UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO



PRIMER PARCIAL: CENTRO DE SERVICIO TÉCNICO

NOMBRES: ABASTO BERBETTY, RODRIGO

REGISTRO: 210000491

MATERIA: ARQUITECTURA DE SOFTWARE

FECHA: 24 – ABR – 2019

SANTA CRUZ – BOLIVIA

[1. Descripción del problema 2](#_Toc6885032)

[2. Flujo de trabajo: Captura de requisitos 3](#_Toc6885033)

[2.1. Requisitos funcionales 3](#_Toc6885034)

[2.1.1. Identificar actores y casos de uso 3](#_Toc6885035)

[2.1.2. Especificación de requisitos funcionales 4](#_Toc6885036)

[2.1.3. Diagrama general de casos de uso 16](#_Toc6885037)

[2.2. Requisitos no funcionales 17](#_Toc6885038)

[2.2.1. Calidad 17](#_Toc6885039)

[2.2.2. Restricción 17](#_Toc6885040)

[3. Flujo de trabajo: Análisis 18](#_Toc6885041)

[3.1. Análisis de la arquitectura 18](#_Toc6885042)

[3.1.1. Identificar módulos 18](#_Toc6885043)

[3.1.2. Relacionar módulos con casos de uso 18](#_Toc6885044)

[3.2. Análisis de casos de uso 19](#_Toc6885045)

[3.2.1. Diagrama de comunicación 19](#_Toc6885046)

[4. Flujo de trabajo: Diseño 26](#_Toc6885047)

[4.1. Diseño de arquitectura 26](#_Toc6885048)

[4.1.1. Diseño Lógico 26](#_Toc6885049)

[4.1.2. Diseño Físico 26](#_Toc6885050)

[4.2. Diseño de la base de datos 27](#_Toc6885051)

[4.2.1. Diseño conceptual 27](#_Toc6885052)

[4.2.2. Diseño lógico 27](#_Toc6885053)

[4.2.3. Diseño Físico 29](#_Toc6885054)

[4.3. Diseño de la interfaz 43](#_Toc6885055)

[4.4. Diseño procedimental 47](#_Toc6885056)

[4.4.1. Diagrama de clases dinámico y diagrama de secuencia 47](#_Toc6885057)

[5. Flujo de trabajo: Implementación 66](#_Toc6885058)

[5.1. Diagrama de componentes 66](#_Toc6885059)

## 1. Descripción del problema

ServiTotal es un centro de servicio técnico nuevo en la ciudad, el cual ofrece mantenimiento preventivo y correctivo (reparaciones) para todo tipo de artefactos electrónicos de oficina, (monitores, impresoras, laptops, etc.) luego de unos meses de funcionamiento, han percibido una serie de dificultades que limitan al negocio en su etapa de mejoramiento, y asentamiento en la ciudad. Estas dificultades tienen que ver con el manejo de la información y datos que reciben al momento de proporcionar un servicio de mantenimiento. Así que se planteó el uso de un software que les pudiera ayudar con lo siguiente:

* Registrar los datos básicos de los clientes, como ser: NIT, nombre, email y teléfono. Además, poder registrar todos los equipos que los clientes traigan para su respectivo mantenimiento. Guardando de estos el número de serie, modelo y marca.
* Clasificar a todos los equipos que se recepcionan, según su tipo. Por ejemplo: monitores, impresoras, fotocopiadoras, etc.
* Registrar a todo el personal técnico que posee la empresa, guardando al menos su número de carnet, nombre completo, especialidad, dirección y teléfono.
* Registrar todas las herramientas que posee la empresa, guardando su descripción y marca.
* Manejar un informe de servicio donde se especificará la fecha de recepción de los equipos, el cliente, el costo total del servicio. Además, tendrá un detalle de los equipos sobre los cuales se trabaja y el costo individual del mantenimiento de cada uno.
* Se deberá detallar también todos los trabajos que se realicen a cada uno de los equipos, con una descripción del trabajo y el costo de dicho trabajo, este costo debe ser definido por el técnico, siendo la sumatoria de todos estos trabajos el costo individual del mantenimiento de un solo equipo.
* Se deben registrar el uso de las herramientas que utilizan los técnicos al momento de realizar un determinado trabajo.

## 2. Flujo de trabajo: Captura de requisitos

### 2.1. Requisitos funcionales

#### 2.1.1. Identificar actores y casos de uso

##### Actores



**Cliente**: Es la persona que trae su equipo para que se le realice el mantenimiento respectivo.

**Técnico**: Es la persona que realiza los trabajos que requiere el servicio haciendo uso de las herramientas del negocio.

**Encargado**: Es la persona a cargo del negocio y de supervisar el trabajo de los técnicos, el realiza además el informe de servicio.

##### Casos de uso

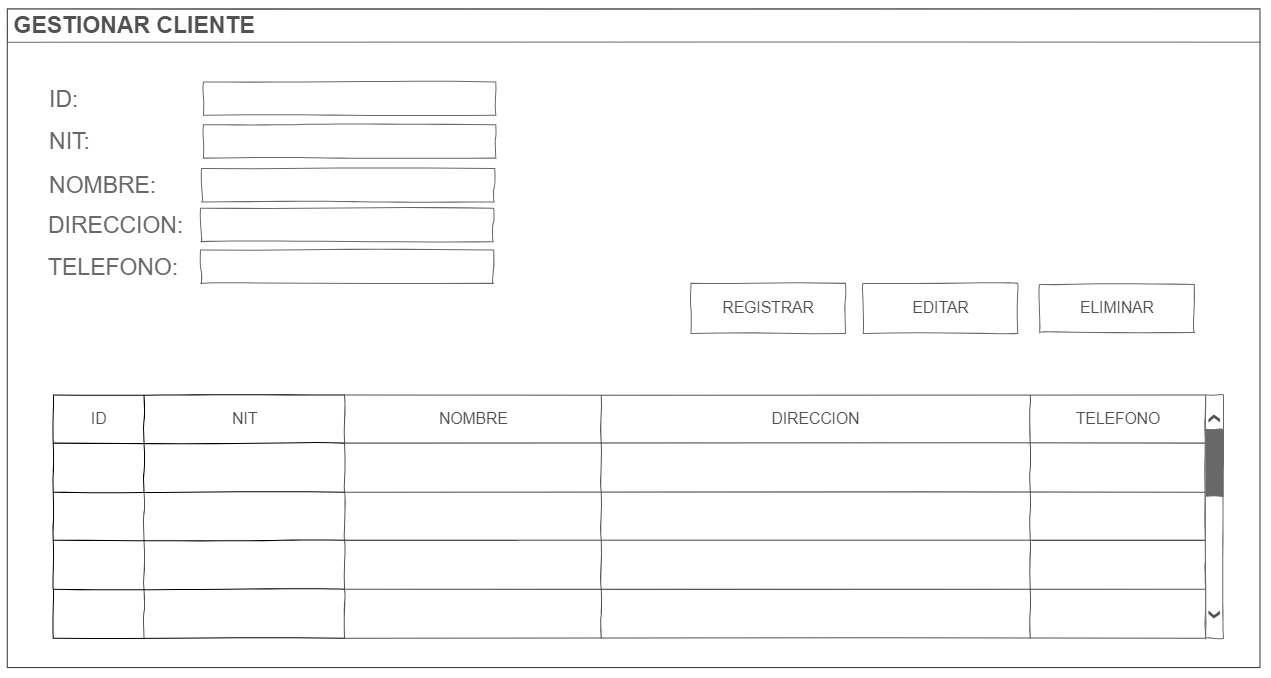
* CU01. Gestionar Cliente
* CU02. Gestionar Equipo
* CU03. Gestionar Tipo de equipo
* CU04. Gestionar Técnico
* CU05. Gestionar Herramienta
* CU06. Gestionar Trabajo
* CU07. Gestionar Informe de Servicio

#### 2.1.2. Especificación de requisitos funcionales

##### CU01. Gestionar Cliente



|  |  |
| --- | --- |
| NOMBRE | CU01. GESTIONAR CLIENTE |
| PROPÓSITO | Almacenar y manipular los datos entregados por el cliente al momento de traer sus equipos al negocio. |
| DESCRIPCIÓN | Permite el registro, la edición y la eliminación de un cliente del negocio. |
| ACTORES | Cliente |
| ACTOR INICIADOR | Cliente |
| PRE-CONDICIÓN | Ninguna. |
| FLUJO | 1. **REGISTRAR**    1. Se llenan los campos para registrar al cliente. Estos son: NIT, nombre, dirección y teléfono.    2. Se procede a guardar los datos proporcionados. Y dichos datos son agregados a la lista de clientes. 2. **EDITAR**    1. Se selecciona algún cliente de la lista.    2. Los datos del cliente serán mostrados en los campos de edición.    3. Se editan los campos.    4. Se procede a guardar todos los datos del cliente. Y se actualiza la lista de cliente. 3. **ELIMINAR**     1. Se selecciona algún cliente de la lista.    2. Los datos del cliente serán mostrados en los campos de edición.    3. Se procede a eliminar al tipo de equipo seleccionado. |
| POST-CONDICIÓN | 1. Se registró un nuevo cliente.  2. Se editó un cliente.  3. Se eliminó un cliente. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.3. No se seleccionó algún cliente a modificar.  3.3. No se seleccionó algún cliente a eliminar. |



##### CU02. Gestionar Equipo



|  |  |
| --- | --- |
| NOMBRE | CU02. GESTIONAR EQUIPO |
| PROPÓSITO | Almacenar y manipular los datos entregados por el cliente sobre sus equipos entregados. |
| DESCRIPCIÓN | Permite el registro, la edición y la eliminación de un equipo traído al negocio por un cliente. |
| ACTORES | Cliente |
| ACTOR INICIADOR | Cliente |
| PRE-CONDICIÓN | Debe existir al menos un tipo de equipo. |
| FLUJO | 1. **REGISTRAR**    1. Se llenan los campos para registrar del equipo traído por el cliente. Estos son: modelo, número de serie, marca, observación. Además, se debe seleccionar el tipo de equipo al que corresponde.    2. Se procede a guardar los datos proporcionados. Y dichos datos son agregados a la lista de equipos. 2. **EDITAR**    1. Se selecciona algún equipo de la lista.    2. Los datos del equipo serán mostrados en los campos de edición.    3. Se editan los campos.    4. Se procede a guardar todos los datos del equipo. Y se actualiza la lista de equipos. 3. **ELIMINAR**    1. Se selecciona el equipo que desee eliminar de la lista.    2. Los datos del equipo serán mostrados en los campos de edición.    3. Se procede a eliminar al equipo seleccionado. |
| POST-CONDICIÓN | 1. Se registró un nuevo equipo.  2. Se editó un equipo.  3. Se eliminó un equipo. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.4. No se seleccionó un equipo a modificar.  3.3. No se seleccionó un equipo a eliminar. |



##### CU03. Gestionar Tipo de equipo



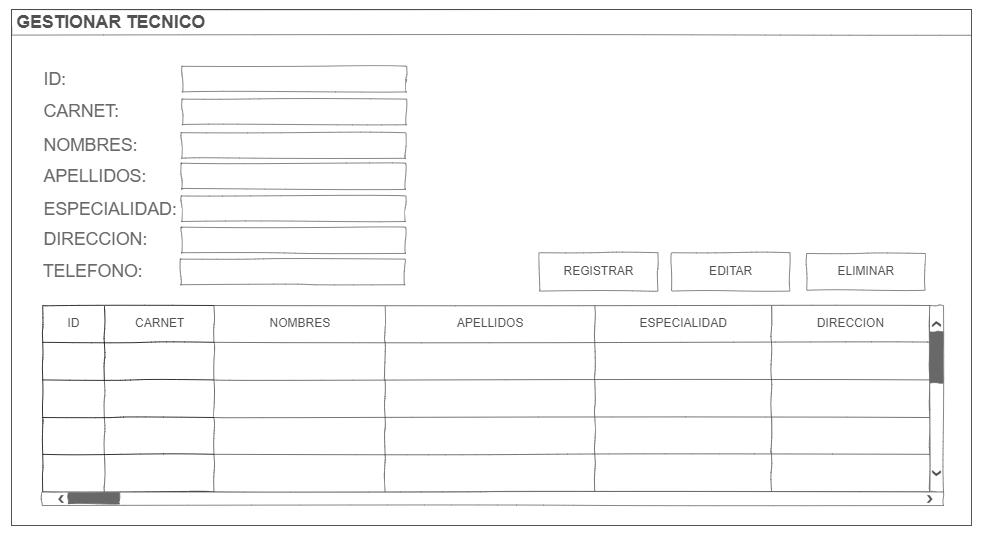
|  |  |
| --- | --- |
| NOMBRE | CU03. GESTIONAR TIPO DE EQUIPO |
| PROPÓSITO | Almacenar y manipular los diferentes tipos de equipos que recibe el negocio. |
| DESCRIPCIÓN | Permite el registro, la edición y la eliminación de un tipo de equipo que recibe el negocio. |
| ACTORES | Encargado |
| ACTOR INICIADOR | Encargado |
| PRE-CONDICIÓN | Ninguna |
| FLUJO | 1. **REGISTRAR**    1. Se llenan los campos para registrar el tipo. Es decir, solo el nombre.    2. Se procede a guardar los datos proporcionados. Y dichos datos son agregados a la lista de tipos. 2. **EDITAR**    1. Se selecciona algún tipo de la lista.    2. Los datos del tipo de equipo serán mostrados en los campos de edición.    3. Una vez editado el campo deseado se procede a guardar todos los datos del tipo. Y se actualiza la lista de tipos de equipo. 3. **ELIMINAR**     1. Se selecciona algún tipo de la lista.    2. Los datos del tipo serán mostrados.    3. Se procede a eliminar al tipo de equipo seleccionado. |
| POST-CONDICIÓN | 1. Se registró un nuevo tipo de equipo.  2. Se editó un tipo de equipo.  3. Se eliminó un tipo de equipo. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.3. No se seleccionó un tipo de equipo a modificar.  3.3. No se seleccionó un tipo de equipo a eliminar. |



##### CU04. Gestionar Técnico



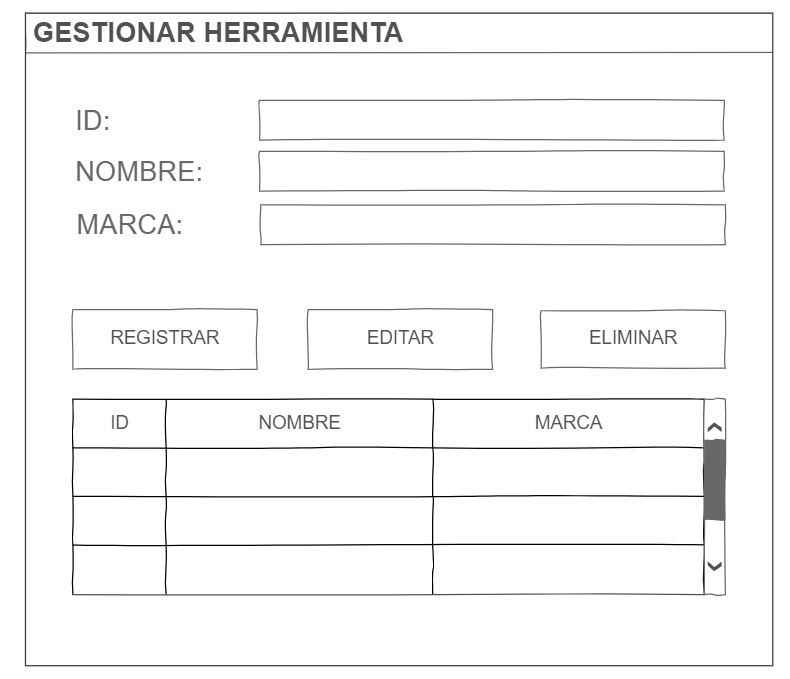
|  |  |
| --- | --- |
| NOMBRE | CU04. GESTIONAR TÉCNICO |
| PROPÓSITO | Almacenar y manipular los datos entregados por los técnicos los cuales posee el negocio. |
| DESCRIPCIÓN | Permite el registro, la edición y la eliminación de un técnico del negocio. |
| ACTORES | Técnico |
| ACTOR INICIADOR | Técnico |
| PRE-CONDICIÓN | Ninguna |
| FLUJO | 1. **REGISTRAR**    1. Se llenan los campos para registrar al técnico. Estos son: número de carnet, nombres, apellidos, especialidad, dirección y teléfono.    2. Se procede a guardar los datos proporcionados. Y dichos datos son agregados a la lista de técnicos. 2. **EDITAR**    1. Se selecciona algún técnico de la lista.    2. Los datos del técnico serán mostrados en los campos de edición.    3. Se procede a guardar todos los datos del técnico. Y se actualiza la lista de técnicos. 3. **ELIMINAR**     1. Se selecciona algún técnico de la lista.    2. Los datos del técnico serán mostrados en los campos de edición.    3. Se procede a eliminar al técnico seleccionado. |
| POST-CONDICIÓN | 1. Se registró un técnico.  2. Se editó un técnico.  3. Se eliminó un técnico. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.3. No se seleccionó un técnico a modificar.  3.3. No se seleccionó un técnico a eliminar. |



##### CU05. Gestionar Herramienta



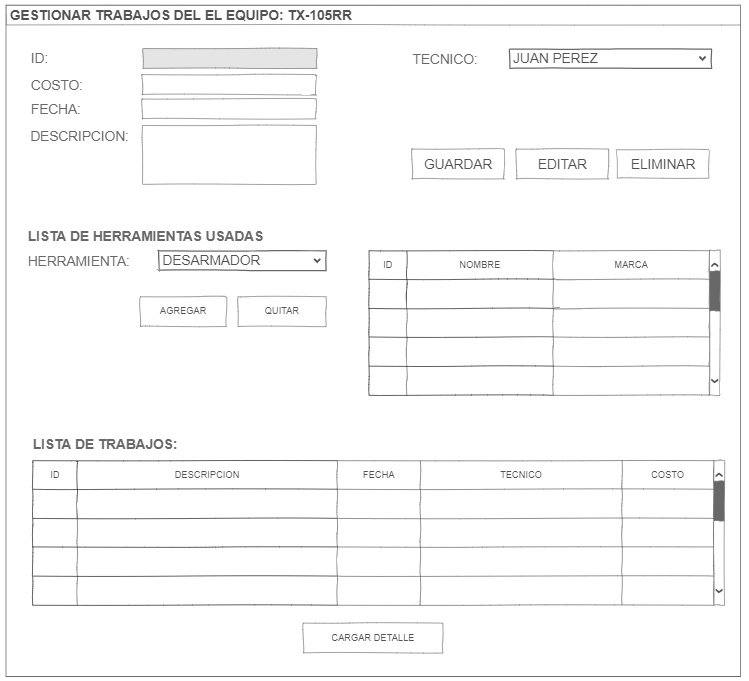
|  |  |
| --- | --- |
| NOMBRE | CU05. GESTIONAR HERRAMIENTA |
| PROPÓSITO | Almacenar y manipular los datos de las herramientas del negocio, las cuales son utilizadas dentro de los servicios realizados a los equipos. |
| DESCRIPCIÓN | Permite el registro, la edición y la eliminación de una herramienta del negocio. |
| ACTORES | Encargado |
| ACTOR INICIADOR | Encargado |
| PRE-CONDICIÓN | Ninguna |
| FLUJO | 1. **REGISTRAR**    1. Se llenan los campos para registrar la herramienta. Es decir, solo el nombre y marca.    2. Se procede a guardar los datos proporcionados. Y dichos datos son agregados a la lista de herramienta. 2. **EDITAR**    1. Se selecciona alguna herramienta de la lista.    2. Los datos de la herramienta serán mostrados en los campos de edición.    3. Se procede a guardar todos los datos de la herramienta. Y se actualiza la lista de herramientas. 3. **ELIMINAR**     1. Se selecciona alguna herramienta de la lista.    2. Los datos de la herramienta serán mostrados en los campos de edición.    3. Se procede a eliminar la herramienta seleccionada. |
| POST-CONDICIÓN | 1. Se registró una nueva herramienta.  2. Se editó una herramienta.  3. Se eliminó una herramienta. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.3. No se seleccionó una herramienta a modificar.  3.3. No se seleccionó una herramienta a eliminar. |



##### CU06. Gestionar Trabajo



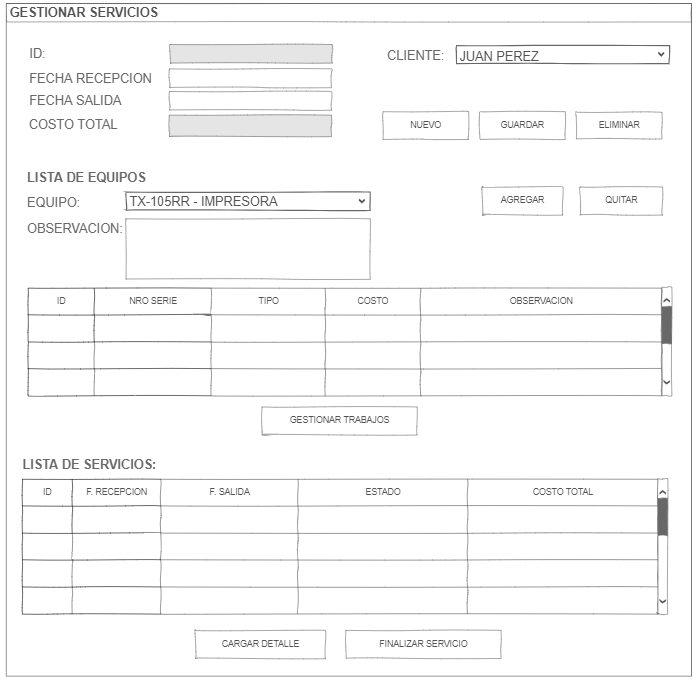
|  |  |
| --- | --- |
| NOMBRE | CU06. GESTIONAR TRABAJO |
| PROPÓSITO | Almacenar y manipular los trabajos realizados por los técnicos dentro del servicio de mantenimiento para un equipo. |
| DESCRIPCIÓN | Permite el registro, la edición y la eliminación de un trabajo en específico realizado por un técnico durante el servicio de mantenimiento de un equipo. |
| ACTORES | Técnico |
| ACTOR INICIADOR | Técnico |
| PRE-CONDICIÓN | Debe existir un informe de servicio con al menos un equipo. Además, debe haber al menos una herramienta registrada al igual que un técnico. |
| FLUJO | 1. **COMO ACCEDER**    1. En el detalle del informe de servicio se debe seleccionar un equipo para poder gestionar sus trabajos correspondientes 2. **REGISTRAR**    1. Se llenan los siguientes campos para registrar el trabajo realizado: descripción, fecha, costo y se debe seleccionar un técnico.    2. Se procede a guardar los datos proporcionados. Además, también se actualizará el costo del servicio de mantenimiento por ese equipo y por todo el servicio en general. Y dichos datos son agregados a la lista de trabajos de ese equipo. 3. **EDITAR**    1. Se selecciona algún trabajo de la lista.    2. Los datos del trabajo serán mostrados en los campos de edición. Y se procede de la misma manera que al registrar el trabajo.    3. Se procede a guardar todos los datos del trabajo. Además, también se actualizará el costo del servicio de mantenimiento por ese equipo y por todo el servicio en general. Y dichos datos serán actualizados en la lista de trabajos de ese equipo. 4. **ELIMINAR**     1. Se selecciona algún trabajo de la lista.    2. Los datos del trabajo serán mostrados.    3. Se procede a eliminar el trabajo seleccionado. Además, también se actualizará el costo del servicio por ese equipo y por todo el servicio en general. 5. **DETALLE**    1. Para poder agregar o quitar una herramienta del detalle del trabajo será necesario antes seleccionar el trabajo de cual se quiera agregar, esto habilitará la sección del detalle donde se cargará las herramientas ya agregadas.    2. Una vez habilitado el detalle se debe agregar las distintas herramientas que fueron utilizadas en el trabajo. Esto se hará a través de un selector de herramientas. Cada vez que se agregue una herramienta se está guardando directamente.    3. Para quitar una herramienta solo se debe seleccionarla en la lista del detalle y presionar quitar, a medida que se vaya quitando se irá guardando. |
| POST-CONDICIÓN | 2. Se registró un nuevo trabajo.  3. Se editó un trabajo.  4. Se eliminó un trabajo. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.3. No se seleccionó un trabajo a modificar.  3.3. No se seleccionó un trabajo a eliminar. |



##### CU07. Gestionar Informe de Servicio



|  |  |
| --- | --- |
| NOMBRE | CU07. GESTIONAR INFORME DE SERVICIO |
| PROPÓSITO | Almacenar y manipular los datos del servicio de mantenimiento a los equipos traídos por un cliente. |
| DESCRIPCIÓN | Permite el registro, la edición, finalización y la anulación de un informe de servicio de mantenimiento. |
| ACTORES | Encargado |
| ACTOR INICIADOR | Encargado |
| PRE-CONDICIÓN | Debe existir al menos un cliente registrado junto con sus equipos. |
| FLUJO | 1. **REGISTRAR**    1. Se inserta la fecha de recepción de los equipos, se selecciona un cliente. El campo costo total no será editable, ya que este se calcula por medio de la sumatoria de los costos del mantenimiento individual de cada equipo.    2. Se procede a guardar los datos proporcionados. Y dichos datos son agregados a la lista de informes de servicio. 2. **EDITAR**    1. Se selecciona algún informe de servicio no terminado de la lista de servicios.    2. Los datos del informe de servicio serán mostrados en los campos de edición.    3. Luego de actualizar los campos deseados se procede a guardar. Y se actualiza la lista de informes de servicios. 3. **ANULAR**     1. Se selecciona algún servicio de la lista.    2. Los datos del servicio serán mostrados en los campos de edición.    3. Se procede a anular al servicio seleccionado. 4. **FINALIZAR INFORME**   Cuando se considere que todos los equipos han terminado con todos sus trabajos necesarios estos podrán ser finalizados. Para ello hacemos lo siguiente:   * 1. Seleccionar el Informe de servicio que se desea finalizar.   2. Rellenar la fecha de finalización y selecciona finalizar informe.  1. **DETALLE**    1. Para poder agregar o quitar un equipo del detalle del informe será necesario antes seleccionar el informe de cual se quiera agregar, esto habilitará la sección del detalle donde se cargarán los equipos ya agregadas.    2. Una vez habilitado el detalle se debe agregar los equipos sobre los cuales se trabajarán. Esto se hará a través de un selector de equipos. Cada vez que se agregue un equipo se está guardando directamente.    3. Para quitar un equipo solo se debe seleccionarlo en la lista del detalle y presionar quitar, a medida que se vaya quitando se irá guardando. |
| POST-CONDICIÓN | 1. Se registró un nuevo informe de servicio.  2. Se editó un informe de servicio.  3. Se anuló un informe de servicio.  4. Se finalizó un informe de servicio. |
| EXCEPCIONES | 1.2. Todos los campos son obligatorios.  2.3. No se seleccionó un informe a modificar.  3.3. No se seleccionó un informe a anular.  4.4. No se seleccionó un informe para ser finalizado. |



#### 2.1.3. Diagrama general de casos de uso



### 2.2. Requisitos no funcionales

#### 2.2.1. Calidad

* El software deberá tener un tiempo óptimo de respuesta a eventos (menos de 3 segundos).
* El usuario final no deberá tardar más de una hora en la compresión del uso correcto del software.
* El software debe contar con un manual de usuario estructurado adecuadamente, con imágenes y ejemplos.

#### 2.2.2. Restricción

* El tiempo de desarrollo del software no debe ser superior a 2 meses.
* El software no podrá ocupar más de 512MB de espacio en disco.
* El software debe ser compatible con todas las versiones de Windows superiores a Windows 7 o con Ubuntu.
* El software funcionará en un solo equipo que trabaja con MySQL.

## 3. Flujo de trabajo: Análisis

### 3.1. Análisis de la arquitectura

#### 3.1.1. Identificar módulos

##### Módulo de Administración



#### 3.1.2. Relacionar módulos con casos de uso



### 3.2. Análisis de casos de uso

#### 3.2.1. Diagrama de comunicación

##### CU01. Gestionar Cliente



##### CU02. Gestionar Equipo



##### CU03. Gestionar Tipo de equipo



##### CU04. Gestionar Técnico



##### CU05. Gestionar Herramienta



##### CU06. Gestionar Trabajo



##### CU07. Gestionar Informe de Servicio



## 4. Flujo de trabajo: Diseño

### 4.1. Diseño de arquitectura

#### 4.1.1. Diseño Lógico



#### 4.1.2. Diseño Físico



### 4.2. Diseño de la base de datos

#### 4.2.1. Diseño conceptual

#### 4.2.2. Diseño lógico

**Cliente**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PK** |  |  |  |  |
| **id** | nit | nombre | direccion | telefono |

**Tecnico**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PK** |  |  |  |  |  |  |
| **id** | ci | nombre | apellido | especialidad | direccion | telefono |

**Tipo**:

|  |  |
| --- | --- |
| **PK** |  |
| **id** | nombre |

**Herramienta**:

|  |  |  |
| --- | --- | --- |
| **PK** |  |  |
| **id** | descripcion | marca |

**Equipo**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PK** |  |  |  | **FK** |
| **id** | modelo | nro\_serie | marca | **tipo\_id** |

**Informe de Servicio**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PK** |  |  |  |  | **FK** |
| **id** | fecha\_recepcion | fecha\_finalizacion | costo\_total | estado | **cliente\_id** |

Detalle informe de servicio:

|  |  |  |  |
| --- | --- | --- | --- |
| **PK/FK** | **PK/FK** |  |  |
| **informe\_servicio\_id** | **equipo\_id** | costo | Observacion |

Trabajo:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PK** |  |  |  |  | **FK** |
| **id** | fecha\_inicio | fecha\_fin | costo | descripcion | **tecnico\_id** |

|  |  |
| --- | --- |
| **FK** | **FK** |
| **detalle\_informe\_informe\_servicio\_id** | **detalle\_informe\_equipo\_id** |

Detalle trabajo:

|  |  |
| --- | --- |
| **PK/FK** | **PK/FK** |
| **trabajo\_id** | **herramienta\_id** |

#### 4.2.3. Diseño Físico

##### 4.2.3.1. Tabla de volumen

**Cliente**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador del cliente |
| nit | Texto | 20 | - | No | NIT del cliente |
| nombre | Texto | 45 | - | No | Nombre del cliente |
| direccion | Texto | 100 | - | No | Dirección del cliente |
| telefono | Texto | 10 | - | No | Nro de teléfono o celular del cliente |

**Técnico**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador del técnico |
| ci | Texto | 15 | - | No | Carnet de identidad del técnico |
| nombre | Texto | 45 | - | No | Nombres del técnico |
| apellido | Texto | 45 | - | No | Apellidos del técnico |
| especialidad | Texto | 45 | - | No | Área de especialidad del técnico |
| direccion | Texto | 100 | - | No | Dirección del domicilio del técnico |
| telefono | Texto | 10 | - | No | Nro de teléfono o celular del técnico |

**Tipo**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador del tipo de equipo |
| nombre | Texto | 45 | - | No | Nombre del tipo de equipo. |

**Herramienta**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador de la herramienta |
| descripcion | Texto | 45 | - | No | Nombre y descripción física de la herramienta. |
| marca | Texto | 45 | - | No | Nombre de la marca de la herramienta. |

**Equipo**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador del equipo |
| modelo | Texto | 45 | - | No | Nombre del modelo del equipo |
| nro\_serie | Texto | 45 | - | No | Nro de serie del equipo |
| marca | Texto | 45 | - | No | Nombre de la marca del equipo |
| tipo\_id | Numérico | 11 | FK | No | Identificador foráneo al tipo de equipo |

**Informe de servicio**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador del informe |
| fecha\_recepcion | Texto | 20 | - | No | Fecha de recepción de los aparatos |
| fecha\_finalizacion | Texto | 20 | - | No | Fecha de finalización del servicio |
| costo\_total | Numérico (Float) | - | - | No | Costo total del servicio |
| estado | Texto | 20 | - | No | Estado actual del servicio |
| cliente\_id | Numérico | 11 | FK | No | Identificador foráneo al cliente |

**Detalle de informe de servicio**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| informe\_servicio\_id | Numérico | 11 | PK/FK | No | Identificador foráneo al informe |
| equipo\_id | Numérico | 11 | PK/FK | No | Identificador foráneo al equipo |
| costo | Numérico (Float) | - | - | No | Costo individual al servicio |
| observacion | Texto | 100 | - | No | Observación de algo en el equipo |

**Trabajo**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| id | Numérico | 11 | PK | No | Identificador del trabajo |
| fecha\_inicio | Texto | 20 | - | No | Fecha de inicio del trabajo |
| fecha\_fin | Texto | 20 | - | No | Fecha de finalización del trabajo |
| costo | Numérico (Float) | - | - | No | Costo individual del trabajo sobre un equipo |
| descripcion | Texto | 100 | - | No | Descripción de lo realizado |
| tecnico\_id | Numérico | 11 | FK | No | Identificador foráneo al técnico que realizo el trabajo |
| detalle\_informe\_informe\_servicio\_id | Numérico | 11 | FK | No | Identificador foráneo al detalle del informe |
| detalle\_informe\_equipo\_id | Numérico | 11 | FK | No | Identificador foráneo al detalle del informe |

**Detalle de trabajo**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE** | **TIPO DE DATO** | **AMPLITUD** | **LLAVE** | **NULO** | **DESCRIPCIÓN** |
| trabajo\_id | Numérico | 11 | PK/FK | No | Identificador foráneo al trabajo |
| herramienta\_id | Numérico | 11 | PK/FK | No | Identificador foráneo a la herramienta usada |

##### 4.2.3.2. Script

-- -----------------------------------------------------

-- Schema arquitectura

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `arquitectura` DEFAULT CHARACTER SET utf8 ;

USE `arquitectura` ;

-- -----------------------------------------------------

-- Table `arquitectura`.`cliente`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`cliente` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nit` VARCHAR(20) NOT NULL,

`nombre` VARCHAR(45) NOT NULL,

`direccion` VARCHAR(100) NOT NULL,

`telefono` VARCHAR(10) NOT NULL,

PRIMARY KEY (`id`));

-- -----------------------------------------------------

-- Table `arquitectura`.`tecnico`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`tecnico` (

`id` INT NOT NULL AUTO\_INCREMENT,

`ci` VARCHAR(15) NOT NULL,

`nombre` VARCHAR(45) NOT NULL,

`apellido` VARCHAR(45) NOT NULL,

`especialidad` VARCHAR(45) NOT NULL,

`direccion` VARCHAR(100) NOT NULL,

`telefono` VARCHAR(10) NOT NULL,

PRIMARY KEY (`id`));

-- -----------------------------------------------------

-- Table `arquitectura`.`tipo`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`tipo` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nombre` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`));

-- -----------------------------------------------------

-- Table `arquitectura`.`herramienta`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`herramienta` (

`id` INT NOT NULL AUTO\_INCREMENT,

`descripcion` VARCHAR(45) NOT NULL,

`marca` VARCHAR(45) NOT NULL,

PRIMARY KEY (`id`));

-- -----------------------------------------------------

-- Table `arquitectura`.`equipo`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`equipo` (

`id` INT NOT NULL AUTO\_INCREMENT,

`modelo` VARCHAR(20) NOT NULL,

`nro\_serie` VARCHAR(20) NOT NULL,

`marca` VARCHAR(20) NOT NULL,

`tipo\_id` INT NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_equipo\_tipo\_idx` (`tipo\_id` ASC) ,

CONSTRAINT `fk\_equipo\_tipo`

FOREIGN KEY (`tipo\_id`)

REFERENCES `arquitectura`.`tipo` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE);

-- -----------------------------------------------------

-- Table `arquitectura`.`informe\_servicio`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`informe\_servicio` (

`id` INT NOT NULL AUTO\_INCREMENT,

`fecha\_recepcion` VARCHAR(20) NOT NULL,

`fecha\_finalizacion` VARCHAR(20) NOT NULL,

`costo\_total` FLOAT NOT NULL,

`estado` VARCHAR(20) NOT NULL,

`cliente\_id` INT NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_informe\_servicio\_cliente1\_idx` (`cliente\_id` ASC) ,

CONSTRAINT `fk\_informe\_servicio\_cliente1`

FOREIGN KEY (`cliente\_id`)

REFERENCES `arquitectura`.`cliente` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE);

-- -----------------------------------------------------

-- Table `arquitectura`.`detalle\_informe`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`detalle\_informe` (

`informe\_servicio\_id` INT NOT NULL,

`equipo\_id` INT NOT NULL,

`costo` FLOAT NOT NULL,

`observacion` VARCHAR(100) NOT NULL,

PRIMARY KEY (`informe\_servicio\_id`, `equipo\_id`),

INDEX `fk\_detalle\_informe\_equipo1\_idx` (`equipo\_id` ASC) ,

CONSTRAINT `fk\_detalle\_informe\_informe\_servicio1`

FOREIGN KEY (`informe\_servicio\_id`)

REFERENCES `arquitectura`.`informe\_servicio` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `fk\_detalle\_informe\_equipo1`

FOREIGN KEY (`equipo\_id`)

REFERENCES `arquitectura`.`equipo` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE);

-- -----------------------------------------------------

-- Table `arquitectura`.`trabajo`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`trabajo` (

`id` INT NOT NULL AUTO\_INCREMENT,

`fecha\_inicio` VARCHAR(20) NOT NULL,

`fecha\_fin` VARCHAR(20) NOT NULL,

`costo` FLOAT NOT NULL,

`descripcion` VARCHAR(100) NOT NULL,

`tecnico\_id` INT NOT NULL,

`detalle\_informe\_informe\_servicio\_id` INT NOT NULL,

`detalle\_informe\_equipo\_id` INT NOT NULL,

PRIMARY KEY (`id`),

INDEX `fk\_trabajo\_tecnico1\_idx` (`tecnico\_id` ASC) ,

INDEX `fk\_trabajo\_detalle\_informe1\_idx` (`detalle\_informe\_informe\_servicio\_id` ASC, `detalle\_informe\_equipo\_id` ASC) ,

CONSTRAINT `fk\_trabajo\_tecnico1`

FOREIGN KEY (`tecnico\_id`)

REFERENCES `arquitectura`.`tecnico` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `fk\_trabajo\_detalle\_informe1`

FOREIGN KEY (`detalle\_informe\_informe\_servicio\_id` , `detalle\_informe\_equipo\_id`)

REFERENCES `arquitectura`.`detalle\_informe` (`informe\_servicio\_id` , `equipo\_id`)

ON DELETE CASCADE

ON UPDATE CASCADE);

-- -----------------------------------------------------

-- Table `arquitectura`.`detalle\_trabajo`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `arquitectura`.`detalle\_trabajo` (

`trabajo\_id` INT NOT NULL,

`herramienta\_id` INT NOT NULL,

PRIMARY KEY (`trabajo\_id`, `herramienta\_id`),

INDEX `fk\_detalle\_trabajo\_herramienta1\_idx` (`herramienta\_id` ASC) ,

CONSTRAINT `fk\_detalle\_trabajo\_trabajo1`

FOREIGN KEY (`trabajo\_id`)

REFERENCES `arquitectura`.`trabajo` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE,

CONSTRAINT `fk\_detalle\_trabajo\_herramienta1`

FOREIGN KEY (`herramienta\_id`)

REFERENCES `arquitectura`.`herramienta` (`id`)

ON DELETE CASCADE

ON UPDATE CASCADE);

-- \*\*\*\*\*\*\*\*\*\* POBLACION \*\*\*\*\*\*\*\*\*\*

INSERT INTO CLIENTE VALUES (NULL, '12345', 'FARMACORP', 'DOBLE VIA', '3542102');

INSERT INTO CLIENTE VALUES (NULL, '78954', 'INQUIFARMED', 'RADIAL 17 1/2', '3532412');

INSERT INTO CLIENTE VALUES (NULL, '96542', 'UAGRM', 'AV. BUSCH', '3333542');

INSERT INTO CLIENTE VALUES (NULL, '87456', 'COPIJSUD', 'AV. CUMORAH', '3587765');

INSERT INTO CLIENTE VALUES (NULL, '13335', 'VOLTEC', 'AV. BANZER', '3974454');

INSERT INTO TECNICO VALUES (NULL, '815245', 'JUAN', 'PEREZ', 'IMPRESORAS', 'DOBLE VIA', '3542102');

INSERT INTO TECNICO VALUES (NULL, '815245', 'JORGE', 'TORRES', 'MONITORES', 'DOBLE VIA', '3542102');

INSERT INTO TECNICO VALUES (NULL, '815245', 'MARIO', 'TOLEDO', 'TELEVISORES', 'DOBLE VIA', '3542102');

INSERT INTO TECNICO VALUES (NULL, '815245', 'NORMA', 'GUTIERREZ', 'TECLADOS', 'DOBLE VIA', '3542102');

INSERT INTO TECNICO VALUES (NULL, '815245', 'CARMEN', 'ARANCIBIA', 'CASES', 'DOBLE VIA', '3542102');

INSERT INTO TIPO VALUES (NULL, 'MONITOR');

INSERT INTO TIPO VALUES (NULL, 'IMPRESORA');

INSERT INTO TIPO VALUES (NULL, 'FUENTE');

INSERT INTO TIPO VALUES (NULL, 'TECLADO');

INSERT INTO TIPO VALUES (NULL, 'PARLANTE');

INSERT INTO HERRAMIENTA VALUES (NULL, 'DESARMADOR ESTRELLA AMARILLO PEQUEÑO', 'GENERICO');

INSERT INTO HERRAMIENTA VALUES (NULL, 'DESARMADOR PLANO AZUL MEDIANO', 'TRAMONTINA');

INSERT INTO HERRAMIENTA VALUES (NULL, 'ALICATE DE PUNTA ROJO', 'GENERICO');

INSERT INTO HERRAMIENTA VALUES (NULL, 'ALICATE DE CORTE ROJO', 'UNITED');

INSERT INTO HERRAMIENTA VALUES (NULL, 'SOLDADOR ELECTRONICO NEGRO', 'VOLTRON');

INSERT INTO EQUIPO VALUES (NULL, 'JS-3432', 'ASD12312', 'LG', 3);

INSERT INTO EQUIPO VALUES (NULL, 'SD455D4', '4561231', 'ASUS', 1);

INSERT INTO EQUIPO VALUES (NULL, 'TR8454S', '8945212', 'SAMSUNG', 5);

INSERT INTO EQUIPO VALUES (NULL, 'ASD78A', '2N121312', 'EPSON', 3);

INSERT INTO EQUIPO VALUES (NULL, 'QWHE12J', '3412123W', 'EPSON', 1);

INSERT INTO EQUIPO VALUES (NULL, 'FDKSA02', 'R4322', 'HP', 2);

INSERT INTO EQUIPO VALUES (NULL, 'KSFD123', '675876D', 'CANON', 4);

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\* PROCEDIMIENTOS \*\*\*\*\*\*\*\*\*\*\*\*

DROP procedure IF EXISTS `proc\_actualizar\_detalle\_informe`;

DELIMITER $$

USE `arquitectura`$$

CREATE PROCEDURE proc\_actualizar\_detalle\_informe (IN i\_id int, in e\_id int)

BEGIN

-- Variables donde almacenar lo que nos traemos desde el SELECT

declare resultado float default 0;

declare v\_costo float default 0;

-- Variable para controlar el fin del bucle

DECLARE fin INTEGER DEFAULT 0;

-- El SELECT que vamos a ejecutar

DECLARE trabajos\_cursor CURSOR FOR

SELECT costo FROM trabajo WHERE detalle\_informe\_informe\_servicio\_id=i\_id and detalle\_informe\_equipo\_id=e\_id;

-- Condición de salida

DECLARE CONTINUE HANDLER FOR NOT FOUND SET fin=1;

OPEN trabajos\_cursor;

get\_trabajos: LOOP

FETCH trabajos\_cursor INTO v\_costo;

IF fin = 1 THEN

LEAVE get\_trabajos;

END IF;

set resultado = resultado + v\_costo;

END LOOP get\_trabajos;

UPDATE detalle\_informe

SET

costo = resultado

WHERE

informe\_servicio\_id = i\_id AND equipo\_id = e\_id;

CLOSE trabajos\_cursor;

END$$

DELIMITER ;

DROP procedure IF EXISTS `proc\_actualizar\_informe`;

DELIMITER $$

USE `arquitectura`$$

CREATE PROCEDURE proc\_actualizar\_informe (IN i\_id int)

BEGIN

-- Variables donde almacenar lo que nos traemos desde el SELECT

declare resultado float default 0;

declare v\_costo float default 0;

-- Variable para controlar el fin del bucle

DECLARE fin INTEGER DEFAULT 0;

-- El SELECT que vamos a ejecutar

DECLARE detalles\_cursor CURSOR FOR

SELECT costo FROM detalle\_informe WHERE informe\_servicio\_id=i\_id;

-- Condición de salida

DECLARE CONTINUE HANDLER FOR NOT FOUND SET fin=1;

OPEN detalles\_cursor;

get\_detalles: LOOP

FETCH detalles\_cursor INTO v\_costo;

IF fin = 1 THEN

LEAVE get\_detalles;

END IF;

set resultado = resultado + v\_costo;

END LOOP get\_detalles;

UPDATE informe\_servicio

SET

costo\_total = resultado

WHERE

id = i\_id;

CLOSE detalles\_cursor;

END$$

DELIMITER ;

-- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* DISPARADORES \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DROP trigger IF EXISTS `disp\_actualizar\_insert`;

DELIMITER $$

USE `arquitectura`$$

CREATE trigger disp\_actualizar\_insert after insert on trabajo

for each row

BEGIN

call proc\_actualizar\_detalle\_informe(new.detalle\_informe\_informe\_servicio\_id, new.detalle\_informe\_equipo\_id);

call proc\_actualizar\_informe(new.detalle\_informe\_informe\_servicio\_id);

END$$

DELIMITER ;

DROP trigger IF EXISTS `disp\_actualizar\_update`;

DELIMITER $$

USE `arquitectura`$$

CREATE trigger disp\_actualizar\_update after update on trabajo

for each row

BEGIN

call proc\_actualizar\_detalle\_informe(old.detalle\_informe\_informe\_servicio\_id, old.detalle\_informe\_equipo\_id);

call proc\_actualizar\_informe(old.detalle\_informe\_informe\_servicio\_id);

END$$

DELIMITER ;

DROP trigger IF EXISTS `disp\_actualizar\_delete`;

DELIMITER $$

USE `arquitectura`$$

CREATE trigger disp\_actualizar\_delete after delete on trabajo

for each row

BEGIN

call proc\_actualizar\_detalle\_informe(old.detalle\_informe\_informe\_servicio\_id, old.detalle\_informe\_equipo\_id);

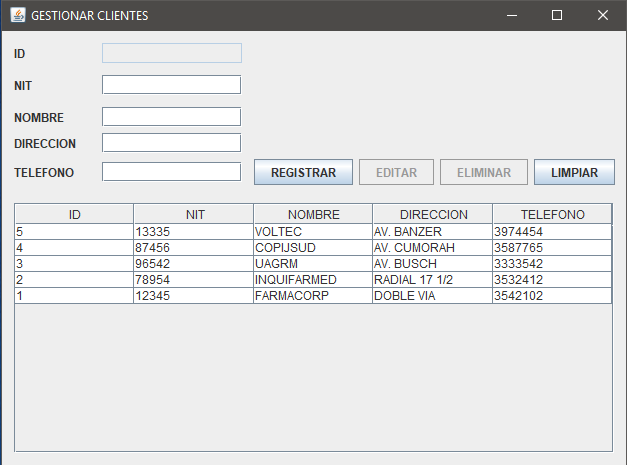
call proc\_actualizar\_informe(old.detalle\_informe\_informe\_servicio\_id);

END$$

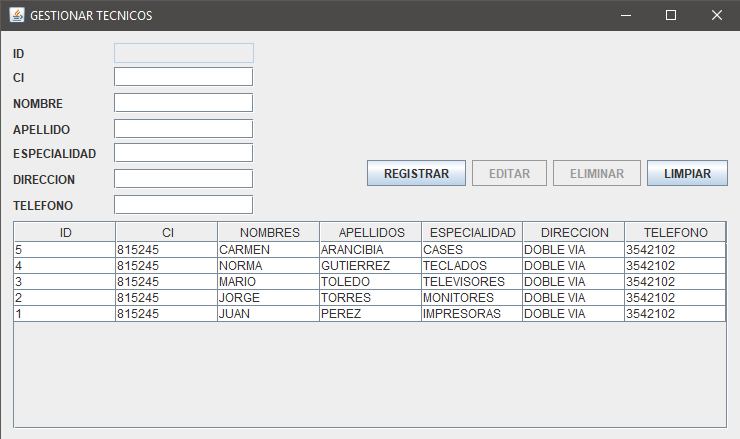
DELIMITER ;

### 4.3. Diseño de la interfaz

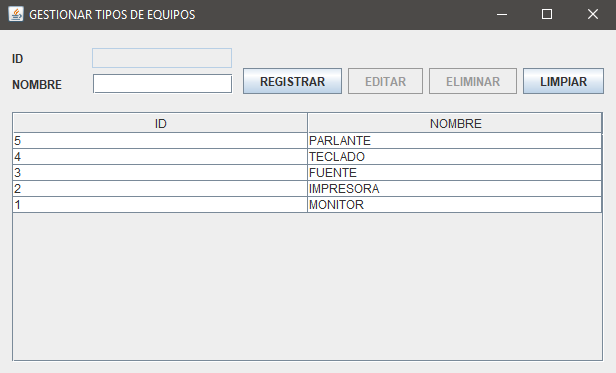
**Gestionar Clientes**

****

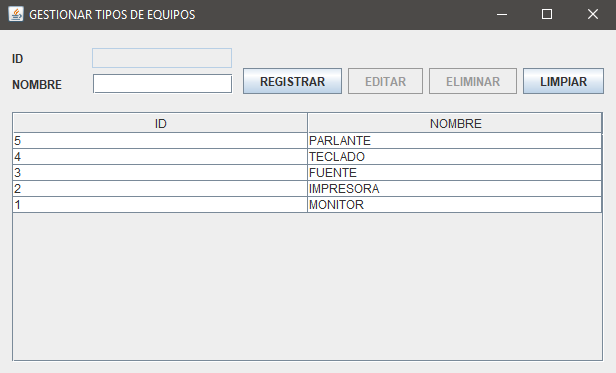
**Gestionar Técnicos**

****

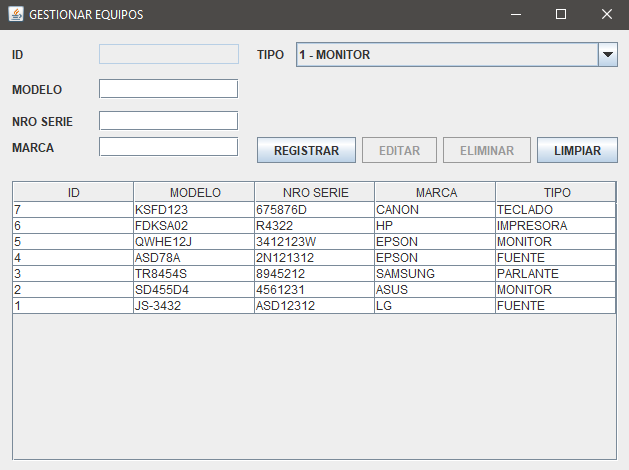
**Gestionar Tipos de Equipos**

****

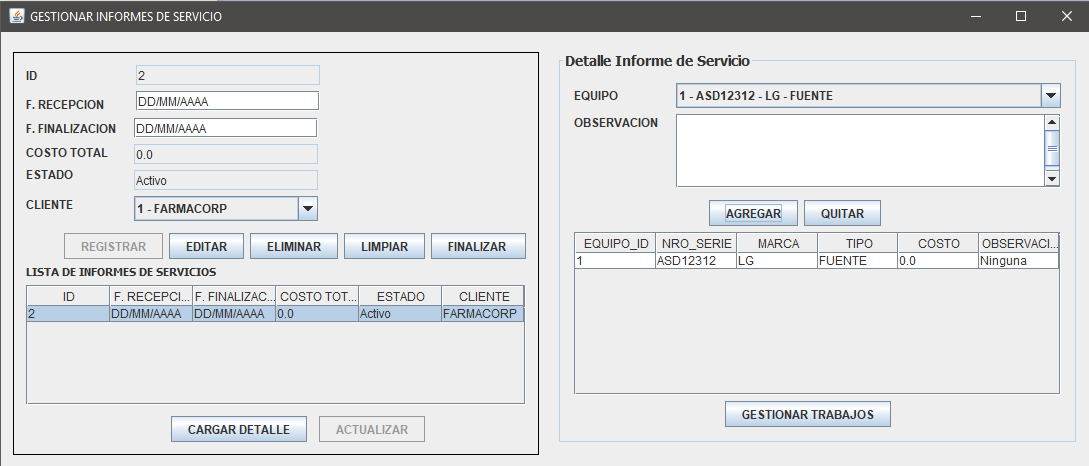
**Gestionar Herramientas**

****

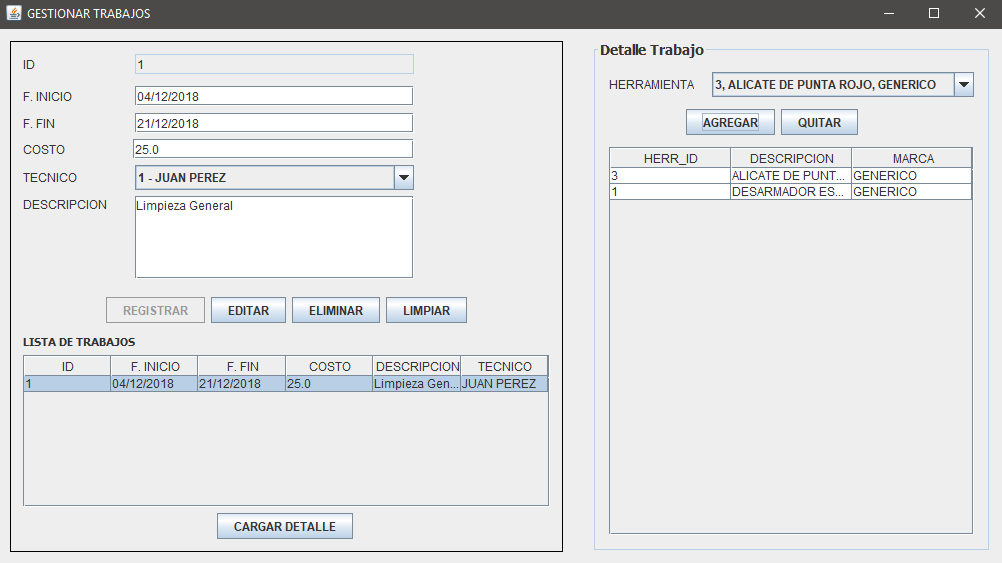
**Gestionar Equipos**

****

**Gestionar Informes de Servicio**

****

**Gestionar Trabajos**

****

### 4.4. Diseño procedimental

#### 4.4.1. Diagrama de clases dinámico y diagrama de secuencia

CU01. Gestionar Cliente





CU02. Gestionar Equipo





CU03. Gestionar Tipo de equipo





CU04. Gestionar Técnico





CU05. Gestionar Herramienta





CU06. Gestionar Trabajo









CU07. Gestionar Informe de Servicio











## 5. Flujo de trabajo: Implementación

#### 5.1. Diagrama de componentes

