



Informe 3 - Ingeniería de Software

Proyecto: Integración de seguimiento de solicitudes de compra

Grupo: Innombrable

29 de agosto de 2018

Nombre y apellidos	Email	Rol USM
Sofía Carrasco C.	sofia.carrascoc@sansano.usm.cl	201521061-6
Michell Concha G.	michell.concha.14@sansano.usm.cl	201451529-4
Hugo Sepúlveda A.	hugo.sepulvedaa@sansano.usm.cl	201573515-8

1. Requisitos claves (actualizados)

1.1. Funcionales

Requisitos funcionales	Descripción y medición
Método para realizar solicitudes.	Es necesario crear un formulario que sea capaz de describir exactamente qué se necesita, e informarlo a quienes se encargan de autorizarlo y proveerlo.
Método para gestionar órdenes de compra.	Se deben poder crear órdenes a través de las selecciones de las cotizaciones hechas, además de asociarla con un despacho y la respectiva facturación, junto con el ERP.
Seguir el estado del proceso de solicitud, orden y despacho.	Es vital crear una orden de compra y saber las fechas de despacho por parte del proveedor, para no afectar el avance de las obras.
Enviar notificaciones sobre el estado de los pedidos.	La información sobre el estado de los pedidos debe ser distribuida a las partes interesadas, y una opción simple y directa coincidir con el formato que entrega el ERP Laudusson las notificaciones.
Existencia de inventario de los objetos en la bodega.	Si lo pedido está en bodegas de GPI, su búsqueda y disponibilidad puede ser facilitada por el bodeguero gracias a un sistema con la información.

1.2. Extra-funcionales

Requisitos extra-funcionales	Descripción y medición
[Confidencialidad] Restricción de información y privilegios según tipo de usuario (cargo).	Existe información que sólo puede ser vista por personas específicas, según sus cargos y atribuciones, por políticas internas de GPI.
[Estabilidad] Considerar fallas de conexión de los clientes internos al hacer solicitudes.	En caso de problemas de conectividad, el cliente interno debe recibir un feedback que garantice si su solicitud fue o no registrada en el sistema, posterior a la pérdida de conectividad.
[Disponibilidad] Limitar envíos de solicitudes a horario acotado.	Permite a bodegueros organizar y priorizar solicitudes, dando tiempo suficiente para contacto con proveedores. Fuera del horario establecido, no se permite el envío de formularios de solicitud.
[Interoperabilidad] Integración con ERP Laudus.	GPI ocupa un ERP llamado Laudus para administrar su información, recursos y servicios, por lo que es necesario comunicarse con este.
[Integridad] Consistencia de los datos en Base de Datos.	Debido al tipo de información que maneja GPI, los datos deben ser consistentes. Esto se logra con un sistema de gestión de BD para el caso. Es posible observar esto comparando información real con la gestionada por el sistema, o comparándola dentro del mismo en distintas vistas.

2. Historias de usuario (actualizadas)

Revisión de inventario
Yo, como bodeguero, quiero saber qué materiales se encuentran en bodega, para decidir si resuelvo la solicitud o no.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none"> ■ El bodeguero puede ver la lista de materiales en el inventario. ■ El bodeguero puede ver el stock de un determinado material. ■ El bodeguero puede aprobar una solicitud. ■ El bodeguero puede marcar una solicitud como resuelta.
<u>Conversación:</u> <ul style="list-style-type: none"> ■ Tanto el bodeguero de obra como el de bodega central deben poder tener acceso al inventario. ■ El bodeguero puede entregar un material similar al solicitado que cumpla con la misma función. ■ El bodeguero filtra solicitudes en caso de error.
<u>Cambios:</u> <ul style="list-style-type: none"> ■ Se agrega aprobación de solicitud como criterio de aceptación, por nueva conversación que influye directamente en cómo se modifica el inventario.

Notificaciones de atraso
Yo, como encargado de compras, quiero recibir notificaciones de problemas con fechas de los pedidos, para reiterar la solicitud al proveedor.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none"> ■ El encargado de compras recibe una notificación a través del sistema cuando se acerca la fecha de entrega de una orden de compra o se ha excedido. ■ El encargado de compras puede reiterar la solicitud al proveedor.
<u>Conversación:</u> <ul style="list-style-type: none"> ■ La fecha de entrega puede ser fijada por el encargado de compras en base a la urgencia de adquirir los materiales solicitados en la orden de compra. ■ La reiteración de la solicitud puede ser a través del sistema o no (por ejemplo, a través de un llamado telefónico).
<u>Cambios:</u> <ul style="list-style-type: none"> ■ Se complementa primer criterio, con el exceder de la fecha.

Recepción de pedidos
Yo, como bodeguero, quiero informar rápidamente el estado de un material al ser recepcionado o despachado.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none"> ■ El cliente interno y el encargado de compras reciben una notificación sobre el estado del pedido y despacho. ■ El proveedor recibe una notificación sobre el estado del pedido y devolución. ■ El bodeguero es capaz de actualizar el estado del pedido y la ubicación del despacho.
<u>Conversación:</u> <ul style="list-style-type: none"> ■ El encargado de compras puede además notificar el estado del pedido al proveedor a través de otros medios (por ejemplo, un llamado telefónico). ■ Se debe notificar además la fecha de recepción y detalles.
<u>Cambios:</u> <ul style="list-style-type: none"> ■ Se complementa la conversación, con la importancia de los detalles de recepción.

Petición de Materiales
Yo, como cliente interno, quiero realizar una petición de materiales, para un término de obra oportuno.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none"> ■ El cliente interno tiene acceso a un formulario que permite realizar correctamente un pedido. ■ El cliente interno puede ver todos sus pedidos. ■ El cliente interno puede ver si su pedido fue aprobado. ■ El cliente interno recibe notificaciones sobre el estado de su pedido.
<u>Conversación:</u> <ul style="list-style-type: none"> ■ El cliente interno será quien haga una solicitud, a causa de la necesidad de la obra en la que participa. ■ Este debe ver cuál es el estado de su solicitud, para estimar los tiempos de llegada de los objetos que puedan afectar al tiempo de realización de la obra.
<u>Cambios:</u> <ul style="list-style-type: none"> ■ Se agrega criterio sobre ver todos los pedidos.

Visualización de solicitudes
Yo, como encargado de compras, quiero poder visualizar las solicitudes no resueltas, para optimizar la realización de órdenes de compra.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none"> ■ El encargado de compras puede ver todas las solicitudes no resueltas. ■ El encargado de compras puede ver el detalle de cada una de las solicitudes (no resueltas).
<u>Conversación:</u> <ul style="list-style-type: none"> ■ Actualmente existe un libro de solicitudes que cumple ese rol. ■ Actualmente la resolución de las solicitudes es informada de forma telefónica con los clientes internos, lo que podría evitarse con un sistema que sea capaz de notificar el estado de las solicitudes. ■ Las solicitudes deben estar ordenadas por fecha y clasificadas por prioridad, similar a como lo están actualmente.
<u>Cambios:</u> <ul style="list-style-type: none"> ■ Ninguno.

Realización de una orden de compra
Yo, como encargado de compras, quiero realizar una orden de compra, para adquirir prontamente los materiales faltantes.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none">■ El encargado de compras tiene acceso a un formulario que permite realizar correctamente una orden de compra a un determinado proveedor, en base a solicitudes no resueltas.■ El encargado de compras puede ingresar una fecha límite a partir de la cual el pedido es considerado atrasado.■ Se le envían al proveedor las órdenes de compra.
<u>Conversación:</u> <ul style="list-style-type: none">■ Se le debe facilitar al encargado de compras, la clasificación de las solicitudes para que estas sean más ordenada cuando se les comunique al proveedor.■ Las órdenes de compras son un punto crítico, que se definen según los criterios del encargado de compras para priorizar, después de recopilar las solicitudes.
<u>Cambios:</u> <ul style="list-style-type: none">■ En criterios se detalla que la orden depende de solicitudes no resueltas.

Integración de Laudus al sistema
Yo, como encargado de compras, quiero utilizar Laudus, para respaldar el proceso de compras.
<u>Criterios de aceptación:</u> <ul style="list-style-type: none">■ El encargado de compras recibe una notificación desde el sistema, que informa la existencia de una nueva solicitud.■ El encargado de compras solicita una cotización a un proveedor.■ El encargado de compras crea y registra una orden de compra en Laudus, a través del sistema.■ El encargado de compra registra una factura en Laudus a través del sistema.
<u>Conversación:</u> <ul style="list-style-type: none">■ El uso del ERP Laudus ya está establecido en GPI, lo que obliga a que el nuevo sistema interactúe con él.■ Existen algunos documentos cuya información ya está estructurada por Laudus, lo que establece un formato para ingresar dicha información al sistema.
<u>Cambios:</u> <ul style="list-style-type: none">■ Se elimina un criterio respecto a la creación de las órdenes, abordado en otra historia de usuario.

3. Árbol de utilidad (actualizado)

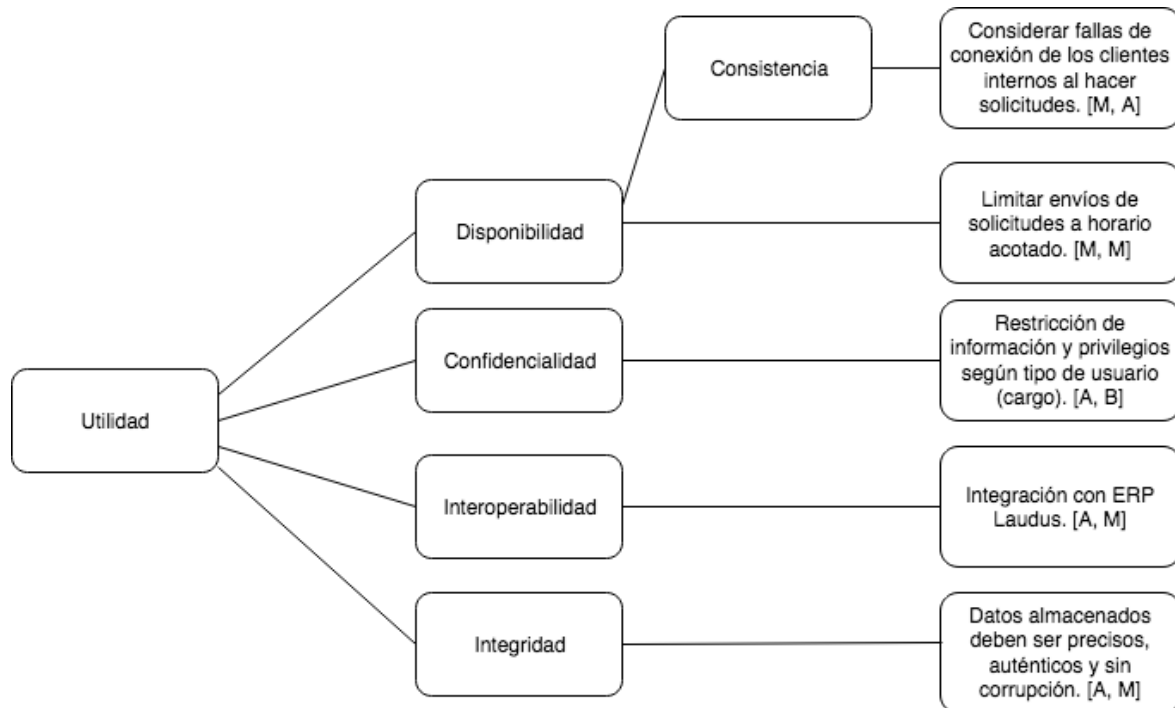
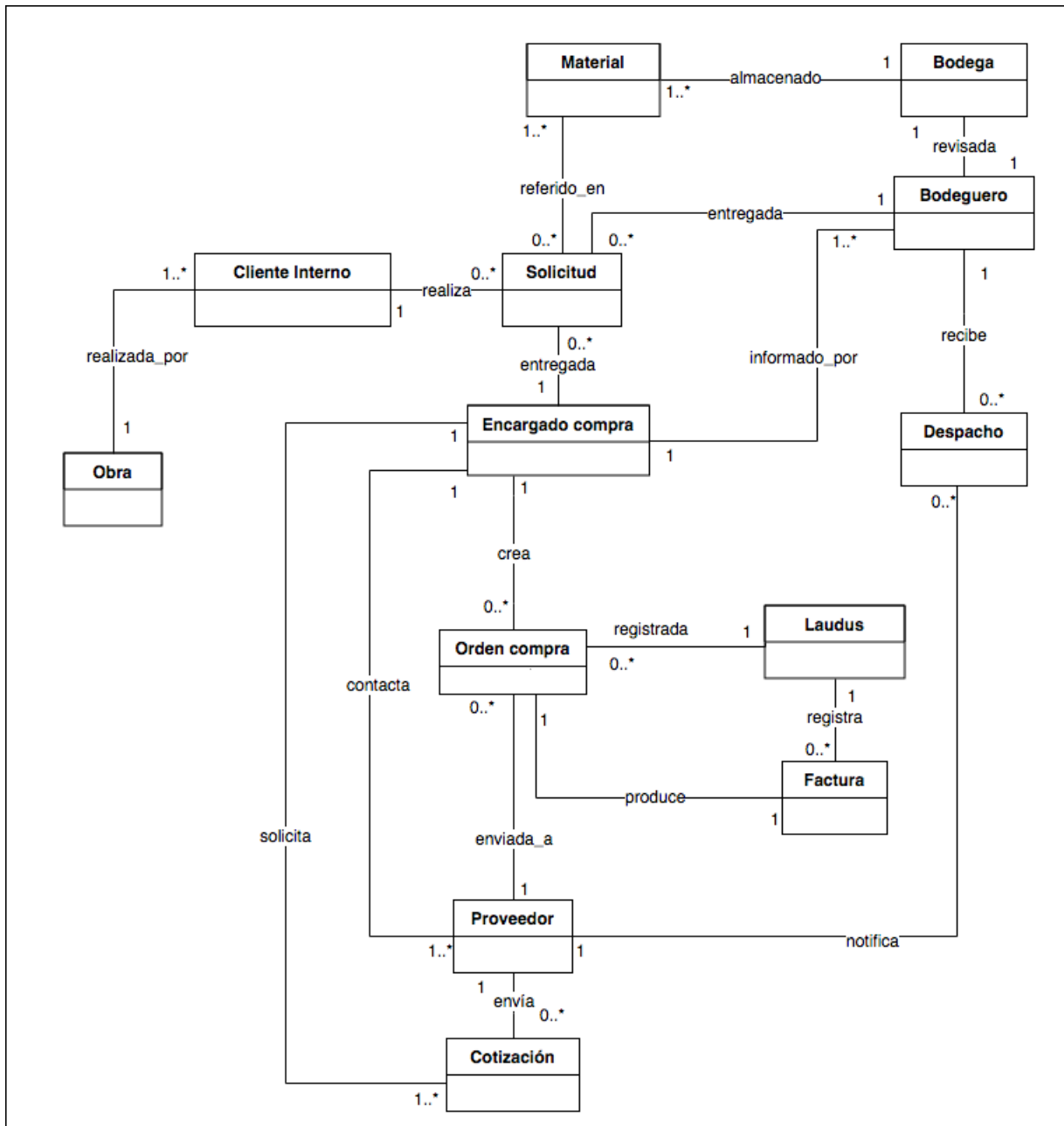


Figura 1: Árbol de utilidad y priorización de requisitos.

4. Modelo de software (actualizado)

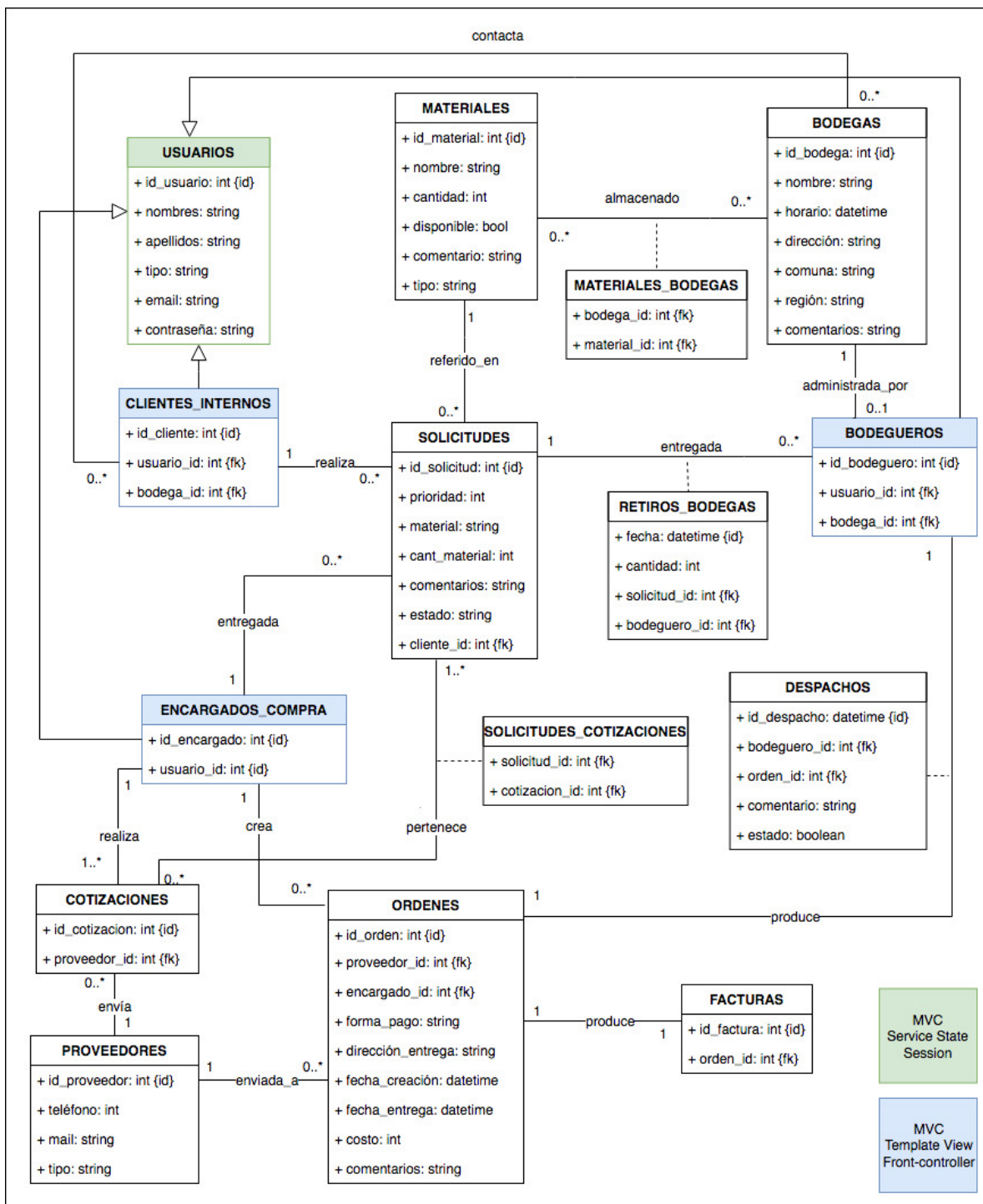
4.1. Modelo de Dominio



Entidad	Descripción
Obra	Lugar físico donde se requiere un material.
Cliente Interno	Persona que puede realizar solicitudes.
Solicitud	Formalización y especificación de pedido de material.
Material	Elemento específico requerido.
Bodega	Lugar físico donde se almacenan los materiales.
Bodeguero	Persona encargada de la administración de bodega.
Encargado compra	Persona encargada de gestionar proceso de compras.
Despacho	Paquete con elementos solicitados desde una obra.
Orden compra	Documento en que se detalla la compra a realizar.
Factura	Documento que acredita y detalla la compra-venta de materiales.
Laudus	ERP utilizado para registrar proceso de compras.
Proveedor	Entidad que ofrece y provee materiales.
Cotización	Documento que especifica los costos asociados a materiales.

Tabla 1: Entidades y descripciones del modelo de dominio.

4.2. Modelo de clases



Clase	Descripción
USUARIOS	Personas que interactúan con el software.
MATERIALES	Elementos específicos requeridos.
MATERIALES_BODEGAS	Asociación entre un material específico y la bodega donde se encuentra.
CLIENTES_INTERNOS	Tipo de usuario que interactúa con las clases de bodegas y solicitudes.
BODEGAS	Lugares físicos donde se almacenan los materiales.
BODEGUEROS	Tipo de usuario encargado de la administración de bodega.
SOLICITUDES	Formalización y especificación de pedido de material.
RETIROS_BODEGAS	Asociación que se hace efectiva al momento de recibir un pedido en bodega.
DESPACHOS	Paquete con elementos solicitados desde una obra.
FACTURAS	Documento que acredita y detalla la compra-venta de materiales.
ENCARGADOS_COMPRA	Tipo de usuario que gestiona el proceso de compra.
COTIZACIONES	Documento que especifica los costos asociados a materiales.
PROVEEDORES	Entidad que ofrece y provee materiales.
ORDENES	Documento en que se detalla la compra a realizar.
SOLICITUDES_COTIZACIONES	Asociación entre la creación de una cotización y las solicitudes que intenta resolver.

Tabla 2: Clases y descripciones de las clases del modelo de clases.

Intención	Patrón de diseño	Razonamiento
Relacionar sistema de software con gestor de base de datos MySQL.	Active Record	Active Record permite manejar datos ya modelados (relacional) para un solo gestor de BD, suficiente para lo que requiere este proyecto.
Hacer consultas según parámetros específicos, con otros objetos asociados.	Query Object	Complementando con Active Record, resulta muy eficiente agregar atributos que resulten de las relaciones entre datos de distintas tablas, lo que se puede realizar a través de Active Record y las instancias de los objetos consultados.
Resolver peticiones que son distintas según tipo de perfil.	Front-controller	aunque existen similitudes en los tipos de datos que manejan los usuarios, estos deben ser presentados de distintas maneras, con distinta manipulación y con mayor simpleza frente a los mismos.
Mostrar información en vistas de forma estática y dinámica.	Template View	Rails ya integra una forma de incrustar objetos consultados a través de su ORM directamentea HTML, de forma que se puedan mostrar conjuntos de objetos con facilidad.
Administrar las sesiones por tipo de usuario	Server session state	Se prefiere las sesiones en server por mayor seguridad en cuanto a las duraciones de las sesiones, permitiendo una mejor restricción
Crear relación entre componentes y datos en el sistema de software	Model-View-Controller (MVC)	Como arquitectura general, MVC provee todo lo necesario para la comunicación de la información en vistas y clases que encapsulen lo que requiere cada tipo de usuario, respecto a los datos y reglas del negocio. Además, encaja perfectamente con los patrones de diseño antes mencionados.

Tabla 3: Selección de patrones.

5. Trade-offs entre tecnologías (actualizados)

Se establecerán comparaciones tecnológicas entre elementos principales de un software para cumplir con los requisitos nombrados a través de este informe.

Tecnología principal usada: Ruby on Rails

- Base de datos: MySQL
- Back-end: Ruby on Rails
- Front-end: Ruby on Rails (Ruby embebido)
- Diseño: Foundation 6

Alternativa: Stack MEAN

- Base de datos: MongoDB
- Back-end: NodeJS/ExpressJS
- Front-end: AngularJS
- Diseño: Bootstrap

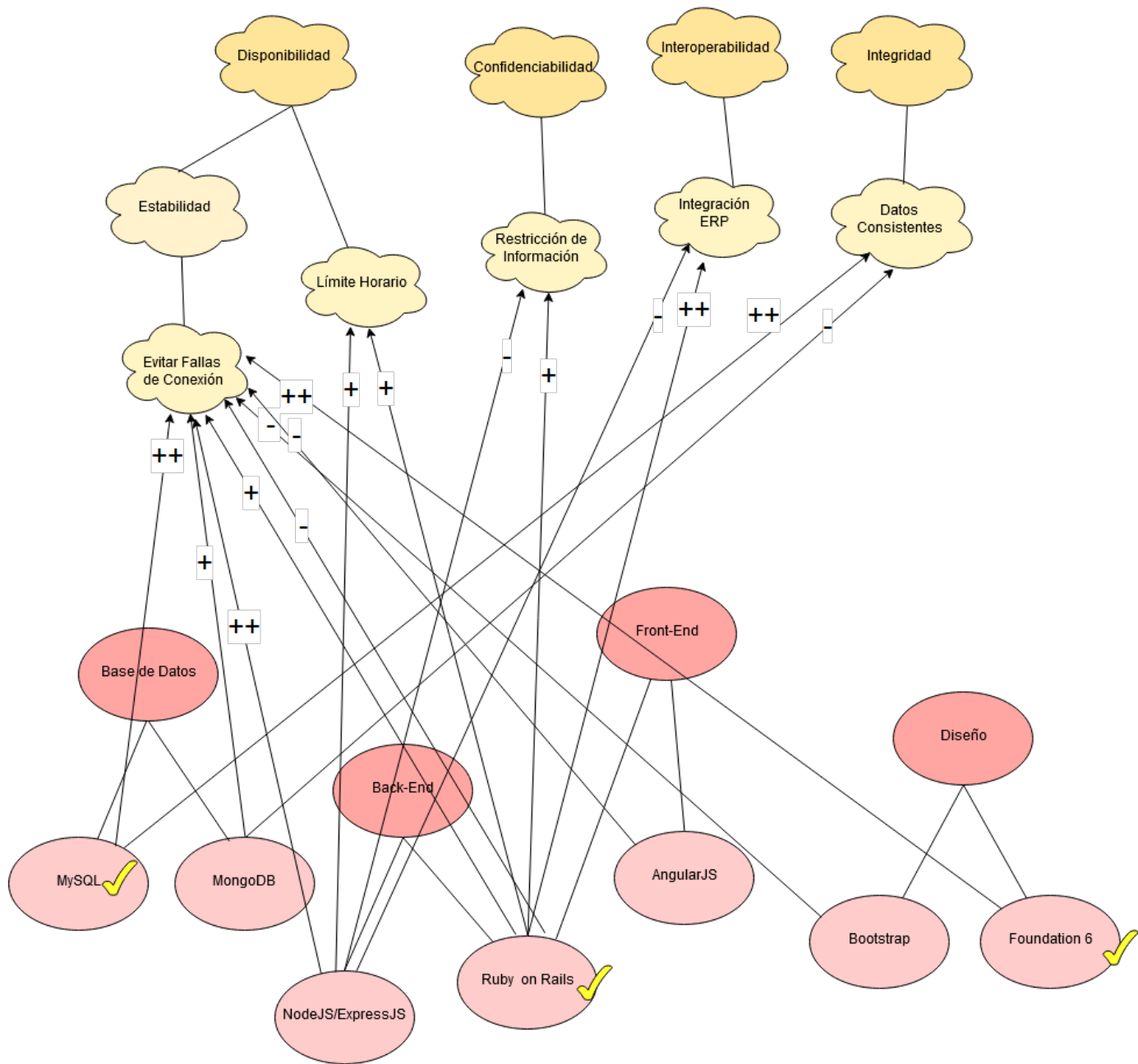


Figura 2: Softgoal Interdependency Graph (SIG).

Decisión	Softgoals	Evaluación	Razonamiento
Ocupar MySQL como gestor de la base de datos	Integridad	++	Soporta gran cantidad de datos y posee gran estabilidad, favoreciendo que los datos sean consistentes.
Ocupar MongoDB como gestor de la base de datos no relacional	Integridad	-	El cómo se almacenan los datos en la base de datos puede afectar negativamente la consistencia de los datos.
Utilizar Ruby on Rails como Back-end	Disponibilidad	+	Permite abstraer problemas de implementación rápida y frecuentemente, además de hacer el código mantenible.
Utilizar NodeJS como Back-end	Disponibilidad	+	Al ser JavaScript uno de los lenguajes más populares, se puede encontrar fácilmente soluciones a problemas de implementación.
Utilizar Ruby on Rails como Back-end	Interoperabilidad	++	Existe variedad de formas para manejar correctamente la información, permisos e integración con otros sistemas.
Utilizar NodeJS como Back-end	Interoperabilidad	+	La comunidad de NodeJS está en constante crecimiento, por lo que la base de conocimientos está extensamente disponible.
Utilizar Ruby on Rails como Back-end	Confidencialidad	+	Reutilizar código libre para mejorar la confidencialidad del sistema.
Utilizar NodeJS como Back-end	Confidencialidad	+	Ya existe código en JavaScript enfocado en mejorar la confidencialidad del sistema.
Back-end en NodeJS	Disponibilidad	++	Sistema de software web es más rápido.
Back-end en Ruby on Rails	Disponibilidad	+	Existen soluciones de alta disponibilidad para Ruby on Rails.
BD no relacional con MongoDB	Disponibilidad	+	Tiempo acotado entre falla primaria y recuperación.
BD con MySQL	Disponibilidad	++	Se puede implementar una solución de alta disponibilidad con MySQL Cluster.
Utilizar Bootstrap para el diseño de la página	Disponibilidad	-	Códigos no son precisamente ligeros.
Utilizar AngularJS como Front-end	Disponibilidad	-	Con gran cantidad de datos o de usuarios conectados a la vez, la interfaz usuaria puede sufrir tiempos de respuesta severamente lentos.
Front-end en Ruby on Rails	Disponibilidad	-	No asegura que el sistema de software web sea rápido.
Utilizar Foundation para el diseño de la Página	Disponibilidad	++	Al ser ligero posee un renderizado más rápido y eficiente en el consumo de internet.

Tabla 4: Trade-offs entre opciones tecnológicas.

6. Deuda técnica incurrida

Deuda técnica	Razonamiento	Impacto
Acceso, autenticación y autorización inseguro	Rápido y fácil de implementar; se pueden utilizar paquetes del framework para autenticación y autorización a cierta información según el tipo de usuario en una próxima iteración.	Inseguridad por acceso no autorizado a información sensible para GPI.
Conexión con API de Alegra para ERP	No hubo acceso para el ERP Laudus, por lo tanto se busca implementar un stub o ERP con una API similar (como ERP Alegra).	Problemas al tener registro necesario para facturas y estándares legales de contabilidad de GPI.
Creación de órdenes de compra incompleta	Implementación complicada solo por las autorizaciones para aprobación, se complica al registrar facturas de estas en ERP.	Sin la resolución de las órdenes, no se puede resolver una solicitud por no haber contacto con proveedor.
Diseño de vistas básico	Fácil y rápido de implementar.	Curva de aprendizaje lenta del usuario.

Tabla 5: Deuda técnica.