Raya roja

diseño TECNICO de sistema

Cliente: Ingeniero Saulo Pineda

Aplicación: Sistema de Reservas de Hotel

Versión: 1.0

Autor: Equipo Kaizen

Plantilla: 01 DTSW – Diseño Técnico de Sistema v 1.0

Contenido

[1. OBJETIVO 3](#_Toc474614154)

[2. ESQUEMA GENERAL DE DISEÑO 4](#_Toc474614155)

[3. DECISIONES GENERALES DE DISEÑO 5](#_Toc474614156)

[4. VISTA LÓGICA y Fisica 8](#_Toc474614157)

[5. CiESTIONES ESPECÍFICAS DE DISEÑO 8](#_Toc474614158)

[5.1. DISEÑO PARA EL RENDIMIENTO DEL SISTEMAS 8](#_Toc474614159)

[5.2. DISEÑO TÉCNICO INTERFAZ DE USUARIO 9](#_Toc474614160)

[5.3. SEGURIDAD 9](#_Toc474614161)

Control de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| Versión | Responsable | Fecha | Descripción del cambio |
| 1.0 | Grupo Kaizen | 22/02/17 | Elaboración del Documento |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# OBJETIVO

El objetivo de este documento es la realización del diseño técnico de todos los procesos que detallan los casos de uso mencionados en el documento llamado **Especificación funcional detallada de requisitos del sistema.**

Podemos mencionar dos objetivos específicos:

1. Describir el modelo del diseño técnico que se ofrece para desarrollar el proyecto según los procedimientos elaborados.
2. Explicar las relaciones existentes entre los elementos del sistema (base de datos establecidos).

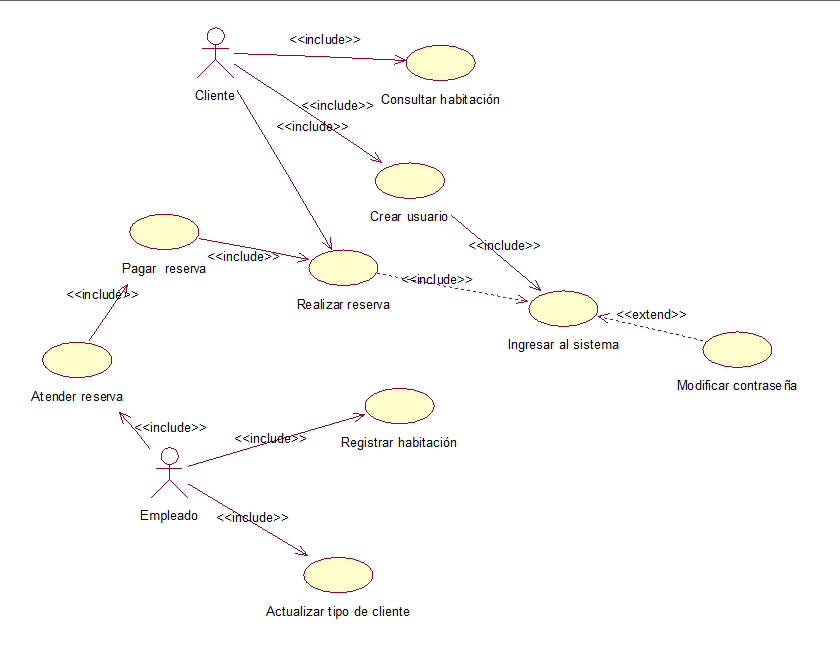
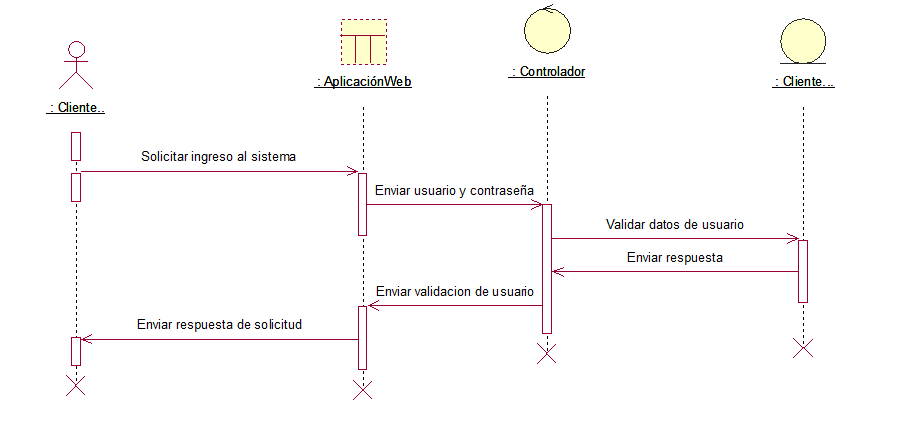
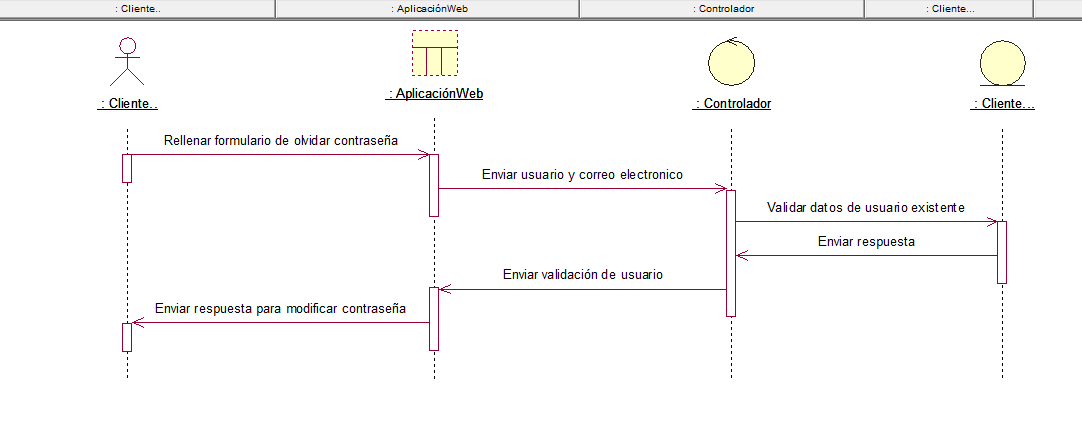
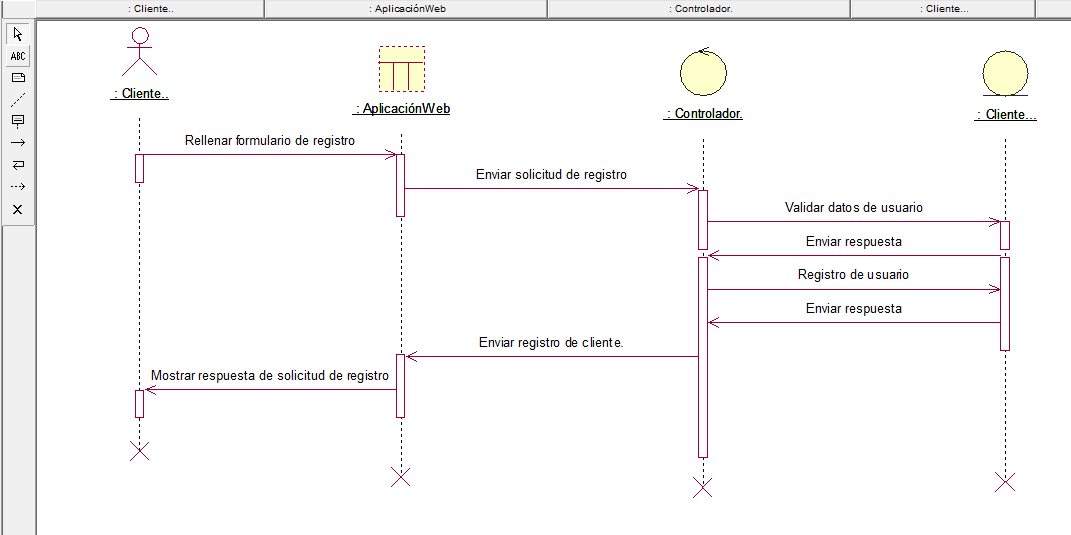
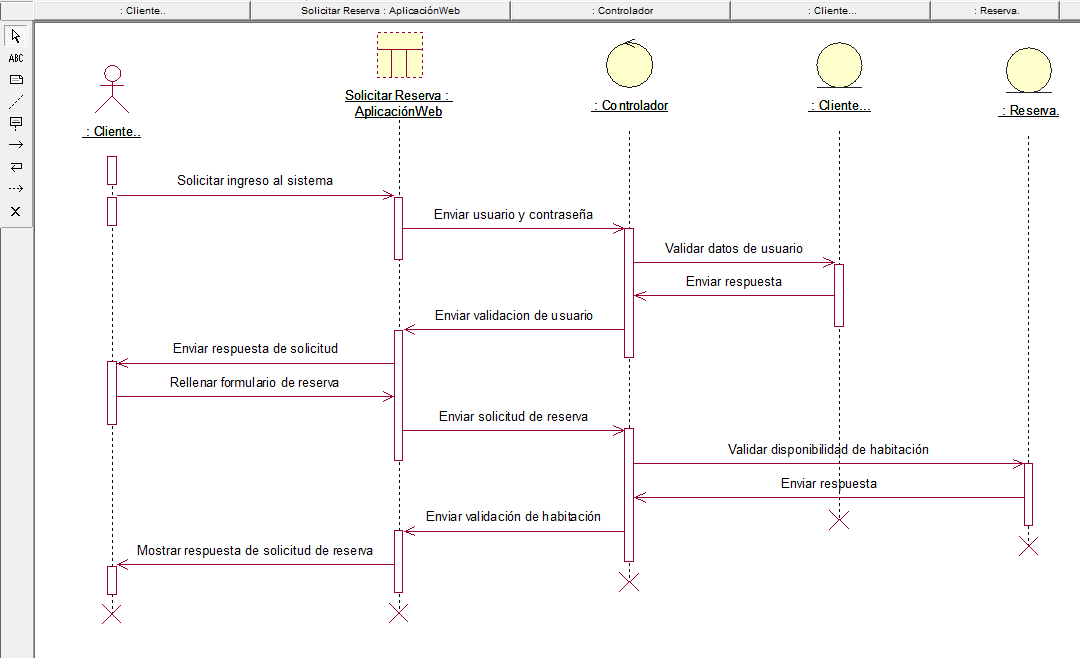
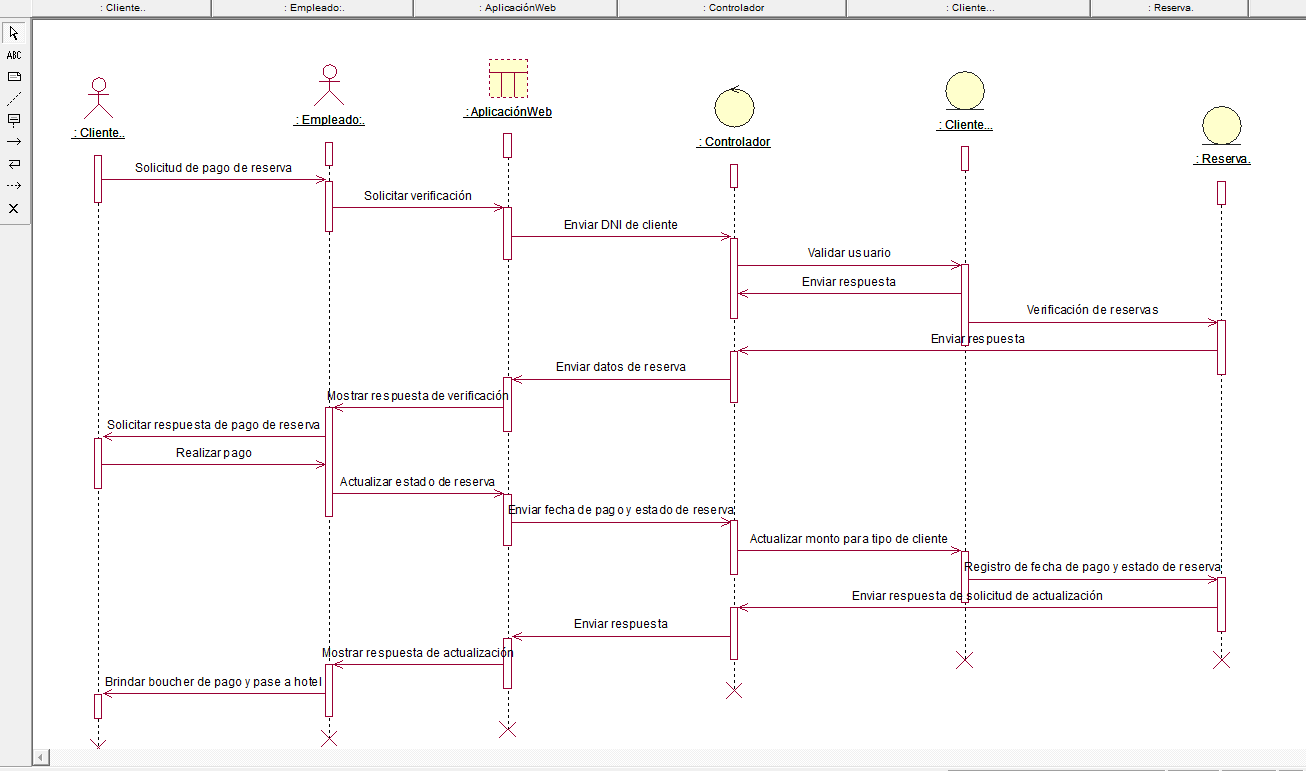
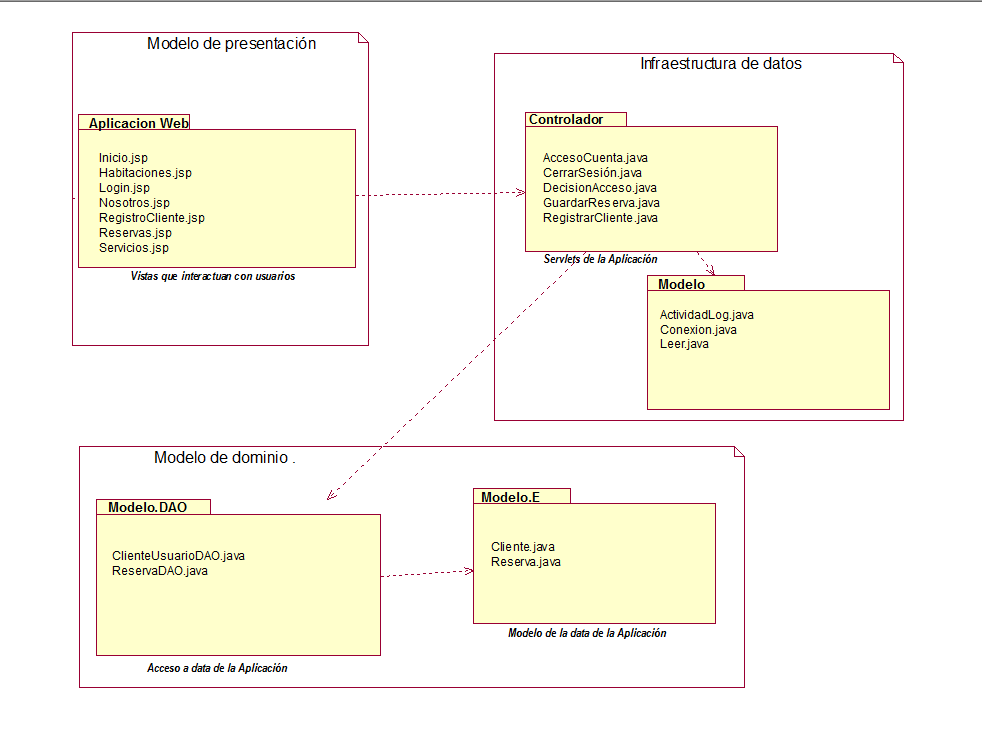
**Consideraciones**

1. Se usará código libre para el desarrollo del proyecto.

**Revisiones del diseño técnico**

1. El equipo de proyecto realizará un diseño documentado que detallara cada proceso que se ha elaborado para realizar este proyecto.
2. En la versión final contrastaremos lo cambios realizado con el documento inicial con el objetivo de ver el valor ganado del proyecto.

# ESQUEMA GENERAL DE DISEÑO

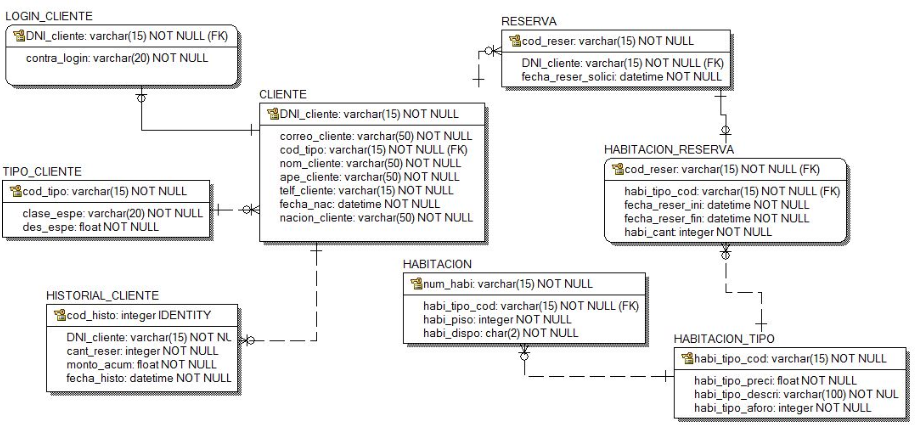
      *.*

|  |  |  |
| --- | --- | --- |
| **Package** | **Descripción** | **Requisitos Implementados** |
| Aplicación Web | Contiene las interfaces del sistema que van a interactuar con el cliente. | * Inicio.jsp * Reservas.jsp * Login.jsp * Nosotros.jsp * Habitaciones.jsp * Login.jsp * Registro.jsp |
| Controlador | Contiene los Servlets que recibirán los datos para validar la información con la base de datos. | * AcesoCuenta.java * CerrarSesión.java * DecisiónAcceso.java * GuardarReserva.java * RegistrarCliente.java |
| Modelo | Contiene las clases que permitirán la conexión segura con la base de datos. | * ActividadLog.java * Conexión.java * Leer.java |
| Repositorio | Contiene los procedimientos predefinidos que se van consultar en la base de datos de una manera segura. | * Usp\_actualizarReserva * Usp\_actualizartipocliente * Usp\_disponibilidadportipo * Usp\_frecuenciaAñoMesporDNI * Usp\_frecuenciaAñoporDNI * Usp\_getclavecliente * Usp\_getDNIcliente * Usp\_montoAñoMesporDNI * Usp\_montoAñoporDNI * Usp\_nuevareserva * Usp\_nuevocliente * Usp\_tamañohistorial |
| Entidades | Contiene las clases que se han establecida para el desarrollo del sistema. | * Cliente * Reserva |

# DECISIONES GENERALES DE DISEÑO

|  |  |
| --- | --- |
| **Información general del aplicativo** | |
| Nombre administrativo de la aplicación | Servicio de reservas de Hotel |
| Nombre técnico de la aplicación | Sistema Web Kaizen |
| Espacio de nombres | Kaizen.SRH |
| Tipo de aplicativo | Aplicación Web |
| **Plataforma** | |
| Escenario | Portal |
| Estilo arquitectónico | 3 capas |
| Tecnologías y versiones | * JDK 8.091 * SqlServer 2008 R2, * J-Query 1.6 |
| * Lenguaje de programación | * JavaScript * CSS 3 * HTML 5 |
| Tecnologías específicas aplicadas en cada capa | |  |  |  | | --- | --- | --- | | **Capas** | **Proyectos** | **Tecnologías** | | Presentación | Aplicación Web | Tecnologías que se usan:   * XML * CSS3 * HTML5 * JAVASCRIPT * JQUERY 1.6 | | Dominio | Modelo.E | Librería de clases que usaremos en la aplicación.  Tecnologías que se usan:   * JDK 8.091 | | Modelo.DAO | Librería de acceso a clases que usaremos en la aplicación para acceder a la base de datos.  Tecnologías que se usan:   * JDK 8.091 | | Infraestructura | Controlador | Librería de clases que nos ayudaran a transferir datos en el sistema.  Tecnologías que se usan:   * JDK 8.091 | | Modelo | Librería de clases que nos permitirán la conexión con la base de datos.  Tecnologías que se usan:   * JDK 8.091 | |
| Tecnologías específicas para la interoperatividad con otros sistemas | No aplicado |
| Componentes corporativos utilizados | Hojas de estilo corporativas del escenario correspondiente. |
| Componentes de terceros utilizados | * Jquery 1.6 * Node.js |
| **Aspectos horizontales** |  |
| Gestión de excepciones | PORTAL |
| Gestión de trazas | PORTAL |
| Seguridad (Autenticación) | n/a |
| Seguridad (Autorización) | Basada en roles definidos en BD “KAIZENHOTEL” |
| Gestión de la configuración | Basado en configuración almacenada en BD |
| Gestión del multi-idiomado | Basado en configuración almacenada en BD |
| Auditoría de datos | n/a |
| Ínstrumentación/logging | n/a |
| **Decisiones relativas a la metodología** | |
| Nivel mínimo de Calidad técnica exigido | Debe realizarse las pruebas unitarias |
| Tecnología de control de versiones | Herramienta para la gestión de control de versiones GitHub. |
| Puesto de desarrollo | SQL Server 2008 R2 |
| **Otras decisiones generales** | |
| N/A | |

# VISTA LÓGICA y Fisica



Se describen aspectos sobre el diseño de la aplicación.

# CuESTIONES ESPECÍFICAS DE DISEÑO

## DISEÑO PARA EL RENDIMIENTO DEL SISTEMAS

**Optimizaciones en el acceso a datos**

* Uso de patrón DAO para tener una mayor eficiencia en el requerimiento de los datos.
* Uso de patrón MVC para la separación por capas de la aplicación web.

**Algoritmos y Recursos**

* Uso de patrones de diseño para tener un orden y poder optimizar el código.
* Uso de código abierto para agilizar el desarrollo de la aplicación en las galerías dinámicas de imágenes.
* Uso de frameworks para el envío de datos a los servlets.

**Carga de páginas HTML**

* Visualizaciones de prueba con la ayuda de Apache Tomcat.
* Pasar parámetros por Ajax
* Simplificación de archivos css

## DISEÑO TÉCNICO INTERFAZ DE USUARIO

* Estilos CSS y tecnología Flexbox para un diseño adaptable.
* Uso de frameworks como J-Query para un interfaz amigable con el usuario.

## SEGURIDAD

**Roles involucrados en el aplicativo**

1. Administrador de la base de datos

**Mapeos a permisos**

N/A

**Mecanismo de autenticación y de autorización**

Autenticación

* + Integrada con Windows
  + Servidores de prueba