**SIGESTPROSO**

**Modelo Diseño**

***Grupo 1***

Patricia Cadenas Quijano

Javier García Tomillo

Mario Muñoz Sanz

Mario Santos Serrano

# Historial de revisiones del documento

|  |  |  |  |
| --- | --- | --- | --- |
| *Fecha* | *Versión* | *Descripción* | *Autor/es* |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**1 – Introducción**

**2 – Objetivo**

**3 – Arquitectura hardware**

**3.1 – Diagrama de despliegue**

**4 - Modelo Estructural**

**4.1 – Arquitectura software**

**4.1.1 – Capa presentación**

**4.1.2 – Capa lógica**

**4.1.3 – Capa datos**

**4.2 – Diagrama diseño**

**4.3 – Descripción de las clases**

**4.3.1 - Usuario**

**4.3.2 - Trabajador**

**4.3.3 - JefeProyecto**

**4.3.4 - Desarrollador**

**4.3.5 - Administardor**

**4.3.6 - ResponsablePersonal**

**4.3.7 - Vacaciones**

**4.3.8 - Proyecto**

**4.3.9 - Fase**

**4.3.10 - Iteracion**

**4.3.11 - Actividad**

**4.3.12 – Tarea personal**

**4.3.13 – Artefacto**

**4.3.14 – InformeTareas**

**4.3.15 - Rol**

**4.3.16 – TipoTarea**

**4.3.17 - Configuracion**

**4 – Diagramas de secuencia**

**1 – Introducción**

El objetivo del presente documento es la especificación del modelo de diseño completo del sistema de “Seguimiento Integrado de la GESTión Temporal de PROyectos de Software (SIGESTPROSO)”. Incluye el diagrama de despliegue, diagrama de diseño, arquitectura software y diagramas de secuencia. Este documento será referencia para el uso posterior de los desarrolladores.

**2 – Referencias**

* Modelo de datos

**3 – Arquitectura hardware**

Los usuarios accederán a la aplicación a través de Internet por medio de sus navegadores web. Tendremos dos nodos, el cliente o máquina de un usuario cualquiera, y el servidor donde se aloja la aplicación, formando un claro sistema cliente/servidor.

El sistema cliente/servidor es una arquitectura muy extendida basada en una clara separación entre la interfaz de usuario del sistema (que reside en el cliente) y los datos persistentes del sistema (que residen en el servidor). Los sistemas cliente/servidor son un extremo del espectro de los sistemas distribuidos y requieren tomar decisiones sobre la conectividad de red de los clientes a los servidores y sobre la distribución física de los artefactos software del sistema a través de los nodos.

**3.1 – Diagrama de despliegue**

**DiagramaDespliegue.wmf**

**4 - Modelo Estructural**

**4.1 – Arquitectura software**

La aplicación a desarrollar consiste en una aplicación web que permita la gestión de los proyectos software en los que trabaja una empresa. Podemos pensar rápidamente en tres tareas diferenciadas dentro de este sistema:

* Interfaz de usuario
* Operaciones del negocio, es decir, el trabajo real de la aplicación
* Gestión de la información almacenada

Viendo esto, vamos a dividir el sistema en **3 capas**. De esta forma tendremos una aplicación que permitirá dividir fácilmente sus tareas de desarrollo, mantenimiento y

escalabilidad.

Esta separación en capas presenta una ventaja muy importante: En un momento dado, sería posible la sustitución de una de las capas por una completamente nueva sin un impacto en el resto del sistema. Por ejemplo, se podría substituir el SGBD sin apenas reprogramar los archivos contenidos en la capa de *Lógica*.

Como inconveniente a esta arquitectura de capas, implicará un sobreesfuerzo en la etapa de análisis y una mayor complejidad en el desarrollo en el momento de la codificación si queremos respetar una separación tan estricta de todas las capas.

**4.1.1 – Capa Presentación**

Como su propio nombre indica, esta capa contiene la interfaz del sistema, formada por varios formularios web. Aquí se van a ejecutar las instrucciones que el usuario invoque desde la capa de Presentación y también las solicitudes de información de la base de datos.

**4.1.2 – Capa Lógica**

En esta capa se sitúan las entidades y operaciones reales del sistema. Aquí se van a ejecutar las instrucciones que el usuario invoque desde la capa de Presentación y también las solicitudes de información de la base de datos. Dentro de esta capa se sitúan las clases mostradas en el diagrama de diseño.

**4.1.3 – Capa Datos**

En esta capa encontraremos todas las clases y operaciones necesarias para el acceso al sistema gestor de bases de datos así como a la información contenida en el mismo.

**4.2 – Diagrama diseño**

El diagrama de diseño se corresponde con el conjunto de clases y asociaciones que se ven a continuación. Forman la capa lógica de la aplicación.

Adjuntamos la imagen y una dirección Web desde la que visualizar más claramente el mismo: [WWW.MODELODISEÑO.ES](http://WWW.MODELODISEÑO.ES)

**4.3 – Descripción de las clases**

**4.3.1 – Usuario**

Modela a un usuario, aún sin identificarse en el sistema. De esta clase heredan el resto de clases del sistema.

**4.3.2 – Trabajador**

Representa a los usuarios principales del sistema: el Jefe de Proyecto y el Desarrollador, los cuales heredan de la misma.

**4.3.3 – JefeProyecto**

Usuario del sistema encargado de dirigir los diferentes proyectos creados en la aplicación.

**4.3.4 – Desarrollador**

Usuario del sistema que registra los tiempos empleados en las diferentes tareas desarrolladas para llevar a cabo las actividades adjudicadas.

**4.3.5 – Administardor**

Usuario del sistema encargado de configurar la aplicación cargando los datos iniciales y los proyectos.

**4.3.6 – ResponsablePersonal**

Usuario del sistema orientado a la gestión de los trabajadores existentes en la aplicación con su creación y la obtención de informes.

**4.3.7 – Vacaciones**

Modela un período de vacaciones escogido por el usuario. Se restringe a poder escoger 4 semanas de vacaciones con inicio en lunes y fin en domingo, divididas en tres períodos vacacionales como máximo.

**4.3.8 – Proyecto**

Modela a los diferentes proyectos software sobre los cuales se realizará el seguimiento temporal. El proyecto está asociado con las diferentes fases del proceso unificado, así como con los trabajadores implicados en el mismo.

**4.3.9 – Fase**

Modela las cuatro fases del Proceso Unificado: Inicio, Elaboración, Construcción y Transición.

**4.3.10 – Iteracion**

Representa el concepto Iteración en el marco del Proceso Unificado. El Jefe de Proyecto decidirá la creación de una o más iteraciones en cada Fase.

**4.3.11 – Actividad**

Modela las distintas actividades relacionadas con una iteración del Proceso Unificado. Tienen un tiempo de duración y un rol. Su finalización está marcada por la aprobación del Jefe de Proyecto y por la generación de un artefacto.

**4.3.12 – Tarea personal**

Representa las distintas tareas personales que un trabajador realiza para completar una actividad.

**4.3.13 – Artefacto**

Representa el documento que, junto con el Informe de Tareas aceptado, debe acompañar cada actividad para poder darse por cerrada.

**4.3.14 – InformeTareas**

Representa el resumen pasado por el Desarrollador al Jefe de Proyecto que, junto con el artefacto, debe acompañar cada actividad para poder darse por cerrada.

**4.3.15 – Rol**

Modela las diferentes relaciones de categoría-rol existentes en el sistema.

**4.3.16 – TipoTarea**

Enumerado que representa las diferentes opciones para el tipo de una Tarea Personal.

**4.3.17 – Configuracion**

Clase necesaria para gestionar aspectos tales de la configuración como la categoría máxima para las relaciones rol-categoría o el número máximo de proyectos en los que puede participar un trabajador.

**5 – Diagramas de secuencia**

**5.1.1 – Casos de uso de Usuario**

**Iniciar sesión**

**Iniciar_sesion.wmf**

**Cerrar sesión**

**Cerrar_sesion.wmf**

**5.1.2 – Casos de uso de Administrador**

**Dar alta proyectos**

**Dar_alta_proyectos.wmf**

**Configurar sistema**

**Configurar_sistema.wmf**

**5.1.3 – Casos de uso de Responsable personal**

**Dar alta cuenta trabajadores**

**Alta_cuentas_trabajadores.wmf**

**Seguimiento de desarrolladores**

**seguimiendo_de_desarrolladores.wmf**

**Obtener informe acumulación trabajo**

**5.1.4 – Casos de uso de Trabajador**

**Obtener resumen proyectos finalizados**

**Escoger vacaciones**

**5.1.5 – Casos de uso de Jefe Proyecto**

**Obtener informe seguimiento temporal**

**Definición proyecto**

**Editar fase**

**Finalizar actividad/etapa/iteración**

**5.1.6 – Casos de uso de Desarrollador**

**Registrar tiempo tareas**

**Gestionar artefacto**

**Consultar actividades y duración**