



Departamento de Informática
Universidad Técnica Federico Santa María



Informe de Proyecto – INF-225-2018-1- CSJ

Proyecto:

Gestionador de Solicitudes Para GPI

03-08-2018

Grupo: *"HalfSoftware"*

Integrantes:

Nombres y Apellidos	Email	ROL USM
Javier Henríquez	javier.henriquez.13@sansano.usm.cl	201304553-7
Cristian Aguilar	cristian.aguilar.13@sansano.usm.cl	201304741-6

1. Contexto del proyecto

Objetivo de proyecto

Antes de indicar los objetivos que intenta cubrir este proyecto, recordemos cual es el contexto por el cual se originó.

El proyecto toma lugar cuando surge la necesidad de mejorar la comunicación y eficiencia en las distintas áreas de la empresa GPI. Dado esto, el problema que la empresa desea abordar, es la gestión y generación de solicitudes de trabajo y órdenes de compra, este problema varía su dificultad de acorde a la complejidad de la organización de la empresa, además que cada empresa presenta sus particularidades al momento de operar, como es en el caso de esta, donde existe un ERP el cual tiene que ser compatible con nuestra solución o al menos no debe entorpecer la operatividad de este.

Para enfrentar este problema, *HalfSoftware* trató de modelar la funcionalidad al interior de la empresa, y tratar de extraer los elementos y acciones en los cuales se podrían intervenir para asegurar una mejor comunicación. Debido a que el exceso de confianza, el desorden y la necesidad de abarcar todo puede conducir al desastre, se planteó abarcar pequeñas metas y simplificar el problema real a sub-problemas presentes, con la idea de aplicar escalabilidad en nuestra solución. Esto permitirá crear una solución modular, la cual nos dará acceso a añadir y eliminar funciones dentro del sistema.

Dado lo anteriormente descrito, nuestro proyecto se centrará en generar una plataforma donde las solicitudes sean gestionadas de manera ordenada incluyendo detalles de los materiales requeridos, la urgencia de la solicitud y otros campos específicos de la obra a la que pertenece.

Contenido del Informe a Entregar

<i>Objetivo.....</i>	<i>2</i>
<i>Contenido del Informe a Entregar</i>	<i>3</i>
1. Requisitos clave (Actualizado).....	4
2. Árbol de Utilidad (Actualizado)	5
3. Modelo de Software.....	6
4. Trade-offs entre tecnologías.....	8

2. Requisitos clave (Actualizado)

Tabla 1: Requisitos funcionales (actualizados)

A continuación se presentan los requisitos funcionales y extra-funcionales que abarcan el proyecto, se distinguen los cambios actualizados en color azul los requisitos agregados se identifican con la simbología (*).

Req. Funcional	Descripción y medición (máximo 2 líneas)
Crear una solicitud de materiales (*)	Un usuario perteneciente a una obra podrá crear una solicitud de materiales con una lista asociada.
Ver materiales por solicitud (*)	Todos los usuarios involucrados podrán ver la lista de materiales asociados a una solicitud de materiales.
Comparar solicitud con elementos en bodega	El sistema debe ser capaz de analizar que materiales y en qué cantidad se encuentran presentes en bodega.
Descontar elementos de bodega.	Se deben descontar la cantidad de materiales de la base de datos para cierta solicitud.
Generar solicitud de compra	La plataforma debe ser capaz de generar la solicitud de compra con los materiales y cantidades necesarias.
Anular solicitud de compra.	Una solicitud puede tener el estado “anulada” en caso de haberse cometido algún error o desincronización.
Actualizar estado de la solicitud de materiales	La plataforma actualiza automáticamente el estado de una solicitud al pasar etapas (obras, bodega, adquisiciones).
Generar recepción de materiales	El sistema debe generar un documento que verifique los materiales recibidos.
Reservar materiales en bodega (*)	Si bodega cuenta con existencias de un material, estas deben reservarse para la obra.
Estado de materiales(*)	Los materiales tienen un estado independiente del estado de la solicitud, dependiendo de su disponibilidad o no.
Comunicación con Laudus(*)	El encargado de compras puede registrar las solicitudes de compras con el ERP Laudus.

Tabla 2: Requisitos extra-funcionales (actualizados)

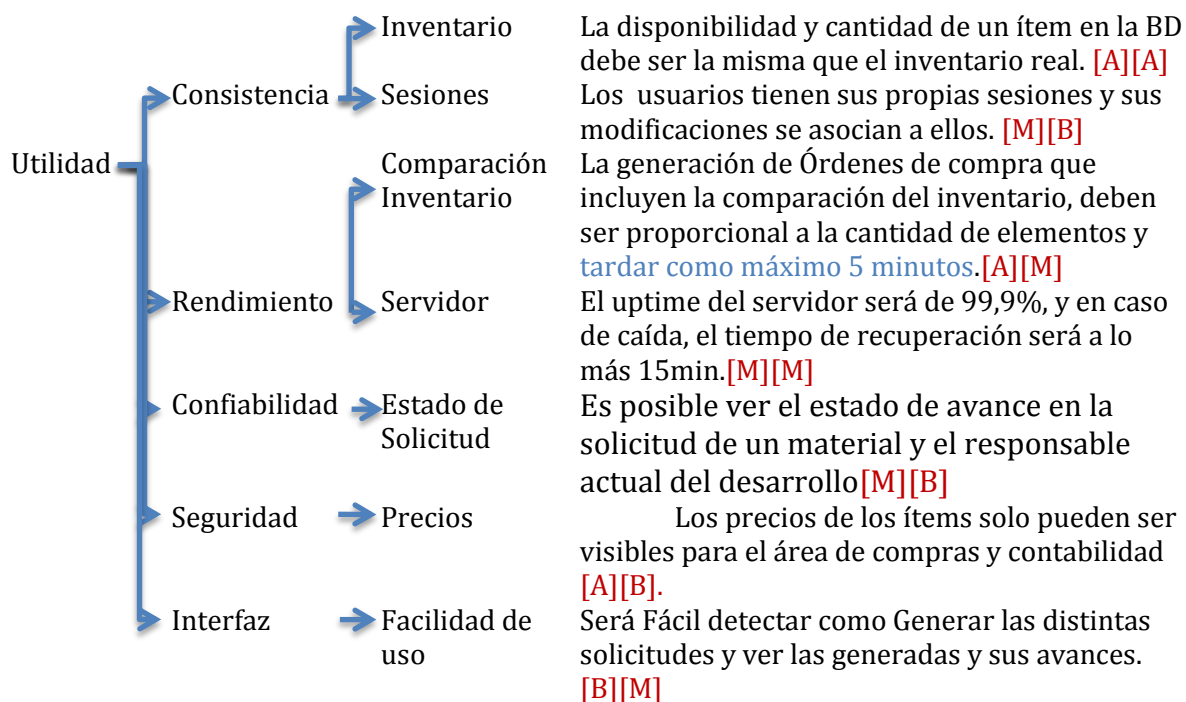
Req. extra-funcional	Descripción y medición (máximo 2 líneas)
Disponibilidad	Podrá atender a 30 usuarios simultáneamente.
Rendimiento	El uptime del servidor será de 99,9%, y en caso de caída, el tiempo de recuperación será a lo más 15min.
Rendimiento	La generación de Órdenes de compra que incluyen la comparación del inventario, deben ser proporcional a la cantidad de elementos y tardar como máximo 5 minutos.
Seguridad	Los precios de los ítems solo pueden ser visibles para el área de compras y contabilidad
Interfaz	Será Fácil detectar como Generar las distintas solicitudes y ver las generadas y sus avances
Consistencia(*)	Los usuarios tienen sus propias sesiones y sus modificaciones se asocian a ellos.
Consistencia (*)	La disponibilidad y cantidad de un ítem en la BD debe ser la

	misma que el inventario real.
Confiabilidad (*)	Es posible ver el estado de avance en la solicitud de un material y el responsable actual del desarrollo

3. Árbol de Utilidad (Actualizado)

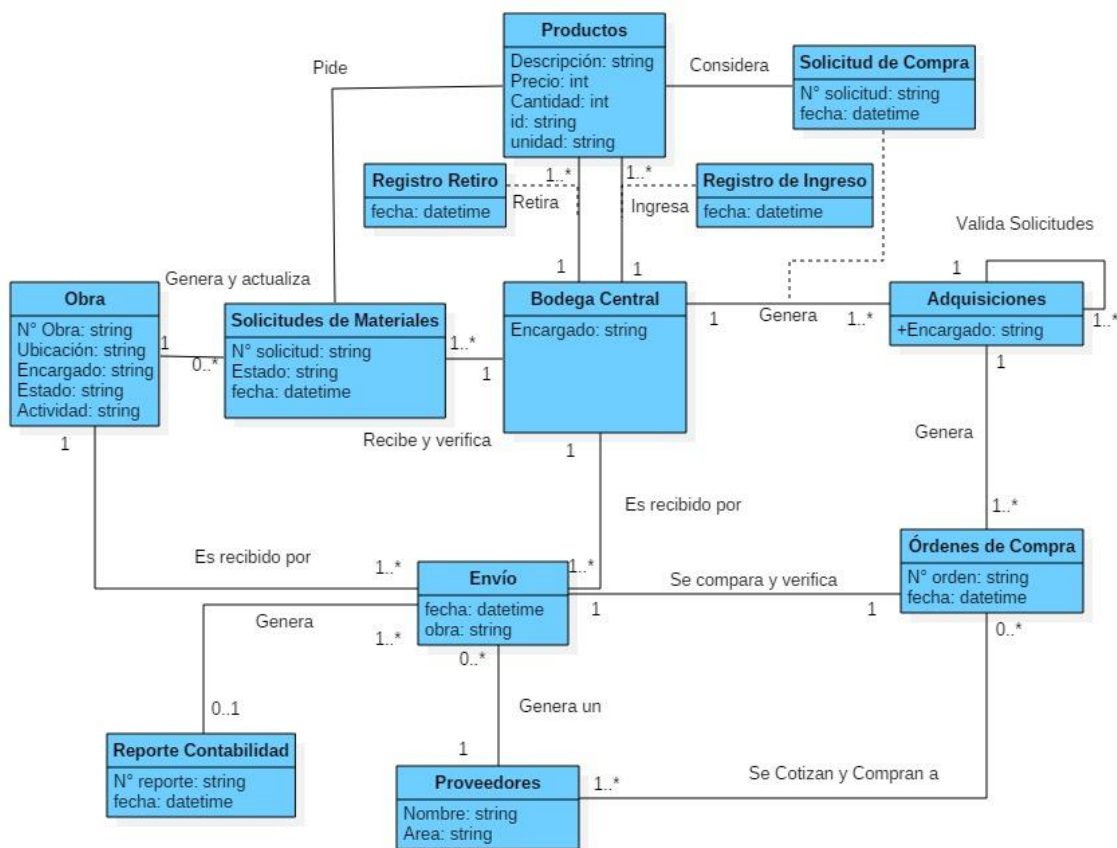
Se actualiza el árbol de utilidad considerando la actualización de los requisitos extra funcionales actualizados. Se consideran importancia y dificultad.

Ilustración 1: Árbol de utilidad



4. Modelo de Software

Modelo de Dominio:



La actualización del modelo de dominio contempla 2 cambios principales además de los atributos de clase, primero se le da más importancia a la clase solicitud que contiene información del avance de la solicitud de materiales en su valor “estado” según el avance en los procesos de GPI, por lo tanto ante una devolución se actualiza el estado de la solicitud.

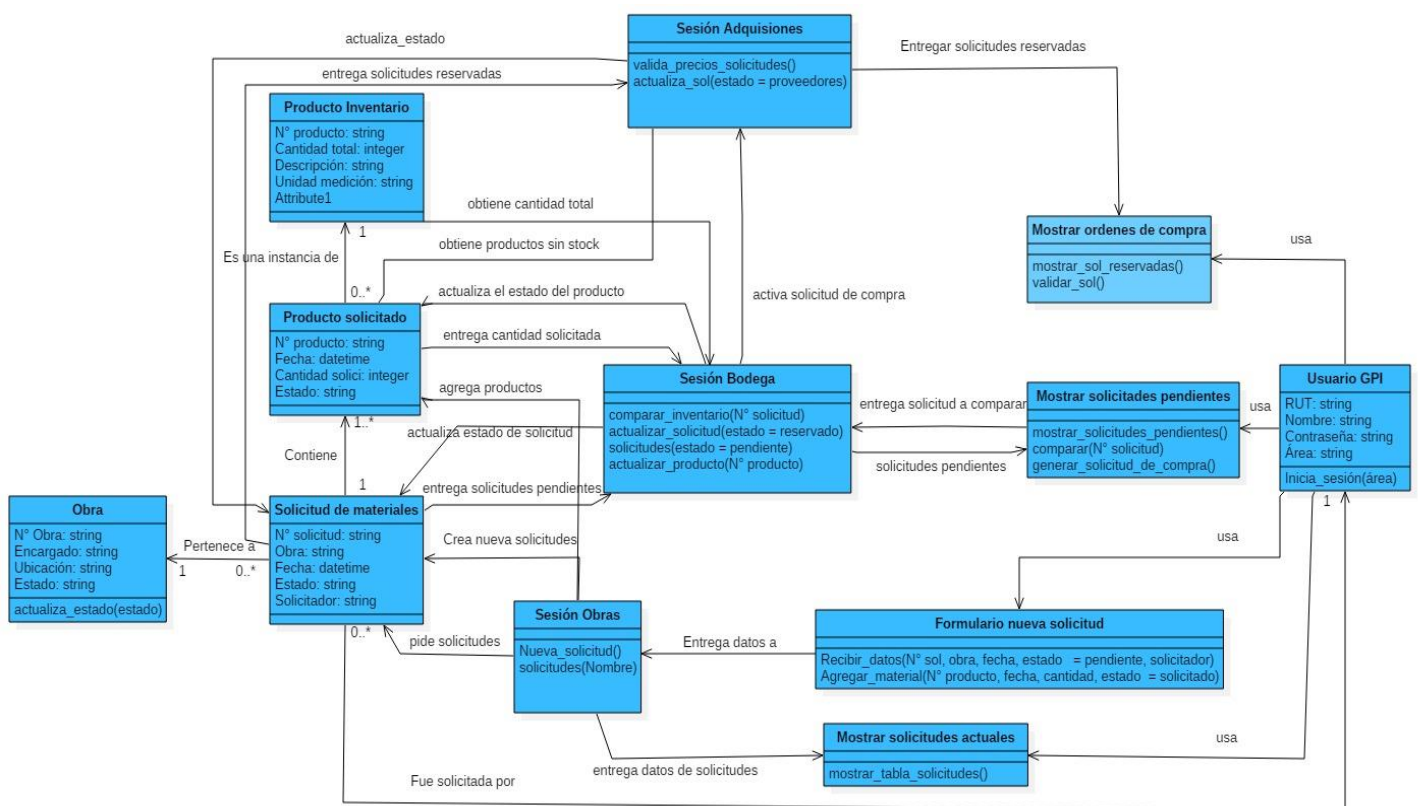
El segundo cambio corresponde a quitar la entidad “encargado de recepciones” y asumir que cada sector tiene personal asociado que cumple funciones relacionadas con otras entidades, en este caso envió, esto gracias a la abstracción del modelo de dominio.

Entidad	Descripción (1 línea)
Bodega Central	Encargado de Administrar los Materiales y Solicitudes
Obra	Se lleva a cabo operaciones, usa y pide materiales
Solc. de Materiales	Representa todas las solicitudes de materiales realizadas
Productos	Representa todos los materiales con código y cantidad
Registro Retiro	Guarda los retiros de material por caducidad o envío
Registro Ingreso	Detalla los productos ingresados con los datos de la recepción
Solicitud de Compra	Detalla los productos que faltantes en bodega para su compra

Adquisiciones	Verifica, Evalúa Solicitudes y las redirige a los Proveedores
Órdenes de Compra	Detalles de productos a cotizar ya validados en Adquisiciones
Proveedores	Recibe órdenes de compra de su ámbito y envía materiales
Envío	Representa los productos enviados desde el proveedor
Rpte. Contabilidad	Representa reportes de productos devueltos y defectuosos

Modelo de Software:

El modelo de software muestra más a la izquierda las clases representantes del “modelo” con los objetos con operaciones CRUD almacenados en la base de datos, las clases sesiones representan al controlador en un modelo MVC donde obtienen datos desde el modelo, se actualizan y se muestran en las vistas representadas en las clases de más a la derecha. El usuario interactúa con las vistas y queda registrado en las solicitudes que crea.



El modelo de software representa un modelo ideal de construcción **final**, en la entrega añadida a github se considera implementada la sesión obras y bodega. También es importante mencionar que se desarrolló la entrega siguiendo un patrón MVC con el modelo en una base de datos mysql, las vistas en html, que incluyen código php que trabaja directamente como controlador, aunque la separación entre estos últimos no es explícita como podría resultar en proyecto Django.

Tabla 3: Selección de Patrones

Intención	Patrón de Diseño	Razonamiento
Manejar solicitudes, crear, actualizar y comparar materiales en el inventario	Model-View-Controller (MVC)	El modelo facilita las interacciones de objetos con operaciones de creación y actualización, lo que es necesario para gestionar solicitudes y los materiales asociados a cada una de esta.
Crear nuevas instancias de solicitudes complejas	Builder	Este modelo ayuda a crear nuevas solicitudes de materiales, donde la complejidad esta en asociar al personal que lo solicita y a la obra a la que esta designada, también los materiales asociados.

5. Trade-offs entre tecnologías

Se darán a conocer los trade-offs, los cuales serán descritos por medio de softgoals y operacionalizaciones, luego se darán a conocer los detalles de las decisiones y los softgoals que cubren, con sus respectivas evaluaciones.

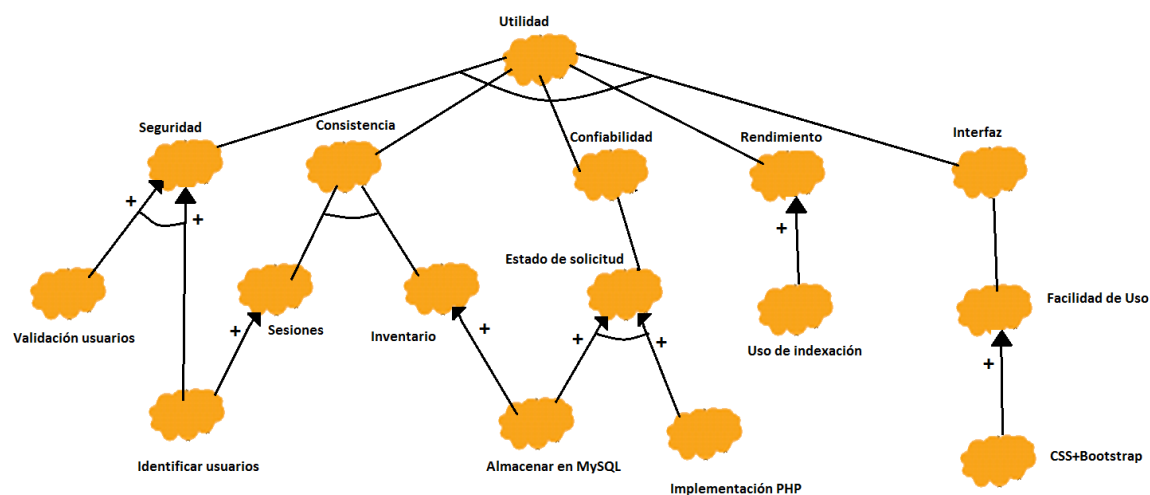


Ilustración 2: Softgoal Interdependency Graph (SIG)

Tabla 4: Trade-offs entre opciones tecnológicas

Decisión	Softgoal	Evaluación	Razonamiento
Implementación en PHP.	Confiabilidad	++	PHP es fácil de implementar y sirve como medio para contactar a la base de datos.
Almacenar datos en MySQL	Consistencia / Confiabilidad	++	Para el servicio que se intenta entregar, una base de datos relacional es ideal para garantizar la integridad de los datos y evitar duplicidad.
Usar CSS+Bootstrap en los HTML	Interfaz	++	Estas herramientas aportan un valor estético en la muestra de datos, lo que nos permite mostrar una interfaz agradable a los ojos del usuario.
Identificar usuarios.	Seguridad	++	Cada usuario autenticado tiene un rol asignado dentro de la empresa y se deben diferenciar.
Validar acceso según usuario	Seguridad	++	Según el tipo de usuario se deben mostrar las páginas respectivas para mantener confidencialidad.
Agregar indexación en la base de datos	Latencia	+	El uso de indexación en bases de datos relacionales es una buena práctica para disminuir el tiempo de respuesta en bases con grandes cantidades de datos