Aplicación Web Bufete de Abogados

Versión 1.0

Revisión Histórica

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Versión** | **Descripción** | **Autor** |
| 12/09/15 | 1.0 | Descripción, alcance y desarrollo de otros puntos de este documento. | Andrés Felipe Perdomo F  Juan José Perdomo F  William David Mesa |
| 13/09/15 | 1.0 | Correcciones de puntos de la plantilla. | Andrés Felipe Perdomo F  Juan José Perdomo F  William David Mesa |
| 13/09/15 | 1.0 | Revisión de la plantilla. | Andrés Felipe Perdomo F  Juan José Perdomo F  William David Mesa |
| 22/09/2015 | 2.0 | Correcciones en los casos de uso, en el diagrama de componentes y en el diagrama de despliegue. | Andrés Felipe Perdomo F  Juan José Perdomo F  William David Mesa |

Tabla de Contenidos

1. Introducción 4

1.1 Propósito 4

1.2 Alcance 4

1.3 Definiciones, Acronimos, and Abreviaturas 4

Véase documento Glosario 4

1.4 Referencias 4

1.5 Vista general 4

2. Representación Arquitectónica 5

3. Metas arquitectónicas y restricciones 6

3.1 Metas 6

3.2 Restricciones 6

4. Vista casos de uso 7

5. Vista lógica 9

5.1 Vista general 9

5.2 Use-Case Realizations 10

6. Vista de procesos 10

7. Vista de despliegue 10

8. Vista de implementación 12

8.1 Niveles o Capas 12

8.1.1 Capa GUI 12

8.1.2 Capa Lógica 12

8.1.3 Capa Base de Datos 12

8.2 Vista general 12

9. Tamaño y rendimiento 13

10. Calidad 14

10.1.1 Seguridad 14

10.1.2 Disponibilidad 15

10.1.3 Mantenibilidad 15

10.1.4 Usabilidad 16

10.1.5 Respaldo 16

Documento arquitectura software

# Introducción

Uno de los aspectos más relevantes a tener en cuenta cuando se construye software es la definición y desarrollo de la arquitectura del software, debido a que esta nos permite representar la estructura del sistema y establecer la forma de comunicación entre las personas involucradas en el desarrollo del proyecto, además de proporcionarnos herramientas para realizar el análisis y la toma de decisiones de las mejores propiedades, tecnologías y herramientas para la construcción del sistema.

En este documento se encontrara la especificación del diseño, las tecnologías y herramientas que son necesarias para la arquitectura del sistema, las vistas a nivel arquitectónico, las metas y restricciones arquitectónicas y la especificación de los atributos de calidad con los que contara el sistema.

## Propósito

Proveer una visión general arquitectónica del sistema, usando un número de vistas arquitectónicas diferentes para representar diferentes aspectos del sistema. Su intención es capturar y transmitir las decisiones arquitectónicas significativas hechas sobre el sistema.

## Alcance

El sistema “aplicación web para un bufete de abogados” tendrá dentro de su alcance, todos los procesos relacionados con la gestión y administración de los casos judiciales, en primer lugar los procesos básicos, crear, editar, eliminar, almacenar y listar los casos judiciales, en segundo lugar procesos de negocio como las notificaciones al cliente, la consulta del estado del caso tanto para el abogado como para el cliente, donde se le brindara información al cliente del proceso en el que se encuentra su caso, el tiempo que se demora e información detallada del caso, esto se realizara a través de un código que será entregado de manera confidencial al cliente con el cual podrá consultar toda la información relacionada a su caso judicial, además los usuarios podrán hacerle preguntas a los abogados por medio de un chat; así que para lograr una correcta construcción este sistema es importante y prioritario identificar la arquitectura de software que se va a implementar para un buen desarrollo del proyecto.

## Definiciones, Acronimos, and Abreviaturas

Ver Glosario.docx.

## Referencias

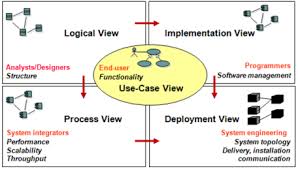
* Documento de especificación de realización de casos de uso

## Vista general

En este documento se encuentra la descripción de la arquitectura, como también cada una de las vistas implementadas y sus descripciones; se especificaran los requisitos funcionales y los atributos de calidad que va a manejar el sistema de información, como también las tecnologías y las herramientas necesarias que se van a implementar para llegar al respectivo desarrollo.

# Representación Arquitectónica

Utilizando el Modelo “4+1” (Kruchten, 1995) se expondrá la arquitectura del sistema “aplicación web para un bufete de abogados”. Este modelo consiste en una seria de vistas (Casos de Uso, Lógica, Procesos, Implementación y Despliegue) las cuales permiten a cualquier Stakeholder encontrar lo que necesita en la Arquitectura. Estas vistas son plasmadas a través de diversos tipos de diagramas de UML (Unified Modeling Language).



**Figura 1.** Las vista del Modelo “4+1” definida en IEEE

1. **Vista Lógica:**

La vista lógica describe el modelo de objetos del diseño, tal como diagramas de entidad y relación, aquí se encuentra el diagrama de clases.

2. **Vista de procesos:**

La vista de procesos describe los aspectos de concurrencia y sincronización del diseño, aquí se encuentran los diagramas de secuencia y de comunicación.

3. **Vista de implementación**

La vista de implementación hace referencia a los componentes de Software, describe las capas y subsistemas de la aplicación, aquí se encuentran los diagramas de componentes.

4. **Vista de despliegue**

La vista de despliegue describe el mapeo o ubicación de los componentes Hardware, y detalla los aspectos de distribución del Sistema, aquí se utiliza el Diagrama de Despliegue.

**5. Vista de casos de uso**

La vista de casos de uso describe el conjunto de Casos de Uso más significativos y de funcionalidad crítica, aquí se encuentra el diagrama de Casos de Uso.

*\*En cada vista se indicara posteriormente los patrones que se utilizaran para su desarrollo.*

# Metas arquitectónicas y restricciones

## Metas

1. Realizar una aplicación cómoda de manejar, de fácil uso, compresión e interacción para los usuarios y así cumplir con el atributo de calidad “usabilidad”.
2. Permitir que dependiendo del perfil del usuario se deshabilitarán opciones de manejo del Sistema para proteger información confidencial.
3. Garantizar el cumplimiento de la implementación de cada uno de los casos de uso mencionados en el punto.
4. Brindar privacidad de la información de los casos jurídicos del cliente, utilizando un método o estrategia de carácter confidencial con el cliente.
5. Brindar una visión general de la aplicación web a través de estas las vistas del modelo 4+1, que permitan un entendimiento más claro de la arquitectura de la aplicación, explicando las interrelaciones y las funcionalidades entre los componentes que lo conforman.
6. Lograr el mayor porcentaje posible en la digitalización de los contenidos de los casos jurídicos manejados por el bufete de abogados, por medio del uso de la información digital almacenada en la aplicación web.

## Restricciones

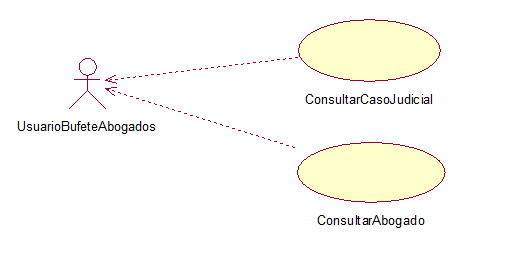
* La aplicación web deberá contar con una disponibilidad como mínimo del 85% del tiempo en un día, esto porque en horarios nocturnos no será muy utilizada la aplicación por los administradores, en este caso los abogados, por lo que en horarios nocturnos no se realizara ninguna actualización del contenido o se modificara el estado de un caso, a no ser que suceda algo excepcional.
* En cuanto a la concurrencia, la aplicación web deberá soportar un máximo de 150 usuarios conectados al mismo tiempo en el día, esto debido a que los casos jurídicos atendidos solo serán nacionales y el número de clientes no es muy grande debido a que el bufete de abogados es pequeño, por lo que la aplicación no tendrá mucha concurrencia
* Las computadoras que serán utilizadas por el bufete de abogados, para el uso de la aplicación, deberán contar con el servicio de internet, de no ser así todos los cambios realizados se perderán.
* La base de datos del sistema deberá ser implementada en MySQL. Debido a su velocidad para realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
* Para el cliente acceder a la información de un caso jurídico, este debe de obtener previamente el código correspondiente al caso buscado.
* Para que alguno de los abogados del bufete pueda acceder como administrador de la aplicación web, y realizar cambios a los contenidos, este deberá loguearse como administrador primero.

# Vista casos de uso

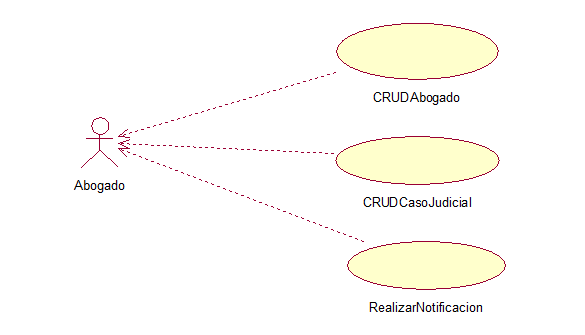
En esta sección se muestran los casos de uso para la arquitectura del software, también se identifican los actores participantes dentro de estos. Dentro de esta sección se identifica y se describe brevemente cada caso de uso que se implementara dentro del desarrollo del software.

|  |  |
| --- | --- |
| CASO DE USO | DESCRIPCIÓN |
| Caso de Uso 1 | CRUD de los casos judiciales de los usuarios. Permite a un Usuario del bufete de abogados registrar, actualizar, eliminar y buscar un caso judicial. |
| Caso de uso 2 | Consultar información de casos judiciales. Permite a un Usuario consultar la información de sus casos judiciales, su estado, descripción y toda la información concerniente respecto a este. |
| Caso de uso 3 | Realizar notificaciones a los usuarios; Permite realizar notificaciones al usuario cuando la información de alguno de sus casos judiciales sea modificada. |
| Caso de uso 4 | CRUD de abogados. Permite a un usuario del bufete de abogados registrar, actualizar, eliminar y buscar información de la información de su perfil. |
| Caso de uso 5 | Consultar información del perfil de un abogado. Permite a un usuario consultar la información del perfil de cualquier abogado del bufete de abogados. |
| Caso de uso 6 | Realizar comunicación entre el usuario y el abogado. Permite realizar una comunicación mediante un chat entre un usuario y algún abogado del bufete de abogados. |

**Tabla1. Casos de Uso y Descripción**

****

**Figura2. Imagen de los casos de uso del Abogado**

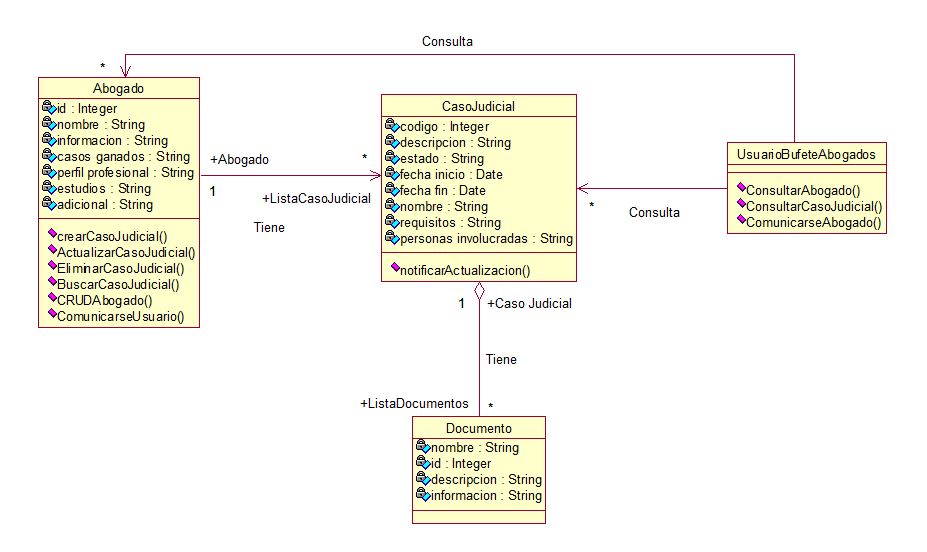


**Figura3. Imagen de los casos de uso del usuario del Bufete de abogados**

# Vista lógica

La vista lógica del sistema está compuesta de cuatro clases principales. Contará con la clase Caso Judicial la cual permitirá realizar el módulo que permitirá hacer CRUD e interactuar con la información de los casos judiciales. En el módulo de los abogados encontramos la clase abogados que permitirá hacer un CRUD de la información de los abogados con sus respectivos atributos. También encontramos la clase Documento, esta permite manejar los documentos asociados a cada caso judicial con su respectiva información y por ultimo encontramos el módulo de los usuarios del bufete de abogados la cual permite realizar las consultas de la información respectiva de los casos judiciales y de los abogados.

## Vista general



**Figura5. Diagrama de clases**

## Use-Case Realizations

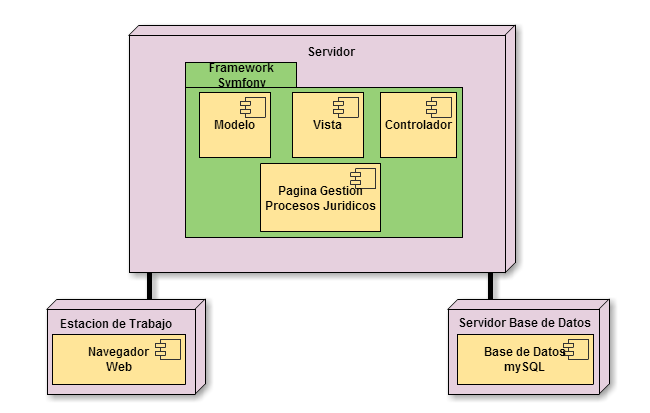
Véase la carpeta de Realización de casos de uso.

# Vista de procesos

Para la visualización de los correspondientes diagramas de secuencia y colaboración de los casos de uso de la aplicación Web del bufete de abogados, véase los documentos contenidos en la carpeta Realización de casos de uso.

# Vista de despliegue

La Figura 6, el Diagrama de Despliegue muestra de manera gráfica los nodos que conforman el sistema, además los describe indicando la ubicación de las tareas en los nodos físicos. Cada nodo representa un recurso de ejecución como una computadora o servidor y se encuentra conectado con otros mediante un enlace de comunicación.



**Figura 6. Vista despliegue**

**Nodo Estación de Trabajo**: Este nodo es para indicar cualquier medio tecnológico por el cual un usuario se puede conectar a la aplicación por medio de un navegador web.

**Nodo Servidor**: Este nodo contiene los componentes que se van a tener en cuenta en la aplicación web para la gestión de los procesos jurídicos en el bufete de los abogados, el modelo vista controlador que es el patrón arquitectónico sobre el que está basado symfony, y en si el mismo Framework. Todas las capas de la aplicación estarán basadas en el MVC y el desarrollo de la aplicación será separado según las capas utilizadas.

**Nodo Servidor Base de datos**: Este nodo contiene la información de la base de datos, que va a ser implementada en MySQL.

# Vista de implementación

La vista de implementación seguirá el patrón de diseño MVC (Model View Controller). Este patrón plantea la separación del problema en tres capas: la capa model, la capa controller y la capa view, utilizando para ello el Framework Symfony.

## Niveles o Capas

### **Capa GUI**

El componente GUI maneja las clases que permiten la interacción del usuario con el sistema.

### **Capa Lógica**

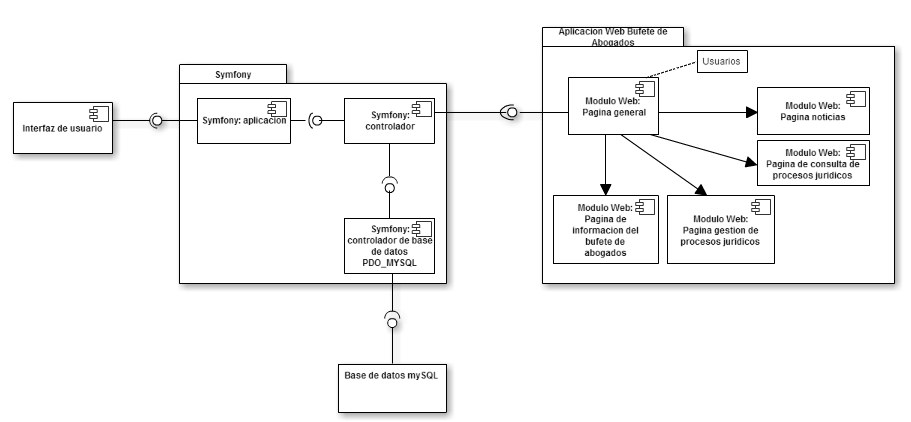
Este componente permite manejar la lógica del negocio de todas las áreas involucradas.

* Administración del Sistema
* Usuarios
* Privilegios
* Comparaciones
* Registro de documentos

### **Capa Base de Datos**

Este componente contiene toda la información que maneja el Sistema, todos los datos.

## Vista general



**Figura 7. Diagrama de componentes**

En el diagrama de componentes se ve reflejada la implementación del Framework Symfony, que utiliza el (MVC) modelo, vista, controlador, nombrado anteriormente. Posteriormente se seleccionara un controller que se encargara de obtener la información de una petición HTTP (Request) y de generar y devolver la respuesta HTTP(Response).

Se deben instalar tanto la extensión PDO como el driver PDO de la base de datos, como en nuestro caso es PDO\_MYSQL debido a que vamos a usar la base de datos mySQL.

# Tamaño y rendimiento

1. La velocidad de ingreso a la aplicación web desde un navegador dependerá del servicio de internet contratado.
2. El espacio en disco necesario para la parte del cliente deberá tener como mínimo 1 GB de espacio libre para su correcto funcionamiento.
3. El espacio en disco necesario del servidor de base de datos deberá ser como mínimo de 10 GB libres en disco para su correcto funcionamiento, debido a los numerosos casos que se encuentran en físico y deberán migrarse a la aplicación web.
4. El sistema Web permitirá el acceso concurrente de varios usuarios, con un estimado de 150 usuarios maximo, de modo que puedan realizar transacciones simultáneas.

# Calidad

### Seguridad

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo de Calidad** | Seguridad | | | |
| **Interés del atributo** | Garantizar la autenticación de los abogados como administradores de la aplicación (nombre de usuario/contraseña cifrada y recuperación segura de la contraseña).  Garantizar la autorización a las funcionalidades de la aplicación haciendo uso de roles, privilegios y estado actual del usuario en el sistema. | | | |
| **Descripción** | Verificar si el sistema puede brindar la privacidad de la información del cliente, si se garantiza el uso de roles y privilegios. | | | |
| **ID Escenario** | Esc1 | | | |
| **Escenario** | El sistema debe de ser altamente seguro cuando usuario desee ingresar en la aplicación | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Usuario identificado correctamente cuando hace uso de la aplicación. |  |
|  |  | **Estímulo** | Operación de modificar información. Cuando el usuario requiera ingresar a realizar determinadas operaciones deberá haberse logueado anteriormente. |  |
|  |  | **Ambiente** | El sistema se encuentra operando de manera normal. |  |
|  |  | **Respuesta** | El sistema mantiene la privacidad de cada caso judicial con la confidencialidad del código que se le asigna a cada caso. |  |
|  |  | **Medida de respuesta** | Las sesiones de los usuarios se encontraran abiertas y para su empleo en la estación de trabajo por lo que otras personas ajenas a la administración no tendrán contacto alguno con estas cuentas. |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

**Tabla 2. Atributo de calidad de seguridad**

### Disponibilidad

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo de Calidad** | Disponibilidad | | | |
| **Interés del atributo** | Contar con una disponibilidad mínima del 85% de las 24 horas del día, los 7 días de la semana. | | | |
| **Descripción** | Verifica que la aplicación pueda ejecutarse constantemente y se encuentre disponible el tiempo necesario para el normal desarrollo de las actividades laborales del juzgado. | | | |
| **ID Escenario** | Esc2 | | | |
| **Escenario** | El sistema debe de estar disponible para el usuario sin generar error alguno. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Sistema externo, o agente extremo |  |
|  |  | **Estímulo** | Mensaje no anticipado |  |
|  |  | **Ambiente** | Operación normal |  |
|  |  | **Respuesta** | Informar al operador y seguir operando. |  |
|  |  | **Medida de respuesta** | sin caídas, disponible y sin errores funcionales. |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

**Tabla 3. Atributo de calidad de Disponibilidad**

### Mantenibilidad

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo de Calidad** | Mantenibilidad | | | |
| **Interés del atributo** | El sistema debe estar en capacidad de incluir nuevas funcionalidades y/o modificar o eliminar funcionalidades existentes durante su puesta en servicio y operación. | | | |
| **Descripción** | Verificar que se puedan incluir nuevas funcionalidades | | | |
| **ID Escenario** | Esc3 | | | |
| **Escenario** | El sistema debe de permitir añadir nuevas funcionalidades, siempre y cuando estas sean aprobadas y no afecten la implementación del proyecto. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Usuario |  |
|  |  | **Estímulo** | Desea agregar funcionalidad |  |
|  |  | **Ambiente** | Etapa de mantenimiento |  |
|  |  | **Respuesta** | Cambio de requerimiento correctamente implementado |  |
|  |  | **Medida de respuesta** | Que no se supere el tiempo acordado con el cliente |  |
|  |  |  |  |  |
| **Importancia** | Media | | | |

**Tabla 4. Atributo de calidad de Mantenibilidad**

### Usabilidad

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Atributo de Calidad** | Usabilidad | | | |
| **Interés del atributo** | Facilitar la navegabilidad, mostrar mensajes informativos y de confirmación como medio de validación de ejecución de la acción por parte del usuario. Cuando se presentan errores por parte del usuario, se mostraran mensajes de alerta que expliquen la falla y la acción correctiva. | | | |
| **Descripción** | Verifica si el sistema es de fácil uso y entendimiento para el usuario | | | |
| **ID Escenario** | Esc4 | | | |
| **Escenario** | El sistema debe de ofrecer fácil y cómoda navegación, un dialogo correcto de las interfaces para la interacción del usuario. | | | |
|  |  |  |  |  |
|  |  | **Fuente del Estimulo** | Usuario |  |
|  |  | **Estímulo** | Maximizar el grado de satisfacción y mejorar la interacción del usuario con las interfaces de la aplicación web |  |
|  |  | **Ambiente** | Ejecución normal |  |
|  |  | **Respuesta** | Mínimo de errores en cuanto a usabilidad y diseño de las vistas |  |
|  |  | **Medida de respuesta** | Sistema integro antes de la ejecución |  |
|  |  |  |  |  |
| **Importancia** | Alta | | | |

**Tabla 5. Atributo de calidad de Usabilidad**