

**Universidad Abierta**

**Interamericana**

**2015**

**Ingeniería en Sistemas Informáticos**

**Trabajo de Campo I**

**Trabajo Final**

**Módulo de seguridad – Software Submition**

**Profesores:** Audoglio, Pablo

Vaquero, Marcelo

**Integrantes:**

Fotia, Diego Imanol

Viti, Emiliano Agustín

# Índice

[Índice 2](#_Toc426449402)

[Especificación de requerimientos 5](#_Toc426449403)

[Funcionales 5](#_Toc426449404)

[No funcionales 5](#_Toc426449405)

[Tecnológicos 5](#_Toc426449406)

[De rendimiento 5](#_Toc426449407)

[Alcance 5](#_Toc426449408)

[Inclusiones 5](#_Toc426449409)

[Exclusiones 5](#_Toc426449410)

[Registro de Stakeholders 6](#_Toc426449411)

[Compradores 6](#_Toc426449412)

[Vendedores 6](#_Toc426449413)

[Software Submition 6](#_Toc426449414)

[Prototipos de interfaz 6](#_Toc426449415)

[Modelo de dominio 8](#_Toc426449416)

[Diagrama de casos de uso 9](#_Toc426449417)

[Matriz de calidad 9](#_Toc426449418)

[Referencias 9](#_Toc426449419)

[Especificaciones de casos de uso 10](#_Toc426449420)

[Diagrama de robustez 11](#_Toc426449421)

[Diagrama de secuencia 12](#_Toc426449422)

[Diagrama de clases 13](#_Toc426449423)

[Modelo de datos 14](#_Toc426449424)

[Módulo de seguridad 15](#_Toc426449425)

[Modelo de dominio 15](#_Toc426449426)

[Diagramas de casos de uso 16](#_Toc426449427)

[Matriz de calidad 18](#_Toc426449428)

[Referencias 18](#_Toc426449429)

[Especificaciones de casos de uso 19](#_Toc426449430)

[Login 19](#_Toc426449431)

[Gestionar usuarios 20](#_Toc426449432)

[Cambiar clave 20](#_Toc426449433)

[Recuperar clave 21](#_Toc426449434)

[Diagramas de robustez 21](#_Toc426449435)

[Login 22](#_Toc426449436)

[Gestionar usuarios 23](#_Toc426449437)

[Cambiar clave 24](#_Toc426449438)

[Recuperar clave 25](#_Toc426449439)

[Diagrama de secuencias 26](#_Toc426449440)

[Login 26](#_Toc426449441)

[Gestionar usuarios 27](#_Toc426449442)

[Cambiar clave 28](#_Toc426449443)

[Recuperar clave 28](#_Toc426449444)

[Diagrama de clases 29](#_Toc426449445)

[Modelo de datos 30](#_Toc426449446)

[Anexo I - Procesos de desarrollo de software 31](#_Toc426449447)

[Fases en las que hemos dividido nuestro proyecto 31](#_Toc426449448)

[Ventajas de utilizar Iconix 32](#_Toc426449449)

[Anexo II - Estudio de viabilidad 32](#_Toc426449450)

[Legal 32](#_Toc426449451)

[Técnica 32](#_Toc426449452)

[Económica 32](#_Toc426449453)

[Anexo III – Seguridad 33](#_Toc426449454)

[Anexo IV – Auditoría 33](#_Toc426449455)

[Anexo V – Métricas 33](#_Toc426449456)

[Conteo de casos de uso 33](#_Toc426449457)

[Ajustes técnicos 34](#_Toc426449458)

[Ajustes del equipo de desarrolladores 34](#_Toc426449459)

[Resumen final 34](#_Toc426449460)

[Anexo VI – Gestión de riesgos 35](#_Toc426449461)

[Riesgos potenciales 35](#_Toc426449462)

[Listado de priorización de riesgos 35](#_Toc426449463)

[Anulación de riesgos y planes de contingencia 35](#_Toc426449464)

[Valoración de riesgos 36](#_Toc426449465)

[Anexo VII – Gestión de calidad 36](#_Toc426449466)

[Anexo VIII – Prueba del software 36](#_Toc426449467)

[Caja negra 36](#_Toc426449468)

[Caja blanca 36](#_Toc426449469)

# Especificación de requerimientos

### Funcionales

- El sistema deberá proveer un servicio de registro de usuarios, con sistema de pagos online.

- Servicio de registro y publicación de software.

- Servicio de compra y descarga de software.

- Servicio de publicidad.

- Servicio de chat entre usuarios del programa.

### No funcionales

- El sistema no deberá incluir servicios de reparación, recuperación, soporte del software proporcionado por los desarrolladores.

### Tecnológicos

- Windows Vista SP2 o superior, Microprocesador de 1ghz, 512mb de RAM, 512mb de espacio libre en HDD.

### De rendimiento

- Se optimizará el proyecto para que ocupe la menor cantidad de RAM, y el menor tiempo de microprocesador posible.

# Alcance

### Inclusiones

Permitirá compra de software online por parte de usuarios registrados, envío de software hecho por desarrolladores

Publicidad automatizada basada en estadística.

Exclusiones

Ni el sistema ni los desarrolladores de este, se harán responsables por fallas en los programas comprados, la cual recaerá directamente en el creador del programa en conflicto.

# Registro de Stakeholders

### Compradores

Expectativas principales:

Plataforma fácil de usar, con términos y condiciones favorables y facilidad de pago.

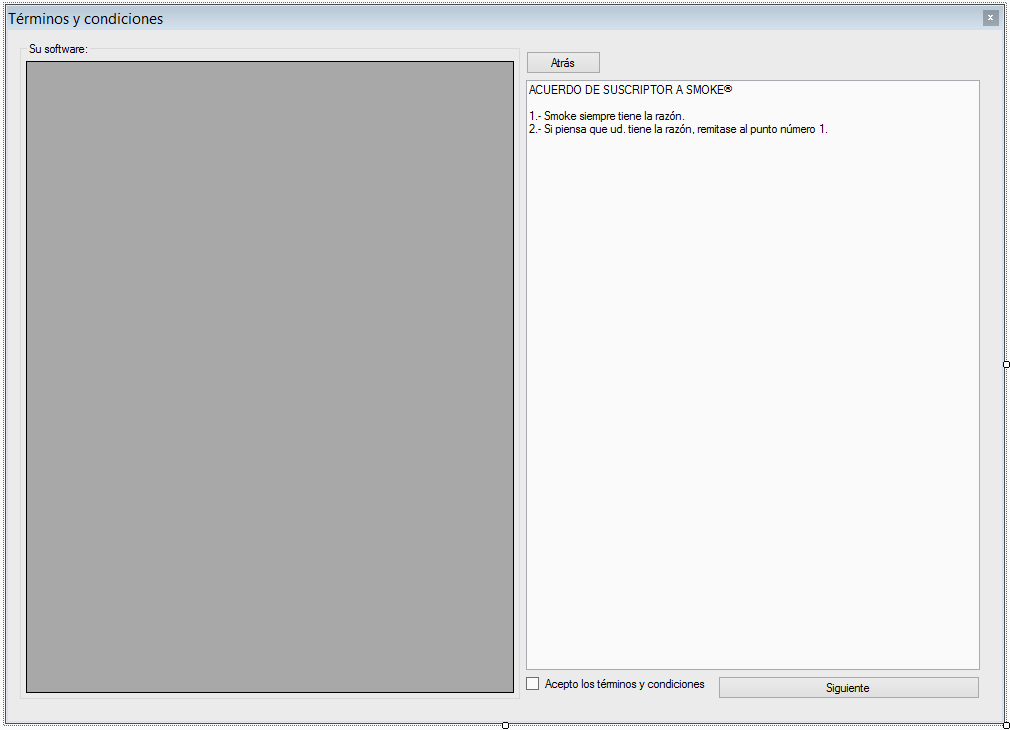
### Vendedores

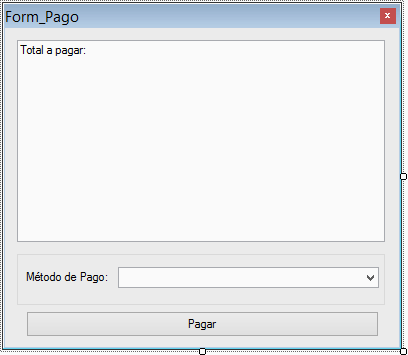
Expectativas principales:

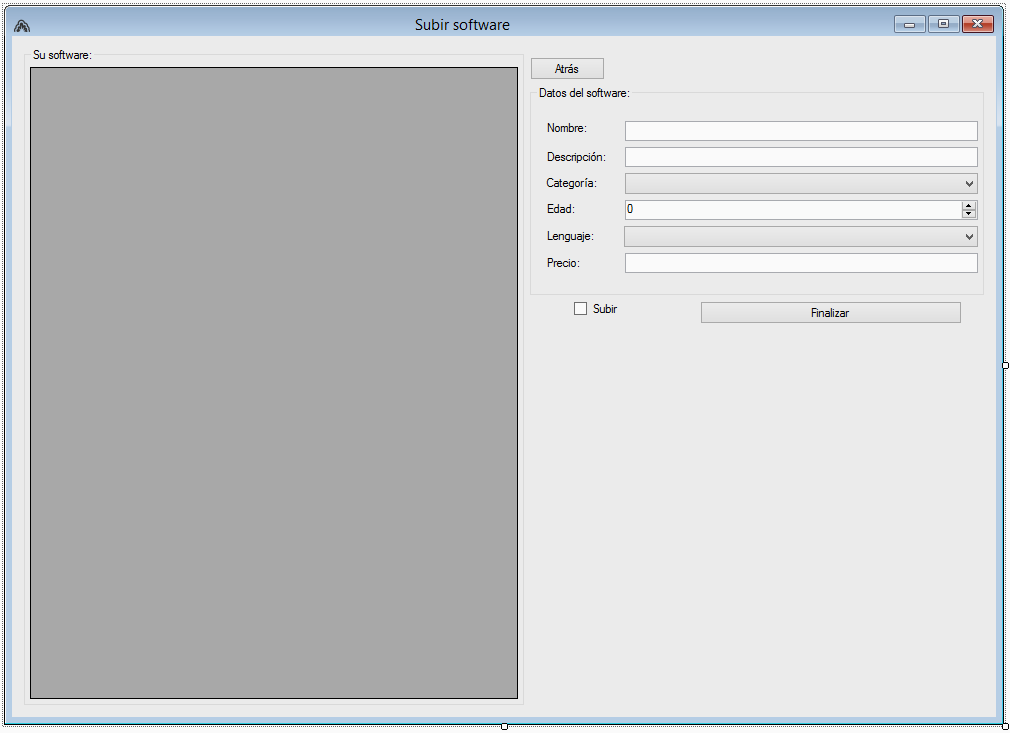
Contrato conveniente en cuanto a publicidad y ventas, equilibrio comisión-servicio.

# Software Submition

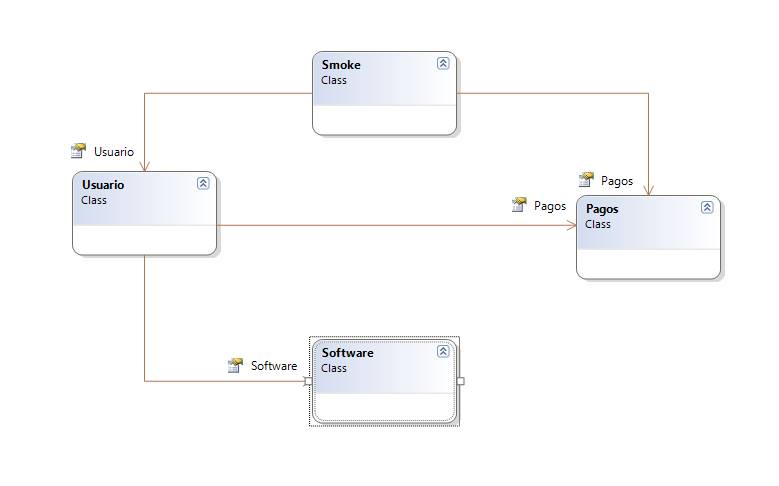
## Prototipos de interfaz



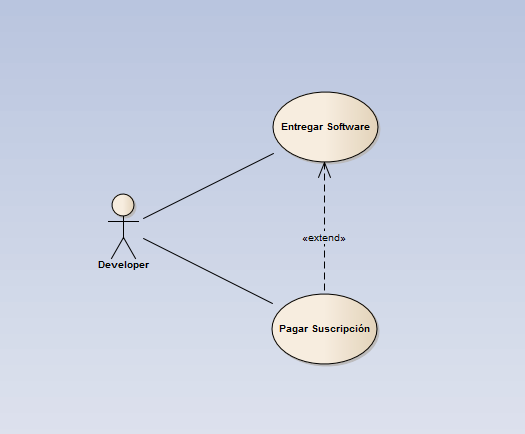




## Modelo de dominio



## Diagrama de casos de uso



## Matriz de calidad

|  |  |  |
| --- | --- | --- |
| CU/R | R1 | R2 |
| CU1 | X | X |

### Referencias

Requerimiento 1: El software tendrá Nombre, Descripción, Categoría, Edad, Lenguaje y Precio

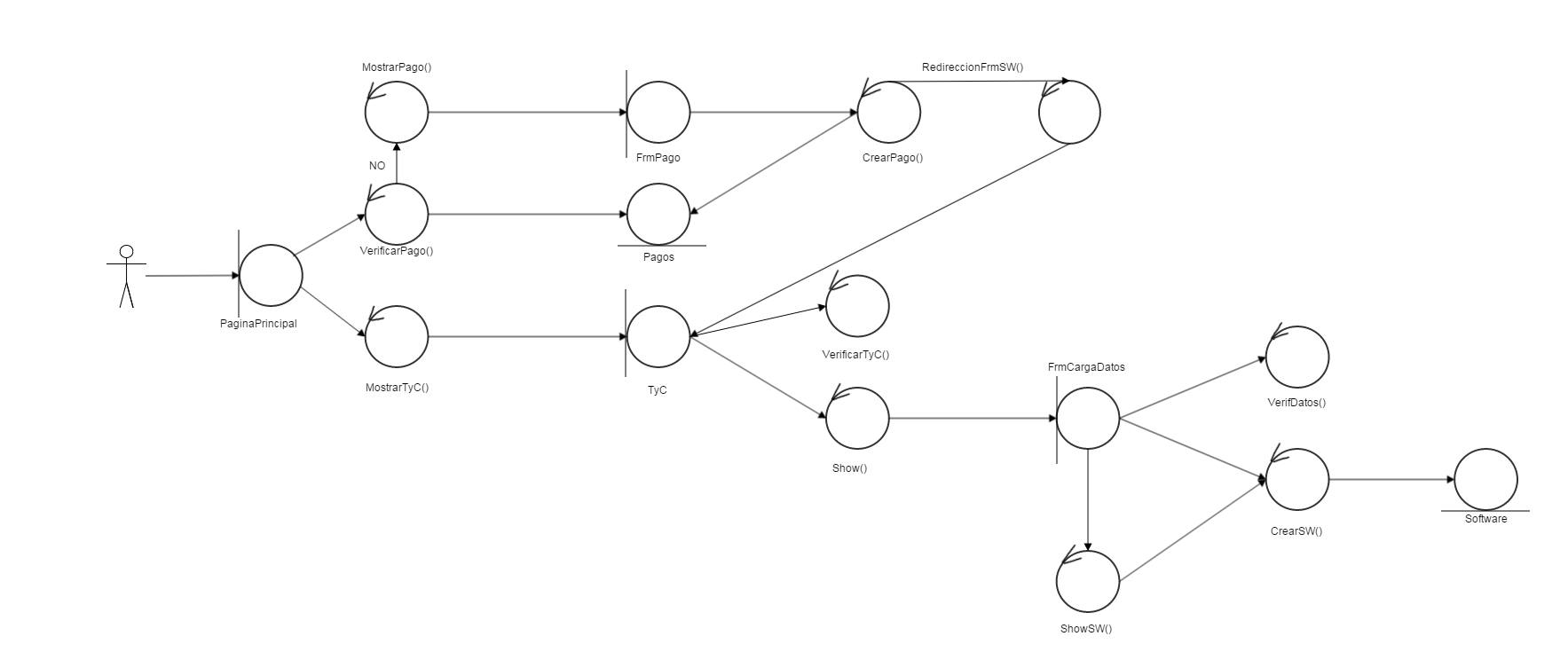
Requerimiento 2: El developer podrá subir su software

Caso de uso 1: Software Submition

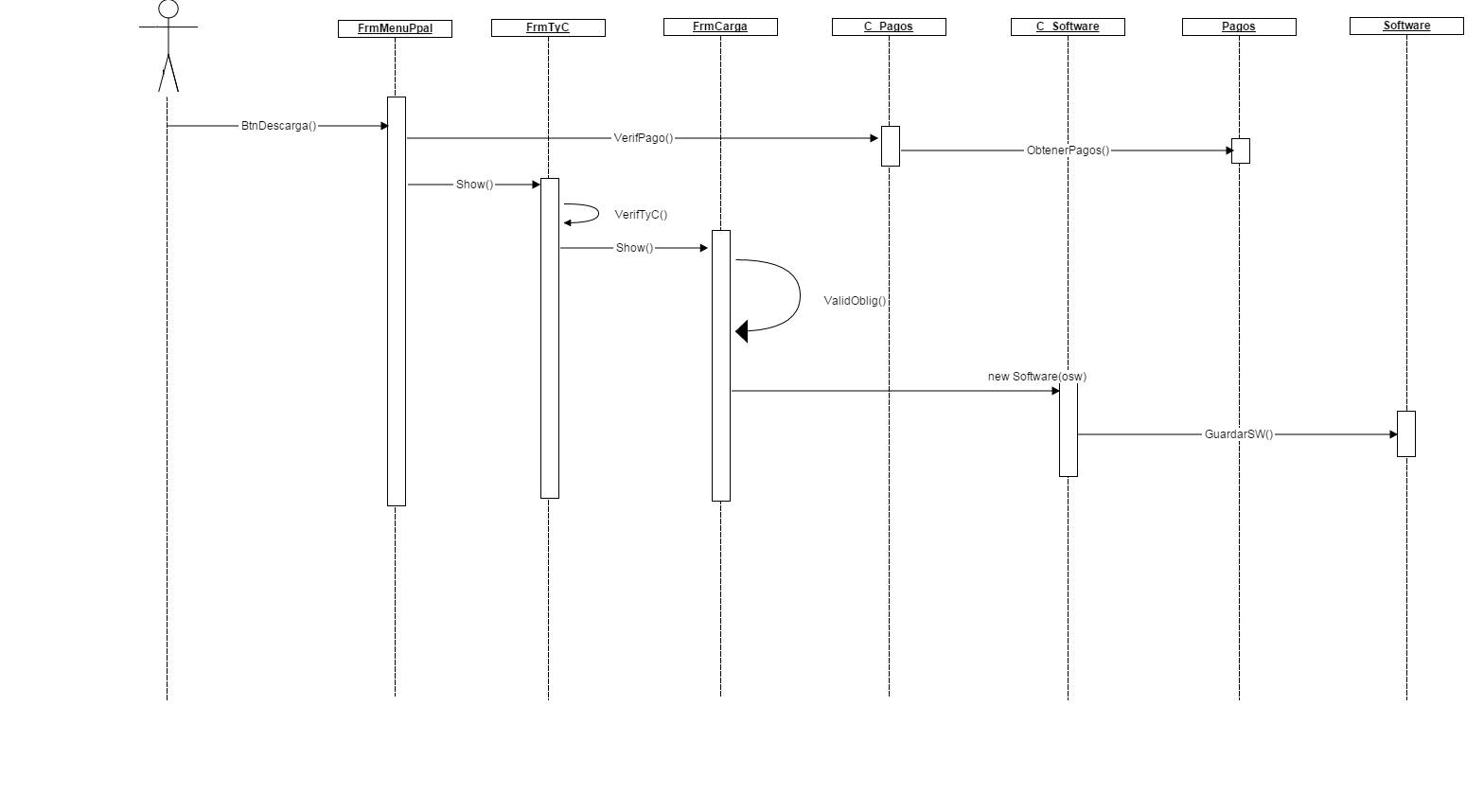
## Especificaciones de casos de uso

|  |  |  |
| --- | --- | --- |
|  | **Caso de Uso** | **Fecha: 10/03/15** |
| **Código** | CUD01 | |
| **Nombre** | Subir Software | |
| **Referencias** |  | |
| **Autor** | Imanol Fotia, Emiliano Viti | |
| **Revisor** |  | |
| **Versión** | 0001 | |
| **Estado** | Validado | |
| **Descripción** | El actor procederá a subir su software al sistema | |
| **Actores** | Sistema, Usuario | |
| **Pre-condición** | Usuario logueado, con cuota al día | |
| **Puntos de Extensión** |  | |
| **Curso Básico** |  | |
| 1 | El Usuario hace click en el botón "Subir Software", que se encuentra en su perfil | |
| 2 | Se abre un nuevo formulario mostrando los términos y condiciones que deben ser aprobados, y el botón de continuar | |
| 3 | El Usuario, acepta los términos y condiciones y oprime el botón siguiente | |
| 4 | El sistema verifica que el usuario tenga su cuota al día | |
| 5 | Si el Usuario ha pagado, se abre un nuevo formulario, mostrando los campos a completar por el usuario, en orden de subir su producto | |
| 6 | El Usuario luego de completar los campo obligatorios, oprime el botón Subir, y el software será agregado al sistema | |
| Curso Alternativo |  |  |
|  |  |  |
| 5.1 | En caso de que el usuario no tenga su cuota al día, se abrirá un nuevo formulario | |
| 5.2 | El usuario deberá elegir su método de pago, y oprimir el botón "Pagar" | |
| 5.3 | Luego de esto el usuario está en condiciones de subir su producto | |
| Pos-condición | Software Subido |  |

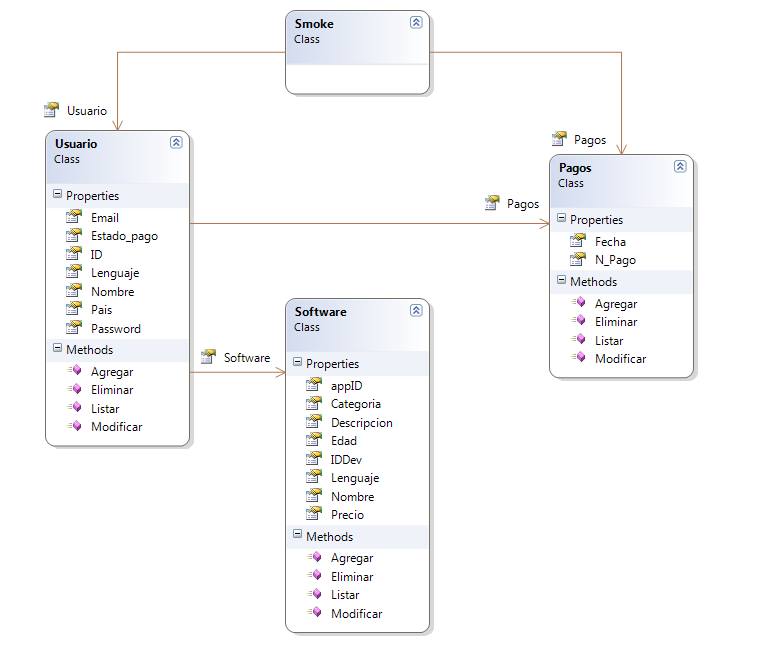
## Diagrama de robustez



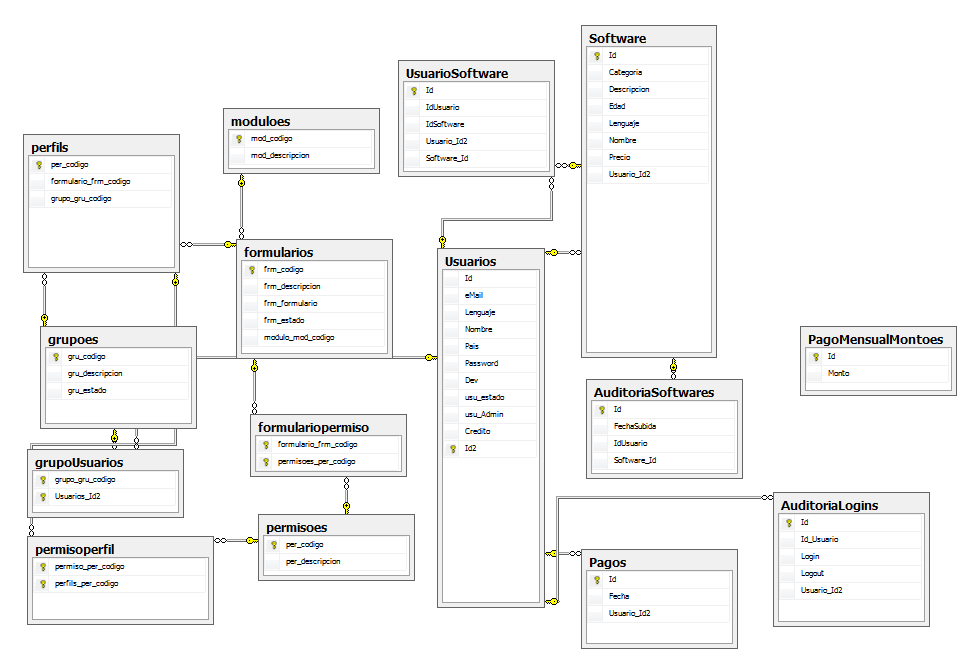
## Diagrama de secuencia



## Diagrama de clases

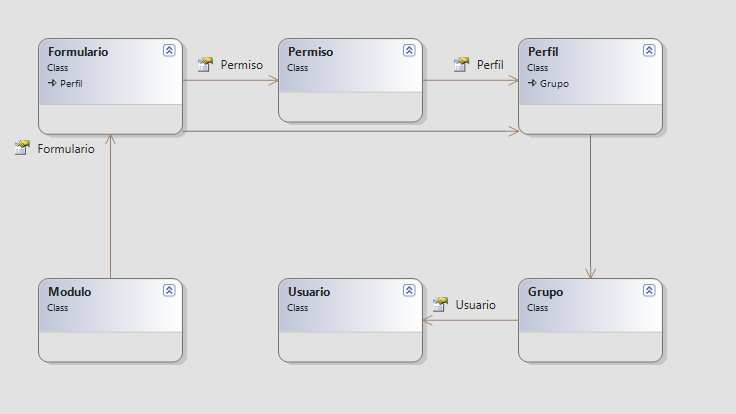


## Modelo de datos

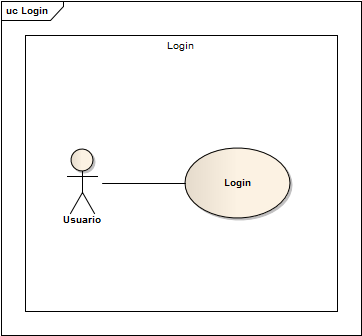
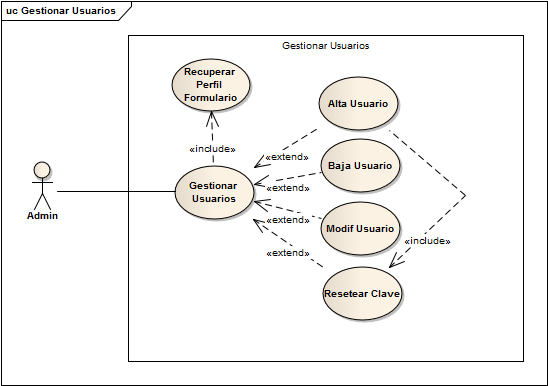
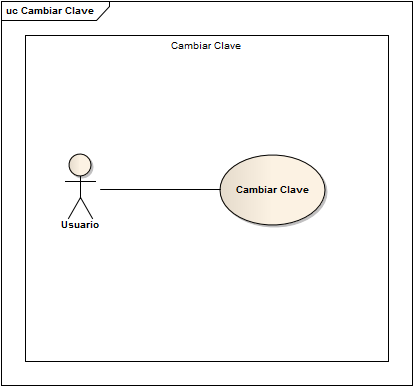
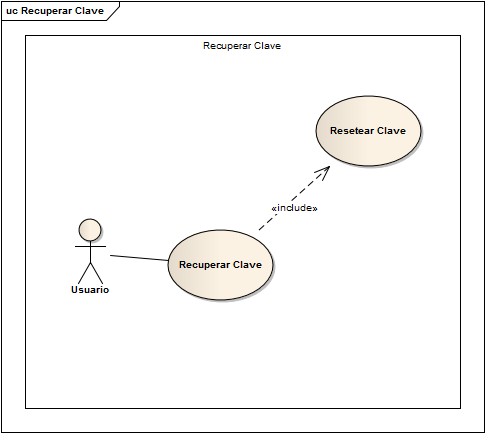
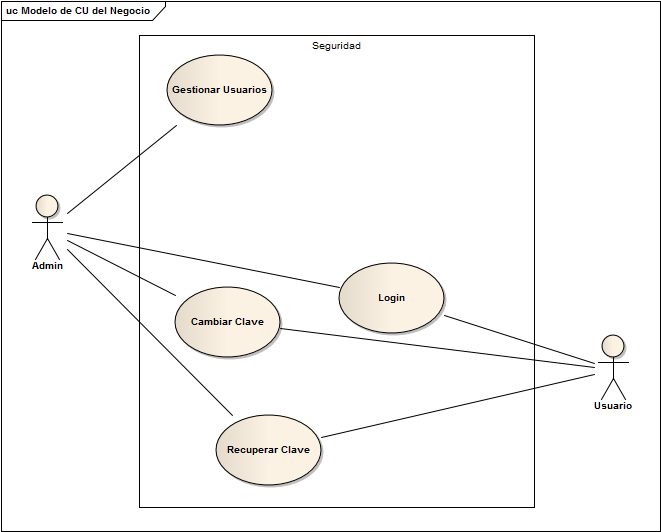


# Módulo de seguridad

## Modelo de dominio



## Diagramas de casos de uso

## Matriz de calidad

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **CU1** | **CU2** | **CU3** | **CU4** |
| **R1** | **X** | **X** | **X** | **X** |
| **R2** | **X** | **X** |  |  |
| **R3** |  |  | **X** |  |
| **R4** | **X** |  |  |  |
| **R5** | **X** | **X** | **X** | **X** |
| **R6** | **X** |  |  |  |
| **R7** |  | **X** |  |  |
| **R8** | **X** |  |  |  |
| **R9** | **X** |  |  |  |
| **R10** | **X** |  |  |  |
| **R11** | **X** |  |  |  |
| **R12** |  |  |  | **X** |

### Referencias

Requerimientos:

1. Gestionar Usuarios. Los usuarios tendrán: Usuario, Clave, Apellido y Nombre, E-mail, y Estado (activo/inactivo).
2. Los usuarios estarán agrupados y podrán pertenecer a uno o varios grupos.
3. Los usuarios podrán cambiar su propia clave.
4. Los usuarios ingresarán al sistema con Usuario y Clave.
5. La Clave de usuario estará encriptada (pero no se podrá desencriptar).
6. Si un usuario pertenece a varios grupos los permisos se sumarán.
7. Gestionar Grupos. Los grupos tendrán: Código, Descripción, y Estado (activo/inactivo).
8. Los grupos tendrán acceso a uno o varios formularios.
9. Los formularios tendrán: Código y Descripción.
10. Los formularios tendrán uno o varios permisos.
11. Un formulario pertenece a un módulo.
12. Un usuario podrá recuperar su clave ingresando: Usuario, y E-mail registrado. La nueva clave le será enviada al E-mail registrado.

Casos de uso:

1. Login.
2. Gestionar usuarios.
3. Cambiar clave.
4. Recuperar clave.

## Especificaciones de casos de uso

### Login

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | | |
| **Código** | CUAN01 | |
| **Nombre** | Login | |
| **Referencias** | R1, R2, R4, R5, R6, R8, R9, R10, R11, R12, R14, R16, R17 | |
| **Autor** | Fotia, Viti |  |
| **Revisor** | Ing. Marcelo Vaquero, Ing. Pablo Audoglio | |
| **Versión** | 0001 | |
| **Estado** | Validado | |
| **Descripción** | El actor ingresará un nombre de usuario y clave para iniciar sesión en el sistema. El sistema deberá recuperar el perfil del usuario para establecer qué permisos posee sobre los menúes y formularios del sistema. | |
| **Actores** | Usuario | |
| **Pre-condición** | Actor existente en el sistema y activo. | |
| **Puntos de Extensión** |  | |
| **Pos-condición** | Actor logueado. | |

### Gestionar usuarios

|  |  |
| --- | --- |
| **Caso de Uso** | |
| **Código** | CUAN02 |
| **Nombre** | Gestionar Usuarios |
| **Referencias** | R1, R2, R5, R7, R13, R14, R15 |
| **Autor** | Fotia, Viti |
| **Revisor** | Ing. Marcelo Vaquero, Ing. Pablo Audoglio |
| **Versión** | 0001 |
| **Estado** | Validado |
| **Descripción** | El actor podrá agregar, modificar, eliminar usuarios y resetear la clave de los usuarios del sistema teniendo en cuenta para estas operaciones: usuario, clave, nombre y apellido, estado, y el o los grupos a los que pertenecerá el usuario. Para llevar a cabo esta gestión podrá visualizar un listado de los usuarios existentes y filtrar por nombre y apellido, por estado y por grupo. |
| **Actores** | Administrador |
| **Pre-condición** | El actor debe estar logueado, poseer permisos sobre el formulario "Usuarios", y hacer click en el ítem de menú "Usuarios" dentro del menú "Seguridad". |
| **Puntos de Extensión** |  |
| **Pos-condición** | El actor ha gestionado usuarios. |

### Cambiar clave

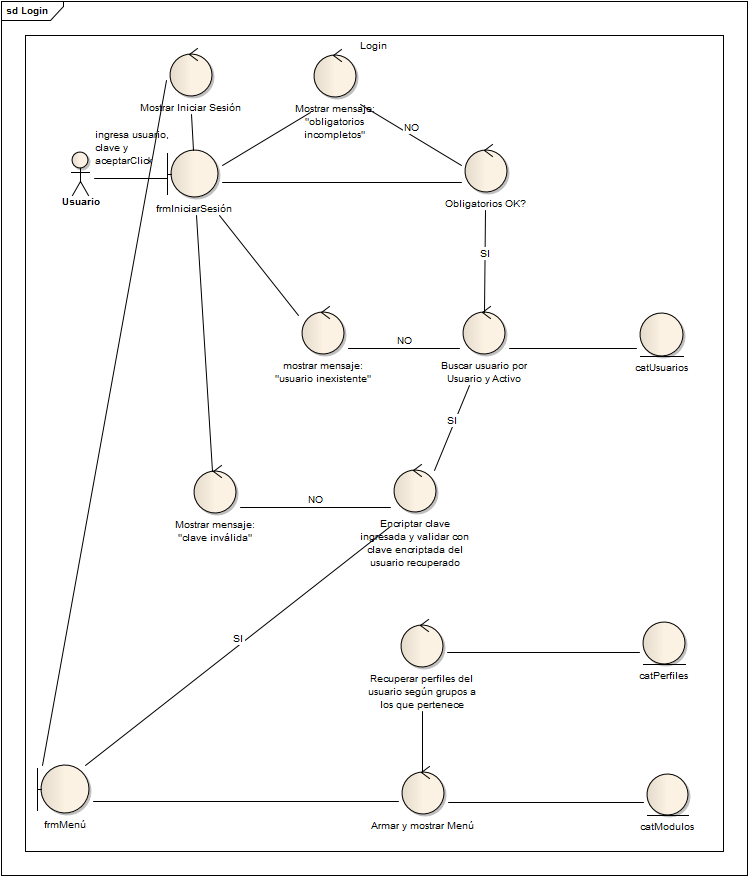
|  |  |
| --- | --- |
| **Caso de Uso** | |
| **Código** | CUAN05 |
| **Nombre** | Cambiar Clave |
| **Referencias** | R1, R3, R5 |
| **Autor** | Fotia, Viti |
| **Revisor** | Ing. Marcelo Vaquero, Ing. Pablo Audoglio |
| **Versión** | 0001 |
| **Estado** | Validado |
| **Descripción** | El actor podrá cambiar clave ingresando su clave actual, y la nueva clave elegidados veces para su verificación. |
| **Actores** | Usuario |
| **Pre-condición** | Actor logueado en el sistema. |
| **Puntos de Extensión** |  |
| **Pos-condición** | Clave cambiada. |

### Recuperar clave

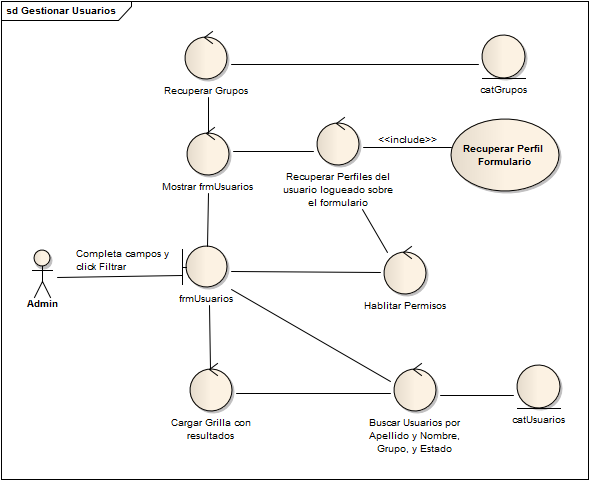
|  |  |  |
| --- | --- | --- |
| **Caso de Uso** |  |  |
| **Código** | CUAN07 |  |
| **Nombre** | Recuperar Clave |  |
| **Referencias** | R1, R5, R14, R18 |  |
| **Autor** | Fotia, Viti |  |
| **Revisor** | Ing. Marcelo Vaquero, Ing. Pablo Audoglio |  |
| **Versión** | 0001 |  |
| **Estado** | Validado |  |
| **Descripción** | El actor podrá recuperar su clave ingresando usuario y dirección de e-mail registrada en el sistema. | |
| **Actores** | Usuario |  |
| **Pre-condición** | El actor debe existir como usuario en el sistema, estar activo y tener registrada una dirección de e-mail válida. | |
| **Puntos de Extensión** |  |  |
| **Pos-condición** | Clave recuperada. |  |

## Diagramas de robustez

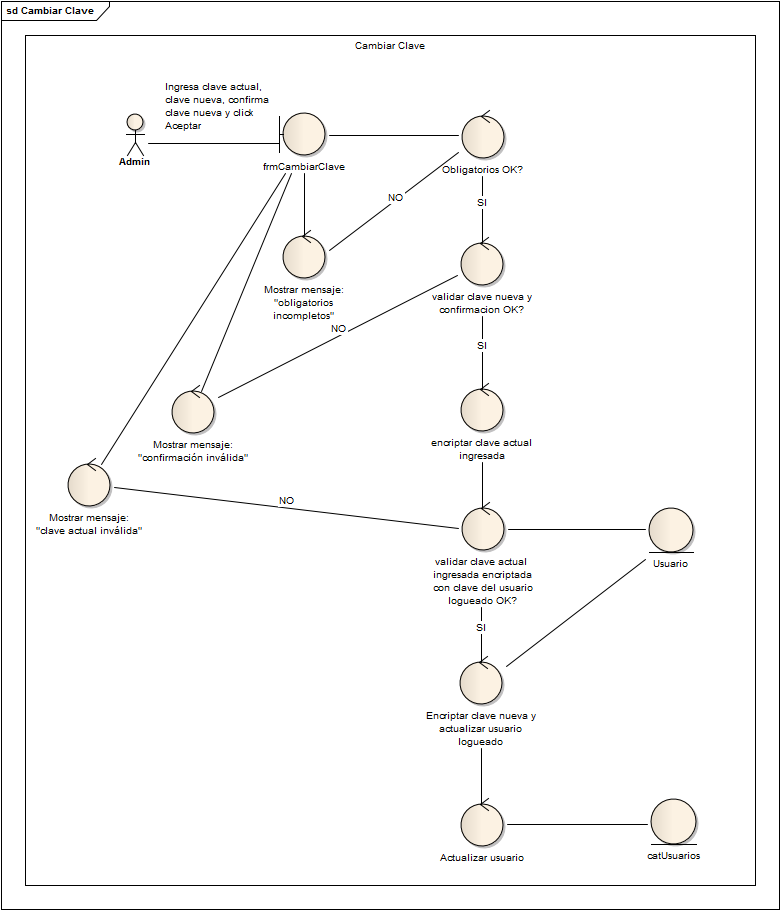
### Login



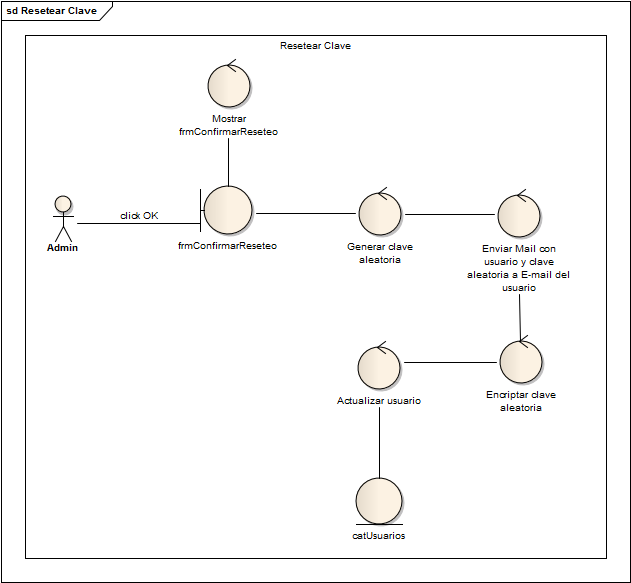
### Gestionar usuarios



### Cambiar clave

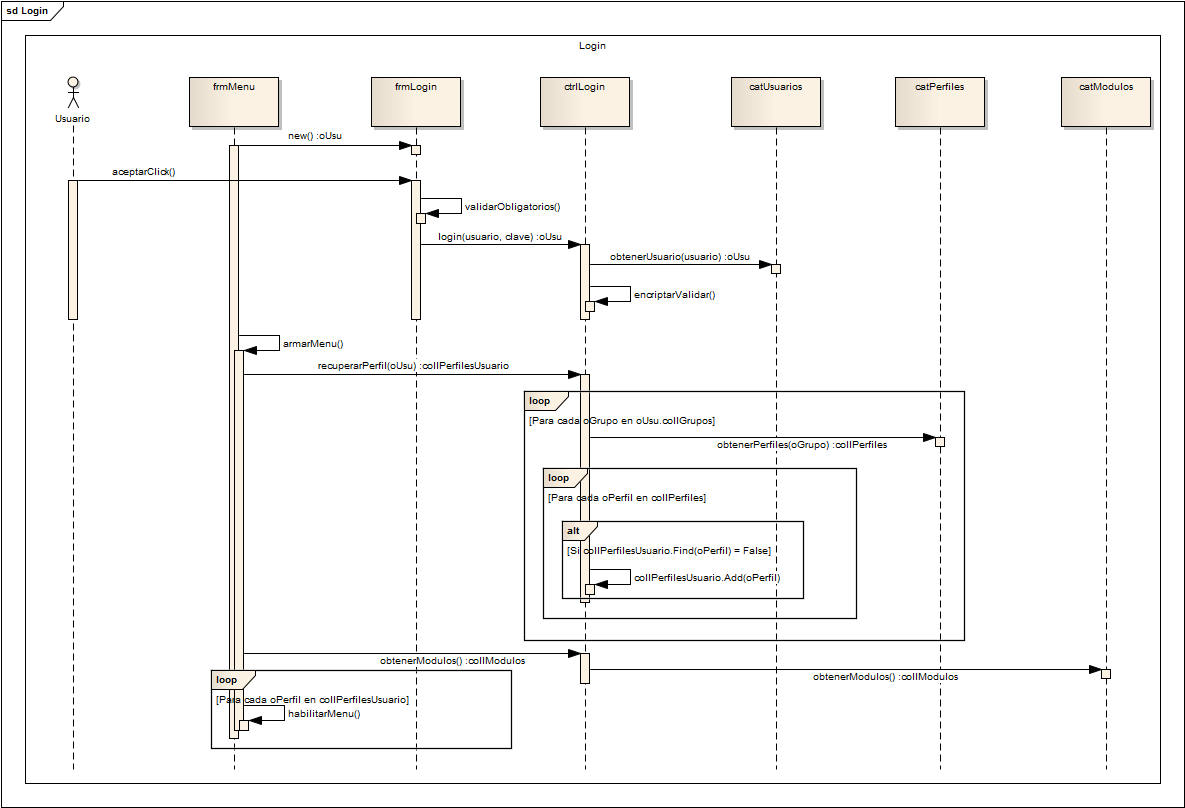


### Recuperar clave

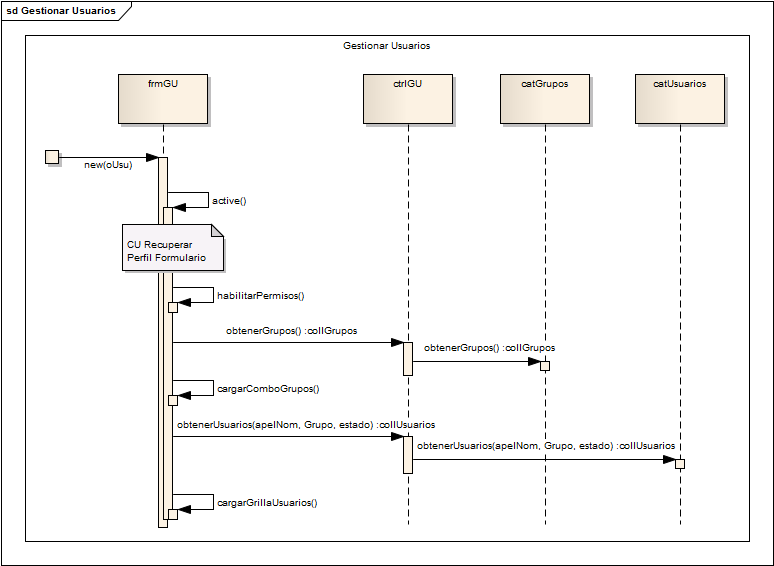


## Diagrama de secuencias

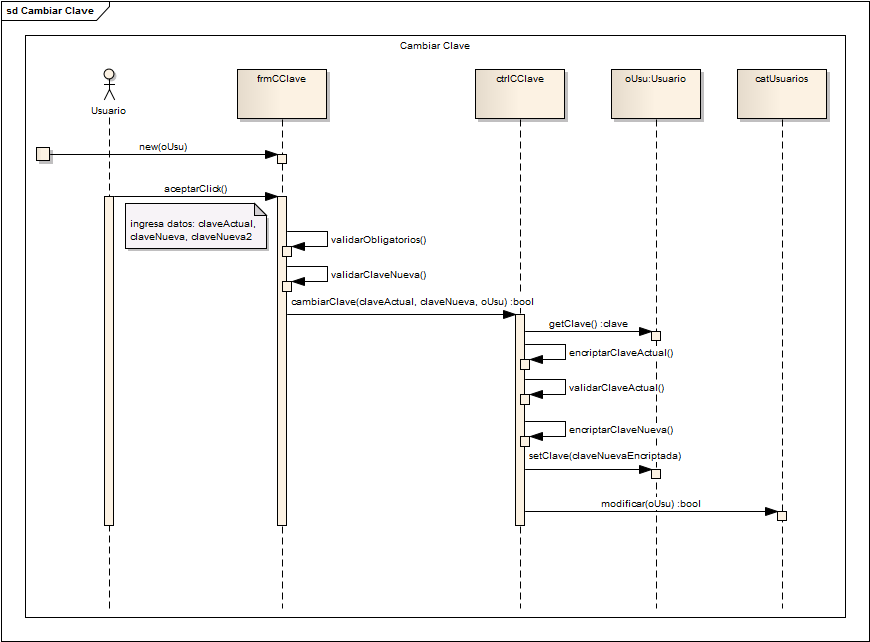
### Login



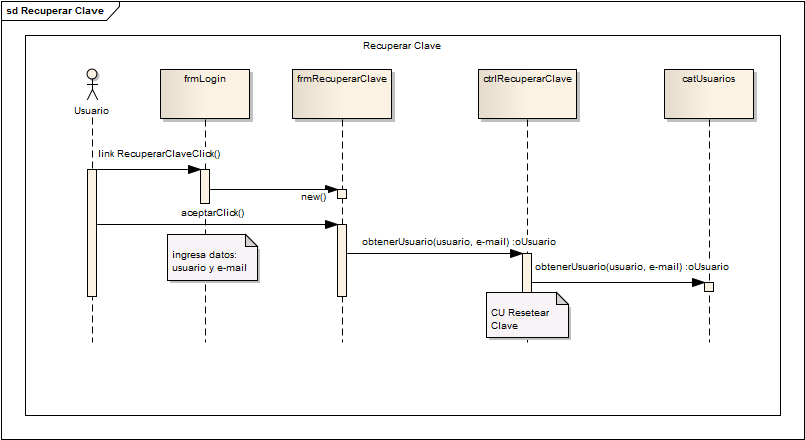
### Gestionar usuarios



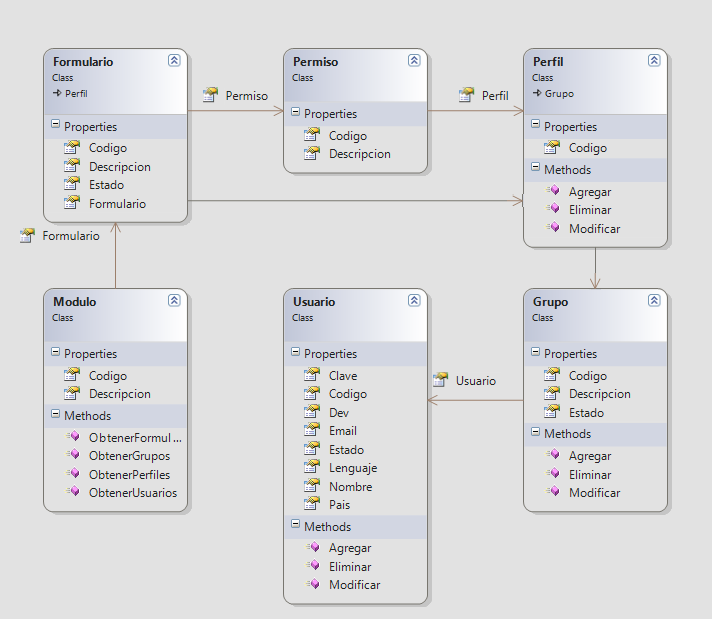
### Cambiar clave



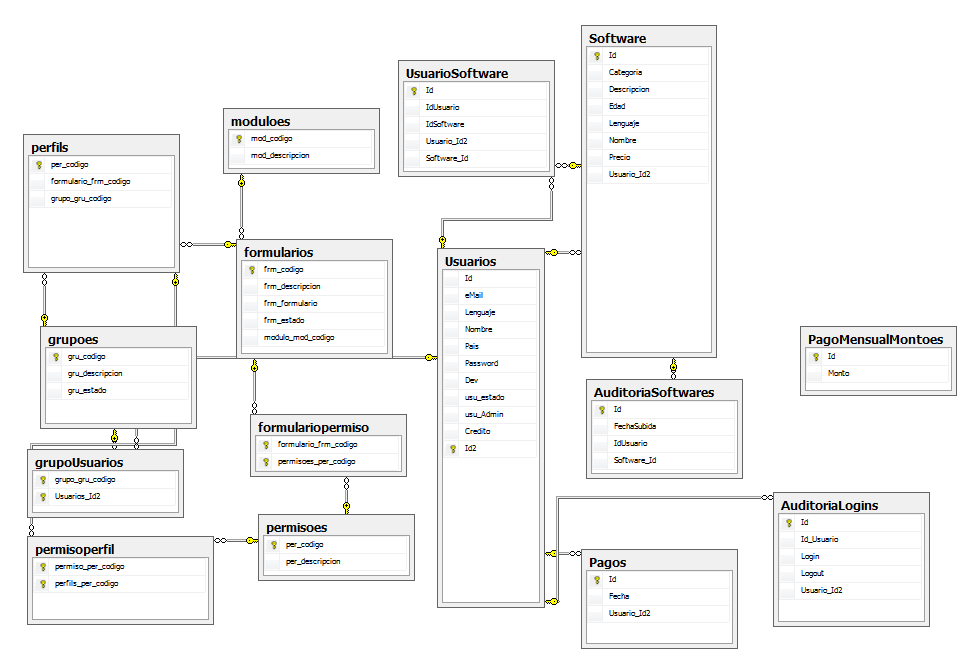
### Recuperar clave



## Diagrama de clases



## Modelo de datos



# Anexo I - Procesos de desarrollo de software

En nuestro proyecto hemos utilizado el proceso de desarrollo Iconix.

Iconix es una metodología pesada-ligera de desarrollo del Software.

Iconix deriva directamente del RUP y su fundamento es el hecho de que un 80% de los casos pueden ser resueltos tan solo con un uso del 20% del UML, con lo cual se simplifica muchísimo el proceso sin perder documentación al dejar solo aquello que es necesario.

## Fases en las que hemos dividido nuestro proyecto

###### Fase 1: Análisis de requisitos

Dentro de esta fase se realizan las siguientes tareas:

* Modelo del dominio
* Elaboración rápida de prototipos
* Modelo de casos de usos

###### Fase 2: Análisis y diseño preliminar

Dentro de esta fase se realizan las siguientes tareas:

* Descripción de los casos de uso
* Diagramas

###### Fase 3: Diseño

Dentro de esta fase se realiza la siguiente tarea:

* Diagramas de secuencia

###### Fase 4: Implementación

Dentro de esta fase se realiza la siguiente tarea:

* Escribir y generar código

## Ventajas de utilizar Iconix

Es un proceso ágil para obtener un sistema informático, dedicado a la construcción de sistemas de gestión de pequeña y mediana complejidad con la participación de los usuarios finales. Por ende, no perdimos documentación, pero no se necesitó realizar tanta documentación como en UML, ayudando a los tiempos de entrega.

# Anexo II - Estudio de viabilidad

## Legal

No existen impedimentos legales que puedan obstaculizar el normal desarrollo de nuestro proyecto.

## Técnica

Actualmente 3 de cada 4 PC’s del mundo cumplen los requisitos mínimos de nuestro software, por lo que no se requiere una ampliación del hardware. Además según datos actuales el 43% de la población del mundo tiene acceso a internet.

El software creado es muy intuitivo, por lo que no requiere una capacitación especial del usuario.

Tampoco se requerirá ninguna instalación extra, ya que el software, como se menciona antes funciona en una PC con características relativamente comunes.

## Económica

El proyecto llevará aproximadamente 870hs distribuidas en dos cuatrimestres, por lo que para el primer cuatrimestre se deberá tener completadas 435hs de trabajo.

En cuanto a recursos humanos el proyecto será completado por dos personas, estudiantes de Ingeniería en Sistemas, el costo hora es inexistente, debido a que se trata de un proyecto universitario, hecho por estudiantes de dicha universidad.

El hardware utilizado para desarrollar el proyecto son las computadoras de ambos estudiantes, ambas capaces de andar .NET Framework y Visual Studio 2010.

No se contemplan gastos de capacitación, debido a que principalmente se utilizarán conocimientos previos.

# Anexo III – Seguridad

Se desarrolla el módulo de seguridad, junto con su documentación en el título “Modulo de seguridad”.

# Anexo IV – Auditoría

Se intentará lograr la trazabilidad del dato en cuanto a los logins y la subida de software.

La trazabilidad del dato del software se logra con el registro del software subido, fecha y hora, y usuario que lo hizo.

En cuanto a los logins se llega a la trazabilidad del dato registrando el usuario que se logueo, y la fecha y hora de entrada y salida.

# Anexo V – Métricas

Para la métrica del esfuerzo necesario en horas-hombre para terminar el proyecto se ha utilizado el método suministrado por la empresa “Neoris”.

Siguiente un resumen de los datos principales utilizados:

## Conteo de casos de uso

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Identificación | | Tipo | | |
| **ID** | **Casos de Uso** | **Simples** | **Medio** | **Complejo** |
| 1 | Subir Software |  |  | 1 |
| 2 | Comprar Software |  |  | 1 |
| 3 | Módulo de Seguridad -1 |  |  | 1 |
| 4 | Módulo de Seguridad -2 |  |  | 1 |
| 5 | Módulo de Seguridad -3 | 1 |  |  |
| 6 | Módulo de Seguridad -4 |  | 1 |  |
| 7 | Tablero |  |  | 1 |

## Ajustes técnicos

|  |  |  |  |
| --- | --- | --- | --- |
| **Factores Técnicos** | **Peso** | **Puntaje (0 - 5)** | **Base para Cálculo** |
| Distributed system | 2 | 0 | 0 |
| Response or throughput performance objectives | 1 | 2 | 2 |
| End user efficiency (online services) | 1 | 5 | 5 |
| Complex Processing | 1 | 4 | 4 |
| Reusable code | 1 | 4 | 4 |
| Easy to install | 0.5 | 3 | 1.5 |
| Easy to use | 0.5 | 5 | 2.5 |
| Portable | 2 | 0 | 0 |
| Easy to change | 1 | 0 | 0 |
| Concurrent | 1 | 5 | 5 |
| Special Security features | 1 | 5 | 5 |
| Direct access for third parties | 1 | 5 | 5 |
| Special user training required | 1 | 0 | 0 |

## Ajustes del equipo de desarrolladores

|  |  |  |  |
| --- | --- | --- | --- |
| **Factores Ambientales** | **Peso** | **Puntaje (0 - 5)** | **Base para Cálculo** |
| Familiar with Development Process | 1.5 | 5 | 7.5 |
| Application experience | 0.5 | 3 | 1.5 |
| Object-oriented experience | 1 | 3 | 3 |
| Lead analyst capability | 0.5 | 2 | 1 |
| Motivation | 1 | 5 | 5 |
| Stable requirements | 2 | 5 | 10 |
| Part-time workers | -1 | 3 | -3 |
| Difficult programming language | -1 | 2 | -2 |

## Resumen final

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Resumen por hojas** | **Simple** | **Medio** | **Alto** | **Total** |
| Actores (UUCP) | 0 | 0 | 1 | 3 |
| Casos de Uso (UUCP) | 1 | 1 | 5 | 90 |
| Ajustes Técnicos (TCF) | 0.94 | | | |
| Ajustes Team (EF) | 0.71 | | | |
|  |  |  |  |  |
| UCP |  |  |  | 62 |
| Productividad |  |  |  | 14 |
| Esfuerzo Total |  |  |  | 869 |

# Anexo VI – Gestión de riesgos

## Riesgos potenciales

|  |  |  |
| --- | --- | --- |
| **Tipo de Riesgo** | **Categoría** | **Riesgos Potenciales** |
| Personas | Proyecto | Conocimientos del personal para desarrollar el producto |
| Personas | Producto/Negocio | Usuarios finales que se opongan al resultado del sistema |
| Estimación | Negocio | La competencia lanza una opción a nuestro cliente |
| Tecnología | Proyecto | Pérdida de datos |

## Listado de priorización de riesgos

|  |  |  |  |
| --- | --- | --- | --- |
| **Nº** | **Riesgo** | **Probabilidad** | **Efectos** |
| 1 | Conocimientos del personal para desarrollar el producto | 2-Bajo | 3-Tolerable |
| 2 | Usuarios finales que se opongan al resultado del sistema | 3-Moderado | 1-Catastrófico |
| 3 | La competencia lanza una opción a nuestro cliente | 4-Alto | 3-Tolerable |
| 4 | Pérdida de datos | 2-Bajo | 1-Catastrófico |

## Anulación de riesgos y planes de contingencia

|  |  |  |
| --- | --- | --- |
| **Nº** | **Riesgo** | **Estrategia** |
| 1 | Conocimientos del personal para desarrollar el producto | Plan de capacitación |
| 2 | Usuarios finales que se opongan al resultado del sistema | Más características |
| 3 | La competencia lanza una opción a nuestro cliente | Innovación |
| 4 | Pérdida de datos | Back-Up |

## Valoración de riesgos

|  |  |
| --- | --- |
| **Tipo de Riesgo** | **Indicadores Potenciales** |
| Personas | Puede pasar al principio del proyecto |
| Estimación | Merma en la cantidad de nuevos clientes |
| Tecnología | Hardware dañado |

# Anexo VII – Gestión de calidad

Se tuvo en cuenta para el desarrollo del proyecto la facilidad de uso del software, la seguridad y confiabilidad de los datos de los usuarios y una apertura hacia la diversidad del software a vender y comprar.

# Anexo VIII – Prueba del software

## Caja negra

Para la prueba de caja negra se ha adjuntado un video realizándose sobre la iteración software submition.

## Caja blanca

La prueba de caja blanca se ha realizado sobre el método utilizado para agregar un usuario, detallándose paso a paso todos los estadios del objeto Usuario.

private void btn\_Siguiente\_Click(object sender, EventArgs e)

{

if (txt\_Verificación.Text != Codigo\_Verificacion) MessageBox.Show("El Codigo ingresado es incorrecto.");

if (txt\_Verificación.Text == Codigo\_Verificacion)

{

//Se instancia el objeto usuarios, con todas las variables con valor nulo

Modelo.Usuarios Usuario = new Modelo.Usuarios();

//Se asigna el ID de usuario, por ej: Usuario.Id = pedro

Usuario.Id = txt\_ID.Text;

//Se asigna el mail del nuevo usuario, por ej: Usuario.eMail = abc@abc.com

Usuario.eMail = txt\_email.Text;

//Asignamos el lenguaje del usuario, por ej: Usuario.Lenguaje = Español

Usuario.Lenguaje = cb\_Lenguaje.SelectedItem.ToString();

//Se agrega el nombre del usuario, por ej: Usuario.Nombre = Pedro

Usuario.Nombre = txt\_Nombre.Text;

//Se agrega el pais del usuario, por ej: Usuario.Pais = Argentina

Usuario.Pais = cb\_Pais.SelectedItem.ToString();

//Antes de asignar el password, este es encriptado, por ej: pass = 123, luego pass = qwidjq2

string pass = MD5.Encriptar(txt\_Password.Text);

//Se guarda el password, Usuario.Password = qwidjq2

Usuario.Password = pass;

//Se definen los privilegios del usuario

Usuario.Dev = 0;

//Se habilita el usuario

Usuario.usu\_estado = true;

//se define el rango del usuario

Usuario.usu\_Admin = false;

//Se guarda el nuevo usuario en la base de datos, los datos del nuevo usuario serán:

* Usuario.Id = pedro
* Usuario.eMail = [abc@abc.com](mailto:abc@abc.com)
* Usuario.Lenguaje = Español
* Usuario.Nombre = Pedro
* Usuario.Pais = Argentina
* Usuario.Password = qwidjq2

C\_Usuarios.Agregar(Usuario);

MessageBox.Show("Usuario creado");

this.DialogResult = DialogResult.OK;

}

}