|  |  |
| --- | --- |
| NOMBRE DEL PROYECTO: | Mis Primeras Letras |

# CONTROL DE CAMBIOS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| VERSIÓN | FECHA  AAAA/MM/DD | DESCRIPCIÓN DEL CAMBIO | ELABORADO POR | AUTORIZADO POR |
| 1.0 | 2020/05/20 | Documento Inicial | Julian Bermudez  Juan Diaz  Jorge Escobar  Fabian Lasso |  |

1. **OBJETIVO**

OBJETIVO GENERAL

Diseñar el Sistema de Información Mis Primeras Letras para un jardín infantil, el cual permita la gestión de matrícula de estudiantes, gestión de pagos, gestión de horarios y salones. Teniendo en cuenta la arquitectura propuesta y respectando los estándares de calidad del software.

OBJETIVOS ESPECÍFICOS

* Usar el modelo de “vistas 4 + 1”, para representar las características del sistema.
* Realizar el diseño del sistema siguiendo el paradigma de desarrollo orientado a objetos- POO.
* Generar insumos y/o artefactos relevantes para la implementación de la aplicación, como diagramas de despliegue y de componentes del sistema, con el fin de obtener una visión global del sistema tanto en software como en hardware.
* Identificar los componentes e interfaces de la estructura del sistema.
* Mostrar la interacción y comportamiento entre los diferentes componentes del sistema.
* Modelar la estructura estática del sistema mediante diagramas de clases, a partir de los casos de uso definidos.
* Modelar la estructura dinámica del sistema a través de diagramas de secuencia, generados a partir de los procesos lógicos del sistema.
* Registrar a los usuarios del sistema con sus respetivos datos, de tal manera que el administrador pueda cambiar los roles de ser necesario.
* Registrar la matrícula de estudiantes tanto nuevos como antiguos teniendo en cuenta las validaciones del grado a cursar como el pago.
* Gestionar el horario de cada uno de los grupos de tal manera que no se crucen los horarios de los profesores ni los salones.
* Generar reportes de estudiantes, salones, y pagos de matrícula. \*

1. **LIMITACIONES DE LA ARQUITECTURA**

Las limitantes que podrían afectar el diseño y/o desarrollo del Sistema Mis Primeras Letras son:

* Debido a que se utilizó en SQL server de manera local, la aplicación podría no soportar un gran volumen de datos.
* Los controles de cambio que afecten directamente a los procesos ´principales del sistema afectando así las tablas principales de la base de datos.
* Cambios de diseño por cambio de tecnología o framework usado en un comienzo y cambiado durante el desarrollo del proyecto.

1. **VISTAS DE ARQUITECTURA**

Teniendo en cuenta el modelo conceptual se realiza una abstracción del diseño arquitectónico del Sistema de Información Mis Primeras Letras basado en el modelo de “vistas 4 + 1”, donde se parte de la vista central de escenarios representada con el diagrama de casos de uso definidos en la etapa de análisis para llegar a las cuatro dimensiones de arquitectura que representan la solución.



Figura No. 1 Vista 4 + 1

* 1. **VISTA DE ALTO NIVEL**

La vista de alto nivel propone el uso de escenarios; los cuales son instancias de los casos de uso, que relacionan las cuatro vistas entre sí, permitiendo obtener una perspectiva general del sistema con el fin de validar la arquitectura propuesta. Estos surgen de la especificación de requerimientos de la etapa de análisis.

* 1. **VISTA LÓGICA**

En esta vista se representa la funcionalidad del sistema, donde se identifican las funciones principales y servicios que ofrece la solución. En esta vista se proporciona la base para comprender la estructura y la organización del sistema Mis Primeras Letras a un alto nivel.

Cada componente y/o módulo de Mis Primeras Letras se desarrolla de manera independiente; sin embargo, la arquitectura de software definida está dada por el uso del patrón de arquitectura MVC (Modelo – Vista – Controlador) que facilita la evolución de la lógica de negocio y la interfaz de usuario por separado, combinado con el patrón DAO (Data Access Object) para la configuración del acceso al nivel de datos. Lo anterior debido al carácter web de la aplicación.

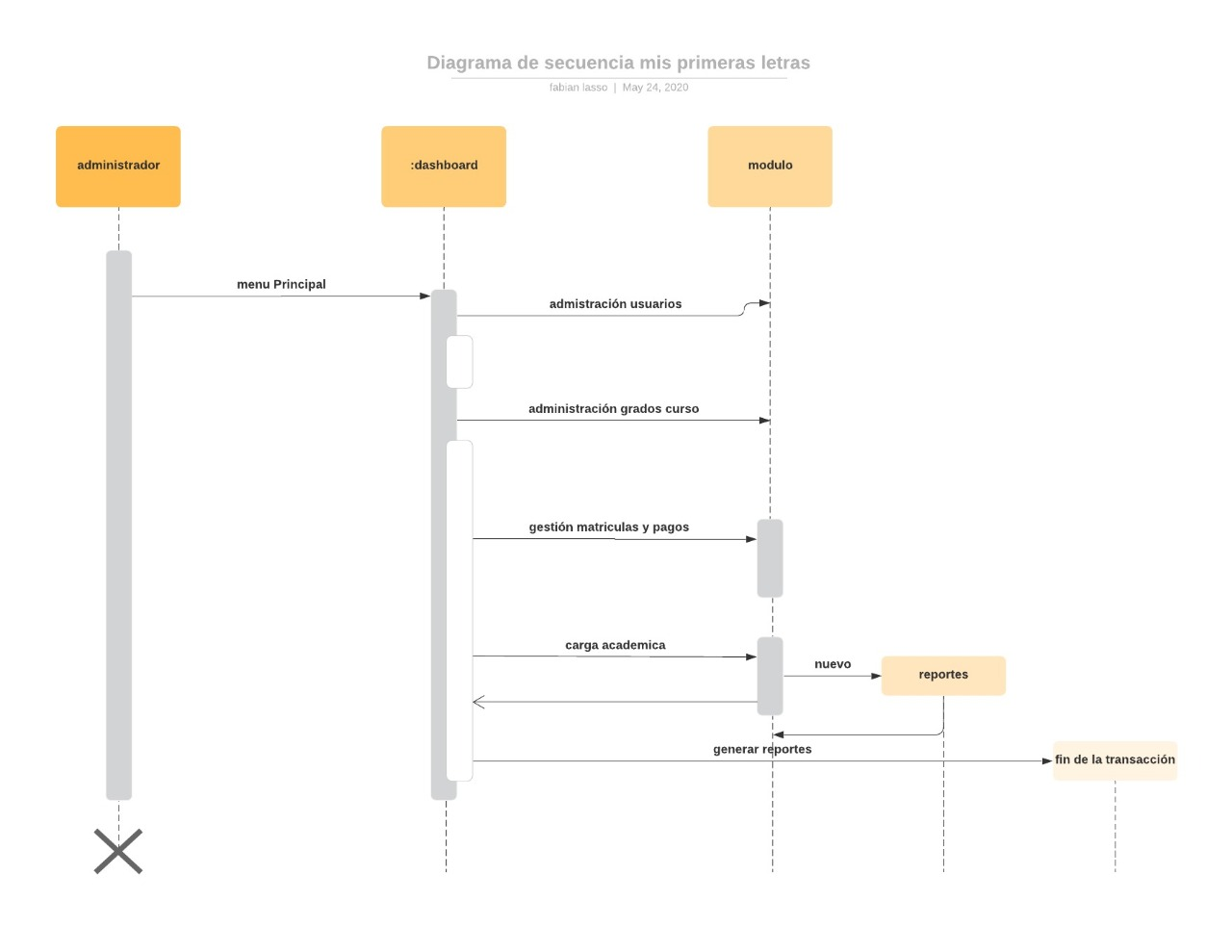


Figura No. 3 Vista Lógica

* 1. **VISTA DE DESPLIEGUE**

En esta vista, también llamada vista física, se representa la gestión del software, se muestra como está dividido el sistema en componentes y las dependencias que hay entre estos. Los diagramas UML que se utilizan para representar esta vista son los Diagrama de componentes y de paquetes.

Para el caso del proyecto Mis Primeras Letras, se distribuye sobre varios nodos de procesamiento (hardware), los cuales se encuentran relacionados con elementos presentes en las vistas de arquitectura restantes (alto nivel, lógica, procesos, implementación y datos). Adicionalmente, la vista de despliegue puede presentar varias configuraciones, como son: ambiente de desarrollo, pruebas, producción, entre otros.

En consecuencia, para describir la vista física en el presente documento, se empleó el siguiente diagrama de despliegue:

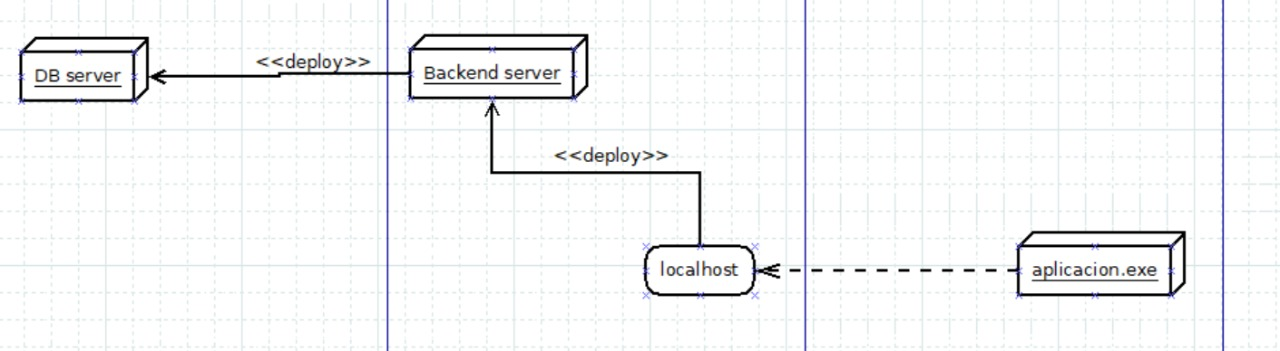


Figura No. 8 Diagrama de Despliegue

* + 1. **Descripción de nodos físicos**

Los servidores destinados para el proyecto alojarán al software correspondiente a servidores de aplicaciones, servidores de base de datos, sistemas operativos, el contenedor de servlets, motor de base de datos, conexión a herramienta para gestión de metadato y componentes o librerías que gestionen la persistencia.

En las siguientes tablas se detallan las características principales de los nodos que componen el Sistema De Información Mis Primeras Letras (ambiente de pruebas/producción):

Tabla 8. Características de los nodos

|  |  |
| --- | --- |
| **SERVIDOR** | **DESCRIPCIÓN** |
| Tipo | *Servidor de aplicaciones y de Base de datos* |
| Procesadores | *Intel Core I5 6300U* |
| Capacidad de Memoria | *8GB* |
| Capacidad de almacenamiento | *1 TB* |
| Sistema operativo + versión | *Windows 10* |
| Tipo de conexión | TCP/IP |
| Puerto | *8080* |
| Información adicional | *El servidor de aplicaciones y de base de datos es el mismo en este caso* |
| **SERVIDOR** | **DESCRIPCIÓN** |
| Tipo | *Servidor de base de datos* |
| Procesadores | *Intel Core I5 6300U* |
| Capacidad de Memoria | *8GB* |
| Capacidad de almacenamiento | *1 TB* |
| Sistema operativo + versión | TCP/IP |
| Tipo de conexión | *8080* |
| Puerto | *1413* |
| Información adicional | *El servidor de aplicaciones y de base de datos es el mismo en este caso.* |

* 1. **VISTA DE IMPLEMENTACIÓN**

La vista de desarrollo se centra en la organización real de los módulos de software en el ambiente de desarrollo del sistema. Para el caso del Sistema de Información Mis Primeras Letras se debe tener en cuenta los requisitos internos relativos a la facilidad de desarrollo, administración del software, reutilización y elementos comunes, y restricciones impuestas por las herramientas o el lenguaje de programación que se use.

Por consiguiente, el software se empaqueta en componentes que a su vez se componen de subsistemas que se organizan en una jerarquía de capas, cada una de las cuales brinda una interfaz estrecha y bien definida hacia las capas superiores del sistema, tal y como se muestra en la Figura 9.

|  |  |
| --- | --- |
| **Ejecutable .EXE** | |
| **Windows Forms** |
| **C#** | |
| **Linq To SQL** | |
| **SQL Server** |

Figura No. 9 Modelo de Implementación

* 1. **VISTA DE DATOS**

El modelo para el gestor documental está dado por las siguientes entidades:

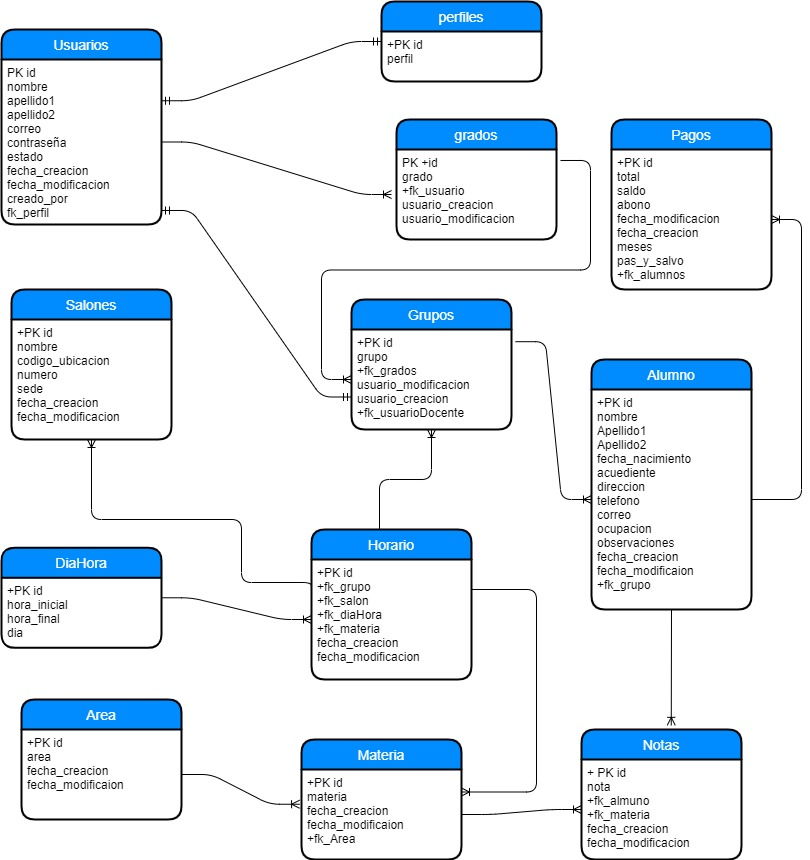


Figura No. 10 Modelo de Datos Gestor Documental

El modelo de base de datos del sistema de administración de usuarios se representa con el siguiente modelo físico.

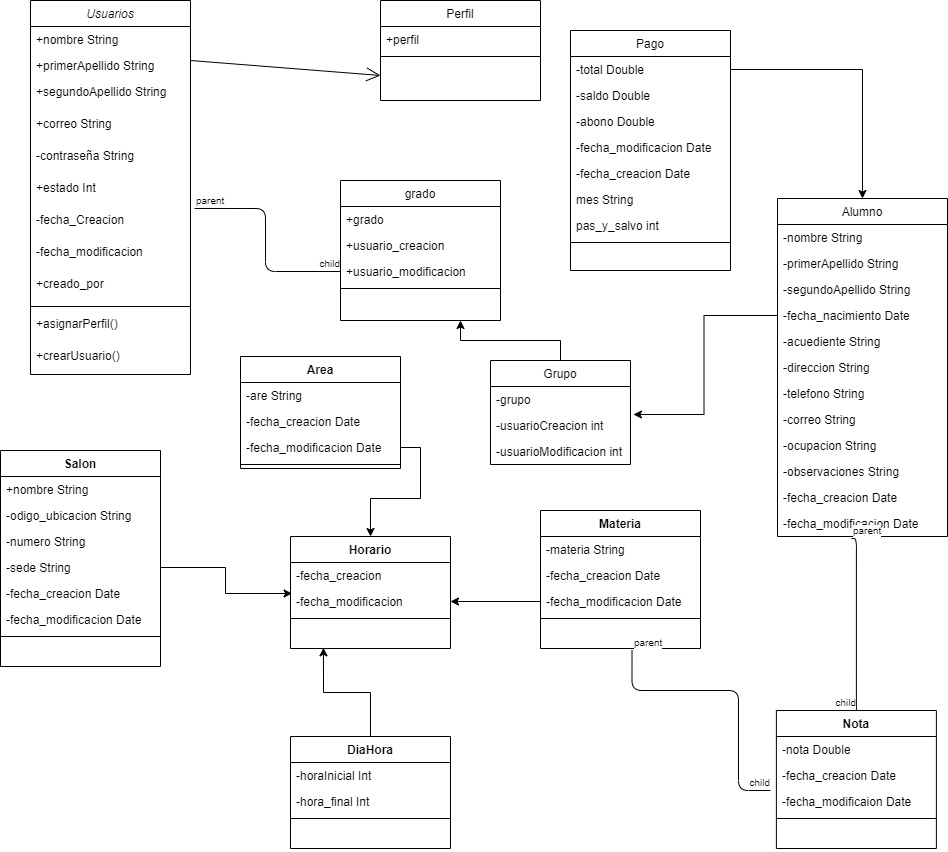


Figura No. 11 Diagrama de Clases

1. **CALIDAD**

Uno de los factores importantes dentro de un sistema de software es la calidad, que consiste en que el software cumpla con los requerimientos funcionales y no funcionales además de cumplir con las expectativas de los stakeholders y presente el menor número de fallos posible.

1. Lista de los atributos de calidad asociados e implementados en el proyecto
2. El sistema debe garantizar la seguridad y la protección de datos del usuario.
3. El sistema debe estar disponible las 24 horas de los 7 días de la semana
4. El sistema debe cumplir con los estándares de usabilidad según los diagras de causos de usos diseñados.

Con base en las anteriores premisas, se realizó el diseño conceptual presentado en los anteriores apartados. Entre otros, el sistema de información emplea el estilo arquitectónico de componentes independientes, haciendo uso de patrones de diseño de software; como es el caso del patrón MVC (Modelo, Vista, Controlador), permitiendo inferir que es una arquitectura modular escalable.