



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

xx/03/2020

Lucas Rios

Contenido

I.	Justificación	0
II.	Alcance del sistema	0
III.	Modelo del sistema	1
IV.	Diseño de Funcionalidades.....	3
V.	Modelado de la BD	4
VI.	Modelado de Interfaz de usuario	7
VII.	Modificación de funcionalidades	10
VIII.	Prueba de BD.....	10
IX.	Pruebas de Interfaz.....	11
X.	Pruebas Finales.....	12
A.	Ventana Inserción de Negocio	12
B.	Ventana Modificación de un Negocio	13
C.	Ventana de Crear nuevo Usuario	15
D.	Ventana Ingresar Nueva Donación.....	18

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

I. JUSTIFICACIÓN

El manejo de donaciones actual del Banco de Alimentos de Santa Fe se sostiene sobre el uso de planillas de cálculo que son manejadas manualmente por un voluntario responsable, el cual cambia constantemente. El resto de los voluntarios se separan en grupos de hasta 4 personas que visitan el Mercado de Alimentos de Santa Fe, donde recorren los distintos puestos en busca de donaciones, las cuales son registradas en una planilla física. Estas planillas son enviadas al voluntario responsable, el cual carga los registros en la planilla de cálculo. Cada una de estas donaciones es registrada en el sistema junto con la información necesaria (producto, fecha de vencimiento etc.), pudiendo haber entradas incompletas en ciertos casos. A partir de estas entradas se elabora un resumen donde se engloban resultados como el total de productos del mismo tipo donado, entre otras cosas. Este resumen se realiza manualmente mediante funciones propias de la planilla. 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. A su vez, como se dijo previamente, el resumen se lleva a cabo manualmente por una única persona, de

manera que puede olvidarse u omitirse un dato, además, debido a que esta cambia constantemente, decisiones de diseño tomadas por uno de los responsables pueden verse modificadas por otro sin notificación alguna.

Mediante este sistema propuesto se busca llevar a cabo una automatización de las tareas de manera que estas se desliguen en su mayor parte del 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 aquellos voluntarios responsables de la carga de la planilla en cada momento, siendo estos mismos los que se encargarán de la obtención de aquellos resúmenes que se obtengan de manera manual. Al momento de ingresar uno nuevo, estos serán validados por la base de datos de manera que se cumplan todos los requisitos necesarios. Además, el resumen se realizará de manera automática y periódica, donde todos los cálculos serán llevados a cabo por la propia base. 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, seguro, 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:

- Grupo de voluntarios
- ¿Visitan un Puesto o un Piso?
 - Número
- Información del donante
 - Nombre del negocio
 - Dueño
 - Información de Contacto
- Información del producto
 - Nombre
 - Peso
 - Fecha de vencimiento (en caso de ser aplicable)

Para llevar a cabo las operaciones de la base de datos se hará uso de funciones en lenguaje PostgreSQL. Esto con el fin de evitar que las consultas sean realizadas manualmente. Las funciones son 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

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 base. 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 encargara 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 mostraran en una lista para seleccionarse
- Un piso siempre tiene el mismo ocupante, por lo que los campos se autocompletarán
- Dependiendo el ocupante, se mostrarán una lista de sugerencias de productos basados en las donaciones anteriores de dicho ocupante

Además, como se generarán informes automáticos en periodos establecidos, también se dispondrá de la opción de generar algunos de los informes previamente declarados cuando sea necesario o de establecer el periodo en que deben llevarse a cabo.

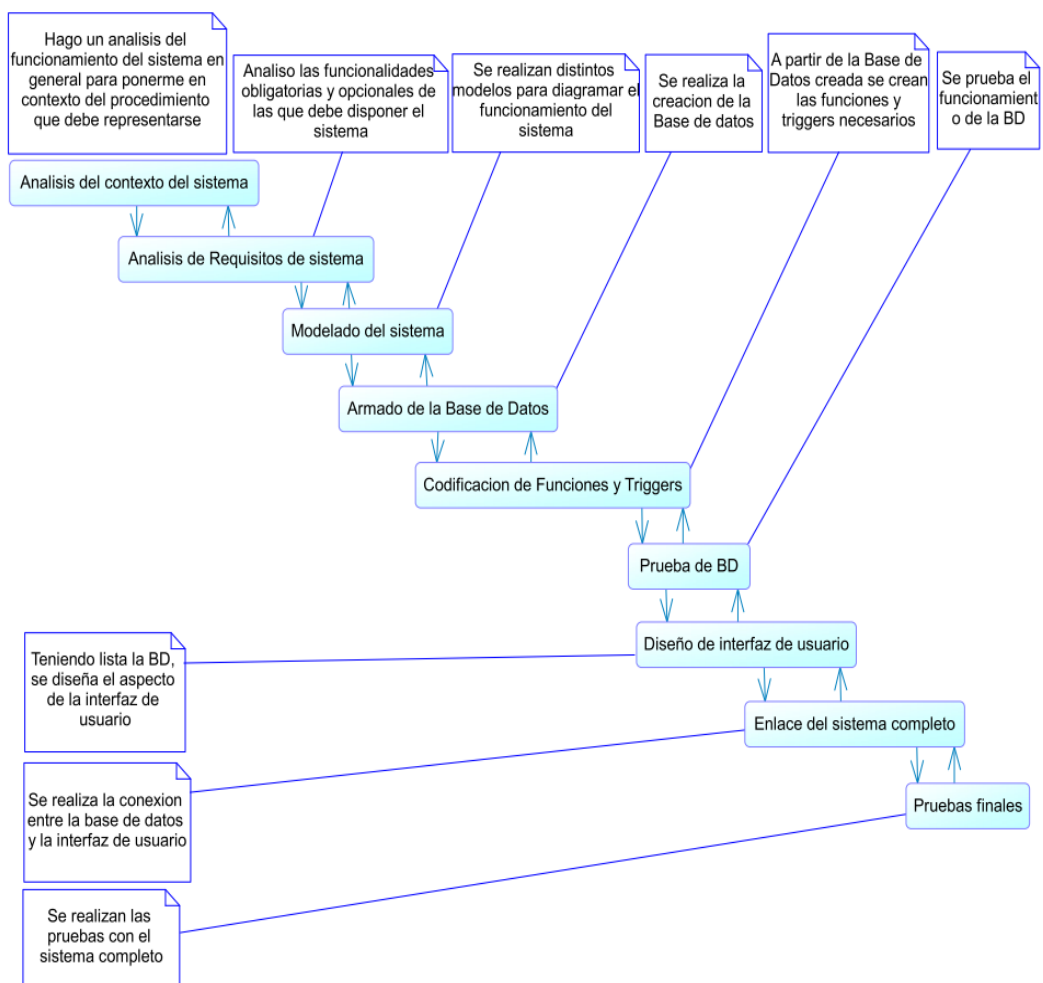
III. MODELO DEL SISTEMA

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):

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 visualizar a los distintos usuarios y cada una de las funcionalidades ofrecidas por el sistema. Debido a que estos no tienen interacción alguna con la BD, para poder operar con esta, se llevara a cabo una interacción con una aplicación que operara como interfaz.

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

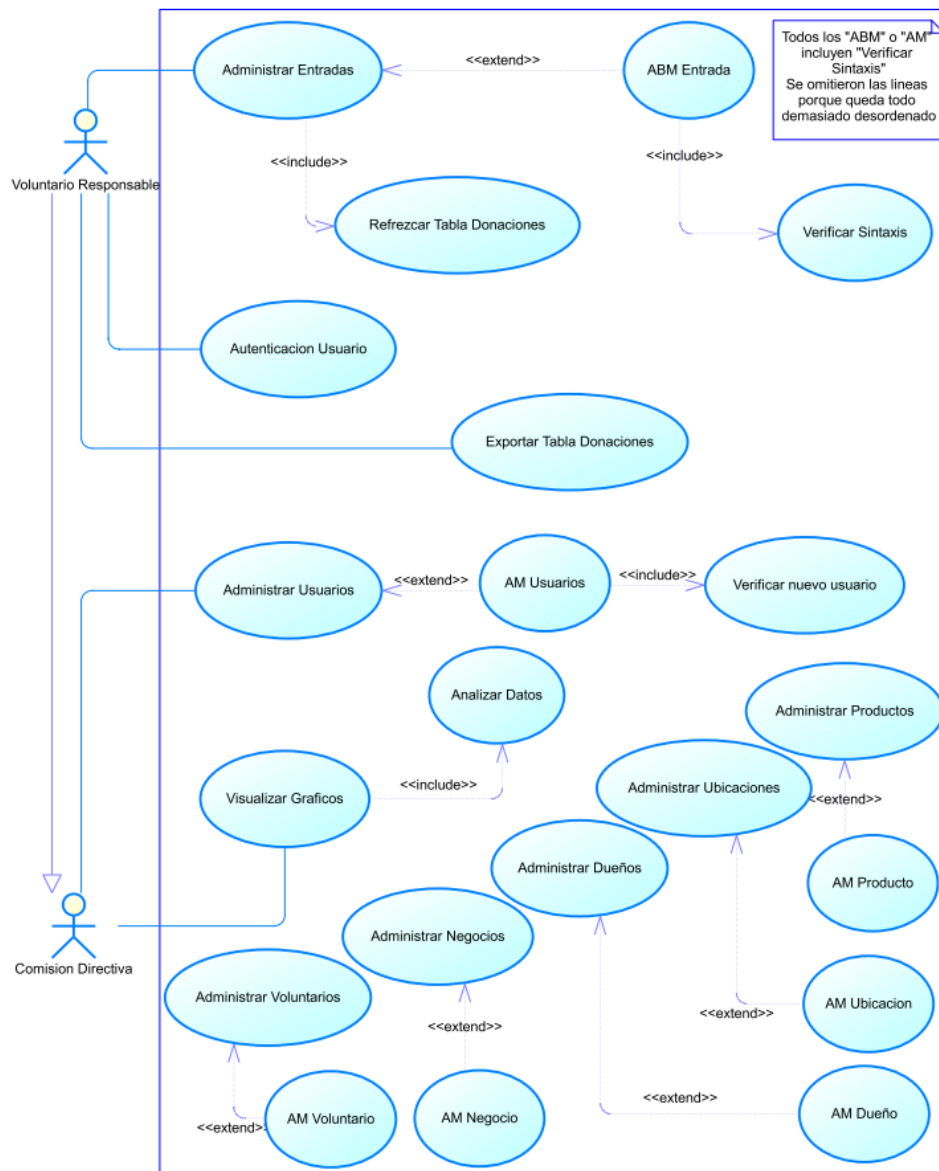
En el siguiente diagrama podemos visualizar cada una de las funciones llevadas a cabo por cada rol, donde la Comisión (CD) utiliza el programa para dos funciones principales:

- Por un lado, obtener información acerca del rendimiento de las donaciones mediante el uso de distintos resúmenes que muestran sus resultados de manera grafica
- Por otro lado, dispondrá de la opción de dar de alta o modificar:
 - Voluntarios
 - Dueños
 - 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 o eliminar 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. Por último, vemos que pueden realizar una exportación de datos, de manera que tengan disponible la tabla de donaciones en una planilla de cálculo clásica. Cabe aclarar que la CD hereda las funcionalidades disponibles para los responsables, pudiendo también llevarlas a cabo.

Diagrama de
Casos de uso



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.

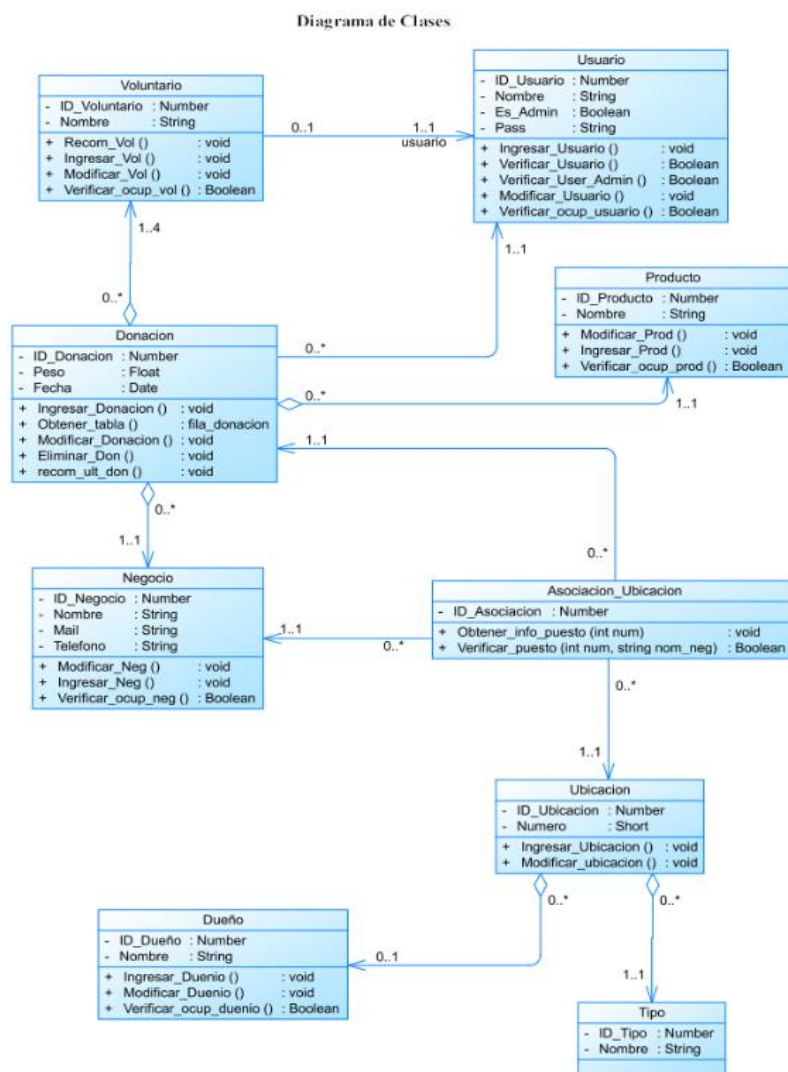
Siguiendo nuestro modelo de desarrollo, el primer paso es la creación de la BD, por lo que el primer modelo que se mostrara es el

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.

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
- Un Voluntario siempre es un usuario (un visitante) pero no viceversa (en caso de los administradores)
- 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 segundo 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.

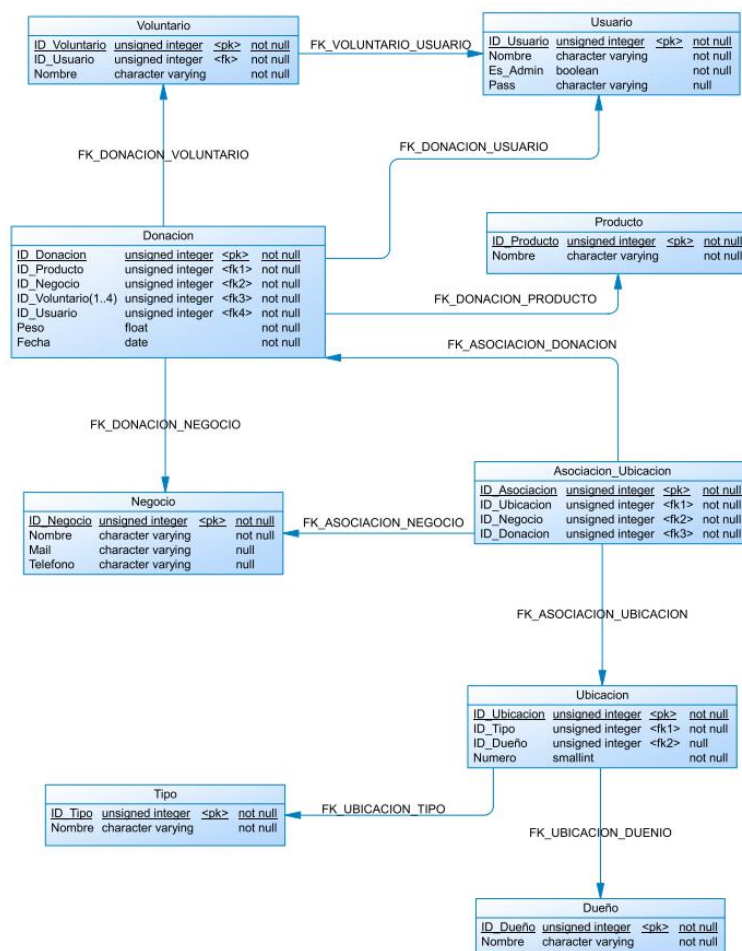
En este modelo vemos principalmente como los IDs de cada tabla se integran en el resto de las tablas permitiendo identificar claramente las relaciones entre ellas y como estas están conformadas:

- Al crear un voluntario, debe existir un usuario al que este se refiera
- Al momento de ingresar una nueva donación, debe seleccionarse un

producto, un usuario, de 1 a 4 voluntarios y un negocio

- 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*

Modelo Físico de Datos





1-Ventana Principal

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, voluntarios, negocios o ubicaciones.

2- Ventana de ingreso de Donación

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:

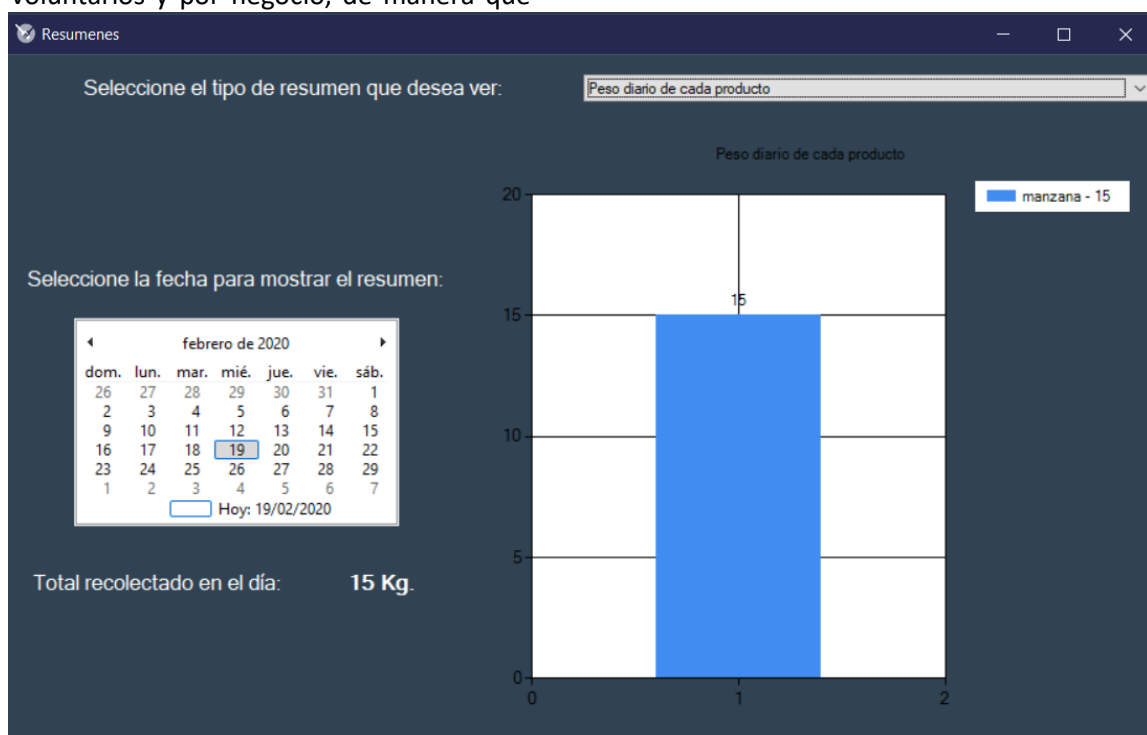
- Al abrir la ventana se seleccionará automáticamente los voluntarios que participaron de la última donación registrada
- Además, se seleccionará el ultimo negocio utilizado

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.

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



3- Ventana de resumen

4- Ventana de agregar Ubicación

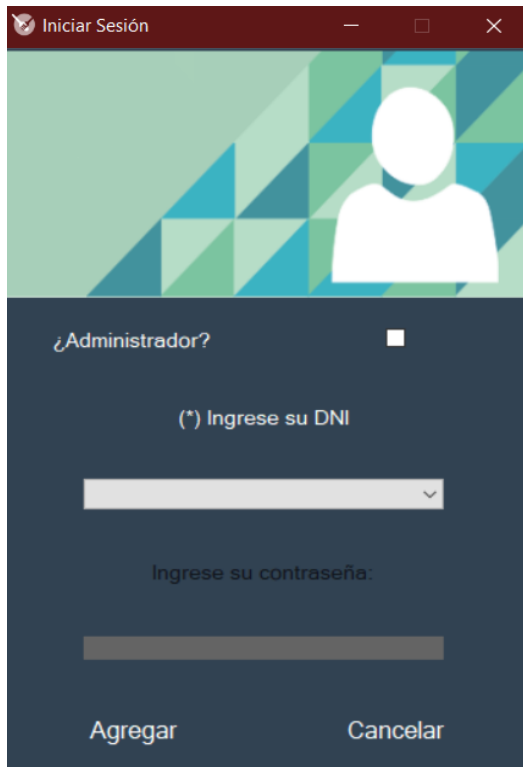
gráfica, debido a la importancia y frecuencia de utilización de dicha información.

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.

5 - Ventana de Nuevo Usuario

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 administradores. Aquí se deja un espacio para ingresar el DNI, que servirá como nombre de usuario, y un campo check administrador que permite indicar si el nuevo usuario es administrador. Este habilita el campo de contraseña que es obligatorio para usuarios administradores. 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.



6- Ventana de Iniciar Sesión

Por último, se muestra la ventana que aparece al iniciar el programa. Esta presenta el check administrador al comienzo, de manera que según este activado o no, se filtra la lista de DNI mostrada abajo, de manera que se muestran los invitados si no está activada y los administradores si lo está. Además, en este último caso, se habilita el campo contraseña para poder ingresar

VII. MODIFICACIÓN DE FUNCIONALIDADES

Luego de llevar a cabo una entrevista con algunos de los potenciales usuarios finales del sistema se establecieron distintas modificaciones para cubrir nuevas necesidades que son visualizadas en este informe.

- Hubo una solicitud de funcionalidad nueva que permita la creación de un nuevo negocio con sus datos aparejados para que este sea usado dentro de las funciones de sugerencias y autocompletados. Esto se extendió al resto de datos,

como los voluntarios, ubicaciones, etc.

- Se solicitó que haya una opción que permita exportar la tabla general (que se observa en la pantalla principal) a un archivo de planilla de texto
- Hubo un cambio en la estructura de la base de datos de manera que el campo *Dueño* ya no es obligatorio de completar
- Otro cambio estructural se dio con las ubicaciones, de manera que sea más fácil manejarlas y permitir agregarlas individualmente
- Se pidió que en la ventana *Resumen* siempre este visible el peso total recolectado en un día, ya que esta es una de las medidas más importantes y recurrentes
- Múltiples detalles de restricción en los campos de texto para evitar posibles errores (indicar decimales únicamente con puntos por ej.)

VIII. 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.

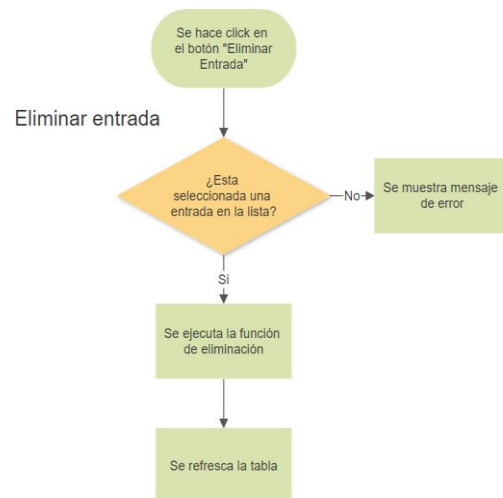
IX. PRUEBAS DE INTERFAZ

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
- Etc.

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).

Anteriormente se mencionó una entrevista realizada 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, voluntarios, 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

X. 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. 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

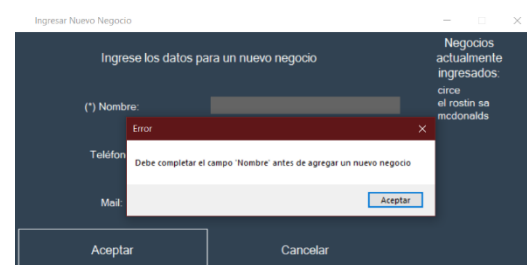
A. Ventana Inserción de Negocio

Para llevar a cabo el testeo de esta primer 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

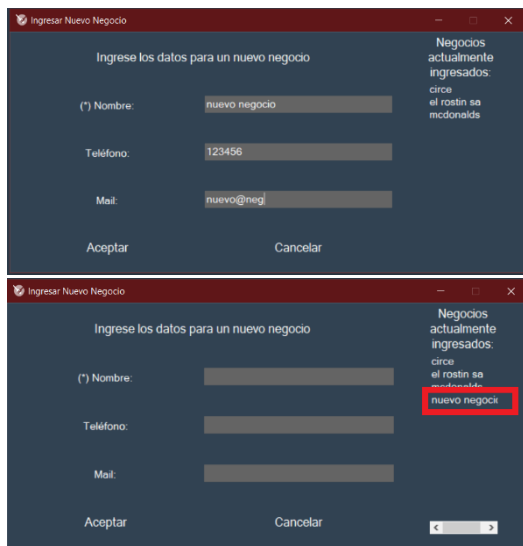
Resultados:

1)



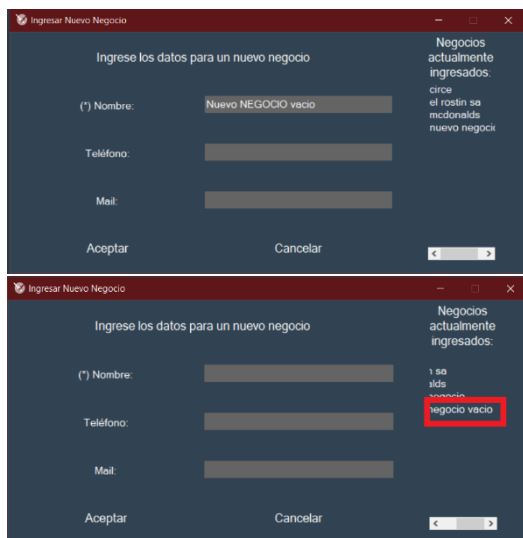
Aquí nos encargamos de comprobar satisfactoriamente que el programa no nos permite avanzar sin completar los campos obligatorios dentro de la BD

2)



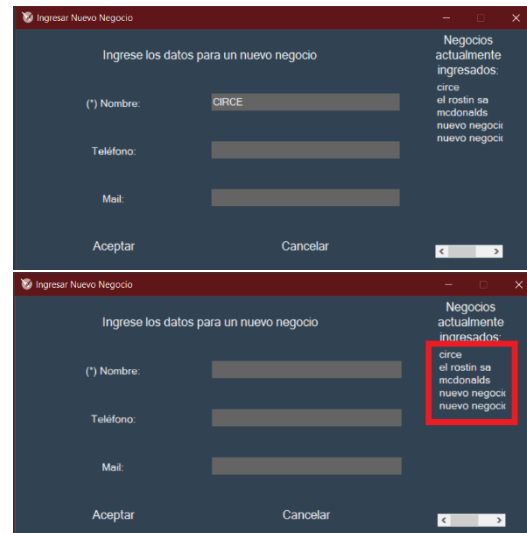
En la primera imagen vemos los campos ingresados y en la segunda vemos el resultado. Señalado en un recuadro rojo podemos ver que aparece en la lista de negocios ingresados el *nuevo negocio*

3)



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)



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 mostro 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:

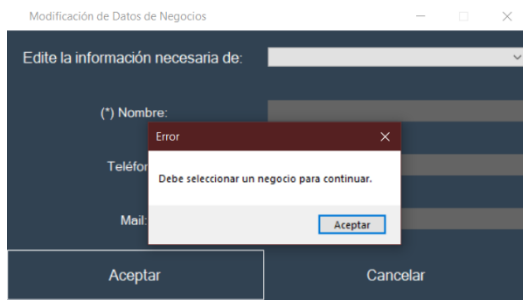
1. Se dejan todos los campos vacíos
2. Se selecciona un negocio y se pulsa *Aceptar*
3. Habiendo seleccionado un negocio, se cambia el nombre por uno preexistente pero escrito en mayúsculas
4. Habiendo seleccionado un negocio, se borra el nombre y se lo deja vacío
5. Se completan todos los campos correctamente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

Resultados:

1)

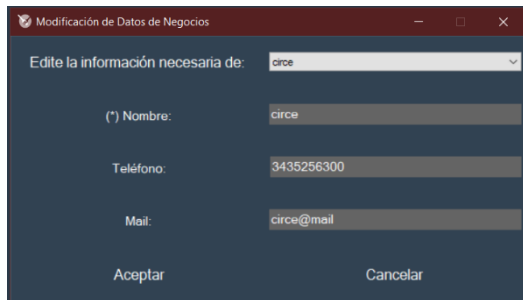
En esta primera prueba fue encontrado un error, ya que al no seleccionar nada, el programa se cerraba porque generaba un error al momento de ejecutar la función.

Una vez solucionado esto, se vuelve a ejecutar la ventana:



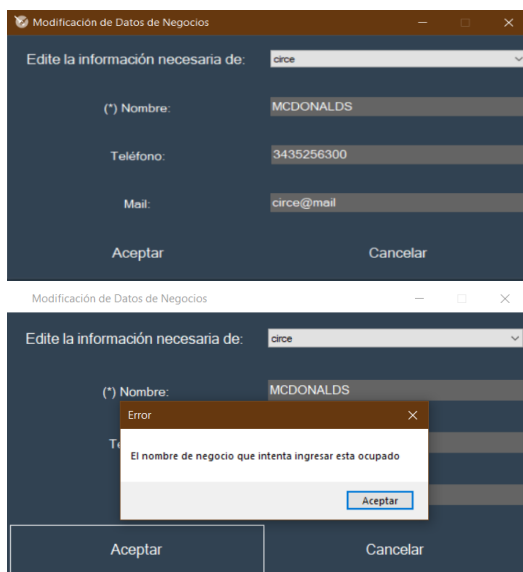
Aquí vemos que, si no se selecciona nada, se solicita que mínimamente se seleccione un elemento a modificar. Caso contrario puede cerrar la ventana mediante el botón cancelar.

2)



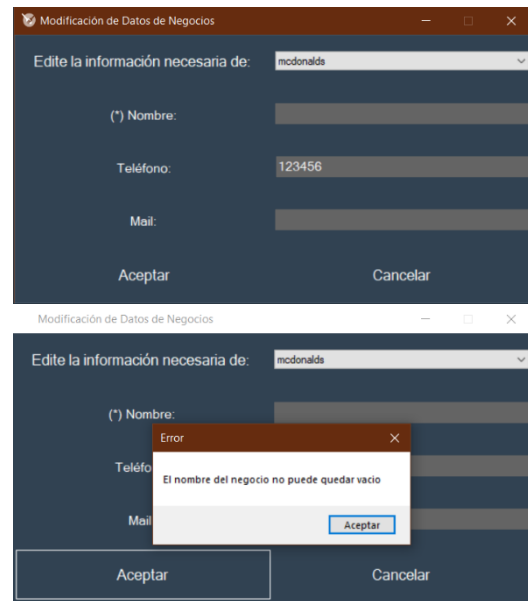
Al seleccionar un elemento en la lista, vemos que se autocompletan los campos restantes de manera satisfactoria. Si pulsamos aceptar la ventana se cierra sin registrar cambio alguno.

3)



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)



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

5)

The first screenshot shows the initial state with 'circe' as the business name, phone number '3435256300', and email 'circe@mail'. The second screenshot shows the name updated to 'nuevo circe' and the email to 'nuevocirce@mail', with the phone number changed to '155256300'. A red box highlights the 'nuevo circe' dropdown. The third screenshot shows the phone number field cleared, while the name and email remain updated.

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.

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 de tipo Administrador 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:

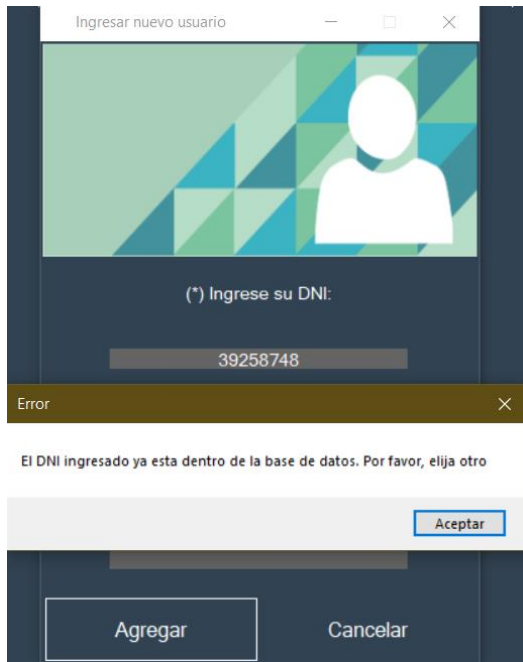
1)

The screenshot shows the 'Ingresar nuevo usuario' window. An error dialog box is displayed with the message 'Debe completar el campo 'DNI''. The background form shows fields for name, DNI, and password, with a checkbox for '¿Administrador?'. The 'Agregar' button is visible at the bottom.

Aquí vemos que se verifican los campos obligatorios al querer agregar un usuario, como el check administrador esta desactivado, el campo contraseña se encuentra deshabilitado.

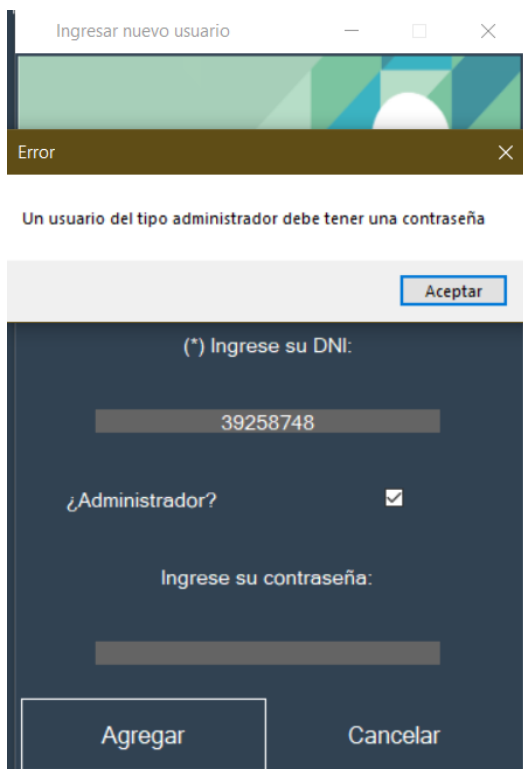
2)

	DNI
	character varying
1	39258748
2	12345678
3	23456789
4	34567891



Aquí se intento usar como DNI uno de los listados en la imagen anterior, los cuales son los que están actualmente ingresados en la BD. Como podemos ver, el programa no nos permite continuar a menos que ingresemos uno nuevo, lo que es un resultado correcto.

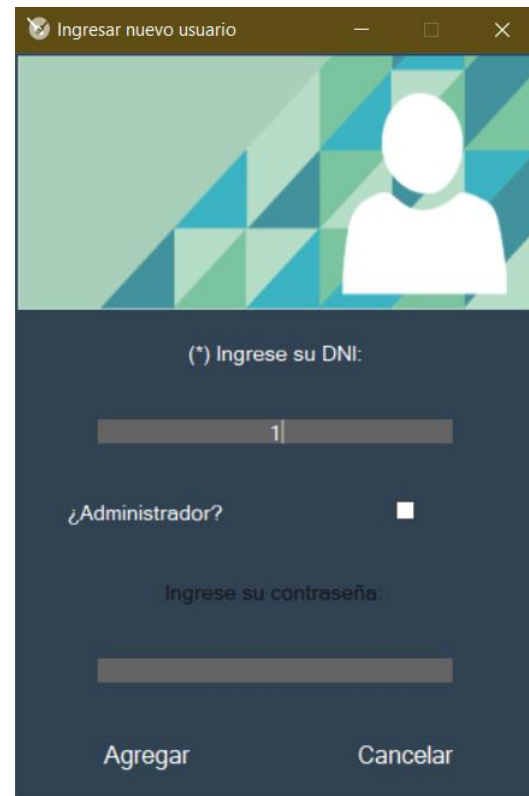
3)



Aquí vemos que si el check Administrador esta seleccionado y se quiere agregar un

nuevo usuario, el programa nos obliga a ingresar una contraseña para poder continuar.

4)



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)

Ingresar nuevo usuario

Error

El campo 'DNI' Solo puede tener 8 caracteres

Aceptar

1234567

¿Administrador? ☐

Ingrese su contraseña:

Agregar Cancelar

Ingresar nuevo usuario

(*) Ingrese su DNI:

23578655

¿Administrador? ☐

Ingrese su contraseña:

Agregar Cancelar

Ingresar nuevo usuario

Error

El campo 'DNI' Solo puede tener 8 caracteres

Aceptar

(*) Ingrese su DNI:

123456789

¿Administrador? ☐

Ingrese su contraseña:

Agregar Cancelar

Ingresar nuevo usuario

(*) Ingrese su DNI:

20554920

¿Administrador? ☒

Ingrese su contraseña:

Agregar Cancelar

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

6)

	DNI character varying	¿Administrador? boolean	Contraseña character varying
1	39258748	true	pass
2	12345678	false	[null]
3	23456789	false	[null]
4	23456789	false	[null]
5	23578655	false	[null]
6	20554920	true	contraseña

En la primera imagen podemos ver la ventana al ingresar un usuario invitado con DNI: “23578655” y en la segunda imagen la misma ventana, pero ingresando un usuario administrador con DNI: “20554920” y contraseña: “contraseña”, la cual aparece oculta. 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)

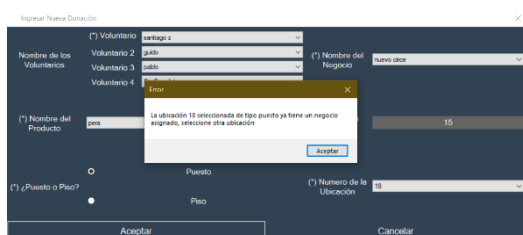


Aquí se muestra el resultado luego de ingresar dos nuevas donaciones. La primera se ingreso en el campo *Peso* “.5” en vez de 0.5, mientras que en la segunda se ingreso “5.” En vez de 5. Esta prueba se realizo para verificar un correcto funcionamiento de la función que ingresa la donación ante distintas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

En esta ultima prueba se verificarán, por un lado, verificaremos 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.

Fecha	Voluntario 1	Voluntario 2	Voluntario 3	Voluntario 4	Negocio	Numero	¿Puesto o Piso?	Telefono	Mail	Duño	Producto	Peso
13/03/2020	santiago z	guido	pablo	Sin Compl...	mcdonalds	18	puesto	123456	Sin Compl...	Sin Compl...	manzana	18
13/03/2020	santiago z	guido	pablo	Sin Compl...	nuevo circo	10	piso		Sin Compl...	nuevocirco	duño c	17.5
20/03/2020	santiago z	guido	pablo	Sin Compl...	nuevo circo	1	puesto		Sin Compl...	nuevocirco	duño a	0.5
20/03/2020	santiago z	guido	pablo	Sin Compl...	nuevo circo	1	puesto		Sin Compl...	nuevocirco	duño a	5
20/03/2020	santiago z	Sin Compl...	guido	Sin Compl...	el roble sa	18	piso		Sin Compl...	Sin Compl...	duño c	15

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.