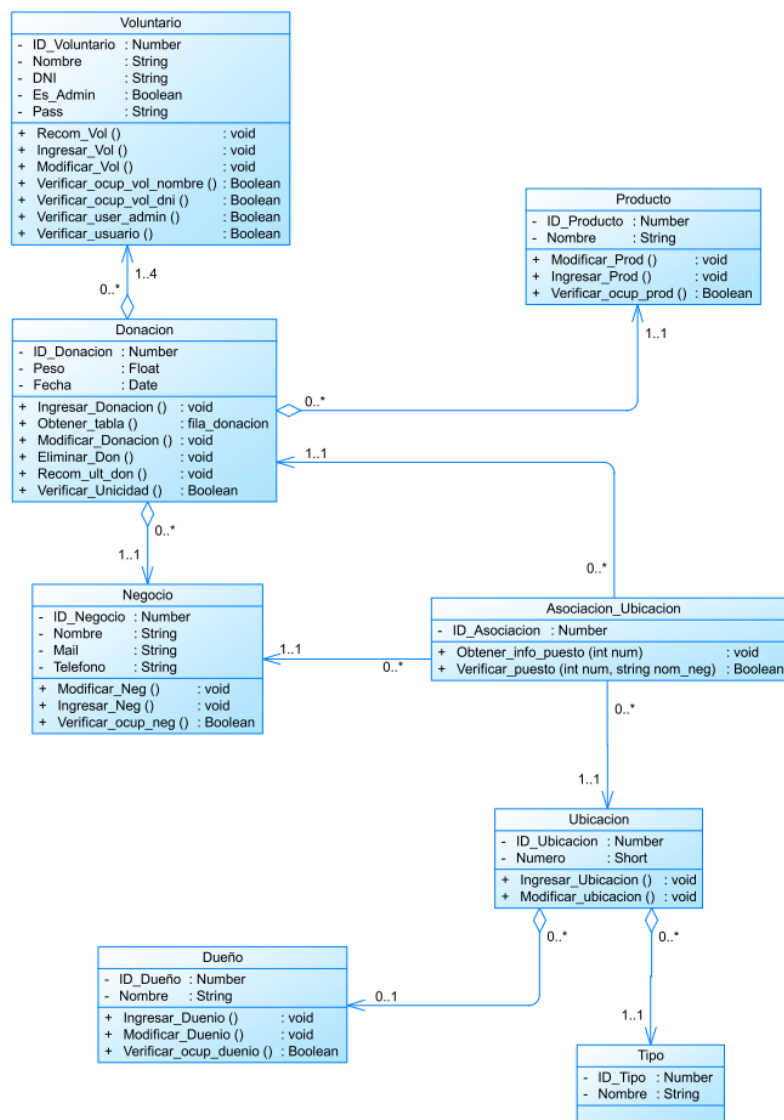
## V. MODELADO DE LA BD

Para comenzar con el desarrollo del sistema, habiendo ya determinado el alcance y objetivos de este, se comienza con el desarrollo de ciertos modelos que nos permiten atacar la complejidad desde distintas perspectivas. Estos modelos son los que se utilizarán como base para la codificación tanto de la Base de datos (BD) como de la aplicación de interfaz de usuario.

Siguiendo nuestro modelo de desarrollo, el primer paso es la creación de la BD, por lo que el primer modelo que se mostrara es el Diagrama de clases (DC). Este modelo nos permite visualizar las distintas tablas que conformaran la BD, además de las cardinalidades con las que cada elemento de la tabla se relaciona con el resto. Otro detalle importante, es la visualización de las funciones en las que interviene cada una de las tablas. Estas serán de suma importancia ya que serán el medio esencial de comunicación entre la BD y la interfaz de usuario. Mediante el modelo visualizado en la imagen vemos que las tablas que relacionan todo el esquema son *Donación* y *Asociación-Ubicación*, de las cuales se

desprenden las relaciones hacia las tablas restantes. Otro dato importante son aquellos parámetros que actúan como *claves de negocio* únicas dentro de ambas tablas, siendo la concatenación <Fecha-Producto-Negocio> en caso de *Donación* y <Negocio-Donación-Ubicación> en el caso de *Asociación-Ubicación*, valores que nos permiten identificar una única fila si no dispusiésemos del ID correspondiente. Este modelo se creó con la idea de poder obtener a través de él la tabla de datos original provista por los voluntarios en la planilla de datos, la cual se visualizará en la ventana principal del programa.

Diagrama de Clases





En la siguiente imagen podemos ver la representación del sistema. Aquí podemos ver distintos detalles sobre como este se conforma:

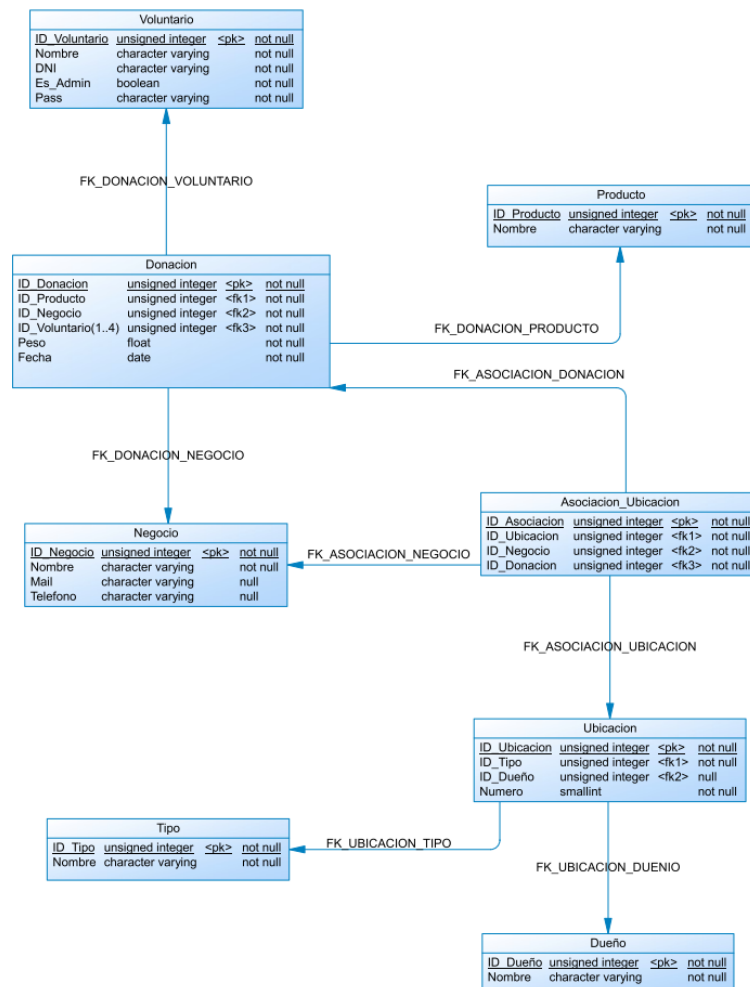
- Cada donación dispone de entre 1 y 4 voluntarios, un único producto, un único negocio y es cargada al sistema por un usuario
- Una ubicación tiene un dueño (opcional) y es de un único tipo
- La tabla Asociación Ubicación se encarga de realizar una asociación entre una donación y la ubicación donde esta se realizó. Esto nos permite que una ubicación pueda pertenecer a dos negocios distintos en distintos momentos de tiempo (lo cual sucede en caso de las ubicaciones de tipo Piso)

Como último modelo, tenemos el Modelo físico de datos (MFD). Este modelo no presenta grandes diferencias con respecto al DC, sin embargo, es un modelo más abstracto que representa cómo se modela la estructura lógica de una BD, permitiéndonos saber cómo los datos se conectan entre sí y cómo se procesan y almacenan dentro del sistema. Por todo esto, es mayormente utilizado para llevar a cabo la codificación de la BD.

En este modelo vemos principalmente como los IDs de cada tabla se integran en el resto de las tablas permitiendo identificar claramente las relaciones entre ellas y como estas están conformadas:

- Al momento de ingresar una nueva donación, debe seleccionarse un producto, de 1 a 4 voluntarios y un negocio

Modelo Físico de Datos



- Ahora podemos ver, como dijimos anteriormente, que la tabla Asociación Ubicación integra una referencia tanto de una donación, como de un negocio y una ubicación, permitiendo la repetición de las ubicaciones con distintos negocios, pero asegurando la unicidad de la combinación *Donación-Ubicación-Negocio*

## VI. MODELADO DE INTERFAZ DE USUARIO

Debido a que los usuarios del sistema no tienen interacción alguna con la BD, para poder operar con esta, se llevará a cabo una interacción con una aplicación que funcionará como interfaz.

En la imagen 1 podemos ver la ventana principal del programa. Aquí se mostrará en grande una tabla que funciona como visualizador de todas las donaciones ingresadas ordenadas en orden de ingreso. A la izquierda se encuentran todos los botones de acción para llevar a cabo las operaciones sobre las donaciones. Estos se encuentran deshabilitados por defecto cuando no se registraron donaciones y se habilitarán una vez que se realice algún ingreso. El calendario en la esquina inferior izquierda se utiliza para seleccionar el día

con el que se registraran las nuevas donaciones (que por defecto selecciona el día del sistema). Por último, el menú superior alberga el resto de las funciones, las cuales estarán habilitadas o no dependiendo el tipo de usuario que haya iniciado sesión:

- Agregar o Modificar usuarios
- Exportar la tabla
- Agregar o modificar productos, dueños, negocios o ubicaciones.
- Eliminar entradas de donaciones

En la imagen 2 se muestra la ventana donde se registra una nueva donación. En esta ventana la gran mayoría de los campos son listas desplegables, las cuales permiten al usuario elegir entre los valores disponibles que son cargados directamente de la base de datos al momento de abrir dicha ventana, de manera que se elimina cualquier posibilidad de error de tipeo, facilitando el completado. El único campo que permite el ingreso de texto presenta funciones de control que se encarga de asegurarse que no se ingresen caracteres incorrectos en él. Esta ventana presenta varias funciones de autocompletado para facilitar la experiencia de uso:



1-Ventana Principal

**Ingresar Nueva Donación**

Nombre de los Voluntarios: (\*) Voluntario (Sin Completar), Voluntario 2 (Sin Completar), Voluntario 3 (Sin Completar), Voluntario 4 (Sin Completar)

(\*) Nombre del Negocio: circe

(\*) Nombre del Producto: [dropdown]

(\*) Peso (en Kg.): [input field]

(\*) ¿Puesto o Piso?   
☐ Puesto   
☒ Piso

(\*) Número de la Ubicación: [dropdown]

Aceptar Cancelar

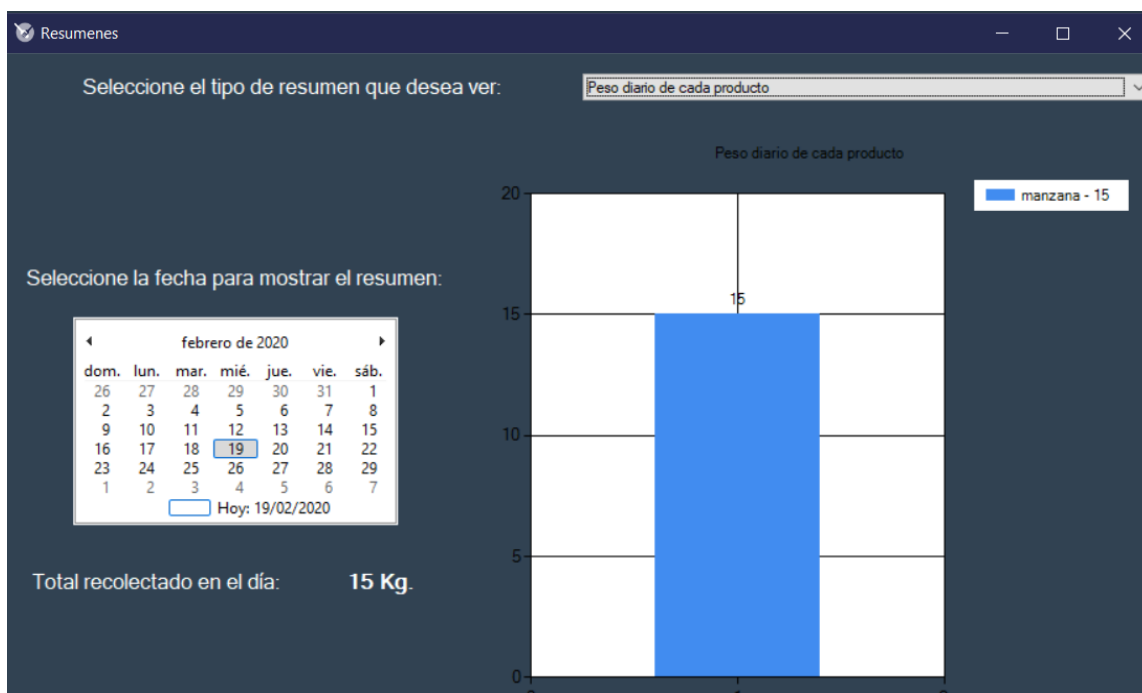
2- Ventana de ingreso de Donación

- Al abrir la ventana se seleccionará automáticamente los voluntarios que participaron de la última donación registrada
- Además, se seleccionará el ultimo negocio utilizado

Estas dos funciones se basan en el funcionamiento de las actuales inserciones de datos, donde la lista de elementos a ingresar viene ordenada por grupo de voluntarios y por negocio, de manera que existen altas probabilidades que dicho grupo registre varias donaciones de dicho negocio consecutivamente, de manera que

se logra ahorrar varios campos a completar, agilizando la inserción.

Por último, se dispone de una función que, al seleccionar una ubicación de tipo Puesto ingresada previamente, seleccionará automáticamente el negocio de dicho puesto ya que, como se mencionó antes, los puestos mantienen un mismo negocio siempre. Debido a esto, también se dispone también de una función que se asegura que no se ingrese en un puesto ya utilizado un nuevo negocio.



3- Ventana de resumen

4- Ventana de agregar Ubicación

En la imagen 3 vemos la ventana de resúmenes. Aquí se permite ver de manera gráfica los resultados de los resúmenes, los cuales pueden elegirse mediante un menú desplegable. Según la elección en este menú y la fecha indicada en el calendario, se muestra el resultado mediante la gráfica como se ve en la imagen. Debajo del calendario se muestra, además, el peso total recolectado el día seleccionado, sin importar el resumen visualizado en la gráfica, debido a la importancia y frecuencia de utilización de dicha información.

En la imagen 4 se muestra un ejemplo de una de las ventanas de agregación de datos, siendo este el caso de *Agregar Ubicación*. Todas las ventanas presentan los campos necesarios para el agregado, siendo estos controlados para que no puedan ingresarse caracteres no admitidos o mayúsculas no deseadas. Además, las ventanas de agregar disponen de una columna a la derecha que muestra los elementos ingresados hasta ese momento. En este caso se muestran los números de ubicaciones junto con el tipo de cada una. Las ventanas de modificar son similares, presentando además una lista que permite seleccionar, de los elementos ya ingresados, cual modificar, y luego se autocompletan los campos restantes con la

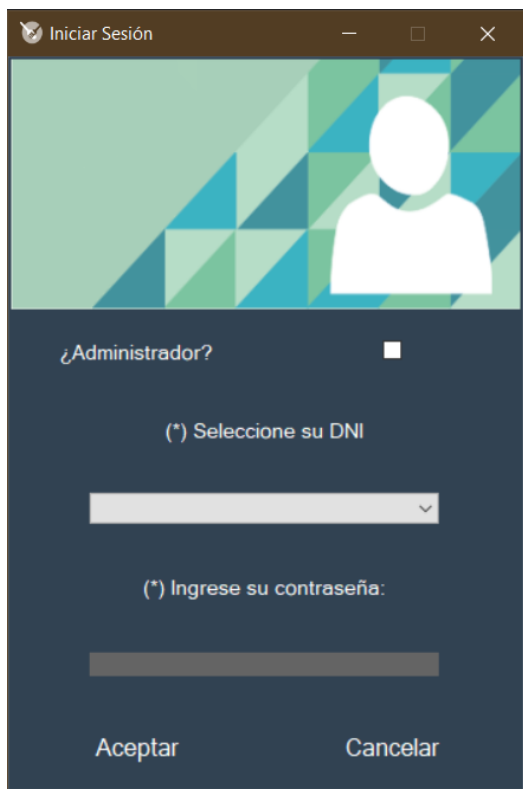
información actual, permitiendo su modificación.

En la imagen 5 se muestra la ventana de Agregar un nuevo usuario. Esta ventana está habilitada únicamente desde un usuario administrador, ya que permite agregar otros usuarios. Aquí se deja un espacio para ingresar el Nombre, DNI y un

5 - Ventana de Nuevo Usuario

campo check administrador que permite indicar si el nuevo usuario es administrador. Por último tenemos un campo contraseña obligatorio para todo tipo de usuario. Esta ventana aparece al iniciar el programa por primera vez con la simple modificación que el check administrador esta activado y no puede modificarse, esto con el fin de obligar al usuario a crear un usuario administrador la primera vez que se inicia el programa.

Por último en la imagen 6, se muestra la ventana que aparece al iniciar el programa. Esta presenta el check administrador al comienzo, de manera que según esté este activado o no, se filtra la lista de DNI mostrada abajo, donde se muestran los invitados si no está activada y los administradores si lo está.



6 - Ventana de Iniciar Sesión

## VII. PRUEBA DE BD

Para llevar a cabo el testeo de la BD se realizaron pruebas simples de cada función, donde se utilizaron datos manualmente ingresados que ayudasen a verificar cada uno de los controles de la función.

- Por un lado, se verifica que la funcionalidad actúe de manera adecuada en un caso normal sin ningún tipo de complicaciones (se ingresan todos los valores obligatorios y opcionales)
- Por el otro, se verifica que no haya errores ante el ingreso de datos nulos o incorrectos

Un ejemplo de esta verificación es ingresar una nueva *Ubicación*, la cual dispone de un número, un tipo y un dueño. De estos datos, los valores obligatorios son el *número* y *tipo*, por lo que la función no se ejecuta si no recibe alguno de ellos. Además, al igual que el resto de las funciones de inserción, la función distingue si es el primer dato insertado en esa tabla o no, de manera que no ocurran problemas en ningún caso (problema que se encontró en versiones tempranas de la BD). Sin embargo, uno de los requerimientos establecidos indicaba que las ubicaciones de tipo *Puesto* no pueden repetirse con negocios distintos, detalle que la función de inserción no contempla, de manera que una misma ubicación de tipo *puesto* puede estar ocupada por dos negocios distintos. Debido a esto fue necesario crear otra función encargada de verificar la ocupación de un puesto, la cual debe invocarse antes.

Estas pruebas se llevaron a cabo para cada función integrada en la BD de manera individual, de manera que cualquier error detectado (como el indicado arriba) era corregido durante el testeo. El motivo principal de estas pruebas es asegurarnos sobre el correcto funcionamiento básico de la función. De esta manera, una vez que realizamos el llamado desde la interfaz de usuario, podemos concentrarnos en pulir errores puntuales en vez de tener que verificar que esta cumpla con su función correctamente, ya que, de ser ese el caso, la detección de errores sería más costosa, debido a que tenemos que buscar si dicho error se dio por un mal funcionamiento de

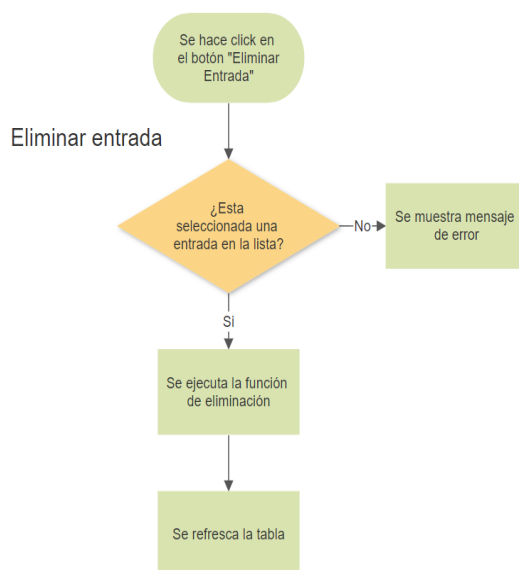
la función, o por algún error al momento en que esta fue invocada por la interfaz.

#### VIII. PRUEBAS DE INTERFAZ

A pesar de que estas pruebas aseguran el funcionamiento individual de cada funcionalidad, muchas de estas se desligan de ciertas verificaciones para que otras se encarguen de ello, de manera que dependen del resultado para un correcto funcionamiento en su totalidad:

- Asegurarse que los campos numéricos contengan únicamente números
- Validación previa de nuevas ubicaciones como la que se comentó previamente
- Validación para evitar valores nulos para ciertos campos
- Verificación de disponibilidad de nombre de usuario

Estas nuevas validaciones son realizadas en la etapa de testeo de interfaz de usuario, donde cada ventana invoca de manera secuencial las distintas funcionalidades, permitiendo o bloqueando la ejecución de estas dependiendo el resultado de las validaciones anteriores.



Este ejemplo es un diagrama de flujo del funcionamiento del botón eliminar entrada. Aquí la ventana se asegura que haya

seleccionada una fila de la tabla (y, por ende, que haya alguna fila en la tabla) antes de llamar a la función eliminación, de esta forma, la función no debe encargarse de verificar esto, ya que la ventana es la que se ocupa. Antes de la etapa de testing, esta verificación no estaba, de manera que se generaba un error al querer eliminar una entrada en la lista vacía (posteriormente se realizó una nueva verificación que deshabilita el botón cuando la tabla queda vacía).

Se realizó una entrevista con los futuros usuarios del sistema, la cual sirvió como un testeo directo por parte de ellos, donde probaron todas las funcionalidades de este:

- Se hicieron sucesivas pruebas de inserción de datos, donde se probaron ingresar valores nulos en distintos campos (obligatorios o no), pesos decimales (separados por puntos, comas etc.) y números de ubicaciones decimales (solo se permite ingresar números enteros)
- El ingreso de letras en campos numéricos
- El funcionamiento de las funciones de autocompletado en caso de selección de puestos
- El entendimiento sobre algunas decisiones de interfaz, como el funcionamiento del calendario en la ventana principal
- Se comprobó la utilidad de la ventana de resumen

Todas las pruebas realizadas sirvieron como retroalimentación para la realización o modificación de varias funciones dentro del sistema:

- Permitir insertar de manera individual productos, usuarios, ubicaciones, etc.
- Rediseñar la ventana de resumen para que muestre más información útil

- Mejorar ciertas restricciones en los campos de texto para evitar errores
- Eliminación de múltiples campos de texto para ser reemplazados por listas seleccionables, de manera que se evita el ingreso de datos, restringiendo únicamente a ciertos valores disponibles

#### IX. PRUEBAS FINALES

Para finalizar el desarrollo del sistema, se llevará a cabo una sucesión de pruebas destinadas a verificar el comportamiento de los distintos apartados del programa mediante diferentes comprobaciones.

Para cada una de las ventanas que serán probadas, nos enfocaremos en simular acciones que pueden ocurrir de manera cotidiana y que si no son tenidas en cuenta, podrían ocasionar una falla en la ejecución del sistema:

- Dejar campos de texto en blanco
- Escribir ciertos caracteres donde no deberíamos (ingresar letras en un campo de números por ej.)
- Utilizar indistintamente mayúsculas y minúsculas
- Verificación de funciones de autocompletado

Realizamos este tipo de pruebas debido a que anteriormente ya validamos que estas funcionen con los datos adecuados, por ende ahora tenemos que enfocarnos en ver cómo se comportan cuando se ingresan valores erróneos o inesperados (si un campo obligatorio está vacío, la función no debería continuar con su ejecución, por ej.). Normalmente se suele decir que una prueba es exitosa cuando presenta errores, de manera que estos puedan ser arreglados en vez de ser descubiertos una vez que el producto es entregado, por lo que todas las verificaciones que se enlistarán más abajo se espera que cubran todas aquellas situaciones que un usuario puede llegar a afrontar debido a un descuido (dejar un

campo en blanco), costumbre propias de escritura (usar “.” o “,” para notación decimal), entre otros.

Debido a que mostrar los resultados de cada una de las funcionalidades haría muy extensa esta sección, se seleccionaron 4 principales que involucran gran parte de la lógica del sistema:

- Inserción de un negocio: todas las ventanas de inserción manejan una lógica idéntica, por lo que comprobando una podemos replicar el proceso para las demás. Se escogió el negocio ya que contiene tanto campos obligatorios como opcionales.
- Modificación de un negocio: siguiendo la lógica comentada anteriormente, lo mismo sucede con las ventanas de modificación.
- Creación de usuario: Al momento de crear un usuario se llevan a cabo distintas validaciones, de acuerdo con el manejo de usuarios, por lo que se escogió esta en vez de la función de “Iniciar sesión”
- Insertar donación: la función clave del sistema que mantiene en mayor o menor medida una relación con el resto de las funcionalidades

##### A. Ventana Inserción de Negocio

Para llevar a cabo el testeo de esta primera ventana se escogieron 5 pruebas que se encargan de ejecutar todas las verificaciones de las que dispone la ventana:

1. Se dejan todos los campos vacíos
2. Se completan todos los campos de manera correcta
3. Se completan únicamente los campos obligatorios y se escribe el nombre intercambiando entre mayúsculas y minúsculas
4. Se utiliza un nombre ya utilizado pero escrito en mayúsculas

## Resultados:

1)



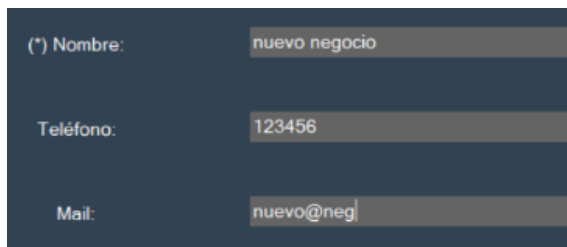
Error

Debe completar el campo 'Nombre' antes de agregar un nuevo negocio

Aceptar

Aquí nos encargamos de comprobar satisfactoriamente que el programa no nos permite avanzar sin completar los campos obligatorios dentro de la BD

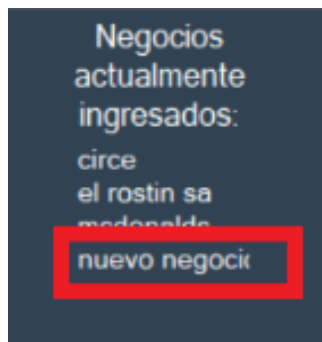
2)



(\*) Nombre: nuevo negocio

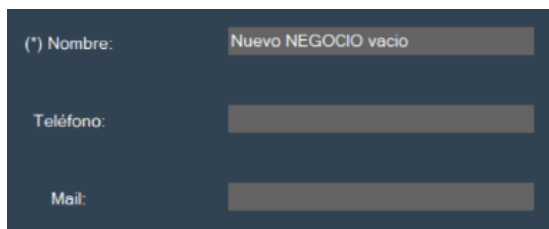
Teléfono: 123456

Mail: nuevo@neg



En la primera imagen vemos los campos ingresados y en la segunda vemos el resultado. Señalado en un recuadro rojo podemos ver que aparece en la lista de negocios ingresados el *nuevo negocio*

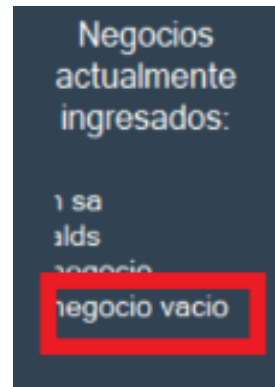
3)



(\*) Nombre: Nuevo NEGOCIO vacio

Teléfono:

Mail:



En la primera imagen vemos los campos ingresados y en la segunda vemos el resultado. Por un lado, vemos que el negocio fue ingresado sin problemas, por otro lado, vemos que este fue ingresado correctamente ya que aparece en la lista marcado con un recuadro rojo, donde además vemos que el nombre fue registrado en minúsculas correctamente

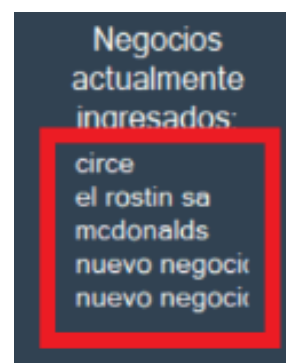
4)



(\*) Nombre: CIRCE

Teléfono:

Mail:



En la primera imagen vemos que se ingresa el nombre de un negocio ya ingresado anteriormente, pero escrito en mayúsculas. En este caso no se mostró una ventana que marque error alguno, sin embargo, vemos que no se produjo cambio alguno en la lista de negocios, sin importar las mayúsculas, *circe* aparece una única vez en la BD.



### B. Ventana Modificación de un Negocio

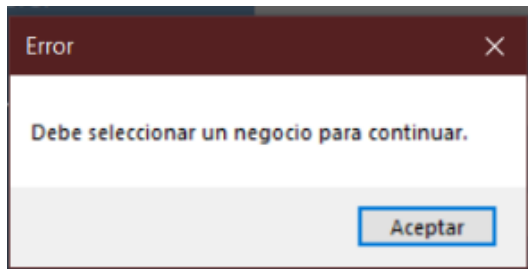
Para esta segunda ventana se eligieron 5 pruebas:

1. Se dejan todos los campos vacíos
2. Se selecciona un negocio y se pulsa *Aceptar*
3. Habiendo seleccionado un negocio, se cambia el nombre por uno preexistente pero escrito en mayúsculas
4. Habiendo seleccionado un negocio, se borra el nombre y se lo deja vacío
5. Se completan todos los campos correctamente y se vuelven a completar eliminando una entrada opcional

#### Resultados:

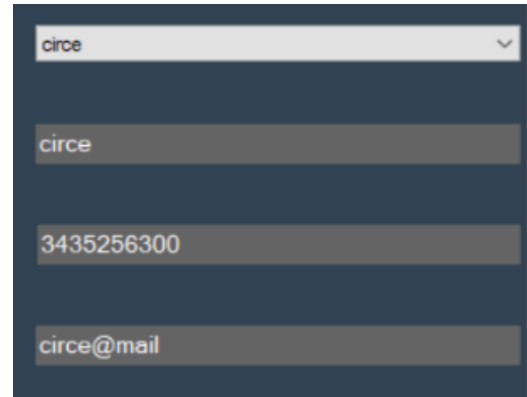
##### 1)

En esta primera prueba fue encontrado un error, ya que al no seleccionar nada, el programa se cerraba porque generaba un error al momento de ejecutar la función. Una vez solucionado esto, se vuelve a ejecutar la ventana:



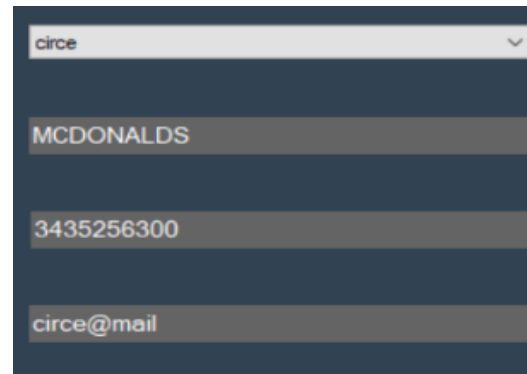
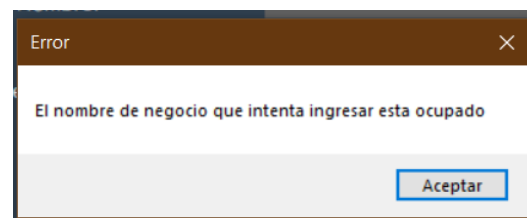
Aquí vemos que, si no se selecciona nada, se solicita que mínimamente se seleccione un elemento a modificar. Caso contrario puede cerrar la ventana mediante el botón cancelar.

##### 2)

A screenshot of a web form for modifying a business. At the top, a dropdown menu shows "circe" with a downward arrow. Below it are four input fields: the first contains "circe", the second contains "3435256300", and the third contains "circe@mail". The fourth field is empty.

Al seleccionar un elemento en la lista, vemos que se autocompletan los campos restantes de manera satisfactoria. Si pulsamos aceptar la ventana se cierra sin registrar cambio alguno.

##### 3)

A screenshot of the same web form as in the previous image, but the dropdown menu now shows "MCDONALDS". The other input fields remain the same: "MCDONALDS", "3435256300", and "circe@mail".

Una vez que seleccionamos un negocio y se completaron los datos, podemos modificar cualquiera de estos. Si modificamos el nombre del negocio y escogemos uno ya utilizado (aunque este escrito en mayúsculas), podemos ver que se muestra una ventana de error indicando que este está siendo utilizado.

##### 4)

Formulario de edición de negocio. El dropdown muestra 'mcdonalds'. El campo de teléfono contiene '123456'.

Formulario de edición de negocio. El dropdown muestra 'nuevo circe'. Los campos de nombre y teléfono contienen 'nuevo circe' y '155256300' respectivamente.

Un cuadro de diálogo de error con el título 'Error' y el mensaje 'El nombre del negocio no puede quedar vacío'. Hay un botón 'Aceptar'.

Formulario de edición de negocio. El dropdown muestra 'nuevo circe'. Los campos de nombre y teléfono contienen 'nuevo circe' y 'nuevocirce@mail' respectivamente.

En esta prueba se encontró otro error, por el cual no se verificaba que los campos obligatorios queden completos al pulsar aceptar, de manera que se podía dejar vacío el nombre de un negocio. Una vez solucionado esto vemos el resultado de las imágenes, donde se muestra un cuadro de error al querer confirmar los cambios.

Por último, realizamos una modificación completa. En la primera imagen vemos los datos previos y en la segunda vemos los datos luego de confirmar los cambios. Si vemos el recuadro rojo, dicha lista muestra como el nombre del negocio fue cambiado exitosamente en la BD y debajo, muestra los campos actuales. Además, en la tercera imagen, vemos que realizamos otra modificación donde dejamos el campo teléfono en blanco, cambio que también se realizó exitosamente.

5)

Formulario de edición de negocio. El dropdown muestra 'circe'. Los campos de nombre y teléfono contienen 'circe' y '3435256300' respectivamente.

### C. Ventana de Crear nuevo Usuario

Para esta ventana se establecieron las siguientes 6 pruebas:

1. Se dejan los campos en blanco y se pulsa *Aceptar*
2. Se intenta agregar un usuario que ya este ingresado
3. Se intenta agregar un usuario sin ingresar una contraseña
4. Se ingresan caracteres que no son números en el campo de DNI
5. Se ingresa un DNI con una cantidad de caracteres distinta de 8

6. Se agrega un nuevo usuario correctamente

**Resultados:**

1)

The screenshot shows a web form titled 'Ingresar nuevo usuario'. It has a header with a geometric pattern and a white user icon. Below the header, there is a label '(\*) Ingrese su Nombre:' followed by an empty text input field. Below this, there is a label '(\*) Ingrese su contraseña:' followed by an empty password input field. At the bottom, there are two buttons: 'Agregar' and 'Cancelar'. An error dialog box is open in the center, with the title 'Error' and the message 'Debe completar todos los campos obligatorios'. There is an 'Aceptar' button in the dialog box.

Aquí vemos que se verifican los campos obligatorios al querer agregar un usuario y no se permite continuar debido a que dichos campos se encuentran vacíos.

2)

The screenshot shows a DNI input field with the label 'dni' and a placeholder 'character varying'. The input field contains the number '39258748'. Below the input field, there is an error dialog box with the title 'Error' and the message 'El DNI ingresado ya esta dentro de la base de datos. Por favor, elija otro'. There is an 'Aceptar' button in the dialog box.

Aquí se intentó usar como DNI el que aparece en la imagen anterior, el cual está actualmente ingresado en la BD. Como podemos ver, el programa no nos permite

continuar a menos que ingresemos uno nuevo, lo que es un resultado correcto.

3)

The screenshot shows the 'Ingresar nuevo usuario' form. The 'Nombre' field is filled with 'Santiago' and the '¿Administrador?' checkbox is checked. The 'Contraseña' field is filled with '20554830'. At the bottom, there are 'Agregar' and 'Cancelar' buttons. An error dialog box is open in the center, with the title 'Error' and the message 'Un usuario debe tener una contraseña'. There is an 'Aceptar' button in the dialog box.

Aquí vemos que si se quiere agregar un nuevo usuario, el programa nos obliga a ingresar una contraseña para poder continuar.

4)

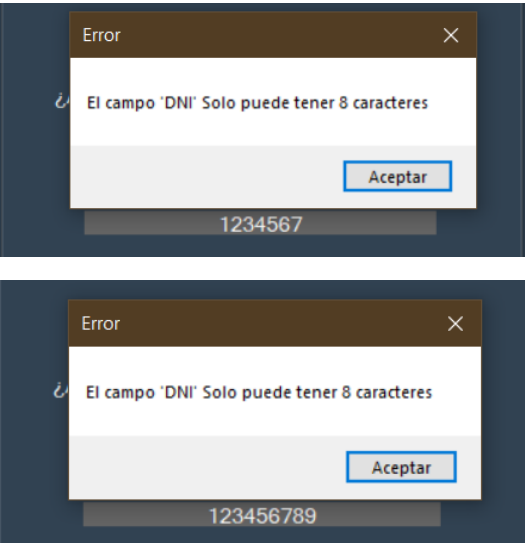
The screenshot shows the 'Ingresar nuevo usuario' form. The 'Nombre' field is filled with 'Santiago', the '¿Administrador?' checkbox is checked, and the 'DNI' field is filled with '1'. The 'Contraseña' field is empty. At the bottom, there are 'Agregar' and 'Cancelar' buttons. No error message is present.

En la imagen vemos la ventana con el campo de texto “DNI” enfocado luego de pulsar las siguientes teclas:

- .
- \*
- Q
- 1
- |
- Espacio
- /

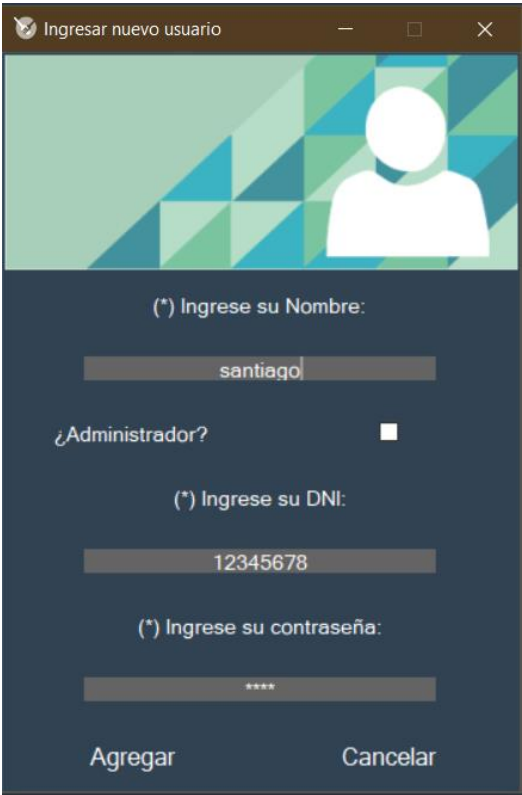
Como podemos ver, la única tecla que hizo efecto fue “1”, lo cual se corresponde con la salida esperada, ya que solo se permite ingresar números. Cabe notar que siquiera el espacio tuvo efecto, ya que el cursor se muestra exactamente enseguida al “1”, por lo que la salida fue correcta.

5)



Aquí vemos que se intentaron ingresar tanto 7 como 9 caracteres en el campo y ninguno fue permitido, especificando la cantidad exacta permitida.

6)



nombre	dni	es_admin
character varying	character varying	boolean
santiago	12345678	false
hugo	87654321	true

En la primer imagen podemos ver la ventana al ingresar un usuario invitado con DNI: “12345678” y en la segunda imagen la misma ventana, pero ingresando un usuario administrador con DNI: “87654321”. En la última imagen vemos que ambos usuarios se almacenaron correctamente en la BD.

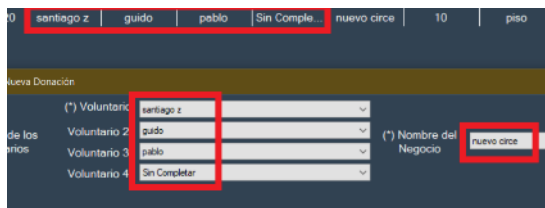
#### D. Ventana Ingresar Nueva Donación

Esta será la última ventana, la cual presenta distintas funciones de verificación, las cuales serán probadas por las siguientes pruebas:

1. Se dejan todos los campos vacíos
2. Se seleccionan voluntarios exceptuando el primero que queda vacío
3. Se intentan ingresar letras y símbolos en el campo *Peso*
4. Se prueba ingresar una donación con peso “. #” o “#.”
5. Se selecciona un puesto ya utilizado y se escoge un negocio distinto
6. Se ingresa una nueva donación correctamente

#### Resultados:

Antes de realizar las pruebas, se muestra una imagen de la ventana al abrirse:



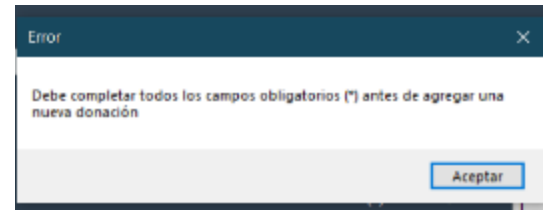
Aquí vemos una de las funcionalidades de autocompletado, donde se seleccionan automáticamente los voluntarios utilizados en la última donación ingresada y el negocio utilizado.

1)



Aquí vemos que, si pulsamos en aceptar sin completar los campos obligatorios, el sistema no nos deja continuar.

2)



En esta imagen, vemos que, a pesar de que todos los campos están completos, si el primer voluntario no se selecciona, el programa considera que falta completar datos obligatorios, por lo que no nos deja continuar.

3)



En la imagen vemos la ventana con el campo de texto “Peso” enfocado luego de pulsar las siguientes teclas:

- .
- .
- \*
- Q
- 1
- |
- Espacio
- /

Aquí vemos que solo se permitió ingresar un único punto y la tecla “1”. Esta salida es correcta, ya que el campo se encarga de verificar que solo se ingresen caracteres numéricos y además permite ingresar únicamente un solo punto, que funciona como separador decimal.

4)

pera	0,5
naranja	5

Aquí se muestra el resultado luego de ingresar dos nuevas donaciones. La primera se ingresó en el campo *Peso* “.5” en vez de 0.5, mientras que en la segunda se ingresó “5.” En vez de 5. Esta prueba se realizó para verificar un correcto funcionamiento de la función que ingresa la donación ante distintas formas de escribir los pesos.

5)

Negocio	Numero	¿Puesto o Piso?
mcdonalds	18	puesto

En esta imagen vemos que ya hay registrada una donación donde la ubicación #18 de tipo Puesto está ocupada por el negocio “mcdonalds”. De esta manera el sistema no debería permitir que ingresemos una nueva donación para dicha ubicación, pero ocupada por otro negocio.



Aquí vemos que la salida es la esperada. El sistema no nos permite seleccionar otro negocio para dicha ubicación.

6)

Fecha	Voluntario 1	Voluntario 2	Voluntario 3	Voluntario 4	Negocio
13/3/2020	santiago z	guido	pablo	Sin Comple...	mcdonalds
13/3/2020	santiago z	guido	pablo	Sin Comple...	nuevo circe
20/3/2020	santiago z	guido	pablo	Sin Comple...	nuevo circe
20/3/2020	santiago z	guido	pablo	Sin Comple...	nuevo circe
20/3/2020	santiago z	Sin Comple...	guido	Sin Comple...	el rostin sa

Numero	¿Puesto o Piso?	Telefono	Mail	Dueño	Producto	Peso
18	puesto	123456	Sin Comple...	Sin Comple...	naranja	19
10	piso	Sin Comple...	nuevocirce...	dueño c	manzana	17,5
1	puesto	Sin Comple...	nuevocirce...	dueño a	pera	0,5
1	puesto	Sin Comple...	nuevocirce...	dueño a	naranja	5
10	piso	Sin Comple...	Sin Comple...	dueño c	pera	15

En esta última prueba se verificarán, por un lado, que a diferencia de una ubicación de tipo *Puesto*, podamos ingresar una donación en una ubicación de tipo *Piso* ya utilizada. Por otro lado, verificaremos que la donación sea ingresada correctamente.

Los datos a ingresar serán:

- Voluntario 1: “Santiago Z”
- Voluntario 2: “Sin Completar”
- Voluntario 3: “Guido”
- Voluntario 4: “Sin Completar”
- Producto: “Pera”
- Peso: 15
- Negocio: “El rostin sa”
- Ubicación: 10 de tipo Piso (ya ingresada en otra donación)

En la ultima imagen podemos ver en la fila seleccionada, la nueva donacion ingresada. Por un lado, los datos seleccionados se ingresaron correctamente, ademas de no presentar problemas con la ubicación seleccionada.