

**Módulo de Sustancias, materiales y desechos
peligrosos del Sistema Integrado de Gestión de la
Unidad de Laboratorios
Documento de la Arquitectura del Software**

Versión <2.0>

Historia de Revisión

Fecha	Versión	Descripción	Autor
29/01/2017	<1.0>	Versión inicial	Lambda
12/02/2017	<1.1>	Añadidos varios diagramas nuevos	Lambda
12/03/2017	<1.2>	Modificación de la descripción de los casos de uso	Lambda
31/01/2018	<2.0>	Actualización de la arquitectura	Samuel Arleo
04/04/2018	<2.1>	Actualización	Samuel Arleo

Tabla de Contenidos

1.	Introducción	4
1.1	Propósito	4
1.2	Alcance	4
1.3	Definiciones, Siglas, y Abreviaciones	4
1.4	Referencias	4
1.5	Vista Global	4
2.	Representación Arquitectónica	4
3.	Metas y Restricciones Arquitectónicas	5
4.	Vista de Casos de Uso	5
5.	Vista Lógica	12
5.1	Visión general	12
5.2	Paquetes de Diseño Significativos Arquitectónicamente	12
5.3	Realizaciones de los casos de uso	12
6.	Vista de Procesos	13
7.	Vista de Implantación	13
8.	Vista de Implementación	14
8.1	Vista General	14
9.	Vista de Datos	15
10.	Tamaño y Desempeño	16
11.	Calidad	16

Documento de la Arquitectura del Software

1. Introducción

1.1 Propósito

El rol del Documento de Arquitectura de Software (DAS) es describir de forma concisa los elementos que conforman el sistema. Representa una base de conocimiento que describe al sistema en un nivel de abstracción inferior en comparación con el documento de Especificaciones de Requerimientos de Software (ERS).

1.2 Alcance

El DAS abarcara la vista lógica, la vista de desarrollo, de proceso, vista física y vista de escenarios. Todas estas vistas pueden ser desplegadas a través del uso de la vista de casos de uso, el plan creativo, el diagrama de contexto y el documento ERS.

1.3 Definiciones, Siglas, y Abreviaciones

Todas las definiciones y vocabulario que puede presentarse en este documento están especificados en el Glosario del proyecto.

1.4 Referencias

- *Glosario.pdf* Clase 2.

1.5 Vista Global

El DAS busca dar una visión clara de los aspectos más relevantes del sistema. A continuación del documento se hace un análisis minucioso de las vistas que conforman el documento DAS.

2. Representación Arquitectónica

Este software está constituido por una base de datos, la cual es el pilar de nuestro sistema. A partir de esta se divide jerárquicamente el sistema dependiendo del rol que posea el usuario. Existen siete roles relevantes para el módulo de SMyDP: director, asistente del director, gestor, jefe de laboratorio, jefe de sección y técnico. Cada uno de estos, tiene acceso a la vista de la lista de sustancias de laboratorio, siendo únicamente el gestor, el encargado de modificarla. A su vez, cada uno de ellos tendrá una vista de inventario, bajo la cual se seleccionará el inventario del espacio físico que desee ver, según sus privilegios (de acuerdo a la dependencia a la que pertenece el usuario). En dicha tabla de inventarios, se posee la opción de acceder a la bitácora, bajo la cual se registran los usos y consumos de las sustancias para cada uno de los distintos espacios físicos.

En la vista de inventario y bitácoras se encuentran únicamente las tablas de los mismos, con la opción de editar, agregar o eliminar según sea el caso. Para la selección de la bitácora, el usuario contará con una vista de las secciones, laboratorios y espacios físicos disponibles, bajo la cual se redireccionará al inventario correspondiente.

3. Metas y Restricciones Arquitectónicas

El proyecto se encuentra en una segunda fase de desarrollo. En la primera, una versión inicial del sistema fue creada donde fueron implementadas diversas funcionalidades de las cuales se encuentran operativas las siguientes:

- Gestión de usuarios: Registro, autenticación, permisos, modificación y eliminación de usuarios.
- Gestión de personal: Listado del personal de la ULAB, consulta de usuarios
- Listado de sustancias (catálogo de sustancias): Ingreso al catálogo, modificación de sustancias, gestión de hojas de seguridad.

Otras funcionalidades desarrolladas que se encuentran medianamente operativas son:

- Gestión de inventarios: Consultas globales de inventarios (no operativa)
- Gestión de bitácoras: Ingreso y salida de materiales (medianamente operativas)
- Gestión de solicitudes y notificaciones: Emisión, modificación, eliminación, recepción, aceptación y certificación de solicitudes (no operativas)

Este hecho ha supuesto una restricción para el desarrollo pues es necesario adaptar las nuevas funcionalidades a las ya programadas en la primera versión, de la cual conviene utilizar todos los elementos operativos posibles mencionados, reprogramar los no operativos y diseñar los nuevos de manera que sean compatibles.

Además, el módulo de servicios, que junto con el módulo de SMyDP conforman el SIGULAB, ha sido desarrollado desde cero, reprogramando las funcionalidades operativas de la primera versión del módulo de SMyDP (excepto el catálogo de sustancias) por problemas de diseño de la base de datos del segundo. Por esta razón, otra restricción comprende la selección de las funcionalidades operativas que comparten ambos módulos.

Por lo tanto, la meta es combinar ambos desarrollos previos, creando un único sistema SIGULAB que saque provecho de todos los elementos funcionales posibles existentes.

4. Vista de Casos de Uso

La vista de casos de uso presenta el subconjunto del modelo de casos de uso arquitectónicamente significativos para el sistema. Describe el conjunto de escenarios que representan alguna funcionalidad central significativa. También describe el conjunto de casos de uso que tienen una cobertura arquitectónica sustancial (que implica muchos elementos arquitectónicos) o que acentúan o ilustran un punto específico y delicado de la arquitectura.

Los casos de uso que suponen un mayor impacto en la arquitectura del sistema vienen dados por la estructura jerárquica de las dependencias y el personal que conforman la ULAB. Las acciones que el usuario puede realizar y el contenido que le es mostrado depende de su cargo y la dependencia de la cual forma parte. La jerarquía de dependencias tiene en su punto más alto a la dirección (única unidad de adscripción que no depende de otra hasta los momentos), luego es seguida por los laboratorios (que pueden estar adscritos a otros laboratorios), quienes a su vez están conformados por secciones. Finalmente, los espacios físicos que componen las secciones son el nivel más bajo de la jerarquía de dependencias.

El personal también posee una jerarquía determinada por la dependencia en la cual labora dicha persona.

CARGO	DEPENDENCIA
Director de la ULAB (quien posee todos los privilegios)	Dirección
Gestor	Dirección
Asistente del director (quien posee los mismos permisos del director pero que debe solicitar aprobación para hacer cambios en el sistema)	Dirección
Jefe de laboratorio	Laboratorios
Jefe de secciones	Secciones
Técnico	Espacios físicos

Algunos de los casos de uso presentados son generalizaciones de los casos de uso presentes en el documento ERS (aquellos que contienen la palabra “gestión”) y comprenden acciones tales como agregación, modificación y eliminación. Esto fue hecho para evitar la repetición de casos de uso que poseen un mismo impacto en la arquitectura.

Por otra parte, se asume que el personal de una dependencia en un determinado nivel de la jerarquía podrá realizar todas las acciones que puede llevar a cabo el personal de las dependencias que tiene a cargo. En la siguiente tabla, el técnico es descrito como el actor que puede gestionar solicitudes de sustancias en espacios físico. Sin embargo, no sólo puede hacerlo un técnico, sino el personal de las dependencias superiores a las cuales rinde cuentas el técnico. Además, un jefe de sección puede gestionar la bitácora de los espacios físicos que tiene a cargo, pero no de los espacios físicos que pertenecen a otras secciones de las cuales no esté a cargo.

<i>Caso de Uso</i>	<i>Actor</i>
Consultar inventario	Usuario
Iniciar sesión	Usuario
Gestionar bitácora	Técnico
Gestionar solicitudes	Técnico
Gestionar lista de SMyDP	Gestor
Manejar Usuarios	Director

Caso de uso: Consultar inventario	
<u>Descripción:</u> Permite a los usuarios consultar el inventario de sustancias y materiales de la ULAB (si el usuario pertenece al personal de la dirección), de laboratorios (para el personal del laboratorio), de secciones (para el personal de la sección) y espacios físicos.	
<u>Precondición:</u> Acceder a la página del sistema.	
FLUJO BÁSICO: <ul style="list-style-type: none"> • El usuario ingresa al sistema. • El sistema muestra la vista inicial de la interfaz web. • El usuario consulta el inventario de la unidad, laboratorio, sección o espacios físicos en los cuales posee los permisos necesarios • El sistema lista todos los materiales con sus respectivos datos 	
ACTOR	SISTEMA
1. Ingresa al sistema.	2. Muestra la interfaz web inicial.
3. Consulta el inventario de la unidad, de un laboratorio, sección o espacio físico	4. Muestra las cantidades totales de cada sustancia y material en la dependencia seleccionada
<u>Postcondición:</u> Ninguna	
<u>Requerimientos especiales:</u> Ninguno.	
<u>Casos de uso relacionados:</u> Ninguno.	
Impacto en la arquitectura: Se utilizará una referencia a la dependencia padre (lista de adyacencia) para poder recorrer la jerarquía de dependencias desde los espacios físicos hasta la dependencia raíz (Dirección de la ULAB). Además, dependiendo del cargo del usuario, se pueden consultar determinados inventarios.	

Caso de uso: Iniciar sesión	
<u>Descripción:</u> Este caso de uso se refiere al ingreso del sistema por parte de un Usuario Normal, para lo cual se le pedirá que ingrese su correo electrónico y su contraseña.	
<u>Precondición:</u> El correo del usuario debe estar registrado en la tabla de usuarios del administrador de usuarios.	
FLUJO BÁSICO: <ul style="list-style-type: none"> • Si el usuario accede al sistema exitosamente, ve la pantalla principal que le permite ingresar en cualquiera de los diferentes módulos. • Si el ingreso no es validado, se le carga otra vez la página con los campos vacíos. El usuario tiene la opción de registrarse en el sistema.	
ACTOR	SISTEMA
1 Ingresa su correo y contraseña	2 Determina si los datos son correctos
3 Entra al sistema o debe volver a ingresar sus datos	

<u>Postcondición:</u> Si los datos son correctos, entonces debe cargarse la página principal del sistema. Si los datos son incorrectos, entonces la página de ingreso se carga de nuevo.
<u>Requerimientos especiales:</u> Para que el usuario pueda registrarse, es necesario que el administrador de usuarios lo haya añadido en su tabla de usuarios.
Impacto en la arquitectura: Solo el director puede gestionar los usuarios. Por esto se agregó la figura de director, quien cuenta con los mismos permisos del gestor, pero que adicionalmente puede gestionar los usuarios del sistema.

Caso de uso: Gestionar bitácora	
<u>Descripción:</u> Este caso de uso se refiere al uso y administración de la bitácora que posee cada sustancia en los diferentes espacios físicos de los laboratorios. En ella se pueden consultar, agregar y modificar los registros que posee.	
<u>Requerimiento:</u> SIGULAB.2	
<u>Precondición:</u> El usuario debe estar autenticado en el sistema y debe tener como mínimo el privilegio de Técnico. Las sustancias deben estar registradas en el inventario.	
FLUJO BÁSICO: El técnico entra en la bitácora de una sustancia específica. Allí Introduce: fecha, cantidad y si se usó, si ingresó o si fue un egreso. El sistema lo registra y guarda en la base de datos. El técnico puede consultar registros previos. El técnico puede modificar registros previos.	
ACTOR	SISTEMA
1 Solicita bitácora de una sustancia	2 Devuelve los registros de la bitácora
3 Introduce los datos de un registro nuevo	4 Guarda en la base de datos
5 Modifica registros existentes	6 Guarda los cambios en la base de datos y notifica al jefe de sección
<u>Postcondición:</u> La bitácora va a tener un nuevo registro asociado a la sustancia. Se modifica un registro existente de alguna sustancia. El sistema muestra los registros de una sustancia.	
<u>Requerimientos especiales:</u> En caso de compra se desea guardar el número de la factura relacionada.	
<u>Casos de uso relacionados:</u> Agregar registro, Eliminar registro, Modificar registro	
Impacto en la arquitectura: Aunque el jefe de sección y el técnico tengan los mismos privilegios en lo que corresponde a la bitácora de sustancias, el jefe de sección debe ser notificado de todos los cambios a la bitácora que haga el personal que tiene a su cargo. Por esta razón se ha diferenciado entre estos dos actores	

Caso de uso: Gestionar lista de SMyDP
<u>Descripción:</u>

Este caso de uso se refiere a la modificación de la lista de las sustancias, materiales y desechos peligrosos de la Unidad de Laboratorios por parte del Técnico.	
Requerimiento: SIGULAB.4	
Precondición: El usuario debe estar autenticado en el sistema y debe tener como mínimo el privilegio de Técnico.	
FLUJO BASICO: El usuario entra a la lista de SMyDP El Sistema presenta una lista con las sustancias registradas, mostrando sus campos. El usuario ingresa los cambios (agrega un nuevo materia o modifica o elimina uno existente) El sistema guarda los cambios	
ACTOR	SISTEMA
1 Entra a la lista de SMyDP	2 Devuelve la lista de sustancias registradas
3 Ingresa los cambios (agrega un nuevo materia o modifica o elimina uno existente)	4 Guarda los cambios
Postcondición: El Sistema guarda los cambios hechos	
Requerimientos especiales: Ser gestor	
<u>Casos de uso relacionados:</u> Consultar lista de SMyDP	
Impacto en la arquitectura: Nuevamente la jerarquía del personal que hará uso del sistema determina la ejecución de este caso de uso. Solo el gestor puede hacer cambios al listado de SMyDP	

Caso de uso: Gestionar usuarios	
<u>Descripción:</u> Este caso de uso se refiere al uso y administración de la lista de usuarios que pueden acceder al sistema, así como de los permisos y privilegios que ellos poseen.	
<u>Requerimiento:</u> SIGULAB.4	
<u>Precondición:</u> El usuario debe estar autenticado en el sistema y debe tener el privilegio de Administrador de Usuarios.	
FLUJO BÁSICO: El Administrador de Usuarios entra en la lista de Usuarios. Se muestra una lista con todos los usuarios registrados y sus privilegios. El Administrador puede: añadir un usuario nuevo, eliminar uno existente o editar sus privilegios. El Sistema guarda los cambios correspondientes.	
ACTOR	SISTEMA
1 Solicita lista de usuarios y privilegios	2 Devuelve los usuarios registrados
3 Puede: añadir, editar o eliminar usuario	4 Guarda en la base de datos los cambios correspondientes
<u>Postcondición:</u> Se añade un nuevo usuario a la lista. Se modifica o elimina un usuario existente. Se modifican los privilegios de un usuario registrado.	
<u>Casos de uso relacionados:</u> Ninguno	

Impacto en la arquitectura: La distinción entre el gestor y el director ha sido necesaria debido al registro de usuarios en el sistema, que debe ser manejada por el director.

5. Vista Lógica

Se explicarán las distintas divisiones del sistema, así como una breve explicación de las funcionalidades de cada una de ellas, incluyendo los distintos escenarios que tendrá el usuario al hacer uso de dicho software.

5.1 Visión general

El módulo de SMyDP ha sido dividido en diversos paquetes que pueden ser apreciados en el modelo de dominio presentado más abajo. Estos paquetes son: bitácoras e inventarios (en rojo), solicitudes y notificaciones (en verde), usuarios y personal (en morado), sustancias (en amarillo) y dependencias (en azul). Se ha hecho esta división lógica para separar las funcionalidades independientes y buscar la modularidad del sistema.

5.2 Paquetes de Diseño Significativos Arquitectónicamente

Bitácoras e inventario:

Descripción:

Contiene la representación del inventario de un espacio físico y las bitácoras de las sustancias en ese inventario. Además define las acciones de adición, eliminación y modificación de registros de las bitácoras así como los procedimientos para calcular la existencia actual de sustancias a nivel de espacios físicos y a nivel de dependencias por medio del recuento de las existencias del mismo material en varias dependencias.

Comunicación con otros módulos:

Se comunica con el módulo de Sustancias y el módulo de Dependencias para obtener datos de los materiales que conforman el inventario y las dependencias a las que corresponden estos.

Solicitudes y notificaciones (por desarrollar):

Descripción:

Se subdivide en solicitudes y notificaciones. La primera se encarga de manejar el flujo de las solicitudes: emisión, eliminación, aceptación, préstamos activos (cuando se presta un material), cancelación de deudas, certificaciones de entrega y recepción del material. La segunda comprende el aviso a los actores involucrados cuando se emite una solicitud (al jefe de sección), cuando se rechaza (al gestor), cuando se elimina (a las secciones con excedente y al gestor), cuando se certifica la entrega del material (al proveedor). Además contiene

notificaciones por otros motivos, como las que se envían al director cuando el asistente hace cambios y las del gestor cuando se modifica el registro de la bitácora de una sustancia.

Comunicación con otros módulos:

Se comunica con el módulo de Bitácoras e inventarios para hacer el ingreso o egreso de sustancias cuando se certifica la recepción o salida de un material de un espacio físico por una solicitud. Usuarios y personal para determinar las acciones que puede realizar el usuario de acuerdo a su cargo y el contenido que puede consultar. También existe comunicación con el módulo de personal, por el envío de notificaciones a los actores involucrados de acuerdo a la sección que emite la solicitud y las que poseen excedente. Finalmente, es necesaria la comunicación con el módulo de Dependencias para conocer cuáles poseen excedente de la sustancia solicitada y quien es el personal que debe ser informado de la transacción.

Usuarios y personal:

Descripción:

Este paquete contiene la representación de los usuarios (utilizadores del sistema) y el personal (miembros de la ULAB). Define la jerarquía del personal de la ULAB así como los privilegios de cada usuario. Además contiene la lógica del registro y aprobación de usuarios por parte del director.

Comunicación con otros módulos:

Se comunica con todos los módulos para ejecutar acciones definidas en estos, como la modificación del catálogo de usuarios o la emisión de solicitudes.

Sustancias:

Descripción:

Este paquete contiene la representación de las sustancias y materiales (ambas representadas como sustancias), el listado de sustancias y las funcionalidades que permiten el registro, eliminación y modificación de una sustancia y manejo de las hojas de seguridad por parte de los gestores de la ULAB, quienes poseen los permisos necesarios para realizar estas acciones.

Comunicación con otros módulos:

Se comunica con el módulo de Usuarios y personal para determinar las acciones que puede realizar el usuario de acuerdo a su cargo y el contenido que puede consultar.

Dependencias:

Descripción:

Define los espacios físicos y dependencias así como la jerarquía de las unidades de adscripción que compone la ULAB.

Comunicación con otros módulos:

Se comunica con todos los módulos que tienen alguna relación con las dependencias para ejecutar acciones como la agregación de los materiales existentes en los espacios físicos para la consulta de inventarios.

5.3 Realizaciones de los casos de uso

Las jerarquías del personal y de las dependencias de la ULAB han sido consideradas en el diseño, lo que facilita la ejecución correcta de los casos de uso. Dependiendo de los permisos que le hayan sido asignados a un usuario por el encargado de manejo de personal, este usuario podrá realizar acciones que afecten únicamente a la dependencia a la cual forma parte y podrá ver contenido de acuerdo a su cargo.

Diagrama de Clases

6. Vista de Procesos

El módulo de SMyDP debe permitir la realización de los procesos “Registro de usuarios”, “Solicitud de sustancia”, “Registro de movimiento en la bitácora de una sustancia” y Consulta del inventario de un espacio físico, dependencia o dirección” pues representan las principales funcionalidades del módulo.

- Proceso: Registro de usuarios.

Descripción:

Este proceso comprende el registro de nuevos usuarios en el sistema y varía dependiendo de si el usuario forma parte del personal de la ULAB o no.

Flujo:

1. Usuario suministra datos personales, rol y dirección de correo
 - 1.1. Si el rol es de técnico, se permite seleccionar uno o más espacios físicos
 - 1.2. Si el rol es de jefe de sección, se permite seleccionar una sección
 - 1.3. Si el rol es de jefe de laboratorio, se permite seleccionar un laboratorio
 - 1.4. Si el rol es de gestor, director o asistente, solo se puede seleccionar como dependencia la Dirección

- Proceso: Solicitud de sustancia.

Descripción:

Comprende la solicitud de una nueva sustancia por parte de una sección a otras con excedente. El proceso involucra la emisión, modificación, eliminación, recepción y aceptación de la solicitud.

Flujo:

1. Técnico solicita una cantidad X de una sustancia. Esta puede ser modificada o eliminada antes de ser aceptada.
 - 1.1. Jefes de secciones con excedente de la sustancia reciben la notificación de una nueva solicitud.
 - 1.1.1. Si la solicitud es aceptada:
 - 1.1.1.1. Se selecciona la cantidad a enviar y los términos de la cesión de la sustancia (préstamo o cesión)
 - 1.1.1.1.1. Si el término es préstamo, se establece la fecha de devolución
 - 1.1.1.1.2. Si el término es cesión no hace falta suministrar más datos
 - 1.1.1.2. Se notifica al solicitante sobre la aceptación de la solicitud
 - 1.1.1.3. Se realiza el envío físico de la sustancia
 - 1.1.1.4. El solicitante certifica la recepción y se agrega el registro a la bitácora (ver proceso “Registro de movimiento en la bitácora”)
 - 1.1.1.5. El suministrador certifica el envío y se agrega el registro a la bitácora
 - 1.1.2. Si no es aceptada, se envía notificación al gestor de la Dirección acerca de la negación de la solicitud.

- Proceso: Registro de movimiento en la bitácora de una sustancia.

Descripción:

Comprende el registro del ingreso o salida de una cantidad determinada de una sustancia.

Flujo:

1. Usuario intenta registrar un nuevo movimiento en la bitácora de una sustancia:
 - 1.1. Si la sustancia no aparece en el listado de sustancias, debe ser comunicado al gestor para que la agregue (ya que el listado funciona como un catálogo y no es posible seleccionar sustancias diferentes a las ya definidas)
 - 1.2. Si la sustancia aparece en el listado, el usuario selecciona la cantidad y tipo de operación (ingreso o salida)
 - 1.2.1. Si el tipo de operación es ingreso, se especifica el origen (almacén o compra)
 - 1.2.1.1. Si el origen es una compra, se incluye el número de factura
 - 1.2.1.2. Si no, se indica el almacén
 - 1.2.2. Si el tipo de operación es salida, se especifica el motivo (extensión, investigación, enseñanza -académico-, cesión o préstamo)
 - 1.2.2.1. Si el motivo es préstamo, se indica la fecha de devolución
 - 1.2.2.2. Si el motivo es extensión, se indica la empresa a la cual se presta el servicio
 - 1.2.2.3. Si el motivo es investigación, se indica el proyecto
 - 1.2.2.4. Si el motivo es enseñanza, se especifica la asignatura
 - 1.2.2.5. Si el motivo es cesión, se indica la sección receptora

- Proceso: Consulta del inventario de un espacio físico, dependencia o de la ULAB

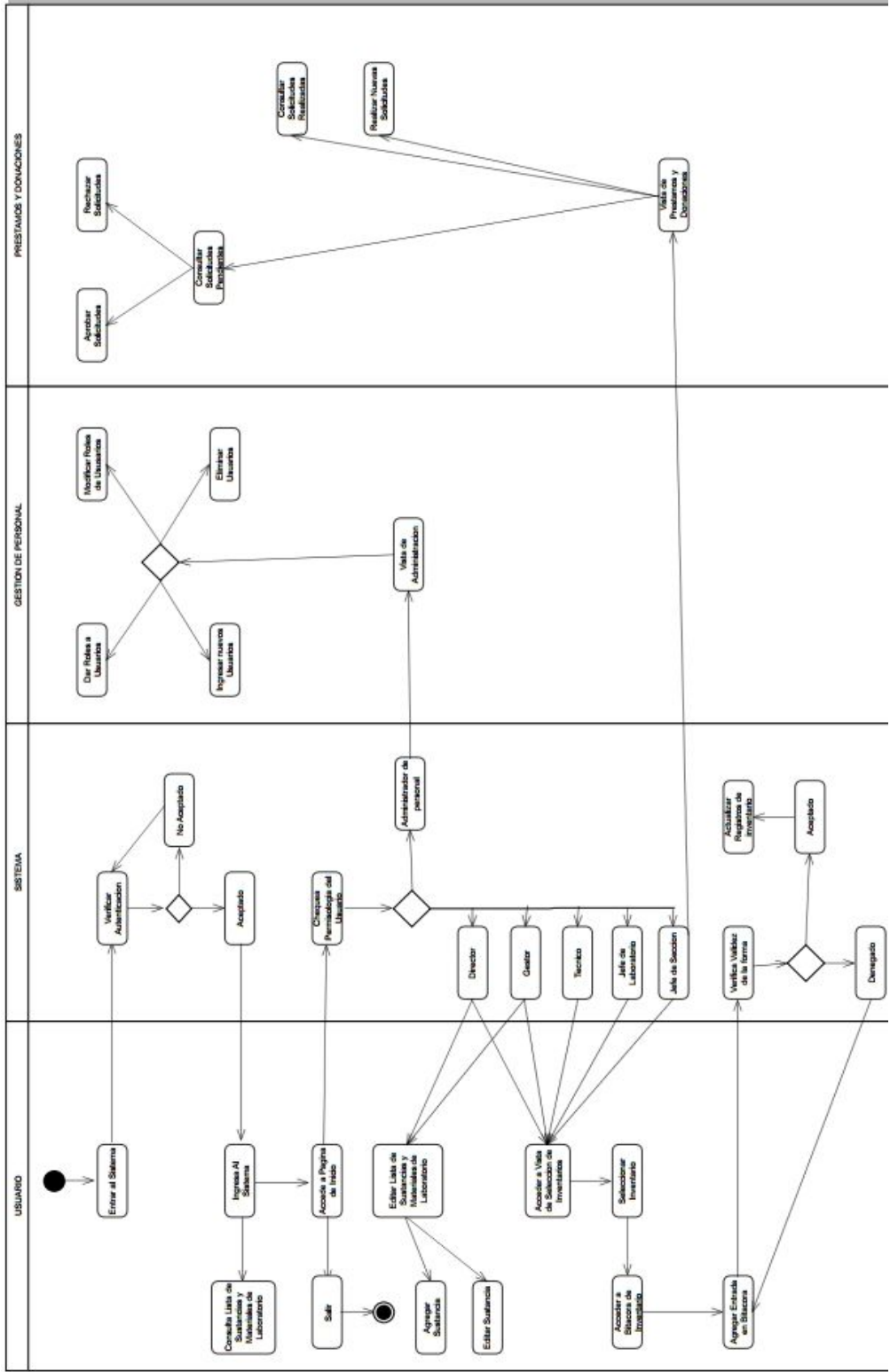
Descripción:

Comprende el listado de los materiales existentes en un espacio físico, dependencia o la ULAB mostrando sus cantidades existentes y el excedente.

Flujo:

1. Usuario consulta el inventario de una dependencia o espacio en el cual posee permisos suficientes para realizar dicha acción.
 - 1.1. Para cada sustancia en los espacios físicos consultados (o pertenecientes a la dependencia o unidad consultada)
 - 1.1.1. Se toma del inventario la existencia y el excedente
 - 1.1.2. Se agregan estos valores a las cantidades de las demás secciones (en caso de ser una consulta de inventario de unidad o dependencia)
 - 1.2. Se generan inventarios a nivel de secciones y se repite el proceso de forma recursiva hasta llegar a la dependencia consultada.

Modelo de actividad



7. Vista de Implementación

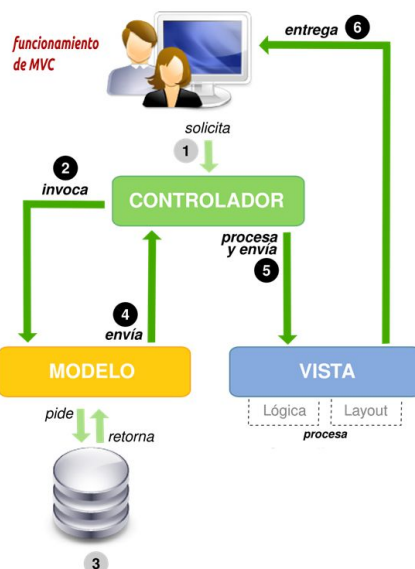
7.1 Vista General

El sistema posee una arquitectura en capas debido al uso del patrón MVC (Modelo-Vista-Controlador) dentro de la cual se establecen:

- Las Vistas como la interfaz web del usuario, la capa más superficial pero no menos importante del sistema. Toda la interacción directa con el mismo se realizará a través de ésta, y es por eso que resulta importante la comodidad. Se buscó que dentro de nuestro sistema se incluyeran los requisitos y el estilo de las páginas web de la Universidad Simón Bolívar, así como la distribución correcta de la información, evitando sobrecargar al usuario o hacer pesada la interacción visual. Se comunica con los Controladores. Y se relaciona directamente con el usuario.

- Los Controladores, como la capa de programación que se encarga de regular todos los procesos y producir el flujo de la aplicación. En este caso se utilizó el lenguaje de programación Python y el framework Web2py. Este lenguaje posee grandes ventajas dentro del mundo del desarrollo de sistemas, como puede ser que es débilmente tipado y muy flexible a la hora de realizar cambios rápidos en el sistema, cualidad necesaria en este tipo de proyectos que serán expandidos a futuro.

- Los Modelos, como la base de datos a utilizar dentro del sistema. En éste caso se trabaja con PostgreSQL debido a su compatibilidad con Web2py. Este framework ofrece una capa de abstracción de la base de datos para hacer consultas y operaciones a un alto nivel, generando por sí solo el código SQL.



8. Vista de Datos

En esta vista se presentará una descripción de las principales decisiones de diseño que determinaron el modelo de datos que puede ser consultado en el diagrama ERE y las dependencias entre algunas de las tablas.

Descripción del modelo de datos:

El diseño del modelo de datos ha sido determinado por las siguientes decisiones de diseño:

- La bitácora es una tabla de registros llamada “Bitacora”. Es una entidad que almacena los cambios en las entradas de la tabla “Inventario” (entidad débil que se identifica con las tablas Sustancia y Espacio Físico).
- El proveedor representa la entidad que surte el material que está siendo ingresado en la bitácora. Esta entidad se especializa en “Dependencia” (cuando el proveedor es una dependencia que ha prestado o cedido el material), “Institución proveedora” (cuando la sustancia proviene de una empresa) y “Espacio físico” (cuando se trata de un almacén). Esto busca que cada registro en la bitácora tenga asociado el identificador de la entidad proveedora, el tipo de la entidad y su nombre, brindando detalles y generando compatibilidad con futuros módulos de compras, proyectos de investigación o asignaturas y docentes.
- El consumidor representa la entidad que recibe o hace uso del material. Esta entidad se especializa en “Tesis” (cuando es un proyecto de investigación), “Dependencia” (cuando se trata de una sección que ha emitido una solicitud), “Materia” (cuando es utilizada con fines académicos) e “Institución Cliente” (cuando es utilizada por extensión). Luego, el registro tendrá asociado el nombre de la entidad consumidora, su identificador y el motivo del consumo del material.
- Las entidades “Institución proveedora” y “Institución cliente” podrían ser generalizadas en “Compañía” de ser necesario. Esto no ha sido incluido en el diagrama para no dificultar su entendimiento y porque esto simplemente se traduce en una misma tabla para las instituciones. Cada institución será referenciada como proveedora o cliente de acuerdo a las necesidades.
- El listado o catálogo de sustancias ha sido representado por la entidad “Sustancia”, en lugar de llamarse inventario explícitamente.
- Se ha modelado la jerarquía de dependencias con una lista de adyacencias donde cada nodo apunta al padre, en este caso a su unidad de adscripción. Para esto se agregó la relación reflexiva “Adscribe” (referencia a la misma tabla). Es importante destacar que la dirección es la única entidad que no posee un valor en la lista de adyacencias por no estar adscrita a otra unidad dentro de la ULAB.
- No se ha incluido explícitamente la jerarquía de usuarios. Los datos del personal han sido resumidos en la entidad “Personal”. Los del usuario del sistema son almacenados en la entidad “AUTH_USER”, generada por Web2py (un usuario no tiene por qué aparecer en ambas tablas). Luego, los privilegios del usuario son obtenidos por medio de la tabla “AUTH_MEMBERSHIP” (“Tiene_rol” en el diagrama ERE para facilitar la comprensión), que mapea usuarios y grupos (equivalentes a roles o cargos).
- Más detalles de las relaciones “Realiza_trabajo”, “Enseña” y “Tiene_a_cargo” pueden encontrarse en el DAS del módulo de servicios.
- Las entidades “Materia” y “Tesis” corresponden a las asignaturas y proyectos de investigación respectivamente, que podrían hacer uso de las sustancias de los espacios físicos o de los servicios.

Diagrama Entidad Relación Extendido



Diccionario de datos:

- Entidades

Sustancia

Dato	Tipo	Descripción	Relación con otros datos
nombre	string	Nombre de la sustancia	N/a
cas	string	Número identificador de la sustancia	N/a
pureza	int	Rango 0-100. Define la pureza de la sustancia	N/a
estado	int	Representa el estado de la sustancia.	Tabla estado
control	int	Representa el nivel de control de la sustancia.	Tabla reginemenes
peligrosidad	string	Representa la peligrosidad de la sustancia	N/a
presentacion	string	Unidad en la que se mide la cantidad de sustancia	N/a
hoja_de_seguridad	string	Link hacia el archivo de la hoja de seguridad de la sustancia	N/a

Unidad de medida:

Dato	Tipo	Descripción	Relación con otros datos
id	int	Identificador de la unidad de medida	N/a
nombre	string	Nombre de la unidad	N/a

abreviatura	string	Unidad abreviada (ml, l, kg, g, etc.)	N/a
-------------	--------	---------------------------------------	-----

Dependencia:

Dato	Tipo	Descripción	Relación con otros datos
nombre	string	Nombre del laboratorio	N/a
id	int	Identificador numérico de la dependencia	N/a
email	string	Correo institucional de la dependencia	N/a
pagina web	string	Sitio web de la dependencia	N/a
extension	string	Contacto de la dependencia	N/a
extension interna	string	Contacto interno de la dependencia	N/a
id_jefe	int	Id del jefe de laboratorio	Tabla de autenticación de usuario

Espacio físico

Dato	Tipo	Descripción	Relación con otros datos
codigo	string	El espacio	N/a
dirección	string	Dirección del espacio	N/a
nombre	string	Nombre del espacio físico	N/a
uso	string	Descripción de lo que se hace en el espacio físico	N/a

dependencia	int	Unidad de adscripción del espacio físico	Tabla Dependencia
técnico	int	Id del técnico(s) del espacio.	Tabla de Personal

Inventario

Dato	Tipo	Descripción	Relación con otros datos
id_sustancia	string	Id de la sustancia	Tabla Sustancia
id_registro	integer	Id del	Tabla sustancias
excedente	float	Cantidad disponible para donación	N/a
total	float	Cantidad total de la sustancia	N/a
id_espacio_fisico	int	Id del espacio físico de la sustancia	Tabla Espacio físico

Bitácora

Dato	Tipo	Descripción	Relación con otros datos
id	int	Identificador del movimiento en la bitácora	N/a
id_sustancia	string	Id de la sustancia	Tabla Sustancia
excedente	float	Cantidad disponible para donación	N/a
total	float	Cantidad total de la sustancia	N/a
id_espacio_fisico	int	Id del espacio físico de la sustancia	Tabla Espacio físico
descripción	string	Motivo de entrada o	N/a

		salida	
--	--	--------	--

Personal

Dato	Tipo	Descripción	Relación con otros datos
c.i.	string	Cedula de identidad	N/a
nombre	string	Nombre del empleado	N/a
apellido	string	Apellido del empleado	N/a
cargo	string	Cargo en la ULAB	N/a
telefono	string	Telefono personal del empleado	N/a
categoria	string		N/a
estatus	string	Registrado o por aprobar	N/a
email	string	Correo del empleado	N/a
pagina_web	string	Página web personal	N/a

Solicitud

Dato	Tipo	Descripción	Relación con otros datos
id	int	Id de solicitud	N/a
id_sustancia	int	Id de la sustancia solicitada	Tabla de Sustancia
cantidad	float	Cantidad solicitada	N/a
descripcion	string	Descripcion de la solicitud	N/a
fecha_caducidad	timestamp	Descripcion de la	N/a

		solicitud	
caducada	bool	Verdadero si ya paso la fecha limite para satisfacer la solicitud	N/a
id_solicitante	int	Id del usuario creador de la solicitud	Tabla de Personal
id_personal_eliminator	int	Id del usuario que elimina la solicitud	Tabla de Personal

Proveedor

Dato	Tipo	Descripción	Relación con otros datos
id	string	Id del proveedor	N/a
Nombre	string	Nombre del proveedor	N/a

Consumidor

Dato	Tipo	Descripción	Relación con otros datos
id	string	Id del consumidor	N/a
Nombre	string	Nombre del consumidor	N/a

- Relaciones

Modifica (Personal *Modifica* Bitacora)

Dato	Tipo	Descripción
autorizado	bool	Verdadero si ya fue autorizado por el jefe de sección. En ese caso se registran los cambios
descripcion	string	Descripción
operacion	string	Modificar o eliminar
cantidad_anterior	float	Cantidad antes de la modificación
fecha	string	Fecha de modificación

Modifica (Personal *Modifica* Bitacora)

Dato	Tipo	Descripción
autorizado	bool	Verdadero si ya fue autorizado por el jefe de sección. En ese caso se registran los cambios
descripcion	string	Descripción
operacion	string	Modificar o eliminar
cantidad_anterior	float	Cantidad antes de la modificación
fecha	string	Fecha de modificación

Puede_satisfacer (Espacio fisico X de la Dependencia Y y Jefe Z *Puede_satisfacer* Solicitud)

Dato	Tipo	Descripción
existencia	float	Cantidad disponible en el espacio físico
id_unidad_adscripcion	int	Id de la unidad de adscripción con excedente

Responde (Personal *Responde* Solicitud)

Dato	Tipo	Descripción
estado	float	Rechazada o aceptada
fecha_aceptacion	timestamp	Fecha de aceptación
fecha_confirmación_recepción	timestamp	Fecha de confirmación de la recepción
fecha_confirmación_entrega	timestamp	Fecha de confirmación de la entrega

Elimina (Personal *Elimina* Solicitud)

Dato	Tipo	Descripción
descripcion	string	Motivos de la eliminación
fecha	timestamp	Fecha de eliminación

Provee_cantidad (Proveedor *Provee_cantidad indicada* en el registro de la Bitácora)

Dato	Tipo	Descripción
tipo	string	Cesión, préstamo, almacén, compra

Recibe_cantidad (Consumidor *Recibe_cantidad* indicada en el registro de la Bitácora)

Dato	Tipo	Descripción
tipo	string	Investigación, extensión, cesión, préstamo, uso académico

9. Tamaño y Desempeño

Hasta el momento, con lo que llevamos desarrollado de nuestro proyecto no conocemos con detalle las restricciones de desempeño, como por ejemplo, rendimiento, tiempos de respuesta, entre otros; así como tampoco las características principales de dimensionamiento del software. Sin embargo,

este documento se irá actualizando a medida que vaya avanzando el sistema para poder completar este tipo de información.

10. Calidad

Una de las cualidades más importantes de nuestro sistema se lo otorga el desarrollo realizado a través de las herramientas de control de versiones (GitHub), y principalmente del Framework Web2py. Las capacidades más importantes del sistema se resumen en los siguientes puntos:

- **Extensibilidad:** El sistema es fácilmente extensible gracias a Web2py, además de los mismos requisitos del diseño ya que éste está planificado de manera tal que sirva como base para elaborar más módulos referentes a la unidad de Laboratorios con gran facilidad.

- **Confiabilidad:** Tanto la documentación como la elaboración de un código robusto basado en casos de uso, pruebas constantes, el carácter cíclico del desarrollo y la implementación de herramientas profesionales y empresariales proporcionan un alto grado de confiabilidad a nuestro sistema. En el peor de los casos la plantilla de riesgos está junto con los procedimientos recomendados en caso de que exista un problema externo al software que pueda afectar al sistema.

- **Portabilidad:** El sistema puede ser implantado en otro servidor de ser necesario, además de poder ser ejecutado desde cualquier sistema operativo basado en Linux o incluso Windows debido a la disponibilidad del framework para ambos.

- **Funcionalidad:** El objetivo de nuestro sistema es cumplir las expectativas del cliente, desarrollando un prototipo completamente funcional y sentar las bases de los próximos módulos a implementar. Para eso además se realiza el desarrollo de una interfaz amigable al usuario, cuidando siempre los requerimientos de estilo, de funcionalidad y de calidad del trabajo para entregar un software integral.