Documento de Arquitectura de Software

Grupo 13

Carmen Barreira

Pablo de Sosa

Ana Lía Castellanos

Mauricio Rodríguez

Contenido

[1 Introducción. 3](#__RefHeading__1137_307693533)

[2 Vista de Casos de Uso 4](#__RefHeading__1139_307693533)

[2.1 Casos de uso identificados 4](#__RefHeading__1141_307693533)

[2.2Diagrama de Casos de Uso Críticos 5](#__RefHeading__971_500311874)

[2.3 Actores 5](#__RefHeading__1145_307693533)

[2.3.1Usuario 5](#__RefHeading__1269_307693533)

[2.3.2 Administrador 5](#__RefHeading__1271_307693533)

[2.4 Especificación de Casos de Uso Críticos 6](#__RefHeading__1147_307693533)

[3 Vista Lógica](#__RefHeading__1149_307693533) 10

[3.1 Estilo Arquitectónico.](#__RefHeading__1151_307693533) 10

[3.2Subsistemas](#__RefHeading__973_500311874) 11

[3.3 Diagramas de Interacción](#__RefHeading__1155_307693533) 13

[4 Vista de Distribución (Deployment). 1](#__RefHeading__1157_307693533)6

[5 MODELO DE DOMINIO. 1](#__RefHeading__1157_307693533)8

[6 Referencias 1](#__RefHeading__1163_307693533)9

# Introducción.

En este documento se describe la arquitectura del sistema que se desarrollará, tomando en cuenta los diferentes aspectos (o vistas), y haciendo énfasis en cada uno de ellos.

Este enfoque se denomina “4+1 View Model Architecture”, y presenta al sistema bajo 5 diferentes vistas: la Vista de Casos de Uso, la Vista Lógica, la Vista de Distribución, la Vista de Implementación y la Vista de Procesos.

OL4RENT es una plataforma informática que tiene como finalidad la publicación de diferentes sitios de renta, a través de los cuales las diferentes empresas interesadas, así como usuarios particulares, podrán publicitar los bienes a rentar. Si bien, en un principio los usuarios serán propios de cada sitio, existe la posibilidad de interacción entre los diferentes sitios registrados en el sistema. Además de dichas características, OL4RENT brinda a sus usuarios la posibilidad de chatear, consumir noticias y novedades de diferentes fuentes, así como compartir información de sus operaciones, entre otras.

El sistema cuenta con tres módulos, Front Office, de acceso público, con la posibilidad de autenticación integrada con Facebook. El módulo “Mobile” ofrece un subconjunto de las funcionalidades del Front Office, con accesibilidad desde dispositivos móviles. Finalmente el Back Office, módulo restringido solamente a usuarios de carácter administrador.

Los sitios registrados en el sistema representan instalaciones independientes. Pero la plataforma brinda la posibilidad de integrar dichas instalaciones, permitiendo ampliar la base de datos de usuarios, así como compartir información de las operaciones realizadas, e inclusive chat en línea entre los usuarios de los diferentes sitios de OL4RENT.

# Vista de Casos de Uso

## Casos de uso identificados

Módulo Front office:

 *Registro de Usuario.*

Permite al usuario anónimo registrarse como nuevo usuario en el portal del sitio al cual ingresó, brindando información personal (nombre, apellido, mail, etc.).

 *Autenticación de usuario*

Los usuarios registrados tienen la posibilidad de loguearse en el portal web para acceder a funcionalidades privilegiadas.

 *Chatear.*

Los usuarios podrán seleccionar un usuario que se encuentre en línea e iniciar una conversación por chat con el mismo.

Módulo Back office:

 *Ingresar nuevo sitio.*

Permite al administrador autenticado dar de alta un nuevo sitio a partir de templates y opciones de configuración provistas por el sistema.

Módulo Móvil:

 *Autenticación de usuario*

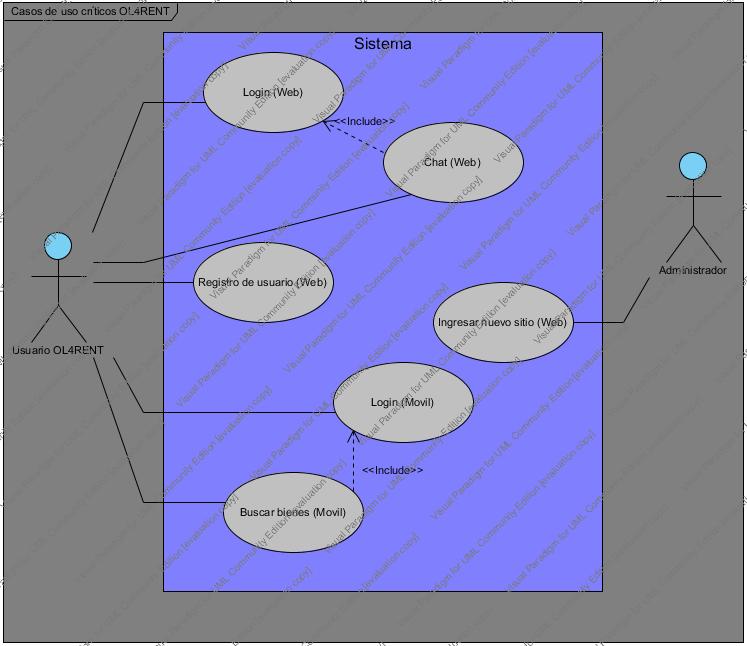
Los usuarios registrados tienen la posibilidad de loguearse para acceder a funcionalidades privilegiadas.

 *Buscar bienes cercanos.*

Permite al usuario logueado la búsqueda de bienes cercanos al usuario tomando como base su posición geográfica.

## 

## Diagrama de Casos de Uso Críticos



## Actores

### Usuario

Personas que están registradas en algún sitio de OL4RENT.

### Administrador

Son quienes crean, configuran y monitorean los diferentes sitios de OL4RENT.

## Especificación de Casos de Uso Críticos

|  |
| --- |
| **Registro de Usuario (módulo Front Office)** |
| **Código**: FO\_RegistroUsuario |
| **Nombre**: Registro de Usuario |
| **Objetivo**: Permite el registro de un nuevo usuario en un sitio. |
| **Actores**: Usuario |
| **Precondiciones**: No existe el usuario en el sistema. |
| **Descripción:** El usuario ingresa sus datos, el sistema valida los datos y registra al usuario en el sistema. |
| **Flujo Normal**:   1. El Usuario ingresa sus datos. 2. El Sistema verifica que los datos ingresados sean correctos. 3. El Usuario queda registrado correctamente en el sistema. |
| **Flujos Alternativos**:  1A. Los datos ingresados no son correctos.   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal.   2A. El usuario ya está registrado en el sistema   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal. |
| **Poscondiciones**: El usuario queda registrado en el sistema. |

|  |
| --- |
| **Login en el sistema (módulo Front Office)** |
| **Código**: FO\_Login |
| **Nombre**: Login Web |
| **Objetivo**: Autenticar al Usuario en el portal web. |
| **Actores**: Usuario |
| **Precondiciones**: No tiene |
| **Descripción:** El Usuario ingresa su nombre y password, el sistema valida los datos e ingresa al usuario al portal web. |
| **Flujo Normal**:   1. El Usuario ingresa su nombre y contraseña. 2. El Sistema verifica que los datos ingresados sean correctos. |
| **Flujos Alternativos**:  2A. La combinación nombre de usuario/password es incorrecta.   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal.   2B. El usuario no está registrado en el sistema   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal. |
| **Poscondiciones**: El usuario queda logueado en el sistema. |

|  |
| --- |
| **Conversación con usuario en línea (módulo Front Office)** |
| **Código**: FO\_Chat |
| **Nombre**: Chatear |
| **Objetivo**: Poder comunicarse con los demás usuarios del sitio. |
| **Actores**: Usuario |
| **Precondiciones**: Estar conectado al chat. |
| **Descripción:** Permite a dos usuarios autenticados en el sitio al cual se registraron previamente, comunicarce entre sí. |
| **Flujo Normal**:   1. El usuario envía mensaje (CU Enviar Mensaje). 2. El usuario recibe mensajes (CU Recibir Mensaje). |
| **Flujos Alternativos**: |
| **Poscondiciones**: |

|  |
| --- |
| **Enviar Mensaje (módulo Front Office)** |
| **Código**: FO\_EnviarMsj |
| **Nombre**: Enviar Mensaje |
| **Objetivo**: Escribir un mensaje a un destinatario |
| **Actores**: Usuario |
| **Precondiciones**: El usuario debe estar conectado al chat.  El destinatario debe estar conectado. |
| **Descripción:** Permite a un usuario conectado enviar un mensaje a otro usuario conectado. |
| **Flujo Normal**:   1. El usuario se conecta al chat. 2. El usuario selecciona a la persona que desea enviar mensaje. 3. Sistema valida el estado del receptor. 4. El usuario selecciona la opción “Enviar”. 5. El mensaje es guardado en el historial de mensajes de los usuarios. |
| **Flujos Alternativos**: |
| **Poscondiciones**: El mensaje queda guardado en historial de mensajes. |

|  |
| --- |
| **Recibir Mensaje (módulo Front Office)** |
| **Código**: FO\_RecibirMsj |
| **Nombre**: Recibir Mensaje |
| **Objetivo**: Recibir mensajes de otros usuarios del chat |
| **Actores**: Usuario |
| **Precondiciones**: El usuario debe conectarse al chat.  El destinatario debe estar conectado. |
| **Descripción:** Permite a un usuario conectado recibir mensajes de otros usuarios conectado. |
| **Flujo Normal**:   1. El usuario se conecta al chat. 2. El usuario selecciona enviar mensajes. 3. Sistema “Recupera mensaje”. 4. Sistema envía respuesta a destinatario. |
| **Flujos Alternativos**: |
| **Poscondiciones**: El usuario recibe mensajes de destinatarios. |

|  |
| --- |
| **Ingresar nuevo sitio (módulo BackOffice)** |
| **Código**: BO\_IngresarSitio |
| **Nombre**: Ingresar nuevo sitio |
| **Objetivo**: Ingresar un nuevo sitio en el sistema. |
| **Actores**: Administrador |
| **Precondiciones**: El Usuario Administrador está logueado. |
| **Descripción:** Permite al administrador autenticado dar de alta un nuevo sitio en el sistema. Para este caso consideramos el flujo principal del caso de uso. |
| **Flujo Normal**:   1. El Administrador ingresa los datos del nuevo sitio a ingresar. 2. El Sistema verifica que los datos ingresados sean correctos. 3. El sitio queda ingresado correctamente en el sistema. |
| **Flujos Alternativos**:  1A. Los datos ingresados no son correctos.   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal.   2A. Ya existe un sitio con ese nombre en el sistema.   1. El sistema despliega un mensaje de error acorde.   Vuelve a paso 1 del Flujo normal. |
| **Poscondiciones**: El sitio queda ingresado en el sistema. |

|  |
| --- |
| **Login en el sistema (módulo Móvil)** |
| **Código**: MO\_Login |
| **Nombre**: Login Móvil |
| **Objetivo**: Autenticar al Usuario en el portal web. |
| **Actores**: Usuario |
| **Precondiciones**: No tiene |
| **Descripción:** Permite a un usuario anónimo autenticarse en el sistema a través de un dispositivo movil. Para este caso consideramos el flujo principal del caso de uso. |
| **Flujo Normal**:   1. El Usuario ingresa su nombre y contraseña. 2. El Sistema verifica que los datos ingresados sean correctos. 3. El usuario queda autenticado correctamente en el sistema. |
| **Flujos Alternativos**:  1A. La combinación nombre de usuario/password es incorrecta.   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal.   2A. El usuario no está registrado en el sistema   1. El sistema despliega un mensaje de error acorde. 2. Vuelve a paso 1 del Flujo normal. |
| **Poscondiciones**: El usuario queda logueado en el sistema. |

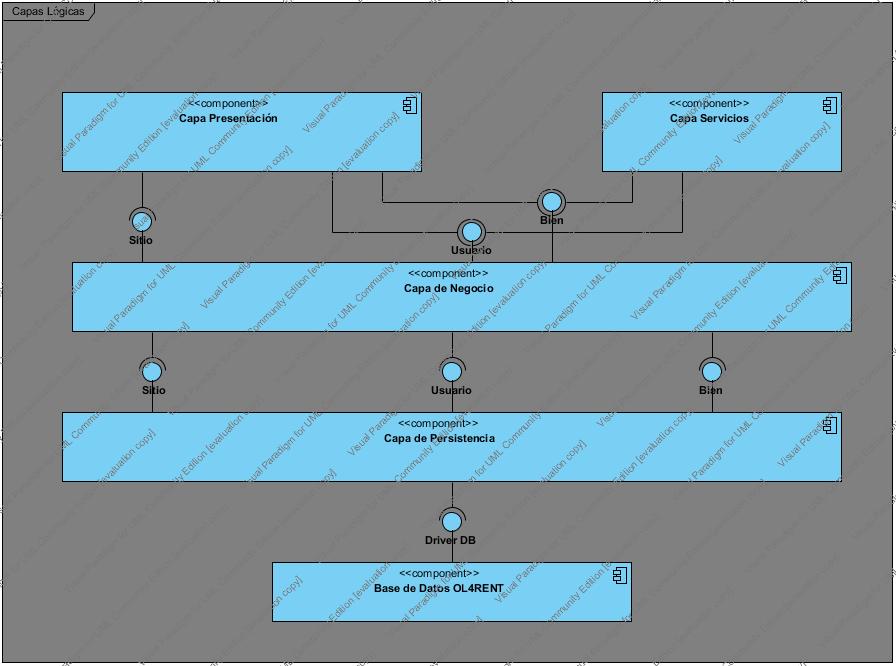
|  |
| --- |
| **Buscar bienes cercanos (módulo Móvil)** |
| **Código**: MO\_BBC |
| **Nombre**: Buscar bienes cercanos |
| **Objetivo**: Buscar vienes cercanos al usuario. |
| **Actores**: Usuario |
| **Precondiciones**: No tiene |
| **Descripción:** Permite al usuario logueado la búsqueda de bienes cercanos al usuario tomando como base su posición geográfica. Para este caso consideramos el flujo principal del caso de uso. |
| **Flujo Normal**:   1. El usuario pide ver los bienes cercanos a su posición. 2. El Sistema obtiene coordenadas de posición del usuario. 3. Se envía lista con bienes al usuario. |
| **Flujos Alternativos**: |
| **Poscondiciones**: |

# Vista Lógica

La vista lógica permite describir el sistema en base a abstracciones fundamentales del diseño orientado a objetos para dar soporte a los requerimientos funcionales. Adoptando un enforque top-down, comenzaremos por descomponer el sistema en un conjunto de subsistemas o "capas lógicas", y a partir de ellos realizaremos sucesivos refinamientos hasta llegar a las unidades lógicas mas pequeñas.

## Estilo Arquitectónico.

El estilo arquitectónico en el cual nos basaremos para el desarrollo de este proyecto será el de "Arquitectura en Capas Estrictas". En este estilo cada capa consume las interfaces definidas de su inmediata inferior. En el caso particular del diseño de esta aplicación, tenemos definidas cuatro capas lógicas: Presentación, Servicios en un mismo nivel jerárquico y sin dependencias entre sí, la Capa Lógica que contiene todas las reglas de negocio y finalmente la Capa de Persistencia.



***3.2 Subsistemas***

Capa de Presentación

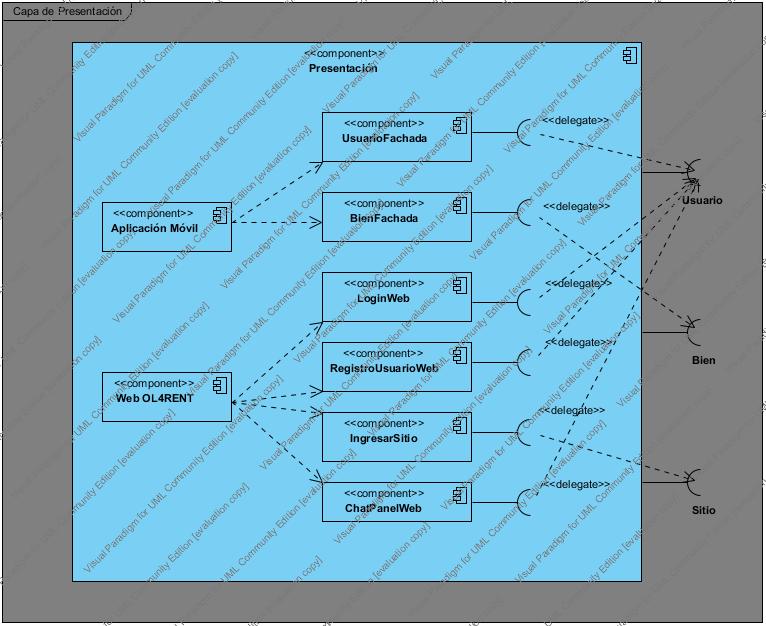
La capa de Presentación contiene todos los componentes que forman la interfaz de usuario (o UI). Dentro de este componente encontramos páginas cshtml (o vistas), estilos así como los controladores definidos para los diferentes casos de uso.

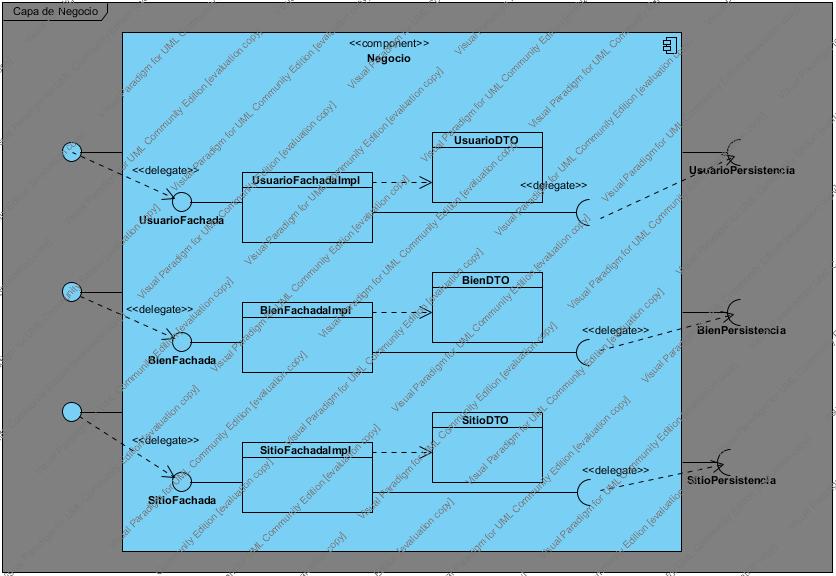
Capa de Servicios

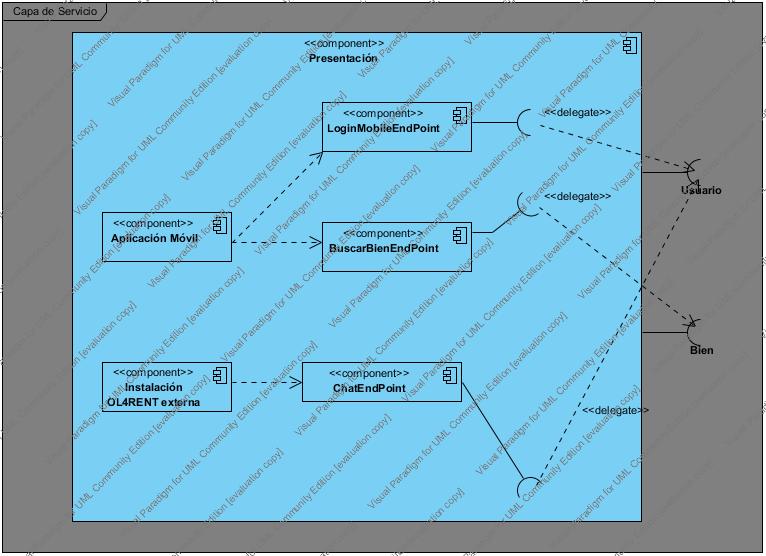
La finalidad de la capa de Servicios es la de exponer un subconjunto de todas las funcionalidades brindadas por el sistema a clientes remotos. Dichos clientes pueden ser tanto usuarios de la aplicación así como otros sistemas (comunicación entre aplicaciones).

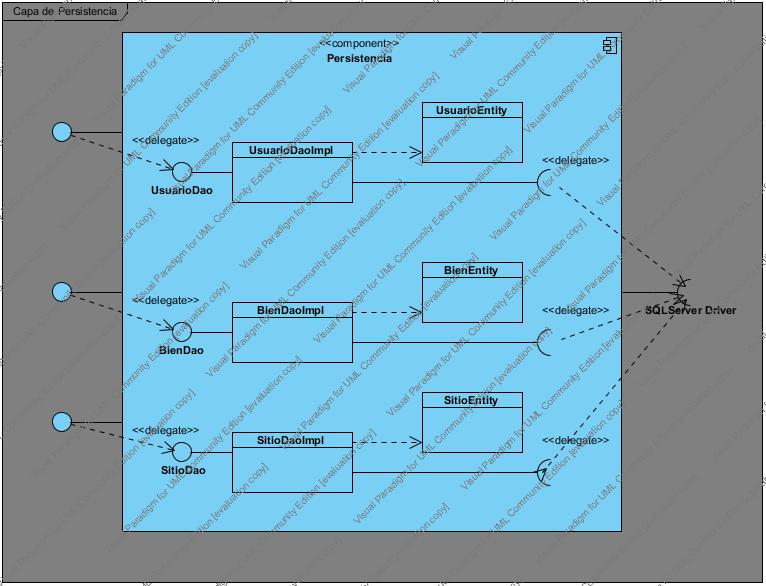
Capa de Negocio

La Capa de Negocio implementa todas las funcionalidades del sistema respetando todas las reglas de negocio definidas. Esta capa también se conoce como Capa Lógica o Capa de Dominio.

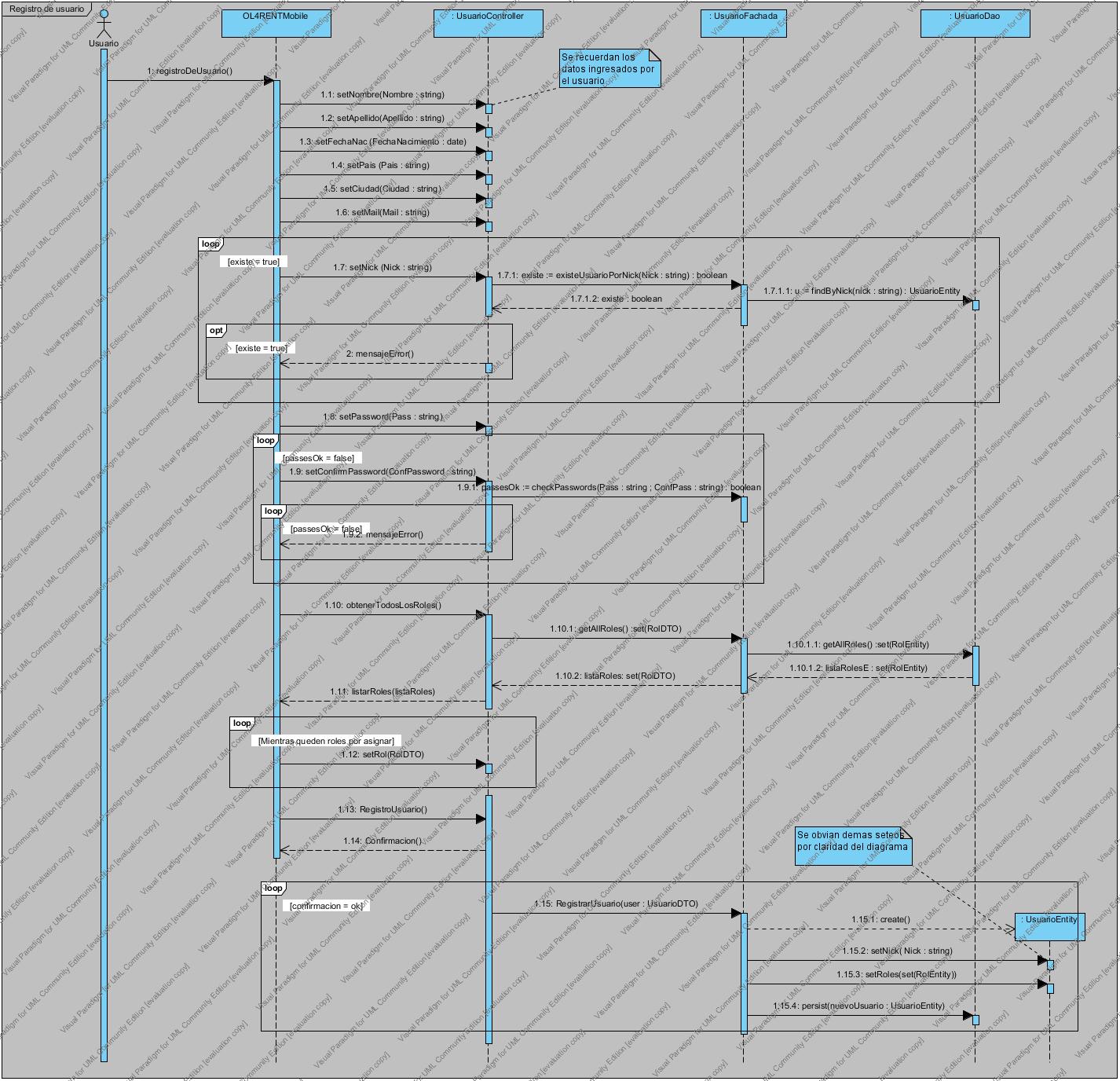


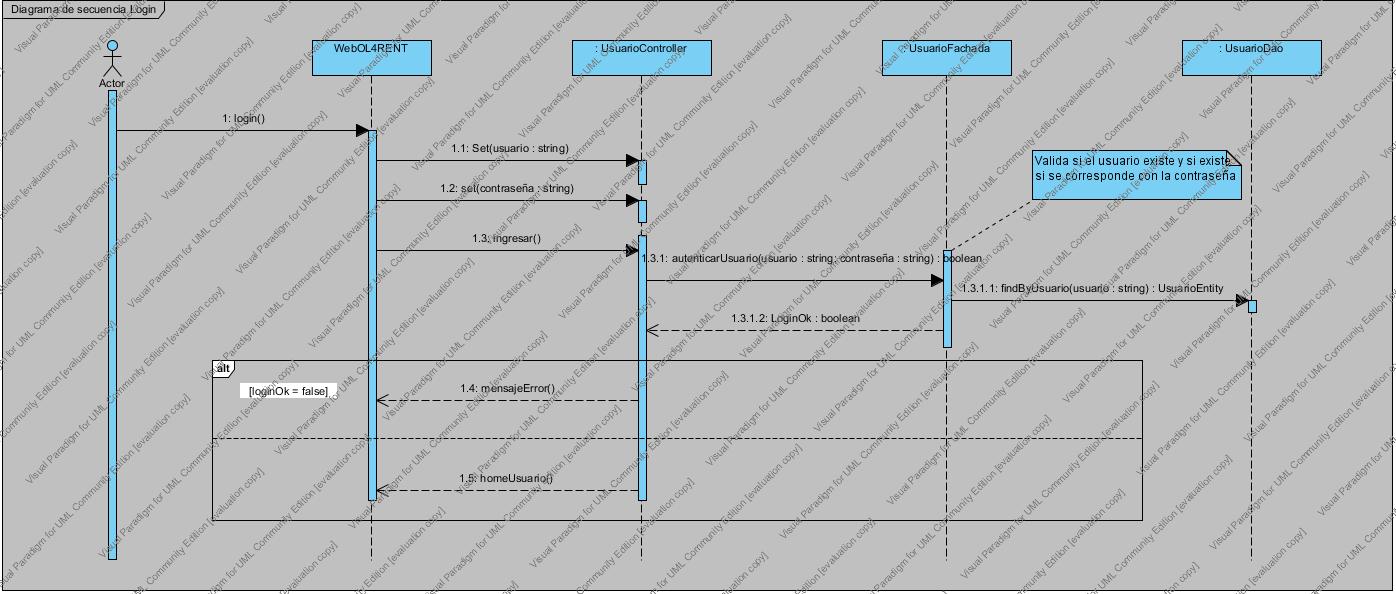


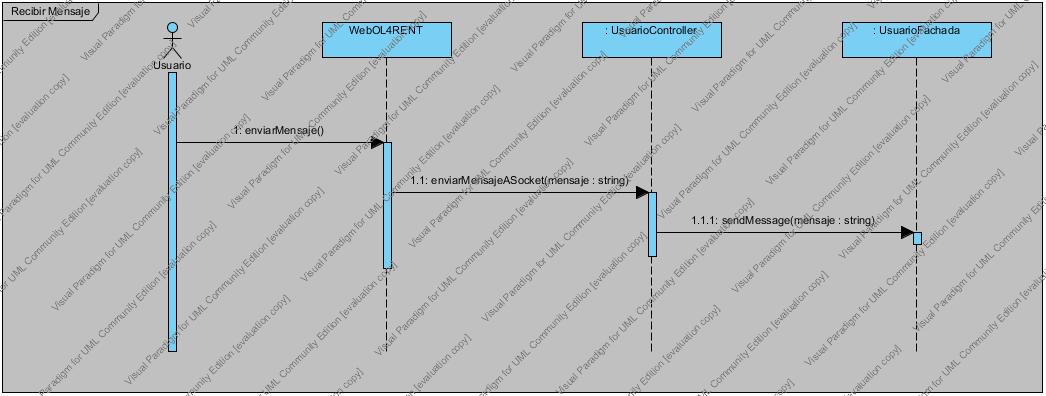


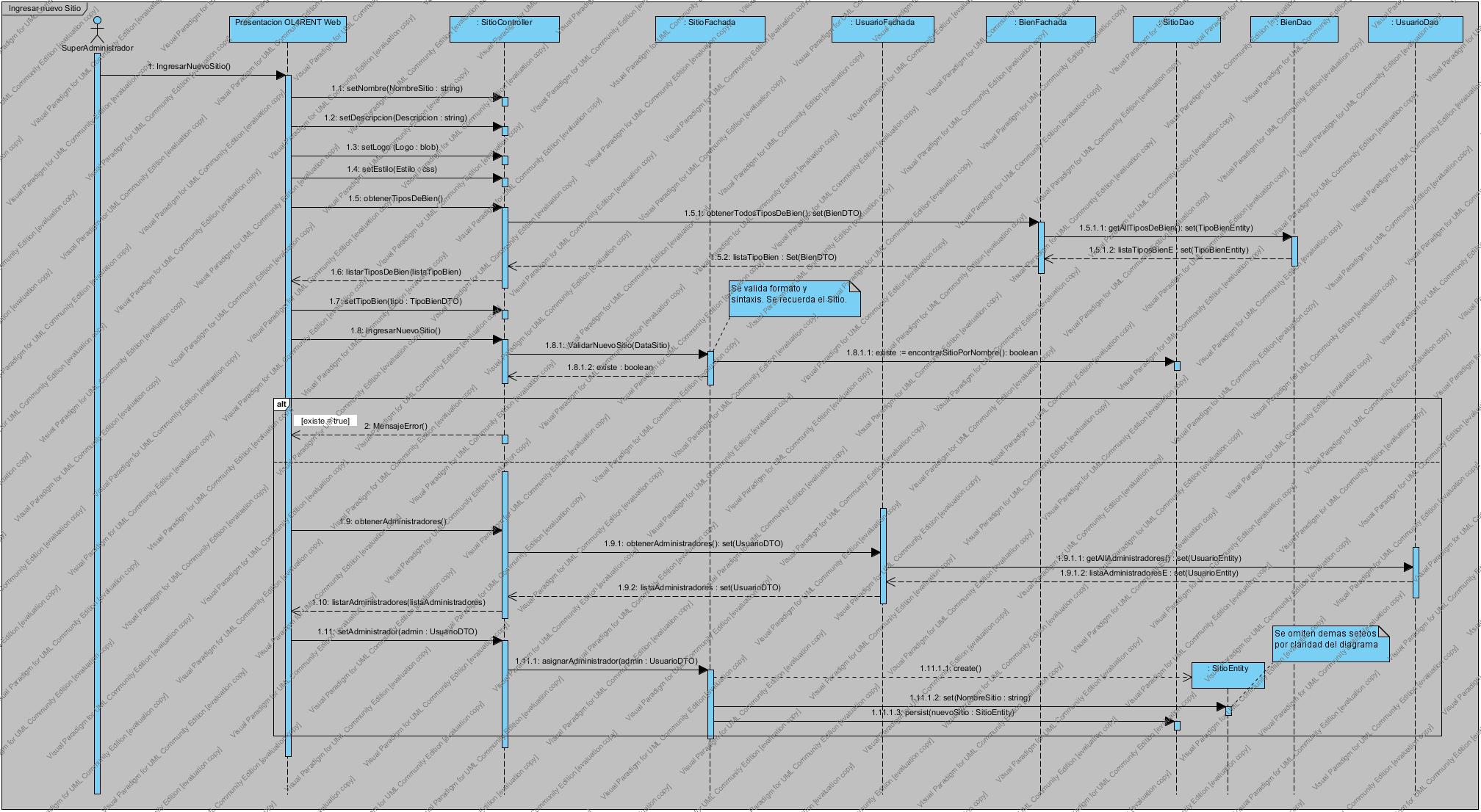
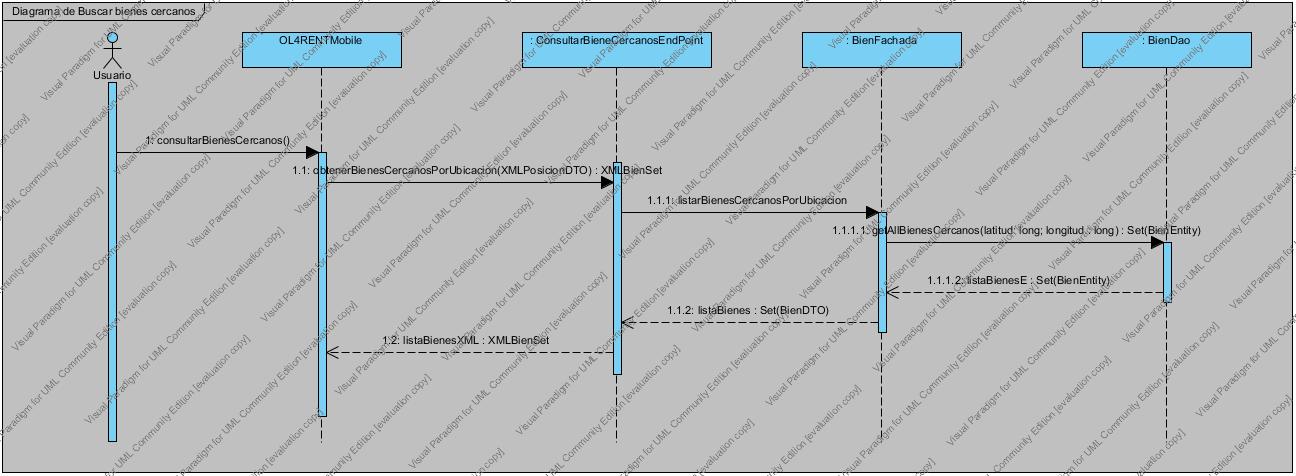


## Diagramas de Interacción









# Vista de Distribución (Deployment).

La vista de distribución permite presentar los diferentes escenarios de asignación de los componentes lógicos del sistema en nodos físicos o tiers. Presentaremos estos escenarios con Diagramas de Deployment de UML, que permite representar la estructura de dichos escenarios en base a nodos, conexiones entre los nodos y componentes en los mismos. Luego daremos una breve descripción de cada tier.

Los nodos que forman el sistema en ambos escenarios son los siguientes:

* **Browser:** Es un navegador estándar instalado en el cliente web como puede ser Mozilla FireFox, Internet Explorer, Chrome, etc.
* **Cliente Web Services:** Puede ser una otra aplicación o usuario que actúe como cliente de los Web Services publicados por OL4RENT.
* **Cliente Mobile:** Una aplicación para dispositivos móviles desarrollada para la plataforma Windows Phone, posiblemente será tanto cliente de la interfaz Web así como cliente Web Services.
* **Servidor de Aplicaciones:** Es el servidor que corre el
* **Servidor de Base de Datos:** Es el servidor que corre el manejador de base de datos. En este caso será SQL Server 2012 de Microsoft.

Presentaremos brevemente dos escenarios para las diferentes etapas del desarrollo de la aplicación. Un primer escenario, pensado para la etapa de desarrollo y otro ideal para la etapa de producción, mencionando para cada uno las ventajas y desventajas.

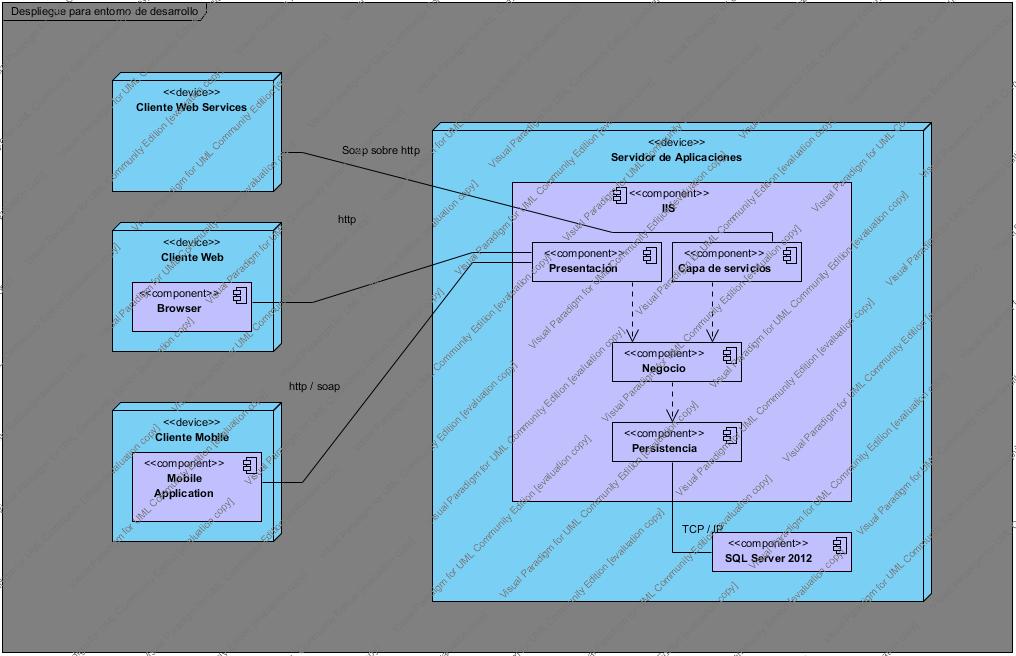
**Escenario de deploy para desarrollo**

El escenario para etapa de desarrollo involucra un único nodo, en el cual se ejecutan todos los componentes del sistema, como son Web browser, simulador Windows Phone, Servidor de Aplicaciones así como el motor de base de datos. Este esquema presenta gran simplicidad y bajo costo al momento de preparar el ambiente de desarrollo. Como contrapartida, presentaría problemas de performance, escalabilidad y seguridad para la aplicación si se considerara para un ambiente de producción.

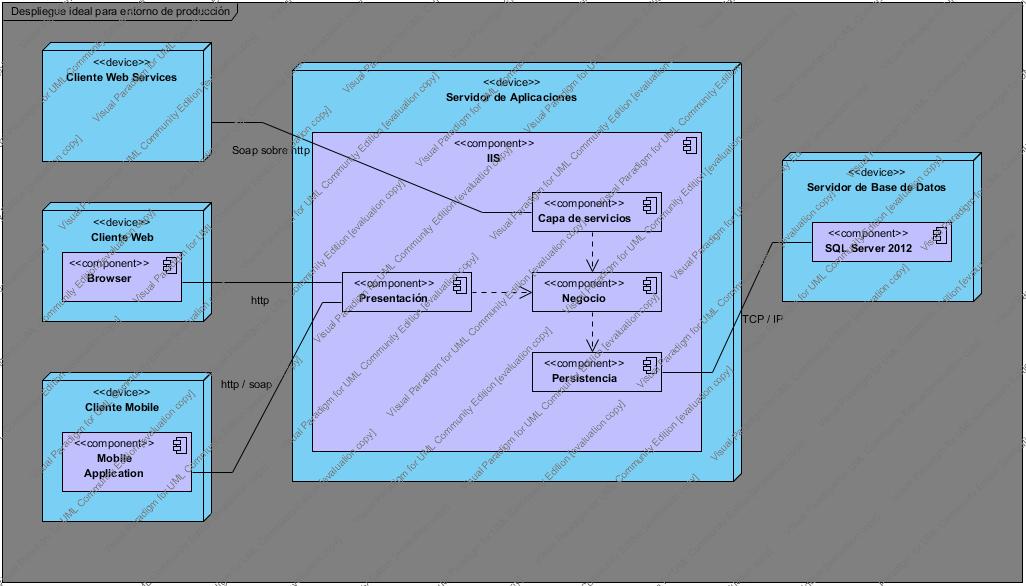
**Escenario de deploy para desarrollo**

Este escenario presenta los componentes implantados en nodos totalmente independientes, un nodo corriendo el Servidor de Aplicaciones, otro con el Servidor de Base de Datos y otros nodos (clientes) ejecutando los browsers y las aplicaciones móviles consumiendo la aplicación. Esto promueve la escalabilidad y performance del sistema, así como la rápida recuperación del sistema ante posibles fallas en alguno de los nodos.

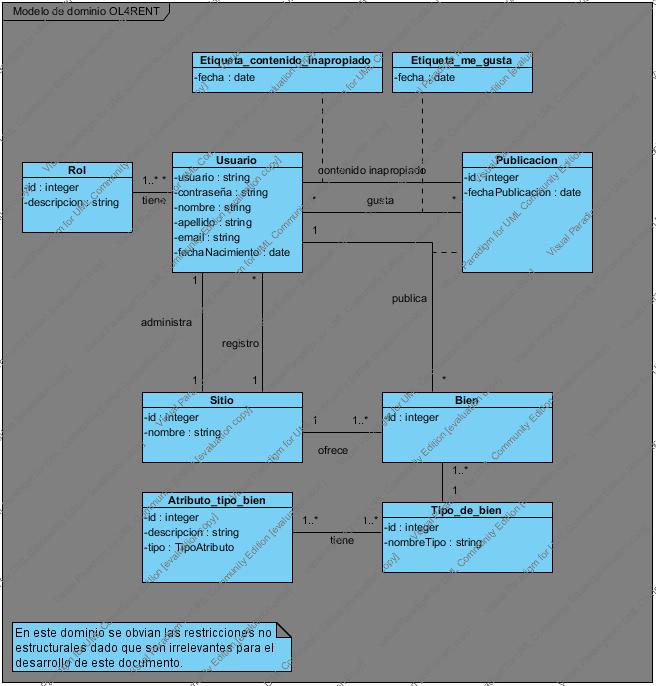
Escenario despliegue de desarrollo



Escenario despliegue de desarrollo



# Modelo de Dominio.

De la etapa de análisis se llegó al siguiente modelo de dominio, que representa los principales conceptos que intervienen en la realidad planteada para la aplicación a desarrollar.

# Referencias

1. Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives. N. Rozanski, E. Woods. Addison-Wesley, 2005
2. Software Architecture in practice, Second Edition. L. Bass, P. Clemens, R. Kazman. Addison-Wesley, 2003
3. An Introduction to Software Architecture. D. Garlan, M. Shaw. 1994. <http://www.cs.cmu.edu/afs/cs/project/able/ftp/intro_softarch/intro_softarch.pdf>
4. Architectural Blueprints — The “4+1” View Model of Software Architecture. Kruchten, Philippe. 1995. <http://www.cs.ubc.ca/~gregor/teaching/papers/4+1view-architecture.pdf>
5. Software Architecture Links. Bredemeyer Consulting. <http://www.bredemeyer.com/links.htm>
6. <https://eva.fing.edu.uy/course/view.php?id=402>.
7. Grupo 13, TSI1 .NET edición 2013. Documento de Arquitectura. 2013