

# **SAD del Sistema de Vetado para Access Now**

Marcos Calderón

Tecnológico de Costa Rica

Proyecto de Ingeniería de Software

Profesora: Ing María Estrada Sánchez Msc.

Enero, 2017

## Tabla de contenido

1.	Introducción .....	4
1.1.	Propósito .....	4
1.2.	Alcance .....	4
1.3.	Referencias .....	4
1.4.	Organización.....	4
2.	Representación de la Arquitectura .....	5
2.1.	Representación .....	5
3.	Vista de Casos de Uso.....	6
3.1.	Descripción del Negocio.....	6
3.1.1.	Sistema de Vetado.....	6
3.1.2.	Procesos de Negocio .....	6
3.2.	Modelo de Dominio .....	7
3.3.	Actores .....	7
3.4.	Casos de Uso .....	8
3.4.1.	Diagrama de Casos de Uso .....	8
3.4.2.	Caso de Uso: Administrar Usuarios, Clientes, Organizaciones.....	9
3.4.3.	Caso de Uso: Realizar Búsqueda .....	10
3.4.4.	Caso de Uso: Crear Nuevo Proceso .....	11
4.	Vista de Restricciones.....	12
4.1.	Normativas .....	12
4.1.1.	Licenciamiento .....	12
4.2.	Estándares .....	12
4.2.1.	UML .....	12
4.2.2.	Interfaz Web.....	12
4.2.3.	Tecnología .....	12
5.	Vista QoS .....	13
5.1.	Usabilidad.....	13
6.	Vista Lógica.....	13
6.1.	Arquitectura del Sistema.....	14
6.2.	Arquitectura Lógica .....	14
6.2.1.	Interfaz de Usuario.....	15
6.2.2.	Servicios de Sistema .....	15

6.2.3.	Servicios de Negocio .....	15
6.2.4.	Infraestructura .....	15
7.	Vista de Procesos .....	16
7.1.	Procesos Distribuidos.....	16
7.1.1.	Interfaz de Usuario.....	16
7.1.2.	Servicios de Infraestructura .....	16
8.	Vista de Datos.....	17
8.1.	Modelo de Datos .....	17
8.2.	Distribución .....	17
8.3.	Servicios de Persistencia .....	18
9.	Vista de Deployment .....	18
9.1.	Tecnología requerida .....	18

# 1. Introducción

El Sistema de Vetado es el proyecto desarrollado por el estudiante Marcos Calderón como parte del curso de Proyecto de Ingeniería de Software. El presente documento provee una vista de alto nivel de la arquitectura del Sistema de Vetado, los objetivos y restricciones, los casos de uso significativos, los estilos de arquitectura aplicados y las principales decisiones de diseño. Este documento da una vista general del resto de los artefactos generados en el proceso de desarrollo. Son dichos documentos quienes guiarán el desarrollo del producto final.

## 1.1. Propósito

Este documento de arquitectura de software (por sus siglas en inglés, SAD) tiene como propósito brindar una visión comprensible de la arquitectura general del Sistema de Vetado, utilizando diferentes vistas de la arquitectura para ilustrar diferentes aspectos del sistema. Captura las decisiones más importantes en lo que respecta a la arquitectura del sistema que fueron tomadas en el proyecto.

Este SAD está dirigido a distintos tipos de actores: docentes y desarrolladores.

Desde el punto de vista de un desarrollador, este documento le brindará, con certeza, una buena razón para darle a la arquitectura de software la importancia que tiene en todo proyecto de desarrollo.

## 1.2. Alcance

Este SAD profundiza principalmente en las vistas de casos de uso y lógica, incluyendo algunos elementos significativos de las otras vistas.

El Sistema de Vetado consta de cinco casos de uso principales, siendo tres de ellos mostrados en esta versión del documento. Se atacará principalmente aquellos casos de uso que involucran interacción con los actores.

## 1.3. Referencias

[RUP] IBM Rational Unified Process. <http://www.rational.com/rup>.

[ERS] Especificación de Requerimientos de Software, versión 1, 2017

## 1.4. Organización

Este documento está organizado en capítulos, correspondiendo cada uno de ellos a los sugeridos en el template provisto en [RUP].

El capítulo 2 presenta la representación de la arquitectura para el caso de estudio, esto es, la descripción de las vistas necesarias y el framework arquitectónico utilizado. Los siguientes capítulos presentan la descripción de las vistas indicadas.

## 2. Representación de la Arquitectura

El Sistema de Vetado es un sistema de información en el dominio de Servicios Informáticos, y concierne principalmente a la gestión de clientes de una organización.

La característica de ser un sistema de información hace que la vista de casos de uso y la vista lógica sean las más relevantes y por ello serán las más extensas.

La arquitectura está representada usando un modelo UML al nivel de abstracción correspondiente de forma que permita visualizar, entender y razonar sobre los elementos significativos de la arquitectura e identificar las áreas de riesgo que requieren mayor detalle de elaboración.

La arquitectura del subsistema se descompone en las siguientes dimensiones:

- Requerimientos: Requerimientos funcionales y no-funcionales del sistema.
- Elaboración: Representación lógica del sistema y representación de tiempo de ejecución.
- Implementación: Vista de módulos implementados, potenciales escenarios de infraestructura y el deployment de los módulos.

La siguiente sección detalla las vistas de la arquitectura que serán utilizadas para cubrir las dimensiones mencionadas.

### 2.1. Representación

La arquitectura del Sistema de Vetado está representada siguiendo las recomendaciones de [RUP]. Las vistas necesarias para especificar dicho subsistema se enumeran a continuación:

- Vista de Casos de Uso: Describe el proceso de negocio más significativo y el modelo del dominio. Presenta los actores y los casos de uso para el sistema.
- Vista de Restricciones: Describe restricciones tecnológicas, normativas, uso de estándares, entre otros, las cuales deben ser respetadas tanto por el proceso de desarrollo como por el producto desarrollado.
- Vista QoS: Incluye aspectos de calidad, y describe los requerimientos no funcionales del sistema.
- Vista Lógica: Describe la arquitectura del sistema presentando varios niveles de refinamiento. Indica los módulos lógicos principales, sus responsabilidades y dependencias.
- Vista de Datos: Presenta los modelos de datos y los servicios de persistencia
- Vista de Deployment: Presenta aspectos físicos como topología, infraestructura informática, e instalación de ejecutables. Incluye además plataformas y software de base.

### 3. Vista de Casos de Uso

Esta vista presenta la percepción que tiene el usuario de las funcionalidades del sistema. Se presenta el proceso de negocio más importante del sistema y los casos de uso críticos que se derivan de éste. Este capítulo provee el contexto y determina el alcance del resto del documento.

Primeramente, se describe el Negocio. Luego se presenta el modelo del dominio para el Sistema de Vetado. Se identifican actores y se detallan los casos de uso significativos.

#### 3.1. Descripción del Negocio

##### 3.1.1. Sistema de Vetado

Una organización sin fines de lucro desea automatizar los servicios brindados por su línea de ayuda. Este departamento posee un sistema de información que satisface parcialmente los requerimientos informáticos reales de la organización. Muchas actividades son registradas en formularios de papel y la obtención de datos estadísticos insume gran cantidad de recursos.

La organización desea mantener en forma central y unificada todas las interacciones que realizan con sus clientes. Como política de la organización se realiza un proceso de verificación de cada nuevo cliente que ingrese a la línea de ayuda en busca de asistencia, por lo cual se debe iniciar un proceso de vetado del cliente para garantizar seguridad y confidencialidad. Es prioritario este requerimiento.

La entidad necesita información estadística. Ésta es utilizada para ser presentada antes sus principales patrocinadores como forma de evidencia de su trabajo en la sociedad civil.

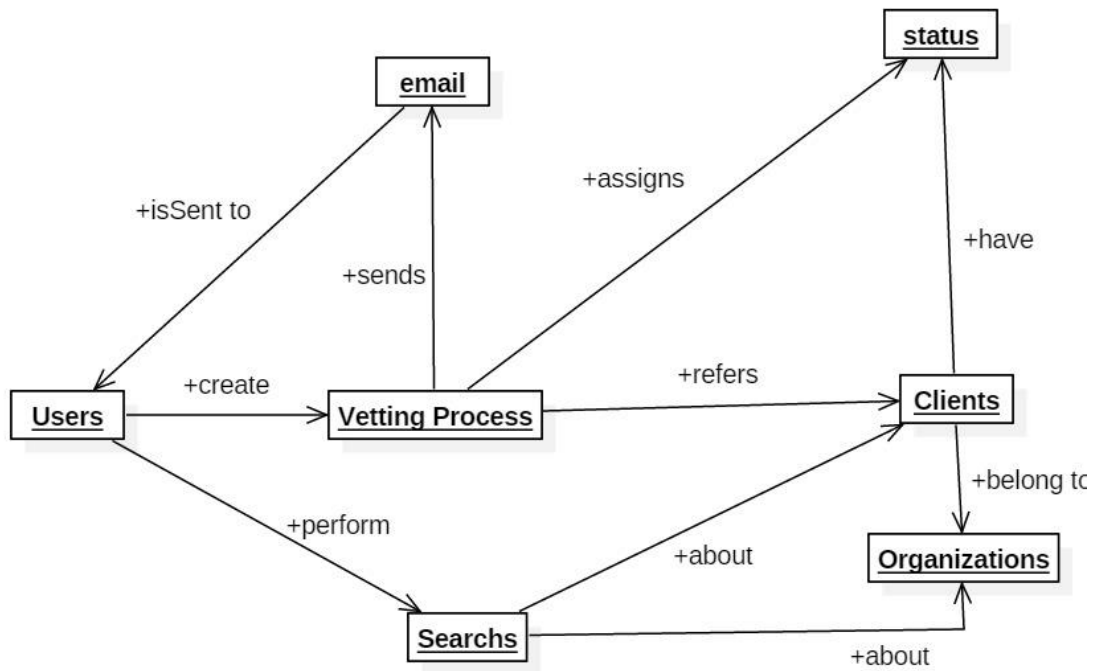
##### 3.1.2. Procesos de Negocio

Los siguientes procesos de negocio son relativos al Sistema de Vetado:

- Creación de Proceso de Vetado
  - Este proceso involucra el inicio de un caso de verificación de cada nuevo cliente que requiera asistencia.
- Consultas Estadísticas
  - Este proceso ocurre cuando se realiza un estudio de la situación de los clientes de la organización. Mediante este proceso se extraerá la información del sistema útil para analizar la diversidad de población de los clientes de la línea de ayuda.

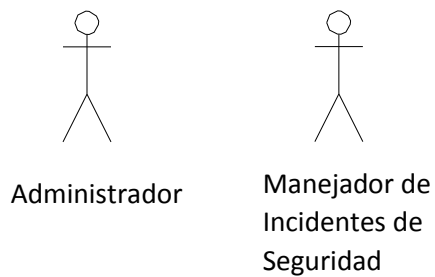
Ambos procesos tienen gran impacto en la arquitectura del sistema; en cambio, las repercusiones sobre ésta es similar en ambos casos.

### 3.2. Modelo de Dominio



### 3.3. Actores

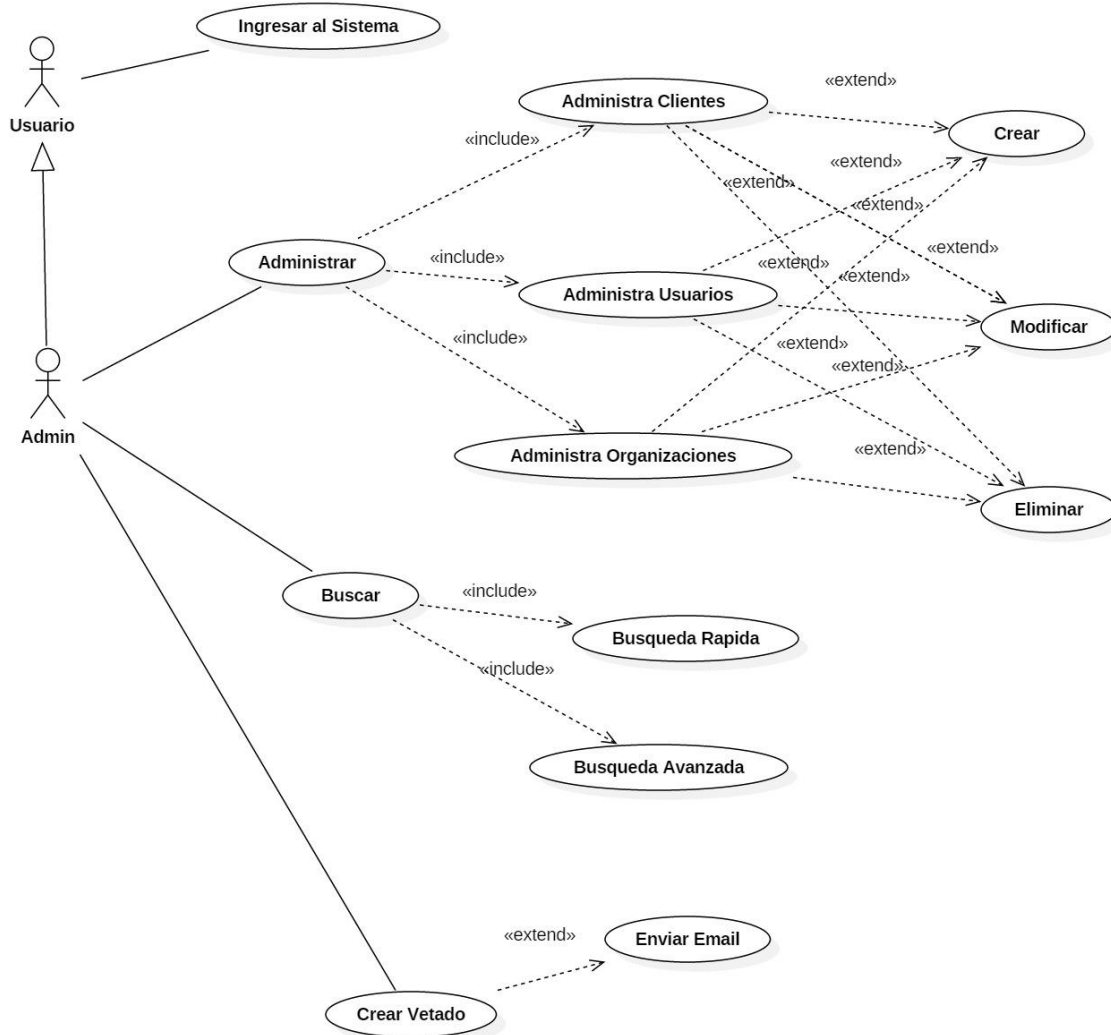
Los siguientes actores son los que interactuarán con el Subsistema de Reservas una vez realizado el deploy.



### 3.4. Casos de Uso

Los casos de uso críticos para el proceso se describen en esta sección. Primero se indica las relaciones entre los casos de usos detectados, para luego presentar la versión expandida de los mismos.

#### 3.4.1. Diagrama de Casos de Uso





### 3.4.2. Caso de Uso: Administrar Usuarios, Clientes, Organizaciones

<b>ID Caso de Uso: CU-01</b>	<b>Creado por: Marcos Calderón</b>
<b>Nombre: Administración</b>	<b>Fecha de creación: 05 enero 2017</b>
<b>Actor: Administrador</b>	<b>Fecha de modificación: 06 enero 2017</b>
<b>Precondiciones:</b>	La base de datos tiene suficiente espacio para almacenar nuevas inserciones y modificaciones
<b>Descripción:</b>	El usuario administrador se encarga de realizar labores de creación, modificación y eliminación de Usuarios, Clientes y Organizaciones
<b>Post-condiciones:</b>	Existen nuevos datos registrados en el sistema y en la base de datos
<b>Flujo normal:</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona las entidades por administrar</li> <li>2. El usuario ejecuta la creación, modificación o eliminación de la entidad seleccionada</li> <li>3. El usuario confirma las acciones realizadas</li> <li>4. El sistema registra las acciones realizadas</li> <li>5. El sistema actualiza la base de datos</li> <li>6. El sistema muestra un mensaje de confirmación</li> </ol>
<b>Flujos alternativos:</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona las entidades por administrar</li> <li>2. El usuario ejecuta la creación, modificación o eliminación de la entidad seleccionada</li> <li>3. El usuario no confirma las acciones realizadas</li> <li>4. El sistema se mantiene sin cambios</li> </ol>
<b>Excepciones:</b>	<ul style="list-style-type: none"> <li>• El usuario cancela las acciones antes de finalizarlas</li> <li>• Se pierde conexión con el sistema</li> </ul>
<b>Prioridad:</b>	Alta
<b>Frecuencia de Uso:</b>	Alta
<b>Reglas de negocio:</b>	N/A

### 3.4.3. Caso de Uso: Realizar Búsqueda

<b>ID Caso de Uso: CU-01</b>  <b>Nombre:</b> Realizar Búsqueda  <b>Actor:</b> Administrador	<b>Creado por:</b> Marcos Calderón <b>Fecha de creación:</b> 05 enero 2017 <b>Fecha de modificación:</b> 06 enero 2017
<b>Precondiciones:</b>	Existe conexión con la base de datos
<b>Descripción:</b>	El usuario administrador se encarga de realizar búsqueda de Clientes y Organizaciones
<b>Post-condiciones:</b>	Resultado es mostrado en el sistema
<b>Flujo normal:</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona Realizar Búsqueda</li> <li>2. El usuario ingresa los parámetros deseados para la búsqueda</li> <li>3. El usuario confirma la búsqueda</li> <li>4. El sistema ejecuta la búsqueda solicitada</li> <li>5. El sistema muestra los resultados de la búsqueda</li> </ol>
<b>Flujos alternativos:</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona Realizar Búsqueda</li> <li>2. El usuario ingresa los parámetros deseados para la búsqueda</li> <li>3. El usuario no confirma la búsqueda</li> <li>4. El sistema se mantiene sin cambios</li> </ol>
<b>Excepciones:</b>	<ul style="list-style-type: none"> <li>• El usuario cancela la búsqueda antes de finalizarla</li> <li>• Se pierde conexión con el sistema</li> </ul>
<b>Prioridad:</b>	Alta
<b>Frecuencia de Uso:</b>	Alta
<b>Reglas de negocio:</b>	N/A

#### 3.4.4. Caso de Uso: Crear Nuevo Proceso

<b>ID Caso de Uso: CU-01</b> <b>Nombre: Nuevo Proceso</b> <b>Actor: Administrador</b>	<b>Creado por: Marcos Calderón</b> <b>Fecha de creación: 05 enero 2017</b> <b>Fecha de modificación: 06 enero 2017</b>
<b>Precondiciones:</b>	La base de datos tiene suficiente espacio para almacenar nuevas inserciones y modificaciones
<b>Descripción:</b>	El usuario administrador se encarga de crear un nuevo proceso de vetado para un cliente, el cual se le asignará un estado de Vetado o No Vetado y se agregará información relevante que fundamente dicha decisión.
<b>Post-condiciones:</b>	Existen nuevos datos registrados en el sistema y en la base de datos
<b>Flujo normal:</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona Vetar Cliente</li> <li>2. El usuario ingresa los valores deseados para el nuevo vetado</li> <li>3. El usuario confirma la creación del vetado</li> <li>4. El sistema ejecuta la creación</li> <li>5. El sistema actualiza la base de datos</li> <li>6. El sistema muestra los resultados de la búsqueda</li> </ol>
<b>Flujos alternativos:</b>	<ol style="list-style-type: none"> <li>1. El usuario selecciona Vetar Cliente</li> <li>2. El usuario ingresa los valores deseados para el nuevo vetado</li> <li>3. El usuario no confirma la creación del vetado</li> <li>4. El sistema se mantiene sin cambios</li> </ol> <ol style="list-style-type: none"> <li>1. El usuario selecciona Vetar Cliente</li> <li>2. El usuario ingresa los valores deseados para el nuevo vetado</li> <li>3. El usuario confirma la creación del vetado</li> <li>4. Los valores ingresados no son válidos</li> <li>5. El sistema muestra un mensaje de error</li> <li>6. El sistema se mantiene sin cambios</li> </ol>
<b>Excepciones:</b>	<ul style="list-style-type: none"> <li>• El usuario cancela la acción antes de finalizarla</li> <li>• Se pierde conexión con el sistema</li> </ul>
<b>Prioridad:</b>	Alta
<b>Frecuencia de Uso:</b>	Alta
<b>Reglas de negocio:</b>	N/A

## 4. Vista de Restricciones

En esta vista se presentan las restricciones normativas, de estándares y de tecnológicas, a las cuales está sujeto tanto el proceso de desarrollo como el producto desarrollado, incluidas en las categorías soporte, implementación e interfaces.

### 4.1. Normativas

Existen restricciones normativas, dictadas por organizaciones gubernamentales y no-gubernamentales, que determinan algunas decisiones del producto desarrollado.

#### 4.1.1. Licenciamiento

No existe regulación de licenciamiento para aplicaciones web en el país donde está radicada la organización. El licenciamiento del producto pesará totalmente sobre la aplicación back-end. Por esta razón el producto no debe limitar la cantidad de usuarios simultáneos que permite la aplicación.

### 4.2. Estándares

#### 4.2.1. UML

Todo artefacto utilizado para comunicación y documentación, tanto entre miembros del equipo de desarrollo como con los clientes y usuarios, está basado en UML.

#### 4.2.2. Interfaz Web

La interfaz de usuario debe estar orientada al web. Debe ser posible visualizar el contenido utilizando cualquiera de los browsers más difundidos.

#### 4.2.3. Tecnología

El desarrollo completo del Sistema debe estar realizado en la plataforma CakePHP, basándose fuertemente en las tecnologías involucradas en la plataforma como ser HTML, CSS. El lenguaje de programación es PHP.

## 5. Vista QoS

Describe los requerimientos no funcionales del Sistema dentro de las categorías usabilidad, confiabilidad y performance

### 5.1. Usabilidad

La interfaz de usuario orientada al web para todos los actores humanos del sistema disminuye la necesidad de capacitación de los usuarios. El sitio es simple, orientado a los casos de uso.

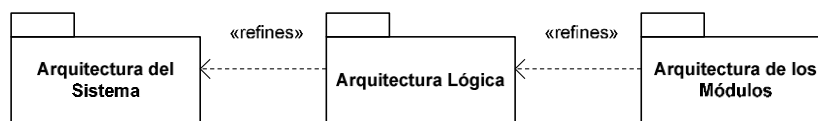
No se proveerá documentación de usuario impresa. Se limitará a un manual técnico en forma digital.

El Sistema de Vetado será utilizado por clientes de todo el mundo. Adicionalmente, la Organización exige que para anunciar servicios en su portal, éstos deben ser provistos en inglés.

## 6. Vista Lógica

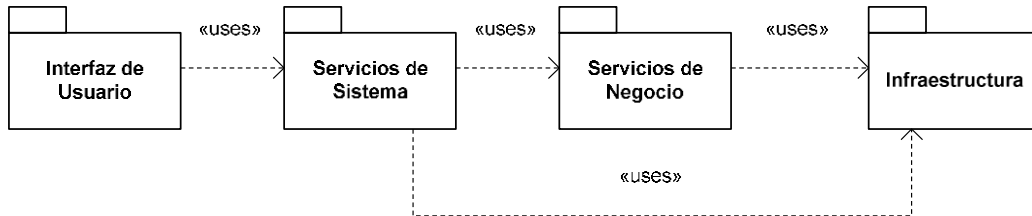
Esta vista presenta tres niveles de arquitectura para el Sistema de Vetado. Cada nivel corresponde a un refinamiento del nivel anterior. El último nivel es el que presenta mayores detalles; en él, se presentan los módulos participantes de la arquitectura junto a un modelo. El modelo utilizado varía según el módulo en cuestión, lo cual se debe principalmente a que los módulos tienen diferentes roles, según su ubicación en el nivel anterior.

Este capítulo se organiza de la siguiente forma:



## 6.1. Arquitectura del Sistema

En el primer nivel se especifica el estilo de arquitectura. El mismo es un estilo en capas, es relajado y se conforma de cuatro capas. El siguiente diagrama presenta la arquitectura del sistema.



- La Interfaz de Usuario tiene como objetivo el manejo de la lógica del usuario. Está conformada por el conjunto de páginas web dinámicas y por módulos que encapsulan la lógica de los casos de uso.
- Los Servicios del Sistema representan los servicios básicos que debe proveer el sistema; estos servicios son directamente utilizados por los módulos de la capa superior.
- Los Servicios de Negocio son servicios de manejo de información del negocio; son servicios aún más básicos que el de la capa superior.
- La capa de Infraestructura contiene todos los módulos necesarios para utilizar los servicios de la plataforma. En esta capa residen principalmente adaptadores de los servicios brindados por la plataforma.

## 6.2. Arquitectura Lógica

La Arquitectura Lógica presenta un refinamiento de la Arquitectura del Sistema. La dimensión Requerimientos, principalmente la Vista de Casos de uso, va a verse realizada por los módulos aquí presentados. Se analizará los módulos presentes en cada capa de la Arquitectura del Sistema, presentando finalmente la Arquitectura Lógica.

### 6.2.1. Interfaz de Usuario

La Vista de Casos de Uso muestra el front-end del sistema. El mismo es generado dinámicamente utilizando tecnología de contenido web dinámico. Desde el punto de vista del back-end se tiene un conjunto de páginas dinámicas generadas a partir de los procesos llevados a cabo por el sistema.

Cada caso de uso indicado en dicha vista requiere, en general, más de una interacción con el usuario. Esta secuencia de páginas es en general variable, dependiendo de las acciones del usuario. La versión expandida de cada caso de uso representa la lógica de dicha interacción.

Esta capa consiste de un módulo por cada caso de uso identificado. Cada módulo contiene la lógica que lleva adelante el caso de uso y un conjunto de páginas dinámicas utilizadas por dicha lógica.

### 6.2.2. Servicios de Sistema

Cada módulo de la capa superior utiliza servicios de esta capa. Se cuenta con una interfaz por caso de uso; esta ofrece los servicios que el módulo que maneja la lógica del caso de uso requiere. Los servicios requeridos por estos casos de uso son cohesivos y por lo tanto pueden ser ofrecidos por el mismo módulo.

### 6.2.3. Servicios de Negocio

Los servicios provistos por estos módulos son los necesarios para poder llevar adelante las operaciones de los módulos de la capa superior.

Los módulos identificados son Manejador de Clientes, Manejador de Usuarios, Manejador de Organizaciones. Cada módulo ofrece una única interfaz con los servicios requeridos.

### 6.2.4. Infraestructura

Los módulos localizados en esta capa son de dos tipos, adaptadores a sistemas existentes, o servicios generales útiles para cualquier aplicación. Estos módulos están disponibles para todos los módulos en las otras capas, y por ello la arquitectura en capas es relajada.

Se cuenta con un Sistema de Mensajería. Este módulo encapsula todo lo relativo con la comunicación con el sistema de envío de mensajes.

En la segunda categoría se encuentra el módulo de Acceso a Datos, el cual encapsula la conexión al motor relacional.

## 7. Vista de Procesos

La Vista de Procesos describe los módulos activos del Subsistema de Vetado; estos son módulos que estarán en ejecución en forma simultánea.

### 7.1. Procesos Distribuidos

#### 7.1.1. Interfaz de Usuario

El Sistema de Vetado es una aplicación basada en el web. Por esta razón cuenta con un grado de distribución a nivel de interfaz de usuario. A nivel de usuario final, corre en su estación de trabajo una aplicación llamada browser. Esta aplicación es la encargada de presentar al usuario la interfaz de la aplicación y de enviar al servidor las acciones que el cliente realiza.

Por el lado del servidor se encuentra otra aplicación general, llamado servidor web. El servidor web en uso es Apache Server. El servidor web recibe las solicitudes del cliente, utilizando el protocolo HTTP, y redirige dichas solicitudes a las páginas dinámicas ubicadas en los módulos de la capa Interfaz de Usuario. El servidor web asocia a cada cliente información de sesión y es él quien se encarga de gestionar la atención a múltiples clientes en forma simultánea.

#### 7.1.2. Servicios de Infraestructura

Los servicios de infraestructura son de dos tipos, aplicaciones legadas que deben ser utilizadas y el propio motor de base de datos. El motor de base de datos corre en forma independiente, atendiendo incluso solicitudes de otras aplicaciones.

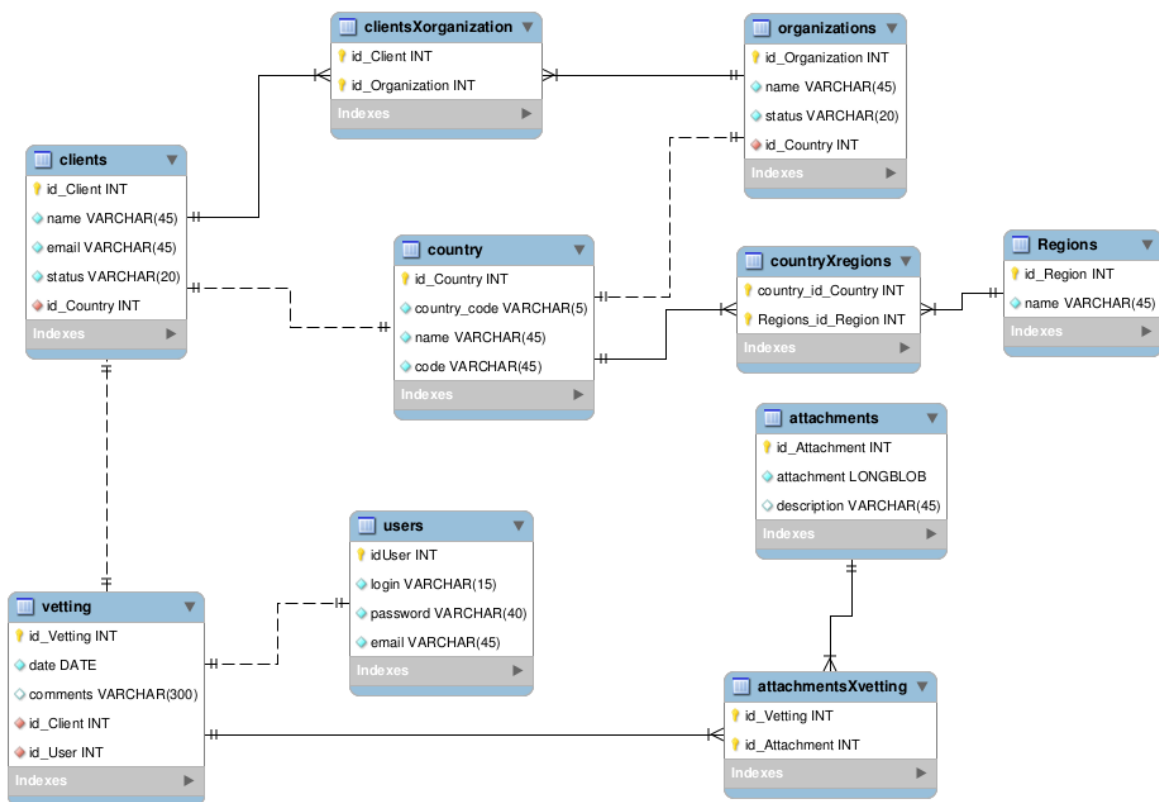


## 8. Vista de Datos

En esta vista se presenta el modelo de datos utilizado, su distribución, y una descripción de los servicios de persistencia utilizados

### 8.1. Modelo de Datos

Se utiliza una única base de datos relacional corriendo sobre el motor de base de datos MySQL. La siguiente figura presenta el modelo de datos.



### 8.2. Distribución

No hay distribución a nivel de datos; todas las tablas radican en la misma base de datos

### 8.3. Servicios de Persistencia

Los servicios de persistencia pueden separarse en dos niveles, los cuales serán mencionados a continuación.

En el nivel inferior se encuentra el motor de base de datos relacional, que para el sistema actual se utiliza el producto MySQL Server

Por encima de éste se ubica la tecnología provista por el framework CakePHP. Esta tecnología brinda servicios de alto nivel, e independientes del producto, para acceder a fuentes de datos.

## 9. Vista de Deployment

La vista de deployment presenta la infraestructura necesaria para instalar el Sistema de Vetado

### 9.1. Tecnología requerida

Las estaciones de trabajo de los clientes (Cliente) deben contar con un browser, con la habilitación de ejecución de JavaScript.