

Arquitectura del Sistema

Horizon

PSI Group



La Arquitectura del software, comprende el conjunto de elementos estáticos, propios del diseño intelectual del sistema, que definen y dan forma tanto al código fuente, como al comportamiento del software en tiempo de ejecución.

Naturalmente este diseño arquitectónico ha de ajustarse a las necesidades y requisitos del proyecto. Este documento describe en términos generales, las ideas principales detrás de la arquitectura escogida para el mismo.



Tabla de contenido

Introducción	4
<i>Propósito</i>	<i>4</i>
<i>Alcance</i>	<i>4</i>
<i>Referencias</i>	<i>5</i>
Representación Arquitectónica	5
<i>Representación</i>	<i>6</i>
Objetivos Arquitectónicos y Restricciones	7
<i>Objetivos Generales</i>	<i>7</i>
<i>Objetivos Específicos</i>	<i>7</i>
<i>Descripción de Procesos</i>	<i>8</i>
Vista de Caso de Uso	8
<i>Descripción de los Actores</i>	<i>9</i>
<i>Contexto del sistema</i>	<i>10</i>
Vista Lógica	11
<i>Paquetes de Diseño significativos para la arquitectura</i>	<i>11</i>
Vista de Procesos	12
Otras Características	15
<i>Características arquitectónicas del sistema</i>	<i>15</i>

Arquitectura del Sistema

Introducción

El presente documento provee una vista de alto nivel de la arquitectura del Sistema en desarrollo, los objetivos y restricciones, los casos de uso significativos, los patrones de arquitectura aplicados y las principales decisiones de diseño.

Este documento da una vista general del resto de los artefactos generados en el proceso de desarrollo.

Propósito

Este documento de arquitectura de software (por sus siglas en inglés, SAD) tiene como propósito brindar una visión comprensible de la arquitectura general, utilizando diferentes vistas de la arquitectura para ilustrar diferentes aspectos del sistema. Captura las decisiones más importantes en lo que respecta a la arquitectura del sistema que fueron tomadas en el proyecto.

Este SAD está dirigido a distintos tipos de actores involucrados en el desarrollo del proyecto, tales como Desarrolladores, Tutores y clientes.

Los desarrolladores puede utilizar este documento como base para la documentación del desarrollo de productos de software en proyectos de diferente porte.

El tutor puede tomar este documento como base para mostrar la importancia de la arquitectura en el desarrollo de software así como el rol del arquitecto de software.

Es de nuestro interés intercambiar con ellos ideas y resultados.

Desde el punto de vista de un desarrollador, este documento le brindará, con certeza, una buena razón para darle a la arquitectura de software la importancia que tiene en todo proyecto de desarrollo.

Alcance

Este SAD profundiza principalmente en las vistas de casos de uso y lógica, incluyendo algunos elementos significativos de las otras vistas.

En este documento se le dará principal importancia a los casos de uso de la funcionalidad principal del sistema. Se atacará principalmente aquellos casos de uso que involucran interacción con los actores.

Finalmente, este documento es la descripción de arquitectura del caso de estudio, no es un instructivo de cómo elaborar un SAD; en otras palabras el lector no encontrará aquí comentarios sobre qué tipo de información debe ser incluida en el documento ni cuáles son los criterios a utilizar en casos generales.

Referencias

Implementación de un sistema de información basado en componentes con J2EE.

S. Bengochea, F. Fajardo, J. Ferré. Reporte técnico 03-17, InCo-PEDECIBA, 2003.

UML Components. J. Cheesman, J. Daniels. Addison Wesley, 2001.

Design Patterns. E. Gamma, et al. Addison Wesley, 1995.

Architectural Blueprints—The “4+1” View Model of Software Architecture, P.

Kruchten, Rational Software Corp. IEEE Software 12 (6), November 1995.

Applying UML and Patterns. C. Larman. Prentice-Hall, 2002.

IBM Rational Unified Process. <http://www.rational.com/rup>, 2003.

Unified Modeling Language. OMG. <http://www.omg.org/uml>, 2003.

Yoopeedoo (UPEDU). <http://www.yoopeedoo.org>, 2003.

Representación Arquitectónica

La características inherentes al sistema hace que la vista de casos de uso y la vista lógica sean las más relevantes y por ello serán las más extensas.

La arquitectura está representada por diferentes vistas utilizando notación UML de forma que permitan visualizar, entender y razonar sobre los elementos significativos de la arquitectura e identificar las áreas de riesgo que requieren mayor detalle de elaboración. Este documento es una forma de comunicar el modelo del subsistema, presentando la información y discusiones estructuradamente.

La arquitectura del subsistema se descompone en las siguientes dimensiones:

Requerimientos: Requerimientos funcionales y no-funcionales del sistema.

Elaboración: Representación lógica del sistema y representación de tiempo de ejecución.

Implementación: Vista de módulos implementados, potenciales escenarios de infraestructura y el deployment de los módulos.

La siguiente sección detalla las vistas de la arquitectura que serán utilizadas para cubrir las dimensiones mencionadas.

Representación

La arquitectura del Subsistema de Reservas está representada siguiendo las recomendaciones de RUP. Las vistas necesarias para especificar dicho subsistema se enumeran a continuación:

Vista de Casos de Uso: Describe el proceso de negocio más significativo y el modelo del dominio. Presenta los actores y los casos de uso para el sistema.

Vista de Restricciones: Describe restricciones tecnológicas, normativas, uso de estándares, entre otros, las cuales deben ser respetadas tanto por el proceso de desarrollo como por el producto desarrollado.

Vista QoS: Incluye aspectos de calidad, y describe los requerimientos no-funcionales del sistema.

Vista Lógica: Describe la arquitectura del sistema presentando varios niveles de refinamiento. Indica los módulos lógicos principales, sus responsabilidades y dependencias.

Vista de Procesos: Describe los procesos concurrentes del sistema.

Vista de Implementación: Describe los componentes de deployment contruidos y sus dependencias.

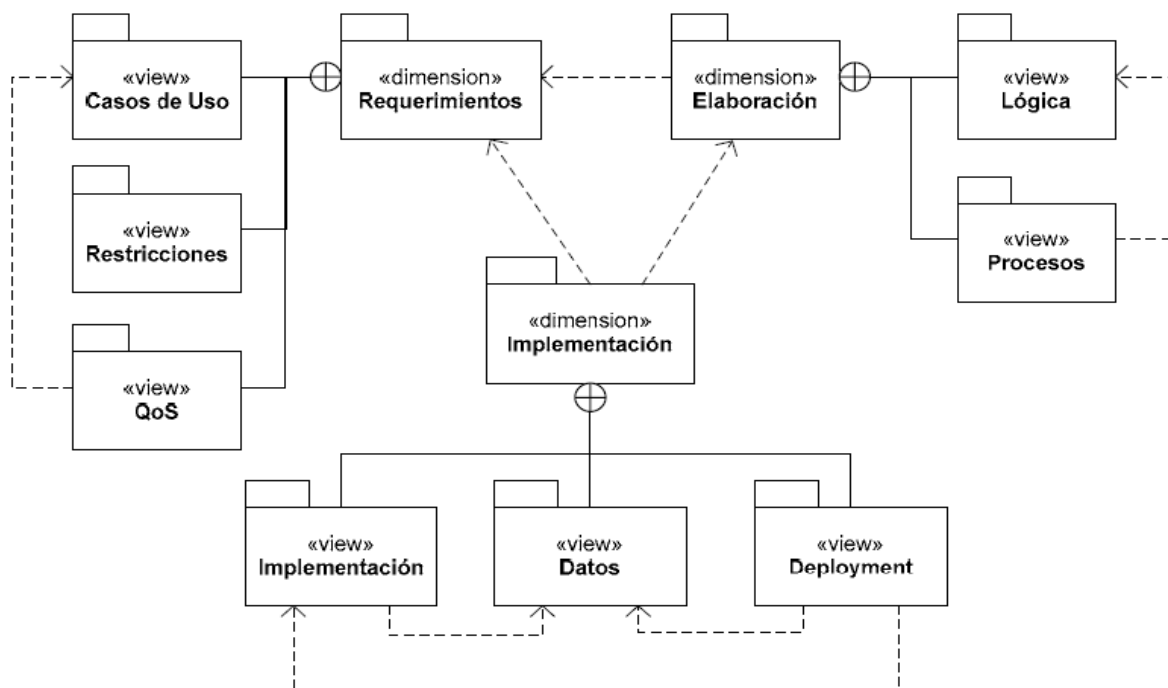
Vista de Datos: Presenta los modelos de datos, los servicios de persistencia y los servicios de transaccionalidad utilizados.

Vista de Deployment: Presenta aspectos físicos como topología, infraestructura informática, e instalación de ejecutables. Incluye además plataformas y software de base.

Estas vistas son las generales para cualquier sistema que se intente desarrollar, para nuestro caso, solo nos explotaremos en las que son significativas para este desarrollo.

La implementación todas estas, están sujetas a las demandas propias del desarrollo.

Las vistas presentadas forman en su conjunto una especificación completa del sistema, la cual se delinea en el siguiente diagrama. Las dependencias entre las vistas indican dependencias entre las vistas tanto a nivel de arquitectura como a nivel de diseño.



Objetivos Arquitectónicos y Restricciones

Objetivos Generales

Los Sistemas de Información Geográfica requieren de una fuerte interacción entre datos obtenidos en tareas de campo, coordenadas de posicionamiento entregadas por GPS, bases de datos e imágenes cartográficas que permitan plasmar las investigaciones y generar información útil para los usuarios. En este contexto, el objetivo general del proyecto será:

Desarrollar un sistema de adquisición, administración y representación de información para los investigadores del área de Química, que permita a los mismos un manejo simple y se presente como una herramienta útil para el uso en las tareas específicas que cotidianamente llevan a cabo.

Objetivos Específicos

Para lograr el objetivo mencionado, se describen a continuación metas específicas que se proponen alcanzar.

Investigar mecanismos de comunicación entre java y GPS, a través de la utilización del API de Comunicaciones de Java desarrollado por Sun, sobre el cual no se ha trabajado aún en esta Unidad Académica, como así también conocer sobre el protocolo NMEA y otros utilizados por los Sistemas de Posicionamiento.

Investigar técnicas de calibración de mapas y referenciación de información.

Diseñar un esquema de base de datos eficiente y acorde a las necesidades de los usuarios, de manera de plasmar correctamente la información obtenida en las etapas de relevamiento.

Integrar los conocimientos mencionados en una solución de software que se convierta en una herramienta práctica para los Investigadores a la hora de realizar sus tareas de campo.

Descripción de Procesos

El proceso comienza cuando un cliente solicita un estudio a la Universidad que requiere ciertos análisis químicos; los mismos son realizados por el área de química que es la encargada de llevar a cabo los estudios de suelo y agua; los cuales difieren de acuerdo a las necesidades y presupuesto del cliente, y si se realizará en un terreno ya explotado o por el contrario el territorio es virgen y se desea generar una línea base que permita controlar a futuro.

Para comenzar la evaluación, dicha área evalúa que se va a medir, lo cual depende de lo que se vaya a realizar en el futuro; una vez realizado esto, se desplaza al lugar que requiere el estudio. Una vez se han definido los límites de cada unidad se procede a tomar varias muestras de la zona tomando sitios de muestreo al azar, asegurándose de cubrir toda el área de estudio para poder ver en que condiciones está hoy el lugar.

Si el estudio es de suelo, se obtienen muestras de cada sitio, y distintas capas del suelo; mientras que si es de cuerpos de agua, se obtienen distinto número de muestras dependiendo del tamaño del cuerpo. Si el mismo es de gran tamaño pueden efectuarse en el centro, las orillas y a distintas profundidades.

Ciertas muestras efectuadas deben ser enviadas a laboratorios externos para su correspondiente análisis. Otras son analizadas en los laboratorios de la Universidad, donde a través de procedimientos establecidos se obtienen los resultados que luego serán volcados en los informes que se entregarán al cliente, conteniendo conclusiones acerca de la investigación realizada.

Vista de Caso de Uso

Esta vista presenta la percepción que tiene el usuario de las funcionalidades del sistema. Se presenta el proceso de negocio más importante y los casos de uso críticos que se derivan de éste. Este capítulo provee el contexto y determina el alcance del resto del documento.

Primeramente se describe el Negocio. Luego se presenta el modelo del dominio para el Sistema. Se identifican actores y se detallan los casos de uso significativos.

Descripción de los Actores

Nombre del Actor: Investigador

Definición : Es la persona encargada de administrar en su totalidad los estudios, es quien puede generar mapas y exportar la información almacenada en el sistema

Nombre del Actor: Administrador

Definición : Esta encargado de la creación y manejo de los distintos usuarios, como así también de sus privilegios, también es quien debe hacer los correspondiente backups al sistema

Nombre del Actor: Laboratorista

Definición : Persona responsable de realizar las actividades de laboratorio a las muestras tomadas por el Investigador de campo, es quien debe completar las planillas adecuadas para cada muestra.

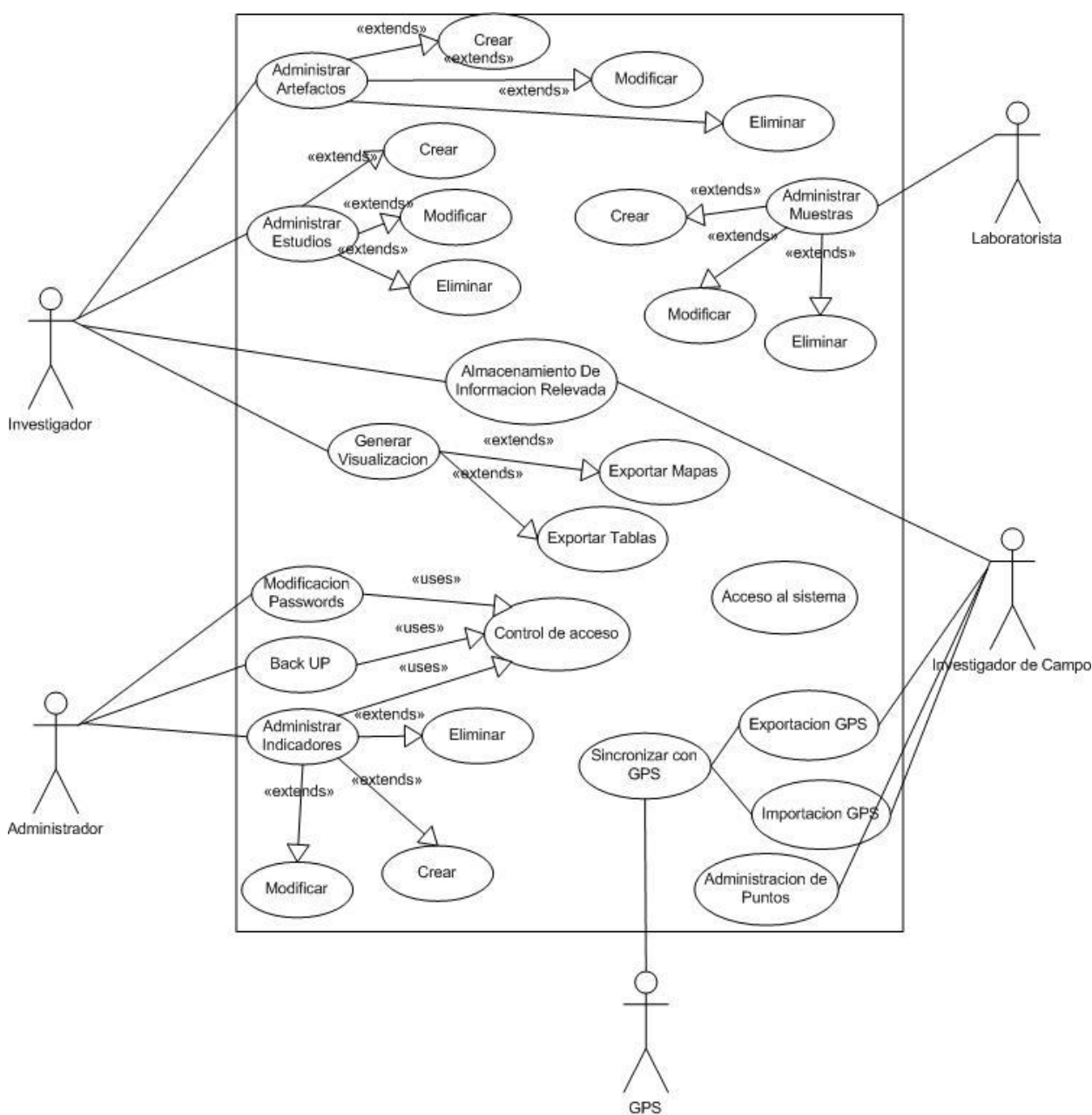
Nombre del Actor: Investigador de Campo

Definición : Es quien efectúa las muestras en el campo y es el quien debe cargar las tablas de cada una de ellas, para que el laboratorista realice su trabajo

Contexto del sistema

Los casos de uso críticos para el proceso se describen en esta sección. Primero se indica las relaciones entre los casos de usos detectados y luego se presenta la versión expandida de los mismos.

En el siguiente diagrama se puede observar como los distintos casos de uso interactúan unos con otros y como esta compuesto el contexto general del sistema.



Vista Lógica

Paquetes de Diseño significativos para la arquitectura

Según lo diseñado el sistema está compuesto por un conjunto de paquetes, cada uno con una función específica.

A continuación vemos los distintos paquetes interactuando entre si.

Diagrama de paquetes Inicial

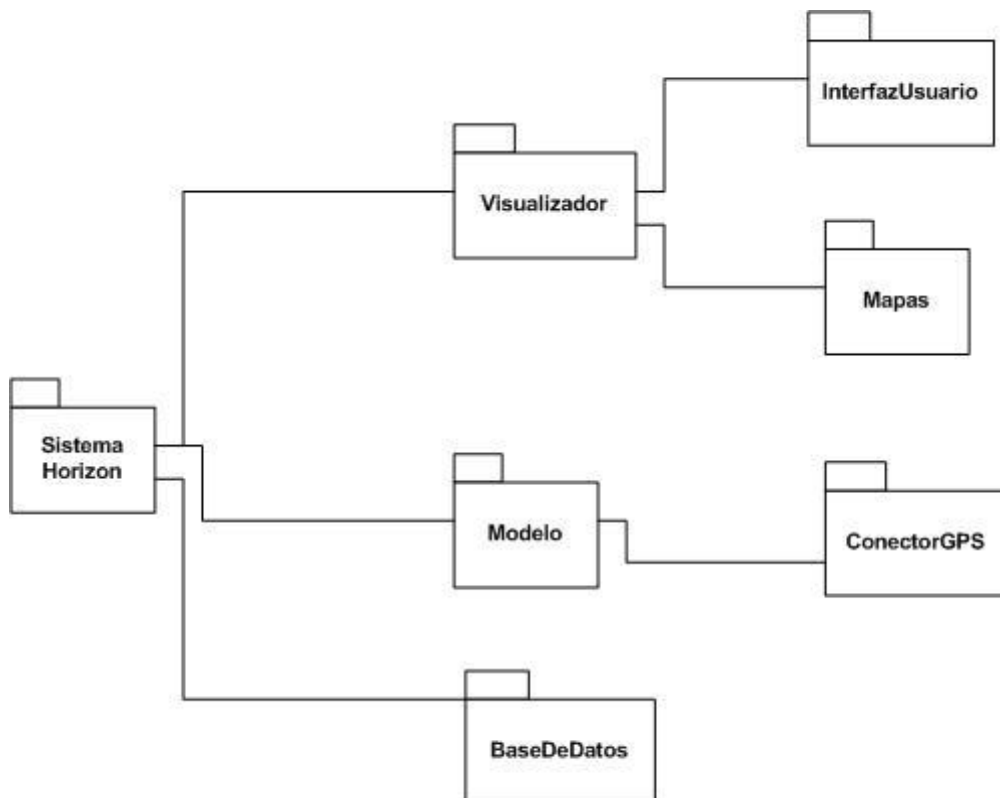
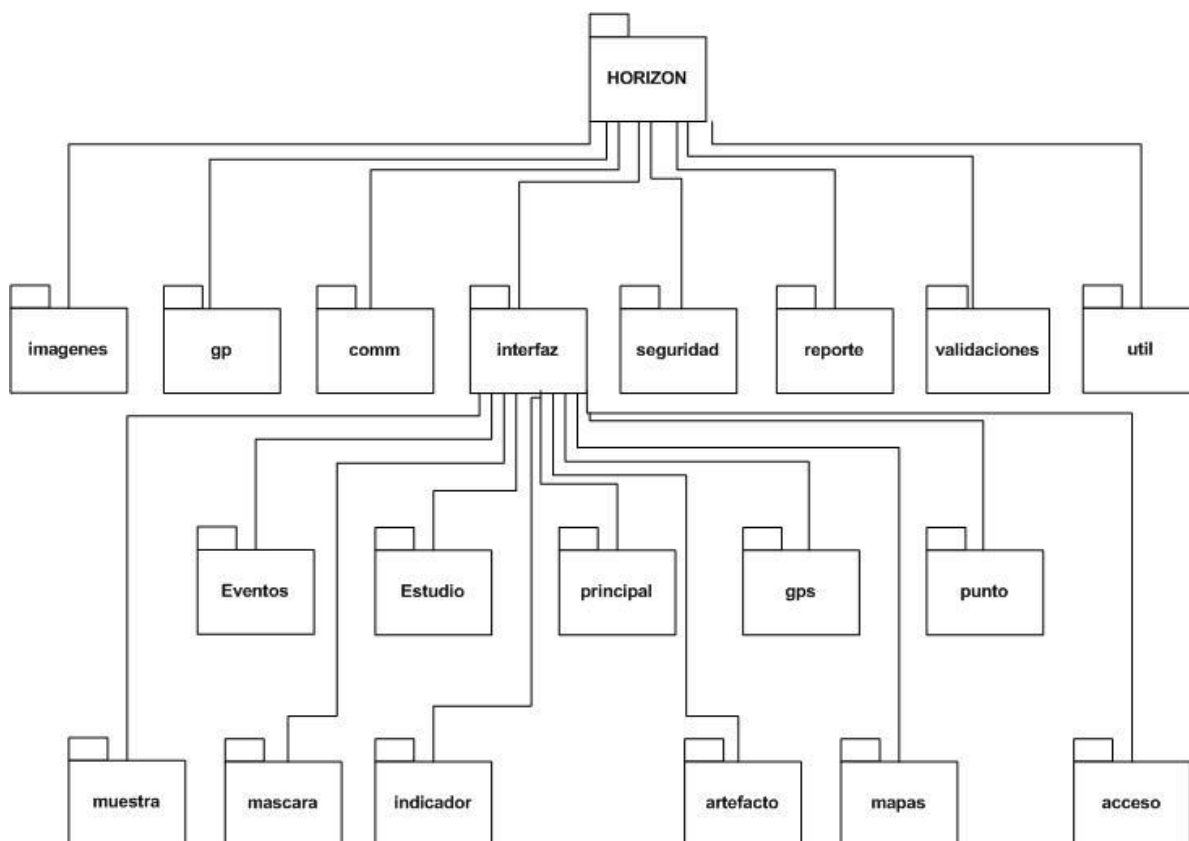


Diagrama de paquetes Final



Vista de Procesos

El proceso comienza cuando un cliente solicita un estudio a la Universidad que requiere ciertos análisis químicos; los mismos son realizados por el área de química que es la encargada de llevar a cabo los estudios de suelo y agua; los cuales difieren de acuerdo a las necesidades y presupuesto del cliente, y si se realizará en un terreno ya explotado o por el contrario el territorio es virgen y se desea generar una línea base que permita controlar a futuro.

Para comenzar la evaluación, dicha área evalúa que se va a medir, lo cual depende de lo que se vaya a realizar en el futuro; una vez realizado esto, se desplaza al lugar que requiere el estudio. Una vez se han definido los límites de cada unidad se procede a tomar varias muestras de la zona tomando sitios de muestreo al azar, asegurándose de cubrir toda el área de estudio para poder ver en que condiciones está hoy el lugar.

Si el estudio es de suelo, se obtienen muestras de cada sitio, y distintas capas del suelo; mientras que si es de cuerpos de agua, se obtienen distinto número de muestras dependiendo del tamaño del cuerpo. Si el mismo es de gran tamaño pueden efectuarse en el centro, las orillas y a distintas profundidades.

Ciertas muestras efectuadas deben ser enviadas a laboratorios externos para su correspondiente análisis. Otras son analizadas en los laboratorios de la Universidad, donde a través de procedimientos establecidos se obtienen los resultados que luego serán volcados en los informes que se entregarán al cliente, conteniendo conclusiones acerca de la investigación realizada.

Según el diseño que se le dio al sistema, nos encontramos con las siguientes funcionalidades principales

Acceso al Sistema

Descripción : Permitirá al usuario ingresar al sistema ya sea como administrador o investigador, aportando seguridad al mismo.

Administración de Acceso

Descripción : Permitirá al administrador poder cambiar la clave de acceso de los usuarios, realizar Backups, agregar nuevos mapas al sistema y crear nuevos indicadores tanto de agua como de suelo.

Administración de Estudios

Descripción : Permitirá crear, modificar y eliminar estudios realizados por el área de química en sus trabajos de campo.

Administración de Artefactos

Descripción : Permitirá el manejo de creación, modificación y eliminación de los artefactos pertenecientes a un estudio.

Administración de Puntos

Descripción : Permitirá crear, modificar y eliminar puntos que hacen referencia a los artefactos estudiados por el área de

química.

Administración de Muestras

Descripción : Permitirá crear, modificar y eliminar muestras tomadas por el área de química en sus trabajos de campo.

Administración de Indicadores

Descripción : Englobará el manejo de indicadores de suelo y agua, para luego ser utilizados en los análisis de laboratorio correspondientes.

Generar Backup

Descripción : Permitirá realizar copias de respaldo de la BD del sistema como así también la importación de las mismas si fuese necesario

Exportar Datos

Descripción : Se encargará de sincronizar la aplicación con el GPS, para así poder exportar los puntos pertenecientes a un artefacto facilitándonos coordenadas exactas del lugar.

Importar Datos

Descripción : Se encargará de sincronizar la aplicación con el GPS, para así poder importar los puntos al sistema facilitándonos la ubicación de algún objeto.

Generar Visualización

Descripción : Permitirá volcar sobre una imagen los puntos donde se encuentre el o los artefactos del estudio, permitiéndonos una mejor visualización de los mismos.

Otras Características

Características arquitectónicas del sistema

- Control de Acceso para usuarios.
- Encriptación de contraseñas con el algoritmo más seguro hasta la fecha MD5.
- Generación de mapas georeferenciados en base a imágenes previamente cargadas.
- Exportación de datos generados por el sistema a través de reportes
- Comunicación con hardware específico, utilizando un protocolo que cumpla con las especificaciones de GARMIN.
- Conexión a una base de datos relacional (MySQL).
- Validaciones con mascaras.
- Uso de patrones de diseño complejos
- Código documentado según estándares
- Cumplimiento de estándares de codificación
- Plataforma independiente.
- Código abierto
- Instalador independiente e integrado
- Documentado completamente.
- Ciclo de desarrollo iterativo e incremental
- Manejo de errores.