



# **ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN**

**“ANÁLISIS, DISEÑO E IMPLEMENTACIÓN DE UN PORTAL WEB PARA  
ADMINISTRAR LA INFORMACIÓN DE LOS PROYECTOS DE  
ORGANIZACIONES SIN FINES DE LUCRO QUE SON REALIZADOS EN  
CONJUNTO CON EL MUNICIPIO DE GUAYAQUIL”**

INFORME DE

**PROYECTO DE GRADUACIÓN**

Previa a la obtención del Título de:

**INGENIERO EN COMPUTACIÓN**

**ESPECIALIZACIÓN SISTEMAS TECNOLÓGICOS**

Presentada por:

**MÓNICA MARLENE BAQUERIZO ANASTACIO**

**DARWIN ALBERTO CORONEL AILLA**

GUAYAQUIL - ECUADOR

2010

## **AGRADECIMIENTO**

A Dios, por abrigarnos bajo sus alas, por ser La Esperanza, El Castillo, Torre Fuerte, nuestra inyección de ánimos, por ayudarnos a terminar esta etapa de la vida. Nuestra carrera está en tus manos.

A nuestras familias, apoyo incondicional de nuestras vidas.

Al Municipio por abrirnos las puertas y servir con nuestro conocimiento a la sociedad.

A la ESPOL, a nuestros profesores, y en particular a la Ing. Anita Tapia, por dirigir este proyecto de graduación.

A nuestros amigos, que con sus consejos y ayuda, han contribuido a culminar nuestra carrera universitaria.

## DEDICATORIA

A mi Papi Vinicio, que con sus consejos siempre me supo guiar por el camino del bien.

A mi Mami Gladys, madre, amiga, consejera, la perseverancia la aprendí de ti.

Padres, este triunfo es de ustedes. Soy su fruto.

A mis hermanas Belén, Daniela, y mi Abuelita Flora, por su apoyo incondicional con quienes siempre puedo contar.

Darwin y Mimí, gracias por ser amigos fieles.

Ing. Cristina Abad, Ing. Carmen Vaca e Ing. Marcelo Loor, gracias por sus conocimientos impartidos.

**Mónica Baquerizo**

## **DEDICATORIA**

A mis Padres, Flavio y Mercy por el apoyo brindado durante toda mi vida.

A mis hermanos Andrés y Mirian, por sus valiosos consejos que me ayudaron a alcanzar la meta.

A mi hija Emily, fuente de inspiración para culminar mis estudios.

A mi esposa apoyo sincero y leal.

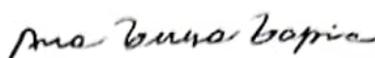
**Darwin Coronel Ailla**

## TRIBUNAL DE SUSTENTACIÓN



---

Ing. Jorge Aragundi  
PRESIDENTE



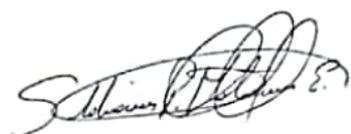
---

Ing. Ana Tapia  
DIRECTOR DE PROYECTO DE  
GRADUACIÓN



---

Ing. Marcelo Loor  
MIEMBRO DEL TRIBUNAL



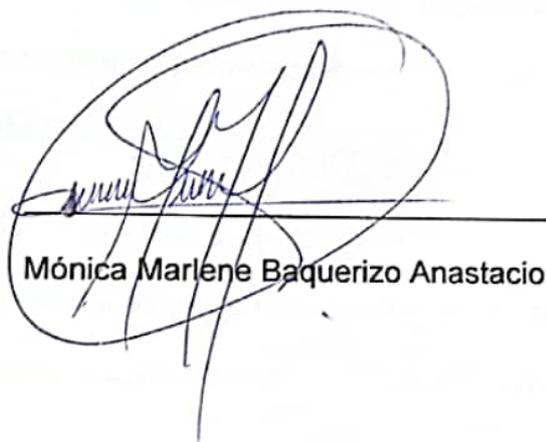
---

Ing. Soldiamar Matamoros  
MIEMBRO DEL TRIBUNAL

## DECLARACIÓN EXPRESA

"La responsabilidad del contenido de Proyecto de Graduación, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL".

(Reglamento de Graduación de la ESPOL)



Mónica Marlene Baquerizo Anastacio



Darwin Alberto Coronel Ailla

## **RESUMEN**

En el presente proyecto “Análisis, Diseño e Implementación de un portal web para administrar la información de los proyectos de organizaciones sin fines de lucro que son realizados en conjunto con el Municipio de Guayaquil” muestra el desarrollo de una aplicación web que será de utilidad para la M. I. Municipalidad de Guayaquil, como herramienta de apoyo para tomar decisiones con respecto a la canalización de los diferentes recursos asignados a las organizaciones y poder ejecutar proyectos más eficaces y eficientes.

El Capítulo 1 se denomina “Antecedentes y Justificación”, en el mismo se describe el problema actual, los objetivos y la justificación de este Proyecto de Graduación.

En el Capítulo 2 titulado “Análisis del Sistema”, se definen los requerimientos funcionales, no funcionales y la selección de las herramientas más adecuadas para el desarrollo del Proyecto.

El Capítulo 3 “Diseño”, describe el porqué y cómo se presentan los formularios, de los elementos utilizados, del diseño de la base de datos y las pruebas, que nos ayudan a saber que tan usable es el sistema.

En el Capítulo 4 “Implementación”, se describe cómo se ha implementado la seguridad, formularios dinámicos, actualización de datos y reportes. Así también las características de hardware y software mínimas que se necesita para que el sistema funcione a cabalidad.

En el Capítulo 5 “Pruebas”, se procede a respectivas pruebas que posibilitan la evaluación de la funcionalidad y usabilidad del prototipo implementado.

Finalmente, se presentan conclusiones y recomendaciones, se incluyen además los anexos empleados en el desarrollo de esta tesis.

# ÍNDICE GENERAL

RESUMEN.....	I
ÍNDICE GENERAL.....	III
ABREVIATURAS .....	V
ÍNDICE DE FIGURAS .....	VI
ÍNDICE DE ANEXOS .....	VII
INTRODUCCIÓN.....	VIII
CAPÍTULO 1 .....	10
ANTECEDENTES Y JUSTIFICACIÓN .....	10
1.1. Descripción del problema.....	10
1.2. Objetivos.....	13
1.2.1. Generales .....	13
1.2.2. Específicos .....	13
1.3. Alcance .....	14
1.4. Justificación .....	15
CAPÍTULO 2 .....	17
ANÁLISIS DEL SISTEMA.....	17
2.1. Requerimientos Funcionales .....	17
2.2. Requerimientos no funcionales .....	18
2.3. Análisis de alternativas y selección de las herramientas más adecuadas ...	19
CAPÍTULO 3 .....	24
DISEÑO .....	24
3.1. Diseño de Formularios.....	24
3.1.1. Identificación Institucional.....	28

3.1.2. Población .....	36
3.1.3. Línea de Acción.....	40
3.2 Diseño de la base de datos.....	41
3.3 Diseño uml.....	49
3.4 Diseño de Pruebas.....	54
CAPÍTULO 4 .....	57
IMPLEMENTACIÓN .....	57
4.1 Requerimientos del servidor .....	61
4.1.1 Hardware.....	61
4.1.2 Software .....	61
4.2 Requerimientos del cliente .....	62
4.2.1 Hardware.....	62
4.2.2 Software .....	62
CAPÍTULO 5 .....	63
PRUEBAS.....	63
5.1 Pruebas de Usabilidad .....	63
5.2 Métricas.....	64
5.3 Resultados obtenidos .....	68
CONCLUSIONES Y RECOMENDACIONES.....	70
ANEXOS .....	74
BIBLIOGRAFIA.....	136

## **ABREVIATURAS**

<b>RIOS</b>	Red de Información de Organizaciones Sociales
<b>ROUS</b>	Registro Único de Organizaciones de la Sociedad Civil
<b>GRUPO ETARIO</b>	Relativo o perteneciente a la edad de las personas.

## ÍNDICE DE FIGURAS

Figura 1.1.	Esquema de funcionamiento del sistema .....	11
Figura 3.1.	Formulario Registrar una organización.....	26
Figura 3.2.	Ejemplo de Ingreso de Contraseña .....	27
Figura 3.3.	Ingreso al sistema.....	27
Figura 3.1.	Formulario de Identificación Institucional. Paso1.....	31
Figura 3.2.	Formulario de Identificación Institucional. Paso2.....	32
Figura 3.3.	Formulario de Identificación Institucional. Paso3.....	33
Figura 3.4.	Formulario de Identificación Institucional. Paso4.....	34
Figura 3.5.	Formulario de Identificación Institucional. Paso5.....	35
Figura 3.1.	Formulario de Población. Paso1 .....	38
Figura 3.2.	Formulario de Población. Paso 2.....	38
Figura 3.3.	Formulario de Población. Paso3.....	39
Figura 3.4.	Formulario de Línea de Acción.....	40
Figura 5.1.	Usabilidad del Sistema .....	66
Figura 5.2.	Motivación del Sistema .....	66
Figura 5.3.	Retroalimentación del Sistema .....	67

## INDICE DE ANEXOS

ANEXO REQUISITOS DEL DESARROLLADOR .....	75
ANEXO DE ANÁLISIS DE ALTERNATIVAS DE LAS HERRAMIENTAS MÁS ADECUADAS.....	89
ANEXO DIAGRAMA DE CLASES .....	102
ANEXOS DE UML .....	108
ANEXO DE DIAGRAMAS DE BASE DE DATOS .....	125
ANEXO DISEÑO DETALLADO .....	131
ANEXOS DE LAS PRUEBAS .....	134

# INTRODUCCIÓN

En la actualidad, la tecnología acapara todas las áreas de la sociedad, haciendo que se renueve la manera en que se guardan los datos. Antes se guardaban en papeles escritos a mano, luego tuvo el apogeo la máquina de escribir manual, después la eléctrica para luego posicionarse la computadora, que almacena datos de manera digital. Todas las instituciones buscan evolucionar sus procesos de acuerdo al avance tecnológico que les facilita herramientas y les ayudan a mejorar su desempeño en diferentes áreas haciéndolas más eficientes y eficaces.

Durante años la M. I. Municipalidad de Guayaquil ha ayudado a organizaciones sin fines de lucro con el fin de disminuir los problemas sociales que aquejan a la ciudadanía. Con el pasar del tiempo estas organizaciones han crecido y con ellas la cantidad de información asociada a los servicios que prestan, comunidades que atienden, e información básica institucional, que no se sabe si está actualizada. La información que se tiene de las organizaciones está en diferentes formatos, haciendo complicada la manipulación de la misma. La tecnología permite mejorar los procesos de

almacenamiento de información mediante herramientas computacionales que ayudan a manipularla de acuerdo a las necesidades del cliente.

La meta de este proyecto de tesis es implementar una herramienta de trabajo para que el M.I. Municipalidad de Guayaquil pueda tener información actualizada y mediante los reportes tener una mejor visión de cómo están trabajando las organizaciones, a qué se dedican, cuántas personas atienden, cual es el enfoque que tienen y de las estrategias que se plantean para poder llegar a los ciudadanos beneficiarios.

Por otra parte, buscamos que este proyecto ayude a canalizar los recursos que la M. I. Municipalidad asigna a las organizaciones aumentando la eficiencia en la ejecución de los proyectos.

# **CAPÍTULO 1**

## **ANTECEDENTES Y JUSTIFICACIÓN**

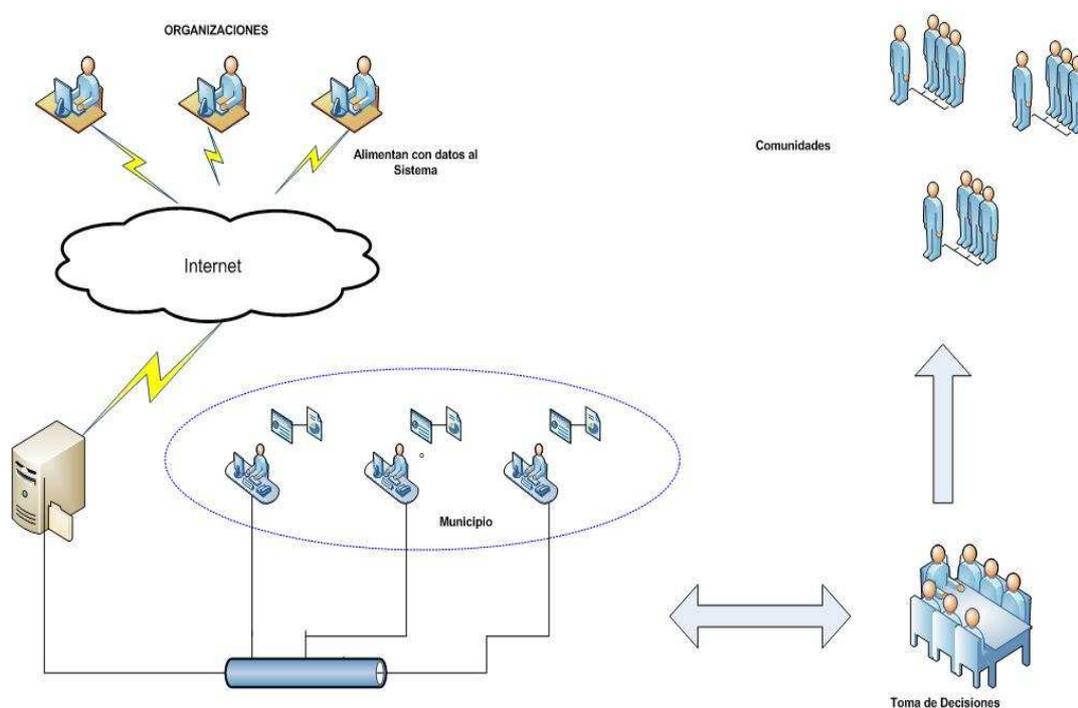
### **1.1. Descripción del problema**

En la ciudad de Guayaquil existen varios problemas sociales, como la pobreza, maltrato infantil, falta de educación en ciertos sectores de la urbe, entre otros. Existen organizaciones que buscan contrarrestar estos problemas como Fundaciones, albergues, iglesias las cuales realizan actividades con el fin de ayudar a los sectores más afectados.

La Muy Ilustre Municipalidad de Guayaquil, empeñada en impulsar el desarrollo integral del cantón, resolvió fortalecer áreas sociales, productivas y de servicios para la cual constituyó la Dirección de Acción Social y Educación – DASE – que contribuye al desarrollo humano y social de la ciudad. Para esto, el Municipio convoca a las organizaciones: no gubernamentales, organismos del sector público, grupos comunitarios, universidades, instituciones, empresas

privadas, eclesiásticas, colegios de profesionales, organizaciones de voluntariado, y cada una, desde su experticia aporta al desarrollo social del cantón a través de un trabajo concertado, articulado, coordinado y complementario.

La información que el Municipio de Guayaquil tiene de las organizaciones está fragmentada e incompleta y en diversos formatos tanto digital como físico, lo cual hace difícil obtener información técnica y gerencial sobre los servicios, proyectos que ejecutan, población que se está atendiendo, entre otros, que permita planificar acciones que tengan mayor cobertura e impacto.



**Figura 1.1. Esquema de funcionamiento del sistema**

La Muy Ilustre Municipalidad de Guayaquil tiene grupos de personas que están involucradas en ayudar a ciertos sectores con problemas sociales, a los cuales se las denomina mesas de concertación, donde cada mesa trabaja en una de las siguientes áreas:

- Infancia
- Jóvenes
- Adultos Mayores
- Discapacidades
- Género, con énfasis en violencia intrafamiliar
- Nutrición

En el esquema presentado, se observa la forma de cómo el sistema recolectará información de las organizaciones, que llegará a un servidor para ser consultada por las mesas de concertación, tomar decisiones y ayudar a las comunidades.

A continuación, algunos de los problemas que se presentan:

- La información de algunas de las organizaciones está documentada físicamente.
- Existe duplicación y desactualización de la información. No se tiene información clara, concisa y disponible.
- Es difícil tomar decisiones para que las organizaciones brinden ayuda a una comunidad que no está siendo atendida.

- Desconocimiento de la cantidad de proyectos que se ejecutan en un área de desarrollo, de los servicios que presta cada organización para el proyecto, cantidad de personas beneficiarias y de la infraestructura física disponible.

## 1.2. **Objetivos**

### 1.2.1. **Generales**

- Realizar un sistema que ayude a organizar la información de las organizaciones inscritas en las Mesas de Concertación.
- Ayudar a la toma de decisiones que el Municipio requiere para llevar a cabo un proyecto benéfico.

### 1.2.2. **Específicos**

- Diseñar formularios para levantar y procesar datos organizacionales sobre: información general, servicios y cobertura de las organizaciones que participan en las áreas de proyectos relacionadas al Municipio.
- Presentar información, relacionada a las áreas de desarrollo, género y tipo de ayuda del Municipio, que facilite a la toma de decisiones y a la canalización de recursos económicos, de infraestructura, logísticos y de capacitación al momento de ejecutar un proyecto.

- Facilitar la actualización de información de las organizaciones.

### 1.3. Alcance

El sistema que requiere la M.I Municipalidad de Guayaquil, es extenso, por lo cual se ha definido la implementación de un sistema web creado a partir de módulos. A continuación el detalle.

- ✓ Página principal.- página informativa del sistema y mesas de concertación.
- ✓ Ingreso de usuarios al sistema

El sistema web, tendrá diferentes opciones que serán acordes al perfil del usuario: organizaciones y mesas de concertación.

Los módulos para las organizaciones son:

- Identificación Institucional
  - Datos generales de las organizaciones
  - Participación en Mesa(s) de Concertación
- Población
  - Sexo con el que trabaja
  - Grupo Etario al cual está dirigido
  - Cobertura
- Línea de Acción y Servicios

- Línea de acción
- Servicios que prestan

Las organizaciones podrán realizar las siguientes acciones:

- Ingresar, modificar y consultar el formulario solicitado que contiene: datos generales, participación en la(s) mesa(s) de concertación, sexo, grupo etario, cobertura, línea de acción y servicio al que atiende.

Las mesas de concertación tendrán acceso a diversos reportes que contestan las siguientes interrogantes:

- ¿Cuántas y cuáles organizaciones prestan servicios en algún área de desarrollo?.
- ¿Cuántas y cuáles organizaciones tienen una línea de acción, con cierto servicio específico?.
- ¿Cuántas y cuáles organizaciones ayudan según su grupo etario?.
- ¿Cuántas personas beneficiarias hubo por organización?.
- ¿Cuáles organizaciones están asociadas a una mesa de concertación específica?.

#### **1.4. Justificación**

Debido a que la información de las organizaciones está físicamente y digitalmente documentada, para la Muy Ilustre Municipalidad de

Guayaquil es difícil tener datos veraces e inmediatos de las diferentes Mesas de Concertación.

Lo anterior, ha provocado que los reportes que se obtienen de las Mesas de Concertación sean erróneos, ya que existe duplicación y desactualización de datos, y una ardua tarea al momento de tomar decisiones.

Debido a esto, se ha decidido realizar el Sistema de Red de Información de Organizaciones Sociales (RIOS). Este sistema facilita la administración de la información de forma eficiente y con esto mejorar la toma de decisiones.

## **CAPÍTULO 2**

### **ANÁLISIS DEL SISTEMA**

#### **2.1. Requerimientos Funcionales**

Definen el comportamiento interno del software como: detalles técnicos, manipulación de datos, funcionalidades específicas que definen el comportamiento del sistema. Entre los requerimientos generales tenemos:

1. Una organización que no existe en el portal, podrá: registrarse, y una vez registrada podrá actualizar y consultar su información.
2. Una organización podrá registrarse, si llenan los campos requeridos en el formulario de ingreso.
3. Los reportes generados se podrán exportar a formatos portables.
4. Las entradas serán validadas según lo indicado por el usuario.
5. Se emitirán mensajes de error, advertencia y confirmación que

retroalimenten de la mejor forma, las acciones realizadas por el usuario.

Cada uno de los requerimientos se detalla en el anexo "REQUISITOS DEL DESARROLLADOR". El detalle de los requerimientos incluye: **código**: que están enumerados RD-0000, **nombre del requisito**: nombre representativo a la funcionalidad, **descripción**: indica la funcionalidad, excepciones y comportamiento propiamente dicho, **dependencias**: referencias a otros requisitos, **observaciones**: campos de entrada al sistema, **tabla de campos**: tipo de datos que se utilizan con sus respectivos nombres, identificador y longitud.

## 2.2. Requerimientos no funcionales

Son aspectos del sistema visibles para el usuario, que no están relacionados de forma directa con el comportamiento funcional del sistema. Entre estos tenemos:

- Se podrá acceder al sistema mediante cualquier navegador de Internet.
- El sistema deberá ser de fácil uso para las Mesas de Concertación y las Organizaciones.
- Después de la capacitación a los líderes de las Mesas de

Concertación, les deberá ser posible utilizar todas las funciones del sistema.

### **2.3. Análisis de alternativas y selección de las herramientas más adecuadas**

Decidir la plataforma más estable para que se ejecute un sistema es crucial, ya que de éste dependen factores como disponibilidad, seguridad e integridad de datos. Se analizará el sistema operativo, motor de base de datos, lenguaje de programación para que el Sistema de Red de Información de Organizaciones Sociales (RIOS) a desarrollar permita el almacenamiento y manipulación de información, en cualquier momento, que sea correcta desde el punto de vista lógico y que solo pueda ser accedida por las personas adecuadas.

Las herramientas a analizar son:

#### **Sistema operativo**

- Linux
- Windows 2008 Server

#### **Motor de Bases de Datos**

- MySql
- SQLSERVER

### **Lenguaje de Programación**

- PHP
- ASP
- JSP

En el anexo de “ANÁLISIS DE ALTERNATIVAS DE LAS HERRAMIENTAS MÁS ADECUADAS” se detalla ampliamente los beneficios y contras de cada una de las herramientas.

A continuación una tabla comparativa de las distintas combinaciones de plataformas, motor de datos y scripts de programación. Se ha considerado una calificación cuyo valor va del 1 al 5, donde 5 es la mejor calificación, estos valores son asignados a la velocidad de ejecución.

<b>Combinación</b>	<b>Costo Referencial</b>	<b>Multiplataforma</b>	<b>Velocidad de Ejecución</b>
<b>Windows+Microsoft SQL server+ASP</b>	\$5058.93	No	3
<b>Windows+ MYSQL + ASP</b>	\$3883.93	No	4
<b>Windows+ MYSQL + PHP/ JSP</b>	\$800	Si	4
<b>Linux+ MYSQL + PHP</b>	\$0	Si	5
<b>Linux+ MYSQL + JSP</b>	\$0	Si	5

**Tabla 2.1. Comparación de Plataformas**

En el análisis detallado anteriormente observamos que el costo referencial de la combinación de tecnologías cuyos valores son cero, es porque son herramientas libres y gratuitas. Existe un valor agregado que es el tiempo empleado en el aprendizaje de las herramientas, en este caso también es cero ya que ambos programadores tienen experticia en ellas.

### **Elección de las herramientas**

Después del análisis realizado, hemos elegido la combinación que nos brinda una multiplataforma y lenguaje libre: Linux, PHP y Mysql. El proyecto que vamos a realizar hace a Mysql una buena elección, debido a que no depende de una compañía específica para dar soporte técnico, sino que las apoya un número importante de desarrolladores que colaboran con plugins, frameworks alrededor de todo el mundo, y no se está forzado a pagar actualizaciones anuales.

Por motivos de experticia de los programadores elegimos PHP, ya que es orientado a objetos y desarrollaremos utilizando un modelo MVC lo cual hace al sistema escalable y de fácil mantenimiento.

Vemos que Mysql se la puede configurar para que dé un alto rendimiento en las consultas que el sistema necesita y que PHP es un muy buen lenguaje de interpretación.

Para las pruebas se va a utilizar Windows XP, debido a que en la mayoría de las computadoras móviles, es el sistema por defecto.

El costo de las herramientas utilizadas es \$0, lo cual beneficia al Municipio de Guayaquil, ya que no tienen que realizar ningún gasto en herramientas.

De forma similar, las personas de la Organización que brindarán servicios de soporte están familiarizados con esta selección.

## **CAPÍTULO 3**

### **DISEÑO**

#### **3.1. Diseño de Formularios**

Para que las organizaciones ingresen sus datos fácilmente, se ha analizado la forma en que se presentarán los formularios, de tal manera de que sea comprensible, organizada, interactiva y fácil para los usuarios del sistema.

Los elementos que presentarán los formularios son:

- Cajas de texto, cuando se incluye una redacción corta de hasta 50 caracteres.
- Áreas de Texto, cuando se incluye una redacción larga superior a 50 caracteres.
- Opción de Selección Unitaria (Radio buttons), para elegir entre dos opciones.

- Opción de Selección Múltiple (Checkbox), cuando el usuario elije una o varias opciones.
- Listas (listbox), cuando el usuario elije solo una opción de las listadas.
- Botones, para ingresar el formulario, agregar o eliminar un campo de entrada. Se permitirá aumentar o disminuir los campos solicitados, hasta el límite establecido en los requerimientos. Esto le da flexibilidad e interacción al sistema. Los detalles están en el anexo “REQUISITOS DEL DESARROLLADOR”.

Para ingresar al sistema el usuario se debe registrar. Para el registro de la organización se ha diseñado el siguiente formulario:

## Registrar Organización

Nombre de la Organización:

Representante Legal:

RUC:

Correo Electrónico:

Dirección:

Usuario:

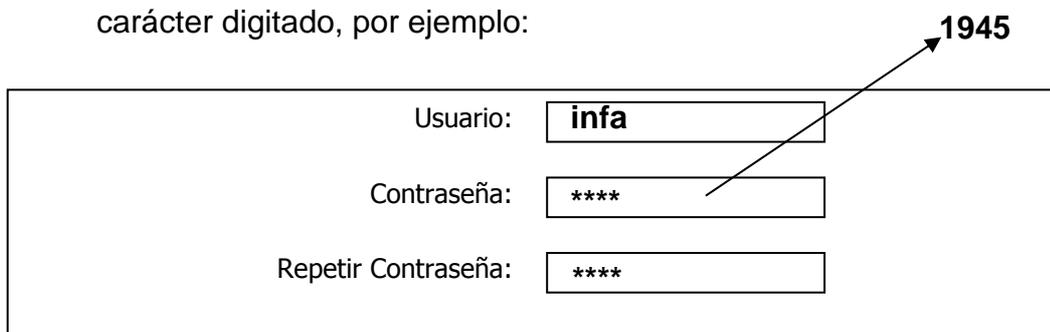
Contraseña:

Repetir Contraseña:

**Figura 3.1. Formulario Registrar una organización**

En este formulario observamos que existen tres tamaños diferentes de cajas de texto para que el usuario pueda observar lo que digita. Los campos pequeños son aquellos que tienen una entrada máxima de 10 caracteres. Los medianos hasta 50 y los grandes hasta 100 caracteres.

Por motivos de seguridad los campos contraseña y repetir contraseña son de tipo “password”, es decir que no será visualizada al momento de digitarla, se observarán un asterisco por cada carácter digitado, por ejemplo:



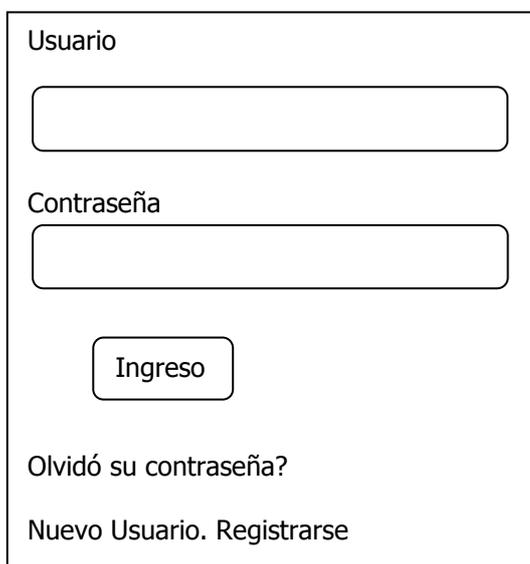
El formulario muestra tres campos de entrada:

- Usuario:
- Contraseña:
- Repetir Contraseña:

Una flecha apunta desde el campo de contraseña hacia el número 1945.

**Figura 3.2. Ejemplo de Ingreso de Contraseña**

Una vez que se ha registrado, el usuario procede a ingresar al sistema por medio del siguiente esquema:



El esquema de ingreso al sistema incluye:

- Etiqueta: Usuario
- Campo de entrada para el usuario.
- Etiqueta: Contraseña
- Campo de entrada para la contraseña.
- Botón: Ingreso
- Enlaces: Olvidó su contraseña? y Nuevo Usuario. Registrarse

**Figura 3.3. Ingreso al sistema**

Las cajas de texto son amplias, y se muestran dos opciones en caso de pérdida de contraseña o si es un usuario que desea realizar un nuevo registro.

Toda organización para ingresar sus datos debe llenar tres tipos de formularios:

- Identificación Institucional
- Población
- Servicios

Los formularios se han dividido en pasos interactivos y organizados.

En caso que el usuario quiera retroceder para cambiar algún dato, lo podrá realizar en cualquier momento utilizando los botones “Atrás” y “Siguiente” realizando los cambios que considere antes de realizar el ingreso definitivo pulsando el botón Aceptar.

### **3.1.1. Identificación Institucional**

Este formulario se ha dividido en cinco pasos. Los elementos que se han utilizado dependen de la información que se solicita. La información se alinea a la derecha, para que se vea mejor organizado y tenga una presentación más agradable.

Paso 1: Datos básicos, se han elegido las cajas de texto, que tienen un tamaño apropiado para cuando el usuario que escribe, observe lo que digita.

En el campo ROUS, se utiliza la opción de selección unitaria, porque el usuario solo puede elegir una de las dos opciones, evitando errores al momento de seleccionar.

Paso 2: Contactos, en esta sección existen cajas de textos para solicitar sitio web y teléfono(s). En el campo teléfono se le da la flexibilidad al usuario de ingresar varios teléfonos hasta el número permitido en los parámetros de requerimientos. Los detalles están en el anexo “REQUISITOS DEL DESARROLLADOR RD-003”.

Paso 3: Estrategia, se utilizan áreas de texto, debido a que la información solicitada es más amplia y necesita que tenga una mejor visualización de lo que se escribe. Los campos solicitados son: misión, visión y objetivos. En el campo de objetivos, se le da la flexibilidad al usuario para que pueda elegir hasta el número permitido en los parámetros de requerimientos. Los detalles están en el anexo “REQUISITOS DEL DESARROLLADOR RD-003”.

Paso 4: Ubicación, en esta sección se solicitan los datos de la ubicación de matriz y sucursales de la organización. Se utilizan listas, para elegir el área de desarrollo en la cual se encuentran ubicadas. Obligatoriamente solo se elije una opción, disminuyendo el porcentaje de error en el formulario. Cuando se pase el mouse sobre el formulario, se le dará una pequeña información al usuario sobre el tema que va a llenar. En el campo sucursal, se le da la flexibilidad al usuario para que pueda elegir hasta el número permitido en los parámetros de requerimientos. Los detalles están en el anexo “REQUISITOS DEL DESARROLLADOR RD-003”.

Paso 5: Mesas Asociadas, una organización puede estar asociada a una o varias mesas de concertación, es por este motivo que al usuario se le da la flexibilidad de aumentar o disminuir el número de mesas a las que esté relacionada. Los datos de entrada son la asociación a la mesa y el contacto de la organización por cada mesa que adjunte. El máximo de asociaciones que el usuario puede aumentar está dado por el número de mesas de concertación. Los detalles están en el anexo “REQUISITOS DEL DESARROLLADOR RD-003”.

Paso 1	Paso 2	Paso 3	Paso 4	Paso 5
Datos Básicos	Contactos	Estrategia	Ubicación	Mesas Asociadas

Código Predial:

Código de Censo:

Casilla Postal:

RUP:

ROUS: SI NO

Figura 3.1. Formulario de Identificación Institucional. Paso1

Paso 1	Paso 2	Paso 3	Paso 4	Paso 5
Datos Básicos	Contactos	Estrategia	Ubicación	Mesas Asociadas

Sitio Web:

Teléfono:

Figura 3.2. Formulario de Identificación Institucional. Paso2

Paso 1	Paso 2	Paso 3	Paso 4	Paso 5
Datos Básicos	Contactos	Estrategia	Ubicación	Mesas Asociadas

Misión:

Visión:

Objetivos:  +  -

Figura 3.3. Formulario de Identificación Institucional. Paso3

Paso 1	Paso 2	Paso 3	Paso 4	Paso 5
Datos Básicos	Contactos	Estrategia	Ubicación	Mesas Asociadas

Ubicación de Matriz:

Sucursales:

Figura 3.4. Formulario de Identificación Institucional. Paso4

Paso 1	Paso 2	Paso 3	Paso 4	Paso 5
Datos Básicos	Contactos	Estrategia	Ubicación	Mesas Asociadas

Mesa Asociada:

**Datos del Delegado**

Nombre:

Cargo:

Correo Electrónico:

Teléfono Convencional:

Teléfono Celular:

Dirección:

Figura 3.5. Formulario de Identificación Institucional. Paso5

### 3.1.2. Población

El formulario de población, está dividido en tres pasos: sexo, grupo etario y cobertura institucional.

Paso 1: sexo, se ha decidido utilizar opciones de selección múltiple, para que el usuario pueda elegir entre las opciones: masculino, femenino o ambas.

Paso 2: grupo etario, también tenemos opciones de selección múltiple para elegir, esto aumenta la amigabilidad del sistema. Las opciones que se muestran dependen de la información que se encuentra en la base de datos. Actualmente las opciones son: niños, adolescentes, jóvenes, adultos y adultos mayores.

Paso 3: cobertura institucional, hemos decidido organizar la información de manera horizontal, para que el usuario pueda tener una legibilidad más cómoda, y si el requiere aumentar filas, lo pueda realizar fácilmente. En esta sección llenará: grupo etario, sexo, número de beneficiarios y áreas de desarrollo. Los elementos a utilizar son las listas y solo en el número de beneficiarios se utiliza campo de texto.

Cuando el usuario pase el ratón sobre el campo área de desarrollo, se le mostrará una pequeña descripción de lo q se

trata las áreas de desarrollo. También habrá un hipervínculo que mostrará un gráfico, del mapa de Guayaquil con sus respectivas áreas de desarrollo para que pueda ubicarlas fácilmente.

Las filas se aumentarán hasta el número permitido en los parámetros de requerimientos. Los detalles están en el anexo "REQUISITOS DEL DESARROLLADOR RD-004".

A continuación los formularios.

Paso 1	Paso 2	Paso 3
Sexo	Grupo Etario	Cobertura Institucional

Masculino

Femenino

Siguiente

Figura 3.1. Formulario de Población. Paso 1

Paso 1	Paso 2	Paso 3
Sexo	Grupo Etario	Cobertura Institucional

Niños

Adolescentes

Jóvenes

Adultos

Atrás

Siguiente

Figura 3.2. Formulario de Población. Paso 2

Paso 1	Paso 2	Paso 3
Sexo	Grupo Etario	Cobertura Institucional

<b>Grupo Etario</b>	<b>Sexo</b>	<b># Beneficiarios</b>	<b>Área de Desarrollo</b>
Elegir una opción 	Elegir una opción 	<input type="text"/>	Elegir una opción 

Figura 3.3. Formulario de Población. Paso3

### 3.1.3. Línea de Acción

Este formulario solo tiene una sección. Se han utilizado dos listas para que el usuario solo pueda elegir una opción entre la lista. Los campos solicitados son línea de acción y servicio.

A pesar de ser una única sección, tiene el mismo formato que las anteriores y lleva un título que le indica al usuario en que formulario se encuentra.

El usuario puede elegir más líneas de acciones y servicios hasta el número permitido en los parámetros de requerimientos. Los detalles están en el anexo "REQUISITOS DEL DESARROLLADOR RD-005".

A continuación el formulario.

El formulario, titulado "Línea de Acción", contiene los siguientes elementos:

- Botones de "+" y "-" para controlar el número de líneas.
- Un campo de selección etiquetado "Línea de Acción" con el texto "Elegir una opción".
- Un campo de selección etiquetado "Servicio" con el texto "Elegir una opción".
- Un botón "Registrar" ubicado al final del formulario.

Figura 3.4. Formulario de Línea de Acción.

## 3.2 Diseño de la base de datos

Para el desarrollo de este proyecto de graduación, se ha analizado el diseño de base de datos para que funcionen los tres módulos que se van a desarrollar, considerando que el sistema se puede expandir.

En esta sección se analizarán las tablas más relevantes del sistema.

Los tipos de tablas que existen en la base de datos son:

- Tablas de Seguridad.
- Tablas de la Organización.
- Tablas de información básica.

### **Tablas de Seguridad**

Las tablas de seguridad son aquellas que guardan el tipo de usuario y los usuarios que van a ingresar al sistema. Las tablas son: tipo\_usuario y usuario.

**Tabla tipo\_usuario:** tiene el tipo de usuario que va a ingresar al sistema. Actualmente existen tres tipos de usuarios: administrador, mesas de concertación y organizaciones. En caso de que se desee un tipo de usuario nuevo, se tendría que crear en esta tabla. El id que se genere, es un FK en la tabla usuario.

**Tabla usuario.-** tiene a los usuarios propiamente dichos. El campo password está encriptado, de esta manera protegemos el acceso al sistema.

### **Tablas de la Organización**

Son las que están relacionadas directamente con la información de cada organización. Existen cuatro tablas que guardan un historial de la información. Cuando se guarda un historial los registros no se modifican ni se eliminan, sino que se insertan. Entre las tablas más relevantes estas tenemos: `org_datos_basicos` , `orgsexo`, `orgcobertura`, `orgcobertura`, `org_linea_servicios`.

**Tabla `org_datos_basicos`:** se concentra los datos básicos de la organización incluido el RUC. El id que se genera en esta tabla es autoincremental.

**Tabla `orgsexo`:** contiene los ids de los sexos que la organización atiende. Esta tabla guarda un historial, es decir que en una actualización, los registros se insertan.

**Tabla `orgcobertura`:** se almacena los ids de el grupo etario, sexo, área de desarrollo y cantidad de personas atendidas. Esta tabla guarda un historial.

**Tabla `org_linea_servicios`:** se almacena los ids de los servicios que presta la organización a la comunidad. Esta tabla guarda un historial.

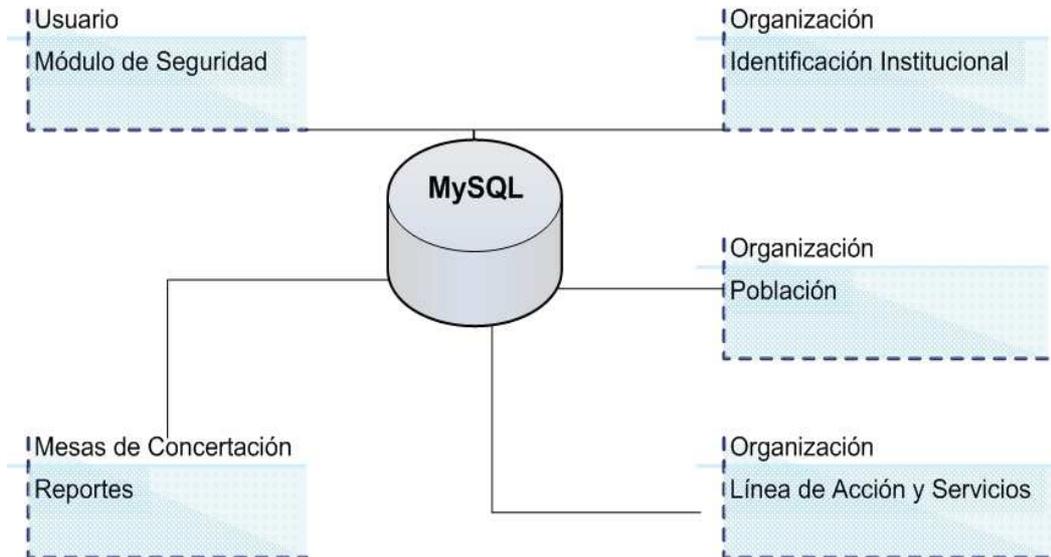
**Tabla `org_grupo_etareo`:** se almacena los ids de el grupo etario atendido. Esta tabla guarda un historial.

**Tabla contacto:** se almacena el contacto a la mesa asociada. Los datos que se almacenan son: nombre, dirección, correo electrónico, teléfono.



## ARQUITECTURA

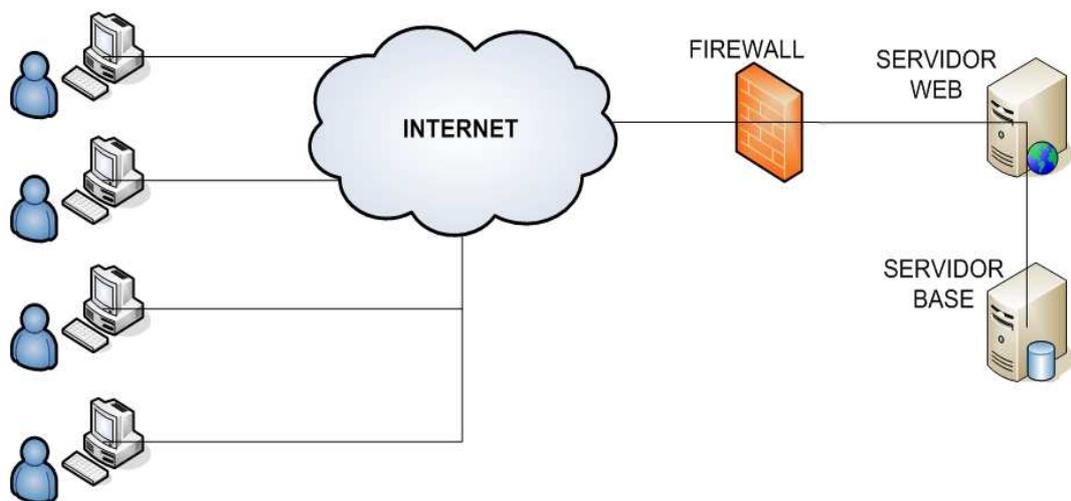
### DIAGRAMA DE REPOSITORIO



### MAPEO HARDWARE – SOFTWARE

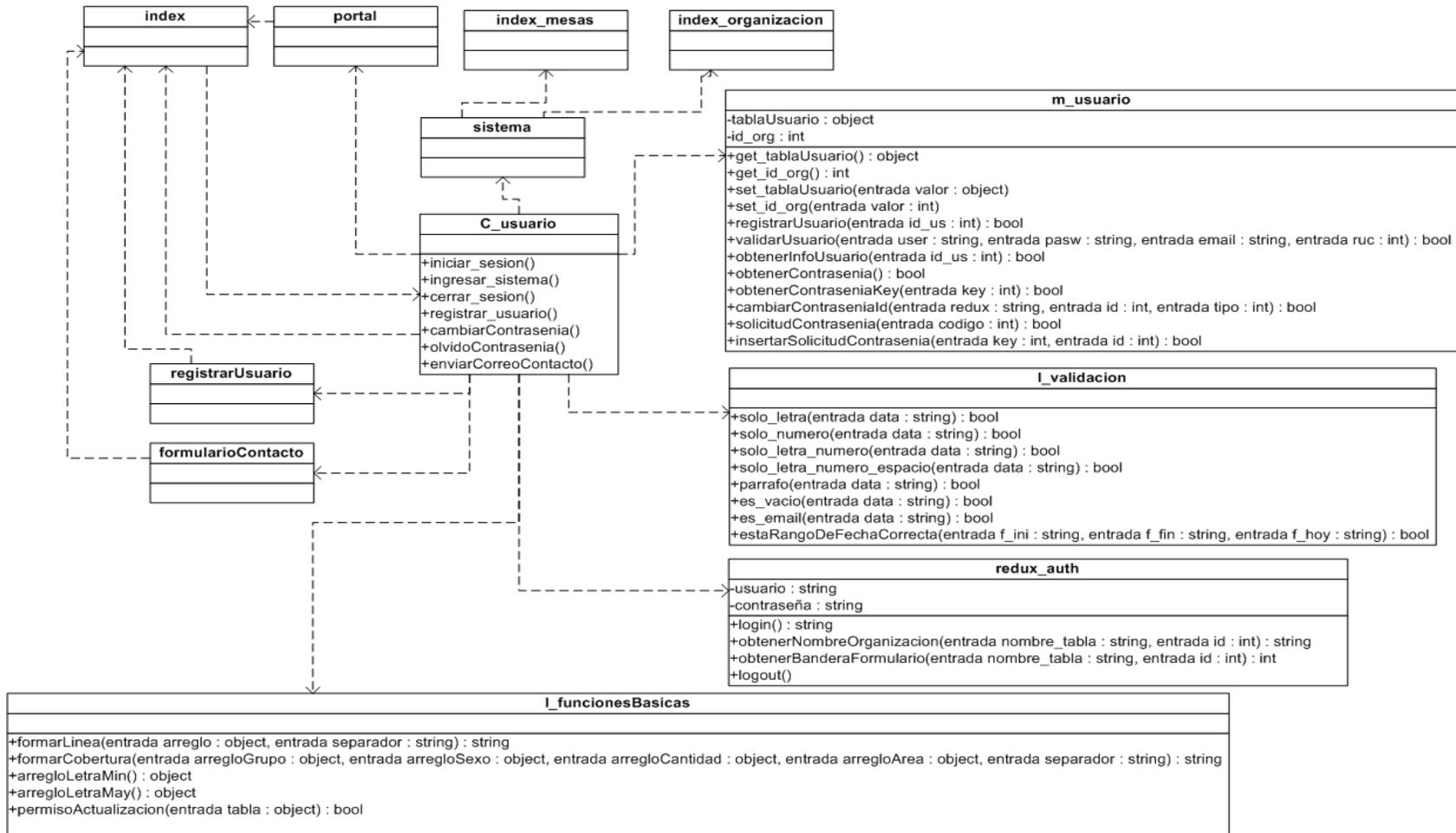
#### ORGANIZACIONES

#### MUNICIPIO DE GUAYAQUIL

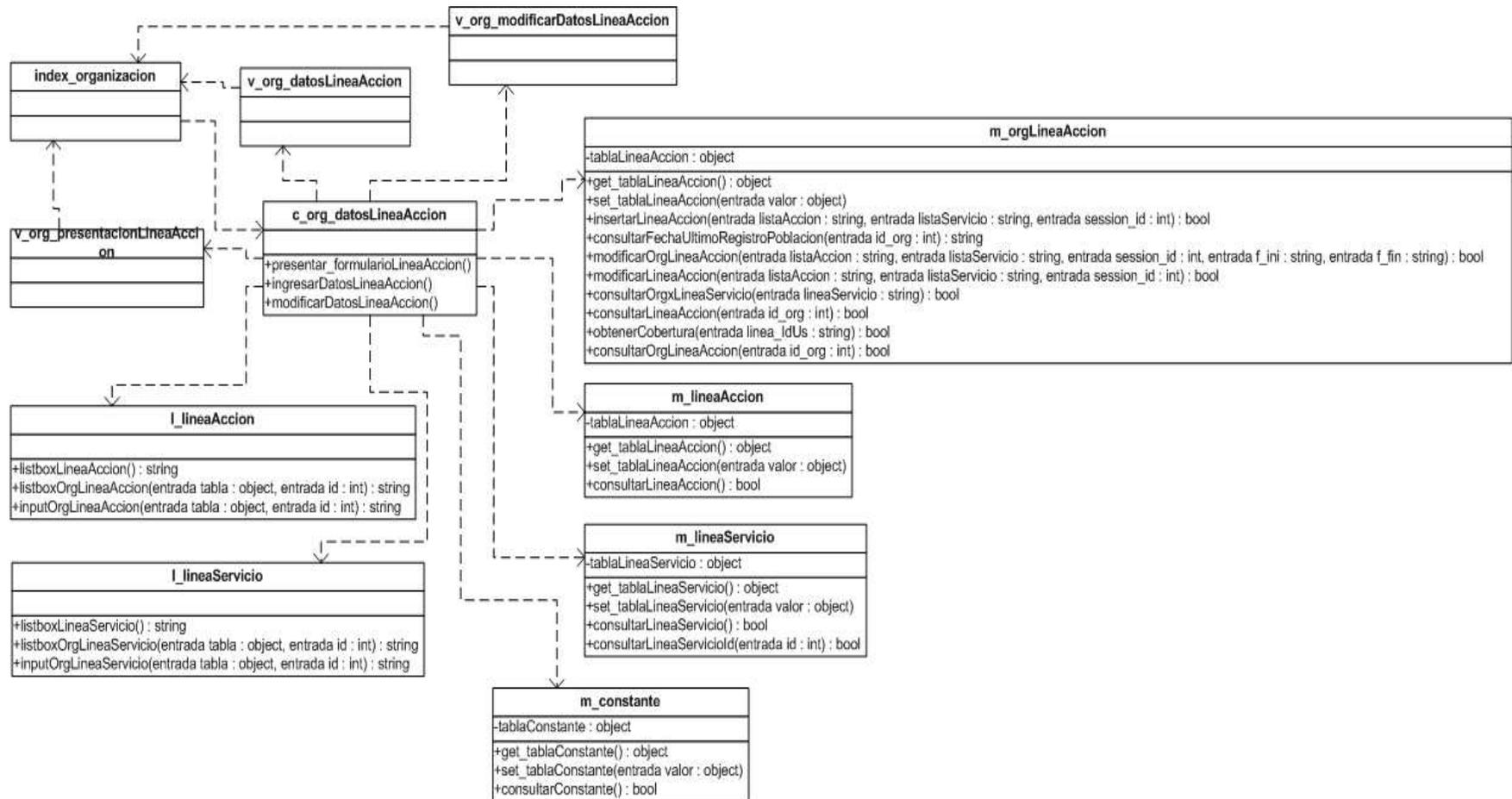


## DIAGRAMA DE CLASES

### CONTROLADOR DE SEGURIDAD



## CONTROLADOR LINEA DE ACCION



**CONTROL DE ACCESO**

<b>Usuario</b>	<b>Funcionalidades a las que tiene acceso</b>
<b>Organización</b>	<ul style="list-style-type: none"> <li>➤ Ingreso de registro al sistema</li> <li>➤ Ingreso de datos básicos.</li> <li>➤ Ingreso de datos de población.</li> <li>➤ Ingreso de datos de línea de acción.</li> <li>➤ Modificación de contraseña.</li> <li>➤ Actualización de datos básicos.</li> <li>➤ Actualización de datos de población.</li> <li>➤ Actualización de datos de línea de acción.</li> </ul>
<b>Mesas de concertación</b>	<ul style="list-style-type: none"> <li>➤ Consulta de organizaciones por área de desarrollo.</li> <li>➤ Consulta de los servicios que tiene las organizaciones.</li> <li>➤ Consulta de organizaciones por grupo etario.</li> <li>➤ Consulta cuantas personas beneficiarias hubo por organización.</li> <li>➤ Consultas cuántas organizaciones están asociadas con las mesas de concertación</li> </ul>

**CONTROL GLOBAL**

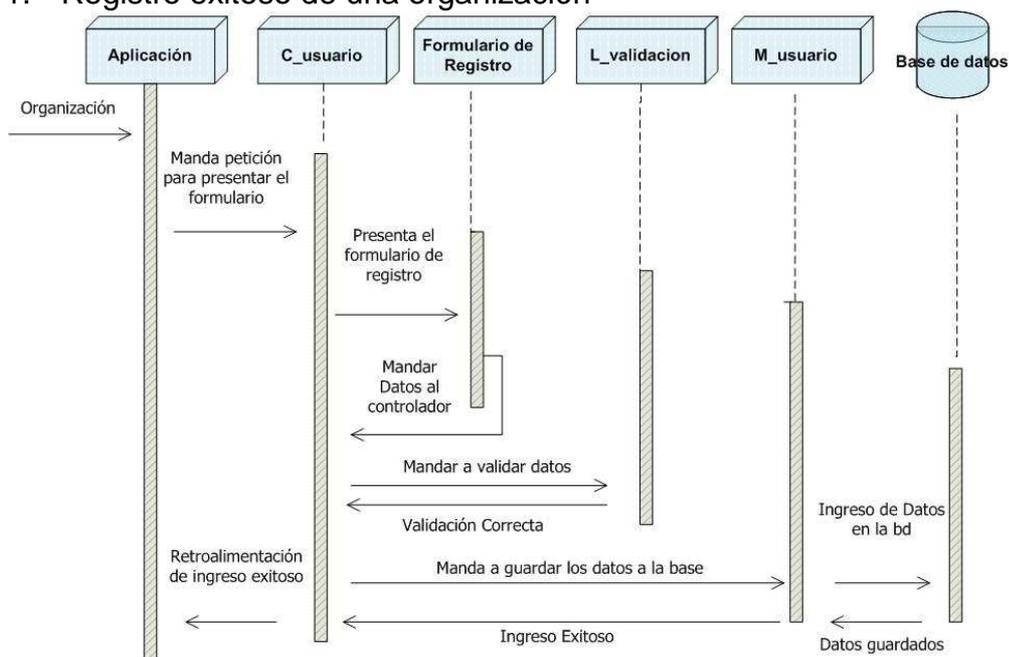
<b>Evento</b>	<b>Descripción</b>
<b>Ingreso al sistema</b>	Si la Organización no se ha registrado previamente no podrá acceder al sistema.
<b>Actualización</b>	Para que las Organizaciones puedan actualizar sus datos, debe de estar su información ingresada en la base de datos. También debe estar habilitada esta opción por el sistema.
<b>Registro de población</b>	La información de sexo y grupo etario debe estar en la base de datos ingresada previamente, porque si no es así el sistema no podrá mostrar dicha información al usuario para que pueda seleccionarla.

### 3.3 Diseño uml

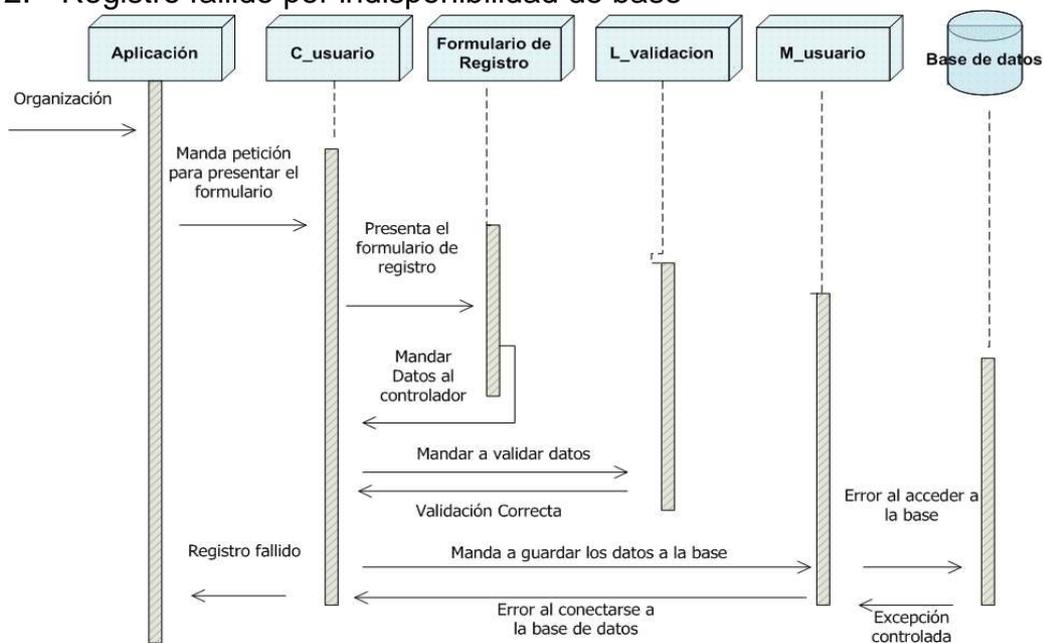
#### Módulo de Seguridad

#### CU01: Registrar a las Organizaciones

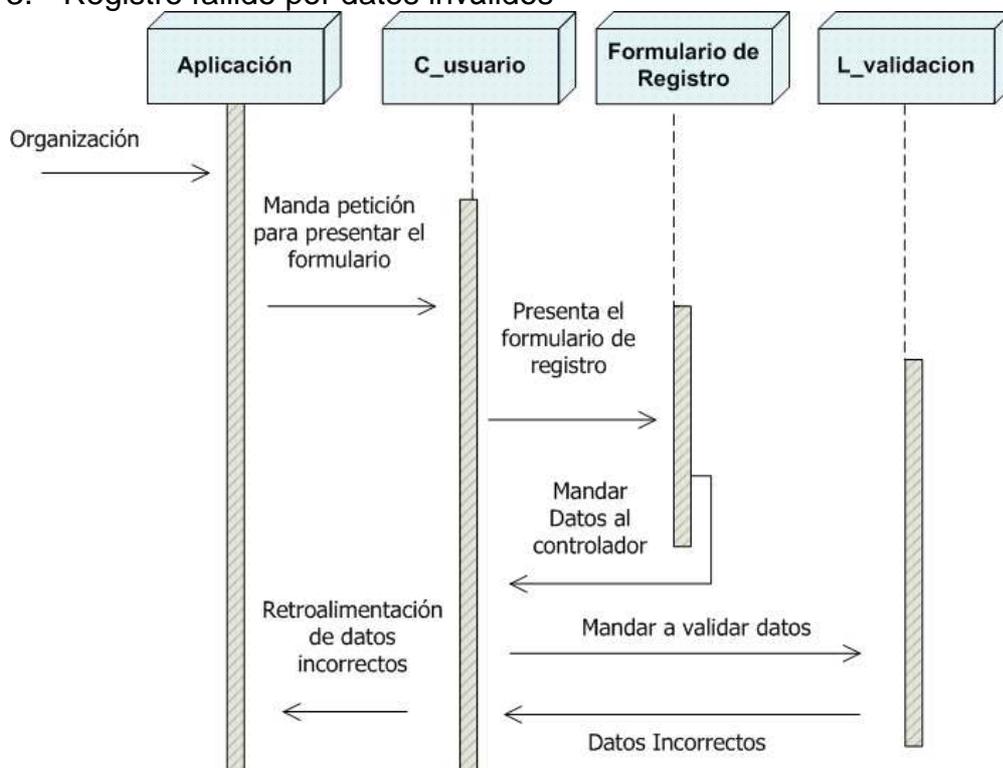
##### 1. Registro exitoso de una organización



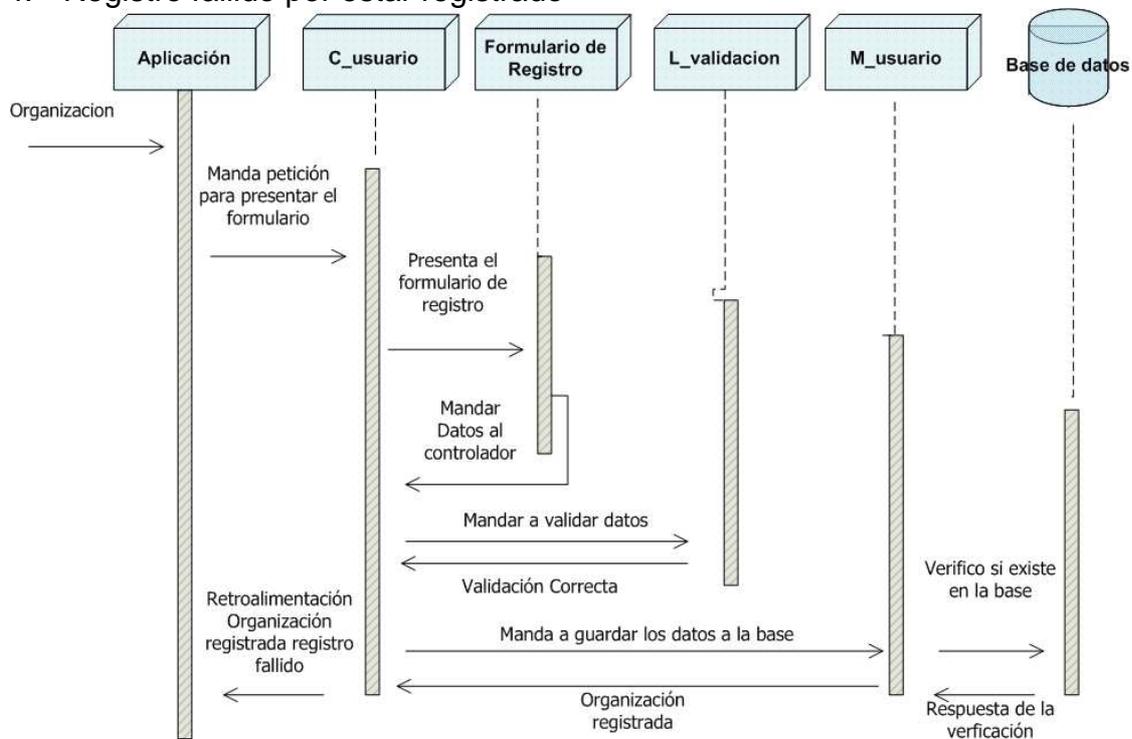
##### 2. Registro fallido por indisponibilidad de base



## 3. Registro fallido por datos inválidos



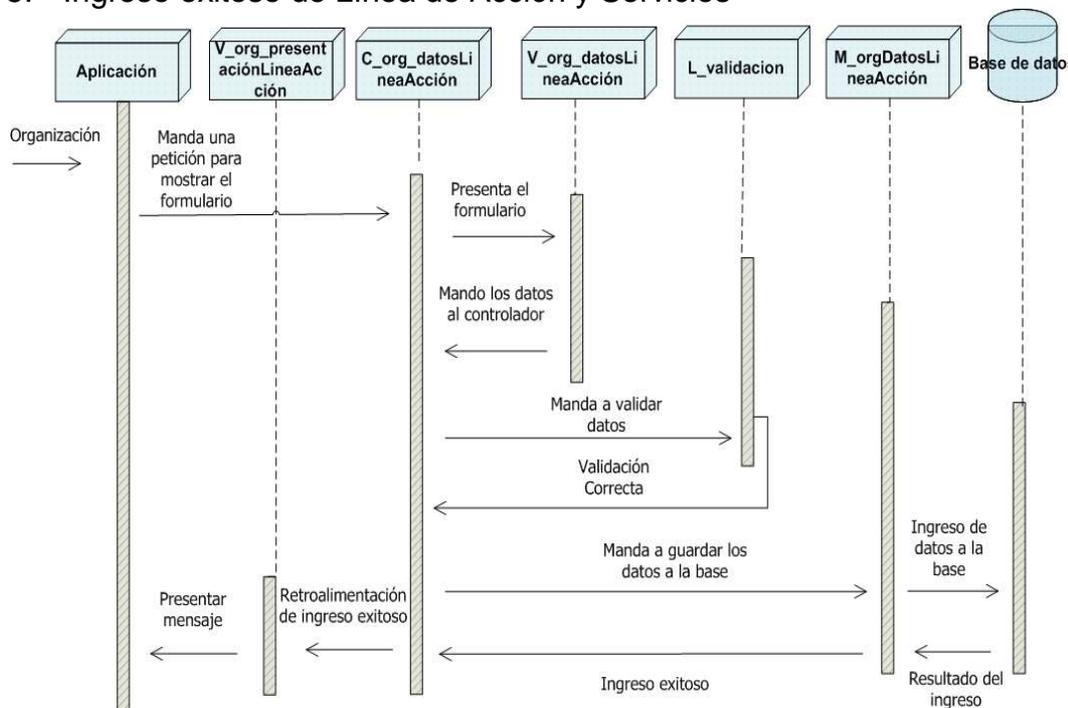
## 4. Registro fallido por estar registrado



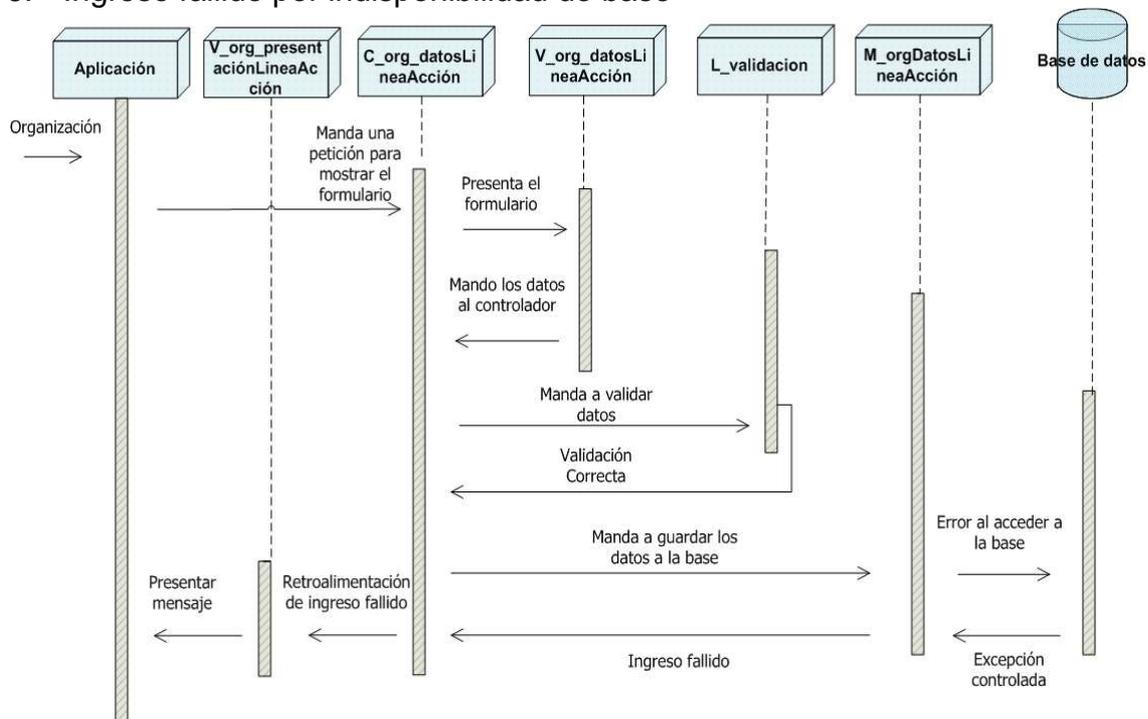
## Módulo de Línea de Acción y Servicios

### CU02: Ingresar Línea de Acción y Servicios

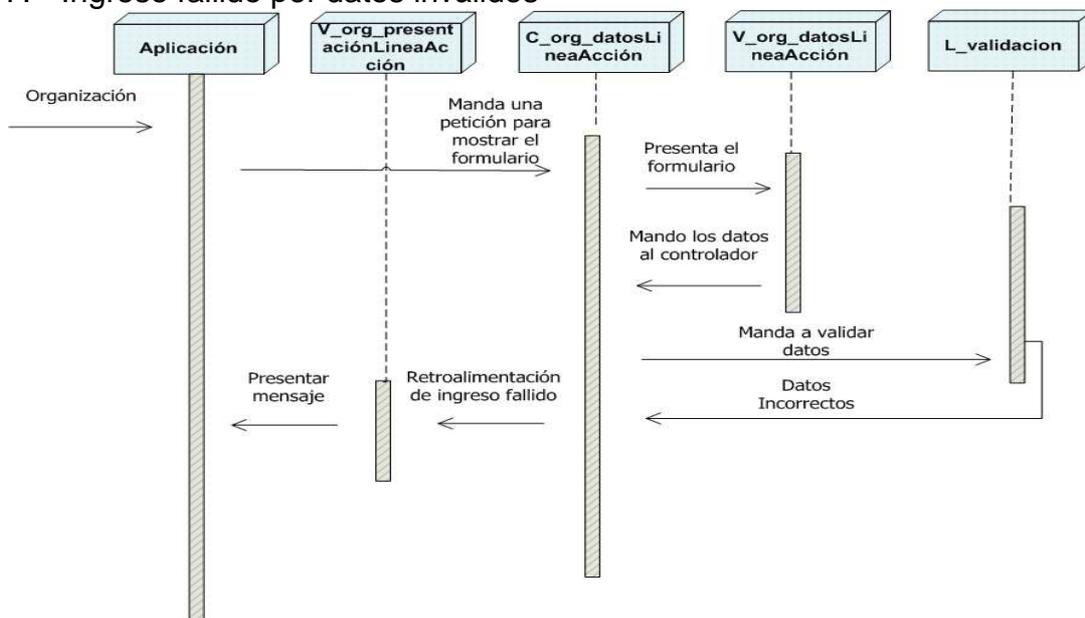
#### 5. Ingreso exitoso de Línea de Acción y Servicios



#### 6. Ingreso fallido por indisponibilidad de base

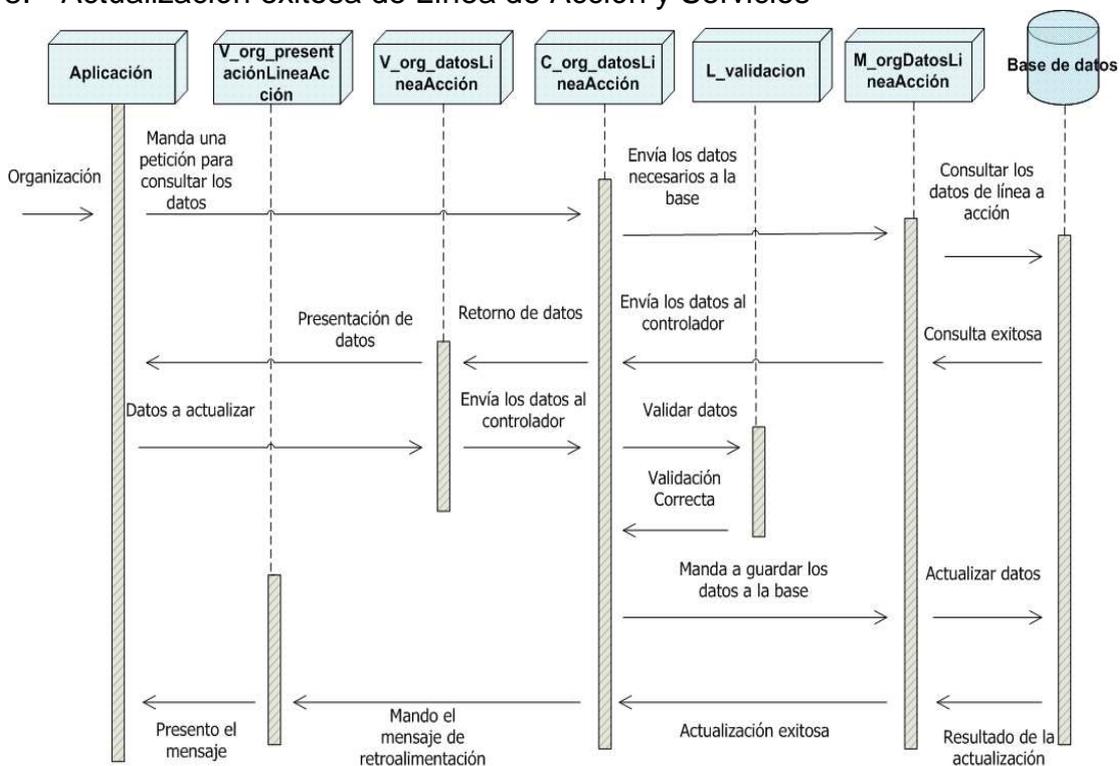


## 7. Ingreso fallido por datos inválidos

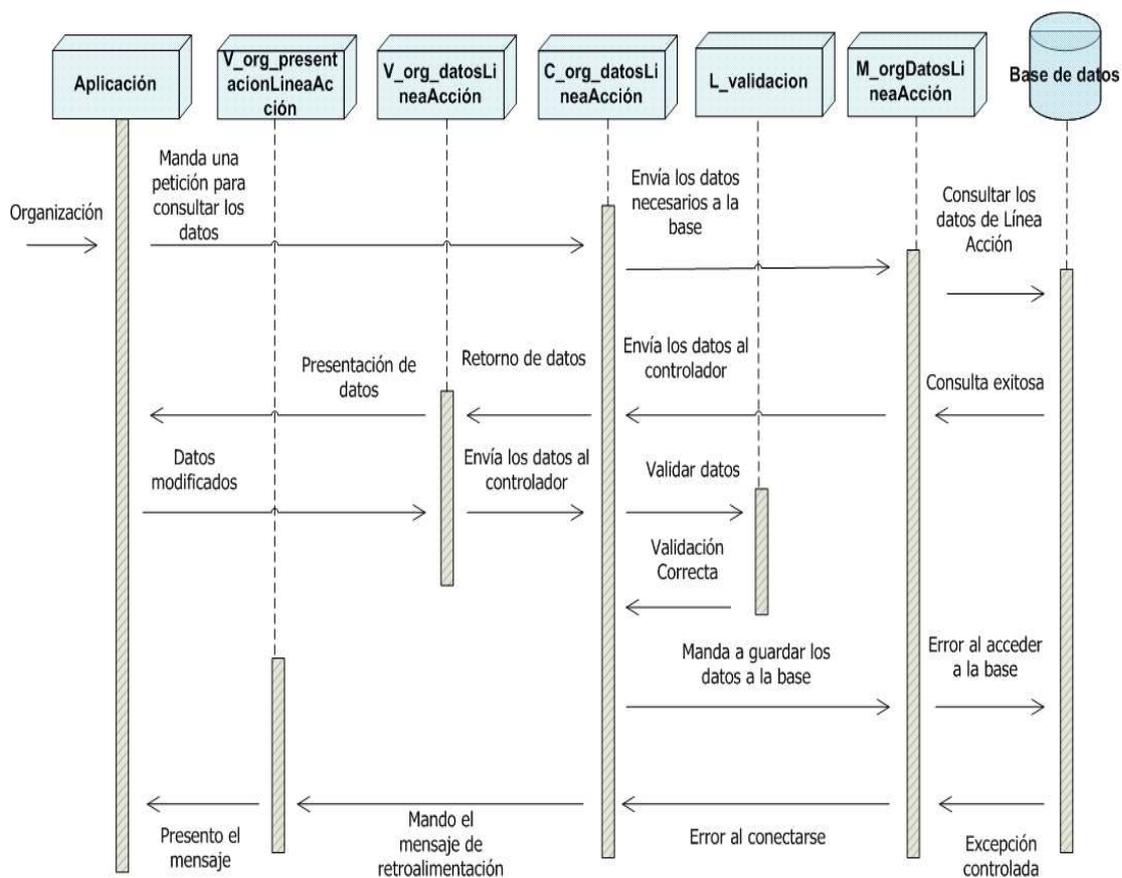
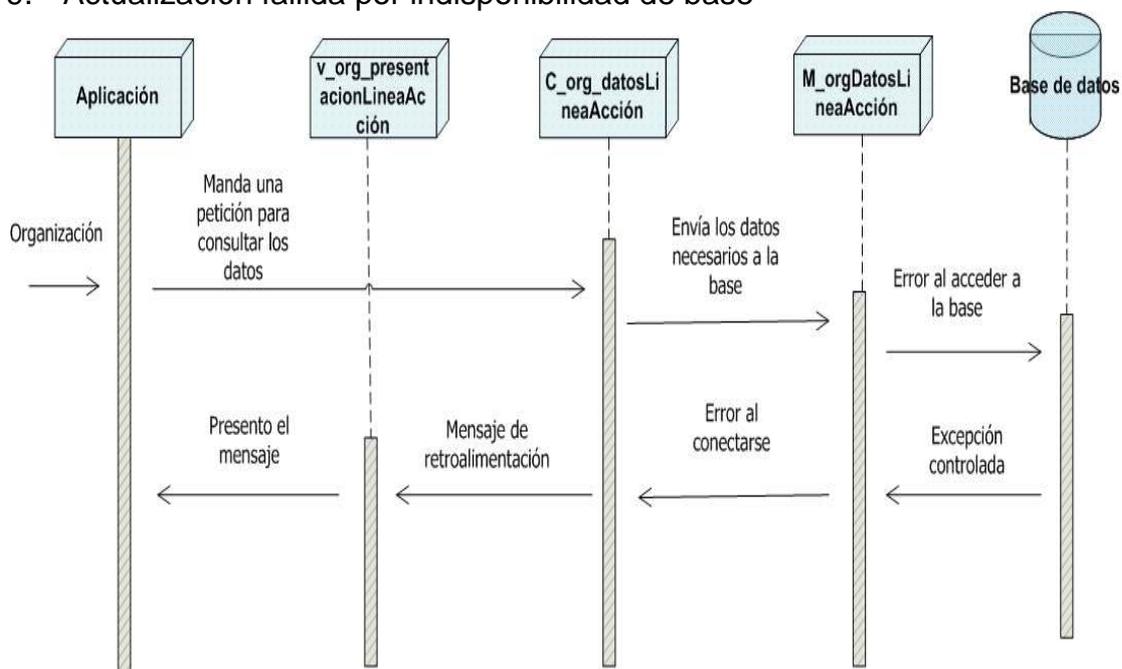


## CU03: Actualizar Línea de Acción y Servicios

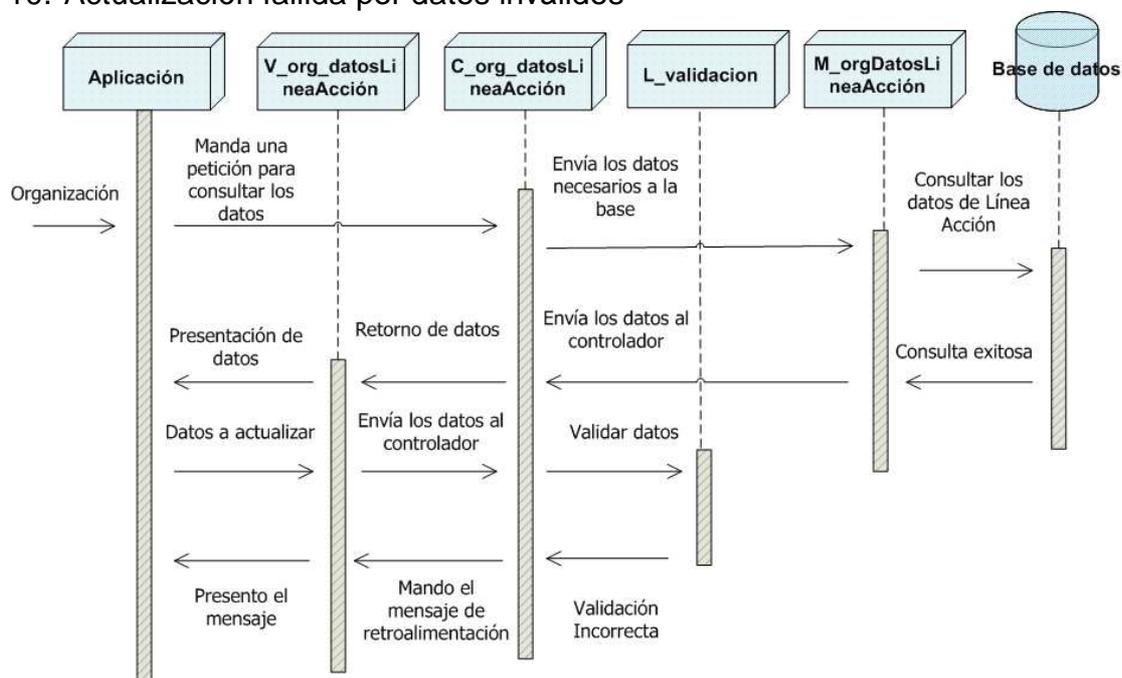
## 8. Actualización exitosa de Línea de Acción y Servicios



## 9. Actualización fallida por indisponibilidad de base



## 10. Actualización fallida por datos inválidos



### 3.4 Diseño de Pruebas

Las pruebas nos permiten verificar la calidad del sistema. Estas, nos ayudan a identificar los posibles fallos de implementación, calidad o usabilidad que tiene RIOS.

Comúnmente las organizaciones sin fines de lucro tienen como cabeza principal a personas adultas que oscilan entre los 35 años en adelante y que por lo general no usan frecuentemente la computadora. Las pruebas serán realizadas a personas que tengan este perfil que nos permitirá corregir, y mejorar el sistema para que su desempeño sea el más adecuado.

Se realizarán las siguientes pruebas:

- Tiempo que le toma al usuario registrarse
- Tiempo que se demoran las organizaciones en llenar los formularios.
- Encuesta sobre la usabilidad del sistema

### **Prueba\_001: Tiempo que le toma al usuario registrarse**

Esta prueba la consideramos importante debido a que si el registro es demorado, el usuario no se va a registrar.

<b>Sexo</b>	
<b>Edad</b>	
<b>Sabe usar una computadora?</b>	
<b>Hora Inicio</b>	
<b>Hora Fin</b>	
<b>Observación</b>	
<b>Resultados</b>	

**Plantilla de Prueba\_001**

### **Prueba\_002: Tiempo que se demoran las organizaciones en llenar los formularios**

Medimos el tiempo en cada uno de los formularios, para saber si la información presentada es entendida por el usuario. La plantilla es la misma que la de la Prueba\_001.

### Prueba\_003: Encuesta sobre la usabilidad del sistema

Esta encuesta se la realiza para saber si el sistema es amigable e intuitivo. La califica del 1 al 5 siendo 1 la menor valoración y 5 la máxima.

1. Es complicado el ingreso de información al sistema  

1	2	3	4	5
  
2. La información que se presentaba en los formularios era clara y concisa.  

1	2	3	4	5
  
3. Los mensajes de errores o aciertos fueron claros  

1	2	3	4	5
  
4. Fue intuitivo el sistema  

1	2	3	4	5
  
5. Visitaría el portal con frecuencia  

1	2	3	4	5
  
6. Cree usted que necesitaría capacitación para usar el sistema  
SI  NO
  
7. Recomendaría el sistema a otras organizaciones  
SI  NO

## **CAPÍTULO 4**

### **IMPLEMENTACIÓN**

El sistema fue desarrollado con herramientas de uso libre: php como script de programación, codeigniter fue el framework que le dio la modularización al sistema y mysql el motor de base de datos.

RIOS es una aplicación web que tiene una arquitectura cliente-servidor y que va a ser manejada por tres tipos diferentes de usuarios: usuarios informativos, organizaciones y mesas de concertación. Es por este motivo que las interfaces visuales son diferentes dependiendo del tipo de usuario.

Los usuarios informativos tienen acceso solamente a información general y al registro del sistema.

Las organizaciones tienen un acceso extra cuando inician sesión, que son las pestañas en las cuales llenan la respectiva información solicitada.

Los usuarios de mesas de concertación, tienen acceso a los diferentes reportes que han sido programados y a la sección informativa.

### **Seguridades del Sistema**

En el registro de organizaciones, el campo contraseña que ingresa el usuario es encriptado en sha1 y almacenado en la base de datos.

En caso de que el usuario no recuerde la contraseña y solicite el envío, el sistema preguntará el nombre y el ruc de la organización. Los datos ingresados son comparados con los almacenados en la base y si son válidos enviará una notificación al correo electrónico ingresado en el registro. El correo electrónico por motivos de seguridad tendrá un enlace que se genera dinámicamente para ocultar referencias que se encuentren en la base de datos y que pudieran ser visibles al usuario. Haciendo clic sobre el enlace se abrirá una página que solicitará a la organización que digite una nueva contraseña. Una vez que la organización haya cambiado la contraseña, el enlace ya no tendrá validez.

También existe una pestaña en la que se puede cambiar la contraseña, pero esta sólo es visible cuando se ha iniciado sesión.

El municipio tiene configurado el servidor web, haciendo que las aplicaciones de ellos corran sobre un puerto seguro SSL. RIOS, al

albergarse en uno de los servidores del municipio, gozará de las características de correr sobre un puerto seguro y de todas las seguridades que ellos tienen configuradas.

### **Formularios Dinámicos**

Existen tres formularios: Datos Básicos, Población y Línea de Acción. Todos los formularios están divididos en pasos para una mejor organización visual.

Los formularios de Población y Línea de Acción son formularios dinámicos, los cuales para ser presentados tienen que existir en la base de datos.

Existen campos que pueden aumentarse, por ejemplo: en el formulario de línea de acción y servicios los campos se pueden aumentar hasta 3. Este valor se encuentra almacenado en la tabla parámetros, la cual almacena valores constantes del sistema. En caso de que el Municipio quiera aumentar o disminuir la cantidad de entradas en los formularios, puede cambiar en la tabla parámetros el valor que se decida. Una organización solo puede consultar su propia información.

### **Actualización de Datos**

Una organización puede registrarse en cualquier momento. Los formularios estarán disponibles para ser llenados por primera vez.

Existe un tiempo de actualización de datos que solo se realiza una vez al año, y es obligatoria para todas las organizaciones. El Municipio decidirá en qué fecha se actualizarán los datos. Estas variables de tiempo residen en la tabla parámetros.

Las organizaciones tendrán habilitados los formularios para realizar los campos pertinentes durante este periodo de tiempo. Cuando culmine el lapso de actualización, las organizaciones solo podrán consultar su información.

### **Historial de Datos**

Existen tablas que guardan un historial de la organización: org\_cobertura, org\_linea\_servicios, org\_grupo\_etareo, orgsexo. Cuando existen modificaciones estas tablas no modifican los datos, sino que vuelven a insertarlos.

### **Reportes**

En la sección de reportes se ha utilizado tecnología ajax y javascript, la cual nos facilita el retorno de consultas y realizar eventos sin tener que refrescar la página en su totalidad.

Los reportes pueden ser exportados a formato pdf para facilidad del usuario.

## **4.1 Requerimientos del servidor**

### **4.1.1 Hardware**

Para RIOS pueda funcionar se necesita como mínimo:

- Memoria Ram de 512 Mb.
- Disco duro de 40 Gb.

Debido a que el sistema se va a alojar en uno de los servidores del municipio, anticipadamente sabemos que los requerimientos antes mencionados van a ser mejorados.

### **4.1.2 Software**

Para que el sistema pueda correr necesita tener instalado:

- PHP versión 5 o superior.
- Apache 2.0 o superior.
- Mysql 5.0 o superior.

Codeigniter es el framework de php en el que se ha trabajado. Existe un archivo de configuración: database.php, en el que se configura la base de datos, este archivo va a tener que ser modificado con la configuración deseada.

## **4.2 Requerimientos del cliente**

### **4.2.1 Hardware**

RIOS puede ser ejecutado en cualquier computadora que tenga características iguales o superiores a un Procesador Intel Pentium III, 747 MHz 384 MB de RAM.

### **4.2.2 Software**

RIOS puede ser accedido desde cualquier computadora que tenga un navegador e internet.

El sistema ha sido probado con Firefox/3.5.7, Internet Explorer 7.0 y Chrome 4.0, en los cuales no se ha presentado ningún inconveniente.

## **CAPÍTULO 5**

### **PRUEBAS**

#### **5.1 Pruebas de Usabilidad**

En la sección 3.4 se diseñaron las pruebas, que fueron tomadas con el afán de tomar medidas en cuanto a ciertos parámetros en el desarrollo de software para poder mejorar en los aspectos débiles y fortalecer los que están correctos. Entre los parámetros que se analizaron están:

- Tiempo que le toma al usuario registrarse
- Tiempo que se demoran las organizaciones en llenar los formularios.
- Encuesta sobre la usabilidad del sistema

Las pruebas fueron realizadas a 10 personas con el perfil mencionado en la sección 3.4. A continuación el análisis de los resultados.

## **5.2 Métricas**

Las métricas son el análisis de los resultados obtenidos en las pruebas. Estas mediciones nos ayudan a visualizar las condiciones del sistema.

En la prueba de usabilidad se han realizado 10 preguntas, que se han clasificado en: usabilidad, motivación y retroalimentación.

### **Tiempo que le toma al usuario registrarse**

El tiempo promedio que tomó a los usuarios en llenar el formulario de registro fue 3,5 minutos, con un promedio en la experticia del uso en el computador de 3,1 sobre 5.

Los usuarios demostraron interés al registrarse debido que ellos consideraron un beneficio que el Municipio les considere para las diferentes actividades que pudieren realizar.

### **Tiempo que se demoran las organizaciones en llenar los formularios.**

El tiempo promedio en llenar todos los formularios fue: 10,8 minutos.

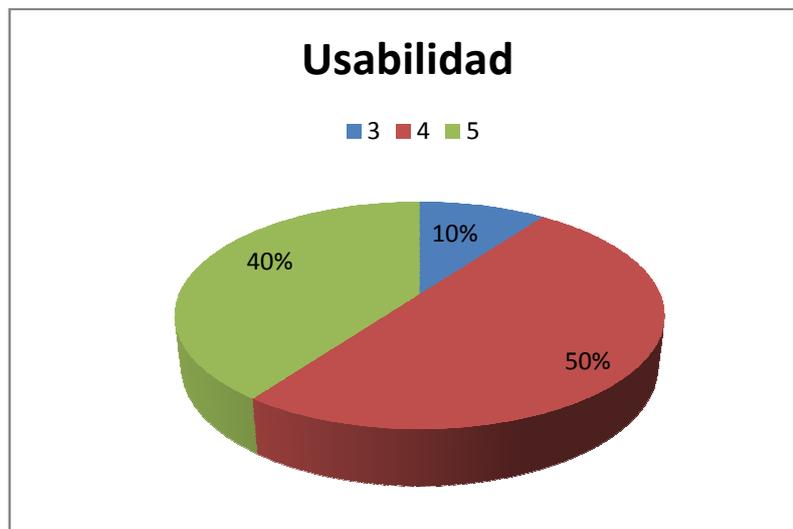
En la mayoría de los casos los usuarios tuvieron errores al momento de ingresar la información en el formulario de Información Básica, pero el sistema valida todas las entradas, por lo cual tuvieron al segundo intento de ingreso de información el 100% de ingresos fueron correctos.

Sobre el formulario de Población y Línea de Servicios consideraron excelente la idea que el Municipio sepa a quienes ayudan y el tipo de servicios que brindan.

### **Encuesta sobre la usabilidad del sistema**

La encuesta tiene varias preguntas que son de diferentes categorías que miden: la usabilidad, la motivación para volver a usar el sistema y la retroalimentación adecuada del mismo.

Los resultados en cuanto al factor usabilidad fueron:



**Figura 5.1. Usabilidad del Sistema**

Como podemos observar los resultados fueron favorables, pero se puede seguir mejorando la interfaz para hacer que el sistema “hable” por sí mismo.

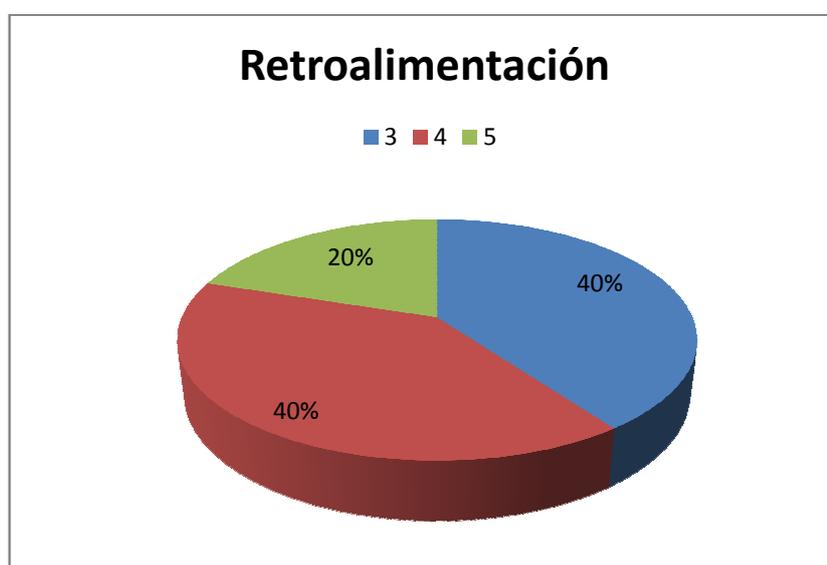
Los resultados del factor motivación fueron los siguientes:



**Figura 5.2. Motivación del Sistema**

Aquí observamos que los resultados no fueron tan favorables, pudiendo percibir que el sistema le falta dar información a las organizaciones.

Los resultados del factor retroalimentación fueron los siguientes:



**Figura 5.3. Retroalimentación del Sistema**

Los resultados fueron buenos, pero debido a que los usuarios que se equivocaron en los formularios no pudieron visualizar rápidamente el mensaje de error, se realizó un cambio en el sistema para poder mejorar la retroalimentación. Cuando ya se hubo mejorado la interfaz, los resultados mejoraron.

### 5.3 Resultados obtenidos

De los resultados expuestos anteriormente, con usuarios cuya edad promedio es de 38 años y un uso del computador de 3 sobre 5, podemos concluir:

- Hay un alto porcentaje de organizaciones que tendrá la predisposición de registrarse, debido a que les interesa que el Municipio los tenga en su base de datos. El tiempo promedio en esta prueba es 3,5 minutos, que es un tiempo prudente para que las organizaciones se registren.
- Al momento de llenar los formularios, el tiempo promedio fue 10,8 min. El mayor tiempo ocupado fue en el formulario de Información Básica debido a que se solicitan datos como misión, visión, código predial, que son datos que no se tienen a la mano, que se tienen que pensar o recurrir a un documento para poderlos visualizar y escribir. La mayoría de usuarios se equivocaron en este formulario, pero los mensajes de error que da el sistema ayudaron a que el formulario sea ingresado con éxito. Los formularios de Población y Línea de Acción se ingresaron sin inconvenientes.
- La usabilidad que tiene el sistema es buena, con un 90% de aceptación, ya que considerando que en promedio las personas que realizaron las pruebas no tienen un uso frecuente en el computador

podieron registrarse y llenar todo lo solicitado, en algunos casos con mayor esfuerzo, pero ninguno pasó de los 22 minutos.

- La motivación que tiene el usuario no es tan buena, no por lo difícil de la interfaz, sino que los usuarios también sugirieron ver la información de las demás organizaciones registradas en el portal, a fin de realizar alianzas o saber que servicios brindan.
- La retroalimentación fue mejorada, debido al comportamiento que se que obtuvieron en las primeras pruebas. Los mensajes de error se colocaron en lugares más visibles, para que el usuario pueda percatarse del error más rápido.

## **CONCLUSIONES Y RECOMENDACIONES**

### **CONCLUSIONES**

En el periodo de análisis, diseño e implementación para realizar este proyecto de graduación, se puede concluir:

- 1) Con el portal RIOS es posible almacenar los datos de las organizaciones de una manera estructurada, ordenada, con un mismo formato, para que las Mesas de Concertación puedan manipularla y consultarla.
- 2) La información presentada por medio de los reportes permite visualizar situaciones que ayudan a la toma de decisiones; con lo cual se espera que las Mesas de Concertación puedan canalizar mejor los recursos al momento de realizar una obra benéfica.
- 3) El sistema se adapta a las posibles adiciones de atributos ya que permite aumentarlos o disminuirlos de acuerdo a las necesidades de las Mesas de Concertación.
- 4) El diseño de los formularios permite la recolección de datos básicos, población y servicios que brindan a la comunidad, de tal manera que las Mesas de Concertación puedan evaluar y seleccionar a las

organizaciones que cumplan con las expectativas del proyecto que se vaya a ejecutar.

- 5) Es posible obtener y actualizar los datos de las organizaciones en tiempo real, de tal manera que la información presentada a las Mesas de Concertación sea la más reciente.
  
- 6) Gracias a la arquitectura del sistema y el diseño de los datos es posible la extensión del portal RIOS a partir de la integración de otros módulos.

## RECOMENDACIONES

- 1) Se sugiere que existan consultas en la sección informativa del portal, para que la sociedad pueda saber información básica de las organizaciones.
- 2) Se recomienda que las organizaciones puedan consultar información entre ellas que les permitirá armar alianzas o simplemente consultar sus servicios.
- 3) A pesar de que RIOS está implementado bajo las versiones de Apache 2.2.8, MySql 5.0.51 y PHP 5.2.6 para futuras implementaciones puede ser ejecutado a versiones superiores compatibles a los mismos.
- 4) En caso de que se quiera expandir el sistema, recomendamos seguir la estructura MVC, ya que tendrá un mejor mantenimiento y orden al implementarlo.

- 5) Si se desea ver los gráficos de los reportes que se han exportado a formato PDF sean iguales a los presentados en el portal, se sugiere que se adquiera la licencia de la librería gráfica FusionCharts.
  
- 6) A partir de la primera versión de RIOS, consideramos que el Municipio debería seguir expandiéndolo, ya que podrían obtener mejores resultados si obtienen información más puntual, y específica acerca de las organizaciones.

## **ANEXOS**

## ANEXO REQUISITOS DEL DESARROLLADOR

<b>ID</b>	RD_001		
<b>Nombre</b>	Registrar Organización		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Registra una organización en el sistema.</li> <li>✓ Una organización solo podrá registrarse una sola vez.</li> <li>✓ El usuario tiene que ser único.</li> <li>✓ Si el registro fue exitoso, se mostrará un mensaje para retroalimentar al usuario.</li> <li>✓ Una vez registrada la organización se mostrará una ventana de acceso al portal.</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	---		
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ La fecha de registro se guarda automáticamente.</li> <li>✓ Se llenan los datos principales de la organización: <ul style="list-style-type: none"> <li>- Nombre de Organización</li> <li>- Nombre de Representante</li> <li>- Ruc</li> </ul> </li> <li>✓ Los datos secundarios se ingresarán cuando el usuario haya iniciado sesión.</li> <li>✓ Se llenan los siguientes campos: <ul style="list-style-type: none"> <li>- Dirección</li> <li>- Correo Electrónico</li> <li>- Usuario</li> <li>- Clave</li> </ul> </li> <li>✓ La organización se genera por defecto con estado vigente.</li> <li>✓ Id de la organización es autoincremental.</li> </ul>		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/ dato</b>	<b>Longit ud</b>

Nombre Organización	nombre_org	String	50
Nombre Representante	representante_org	String	100
Dirección	direccion_cont	String	100
Ruc	ruc_org	String	13
Usuario	usuario_us	String	10
Clave	clave_us	String	10
Correo Electrónico	email_cont	String	50

<b>ID</b>	RD-002		
<b>Nombre</b>	Ingresar al Sistema		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite ingresar al sistema.</li> <li>✓ Se mostrará un mensaje de retroalimentación al usuario.</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-001		
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Solo se podrá ingresar al sistema si la organización se ha registrado previamente.</li> <li>✓ Habrá un campo tipo de usuario, el cual es un FK que indica si el tipo de usuario que ingresa al sistema es: <ul style="list-style-type: none"> <li>- Administrador</li> <li>- Mesa de Concertación</li> <li>- Organización</li> </ul> </li> <li>✓ Según el tipo de usuario se muestra el menú.</li> <li>✓ En caso de haber olvidado la clave, se lo podrá recuperar solicitando el envío de un correo electrónico al mail de registro.</li> </ul>		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
Usuario	usuario_us	String	10
Clave	clave_us	String	10

<b>ID</b>	RD-003		
<b>Nombre</b>	Ingresar Información General		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite llenar datos básicos, contactos, estrategias y asociaciones a las mesas de concertación.</li> <li>✓ Se podrá ingresar varios números de teléfonos, estos son almacenados en una tabla adicional.</li> <li>✓ Se podrá ingresar varios objetivos, estos son almacenados en una tabla adicional.</li> <li>✓ En la tabla parámetros reside el número máximo de teléfonos y objetivos que puede guardar la organización. El valor actual es 3 para los teléfonos y 10 para los objetivos.</li> <li>✓ La información de las sucursales se guarda en una tabla adicional llamada Org_sucursales. El número de sucursales está limitado por el dato en la tabla parametro. Actualmente es 10.</li> <li>✓ Las mesas asociadas y el contacto de la organización por mesa se guardan en tablas separadas.</li> <li>✓ Para las mesas el identificador se guarda en la tabla org_mesas_asociada.</li> <li>✓ Los datos del contacto se guardan en la tabla contacto.</li> <li>✓ Las mesas asociadas pueden aumentarse, y el número máximo depende del número de mesas que están en la base. Actualmente 6.</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-002		
<b>Observación</b>	✓ Para llenar este formulario se necesita haber iniciado una sesión.		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
<b>Datos Básicos</b> Código Predial	predial_org	String	20

Código de Censo	censo_org	String	45
Casilla Postal	casilla_org	String	15
Rup	rup_org	String	13
Rous	rous_org	Bool	--
<b>Contactos</b>			
Página Web	pagweb_org	String	45
Mail	email_cont	String	50
Fax , extensión	teléfono_tl,	String	10
Teléfono	ext_tl	String	10
<b>Información Estratégica</b>	Telefono_tl		
Misión			
Visión	Mission_org	String	500
Objetivos	Vision_org	String	500
<b>Ubicación</b>	Descripcion_op	String	200
Matriz			
Sucursales	id_ad_matriz	Integer	---
<b>Mesas Asociadas</b>	id_ad	Integer	---
Mesa Asociada:			
Nombre del delegado a la Mesa:	Id_ms	Integer	---
Cargo:			
Teléfono1:	nombre_cont	String	50
Teléfono2:	cargo_cont	String	50
Correo Electrónico	telefono_tl	String	50
Dirección	teléfono_tl	String	50
	email_cont	String	50
	direccion_cont	String	100

<b>ID</b>	RD-004
<b>Nombre</b>	Ingresar Población
<b>Descripción</b>	✓ Permite ingresar datos relacionados al sexo, grupo etario y cobertura a la que atiende la organización.
<b>Requisitos Asociados</b>	--
<b>Tipo</b>	Funcional
<b>Dependencias</b>	

<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Estos formularios son dinámicos, por consiguiente los datos son cargados de la base de datos.</li> <li>✓ Todos los datos de esta sección son obligatorios.</li> <li>✓ En grupo etario se puede elegir varias opciones.</li> <li>✓ En la sección de cobertura institucional el mínimo es de un registro, el máximo depende de una variable que reside en la base de datos en la tabla de parámetros. El valor actual es 15.</li> </ul>		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
<b>Sexo</b> Electiva, única.	FK	Integer	
<b>Grupo Etario</b> Electiva, múltiple	FK	Integer	
<b>Cobertura Institucional</b> Grupo etario	FK	Integer	
Sexo	FK	Integer	
Num_beneficiarios	num_beneficiario	Integer	

<b>ID</b>	RD-005
<b>Nombre</b>	Ingresar Línea de Acción
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite ingresar datos relacionados a los servicios que prestan las organizaciones.</li> </ul>
<b>Requisitos Asociados</b>	--
<b>Tipo</b>	Funcional
<b>Dependencias</b>	
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Estos formularios son dinámicos, por consiguiente los datos son cargados de la base de datos.</li> <li>✓ Todos los datos de esta sección son obligatorios.</li> </ul>

	✓ El mínimo de la línea de acción y servicios es uno, el máximo depende de una variable que reside en la base de datos en la tabla de parámetros. El valor actual es 3.		
Nombre	Identificador	Tipo/dato	Longitud
<b>Línea de Acción</b> Electiva.	FK	Integer	
<b>Servicios</b> Electiva	FK	Integer	

### Actualización

<b>ID</b>	RD-006		
<b>Nombre</b>	Actualizar Ingresar Información General		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite modificar datos básicos, contactos e información estratégica.</li> <li>✓ Se podrá ingresar, modificar y eliminar varios números de teléfonos y objetivos.</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-002, RD-003		
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Para modificar esta información, se tuvo que haber iniciado sesión.</li> <li>✓ Solo se podrá modificar la información en una fecha permitida.</li> <li>✓ Se tendrá que validar la información ingresada.</li> <li>✓ La información se modifica en las respectivas tablas.</li> </ul>		
Nombre	Identificador	Tipo/dato	Longitud
<b>Datos Básicos</b>			
Dirección	Direccion_cont	String	100
Código Predial	Predial_org	String	20
Código de Censo	Censo_org	String	45
Casilla Postal	Casilla_org	String	15

Rup	Rup_org	String	13
Rous	Rous_org	Bool	--
<b>Contactos</b>			
Página Web	Pagweb_org	String	45
Mail	Email_cont	String	50
Fax , extensión	Telefono_tl	String	10
Teléfono	Teléfono_tl, ext_tl	String	10
<b>Estrategia</b>			
Misión	Mission_org	String	500
Visión	Vision_org	String	500
Objetivos	Descripcion_op	String	200

<b>ID</b>	RD-007
<b>Nombre</b>	Actualizar Población
<b>Descripción</b>	✓ Permite ingresar, modificar datos relacionados al sexo, grupo etario y cobertura a la que atienden.
<b>Requisitos Asociados</b>	--
<b>Tipo</b>	Funcional
<b>Dependencias</b>	RD-002, RD-004
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Estos formularios son dinámicos, por consiguiente los datos son cargados de la base de datos.</li> <li>✓ Todos los datos de esta sección son obligatorios.</li> <li>✓ En grupo etario se puede elegir varias opciones.</li> <li>✓ En cobertura institucional se pueden aumentar filas.</li> <li>✓ Se tendrá que validar la información ingresada.</li> <li>✓ La información se modifica en las respectivas tablas, pero para llevar un historial del comportamiento de la</li> </ul>

	<p>organización, se insertará cierta información en una tabla historial.</p> <p>✓ La información que se llevará en la tabla historial es:</p> <ul style="list-style-type: none"> <li>- grupo_etario_org</li> <li>- cobertura_org</li> </ul>		
Nombre	Identificador	Tipo/dato	Longitud
<b>Sexo</b> Electiva, única. <b>Grupo Etario</b> Electiva, múltiple <b>Cobertura Institucional</b> Grupo etario Sexo Num_beneficiarios	FK FK FK FK Num_beneficiarios	Integer Integer Integer Integer Integer	

<b>ID</b>	RD-008
<b>Nombre</b>	Actualizar Línea de Acción
<b>Descripción</b>	<p>✓ Permite ingresar, modificar y eliminar datos relacionados a los servicios que prestan las organizaciones.</p>
<b>Requisitos Asociados</b>	--
<b>Tipo</b>	Funcional
<b>Dependencias</b>	RD-002, RD-005
<b>Observación</b>	<p>✓ Estos formularios son dinámicos, por consiguiente los datos son cargados de la base de datos.</p> <p>✓ Todos los datos de esta sección son obligatorios.</p> <p>✓ Se tendrá que validar la información ingresada.</p> <p>✓ La información que se llevará en la tabla historial es:</p> <ul style="list-style-type: none"> <li>- línea_servicios_org</li> </ul>

Nombre	Identificador	Tipo/dato	Longitud
<b>Línea de Acción</b> Electiva.	FK	Integer	
<b>Servicios</b> Electiva	FK	Integer	

<b>ID</b>	RD-009		
<b>Nombre</b>	Olvido de Contraseña		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite modificar la clave de acceso al sistema.</li> <li>✓ Solicitará datos relevantes de la organización.</li> <li>✓ El sistema enviará un correo electrónico con el que se registró.</li> <li>✓ El mail tendrá un link que permitirá modificar la contraseña.</li> <li>✓ Después de cambiar la contraseña, el link enviado ya no tiene validez.</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-001		
<b>Observación</b>	✓ Se podrá modificar cada vez que el usuario lo solicite.		
Nombre	Identificador	Tipo/dato	Longitud
Clave	clave_us	String	10

## Consultas

<b>ID</b>	RD-010		
<b>Nombre</b>	Consultar Datos de la organización		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite consultar todos los datos de la organización.</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-002, RD-003, RD-004, RD-005		
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ La organización solo podrá consultar sus propios datos.</li> <li>✓ Aparecerá la fecha de la última actualización.</li> <li>✓ Aparecerán los datos actualizados.</li> </ul>		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
<b>Datos Básicos</b>			
Código Predial	Predial_org	String	20
Código de Censo	Censo_org	String	45
Casilla Postal	Casilla_org	String	15
Rup	Rup_org	String	13
Rous	Rous_org	Bool	--
<b>Contactos</b>			
Página Web	Pagweb_org	String	45
Correo Electrónico	Email_cont	String	50
Fax , extensión	Teléfono_tl, ext_tl	String	10
Teléfono	Telefono_tl	String	10
<b>Objetivos</b>			
Misión	Mission_org	String	500
Visión	Vision_org	String	500
Objetivos	Descripcion_op	String	200
<b>Sexo</b>			
Electiva, única.	FK	Integer	
<b>Grupo Etario</b>			
Electiva, múltiple	FK	Integer	
<b>Cobertura Institucion</b>			
Grupo etario	FK	Integer	

Sexo Num_beneficiarios	FK num_beneficiarios	Integer Integer	
<b>Línea de Acción</b> Electiva.	FK	Integer	
<b>Servicios</b> Electiva	FK	Integer	

## Reportes

<b>ID</b>	RD-011
<b>Nombre</b>	Reporte de Organizaciones por área de desarrollo
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite visualizar en gráfico de barras las organizaciones que existen en un área de desarrollo.</li> <li>✓ También lista las organizaciones.</li> <li>✓ Cuando se hace clic sobre un área de desarrollo específica, se realiza el reporte RD-012</li> </ul>
<b>Requisitos Asociados</b>	----
<b>Tipo</b>	Funcional
<b>Dependencias</b>	RD-02
<b>Observación</b>	✓ Solo podrá acceder el tipo de usuario: Mesa de Concertación.

<b>ID</b>	RD-012
<b>Nombre</b>	Reporte de Grupo etario atendido por área de desarrollo
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite visualizar en gráfico de pastel los grupos etarios que están siendo atendido en el área de desarrollo especificada.</li> </ul>
<b>Requisitos Asociados</b>	----
<b>Tipo</b>	Funcional

<b>Dependencias</b>	RD-02, RD-011
<b>Observación</b>	✓ Solo podrá acceder el tipo de usuario: Mesa de Concertación.

<b>ID</b>	RD-013		
<b>Nombre</b>	Reporte de Organizaciones por línea de acción y servicio.		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite visualizar las organizaciones que están en una línea de acción y brindan un servicio específico.</li> <li>✓ Cuando se haga clic sobre la institución, se realizará el reporte RD-014</li> </ul>		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-02		
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Solo podrá acceder el tipo de usuario: Mesa de Concertación..</li> <li>✓ Los parámetros de entrada son: <ul style="list-style-type: none"> <li>- Línea de Acción</li> <li>- Servicio</li> </ul> </li> </ul>		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
Línea de Acción Servicio	Id_linea_ac Id_linea_serv	Integer Integer	

<b>ID</b>	RD-014		
<b>Nombre</b>	Reporte de Organizaciones según tipo de servicios.		
<b>Descripción</b>	<ul style="list-style-type: none"> <li>✓ Permite visualizar las organizaciones detalladamente según el tipo de línea de acción y servicio que brinden.</li> <li>✓ El parámetro de entrada es solicitado de la referencia RD-013</li> <li>✓ Se presenta la información de la organización: <b>Básica</b></li> </ul>		

	<ul style="list-style-type: none"> <li>- Dirección</li> <li>- Teléfono</li> <li>- Correo Electrónico</li> </ul> Sexo, grupo Etario y cantidad de personas atendidas
<b>Requisitos Asociados</b>	----
<b>Tipo</b>	Funcional
<b>Dependencias</b>	RD-02, RD-013
<b>Observación</b>	✓ Solo podrá acceder el tipo de usuario: Mesa de Concertación.

<b>ID</b>	RD-0015		
<b>Nombre</b>	Reporte de Organizaciones que ayudan según su grupo etario.		
<b>Descripción</b>	✓ Permite visualizar las organizaciones que atienden al grupo etario ingresado como parámetro.		
<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-02		
<b>Observación</b>	✓ Solo podrá acceder el tipo de usuario: Mesa de Concertación..		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
Grupo Etario	Id_grupo_et	Integer	

<b>ID</b>	RD-0017		
<b>Nombre</b>	Reporte de organizaciones asociadas a una mesa de concertación.		
<b>Descripción</b>	✓ Permite visualizar la cantidad de organizaciones asociadas a una mesa de concertación.		

<b>Requisitos Asociados</b>	----		
<b>Tipo</b>	Funcional		
<b>Dependencias</b>	RD-02		
<b>Observación</b>	<ul style="list-style-type: none"> <li>✓ Solo podrá acceder el tipo de usuario: Mesa de Concertación.</li> <li>✓ El parámetro de entrada: <ul style="list-style-type: none"> <li>- Mesa de Concertación</li> </ul> </li> </ul>		
<b>Nombre</b>	<b>Identificador</b>	<b>Tipo/dato</b>	<b>Longitud</b>
Mesa de Concertación	Id_ms	Integer	

## **ANEXO DE ANÁLISIS DE ALTERNATIVAS DE LAS HERRAMIENTAS MÁS ADECUADAS**

### **Análisis del Sistema Operativo**

La importancia de elegir un sistema operativo es fundamental, ya que se ven afectados factores como: economía, herramientas tecnológicas y selectivas de personal.

Nuestras aplicaciones *“van a correr sobre un sistema operativo, beneficiándose de sus puntos fuertes y viéndose perjudicado por sus debilidades. Por muy potente que sea un determinado aplicativo, si se implementa sobre un sistema operativo pobre, el resultado no puede ser excepcionalmente bueno” (1).*

A continuación mencionaremos algunos criterios que se analizarán:

- Integración con los usuarios
- Estabilidad y Disponibilidad
- Seguridad y Mantenimiento
- Carga de trabajo y Hardware
- Costo

### **Microsoft Windows 2008 Server**

Windows 2008 Server es el sistema operativo más reciente de Microsoft, es útil utilizarlo cuando se tiene red con dominio y active directory, porque es fácil integrar usuarios de Windows a la aplicación. Otra ventaja es que el

entorno gráfico, es muy intuitivo y amigable, comúnmente los usuarios prefieren este entorno, personas con o sin experiencia pueden manipularlo. También tiene la capacidad de correr sesiones paralelas, es decir que varias personas al mismo tiempo pueden acceder a perfiles diferentes y correr sus aplicaciones.

En general la estabilidad del sistema, en muchos de los casos depende de factores de terceros como aplicaciones que pudieran no estar disponibles, incompatibilidad o conflicto entre ellas, haciendo que el sistema no sea estable. Los virus son otra causa de inestabilidad, ya que pueden realizar acciones que el administrador no desee y perjudiquen al S.O.

En cuestión de seguridad, es muy vulnerable a los virus, ni teniendo un buen antivirus actualizado, hace que nuestro sistema no este expuesto a ataques indeseados.

*“Los tiempos de mantenimiento son elevadísimos, en parte causados por el deficiente sistema de ficheros utilizado, NTFS, que obliga a desfragmentar los sistemas de archivo muy frecuentemente. Casi cada actualización del sistema operativo obliga a reiniciar el sistema”. [1]*

Con respecto al rendimiento del sistema operativo indicamos que no es tan favorable porque la carga de trabajo de un antivirus, entorno gráfico que es un devorador de recursos hace mermar su rendimiento, haciendo que el hardware sea muy exigente para correr nuestra aplicación. En consecuencia debemos de costear valores de tener un buen hardware y

antivirus.

Al ser Windows un sistema privativo, es decir que no tenemos acceso al código de implementación del sistema, hay que pagar por el uso de este.

## **Linux**

Linux es un S.O de código abierto que ha tenido acogida en los últimos años.

No es tan fácil integrar usuarios de Windows a alguna aplicación, hay que realizar una serie de configuraciones para que el sistema pueda ejecutarse.

Linux tiene la facilidad de instalar los paquetes que el usuario necesite, es por este motivo que se puede elegir si instalar el entorno gráfico o modo consola. Al estar en modo consola, un usuario no avanzado no sabrá como interactuar con el sistema.

Los virus no son un problema para Linux. La mayoría de los virus son creados para Windows y no corren bajo Linux. Por este lado, no tenemos inestabilidad en el sistema operativo. En cuanto a los drivers son genéricos, esto hace que no haya incompatibilidad o conflicto entre ellas.

Cuando se instala algún nuevo componente no es necesario reiniciar el sistema, sino que por medio de la consola se puede mandar a recargar las librerías nuevas, haciendo que la mayor parte del tiempo haya

disponibilidad.

Linux es seguro. Maneja permisos para modificar, eliminar, crear archivos y ejecutar aplicaciones, no es vulnerable a virus.

El tiempo de mantenimiento es bajo, en Linux no hay que desfragmentar los sistemas de archivos y si existe alguna actualización Linux detiene el servicio, actualiza y lo levanta sin tener que reiniciar el sistema.

La carga de trabajo en el S.O sabiendo configurar de una manera óptima es baja, ya que se puede obviar la instalación del entorno gráfico y con respecto al antivirus no es necesario. Es por este motivo que las características del hardware no son exigentes.

Linux por ser un sistema de código abierto, se lo puede obtener descargando desde la página de los diferentes distribuidores gratis.

Los costos de los sistemas operativos analizados están en la siguiente tabla.

## **COSTOS DE LOS SISTEMAS OPERATIVOS**

<b>Sistema Operativo</b>	<b>Costo \$</b>
Windows 2008 Server	840
Linux	0

### **Análisis del Motor de Bases de Datos**

Es importante elegir un buen motor de base de datos (SGBD) porque la gestión de la empresa gira alrededor de esta. Un SGBD es un repositorio en donde se van a almacenar la información relevante a las transacciones realizadas por la empresa. Por lo tanto se tiene que considerar seguridad, rapidez, costos al momento de elegir un SGBD.

Analizaremos los siguientes parámetros:

- Instalación y Configuración
- Interfaz de usuario
- Mantenimiento y Rendimiento
- Estabilidad
- Costo

### **Microsoft SQL SERVER 2008**

Por lo general la instalación y configuración de los productos Microsoft es fácil y sencilla, porque viene acompañada de una interfaz amigable e intuitiva.

El SQL Server Manager es la herramienta de administración que permite al usuario ver de una forma ordenada diferentes tipos de objetos como: bases de datos, diagrama conceptual, tablas, procedimientos almacenados, triggers, vistas, funciones, tareas programadas entre otros. Es fácil visualizar el contenido de cada uno de los objetos, ya que a medida que se va observando un objeto se puede ver el contenido, atributos, propiedades en la parte derecha.

Otra característica del SQL Server son sus procedimientos almacenados y triggers. Estos son independientes de la aplicación, y se ejecutan en el servidor facilitando su mantenimiento porque en caso de fallos solo se modifica ese procedimiento o trigger sin afectar a la aplicación, esto también influye en el rendimiento porque lo realiza la base de datos, evitando realizar consultas que congestionan la red entre clientes y servidor.

En cuanto a la estabilidad SQL Server tiene un problema, ya que manipulando la tabla se puede duplicar las claves únicas. Las tablas se pueden corromper. Este motor solo funciona bajo la plataforma de Windows.

SQL Server se maneja con licencias es por este motivo que se paga por el servidor y por los clientes que se sirvan de él. Para nuestro análisis se a

considerar la licencia del servidor con 4 instancias.

## **MYSQL**

Instalar y configurar Mysql en Windows es fácil y sencillo como cualquier otra instalación y configuración de alguna aplicación que se desee ejecutar. Bajo el entorno Linux la instalación es algo más compleja en algunos casos y la configuración se lo hace de forma manual.

Mysql tiene algunas herramientas de administración como Mysql Administrator, phpMyAdmin, Mysql para Webmin, donde podemos administrar gráficamente las bases de datos, tablas, procedimientos almacenados, triggers, entre otros. También se puede administrar en modo consola sin depender de ningún software y sistema operativo específico.

Al igual que Sql server tiene procedimientos almacenados y triggers que colabora con el mantenimiento y rendimiento del sistema.

Mysql es estable, porque no permite claves duplicadas. También se puede elegir como se desea almacenar los datos en la bases.

Este motor es multiplataforma, funciona en Windows, Linux, Mac, Solaris entre otros.

Los costos de los motores analizados están en la siguiente tabla:

## COSTOS DE BASES DE DATOS

MOTOR DB	Costo \$
SQL Server (+ 4 calls)	1175
MySQL	0
MySQL con Soporte	595.00 - 4985.00

### Análisis de Scripts de programación

Escoger un lenguaje o script de programación es importante porque es la herramienta en que se va a realizar la aplicación.

Los criterios a analizar es:

- Facilidad de aprendizaje y componentes adicionales
- Multiplataforma
- Velocidad de ejecución
- Soporte
- Costo

### ASP

ASP es una tecnología Web que impulsó Microsoft y es una de las más utilizadas. No es un lenguaje de programación, se puede usar como lenguaje scripting con C#, C++ y visual basic lo que lo hace más manejable al programador, puede elegir la herramienta que desea trabajar.

Aprender ASP es fácil porque el entorno gráfico que ofrece hace que el programador se oriente a la funcionalidad del sistema y se despreocupe de

la presentación de sus ventanas. Permite la visualización fácil los objetos, clases y atributos. Los componentes adicionales son fáciles de adquirirlos pero tienen un costo adicional.

Debido que es un producto de Microsoft solo corre bajo Windows.

La velocidad de ejecución de ASP en un sistema pequeño es muy bueno pero a medida de que es sistema crece se torna lento debido a que en su compilación tiene más código que depurar.

Microsoft tiene un grupo de personas que se encargan de hacer los parches apenas reportan fallos. Debido a que el IDE Visual Studio es pagado Microsoft da soporte.

## **PHP**

PHP es la tecnología web más popular por los desarrolladores, su aprendizaje es rápido y sencillo, debido a que se parece a C que por lo general es el lenguaje que todos desarrolladores dominan. Su programación es estructurada u orientada a objetos.

Los componentes adicionales para PHP son desarrollados por la comunidad de PHP sin costo alguno. Si no existe hay que esperar a que la comunidad lo emita o desarrollarlo uno mismo.

PHP es multiplataforme corre en más de 25 sistema operativos, entre lo más populares tenemos Windows, Linux, Mac, solaris, entre otros.

La velocidad de ejecución es buena, no requiere demasiados recursos del sistema. Si se configura como módulo de Apache y está listo para ser utilizado. Tiene su propio sistema de administrador de recursos.

La extensa comunidad que tiene PHP alrededor mundo hace como soporte al lenguaje debido a que es de código abierto se busca la manera de aportar con frameworks, plugins, foros, tutoriales, APIs para expandir más la fama de la herramienta.

## **JSP**

JSP es una tecnología web bastante popular y que su funcionamiento se basa en scripts utilizando lenguaje java.

El propio lenguaje te obliga a ser ordenado, sin duda alguna, el más ordenado de los tres que hemos analizado, siendo así una opción interesante a la hora de desarrollar proyectos escalables, añadiendo “módulos” (clases), etc. Aunque esta característica se convierte en desventaja al momento de aprender el lenguaje, ya que para una persona que nunca ha programado el tiempo que le tiene que dedicar a la herramienta es bastante considerable a diferencia de los otros dos. Esto es debido a que Java es un lenguaje muy potente, pero un poco más complejo de usar porque es orientado a objetos y la manera de escribir los programas es más rígida. Los componentes adicionales se los puede descargar de la página [Sun Microsystems](#).

JSP se puede crear aplicaciones web que se ejecuten en varios servidores web, de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma.

La velocidad de ejecución de JSP se considera rápida y dispone del API de Java en su totalidad.

Los costos de los scripts analizados están en la siguiente tabla:

#### **COSTOS DE LOS SCRIPTS DE PROGRAMACIÓN**

<b>SCRIPT</b>	<b>Costo \$</b>
ASP.NET	523.93
PHP	0
JSP	0

A continuación una tabla comparativa de los lenguajes analizados, del 1 al 5, donde 5 es la mejor valoración, estos valores son asignados a cada una de las tecnologías.

	<b>ASP.NET</b>	<b>PHP 5</b>	<b>JSP</b>
Facilidad de aprendizaje	4	5	3
Componentes adicionales	3	5	5
Multiplataforma	1	5	5
Velocidad de ejecución	3	5	5
Soporte	5	4	4
Costo	2	5	5
<b>TOTAL:</b>	18	29	27

A continuación se analizan las diferentes combinaciones de plataformas:

#### **Windows+Microsoft SQL server+ASP**

**Costo:** \$5058.93. No disponible para pequeñas y medianas empresas.

Sistema caro e inestable.

**Multiplataforma:** no aplica. SQLSERVER y ASP solo corren bajo Windows.

**Velocidad de ejecución:** mientras más grande más grande se hace el sistema, el tiempo de respuesta es mayor.

#### **Windows+ MYSQL + ASP**

**Costo:** \$3883.93. Esta tecnología aun sigue siendo cara.

**Multiplataforma:** no aplica. ASP solo corre bajo Windows.

**Velocidad de ejecución:** mejora el tiempo de respuesta.

#### **Windows+ MYSQL + PHP/ JSP**

**Costo:** \$800.

**Multiplataforma:** si aplica a nivel de lenguaje y motor de base de datos.

**Velocidad de ejecución:** mejora el tiempo de respuesta.

#### **Linux+ MYSQL + PHP**

**Costo:** \$0.

**Multiplataforma:** si aplica, todos son multiplataforme.

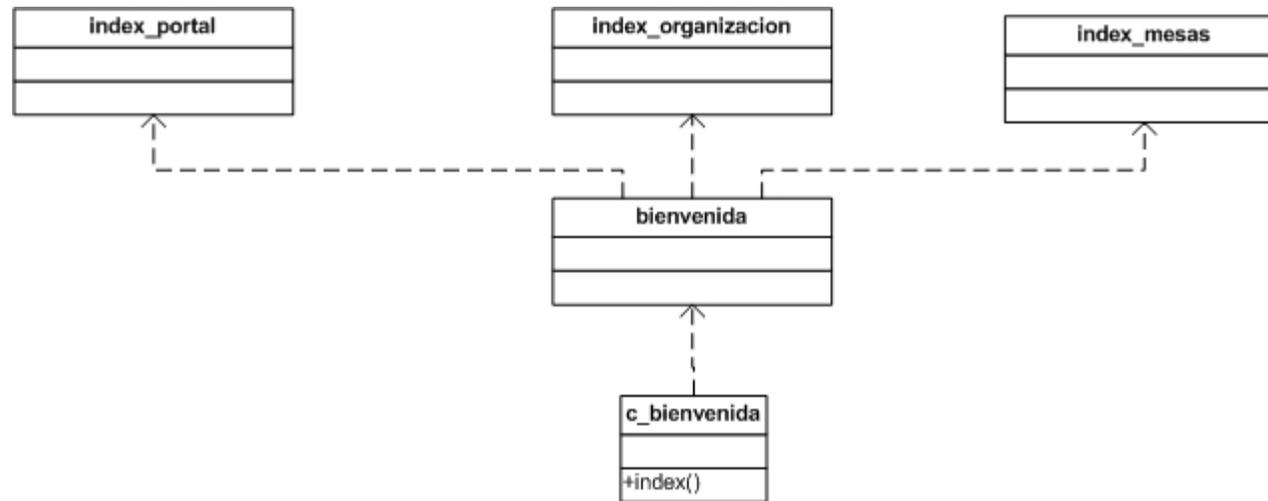
**Velocidad de ejecución:** motor de bases de datos es rápido y el lenguaje ligero.

#### **Linux+ MYSQL + JSP**

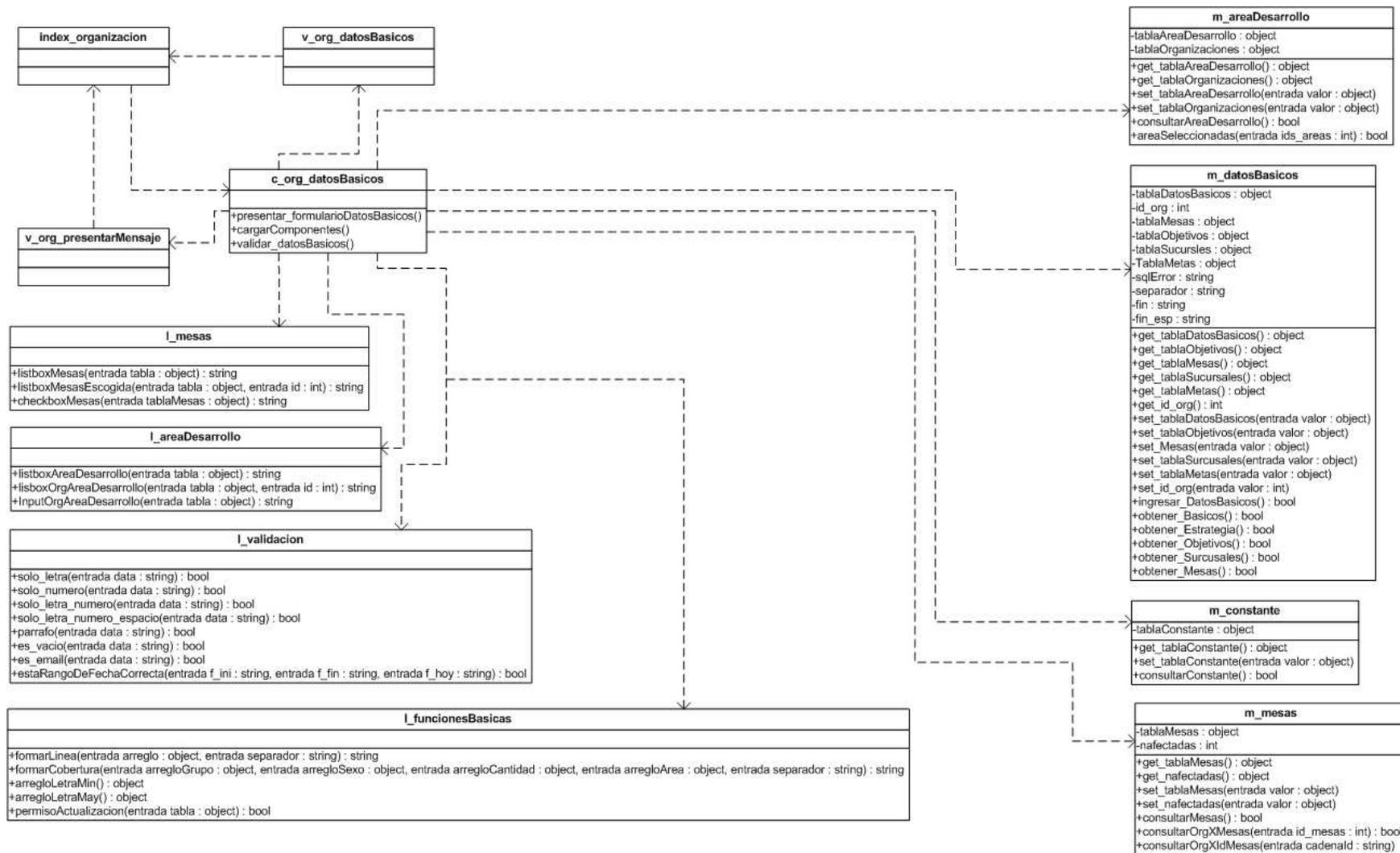
**Costo:** \$0.

**Multiplataforma:** si aplica, todos son multiplataforme.

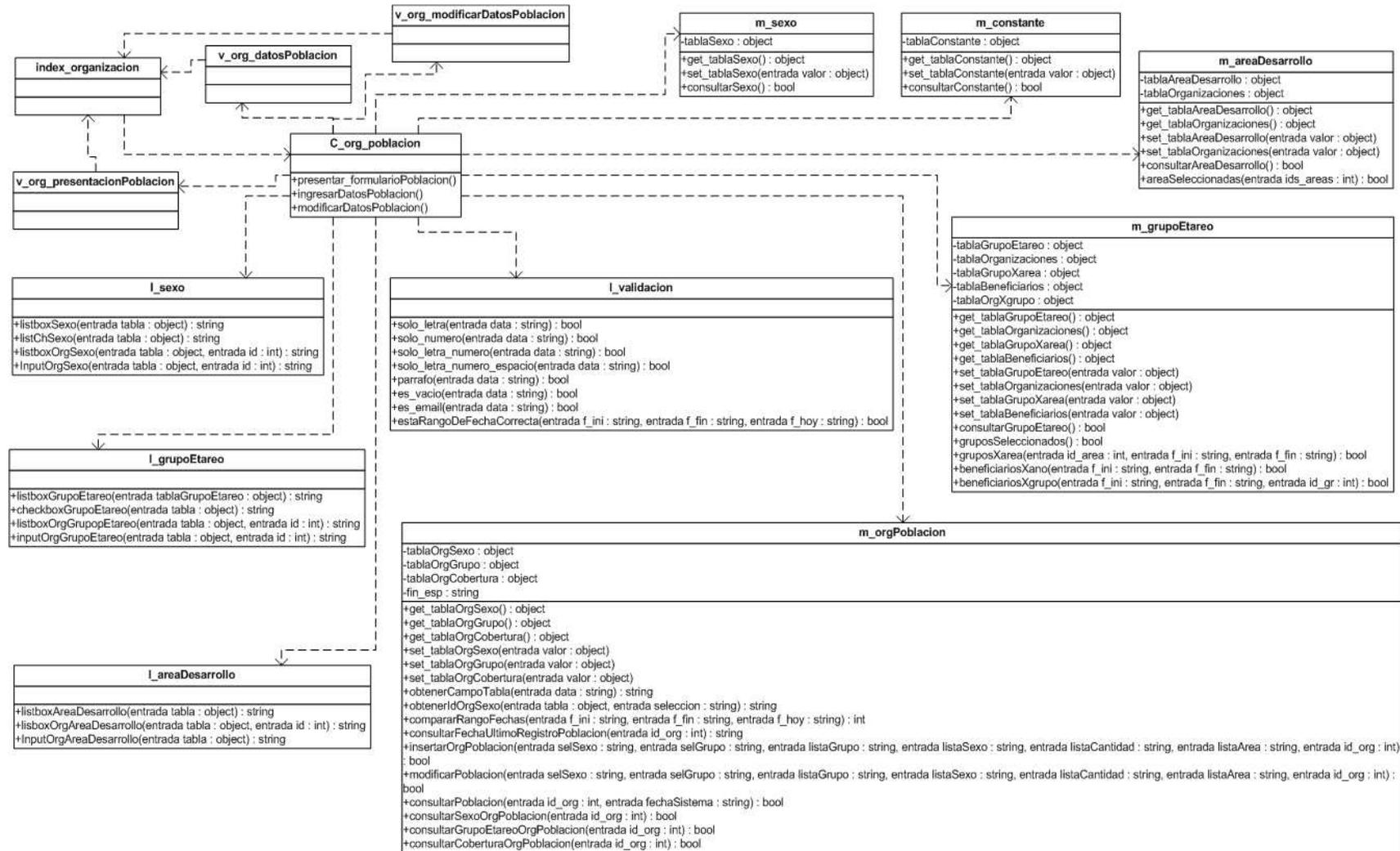
**Velocidad de ejecución:** motor de base de datos es rápido, pero el lenguaje de programación es un poco más lento que PHP.

**ANEXO DIAGRAMA DE CLASES****CONTROLADOR DE BIENVENIDA**

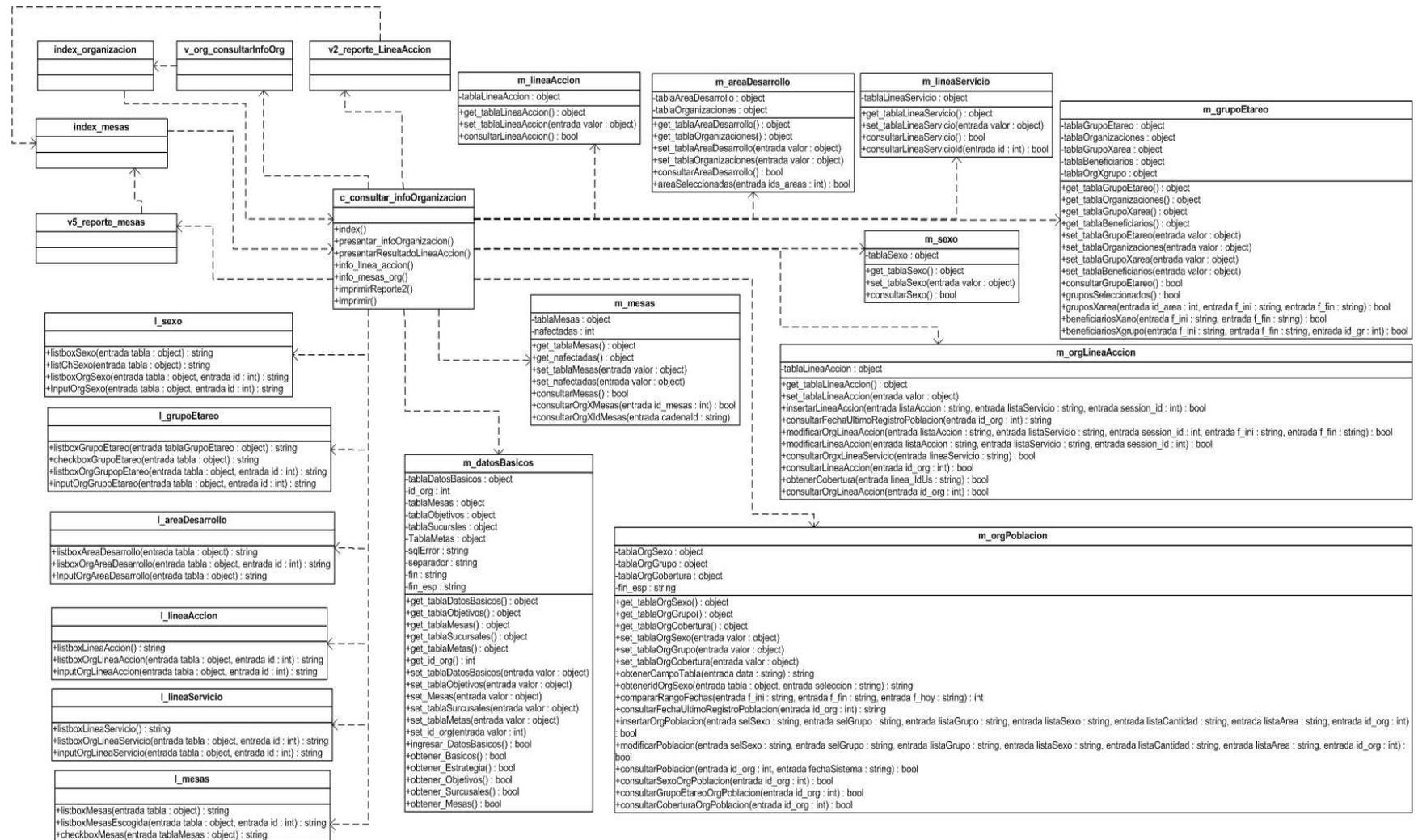
## CONTROLADOR DE DATOS BÁSICOS



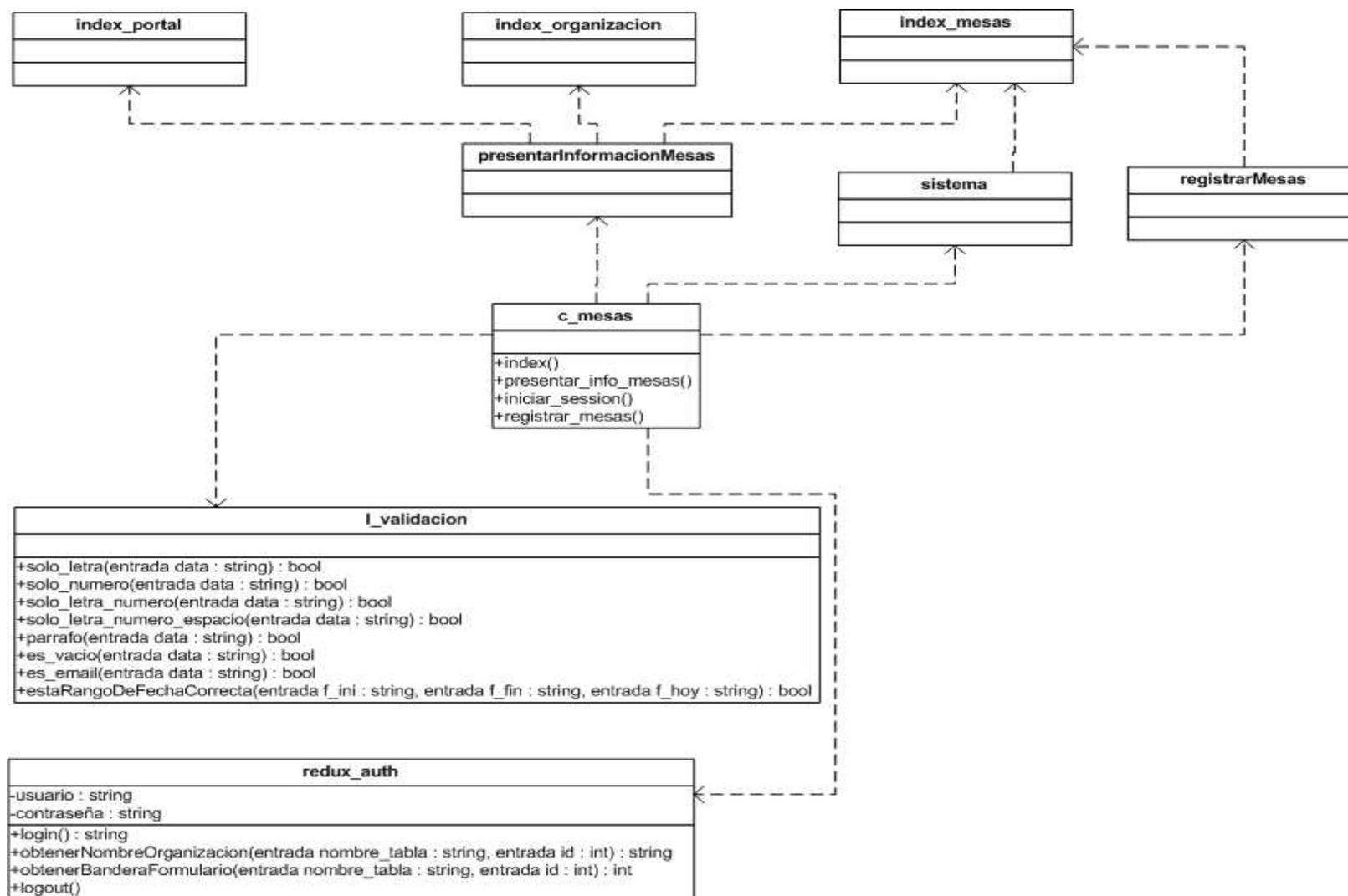
## CONTROLADOR DE POBLACION



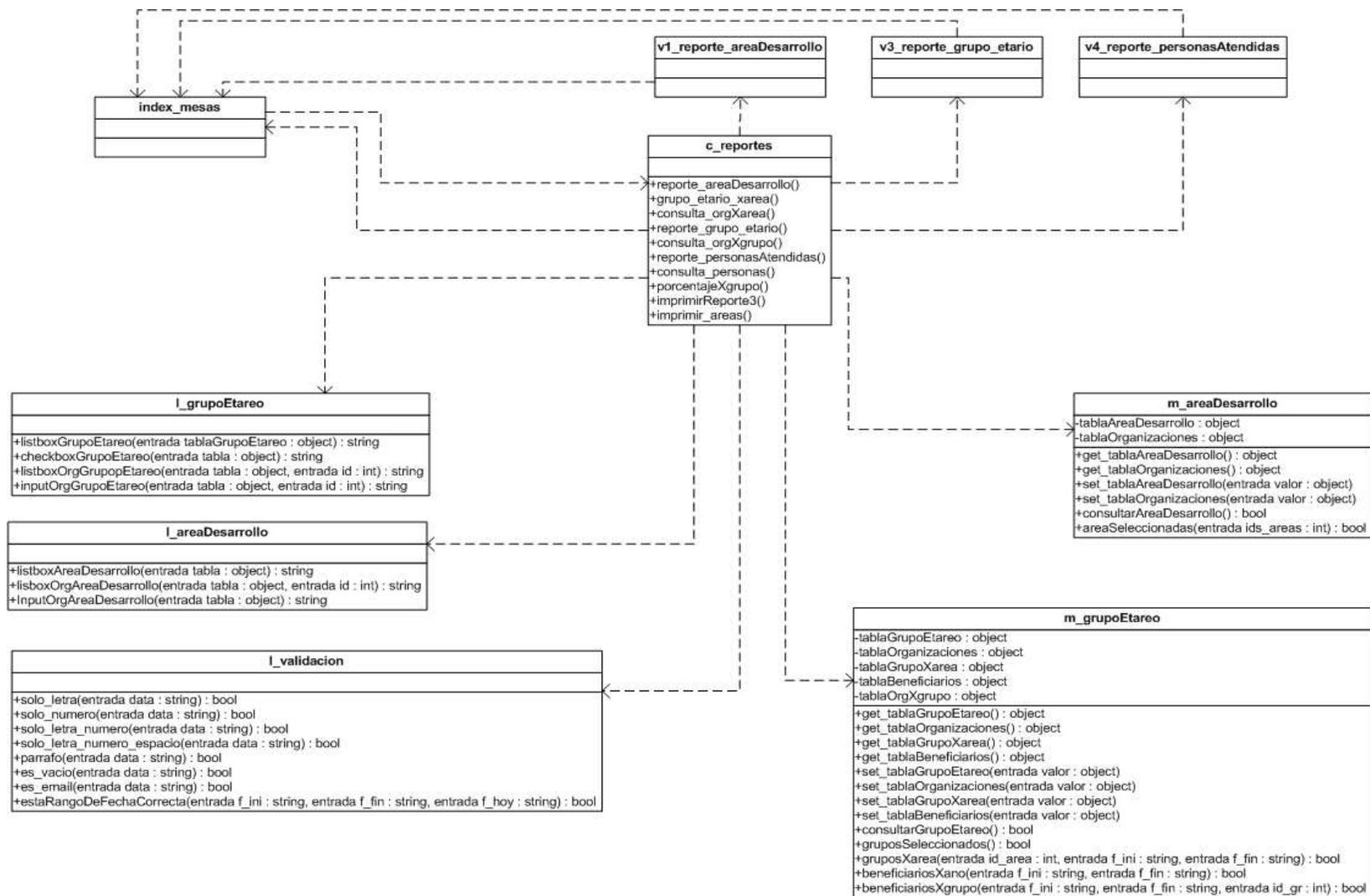
## CONTROLADOR DE CONSULTA



## CONTROLADOR DE MESAS



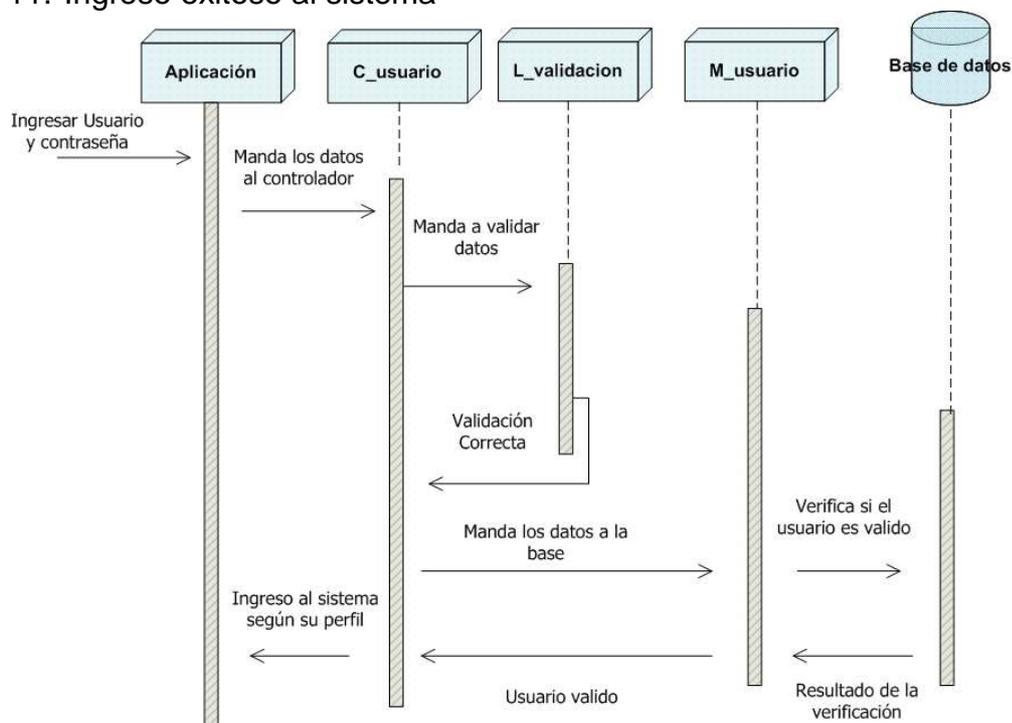
## CONTROLADOR DE REPORTES



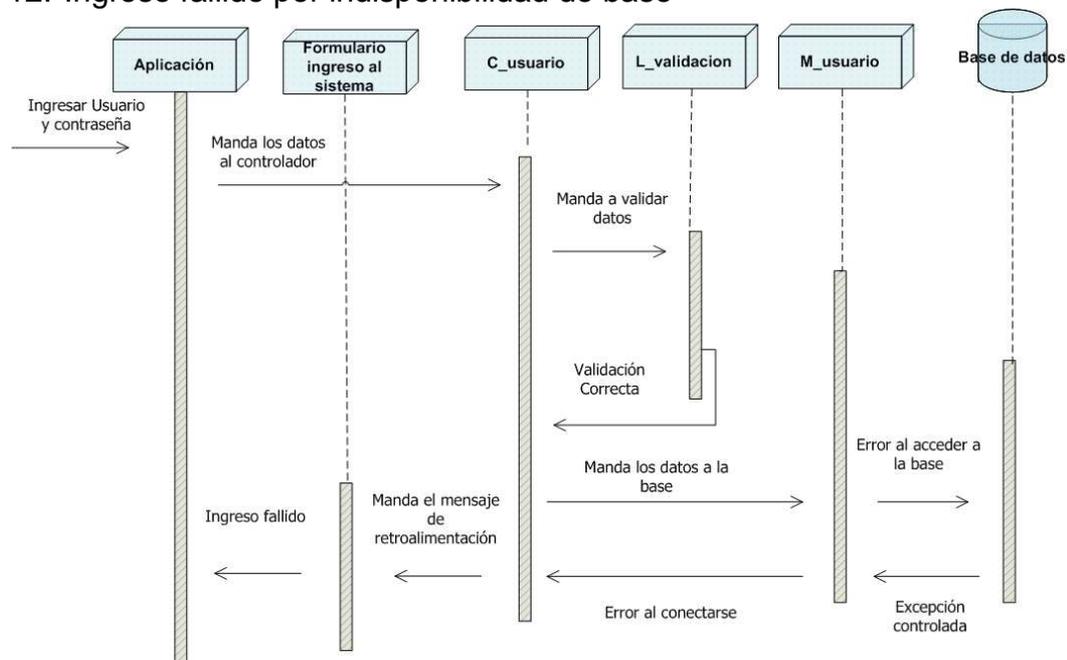
## ANEXOS DE UML

## 5.3.1.1 CU04: Ingresar al Sistema

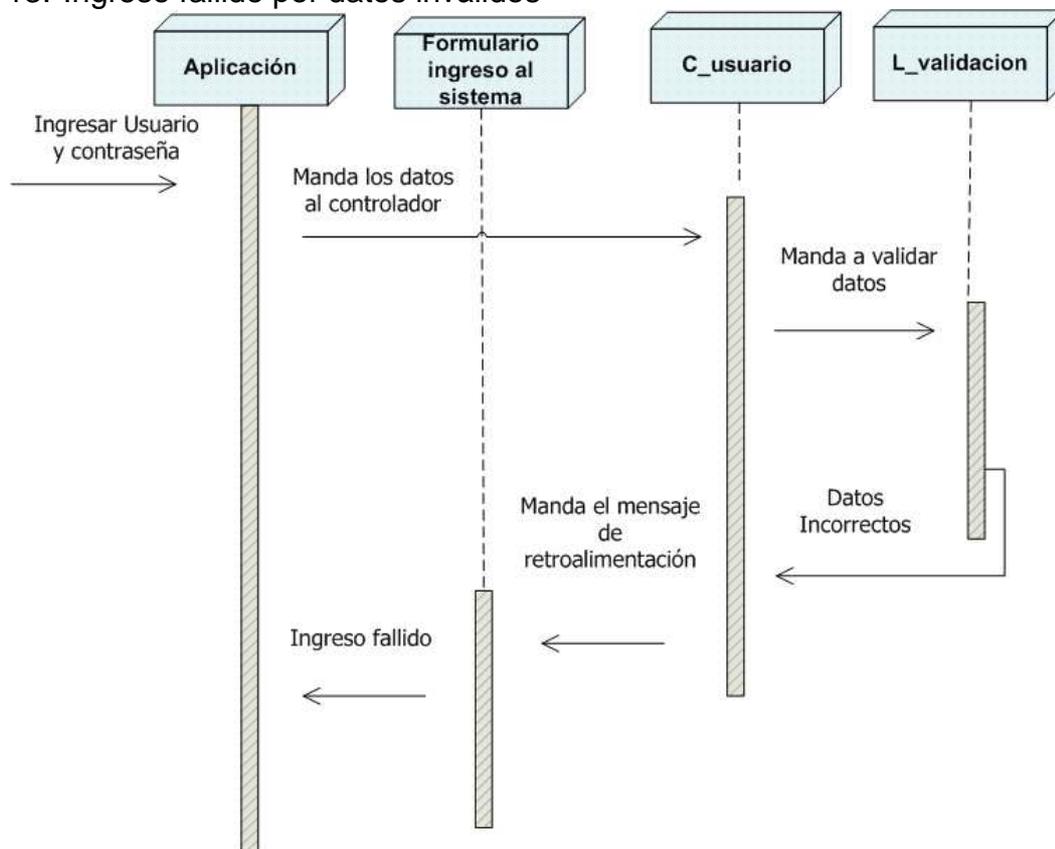
## 11. Ingreso exitoso al sistema



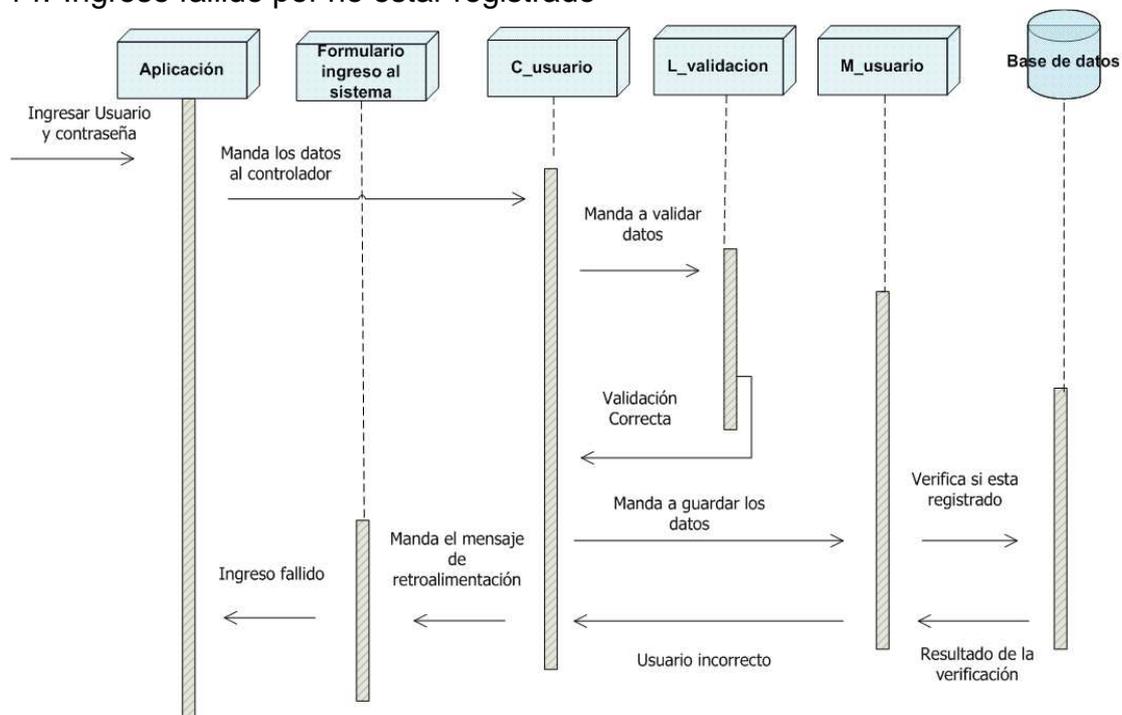
## 12. Ingreso fallido por indisponibilidad de base



## 13. Ingreso fallido por datos inválidos



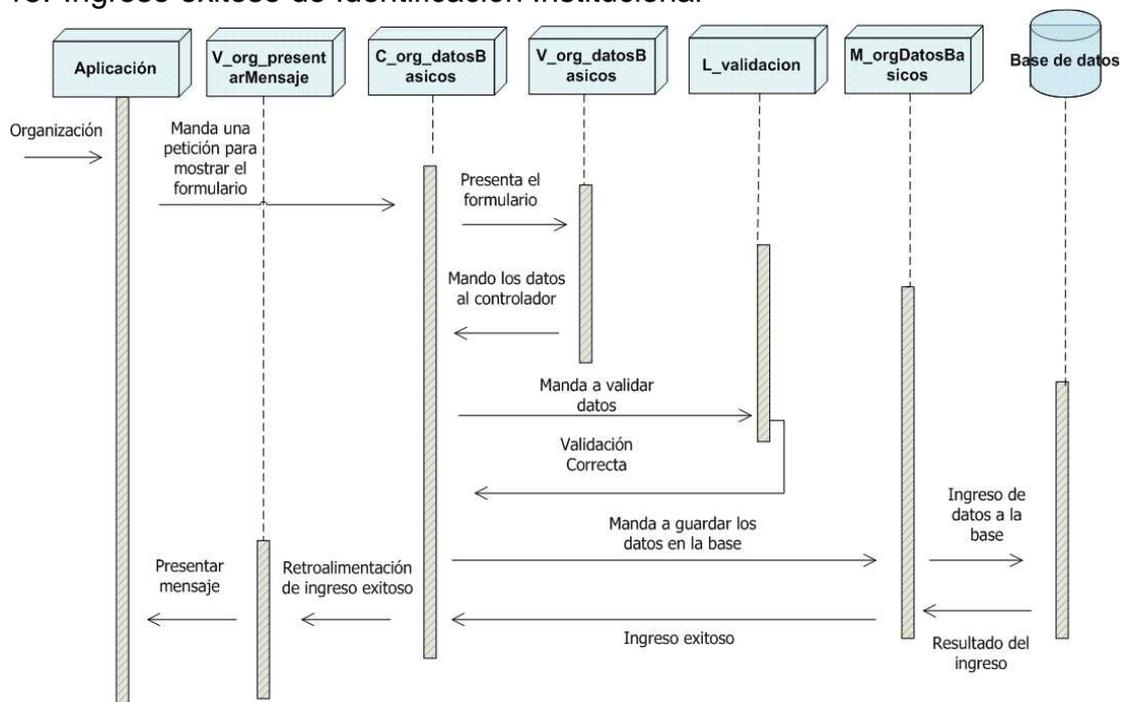
## 14. Ingreso fallido por no estar registrado



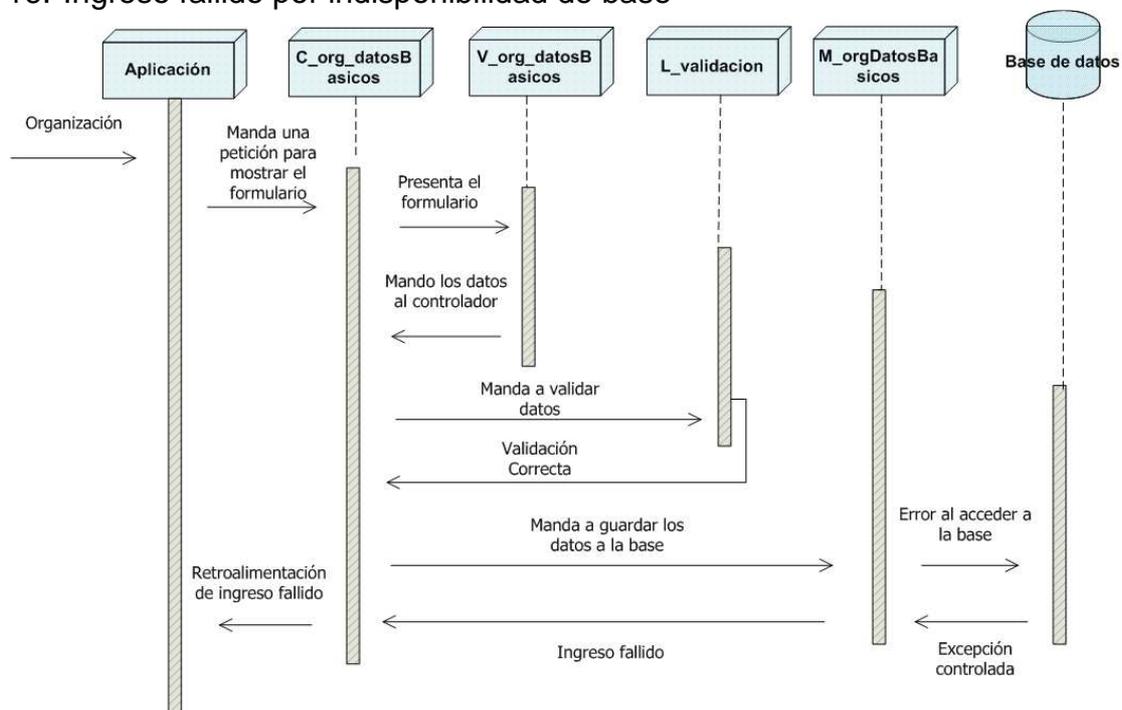
## Módulo de Identificación Institucional

### 5.3.1.2 CU05: Ingresar Identificación Institucional

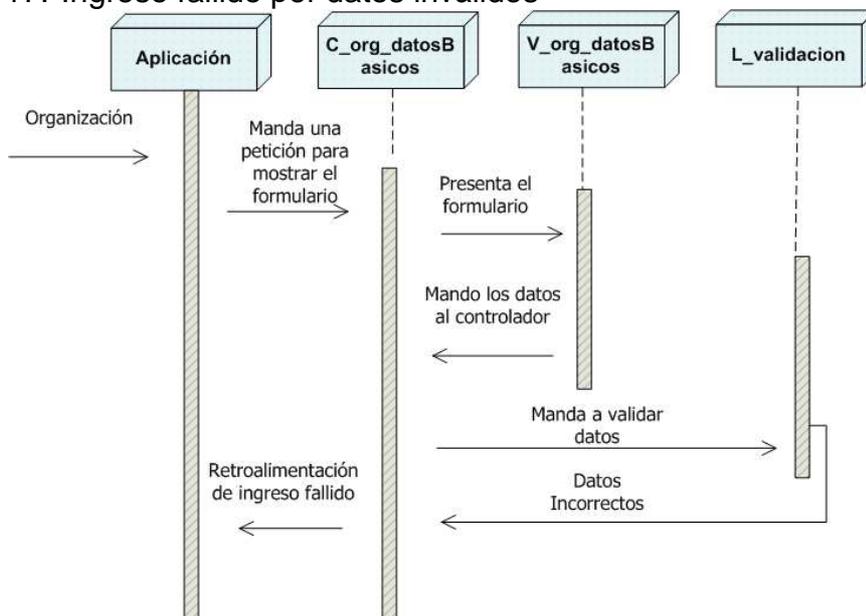
#### 15. Ingreso exitoso de Identificación Institucional



#### 16. Ingreso fallido por indisponibilidad de base

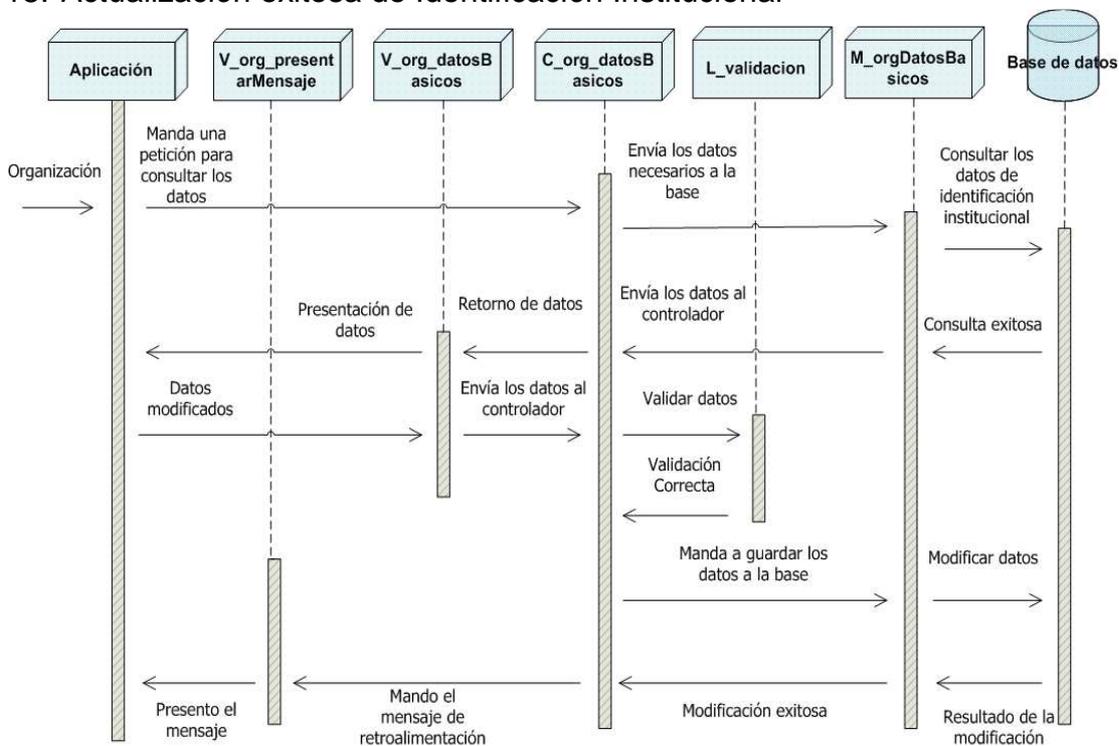


### 17. Ingreso fallido por datos inválidos

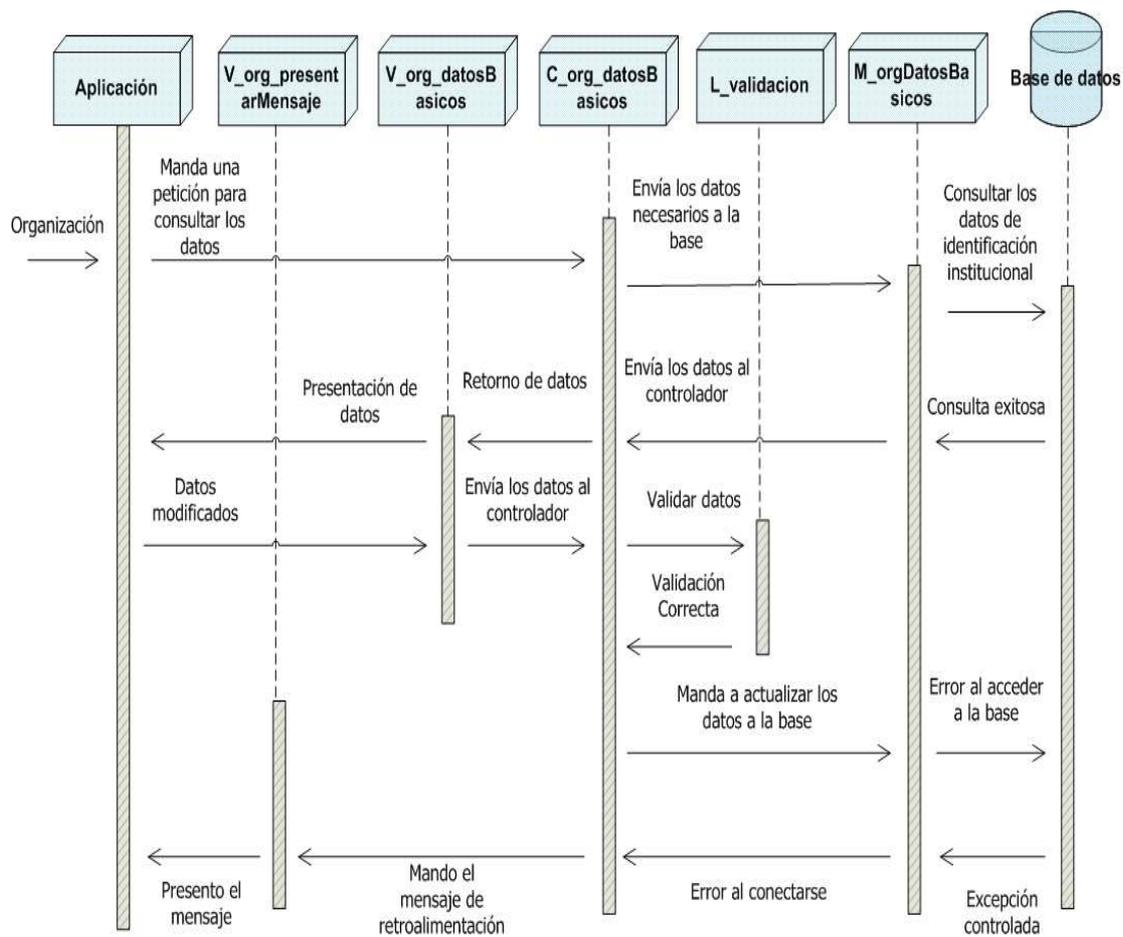
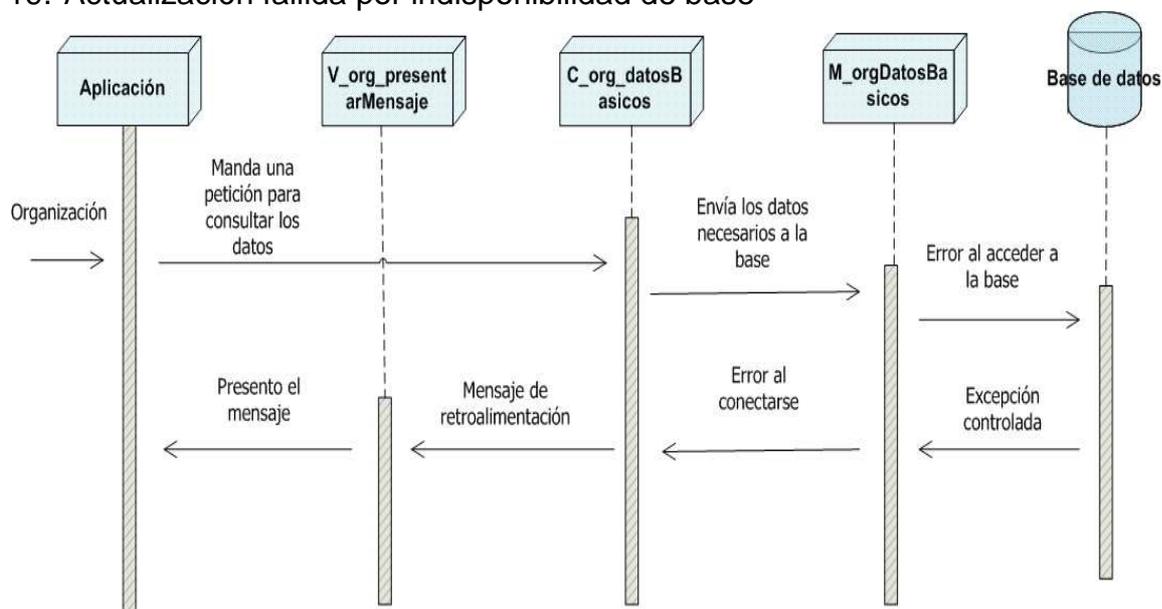


#### 5.3.1.3 CU06: Actualizar Identificación Institucional

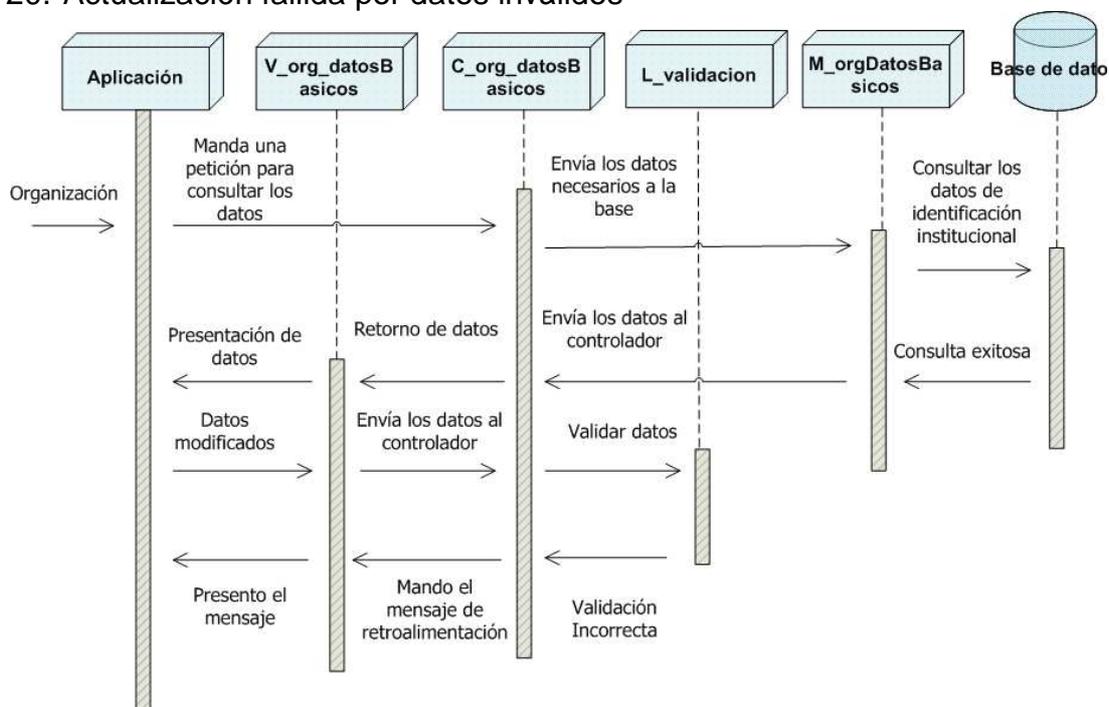
### 18. Actualización exitosa de Identificación Institucional



19. Actualización fallida por indisponibilidad de base



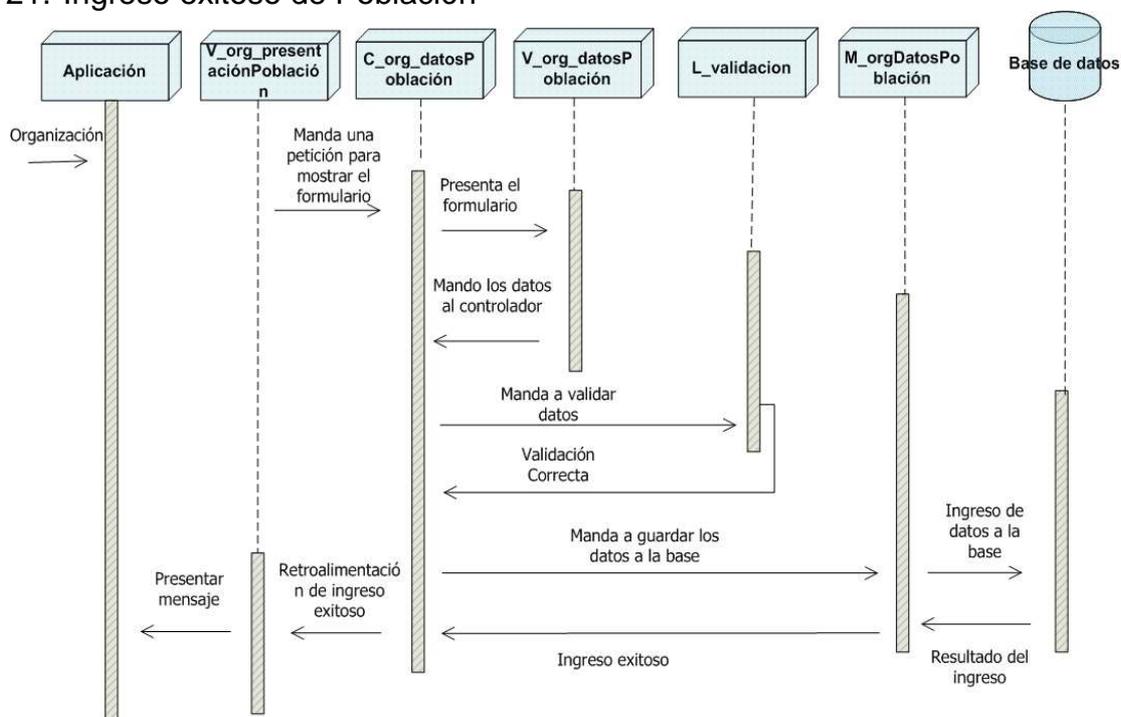
### 20. Actualización fallida por datos inválidos



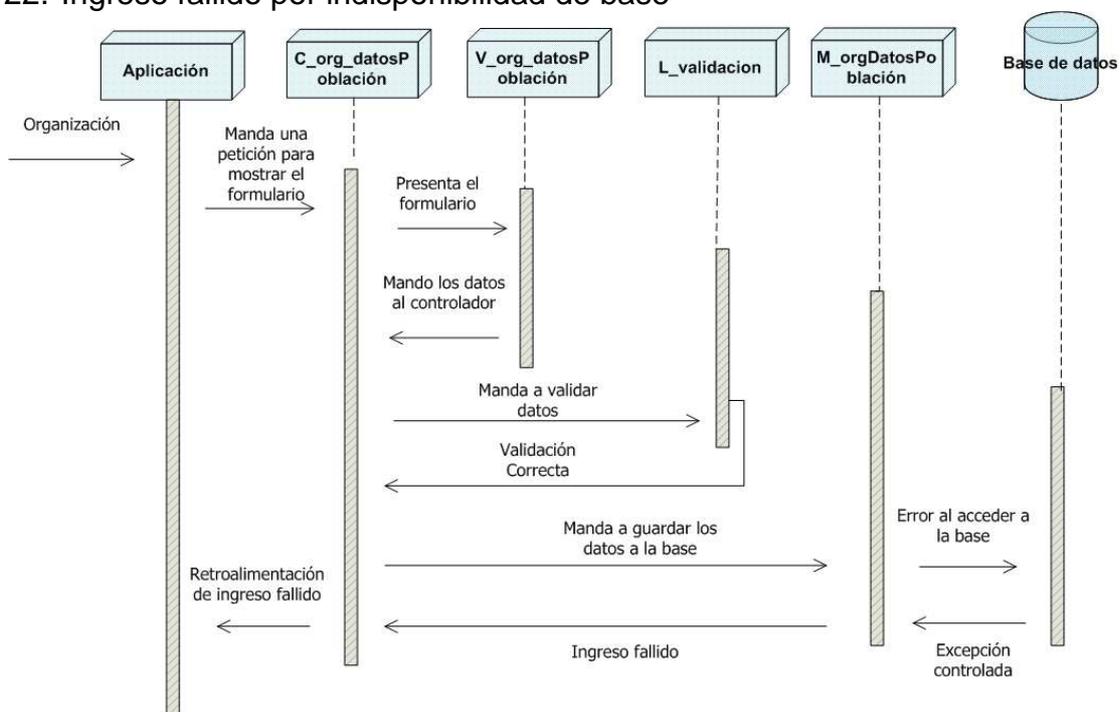
### Módulo de Población

#### 5.3.1.4 CU07: Ingresar Población

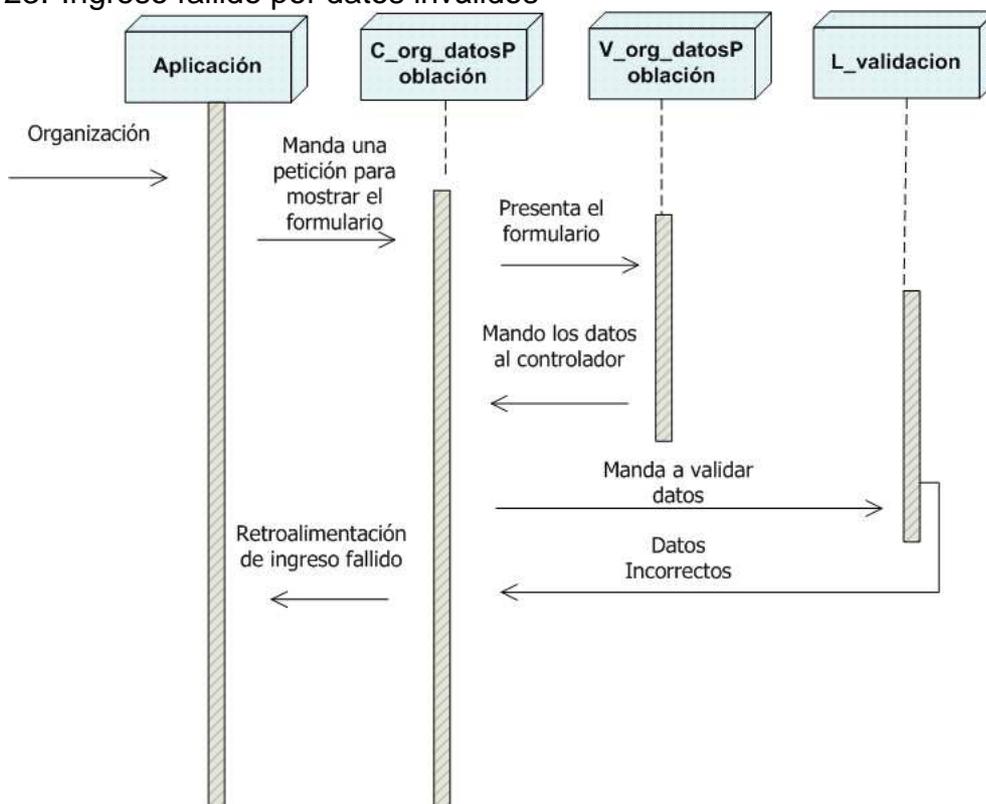
### 21. Ingreso exitoso de Población



### 22. Ingreso fallido por indisponibilidad de base

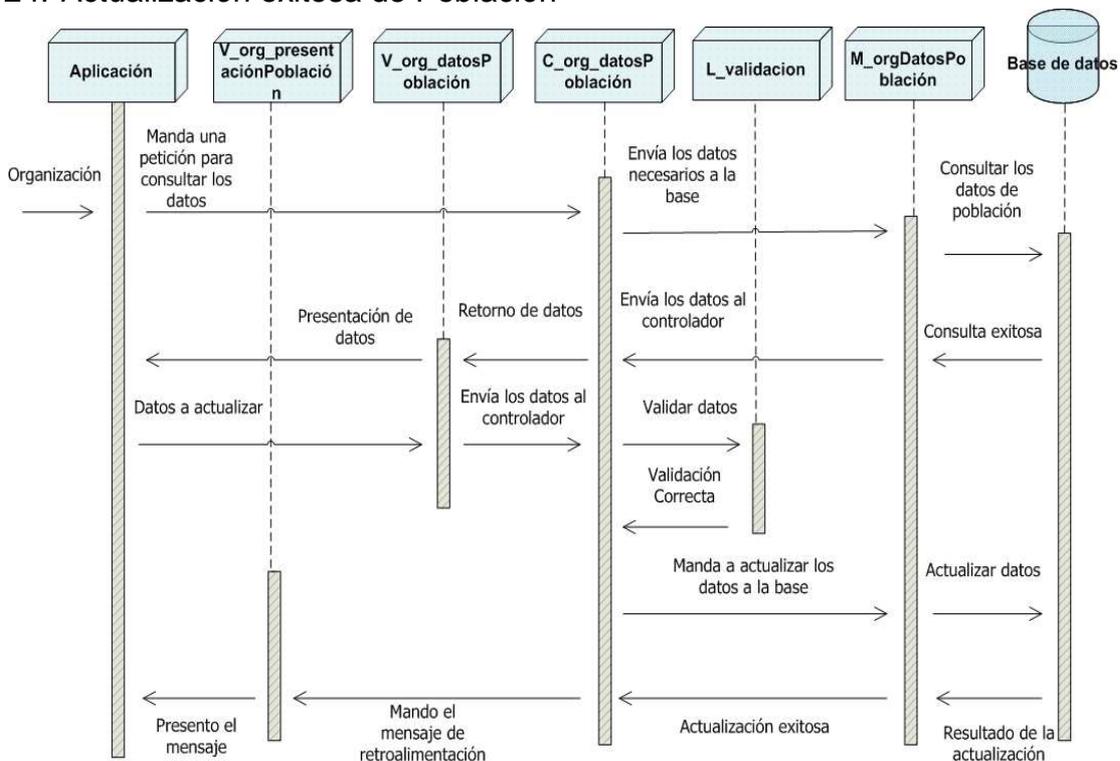


### 23. Ingreso fallido por datos inválidos

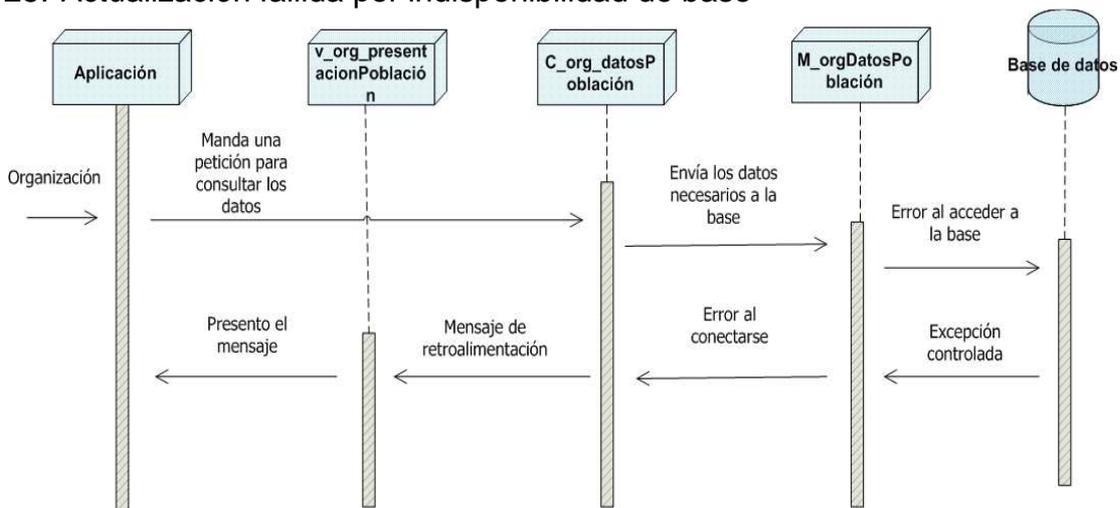


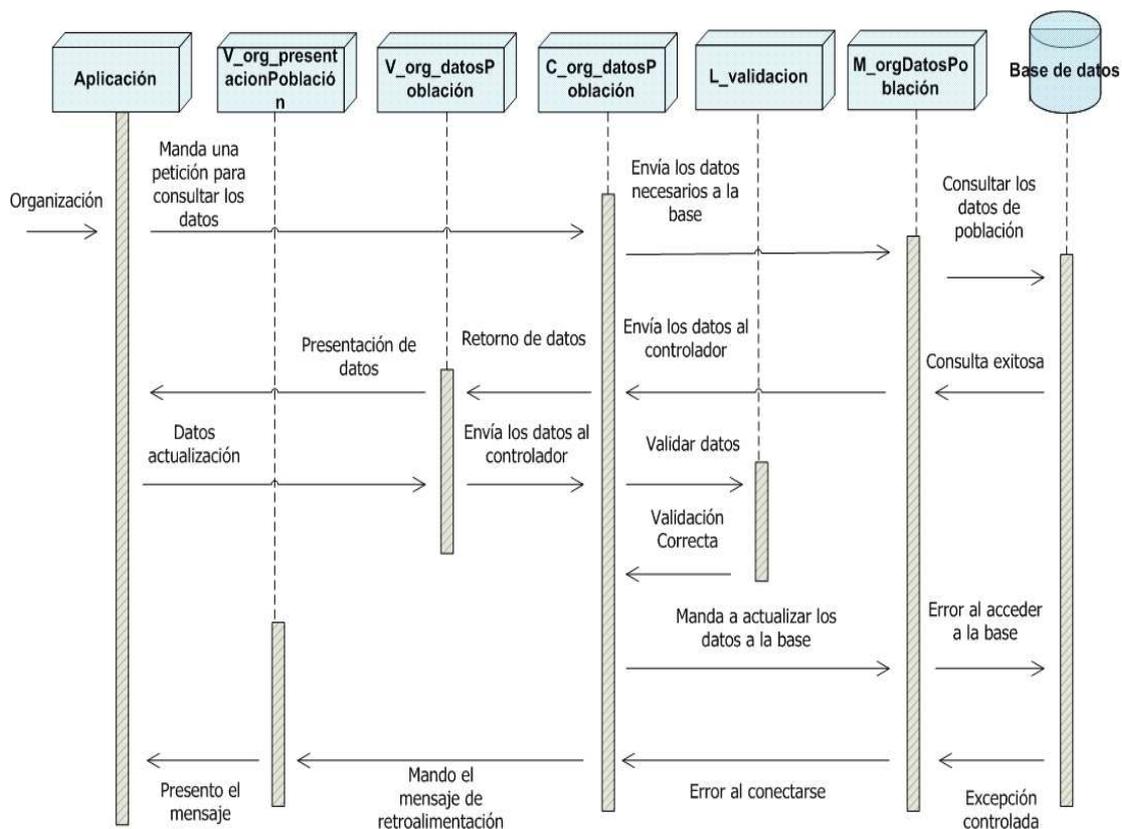
5.3.1.5 CU08: Actualizar Población

24. Actualización exitosa de Población

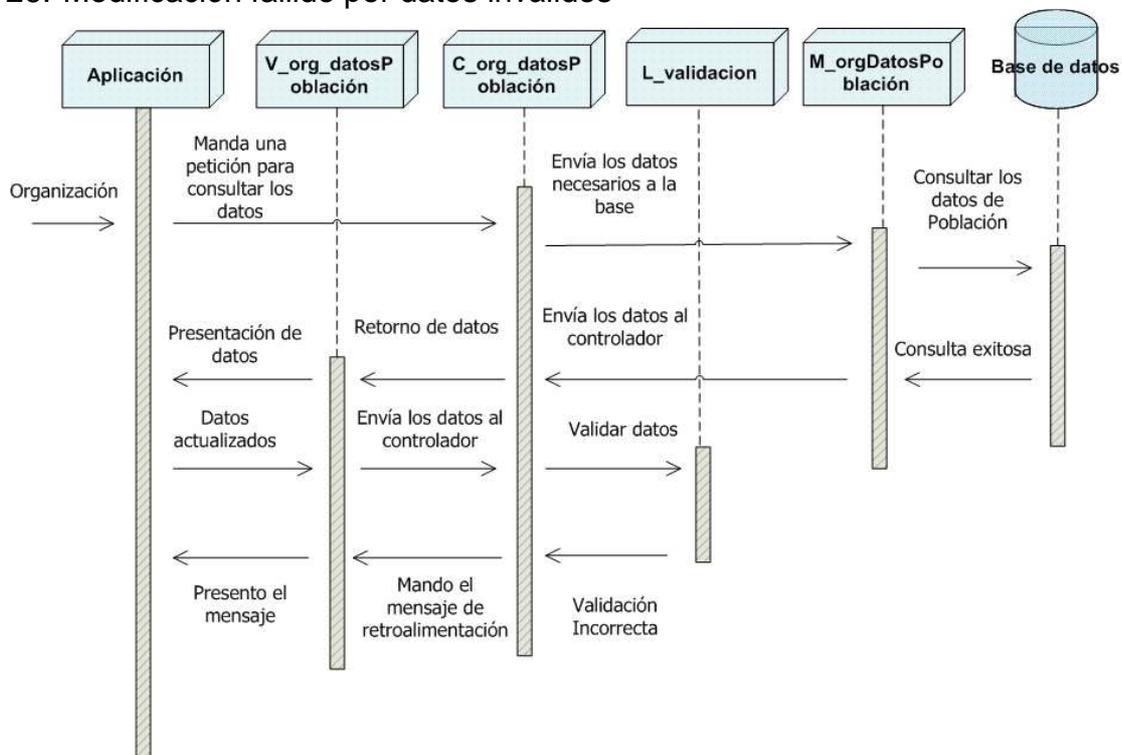


25. Actualización fallida por indisponibilidad de base





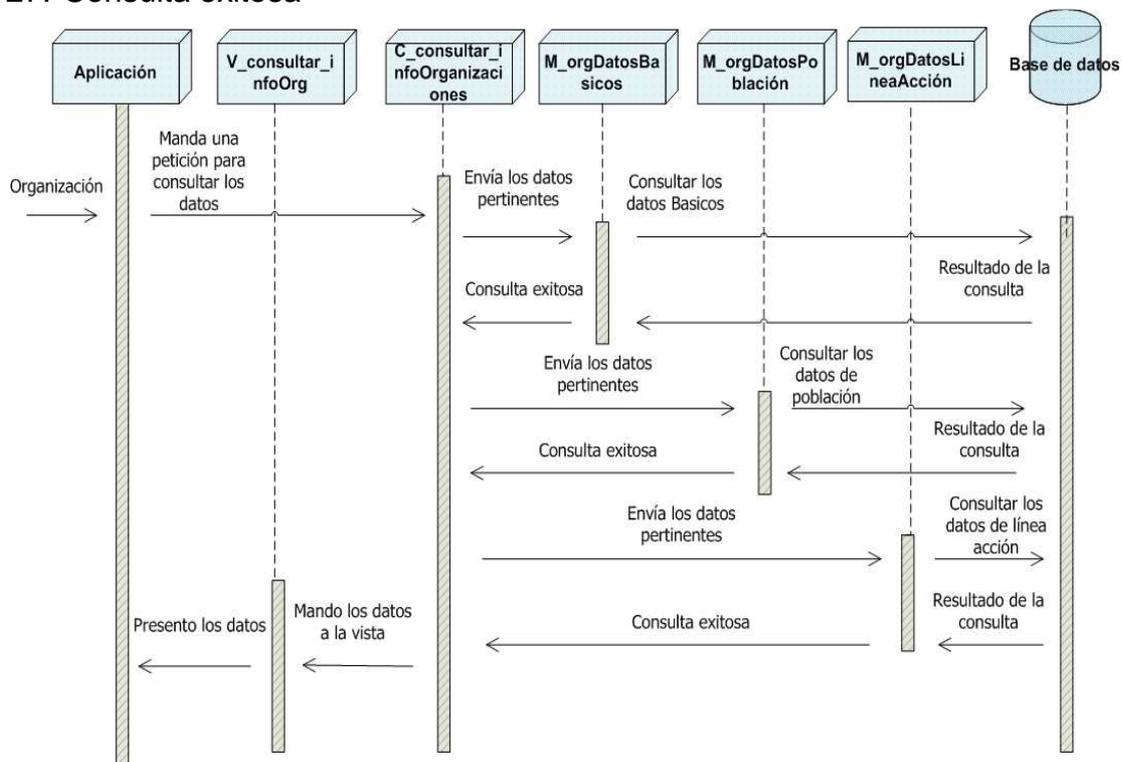
26. Modificación fallido por datos inválidos



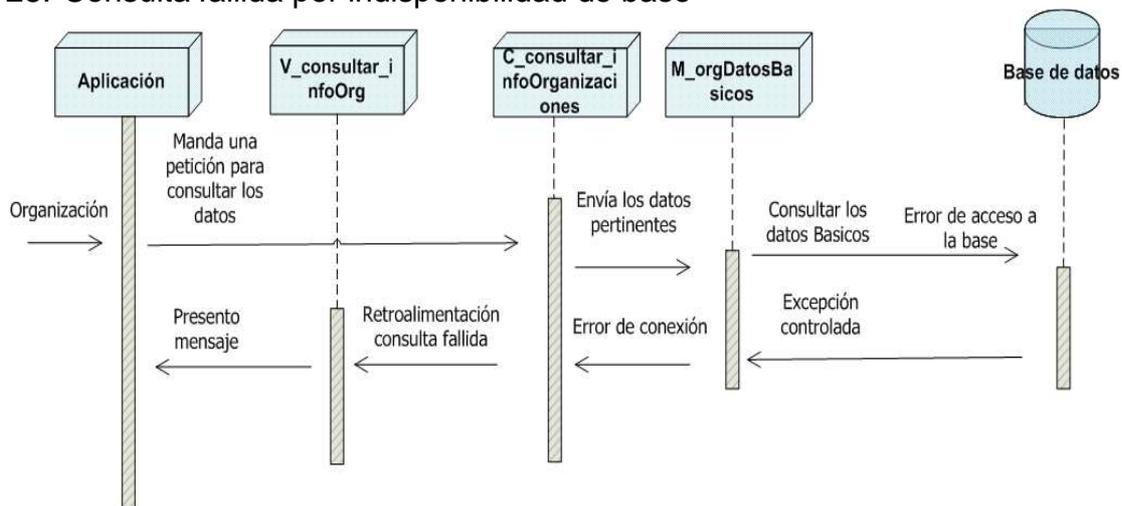
## Módulo de Consultas

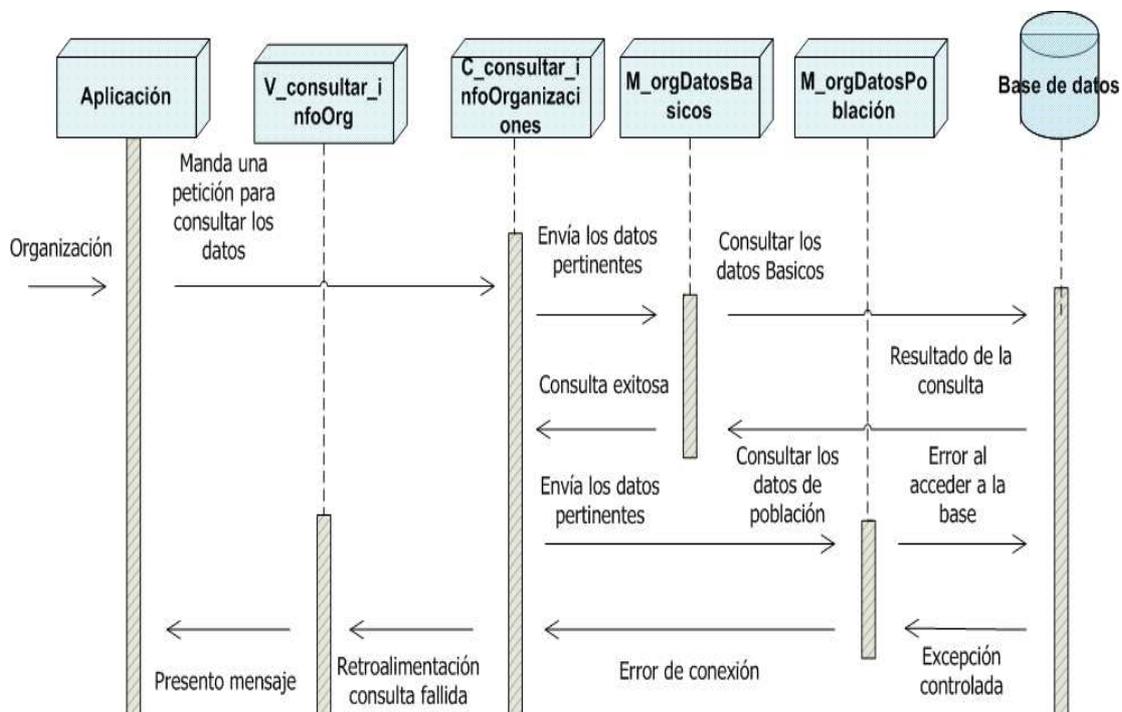
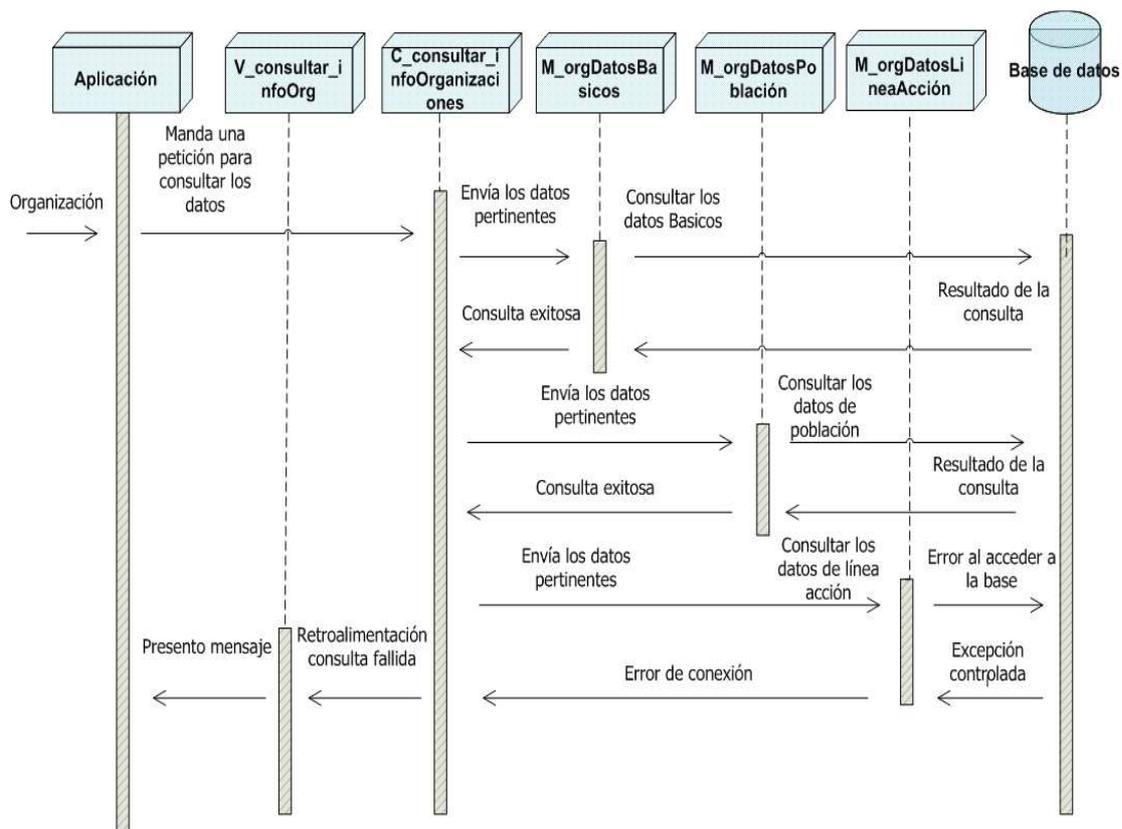
### 5.3.1.6 CU09: Consulta de datos de las Organizaciones

#### 27. Consulta exitosa



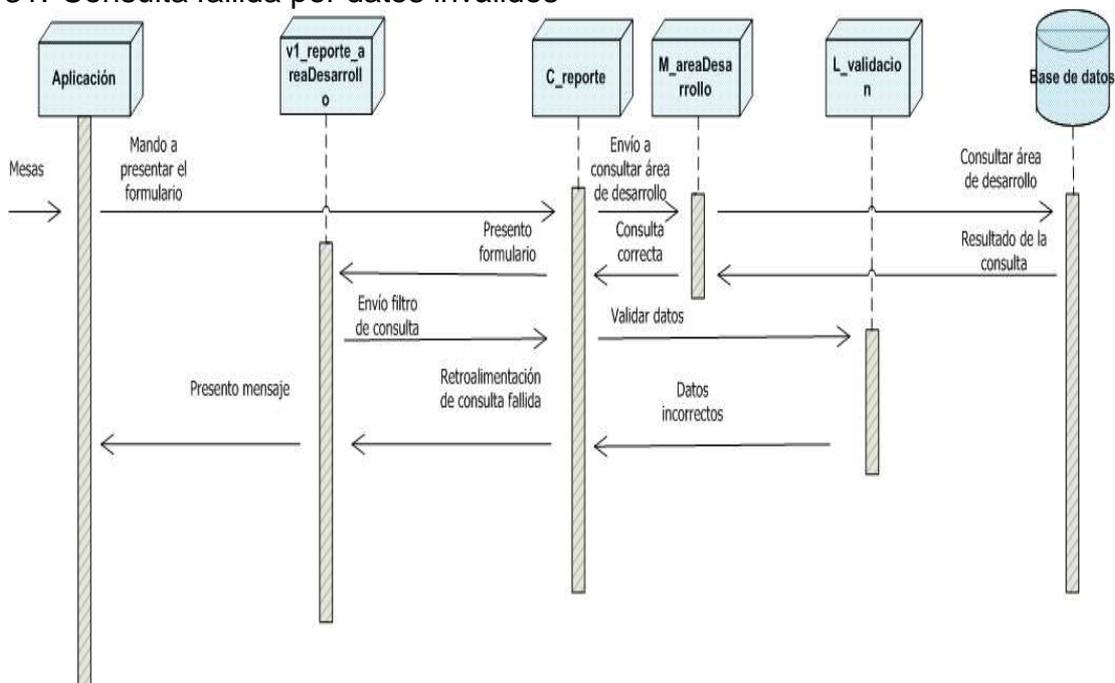
#### 28. Consulta fallida por indisponibilidad de base





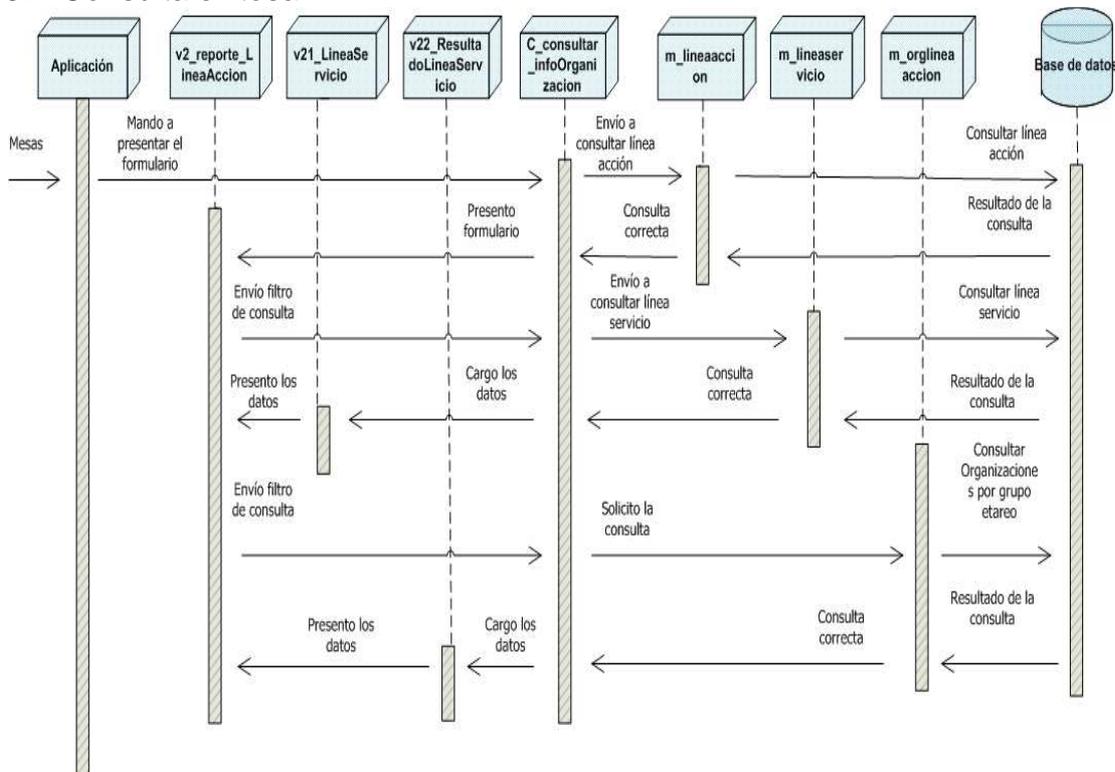


### 31. Consulta fallida por datos inválidos

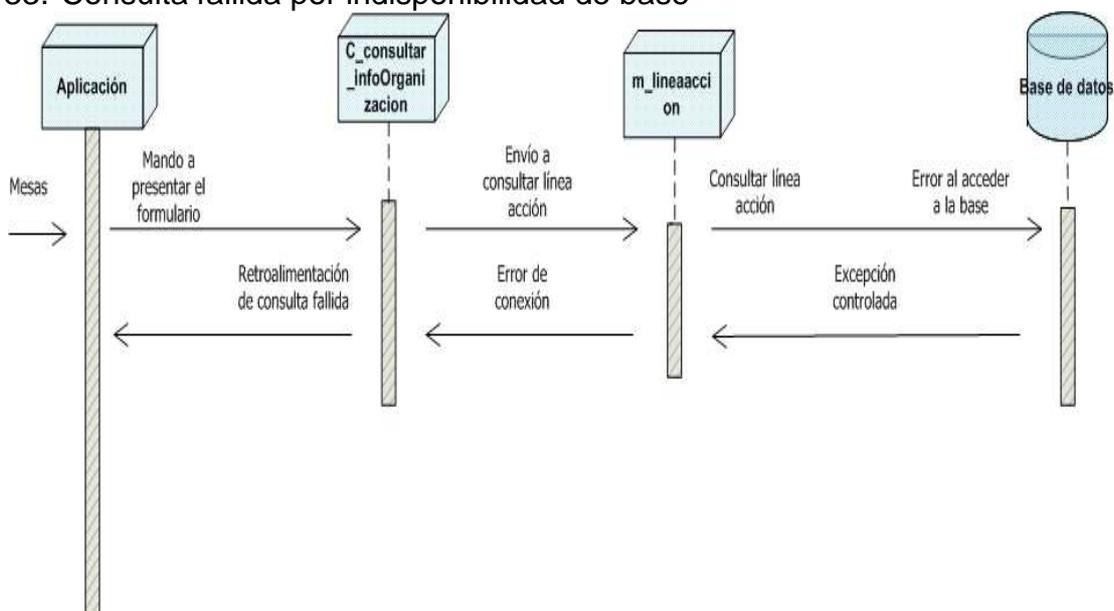


#### 5.3.1.8 CU11: Reporte de Organizaciones por Línea de Acción

### 32. Consulta exitosa

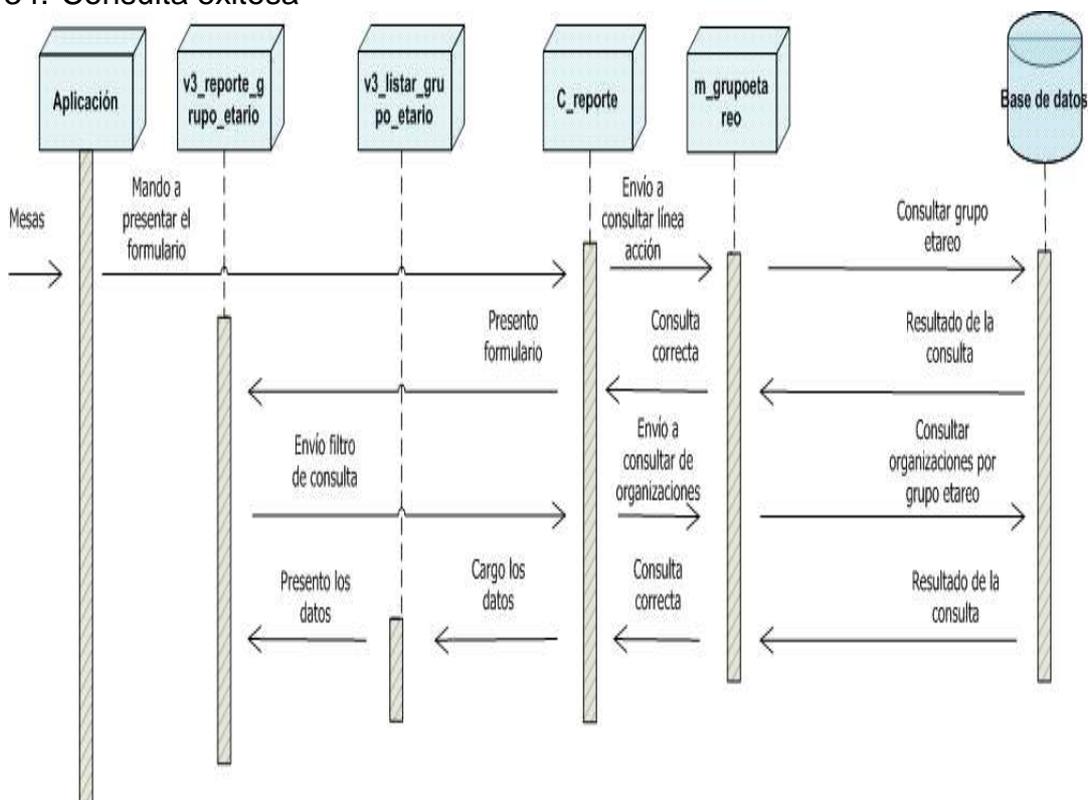


### 33. Consulta fallida por indisponibilidad de base

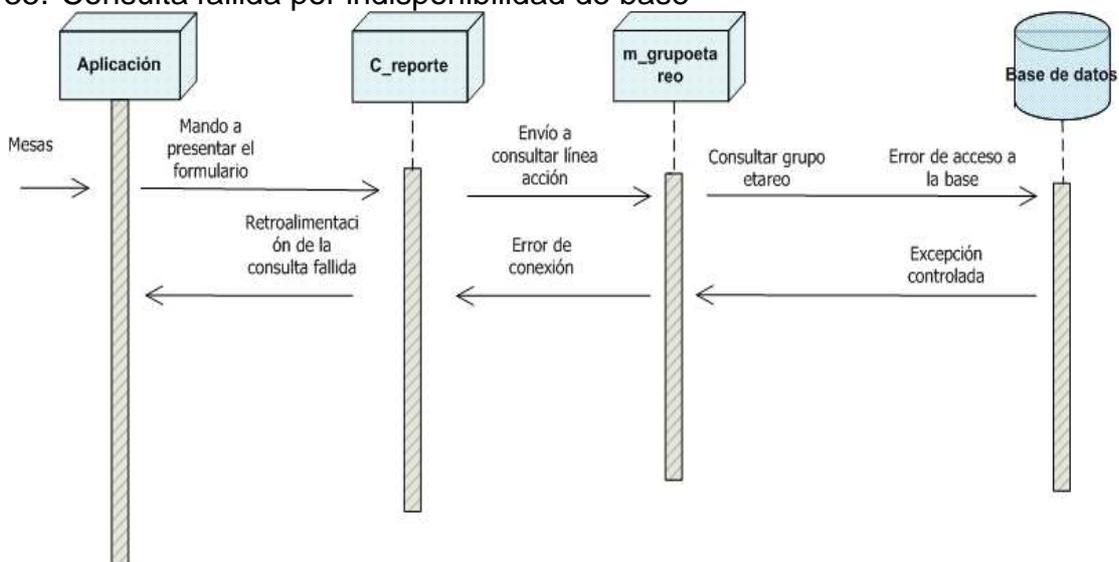


#### 5.3.1.9 CU12: Reporte de Organizaciones por Grupo Etario

### 34. Consulta exitosa

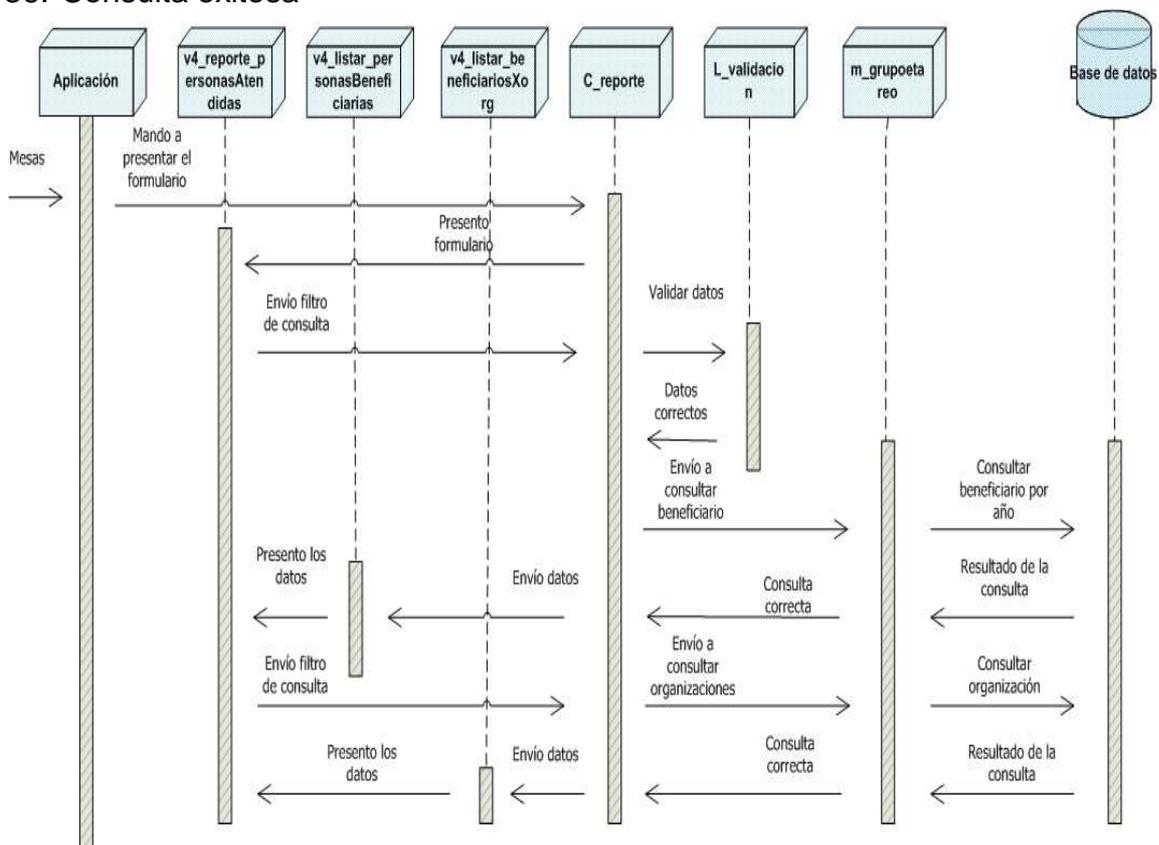


35. Consulta fallida por indisponibilidad de base

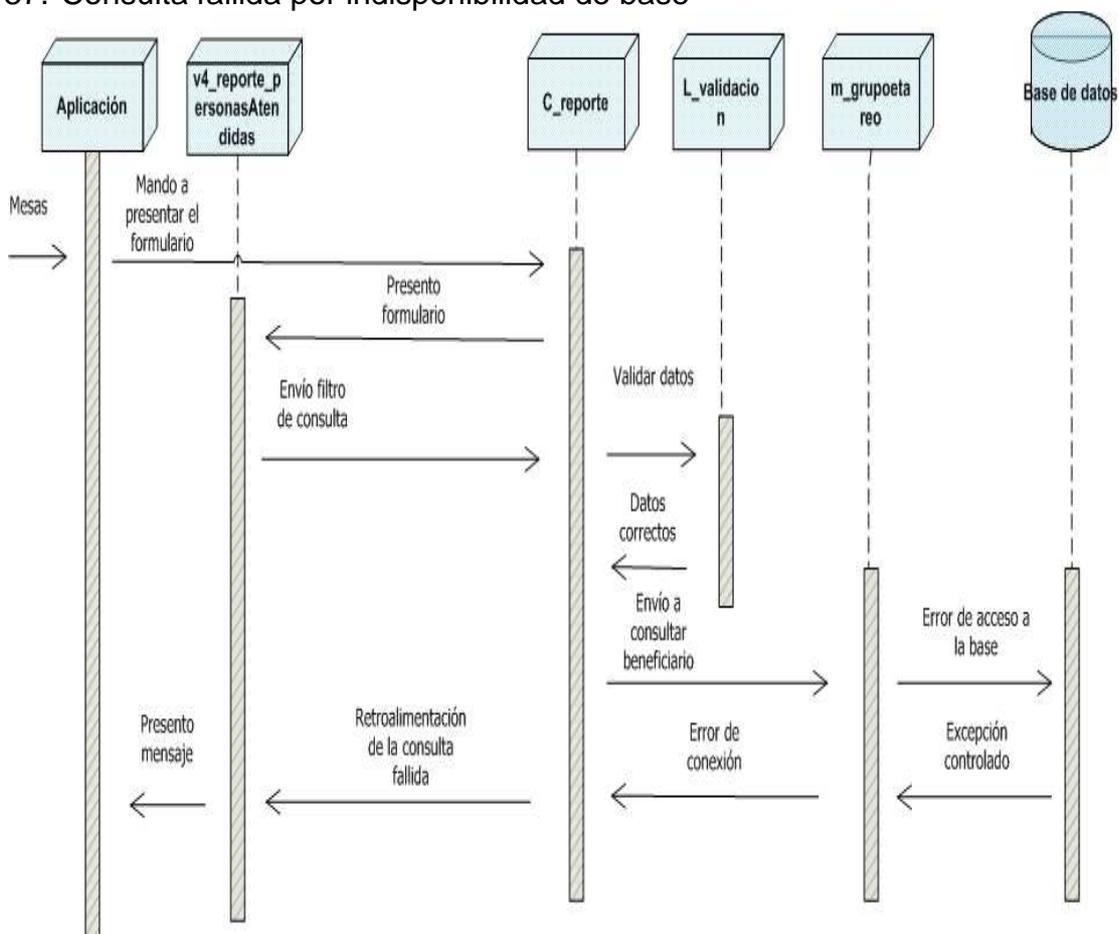


5.3.1.10 CU13: Reporte de Personas Atendidas

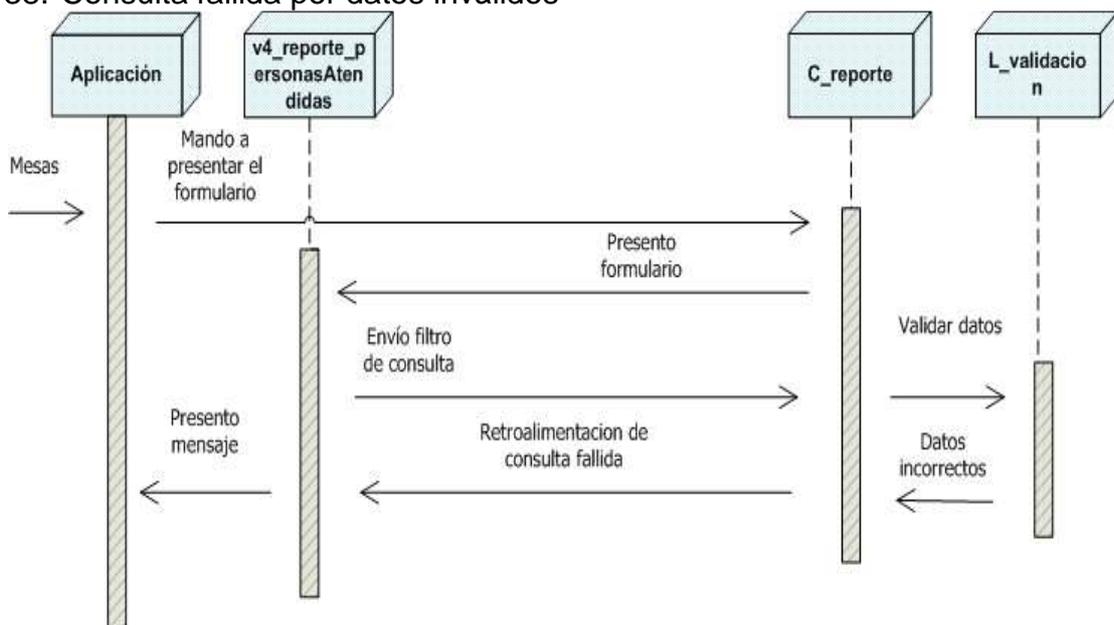
36. Consulta exitosa



37. Consulta fallida por indisponibilidad de base

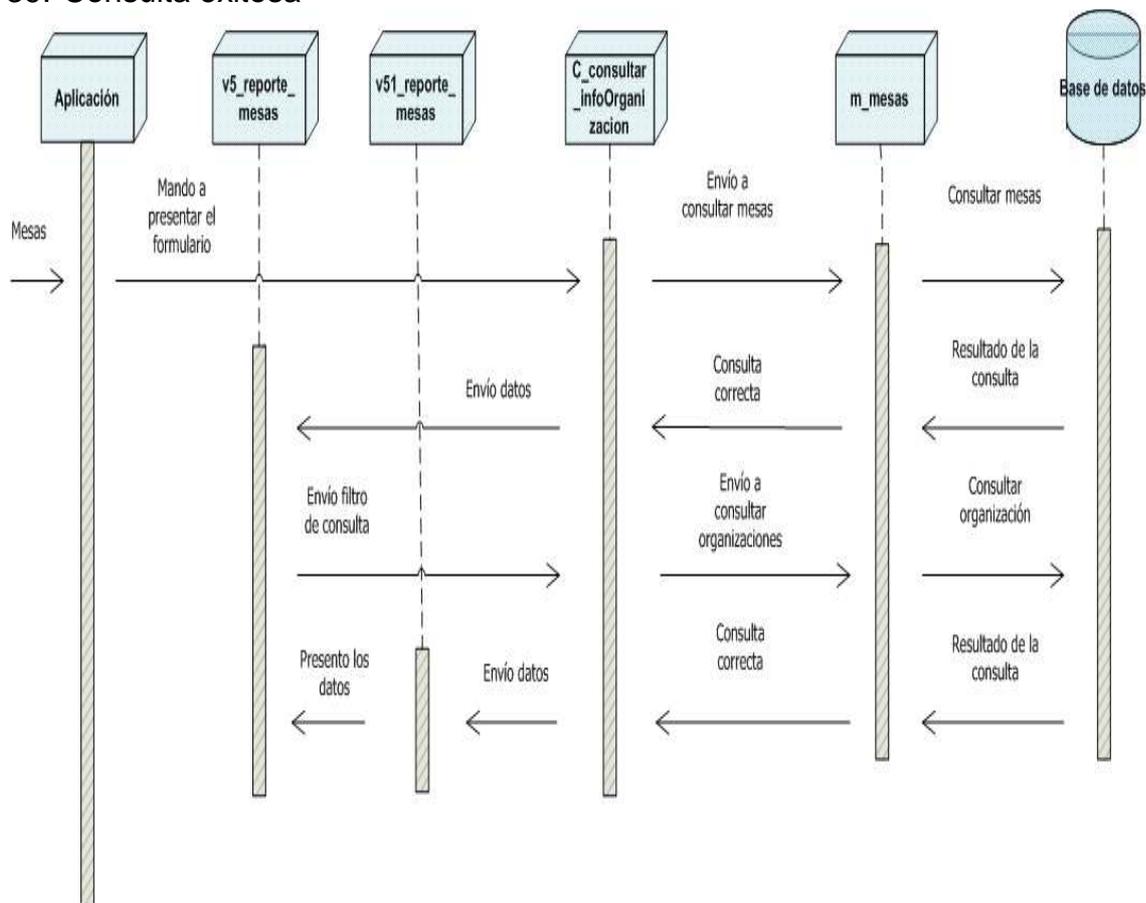


38. Consulta fallida por datos inválidos

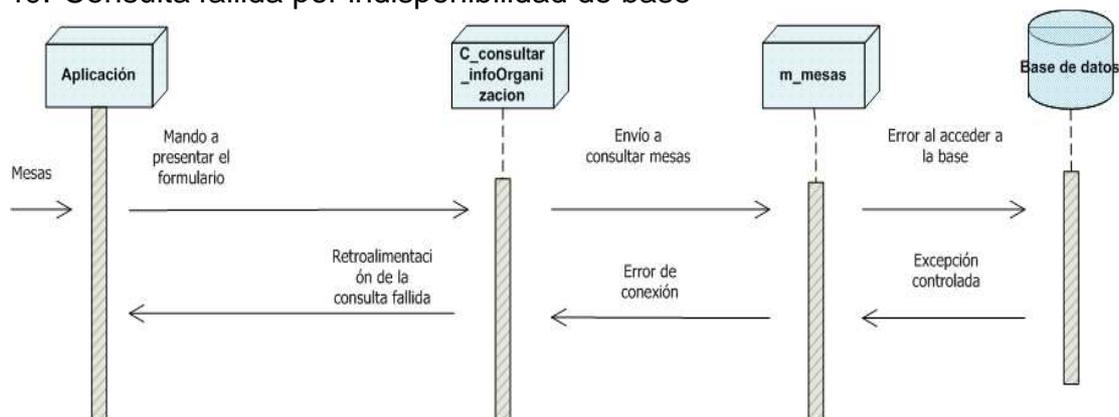


5.3.1.11 CU14: Reporte de Organizaciones por Mesa de Concertación

39. Consulta exitosa

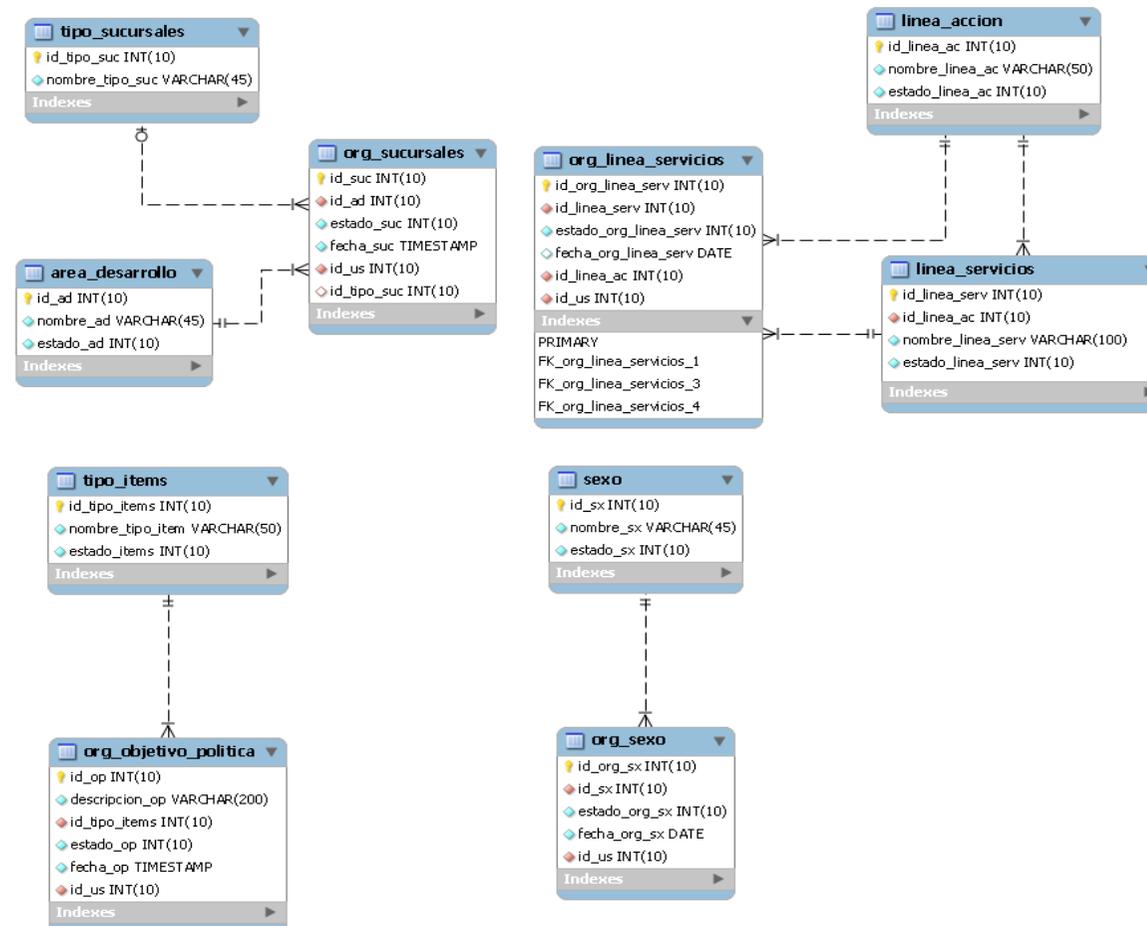


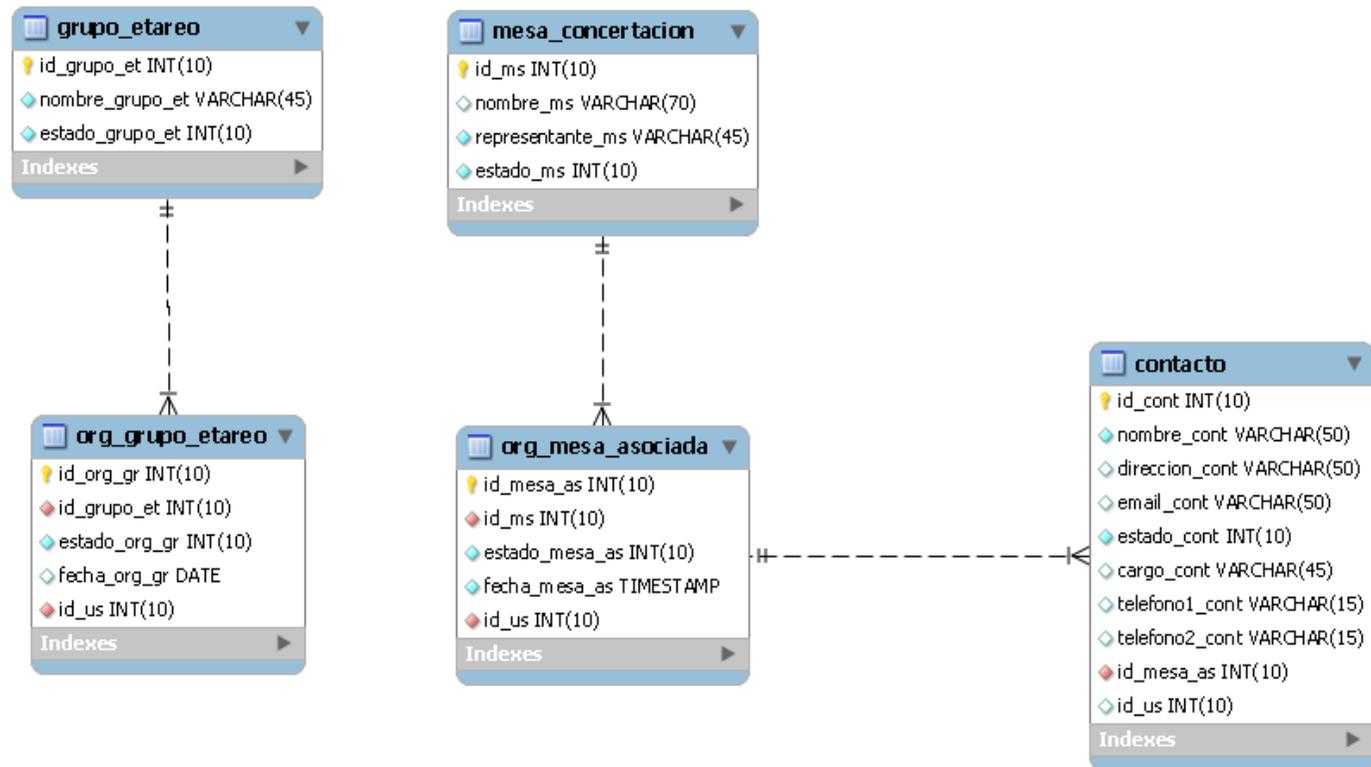
40. Consulta fallida por indisponibilidad de base

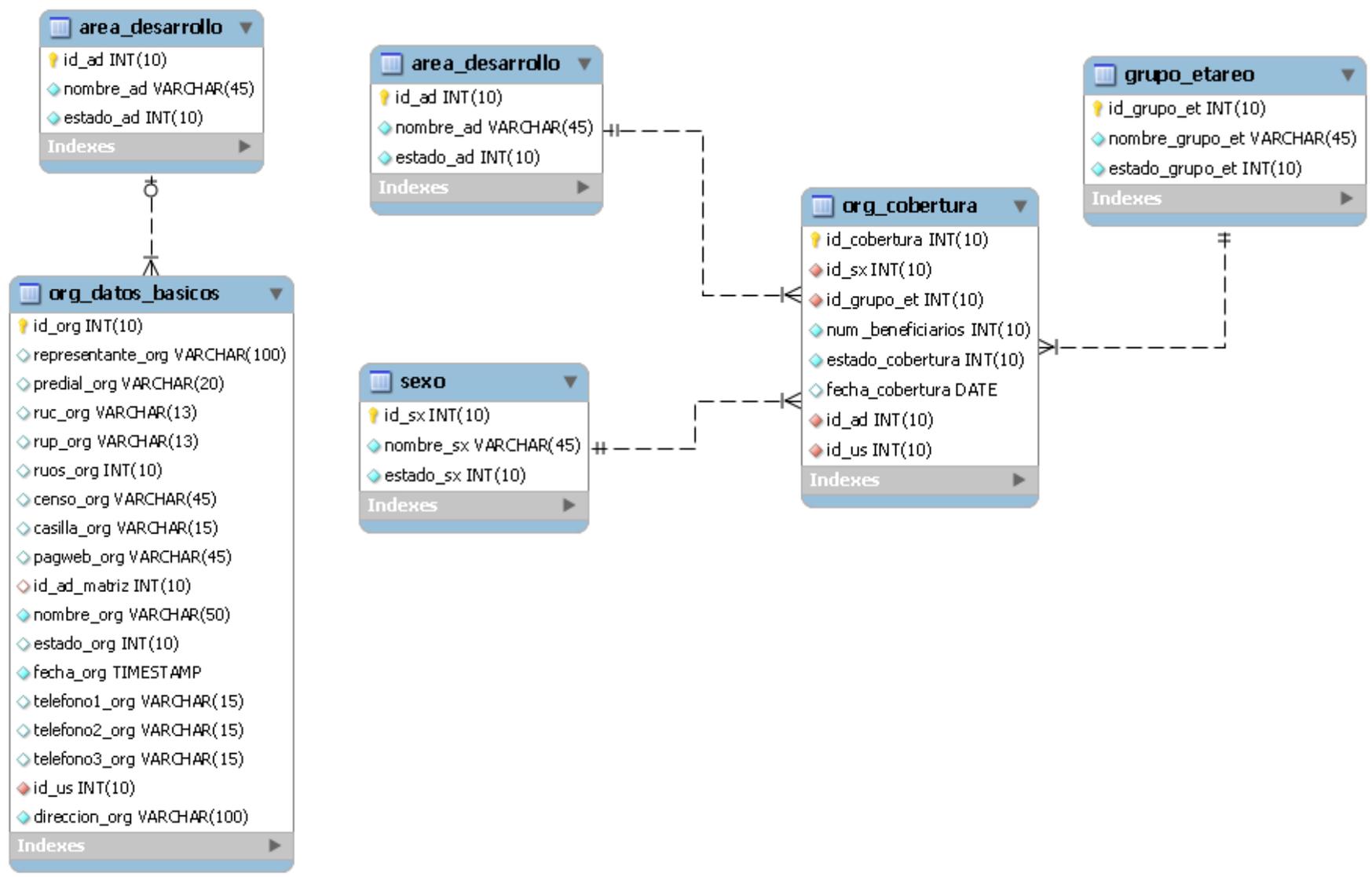


## ANEXO DE DIAGRAMAS DE BASE DE DATOS

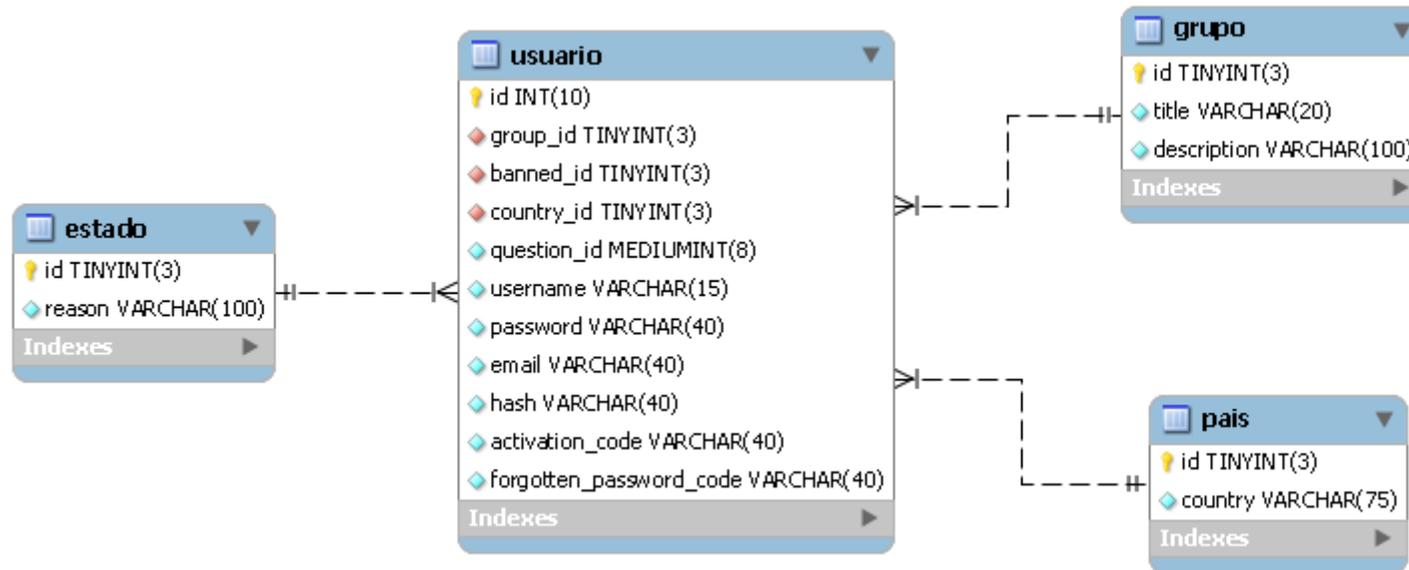
### DIAGRAMA DE TABLAS DE INFORMACIÓN BÁSICA ASOCIADAS A LAS TABLAS DE LA ORGANIZACIÓN



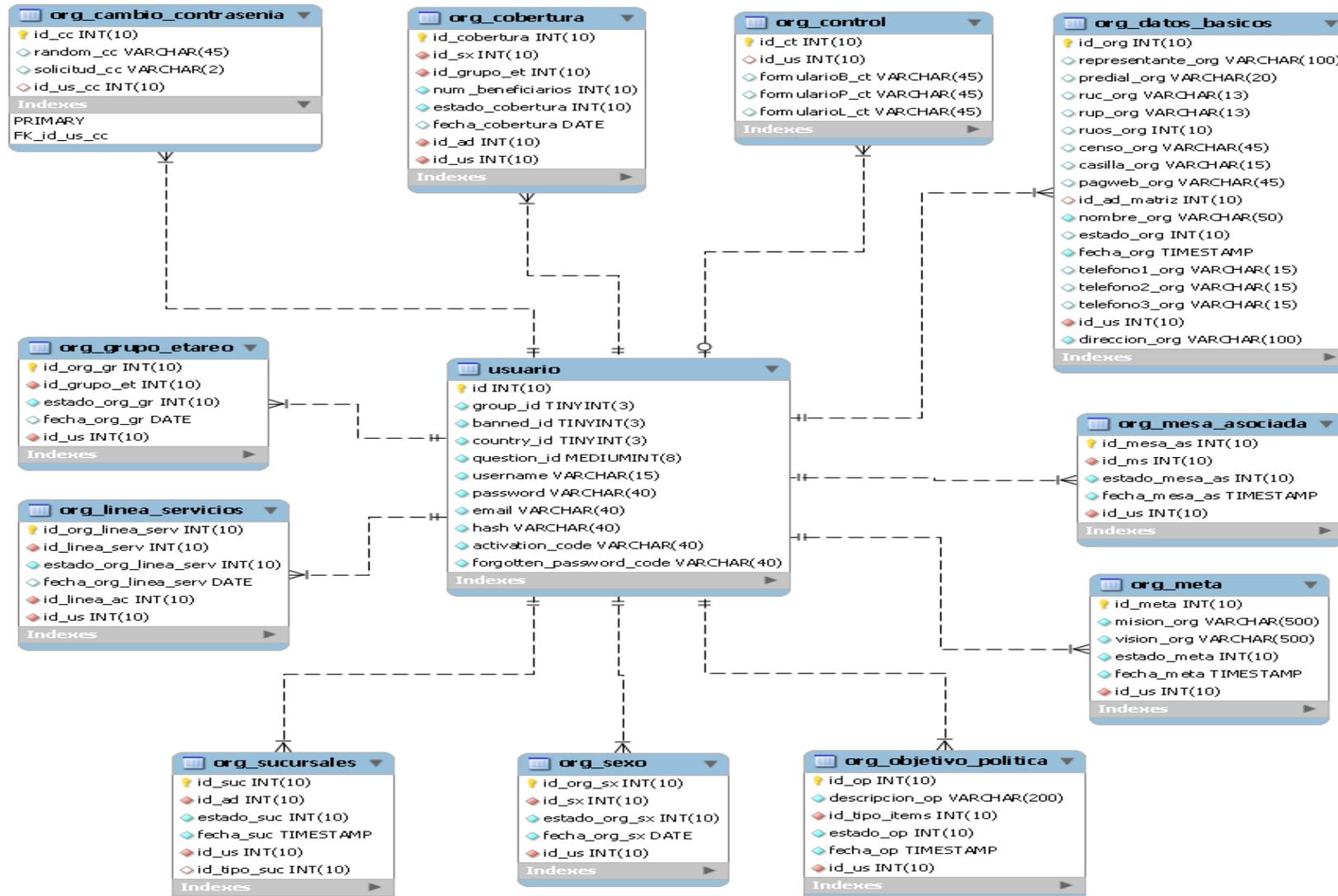




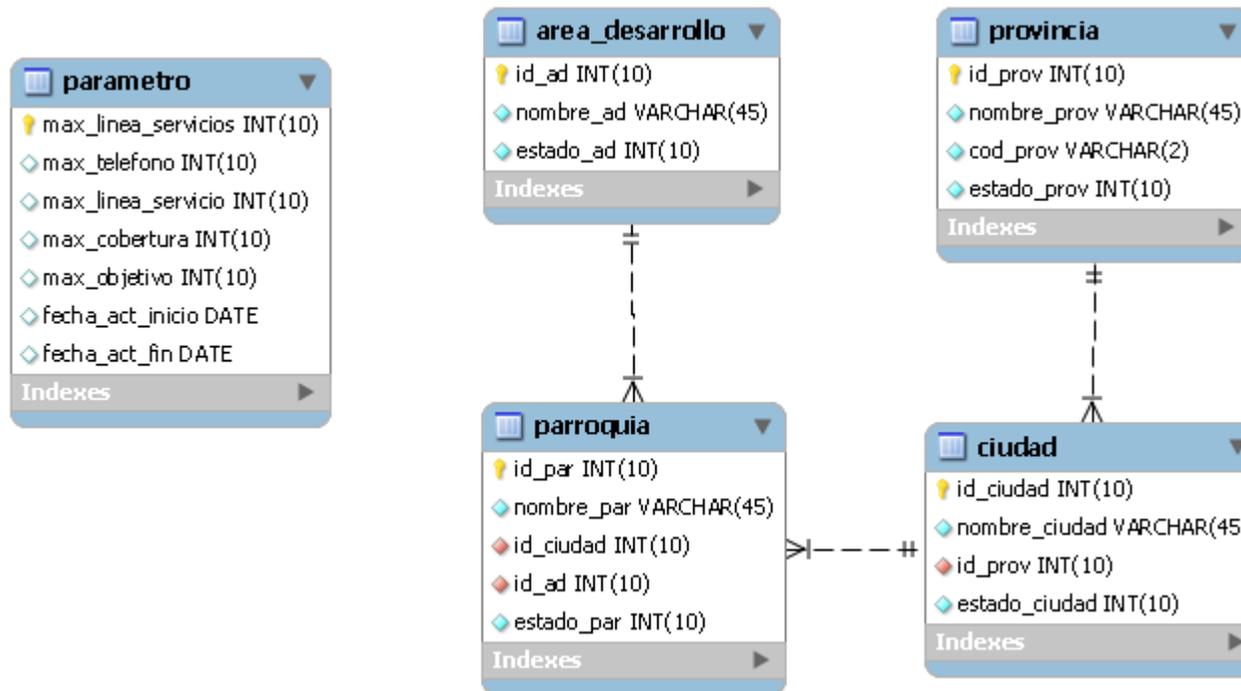
## DIAGRAMA DE TABLAS DE SEGURIDAD



## DIAGRAMA ASOCIADO A LA ORGANIZACIÓN



## DIAGRAMA DE TABLAS DE INFORMACIÓN BÁSICA



## ANEXO DISEÑO DETALLADO

En el sistema Red de Información de Organizaciones Sociales se utiliza como modelo de programación el MVC.

**El modelo vista controlador (MVC)** es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

### Estructura del Sistema

- Librería: L\_nombreClase
- Vista: V\_nombreClase
- Controlador : C\_nombreClase.
- Modelo : M\_nombreClase.
- 

### Modulo de Seguridad

- Registrar Organización
  - C\_usuario
    - Obtener datos de la vista registrarUsuario.
    - Manda mensajes de retroalimentación a la vista.
  - L\_usuario
    - Interactuar entre controlador C\_usuario y el modelo M\_usuario.
    - Manda datos al modelo para que consulte a la base y manda los resultados al controlador.
  - M\_usuario
    - Interactúa con los datos que envía la librería y manda la consulta a la base de datos.
  - L\_validación
    - Validar datos de entrada del usuario.
  - V\_registrarUsuario
    - Presentar formulario de registro.
- Ingresar al Sistema
  - C\_usuario

- Crear la sesión del usuario.
- V\_ingresarSistema
  - Presenta formulario de ingreso al sistema.

### **Modulo Identificación Institucional**

- Ingresar y modificar Identificación Institucional
  - Datos Básicos, Contactos Estrategia, Ubicación y Mesas Asociadas.
    - C\_org\_datosBasicos
      - Obtener los datos de la vista V\_org\_datosBasicos.
      - Mandar a validar lo ingresado por el usuario a la librería L\_validacion.
      - Setear un mensaje de retroalimentación para luego ser enviado a la vista V\_org\_datosBasicos.
    - L\_org\_datosBasicos
      - Manda a validar los datos ingresados por el usuario.
    - L\_validación
      - Validar la información enviada por el controlador C\_org\_datosBasicos, y retorna su resultado hacia el mismo controlador.
    - M\_datosBasicos
      - Ingresa los datos ya validados.
    - V\_org\_datosBasicos
      - Presentar el formulario para el registro de la Organización.
      - En caso de ingresar mal algún dato, presenta los mensajes de retroalimentación.

### **Modulo de Población**

- Ingresar y modificar Población
  - Sexo , Grupo Etario y Cobertura Institucional
    - C\_org\_datosPoblación
      - Obtener información de la base de datos por medio de la librería L\_org\_datosPoblación.
      - Enviar información obtenida a la vista V\_org\_datosPoblación.
      - Validar información obtenida por el usuario.
    - L\_org\_datosPoblación
      - Interactuar con la base y retorna el resultado al controlador C\_org\_datosPoblación.
    - V\_org\_datosPoblación

- Presentar el formulario al usuario para se respectivo registro de población que atienden.

### **Modulo de Línea Acción y Servicio**

- Ingresar y modificar Línea Acción y Servicio
  - C\_org\_lineaServicios
    - Obtener la información de la vista V\_org\_lineaServicios y enviarla a la librería L\_org\_lineaServicios.
    - Setear el mensaje que se va ha presentar como retroalimentación a la vista V\_org\_lineaServicio.
  - L\_org\_lineaServicios
    - Se encarga de mandar a validar los datos ingresados por el usuario.
  - M\_orgPoblacion
    - Interactuar con la base de datos y luego retornar el resultado a la librería.
  - V\_org\_lineaServicios
    - Presentar el formulario para registrar que línea de acción y servicio tiene las organizaciones.

### **Modulo de Consultas**

- Identificación Institucional, Población y Línea de Acción y Servicio.
  - C\_org\_consultaInformacionOrganizacion
    - Obtener la información de la organización de las librerías L\_org\_datosBasicos, L\_org\_datosPoblación y L\_org\_lineaServicios.
    - Enviar la información obtenida a la vista V\_org\_consultarInformacionOrganización.
  - L\_org\_datosBasicos
    - Interactuar con el módulo M\_datosBasicos para obtener la información de las tablas que me den los datos básicos de la organización.
  - L\_org\_datosPoblación
    - Interactuar con el módulo M\_orgPoblacion para obtener la información de las tablas que me den los datos de la población que atiende la organización.
  - L\_org\_lineaServicios
    - Interactuar con el módulo M\_lineaAccion para obtener la información de las tablas que me den los datos de línea de acción y servicio que prestan la organización.
  - V\_org\_consultarInformacionOrganización
    - Presentar la información de la organización.

## ANEXOS DE LAS PRUEBAS

Estas gráficas representan si las organizaciones necesitan una capacitación antes de utilizar el sistema y si lo recomendarían.



Como observamos, lo consideran intuitivo y de fácil manejo.



Una de las razones por las que recomendarían el sistema es porque Municipio les tomaría en cuenta en proyectos benéficos.

## BIBLIOGRAFIA

- [1] **Gómez Santiago**. "Paper: Microsoft SQL Server, MySQL",  
<http://www.scribd.com/doc/15059841/Microsoft-SQL-Server-MySQL-y-PostgreSQL>[Santiago-Gomez-RuizLinux-Plus-Magazine](http://www.scribd.com/doc/15059841/Microsoft-SQL-Server-MySQL-y-PostgreSQL). Obtenido 5 de octubre del 2009.
- [2] **Anónimo**. "MYSQL", <http://es.wikipedia.org/wiki/Mysql>. Obtenido 5 de octubre del 2009.
- [3] **Anónimo**. "Microsoft SQL Server", <http://es.wikipedia.org/wiki/SQLServer>. Obtenido 5 de octubre del 2009.
- [4] **Anónimo**. "PHP", <http://es.wikipedia.org/wiki/PHP>. Obtenido 5 de octubre del 2009.
- [5] **Anónimo**. "ASP.NET", <http://es.wikipedia.org/wiki/ASP.NET>. Obtenido 5 de octubre del 2009.