|  |  |  |  |
| --- | --- | --- | --- |
| **Revisión** | **Fecha** | **Modificación Realizada** | **Realizado Por:** |
| 1 | 25 de Junio del 2018 | Versión Inicial | Jael Armas M. |

**TABLA DE CONTENIDO**

**GLOSARIO DE TÉRMINOS .............................................................................................................. 1-4**

1. **INTRODUCCIÓN ......................................................................................................................... 1-5**
   1. OBJETO DEL DOCUMENTO. ......................................................................................................... 1-5
   2. EQUIPO DE TRABAJO. ................................................................................................................. 1-5
   3. MATERIAL REVISADO. ................................................................................................................. 1-6
2. **ARQUITECTURA ........................................................................................................................ 2-6**
   1. ARQUITECTURA LÓGICA ......................................................................................................... 2-6

# 2.2 Diseño Guiado por Dominio – ASP .NET MVC..................................................................... 2-7

*2.2.1* *Capa de Interfaz de Usuario: .......................................................................................... 2-7*

*2.2.2* *Capa de Aplicación: ........................................................................................................ 2-7*

*2.2.3* *Capa de Dominio ............................................................................................................. 2-7*

*2.2.4* *Capa de Infraestructura: ................................................................................................. 2-8*

2.3 ARQUITECTURA FÍSICA ........................................................................................................... 2-9

1. **MEJORES PRÁCTICAS ........................................................................................................... 3-10**
   1. GIT ......................................................................................................................................... 3-11
2. **CALIDAD .................................................................................................................................... 4-12**
3. **DOCUMENTACIÓN .................................................................................................................. 5-12**
4. **ANEXOS ...................................................................................................................................... 6-13**

6.1 MODELO ENTIDAD RELACIÓN ................................................................................................. 6-13

# 6.2 DIAGRAMA DE CLASES .......................................................................................................... 6-14

# **GLOSARIO DE TÉRMINOS**

Dentro del contexto del presente documento se definen los siguientes términos:

**ASP .NET:** Es un entorno para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores y diseñadores para construir sitios web dinámicos, aplicaciones web y servicios web XML.

**.NET Framework:** Es un framework de Microsoft que hace un énfasis en la transparencia de redes, con independencia de plataforma de hardware y que permite un rápido desarrollo de aplicaciones.

**ADO .NET Entity Framework:** Es un conjunto de API de acceso a datos para el Microsoft .NET Framework Una entidad del Entity Framework es un objeto que tiene una clave representando la clave primaria de una entidad lógica de datastore. Un modelo conceptual Entity Data Model (modelo Entidad-Relación) es mapeado a un modelo de esquema de datastore.

**MVC:** Modelo Vista Controlador, es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

**ASP .NET Razor:** Es una sintaxis de programación ASP.NET utilizada para crear páginas web dinámicas con los lenguajes de programación C # o Visual Basic .NET, Razor es un motor de visualización de sintaxis simple y se lanzó como parte de MVC 3 y del conjunto de herramientas WebMatrix.

**WebMatrix:** Es un generador de sitios web gratuito y conectado a la nube y editor de HTML para Windows, orientado al desarrollo web. WebMatrix permite a los desarrolladores crear sitios web utilizando plantillas incorporadas o populares aplicaciones de código abierto, con soporte completo para ASP .NET, PHP, Node.js y HTML5.

## 1. INTRODUCCIÓN

**1.1 Objeto del Documento.**

Presenta la descripción de cada uno de los componentes de la arquitectura para el desarrollo de la aplicación web Hotelli.

Una Arquitectura de [Software,](http://es.wikipedia.org/wiki/Software) consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del [software](http://es.wikipedia.org/wiki/Software) de un [sistema de información.](http://es.wikipedia.org/wiki/Sistema_de_informaci%C3%B3n)

La arquitectura de la aplicación permite definir la organización interna de la misma estableciendo una funcionalidad por capas y bien definida. Permiten tener mejor control de la ubicación de las reglas de negocio y por lo tanto facilita las tareas de mantenimiento de las mismas.

**1.2 Equipo de Trabajo.**

Las personas que directa o indirectamente intervienen son las siguientes:

|  |  |
| --- | --- |
| **Nombre** | **Cargo** |
| Jael Armas M. | Analista de Infraestructura |
| Jael Armas M. | Directora de Gestión y Registro (Administradora del Contrato) |
| Jael Armas M. | Coordinador de Infraestructura y Seguridad Informática |
| Jael Armas M. | Asesora de Coordinación de Infraestructura y Seguridad Informática |
| Jael Armas M. | Directora de Tecnología y Desarrollo |

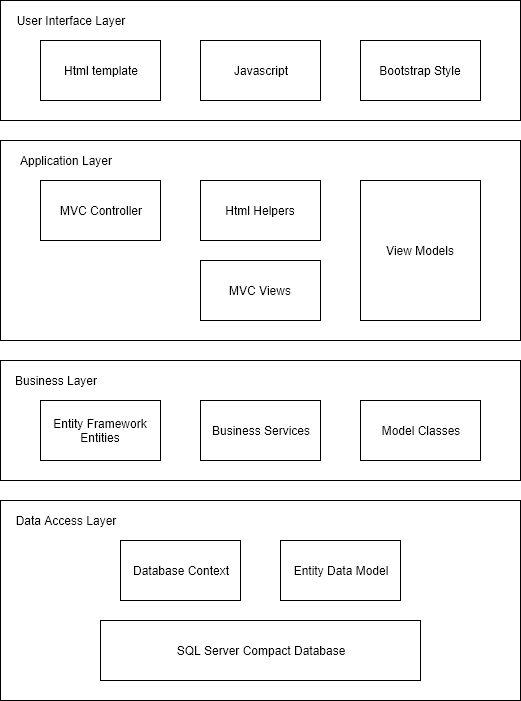
**1.3 Material Revisado.**

* Estándares y herramientas para el desarrollo de aplicaciones ASP .NET.

## 2. ARQUITECTURA

### 2.1 Arquitectura Lógica

El patrón de arquitectura elegido para la creación de la aplicación web Hotelli es ASP .NET MVC, este modelo de arquitectura se encuentra basado en la especificación de Microsoft y en patrones de diseño ampliamente usados, así como estándares abiertos.



Para implementar la arquitectura ASP .NET MVC se usará el servidor de aplicaciones IIS 10 de Windows Server 2016 que está certificado para soportar ASP .NET MVC.

### 2.2 Diseño Guiado por Dominio – ASP .NET MVC

**2.2.1 Capa de Interfaz de Usuario:**

La capa de interfaz de usuario o capa de presentación, se utilizará la especificación de ASP .NET Razor, de WebMatrix de Microsoft, permitiendo utilizar un flujo de trabajo de construcción HTML de manera fácil, rápida y limpia.

**2.2.2 Capa de Aplicación:**

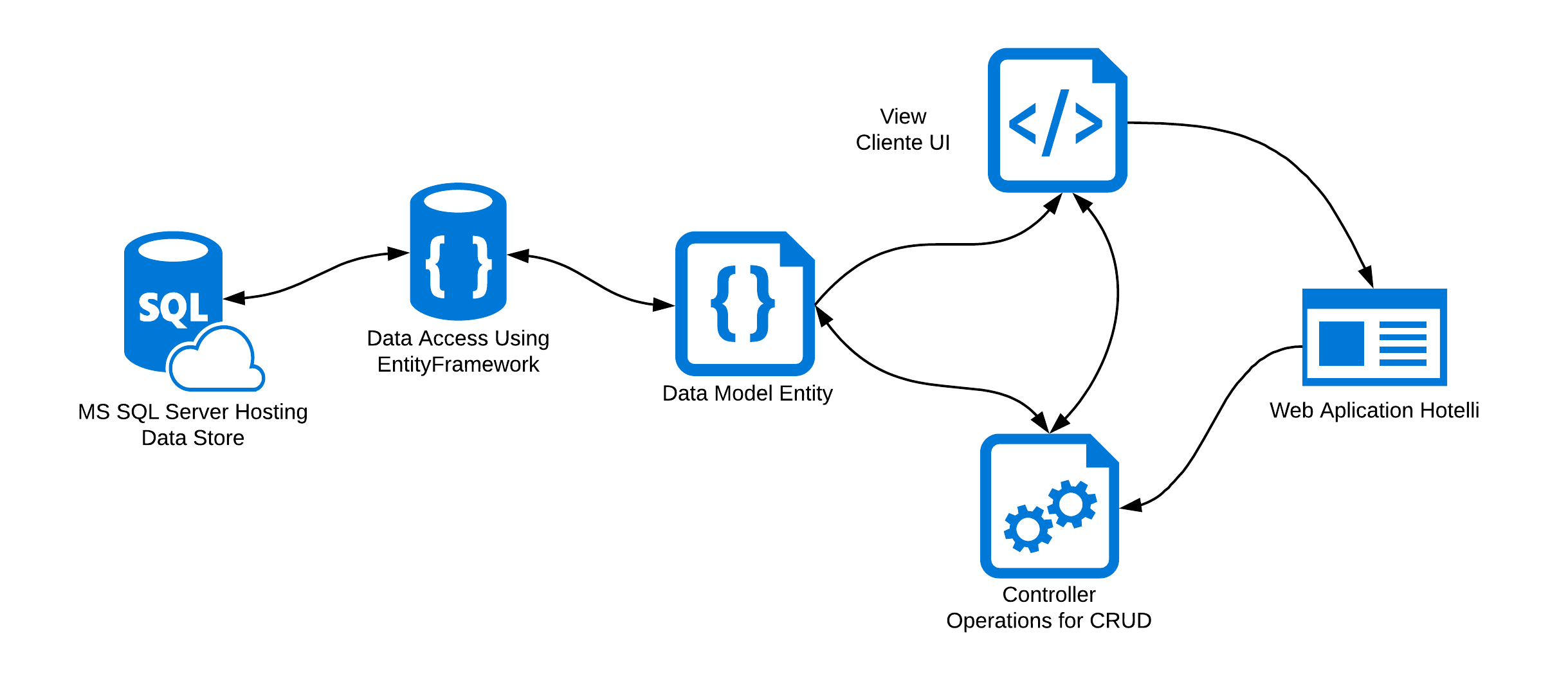
La capa de aplicación incluye principalmente servicios de aplicaciones que usa la capa de negocios para realizar las funcionalidades solicitadas de la aplicación. Utiliza objetos de transferencia de datos para obtener datos y devolver datos a la capa de presentación o servicio distribuido. También puede tratar con autorización, almacenamiento en caché, asignación de objetos, la sesión y así sucesivamente.

**2.2.3 Capa de Negocios:**

La capa de dominio o lógica de negocio contiene la implementación del ORM usando Entity Framework. También contiene las validaciones y reglas propias de negocio.

**2.2.4 Capa de Acceso a Datos:**

La capa de infraestructura contiene los mecanismos necesarios para interactuar con las bases de datos. Se utiliza ASP Entiy Framework como herramienta de mapeo de atributos entre una [base de datos](http://es.wikipedia.org/wiki/Base_de_datos) relacional tradicional y el modelo de [objetos](http://es.wikipedia.org/wiki/Objetos_(programaci%C3%B3n_orientada_a_objetos)) de la aplicación.



### 2.3 Arquitectura Física

A nivel físico el sistema cuenta con un esquema distribuido entre múltiples plataformas:

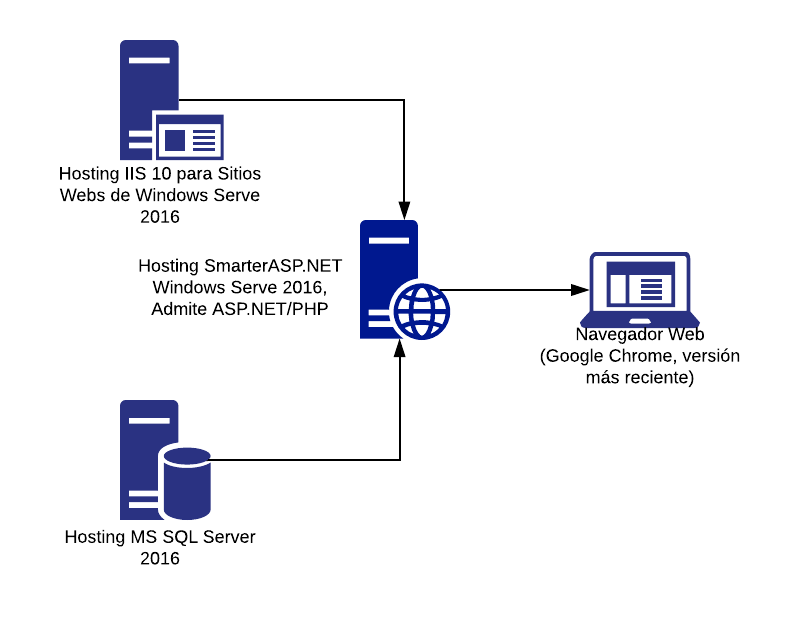
**Windows Server 2016**.- Es una marca que abarca una línea de productos del servidor de Microsoft Corporation y consiste en un sistema operativo diseñado para servidores de Microsoft y una gama de tipos de productos dirigidos al mercado más amplio de negocios.

**IIS 10**.- Internet Information Services, es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows. Este servicio convierte a un PC en un servidor web para Internet o una intranet, es decir que en los ordenadores que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente.

**MS SQL Server**.- Microsoft SQL Server es un sistema de manejo de bases de datos del modelo relacional, desarrollado por la empresa Microsoft. El lenguaje de desarrollo utilizado es Transact-SQL (TSQL), una implementación del estándar ANSI del lenguaje SQL, utilizado para manipular y recuperar datos (DML), crear tablas y definir relaciones entre ellas (DDL).

**Smarter ASP .NET**.- Es uno de los mejores ASP.NET Hosting con grandes fortalezas en servicios de alojamiento ASP.NET. Hasta ahora, ha ayudado exitosamente a los clientes a construir 170,000 sitios web repartidos por todo el mundo.

**Google Chrome**.- Es un navegador web de software privativo o código cerrado6​7​ desarrollado por Google, aunque derivado de proyectos de código abierto.​ Está disponible gratuitamente. El nombre del navegador deriva del término en inglés usado para el marco de la interfaz gráfica de usuario.



## 3. MEJORES PRÁCTICAS

### 3.1 Git

Herramienta para el control de versiones de código que permite llevar el control y auditoría de los cambios que realiza cada desarrollador, y cambiar fácilmente de versión para contar siempre con código estable listo para su uso en producción.

## 4. CALIDAD

Las consideraciones arquitectónicas y de diseño mostradas en este documento contribuyen a elevar la calidad del servicio de la siguiente manera:

***Escalabilidad y Confiabilidad:*** apoyada por el diseño modular y la separación en componentes, esta permite distribuir los componentes entre varios servidores en caso de ser requerido.

***Mantenimiento:*** Los patrones arquitectónicos que se usan permiten que la solución sea lo suficientemente flexible separando claramente cada parte funcional de la solución para soportar cambios en una manera consistente y sencilla.

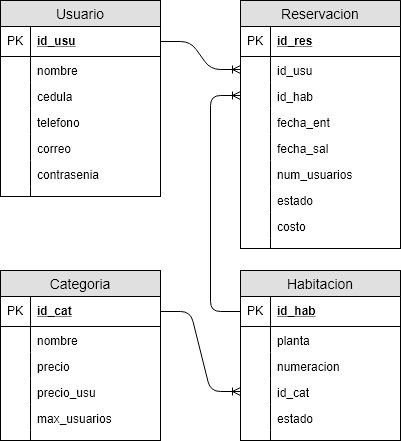
## 5. DOCUMENTACIÓN

Se listan los documentos a entregar junto con el proyecto:

* Diagrama de Arquitectura.
* Diagrama Físico de Base de Datos.

## 6. Anexos

### 6.1 Modelo Entidad – Relación



### 6.2 Diagrama de Clases

