

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

ESCUELA DE SISTEMAS



**DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS Y COMPUTACIÓN.**

TEMA:

**DESARROLLO DE UN SISTEMA INFORMATICO PARA LA TRAZABILIDAD
DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y
NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL
DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN
AMÉRICA LATINA (CISEAL)**

AUTOR:

DIEGO PAUL MONTALVO PAEZ

DIRECTOR:

MTR. LUIS FERNANDO AGUAS BUCHELI

QUITO, 2018

CONTENIDO

CAPITULO 1	7
1. Introducción	8
1.1 Antecedentes	8
1.2 Justificación	8
1.3 Objetivos.....	9
1.3.1 Objetivo General	9
1.3.2 Objetivos Específicos.....	9
1.4 Alcance.....	10
CAPITULO 2	10
2 Introducción al Problema	11
CAPITULO 3	17
3. Herramientas de Desarrollo	17
3.1 Aplicaciones Web.....	17
3.1.1 Definición.....	17
3.1.2 Características	17
3.1.3 Arquitectura de una aplicación Web	18
3.1.4 Ventajas.....	19
3.1.5 Desventajas.....	20
3.2 Lenguaje de programación JAVA	20
3.2.1 Definición.....	20
3.2.2 Historia	21
3.2.3 Características	22
3.2.4 Ventajas.....	23
3.2.5 Desventajas.....	23
3.3 Bases de Postgres	23
3.3.1 Definición.....	23
3.3.2 Historia	24
3.3.3 Características	24
3.3.4 Ventajas.....	25
3.3.5 Desventajas.....	25
3.4 Servidor de Aplicaciones WILDFLY.....	25
3.4.1 Definición.....	25
3.4.2 Historia	26
3.4.3 Características	26
3.4.4 Ventajas.....	26

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACIÓN EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

3.4.5	Desventajas.....	26
3.5	Capa de Presentación (Primefaces).....	27
3.5.1	Definición.....	27
3.5.2	Características	27
3.5.3	Ventajas.....	27
3.5.4	Desventajas.....	27
3.5.5	Adaptación a Móviles desde el navegador.....	28
CAPITULO 4		29
4.	Metodología de Desarrollo Scrum	29
4.1	Definición.....	29
4.2	Entregas Parciales	29
4.4	Roles	32
4.5	Proceso de Scrum.....	33
4.5.1	Introducción.....	33
4.5.2	Planificación	34
4.5.3	Ejecución	35
4.5.4	Reunión Diaria.....	36
4.5.5	Revisión.....	37
4.5.6	Retrospectiva	38
4.5.7	Afinamiento.....	38
CAPITULO 5		40
5.	Alcance de la Aplicación	40
5.1	Requerimientos Generales de URBS – PEED en CISEAL.....	40
5.2	Análisis de Requerimientos de URBS – PEED en CISEAL.....	40
5.3	Metodología para el desarrollo de la aplicación web	41
5.3.1	Planificación	41
5.3.2	Entregas Incrementales	42
5.3.3	Reuniones frecuentes y Afinamiento	43
5.3.4	Entrega del producto final	43
CAPITULO 6		44
6.	Diseño y Desarrollo de la aplicación web	44
6.1	Estructura de la aplicación web	44
6.2	Diseño de Interfaz	44
6.3	Diseño de Base de Datos	45
6.4	Arquitectura de la Aplicación	46
6.4.1	Definición de Arquitectura de aplicación	46
6.4.2	Definición de Arquitectura de aplicación N-Capas.....	46

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

6.5	Acceso a la aplicación web (Login)	47
6.5.1	Roles	49
6.6	Pantalla principal de la aplicación web	50
6.7	Administración de usuarios	52
6.8	Administración de Instituciones	54
6.9	Administración de Marcadores Serológicos	56
6.10	Administración de Kit de Reactivos	57
6.11	Administración de Códigos de Mapa de Trabajo de Serología.....	59
6.12	Administración y procesos del material de referencia de URBS-PEED	60
6.12.1	Recepción de Material de Referencia.....	61
6.12.2	Caracterización de Materia Prima	62
6.12.3	Caracterización de Material de Referencia	64
6.12.4	Caracterización de Laboratorio de Referencia	65
6.12.5	Transformación	66
6.12.6	Situación.....	67
6.13	Auditoría	68
6.14	Reporte (Estadísticas).....	68
CAPITULO 7		71
7.	Instalación de la aplicación web	71
7.1	Verificación del ambiente de ejecución para la aplicación web	71
7.2	Instalación de Java	72
7.3	Instalación de la Base de datos Postgresql.....	77
7.3.1	Instalación de la Base de datos Postgresql.....	77
7.3.2	Instalación de cliente PgAdmin.....	82
7.3.3	Creación de la base de datos	86
7.4	Instalación del Servidor de Aplicaciones WildFly	90
7.4.1	Configuración del Pool Base de datos.....	91
7.4.2	Configuración de validación de credenciales (Login).....	92
7.4.3	Configuración de puertos.....	93
7.5	Ejecución del Servidor de Aplicaciones WildFly.....	94
7.6	Interacción del personal de URBS-PEED con la aplicación web.....	98
CAPITULO 8		99
8.	Conclusiones y Recomendaciones.....	99
8.1	Conclusiones.....	99
8.2	Recomendaciones	100
Bibliografía		102

CONTENIDO ILUSTRACIONES

Ilustración 1 Proceso del URBS-PEED	14
Ilustración 2 Multidispositivo Fuente: https://www.xatakamovil.com/aplicaciones/el-debate-esta-servido-son-mejores-las-apps-con-version-web-o-las-apps-nativas	17
Ilustración 3 Arquitectura de una aplicación web	19
Ilustración 4 Logo Java	21
Ilustración 5 Roles en Scrum	33
Ilustración 6 Proceso de Scrum Fuente: https://proyectosagiles.org/como-funciona-scrum/	39
Ilustración 7 Primefaces.....	45
Ilustración 8 PGAdmin - Tablas de Base de datos.....	45
Ilustración 9 Diagrama entidad - relación	46
Ilustración 10 Arquitectura N-Capas Fuente: https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/ .	47
Ilustración 11 Aplicación Web – Login.....	48
Ilustración 12 Aplicación Web - Pantalla Principal Administrador.....	49
Ilustración 13 Aplicación Web - Pantalla Principal Analista	50
Ilustración 14 Aplicación Web - Pantalla Principal Usuario Consulta	50
Ilustración 15 Aplicación Web - Menú	51
Ilustración 16 Aplicación Web - Perfil de usuario	51
Ilustración 17 Aplicación Web - Zona de Información.....	52
Ilustración 18 Aplicación Web - Cierre de Sesión	52
Ilustración 19 Aplicación Web - Usuarios Lista	53
Ilustración 20 Aplicación Web - Ingreso de Usuario.....	54
Ilustración 21 Aplicación Web - Instituciones Lista	55
Ilustración 22 Aplicación Web - Ingreso de Institución.....	55
Ilustración 23 Aplicación Web - Marcadores Serológicos Lista	56
Ilustración 24 Aplicación Web - Ingreso Marcador Serológico	57
Ilustración 25 Aplicación Web - Kit Lista	58
Ilustración 26 Aplicación Web - Ingreso Kit.....	58
Ilustración 27 Aplicación Web - Códigos de Mapa de Trabajo de Serología Lista.....	59
Ilustración 28 Aplicación Web - Códigos de Mapa de Trabajo de Serología	60
Ilustración 29 Aplicación Web - Material de Referencia Lista	61
Ilustración 30 Aplicación Web - Material de referencia - Recepción	62

Ilustración 31 Aplicación Web - Material de referencia - Caracterización de Materia Prima	
Ilustración 32 Aplicación Web - Material de referencia - Caracterización de Material de Referencia	64
Ilustración 33 Aplicación Web - Material de referencia - Caracterización Laboratorio de Referencia	65
Ilustración 34 Aplicación Web - Material de referencia - Transformación	66
Ilustración 35 Aplicación Web - Material de referencia - Situación.....	67
Ilustración 36 Aplicación Web - Auditoría.....	68
Ilustración 37 Aplicación Web - Reporte Pag 1	69
Ilustración 38 Aplicación Web - Reporte Pag 12.....	70
Ilustración 39 Características del equipo servidor.....	72
Ilustración 40 Descarga de Java Fuente: http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html	73
Ilustración 41 Instalación de Java.....	73
Ilustración 42 Instalación de Java - Selección de Directorio.....	74
Ilustración 43 Instalación de Java - Instalación Exitosa.....	74
Ilustración 44 Configuración de Variables de ambiente de Java	75
Ilustración 45 Creación de variable de ambiente JAVA_HOME	76
Ilustración 46 Creación de variable JAVA_HOME en el Path del sistema operativo	76
Ilustración 47 Verificación de Variables de ambiente Java.....	77
Ilustración 48 Descarga de PostgreSQL Fuente: https://www.enterprisedb.com/downloads/postgres-postgresql-downloads#windows	78
Ilustración 49 Instalación de PostgreSQL	78
Ilustración 50 Instalación de PostgreSQL - Selección de directorio de instalación.....	79
Ilustración 51 Instalación de PostgreSQL - Selección de directorio de datos	80
Ilustración 52 Instalación de PostgreSQL - clave del usuario postgres.....	80
Ilustración 53 Instalación de PostgreSQL - Selección de puerto.....	81
Ilustración 54 Instalación de PostgreSQL - Opciones Avanzadas para Cluster.....	81
Ilustración 55 Instalación de PostgreSQL - Confirmación de Instalación	82
Ilustración 56 Instalación de PostgreSQL - Instalación Exitosa.....	82
Ilustración 57 - Instalación de pgAdmin – Descarga Fuente: https://www.pgadmin.org/download/pgadmin-4-windows/	83
Ilustración 58 Instalación de pgAdmin - Descarga de repositorio Fuente: https://www.postgresql.org/ftp/pgadmin/pgadmin4/v2.1/windows/	83
Ilustración 59 Instalación de pgAdmin	84
Ilustración 60 Instalación de pgAdmin - Licenciamiento y condiciones.....	84

Ilustración 61 Instalación de pgAdmin - Directorio de Instalación	85
Ilustración 62 Instalación de pgAdmin - Configuración Adicional	85
Ilustración 63 Instalación de pgAdmin - Confirmación de Instalación	86
Ilustración 64 Instalación de pgAdmin - Finalización exitosa de Instalación	86
Ilustración 65 Creación de base de datos - Nueva conexión a servidor.....	87
Ilustración 66 Creación de base de datos - Pestaña Conexión	88
Ilustración 67 Creación de base de datos - Conexión a servidor exitosa.....	88
Ilustración 68 Creación de base de datos - nueva base de datos	89
Ilustración 69 Creación de base de datos	89
Ilustración 70 Creación de base de datos - Estructura de la Base de datos.....	90
Ilustración 71 Standalone - subsystem Datasource	91
Ilustración 72 aplicativo web - persistence	92
Ilustración 73 Standalone - subsystem security domain	93
Ilustración 74 Standalone - Configuración de puertos	94
Ilustración 75 pre despliegue de la aplicación.....	94
Ilustración 76 Log WildFly - Inicio de servidor de aplicaciones.....	95
Ilustración 77 Log WildFly - Conexión a base de datos	95
Ilustración 78 Log WildFly - Servicios.....	96
Ilustración 79 Log WildFly - Validación de tablas	96
Ilustración 80 Log WildFly - Finalización de inicio.....	96
Ilustración 81 Verificación de despliegue de aplicación.....	97
Ilustración 82 Verificación de servicio WildFly.....	97
Ilustración 83 Verificación de la aplicación web	98

CAPÍTULO I

1. Introducción

1.1 Antecedentes

En los últimos años la tecnología está fuertemente relacionada con el mejoramiento del manejo de la información en cualquier ámbito existente, uno de los medios en los que más se requiere esta modernización es el tema de trazabilidad en procesos de centros de investigación.

En el CISEAL el proceso de análisis de la materia prima (plasmas reactivos y no reactivos), categorización y resultados que se realizan sobre esta tiene un importante componente de vinculación con la comunidad al trabajar con los bancos de sangre de todo el país, la necesidad de automatizar este proceso y obtener una trazabilidad sobre este es requisito clave para obtener la certificación ISO 9000:2015.

Un sistema que asegure la calidad, privacidad, centralización de la información, así como facilitar la administración del proceso mediante una interfaz amigable al usuario y ofrezca reportes con información precisa es de vital importancia para los interesados.

1.2 Justificación

En el Centro De Investigación para la Salud en América Latina (CISEAL) actualmente la trazabilidad del proceso de análisis de materia prima (plasmas reactivos y no reactivos), categorización y resultados de la misma se realiza mediante herramientas de ofimática como el Excel creando inconsistencias y descentralización de la información, así como complejidad en obtener los reportes que se solicitan.

Entre los objetivos de CISEAL se encuentra obtener la certificación ISO 9000:2015, para esto se necesitan automatizar este proceso y generar esta

trazabilidad necesaria para optimizar el muestreo en laboratorio.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un sistema informático para obtener la trazabilidad de los resultados de la materia prima (plasmas reactivos y no reactivos) en el programa de evaluación externo del desempeño en el Centro De Investigación para la Salud en América Latina (CISEAL) y generar reportes con esta información mediante el uso de software libre.

1.3.2 Objetivos Específicos

- Generar la trazabilidad en el proceso de muestreo resultados de la materia prima (plasmas reactivos y no reactivos) en el programa de evaluación externo del desempeño en el Centro De Investigación para la Salud en América Latina.
- Automatizar el proceso de ingreso de materia prima (plasmas reactivos y no reactivos) así como su categorización y resultados para CISEAL.
- Desarrollo e implementación del sistema con herramientas de software libre y de última tecnología.
- Desarrollar el sistema informático con metodología de Software SCRUM para crear un producto correctamente planificado y funcional.
- Desarrollar el sistema con interfaz amigable, intuitivo y de fácil uso para el usuario.
- Implementar los reportes necesarios para obtener a detalle la información del proceso de los resultados de la materia prima (plasmas reactivos y no reactivos) en el programa de evaluación externo del desempeño del Centro De Investigación para la Salud en América Latina.
- Implementar el sistema, realizando las pruebas respectivas.

1.4 Alcance

La presente disertación de grado concluirá con la entrega de un sistema informático desarrollado con herramientas de software libre, implantado en el programa de evaluación externo del desempeño del Centro De Investigación para la Salud en América Latina.

CAPÍTULO II

2. Introducción al Problema

2.1 Centro De Investigación para la Salud en América Latina (CISEAL)

2.1.1 Introducción del Centro De Investigación para la Salud en América Latina (CISEAL)

El Centro de Investigación para la Salud en América Latina (CISEAL) fue creado en cooperación con el Instituto de Enfermedades Tropicales de la Universidad de Ohio, con el fin de realizar trabajos de investigación biológica y biomédica, actividades de vigilancia epidemiológica y programas de educación y entrenamiento en el Ecuador.

Los principales proyectos de investigación de CISEAL se enfocan al momento en diversos aspectos relacionados a la Enfermedad de Chagas y al mejoramiento de la seguridad y calidad de la sangre donada en el Ecuador.

El CISEAL trabaja juntamente con el Ministerio de Salud Pública del Ecuador, la Cruz Roja Ecuatoriana, y la Universidad de Ohio. La amplia red de colaboradores nacionales e internacionales, les permiten diseñar y ejecutar proyectos de investigación utilizando tecnología de punta, con aplicaciones que benefician directamente a los ecuatorianos.

2.1.2 Visión

“Ser un espacio colaborativo en América Latina, en el cual diversas concepciones científicas se unen en torno al compromiso por construir mejores condiciones de salud, para aquellas personas cuyo bienestar es amenazado por la presencia de enfermedades infecciosas y crónicas.”
(CISEAL, 2016: 06).

2.1.3 Misión

“Sirve como una plataforma internacional para la generación de colaboraciones científicas que amplían el conocimiento, enriquecen la educación y mejoran la salud de la población.” (CISEAL, 2016: 06).

2.2 Introducción del programa de evaluación externo del desempeño del Centro De Investigación para la Salud en América Latina (PEED).

2.2.1 Antecedentes

Los antecedentes que dieron como consecuencia la creación del Programa de Evaluación Externa PEED partieron con el Ministerio de Salud Pública del Ecuador MSP que en su misión, trata de consolidar el trabajo de todos los bancos de sangre del Ecuador para fomentar un ambiente de mejoramiento progresivo y continuo, además de garantizar que todos los plasmas de sangre usados para las transfusiones en el país tengan estándares de calidad altos y que no se corra el riesgo de que alguna persona pueda contagiarse de alguna enfermedad que complique su vida.

De esta manera el PEED es establecido en 2002 con el objetivo de cumplir con la misión y los requerimientos presentados por la más alta institución del país MSP, de esta manera, en 2011, en un oficio del ministro Dr. David Chiriboga, comunica a todos los bancos de sangre del país que tienen de carácter obligatorio participar el Programas de Evaluación Externa que está establecido en la Pontificia Universidad Católica del Ecuador.

Desde esa fecha el PEED sigue trabajando en certificar la calidad de los plasmas que se manejan en todos los bancos de sangres públicos y privados del Ecuador mediante un proceso que ha sido certificado por diferentes entidades correspondientes al medio.

2.2.2 Misión

“Llegar al 2018 como una unidad operativa certificada y acreditada que nos otorgue reconocimiento a nivel nacional.” (PEED 2017: 10).

2.2.3 Visión

“URBS-PEED somos una unidad operativa dentro del CISEAL - PUCE que desarrolla programas de control de calidad y soporte técnico para el tamizaje serológico de los Servicios de Sangre del país pertenecientes a la Red Pública de Salud y la Red Privada Complementaria, con el fin de ofrecer apoyo al programa de Sangre manteniendo estándares de calidad, buenas prácticas de manufactura y reporte de resultados oportunos.” (PEED 2017: 10).

2.2.4 Objetivos

Los objetivos detallados en el mapa estratégico de PEED son, en síntesis, obtener una rentabilidad aceptable para garantizar todo el trabajo operativo y administrativo del programa, incrementar la satisfacción año a año de los involucrados, también alcanzar las metas de los indicadores que arrojan cada proceso para que con esto el personal de PEED también se sienta conforme con su medio y pueda rendir al máximo y explotar toda su experiencia.

PEED maneja el programa externo con el objetivo de recibir muestras periódicamente de los bancos de sangre con sus respectivos resultados verificando que los datos enviados son los correctos, con estos resultados y mediante medidores, se establece una calificación dependiendo de cuantos falsos positivos fueron obtenidos, de esta manera PEED corrobora que la información de los plasmas es verídica y confiable.

También se maneja el programa interno que consiste en colaborar con los bancos de sangre a la verificación de las muestras periódicamente para agilizar más el proceso de mejoramiento de la calidad de plasmas, que se van a utilizar para las transfusiones de sangre.

2.2.5 Valores

- Confidencialidad.
- Responsabilidad.

2.3 Proceso de recepción y tratado de muestras de la materia prima (plasmas reactivos y no reactivos).

En URBS-PEED se manejan distintos procesos que complementan sus objetivos en el mapa general de procesos, a continuación, el detalle:

- Procesos Gobernantes.
- Procesos Principales.
- Procesos de Apoyo.

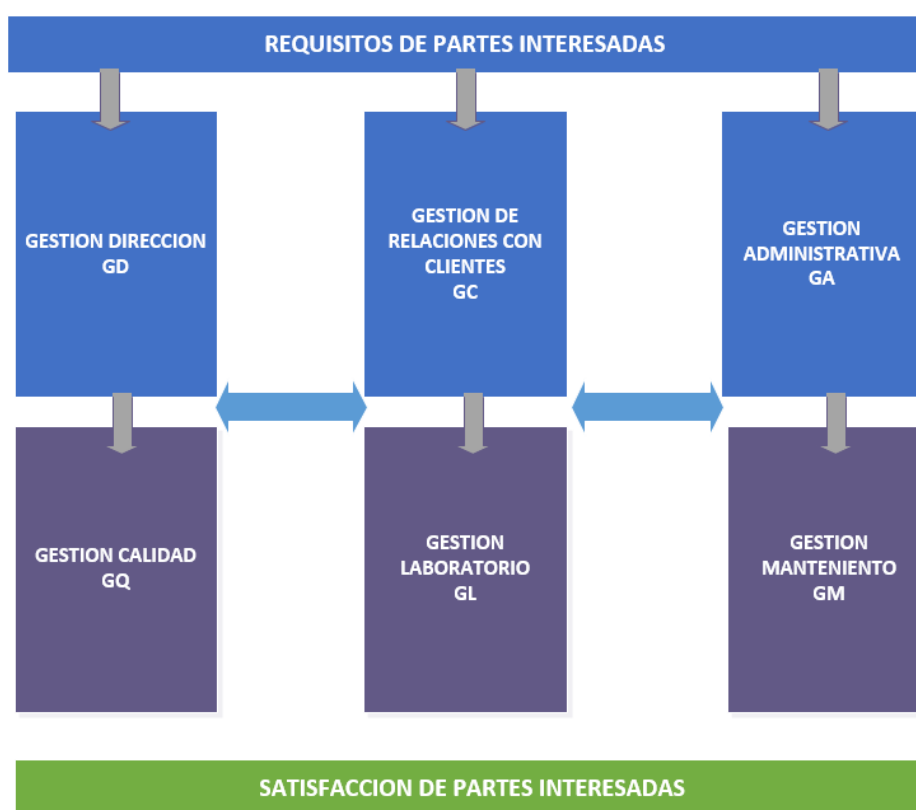


Ilustración 1 Proceso del URBS-PEED

2.3.1 Procesos Gobernantes

Los procesos gobernantes son los que dan el direccionamiento a toda la unidad y manejan que se esté alineado a la misión y visión, cumpliendo con los objetivos estratégicos.

2.3.2 Gestión Dirección

Se encargan de la gestión de todo el equipo de trabajo.

2.3.3 Gestión Calidad

Se encarga de que la gestión del equipo cumpla con las calidades necesarias para cumplir con los objetivos en base a los lineamientos adecuados.

2.3.4 Procesos Principales

Los procesos principales son los que realizan toda la actividad principal dentro del programa.

2.3.5 Gestión de Relación de Clientes

Esta gestión administra la comunicación que se tiene con el cliente, en este caso los bancos de sangre, realizando convenios con ellos.

2.3.6 Gestión Laboratorio

En la gestión de laboratorio se manipula las muestras de plasma para que estas interactúen con los reactivos y los no reactivos con el objetivo de verificar los resultados

- **Pre-análisis:** en esta etapa se hace la recepción, caracterización y almacenamiento de la materia prima.
- **Análisis:** En esta etapa se seleccionan los plasmas que se van a utilizar para verificar que la información tenga o no validez en comparación con lo enviado por el banco de sangre.
- **Post-análisis:** En esta etapa se envía el reporte de lo obtenido a los bancos de sangre para su información.

2.3.7 Procesos Apoyo

Los procesos que ayudan para que se lleven al cabo correctamente los procesos principales.

2.3.8 Gestión Administrativa

Esta gestión se encarga que todos los miembros del equipo tengan los elementos y servicios necesarios para el continuo trabajo.

2.3.9 Gestión de Mantenimiento

Esta gestión se ocupa de que todos los equipos usados para el trabajo estén en el mejor estado para evitar fallas y previniendo inconvenientes.

CAPÍTULO III

3. Herramientas de Desarrollo

3.1 Aplicaciones Web

3.1.1 Definición

Las aplicaciones web obtienen este nombre porque pueden ser ejecutadas a través de la web, esto quiere decir que toda la información que se entrega y recibe se visualiza mediante un navegador, la aplicación es software que debe ser creado en un lenguaje de programación que sea soportado en el navegador.

Las aplicaciones web son la evolución de la arquitectura Cliente-Servidor que se utilizaba para distribuir sistemas a través de una red, este tipo de aplicaciones descarta la configuración de cada uno de los computadores en donde se vaya a trabajar con el software y habilita el acceso de este desde cualquier lugar físico o dispositivo.



Ilustración 2 Multidispositivo Fuente: <https://www.xatakamovil.com/aplicaciones/el-debate-esta-servido-son-mejores-las-apps-con-version-web-o-las-apps-nativas>

3.1.2 Características

Compatibilidad: Una versión única de una aplicación web puede ejecutarse en cualquier sistema operativo, ya que se interactúa con ella

mediante un navegador web, independiente de que se trabaje sobre Windows, Linux, Android, Mac, etc.

Distribución: No requiere distribución individual, al ser accesible por web, no requiere instalación por cada computador.

Actualizaciones: Las actualizaciones de la aplicación web se ejecutan una sola vez en un servidor centralizado que la contiene y así no requiere actualización individual.

Capacidad de cómputo: No requiere de una gran capacidad de cómputo en su hardware, las aplicaciones web solo necesitan un mínimo posible para la ejecución del navegador, la aplicación ejecuta todo el procesamiento de información en los procesadores del servidor que la contiene.

Seguridad: La seguridad se centraliza en el servidor que contiene la aplicación web y no es necesario la instalación de software de seguridad en cada uno de los equipos.

Escalabilidad: Ya que la aplicación web se ubica en un servidor, se puede mejorar sus prestaciones y fusionar con otras aplicaciones web para que trabajen en un solo conjunto.

3.1.3 Arquitectura de una aplicación Web

A continuación, se listan los componentes que forman parte de la arquitectura de una aplicación web.

- **Base de Datos:** Es el servidor cuyo servicio se encarga del almacenamiento y tratamiento de la información.
- **Servidor Web:** Es un servidor el cual se conforma de un conjunto de programas que están en permanente estado de ejecución y que ofrecen una serie de servicios donde la aplicación web corre.
- **Red:** Es el conjunto de protocolos y servicios de comunicación que

interconectan a los servidores con los clientes para que puedan acceder a las aplicaciones web.

- **Navegadores Web:** Es un programa que se ejecuta en el cliente, permite acceder y desplegar las aplicaciones web en el equipo mediante protocolos de comunicación.

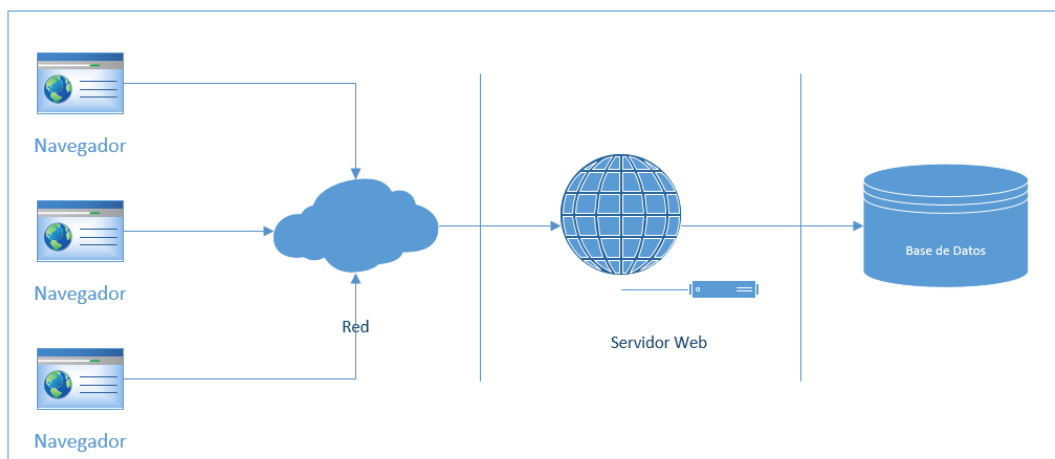


Ilustración 3 Arquitectura de una aplicación web

3.1.4 Ventajas

- **Instalación:** Es una de las grandes ventajas de las aplicaciones web, que no se debe instalar nada en los equipos clientes ya que se utilizan los navegadores web.
- **Carga de trabajo:** No se tiene una carga de trabajo alta en los procesadores de los equipos clientes ya que todo el procesamiento se lleva al cabo en el servidor web.
- **Mantenimiento:** El mantenimiento de una aplicación web es centralizado por lo tanto solo se hacen cambios en el servidor web y estos se verán reflejados en todos los clientes que accedan a esta.
- **Control de usuarios:** El control de acceso a la aplicación se maneja de manera centralizada por lo cual nadie que no tenga los permisos puede acceder a ella.
- **Movilidad:** La aplicación web puede ser visualizada desde cualquier dispositivo o localización mientras esta se encuentre ejecutándose sobre un ambiente de red accesible para los clientes.
- **Errores:** Los errores en la aplicación web son controlados

centralizadamente y corregidos una sola vez en el servidor web, mejorando el ciclo de actualización para los navegadores.

3.1.5 Desventajas

- **Necesidad de conexión a red:** La mayoría de aplicaciones web trabajan sincrónicamente¹, lo que significa que tiene que estar constantemente conectado a una red.
- **Inconvenientes con los navegadores:** Los navegadores más antiguos no soportan algunas de las nuevas funcionalidades visuales de las aplicaciones web.
- **Tráfico en la red:** Cuando una aplicación web tiene una gran cantidad de usuario que acceden a ella, el tráfico aumenta y esto lleva a la lentitud en la respuesta de la aplicación.
- **Apariencia:** Si bien se puede usar la última tecnología para la parte visual de una aplicación web, front-end², esta va a depender de como el navegador despliega los componentes gráficos, de tal manera que el resultado no siempre puede ser el mismo, o el esperado.

3.2 Lenguaje de programación JAVA

3.2.1 Definición

Java es un lenguaje de programación orientado a objetos que permite producir software para distintas plataformas, este fue lanzado por la empresa Sun Microsystems en los años de 1995. Al ser un lenguaje de programación compilado, lo que hace Java es compilar el código que crea el desarrollador y lo ejecuta en lenguaje que la máquina pueda entender, para este concepto, Java hace uso de la Java Virtual Machine o JVM que se ocupa de que el lenguaje pueda ser ejecutado sobre Windows, Linux o Mac.

El lenguaje de programación Java utiliza componentes llamadas

¹ Sincrónicamente: Cuando se ejecutan distintas tareas una seguida de otra sin pausas.

² Front end: Es la interfaz o capa de presentación de un sistema o aplicativo.

Applets los cuales se ejecutan en el navegador, mejorando la interacción gráfica del usuario y el manejo de objetos.

Para desarrollar en Java es necesario el kit de desarrollo o SDK que se lo descarga gratuitamente de la página oficial de Oracle, este SDK tiene un compilador que ayuda con la compilación, interpretación, generación de código, documentación y todo lo necesario para crear una aplicación, este SDK se utiliza en ambientes de desarrollo IDE como Eclipse el cual que facilita la programación.

Por su robustez, Java se usó posteriormente para creación de aplicaciones web con su versión empresarial completa JEE.

3.2.2 Historia

Java obtiene su potencia de su sintaxis, la cual está basada en lenguajes de programación C y C ++.



Ilustración 4 Logo Java

Java nació como una herramienta de programación en 1991 desarrollado por un equipo de trece personas, del cual estaba a cargo James Gosling, este tuvo el nombre inicial de Oak, después cambio de nombre al que en la actualidad ocupa, la ideología de este proyecto era: “Programar solo una vez y ejecutarlo en cualquier ambiente”.

Después de muchas controversias en su evolución, el proyecto tuvo éxito convirtiéndose en uno de los lenguajes de programación más utilizados en el mundo y ejecutándose en muchos de los dispositivos tecnológicos actualmente.

Actualmente el lenguaje de programación Java se encuentra en la versión 8, el cual incluye muchas características que mejoran su rendimiento y lo convierten en uno de los más estables, así como robustos del mercado para realizar aplicaciones.

3.2.3 Características

Las características principales de Java son las siguientes:

- **Simple:** Por la similitud que tiene con lenguajes de programación C y C++, se considera un lenguaje simple y fácil de aprender para el desarrollo.
- **Orientado a objetos:** Java se basa en los conceptos de un lenguaje orientado a objetos, al manejar estructuras encapsuladas para su manipulación.
- **Distribuido:** Permite la creación de aplicaciones distribuidas gracias a la compartición de clases y herramientas en la red.
- **Robusto:** Al ser un lenguaje compilado para la ejecución, permite multi-verificación del código, asegurando así su correcto funcionamiento.
- **Seguro:** Java tiene implementado varias capas de seguridad en sus diferentes entornos, incluido el tiempo de ejecución.
- **Compatible:** Java es un lenguaje de programación el cual se puede ejecutar independiente del sistema operativo que se maneje, puede ser Windows, Unix, Mac, Linux o cualquier sistema operativo móvil.
- **Portable:** Ya que utiliza su Java virtual machine JVM para la ejecución de los programas, se vuelve portable, al poder trabajar sobre cualquier arquitectura.
- **Interpretado y compilado:** Java es un lenguaje de programación que se compila, pero que a la vez puede ser interpretado, de ello su versatilidad, dado que usa byte-code para comunicarse con el sistema operativo.
- **Alto Rendimiento:** Se considera que Java es un lenguaje de programación de alto rendimiento, ya que es de configuración dinámica para su ejecución en la Java virtual machine JVM, además de ser considerado de alto nivel al simplificar el código a

realizar por el desarrollador.

3.2.4 Ventajas

- Es multiplataforma.
- Fácil de entender y aprender.
- Permite configuración de la memoria que se puede utilizar, así se consigue tener varias Java Virtual Machine JVMs ejecutándose al mismo tiempo.
- Tiene una comunidad de desarrollo que se encarga de hacer cada vez más robusto al lenguaje de programación.

3.2.5 Desventajas

- Si la Java Virtual Machine JVM, no está configurada correctamente puede presentar inconvenientes en su rendimiento.
- Su interfaz gráfica GUI utiliza componentes que no permiten mucha de las funcionalidades de un navegador.
- Muchas de las integraciones con bases de datos o repositorios de información no son open source.
- Su propietario Oracle, ralentiza la evolución del lenguaje de programación.

3.3 Bases de Postgres

3.3.1 Definición

Postgres es un motor de administración, centralización, organización y control de información, también conocido como motor de base de datos o entidades relacionadas, es open source, y trabaja sobre multiplataforma Windows, Linux, Mac.

Postgres es uno de los sistemas de bases de datos open source más reconocidos en el mundo debido a su excelente transaccionalidad y su facilidad de implementación con múltiples aplicaciones y lenguajes de programación.

3.3.2 Historia

Postgres nace en 1982 dentro de un proyecto de la Universidad de Berkeley llamado Ingres, con el liderazgo de Michael Stonebraker, Postgres se lleva el mérito de ser uno de los primeros motores de bases de datos prototipo.

Entre diferentes cambios que tuvo el proyecto Ingres, este evolucionó a una versión mejorada la que se denominó post-ingres, luego Postgres, a esto se debe el nombre del motor, así la primera versión fue liberada para la comunidad en 1989, pero no tuvo repercusiones importantes hasta la versión 6.0, liberada en 1996, en la que todas las personas con visión prometedora de los motores de bases de datos, trabajaron en ella y la convirtieron por años en la potencia que es ahora.

3.3.3 Características

- **Alta concurrencia:** Gracias a MVCC, Acceso concurrente multi-versión), el motor de base de datos PostgreSQL consigue que las tablas no se bloqueen mientras otro proceso escribe sobre la misma, de la misma manera se maneja con exactitud los commit hechos por las distintas transacciones y distintos usuarios.
- **Lenguaje SQL:** El lenguaje Sql que maneja es de alto nivel, con un catálogo innumerable de sentencias que mejoran el procesamiento en cálculos.
- **Potencia:** Es el motor de base datos relacionales más potente de los open source.
- **Autenticación:** Soporta Múltiples métodos de autenticación.
- **JSON:** Provee de una serie de características que habilitan la rápida lectura de formatos JSON, para el manejo y comunicación de la información.
- **Postgres EDB:** Es la suite empresarial del motor de base de datos.

3.3.4 Ventajas

- Sin costo de licenciamiento.
- Comunidad que trabaja constantemente en el motor.
- Alta concurrencia.
- Versión empresarial.
- Interactúa con JSON.
- Gran número de documentación para diferentes implementaciones con aplicativos o lenguajes de programación.

3.3.5 Desventajas

- No dispone de soporte oficial.
- No soporta tablespaces, para definir donde ubicar los datos.
- La herramienta para alta disponibilidad de la base de datos tiene precios considerablemente altos.
- Consume muchos recursos comparado con sus similares como MySQL de Oracle.

3.4 Servidor de Aplicaciones WILDFLY

3.4.1 Definición

WildFly es un proyecto de la comunidad, el cual previamente tenía el nombre de JBoss AS, este es un servidor de aplicaciones para la versión empresarial de Java EE, esta netamente programado en Java lo que le da la potencia para trabajar con aplicaciones del mismo lenguaje de programación. Wildfly, al igual que Java puede ser ejecutado sobre cualquier plataforma o sistema operativo como Windows, Linux o Mac, esto lo consigue al trabajar con la Java Virtual Machine JVM.

WildFly tiene la característica de ser software libre y su código es abierto para la constante mejora de la comunidad.

3.4.2 Historia

El servidor de aplicaciones nace de una empresa fundada por Marc Fleury llamada JBoss, después esta fue comprada por la conocida empresa de software Red Hat en 2006, que provisionaría un servidor de aplicaciones con las mismas características pero para uso empresarial, en 2014, el servidor de aplicaciones toma el nombre de Wildfly para convertirse en un proyecto colaborativo, en que la comunidad amplía las funcionalidades y mejora sus características para competir contra grandes como WebSphere de IBM y WebLogic de Oracle.

3.4.3 Características

- Es de código abierto.
- Enfoque Modular, permite la activación, desactivación e integración de las necesidades que se requieran.
- Soporta Rest, JSON y JMX.
- Despliegue rápido de aplicaciones.
- Modificaciones sin necesidad de un redespigüe.
- Soporta clusterización y múltiples nodos.
- Soporta alta disponibilidad.

3.4.4 Ventajas

- Instalación sencilla sobre cualquier sistema operativo.
- Open source.
- Modular.
- Documentación detallada para implementaciones.
- Soporta una consola de administración de fácil uso.

3.4.5 Desventajas

- Integración compleja para arquitecturas orientadas a servicios SOA.
- Versiones empresariales tienen licenciamiento de costo considerablemente alto.
- Soporte solo por parte de la comunidad.

3.5 Capa de Presentación (Primefaces)

3.5.1 Definición

Primefaces es una de las múltiples implementaciones de Java Server Faces JSF, que permiten el desarrollo de la interfaz de usuario o capa Web, este framework de código abierto tiene más de 100 componentes que trabajan de manera adaptable para su uso en dispositivos móviles o en navegadores comunes.

3.5.2 Características

- Simplicidad y mejor rendimiento.
- Intuitivo y de fácil uso.
- Reportes inmediatos de problemas con componentes.
- Tiene una gran comunidad de usuarios con informes de errores y problemas planteados.
- Soporta Ajax y Javascript.
- Diseño innovador.
- Responsive.

3.5.3 Ventajas

- Más de 25 temas para seleccionar plantillas open source.
- Ciclo de vida de JSF fácil de comprender.
- Adaptabilidad a móviles.
- Soporta integración con otras implementaciones de JSF.

3.5.4 Desventajas.

- Las nuevas versiones de componentes tienen diferente implementación que las anteriores y esto no permite la fácil actualización hacia un framework más moderno.

- Nuevos entornos de ejecución para Javascript que mejoran el rendimiento y los componentes de las aplicaciones Web representan una seria competencia.

3.5.5 Adaptación a Móviles desde el navegador

Adaptarse a móviles es la nueva ley para todas las aplicaciones Web, Primefaces provee de su kit de interfaz de usuario PFM PrimeFaces Mobile, la cual utiliza los mismos conceptos para el desarrollo con JSF, pero optimiza estas características para ser visualizada correctamente en dispositivos de bolsillo, este framework está construido en jQuery Mobile y también con las ventajas que ofrece HTML5, lo que optimiza la aplicación en móviles.

CAPÍTULO IV

4. Metodología de Desarrollo Scrum

4.1 Definición

La necesidad de utilizar tecnología en el ciento por ciento de los ámbitos laborales y cotidianos de las personas han dado como resultado la aparición de una cantidad incontable de software nuevo a diario, esto representa un desafío para las empresas las cuales constantemente deben estar innovando en el desarrollo de estas aplicaciones para cumplir sus objetivos de negocio, pero sin descuidar los tiempos en los que se manejan.

En el desarrollo que existe de estas aplicaciones, independientemente del software que se quiera conseguir, existen retos que se presentan a menudo, manejarlos mediante una planificación adecuada, reaccionando a los diferentes cambios que aparezcan dentro del desarrollo, es la función de la metodología agile Scrum.

El enfoque de las metodologías ágiles es promover la toma de decisiones de un proyecto de software, organizando disciplinadamente todos los componentes que participan en el desarrollo de manera colaborativa, para favorecer el rendimiento del equipo.

Scrum se basa en los conceptos de metodologías ágiles para la planificación, ejecución y seguimiento correcto de un proyecto de software.

4.2 Entregas Parciales

Scrum maneja el concepto de entregas parciales regulares, con el objetivo de que el conjunto de estas entregas conforma el producto final, este concepto ayuda a detectar prontamente cualquier cambio que se pueda

solicitar por el cliente, así evitando una posible reestructuración de todo el sistema en un futuro.

Los entregables también refuerzan la confianza del cliente y su satisfacción al obtener rápidas y estructuradas muestras de cómo avanza el desarrollo, las entregas parciales tienen otro valor agregado ya que permite ajustar la planificación de tiempos de desarrollo, ofreciendo flexibilidad y competitividad frente a otras metodologías clásicas de programación de software.

En este mismo concepto, las entregas parciales se utilizan para resolver situación en el caso de que no se esté ofreciendo al cliente lo que está solicitando, cuando se maneja entregas largas, los costes aumentan si llegara a existir esta controversia.

4.3 Características Fundamentales

Con respecto a Scrum, se manejan doce principios los cuales refuerzan y estandarizan la correcta ejecución de la metodología en el proceso de desarrollo de software.

- **Satisfacción del cliente:** El objetivo de todo equipo de desarrollo de software va a ser siempre cumplir con las necesidades y el valor agregado que se puede ofrecer al cliente en el producto a entregar, todos los entregables deben tener la calidad requerida.
- **Nuevos requerimientos:** En el desarrollo de software comúnmente se presentan cambios o nuevos requerimientos sobre la marcha, Scrum nos permite identificarlos a tiempo y adaptarlos en la planificación sin que esto represente un gran cambio en lo ya establecido.
- **Entregas parciales:** Las entregas parciales son unas de las partes claves de Scrum, esto ayuda a la identificación temprana de cualquier inconveniente que se tenga sobre la forma del producto requerido, adicional permite trabajar en diferentes etapas y armar

un flujo de cómo debe desarrollarse el software.

- **Medir el progreso:** Mediante indicadores preestablecidos por la planificación, Scrum permite medir el correcto avance del equipo sobre el proyecto y los inconvenientes que se pueden presentar en el trabajo colaborativo del equipo. La evolución de los procesos no es un elemento subjetivo. Se puede medir con indicadores concretos.
- **Desarrollo sostenible:** La correcta planificación del proyecto de software, garantiza la continuidad del desarrollo, cada etapa tiene su objetivo específico y no se lleva al cabo si no cumple los objetivos parciales y totales del proyecto.
- **Trabajo a la par:** Todos los integrantes del proyecto se encuentran fuertemente vinculados en el proceso colaborativo del desarrollo.
- **Conversación frente a frente:** Mediante reuniones periódicas, los líderes y planificadores del proyecto, se comunican eficazmente cualquier hablando con los miembros del equipo, de esta manera se evita pérdidas de tiempo como, por ejemplo: en cadena de correos electrónicos; también se recomienda tener reuniones periódicas con el cliente para identificar posibles cambios, errores o insatisfacción y así saber remediar cualquiera de estas situaciones a tiempo.
- **Motivación y trabajo en conjunto:** Todos los involucrados en el desarrollo del proyecto tienen que estar alineados a los objetivos a conseguir, manteniendo confianza entre ellos y con la motivación necesaria que les permita trabajar conjuntamente.
- **Diseño:** La estructura del equipo debe estar correctamente diseñada, con las jerarquías y roles necesarios para que los requerimientos no se pierdan o haya situaciones que terminen en controversias, pero a su vez eliminar la burocracia de comunicación para que el equipo tenga acceso a todos sus integrantes y poder trabajar colaborativamente.

- **Simplicidad:** Las actividades o tareas que se asignen a los miembros del equipo tienen que conllevar todo grado de simplicidad posible, si alguna de ellas llega a complicarse, es necesario crear subtareas de esta, hasta cumplirla en su totalidad, de esta manera se reduce el nivel de complejidad y se puede asignar a otro integrante del equipo para colaborar en ella en caso de que se requiera.
- **Organización de los equipos:** Los equipos deben poder organizarse a sí mismos, no excluyendo un monitoreo general de una figura con un rol más alto, esto ayuda a que no crear tanta dependencia entre los integrantes de un equipo entero de desarrollo.
- **Adaptarse a cambios constantes:** Adaptarse rápidamente a cambios es otra de las ventajas de Scrum, los proyectos suelen cambiar durante la marcha y no precisamente terminan de la forma que comenzaron.

4.4 Roles

En Scrum todos los involucrados dentro del proyecto adquieren responsabilidades, las cuales guían al equipo a crear un producto exitoso, la comunicación entre estos miembros y sus actividades están dictadas por roles y características.

- **Product Owner:** Es el propietario del proyecto, el conoce todos los objetivos que el producto debe cumplir y el alcance deseado.
- **Scrum Master:** Es el facilitador dentro de todas las reuniones que tengan que ver con el proyecto, es el encargado de velar por que todas las propuestas ágiles presentadas en la metodología se lleven al cabo exitosamente, además de resolver controversias u obstáculos que se puedan presentar en el camino.

También es el encargado de medir la productividad del equipo, con indicadores que arrojen resultados cumpliendo con cada una de las etapas de Scrum.

- **Equipo:** Es el conjunto de personas multidisciplinarias que, buscan cumplir con el objetivo planteado por el Product Owner, cada uno aporta con sus conocimientos con el fin de apoyar al equipo entero y no solo a sus responsabilidades.

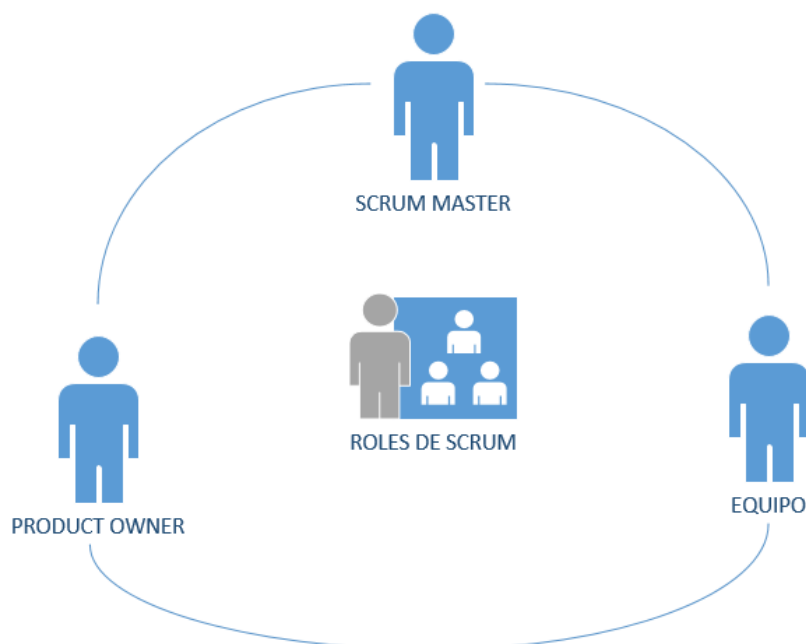


Ilustración 5 Roles en Scrum

4.5 Proceso de Scrum

4.5.1 Introducción

El proceso de Scrum, para el manejo de proyectos, viene dado por la ejecución de bloques que cumplen con tiempos preestablecidos en ser entregados, estos bloques son denominados iteraciones y pueden ser manejados con periodos de entre dos semanas hasta cuatro semanas como máximo, pero no es recordable manejar iteraciones de periodos muy largos.

Cada una de las iteraciones tienen como objetivo entregar una funcionalidad completa que conforma parte del producto de software, estas entregas parciales incrementan las funcionalidades del desarrollo de software y deben ser entregadas al cliente para su posterior prueba y aprobación.

4.5.2 Planificación

La planificación es la primera fase del proceso de Scrum que se realiza, en esta se decide la planificación y el orden de las tareas que se ejecutaran dentro de la iteración, esta es conocida como Sprint Planning y se compone de dos partes fundamentales la cual se lleva al cabo en una reunión:

- **Timebox de requerimientos:** En esta parte de la reunión, que tiene como tiempo máximo unas 4 horas, el cliente expone todos los requerimientos que debe cubrir el software, que son necesarios durante la iteración; luego de esta presentación, el equipo de desarrollo resuelve cualquier duda que se tenga pendiente y lo adapta a los tiempos estimados.
- **Timebox de estrategia:** En esta parte de la reunión que, tiene como tiempo máximo unas 4 horas, el equipo de desarrollo establecerá como se va a llevar al cabo las tareas durante la iteración, a esto es conocido como estrategia de ejecución, se establecen los alcances de la iteración y se crea un pool o lista de tareas a ejecutarse Backlog, cada una con tiempos planificados.

Otro objetivo adicional es que todos los integrantes del equipo estén alineados con la visión del desarrollo, así como la del producto terminado que se entregará para evitar sobre esfuerzos y controversias.

Creando esta planificación, el proceso de Scrum, toma forma y agiliza

la comunicación y productividad que es consecuencia de todos los factores combinados, que cubren los objetivos de la iteración propuesta con el cliente y los miembros del equipo.

4.5.3 Ejecución

Las iteraciones son las partes fundamentales de la ejecución de un proyecto bajo la metodología de Scrum, en contexto un proyecto exitoso es la suma de todas las iteraciones que cumplen exitosamente el alcance propuesto en la fase de planificación.

A esta suma de iteraciones se les conoce como entregas incrementales de producto, de esta manera el cliente puede solicitar un prototipo reducido de funcionalidades para pruebas, adicional puede revisar el estado del software y corregir conceptos en caso de que sea necesario, esto se consigue realizando lo detallado.

Los equipos tienen que realizar constantes reuniones de actualización del trabajo comunicándose entre ellos, aquí se visualizan las tareas pendientes y lo avanzado de manera de ajustar tiempos. En estas reuniones existe un rol muy importante que toma el papel de facilitador, Scrum Master, él toma la tarea de que Scrum esté ejecutándose correctamente y que cada iteración no pierda de vista su alcance.

Es parte importante también del Scrum Master identificar los obstáculos que se vienen presentando y basándose en gráficos con indicadores específicos, corregir estos inconvenientes.

Dentro de las iteraciones se ejecutan las tareas de software que ofrecen más valor agregado al producto, así los avances de las entregas incrementales dan forma al desarrollo y crean la confianza necesaria que se debe tener con el cliente.

Está fuera del objetivo cambiar las tareas que ya se establecieron en la fase de planificación, esto llevaría al desorden del equipo y del avance del producto.

4.5.4 Reunión Diaria

En Scrum la reunión diaria de todos los miembros del equipo, también conocida como Scrum daily meeting, es la parte clave de la metodología.

Las características de esta reunión es cubrir en menos de quince minutos diarios toda la información clave que es necesaria para el avance de las tareas.

Se tocan tres temas fundamentales, las tareas realizadas, las tareas que se están realizando y las tareas a realizar, se exponen estos puntos y se definen los obstáculos que impiden que se lleven al cabo de manera regular, si entre las tareas se define alguna que tiene un grado de dificultad mayor, se establece que otro miembro del equipo puede ayudar.

Las características de las reuniones diarias son las siguientes:

- La puntualidad, las reuniones se realizan a una misma hora y los ausentes tienen una penalidad de incumplimiento.
- Cualquiera persona puede estar dentro de la reunión, pero solo los miembros de equipo pueden comunicarse.
- Se dispone de quince minutos para llevar al cabo la reunión sin importar el número de integrantes del equipo.
- Tienen que tocarse puntos clave ya mencionados, no divagar.
- Se lleva al cabo en un lugar específico para evitar ausencias.
- Cada uno de los integrantes del equipo tiene su turno para participar en la reunión con el fin de evitar el desorden.

Cuando las reuniones diarias cumplen estas características, el Scrum Master identifica correctamente los inconvenientes y los soluciona, todos los integrantes del equipo están al tanto de todo el avance de su equipo y la productividad despierta con el objetivo de cubrir el alcance de cada iteración.

También se identifica el ritmo de trabajo de cada uno de los integrantes mediante indicadores que pueden medir si se tiene un retraso o se está ganando tiempo durante la iteración, los miembros con más experiencia ayudan a los novatos y el ciclo se vuelve completo ya que todos se enriquecen de las habilidades de cada uno de los integrantes del equipo.

Un punto importante que destacar en la reunión es que, los problemas que se identifiquen no se intentan resolver en ese momento, sino que se tratan después, así no se resta tiempo de trabajo al equipo.

Para dar una sensación de que la reunión es ágil, concisa y útil, se la lleva al cabo con los integrantes de pie y con un distintivo que defina de quien es el turno de hablar.

4.5.5 Revisión

La revisión es una reunión informal en la cual el líder de cada equipo presenta cada una de las tareas planificadas en la iteración al cliente, de una manera incremental, los entregables van completando el producto final.

Los requerimientos entregados tienen que cubrir en su totalidad los objetivos de la iteración, es decir cubrir una o varias funcionalidades que agreguen valor y den forma al proyecto.

El cliente tiene la posibilidad de tocar puntos en los cuales no esté satisfecho y ajustar el desarrollo de las funcionalidades, el tiempo máximo para esta reunión es de cuatro horas.

El cliente refuerza su confianza en caso de que los requerimientos cumplan con lo establecido, y en caso contrario se puede identificar prontamente los cambios a realizar y replanificar en la siguiente iteración.

4.5.6 Retrospectiva

Autoevaluarse, esa es la clave de la retrospectiva en Scrum, el equipo analiza a conciencia la forma de trabajo que mantiene durante la iteración y la compara con otras iteraciones pasadas del mismo proyecto u otro con similar estructura, identificando las fortalezas y debilidades, el equipo se pregunta:

- Las actividades que han funcionado bien.
- Las actividades que se deben mejorar.
- Las actividades que se pueden probar en la iteración que viene.
- Lo aprendido en la iteración y la experiencia que pueda servir para la iteración que sigue.
- Los inconvenientes que puedan pasar y sumen negativamente a la siguiente iteración.

Esta reunión tiene un tiempo máximo de tres horas, dependiendo del número de días definidos para la iteración a evaluar.

Es indispensable que el Scrum Master resuelva con los líderes de los equipos los obstáculos identificados, se convierte en un verdadero problema si en cada iteración se notan los mismos problemas.

4.5.7 Afinamiento

Cabe la posibilidad que durante el transcurso de cada una de las iteraciones surjan cambios del cliente, ajustes de tiempo, obstáculos a resolver del equipo, inconvenientes funcionales y no funcionales del desarrollo etc., todos estos cambios pueden ser aumentados, modificados o eliminados de los requerimientos del proyecto y ser afinados, dinamizando el estilo de trabajo de todos los involucrados.

Es indispensable que el cliente tenga claro lo que desea como producto final y lo sepa compartir en las reuniones de planificación, sin llegar a ciclos en los que un cambio se prueba de una y de otra manera con

resultados todos incorrectos.

En este afinamiento también se define la estimación de tiempo nueva, si hay cambios que así lo requieran, afinar la máquina de desarrollo, crea un equipo maduro y experimentado que puede cubrir todos los requerimientos coherentes que el cliente pueda solicitar.

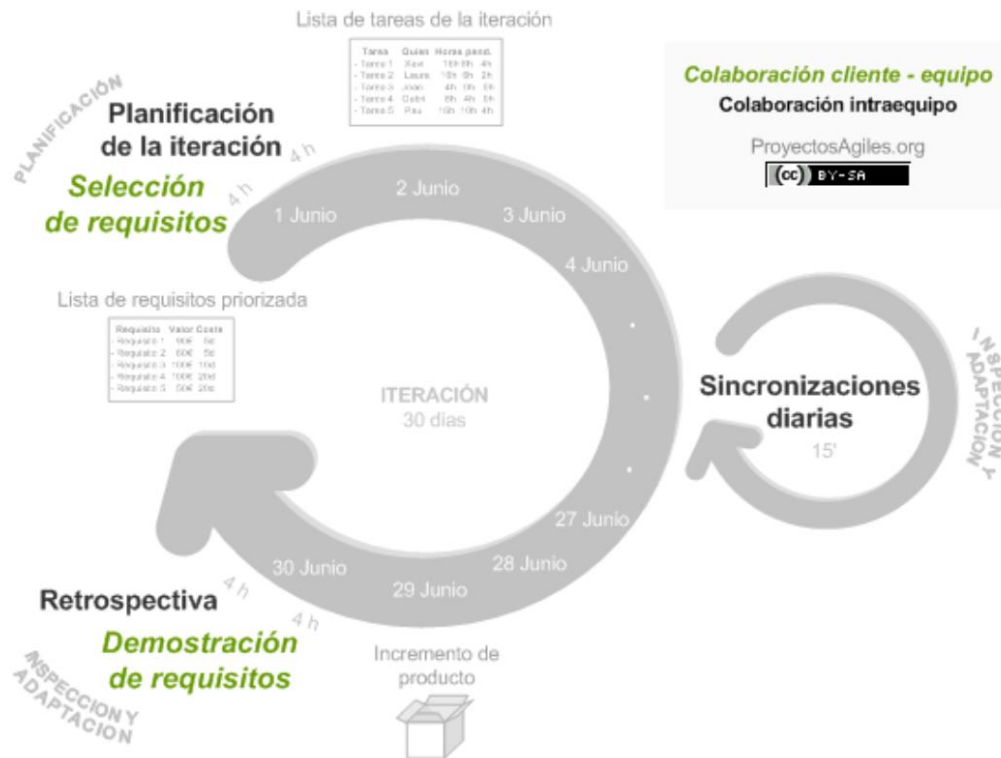


Ilustración 6 Proceso de Scrum Fuente: <https://proyectosagiles.org/como-funciona-scrum/>

CAPÍTULO V

5. Alcance de la Aplicación

5.1 Requerimientos Generales de URBS – PEED en CISEAL

Los requerimientos solicitados por URBS - PEED tienen varios focos de acción a cubrir dentro del software a realizar, el objetivo general a completar es cubrir la trazabilidad de todo el proceso de tratamiento de los plasmas reactivos y no reactivos, esa trazabilidad tiene que ser resuelta por varios objetivos que cumplan con lo necesitado.

5.2 Análisis de Requerimientos de URBS – PEED en CISEAL

Uno de los objetivos a cubrir es la centralización de la información en una sola base de datos, se estaba manejando esta información por medio de archivos Excel el cual permitía tener todos los datos organizados, pero estos archivos son muy vulnerables a cambios involuntarios que afecten la validez de la información.

Estos archivos no disponían de un estándar adecuado para el ingreso de la información, la búsqueda y filtrado de datos, esto causa una pérdida de tiempo para el gestor que está ocupando el archivo y disminuye su productividad, además que los cambios involuntarios de los datos producen que no se mantenga la consistencia con la documentación física.

Otro de los focos a cubrir con el sistema es el control de roles y de acceso a la información, muchos pueden ingresar información, pero no todos pueden editarla, una vez ingresada la información crítica sobre los análisis de plasmas, modificarla no es accesible, por lo cual el manejo de niveles de acceso a las funcionalidades del sistema es clave.

Adicional se requiere auditoría que permita verificar con fechas los cambios y

también se monitorice la información complementaria ingresada sobre las muestras de plasma reactivos y no reactivos, cabe recalcar que el proceso requiere que se trabaje múltiples veces sobre las muestras, ingresando nuevos datos, dependiendo de la información que se añada el sistema tiene distinto comportamiento.

Por último, otro objetivo a satisfacer y el cual cubre la trazabilidad requerida por el sistema, son las estadísticas de toda la información ingresada, mediante gráfico y tablas dinámicas es posible revisar una gran cantidad de información de manera rápida y objetiva, para establecer nuevos medidores que den una perspectiva de como URBS–PEED está trabajando.

5.3 Metodología para el desarrollo de la aplicación web

Para la planificación y el desarrollo del sistema se utilizó la metodología Agile Scrum, esta define una serie de reuniones, alcances, entregables e indicadores para avanzar con el desarrollo de manera eficiente.

5.3.1 Planificación

Se planificó el tiempo estimado para la aplicación y los entregables a realizar en varias reuniones con las personas involucradas:

- Andrés Gonzales: Calidad en URBS - PEED
- Rosa Chiriboga: Directora de URBS - PEED
- Luis Aguas: Tutor de Tesis e Ingeniero en Sistemas y Computación

Con ellos se estableció el alcance que debe tener la aplicación y las funcionalidades que se entregarán en cada entrega para incrementalmente llegar al producto final.

Las funcionalidades por iteración quedaron de la siguiente manera:

ITERACION 1

- Diseño de la aplicación.

ITERACION 2

- Ingreso a la aplicación (Login).
- Diseños de la interfaz de la aplicación.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACIÓN EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

- Diseños de menús.
- Parámetros de configuración.
- Roles de los usuarios.
- Niveles de acceso.
- Administración de Usuarios.

ITERACION 3

- Administración de Instituciones.
- Administración de Marcadores Serológicos.
- Administración de Kits.
- Administración de Códigos Mapas Serología.

ITERACION 4

- Administración de Muestras.
- Muestra: Recepción.
- Muestra: Caracterización de Materia Prima.
- Muestra: Transformación.
- Muestra: Caracterización de Material de Referencia.
- Muestra: Caracterización de Laboratorio de Referencia.
- Muestra: Situación.

ITERACION 5

- Auditoria.
- Reportes.

Con estas funcionalidades, quedó cubierto el alcance de lo solicitado y con todos conformes se procedió al desarrollo.

5.3.2 Entregas Incrementales

Se entregó los prototipos dentro de los plazos de las iteraciones, se realizaba la entrega primero a Scrum Master Luis Aguas para su revisión y conformidad.

Después se entregó los avances a Andrés Gonzales, el cual validaba la funcionalidad de la aplicación con el objetivo de que cumpla con los requerimientos, se ajustaron cambios mínimos los cuales eran revisados

hasta un día posterior a la entrega, estos cambios estaban contemplados dentro del tiempo de la iteración lo cual no significó un atraso, una vez dado el visto bueno se notificaba a la directora de CISEAL.

5.3.3 Reuniones frecuentes y Afinamiento

Se realizaron reuniones constantes con Scrum Master Luis Aguas para detallar algún inconveniente con respecto al desarrollo, el cual fue resuelto inmediatamente.

5.3.4 Entrega del producto final

Se realizó la entrega del desarrollo probando nuevamente toda la funcionalidad del programa, posteriormente se procedió a la instalación, configuración e ingreso de la información en colaboración del personal del URBS-PEED.

CAPÍTULO VI

6. Diseño y Desarrollo de la aplicación web

6.1 Estructura de la aplicación web

La estructura de la aplicación fue basada en el análisis de requerimientos, la planificación realizada y a la metodología llevada al cabo durante el desarrollo, cumpliendo con los estándares de calidad establecidos tanto en forma como en procesos.

La aplicación se estructura cumpliendo los procedimientos necesarios organizados bajo las iteraciones establecidas.

6.2 Diseño de Interfaz

La interfaz de usuario o front-end fue desarrollado utilizando el framework de desarrollo Primfaces, adicional se utilizó una plantilla o tema, propio de la implementación de JSF Primefaces, con el nombre de Ultima.

- Ultima: Es una plantilla de aplicaciones web de categoría premium, este tema es altamente personalizable con cuatro modos diferentes de menú, doce temas, tonalidades personalizables en CSS claro, oscuro, y dos tipos de perfil de usuario. Sumando todas las funcionalidades, se obtienen 384 posibles combinaciones de diseño.
- Es basado en el lenguaje de diseño de materiales o material design, es totalmente sensible, optimizado para táctiles y creado en con SASS, CSS3 y HTML5.

Entre sus características están:

- Completamente adaptable al dispositivo.
- Diseño en base a materiales.
- Completo soporte SaSS.
- Doce temas diferentes para elegir.
- Soportado para móviles.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

- Soportado por la mayoría de navegadores.

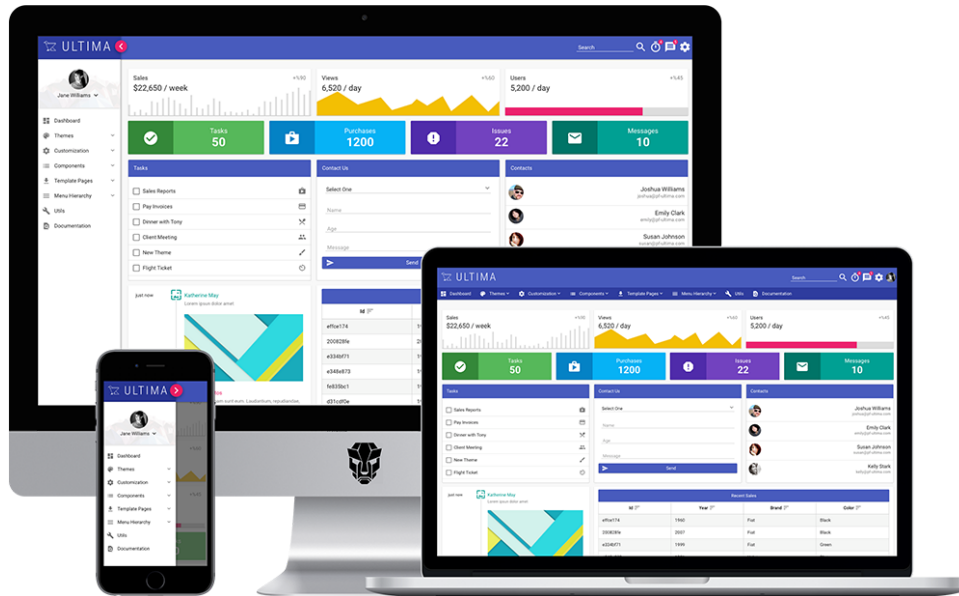


Ilustración 7 Primefaces

6.3 Diseño de Base de Datos

El diseño de base de datos parte como resultado del análisis de requerimientos y es diagramado por las siguientes tablas organizadas con sus respectivas relaciones (entidad – relación).

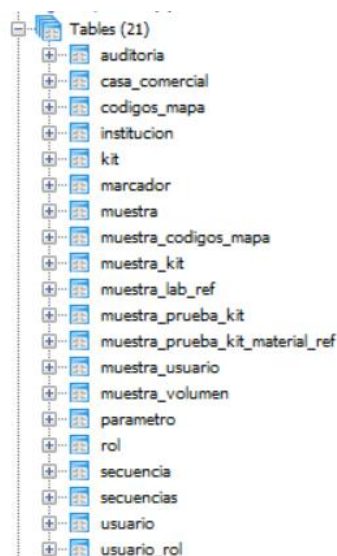


Ilustración 8 PGAdmin - Tablas de Base de datos

Las relaciones de las tablas están establecidas en el diagrama entidad – relación:

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

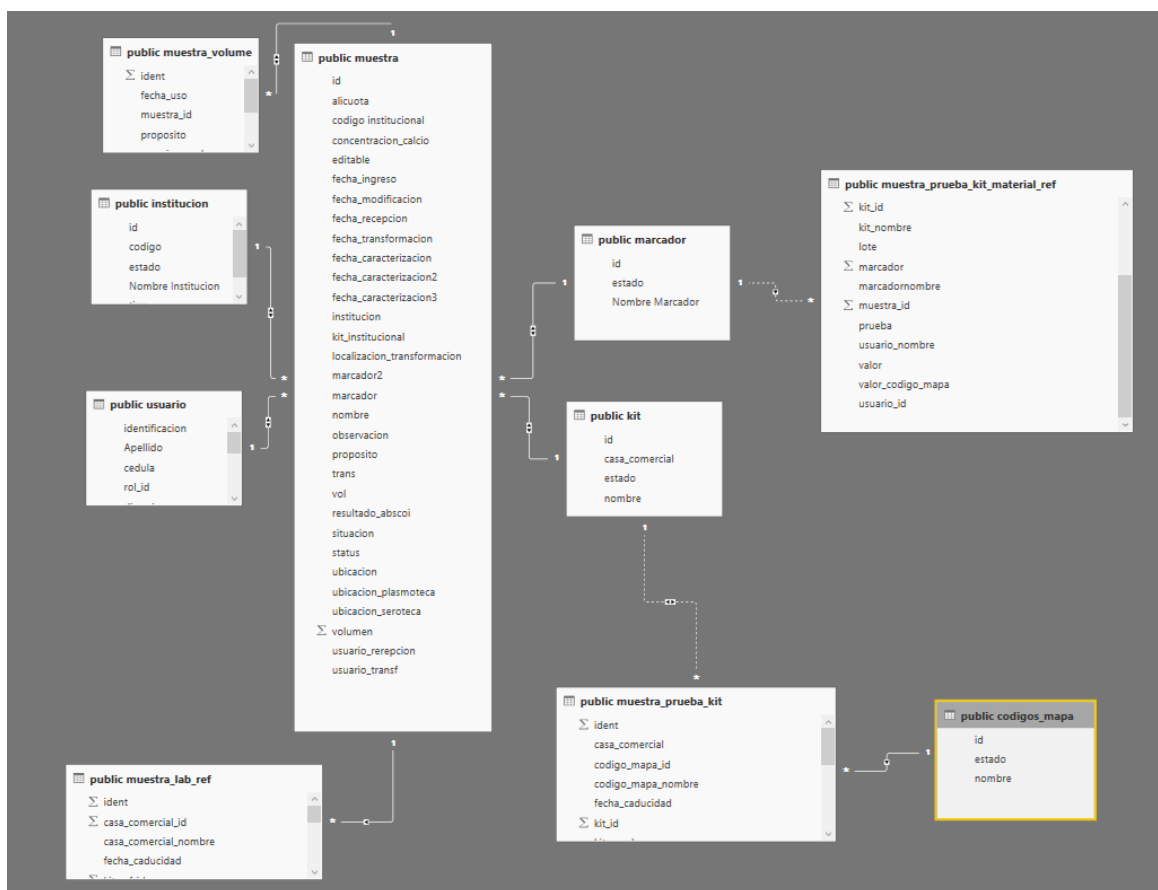


Ilustración 9 Diagrama entidad - relación

6.4 Arquitectura de la Aplicación

6.4.1 Definición de Arquitectura de aplicación

La arquitectura de la aplicación tiene como objetivo separar en capas los diferentes componentes de esta, es una abstracción de como el desarrollo va a trabajar e interactuar con la base de datos, el servidor de aplicaciones y los navegadores.

Bajo las características necesarias para la aplicación web la arquitectura está basada en el modelo n-capas.

6.4.2 Definición de Arquitectura de aplicación N-Capas

Esta está diseñada para mejorar las arquitecturas antiguas y sus limitaciones como lo es cliente-servidor, incrustando una capa en la mitad de la interfaz y del acceso a datos, donde se van a controlar, estandarizar, programar y centralizar todos los procesos del negocio.

Esta arquitectura ofrece también capas extras de seguridad para que los datos sean más complicados de acceder para externos a la aplicación.

- Acceso a datos: En esta capa se accede a la información que está en la base de datos por medio de conectores estandarizados que ofrecen seguridad.
- Lógica de negocios: En esta capa se manipulan todos los datos obtenidos de la capa de acceso a datos, el objetivo es centralizar la información y realizar aquí los procesos como pueden ser cálculos y validaciones.
- Capa de presentación: Esta capa es la que muestra la información a los usuarios y se conectan a la capa de negocios por medio de implementaciones o frameworks.

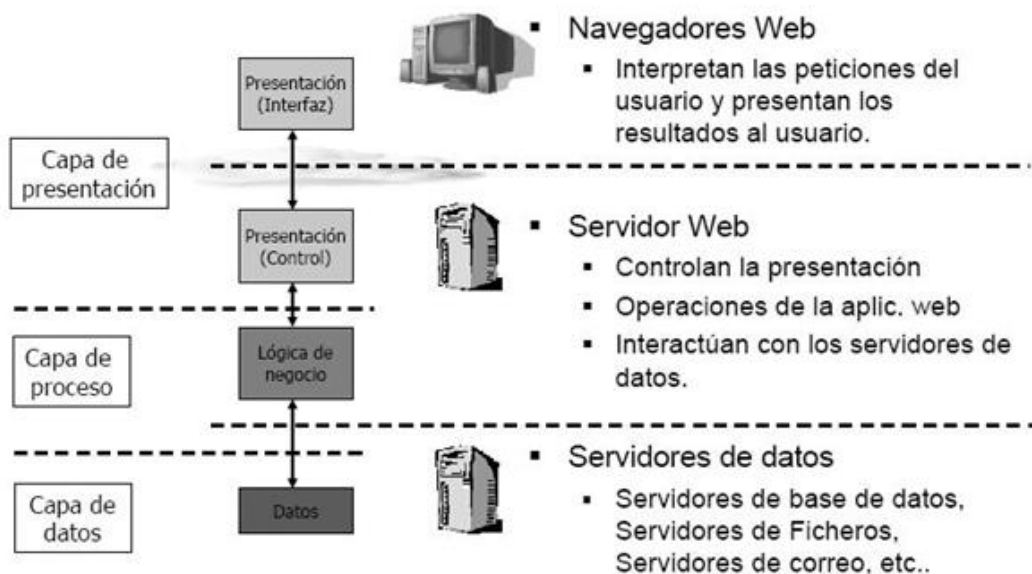


Ilustración 10 Arquitectura N-Capas Fuente: <https://programacionwebisc.wordpress.com/2-1-arquitectura-de-las-aplicaciones-web/>

6.5 Acceso a la aplicación web (Login)

Para acceder a la aplicación web es necesario:

- Estar dentro de la red donde el aplicativo está siendo ejecutado, en este caso es la red de CISEAL ubicado en las instalaciones de la Pontificia Universidad Católica del Ecuador sede Nayón.

- Disponer de un navegador instalado en el equipo.
- Anotar el URL en el navegador:
`http://ciseal4:8080/CISEAL/paginas/index.jsf`

La estructura del URL es la siguiente.

- Http: Es el protocolo estándar de comunicación establecido por la red.
- Ciseal4: Es el equipo o servidor donde está desplegado el aplicativo.
- 8080: Es el puerto por el que se accede al equipo y se consume el servicio o aplicación web.
- CISEAL/paginas/index.jsf: Es la web a visualizar, en este caso representa la pantalla inicial o Login.

Basándose en la arquitectura de aplicaciones web N-Capas se desarrolla el acceso al aplicativo mediante una pantalla que exige credenciales.

Estas credenciales hacen validación con los usuarios registrados en el sistema y verifican su rol para darle acceso a las funcionalidades necesarias.

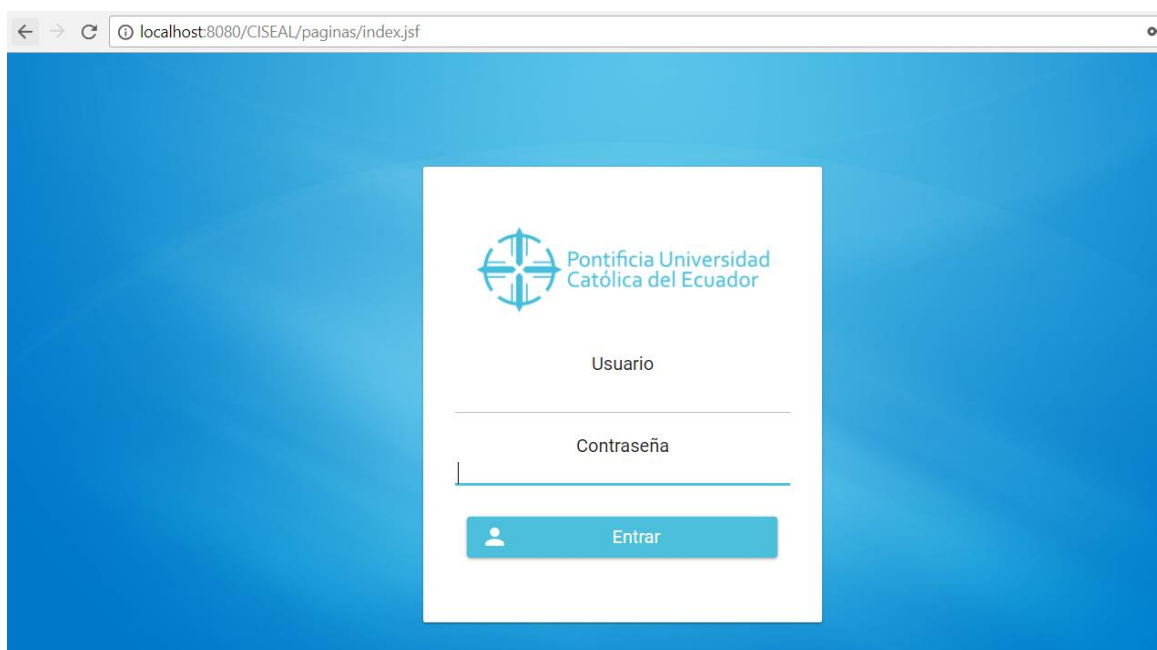


Ilustración 11 Aplicación Web – Login

Ingreso de Texto

- Usuario

- Contraseña

Botón

- Entrar

Funcionalidad

- Ingresar al aplicativo web por medio de credenciales validas, en caso de ingresar credenciales no validas, el aplicativo muestra un mensaje de error.

6.5.1 Roles

Los roles que existen dentro de la aplicación son los siguientes:

- Administrador: Tiene todas las opciones disponibles para configuración, además de todas las funcionalidades habilitadas.

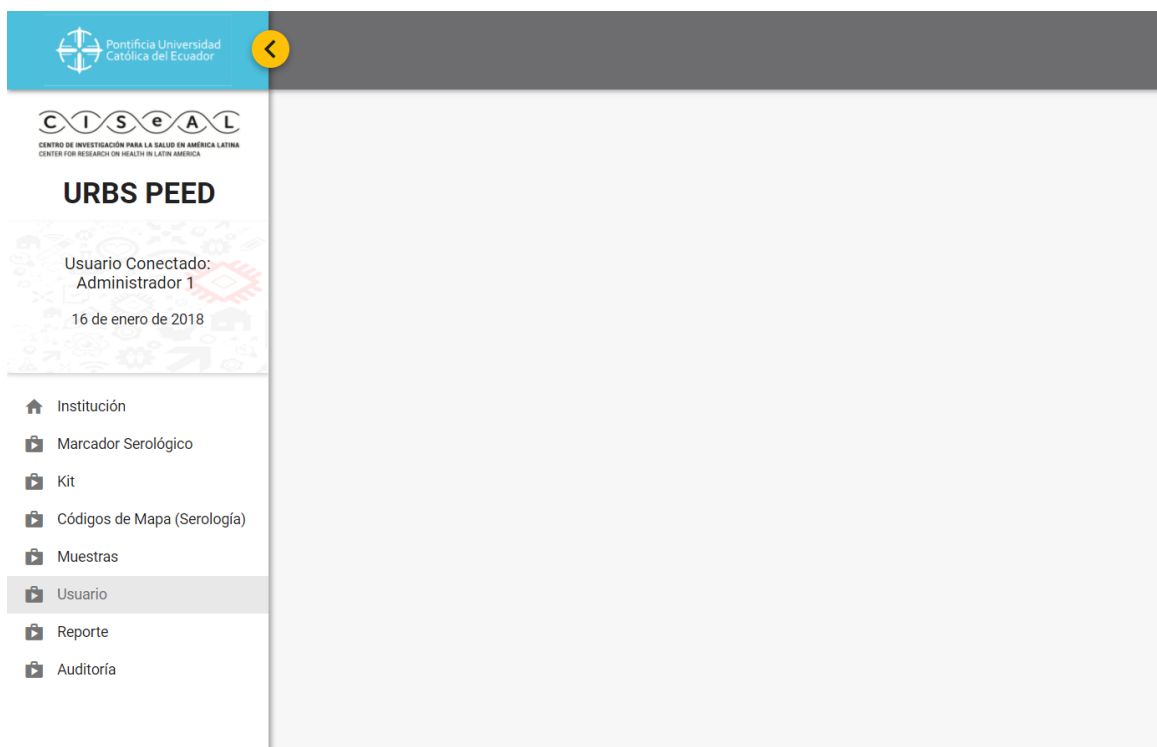


Ilustración 12 Aplicación Web - Pantalla Principal Administrador

- Analista: Tiene funcionalidades restringidas, no puede eliminar muestras.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)



Ilustración 13 Aplicación Web - Pantalla Principal Analista

- Usuario Consulta: Tiene funcionalidades restringidas y adicional de la funcionalidad de reporte, no puede eliminar muestras.

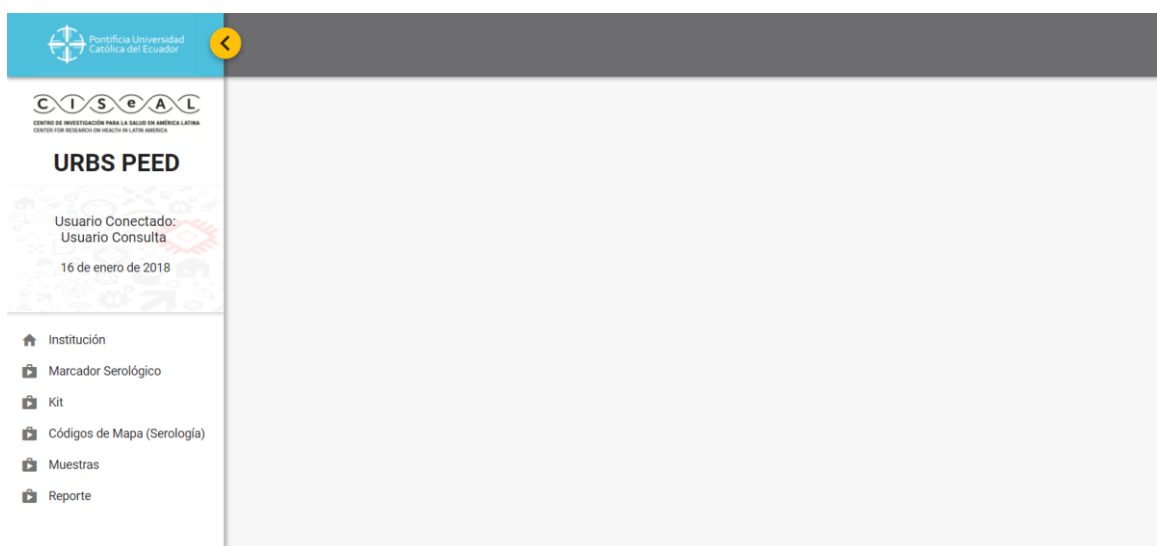


Ilustración 14 Aplicación Web - Pantalla Principal Usuario Consulta

6.6 Pantalla principal de la aplicación web

Una vez ingresado al sistema con las credenciales validas, se despliega la pantalla principal del aplicativo, en esta se despliegan las funcionalidades dependiendo del rol que se tenga.

Dividido en diferentes partes:

- Menú: Es donde se visualizan las administraciones de los procesos.

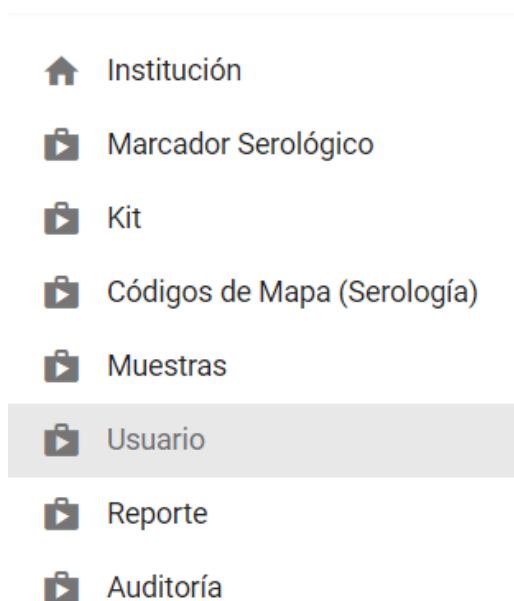


Ilustración 15 Aplicación Web - Menú

- Perfil del usuario: Perfil del usuario conectado e información adicional.



Ilustración 16 Aplicación Web - Perfil de usuario

- Zona de información: Zona donde se visualiza el detalle de las administraciones.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

Nombre	Estado	Editar	Eliminar
Cruz Roja Provincia del Guayas	Activo		
Banco de Sangre de la Junta de Beneficencia de Guayaquil	Activo		
Hemocentro Nacional Cruz Roja Ecuatoriana	Activo		
Hospital de Especialidades Medicas de las Fuerzas Armadas N. 1	Activo		
Hospital Carlos Andrade Marin IESS	Activo		
Hospital Regional Docente Vicente Corral Moscoso	Activo		
Hospital de Especialidades Teodoro Maldonado Carbo	Activo		
Hospital Metropolitano	Activo		
Instituto de Diagnostico y Tratamiento de Enfermedades de la Sangre IDYTES	Activo		
Sociedad de Lucha contra el Cancer SOLCA Nucleo de Cuenca	Activo		

Ilustración 17 Aplicación Web - Zona de Información

- **Cierre de Sesión:** Permite cerrar la sesión del aplicativo.

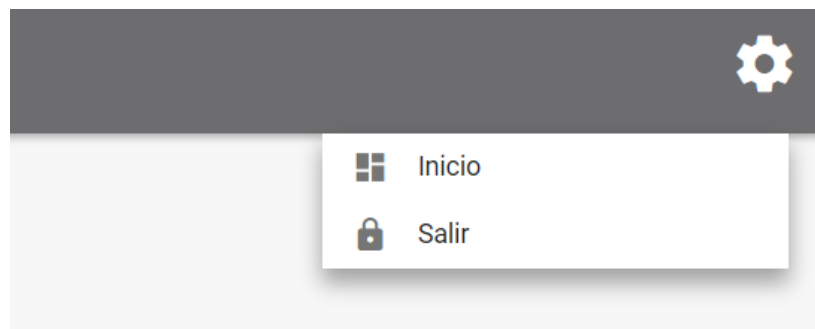


Ilustración 18 Aplicación Web - Cierre de Sesión

6.7 Administración de usuarios

Se administra los usuarios que puede ingresar al sistema, categorizándolos con roles, ellos interactúan con las diferentes funcionalidades dependiendo de su nivel de acceso.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

Usuarios

Nuevo Registro

(1 of 1) << >>

Cédula	Nombres	Apellidos	Estado	Email	Editar	Eliminar
administrador	Administrador	1	Activo	dau@gmail.com		
Analista	Analista	1	Activo	dpaul.mt@gmail.com		
andres	Andres	Gonzalez	Activo	ragonzalez@puce.edu.ec		
dmontalvo	Diego	Montalvo	Activo	dpaul.mt@gmail.com		
usuario	Usuario	Consulta	Activo	dau@gmail.com		

(1 of 1) << >>

Ilustración 19 Aplicación Web - Usuarios Lista

Botón

- Nuevo Registro: Direcciona a la pantalla de ingreso de nuevo usuario.
- Editar: Editar el usuario.
- Eliminar: Eliminar el usuario, este solo se elimina si no tiene relación con otra de las funcionalidades.

Funcionalidad

- Visualizar en una lista todos los usuarios activos o inactivos en el sistema.

Pontificia Universidad Católica del Ecuador

C I S E A L
CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA
CENTER FOR RESEARCH ON HEALTH IN LATIN AMERICA

URBS PEED

Usuario Conectado:
Administrador 1

16 de enero de 2018

Nombre de Usuario*: _____
Cédula*: _____
Nombres: _____
Apellidos: _____
Contraseña*: _____
Dirección Domiciliaria: _____
Email*: _____
Rol*: ADMINISTRADOR ▾
Estado*: Activo ▾

Please enter a password

* Los campos con asterisco son obligatorios.

Guardar Atras

Ilustración 20 Aplicación Web - Ingreso de Usuario

Campos

- Nombre de usuario: Nombre de usuario único en el sistema.
- Cédula: Tiene control de validación de cédula.
- Nombres, Apellidos, dirección, rol.
- Contraseña: Tiene validación de robustez de contraseña.
- Email: Tiene validación de validez de correo electrónico.
- Estado: Activo e inactivo.

Botón

- Guardar, Atras.

Funcionalidad

- Ingresar un nuevo usuario en el sistema.

6.8 Administración de Instituciones

Se administra las instituciones o bancos de sangre las cuales proveen muestras de plasmas activos o no reactivos al URBS-PEED.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

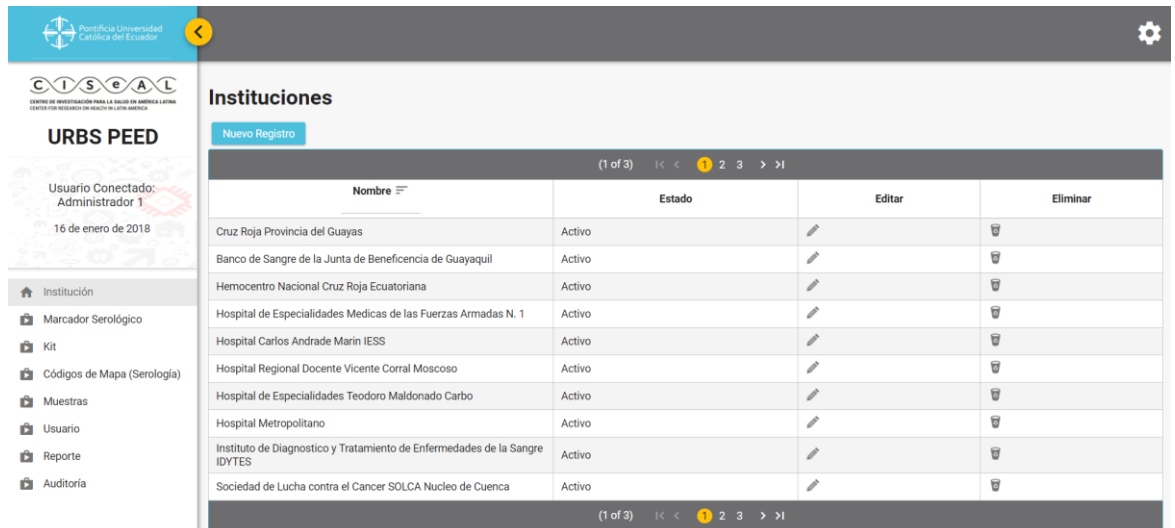


Ilustración 21 Aplicación Web - Instituciones Lista

Filtro

- Nombre

Botón

- Nuevo Registro: Direcciona a la pantalla de ingreso de nueva institución.
- Editar
- Eliminar: Eliminar institución, este solo se elimina si no tiene relación con otra de las funcionalidades.

Funcionalidad

- Visualizar en una lista todos las instituciones activas o inactivas en el sistema.

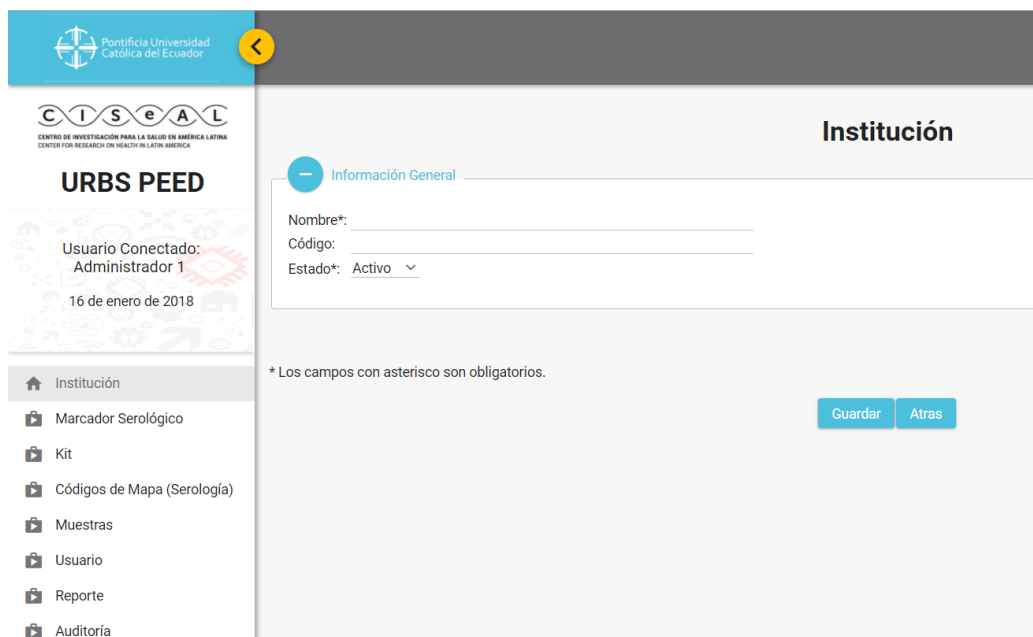


Ilustración 22 Aplicación Web - Ingreso de Institución

Campos

- Nombre, código.
- Estado: Activo e inactivo.

Botón

- Guardar, Atrás.

Botón

- Ingresar una nueva institución en el sistema.

6.9 Administración de Marcadores Serológicos

Se administran los marcadores serológicos, sobre los cuales se clasifican las muestras de plasma reactivos o no reactivos, los marcadores son establecidos por normativas del país y por URBS-PEED.

Nombre	Estado	Editar	Eliminar
HIV	Activo		
HCV	Activo		
HBsAg	Activo		
Chagas	Activo		
Sífilis	Activo		
Anti - HBs	Activo		
Anti - HBc	Activo		
Negativos	Activo		

Ilustración 23 Aplicación Web - Marcadores Serológicos Lista

Filtro

- Nombre

Botón

- Nuevo Registro: Direcciona a la pantalla de ingreso de nuevo marcador serológico.
- Editar
- Eliminar: Eliminar marcador serológico, este solo se elimina si no tiene relación con otra de las funcionalidades.

Funcionalidad

- Visualizar en una lista todos los marcadores serológicos activos o inactivos en el sistema.

The screenshot shows a web application interface. At the top left, there is a header for 'Pontificia Universidad Católica del Ecuador' and a logo for 'CISEAL' (Centro de Investigación para la Salud en América Latina). Below this, the text 'URBS PEED' is displayed, along with 'Usuario Conectado: Administrador 1' and the date '16 de enero de 2018'. A sidebar menu on the left contains items: 'Institución', 'Marcador Serológico' (highlighted), 'Kit', 'Códigos de Mapa (Serología)', 'Muestras', 'Usuario', 'Reporte', and 'Auditoría'. The main content area is titled 'Marcador Serológico' and contains a form with a section 'Información General'. The form has a 'Nombre*' field and an 'Estado*' dropdown menu currently set to 'Activo'. A note below the form states '* Los campos con asterisco son obligatorios.' At the bottom right of the form are two buttons: 'Guardar' and 'Atras'.

Ilustración 24 Aplicación Web - Ingreso Marcador Serológico

Campos

- Nombre
- Estado: Activo e inactivo.

Botón

- Guardar, Atrás.

Botón

- Ingresar un nuevo marcador serológico en el sistema.

6.10 Administración de Kit de Reactivos

Se administran los kits de reactivos estandarizados y proporcionados por varios laboratorios reconocidos internacionalmente como localmente a estos se les denomina casas comerciales, estos reactivos interactúan con las muestras de plasmas para determinar si el marcador serológico establecido por los bancos de sangre es correcto, el personal del URBS-PEED, con del rol de pre-análisis es el encargado de esta interacción.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

Nombre	Casa Comercial	Estado	Editar	Eliminar
HIV Combi PT	ROCHE	Activo		
Anti - HCV II	ROCHE	Activo		
HBsAg II	ROCHE	Activo		
Chagas	ROCHE	Activo		
Syphilis	ROCHE	Activo		
Anti - Hbc	ROCHE	Activo		
Anti - HBs II	ROCHE	Activo		
VIH Ag/Ab Combo Architect	Abbott	Activo		
Anti - HCV Architect	Abbott	Activo		
HBsAg Qualitative II Architect	Abbott	Activo		

Ilustración 25 Aplicación Web - Kit Lista

Filtro

- Nombre, casa comercial.

Botón

- Nuevo Registro: Direcciona a la pantalla de ingreso de nuevo kit de reactivo.
- Editar
- Eliminar: Eliminar el kit de reactivo, este solo se elimina si no tiene relación con otra de las funcionalidades.

Funcionalidad

- Visualizar en una lista todos los kits de reactivos activos o inactivos en el sistema.

Kit de Reactivo

— Información General

Nombre: _____

Casa Comercial: _____

Estado*: Activo ▼

* Los campos con asterisco son obligatorios.

[Guardar](#) [Atras](#)

Ilustración 26 Aplicación Web - Ingreso Kit

Campos

- Nombre, casa comercial.
- Estado: Activo e inactivo.

Botón

- Guardar, Atrás.

Botón

- Ingresar un nuevo kit de reactivo en el sistema.

6.11 Administración de Códigos de Mapa de Trabajo de Serología

Se administran los códigos de mapas de trabajo de serología que son donde las muestras de plasmas reactivos y no reactivos son ubicadas, en estas localizaciones se dispone del plasma y están debidamente documentadas para cumplir con el proceso de caracterización llevado al cabo por el URBS-PEED.

Nombre	Estado	Editar	Eliminar
GLA-R3 HIV	Activo		
GLA-R3 HCV	Activo		
GLA-R3 HBsAg	Activo		
GLA-R3 Chagas	Activo		
GLA-R3 Sífilis	Activo		
GLA-R3 Anti-HBc	Activo		

Ilustración 27 Aplicación Web - Códigos de Mapa de Trabajo de Serología Lista

Filtro

- Nombre

Botón

- Nuevo Registro: Direcciona a la pantalla de ingreso de nuevo código de mapas de trabajo de serología.
- Editar

- Eliminar: Eliminar el código de mapa de trabajo de serología, este solo se elimina si no tiene relación con otra de las funcionalidades.

Funcionalidad

- Visualizar en una lista todos los códigos de mapas de trabajo de serología activos o inactivos en el sistema.



Ilustración 28 Aplicación Web - Códigos de Mapa de Trabajo de Serología

Campos

- Nombre
- Estado: Activo e inactivo.

Botón

- Guardar, Atrás.

Botón

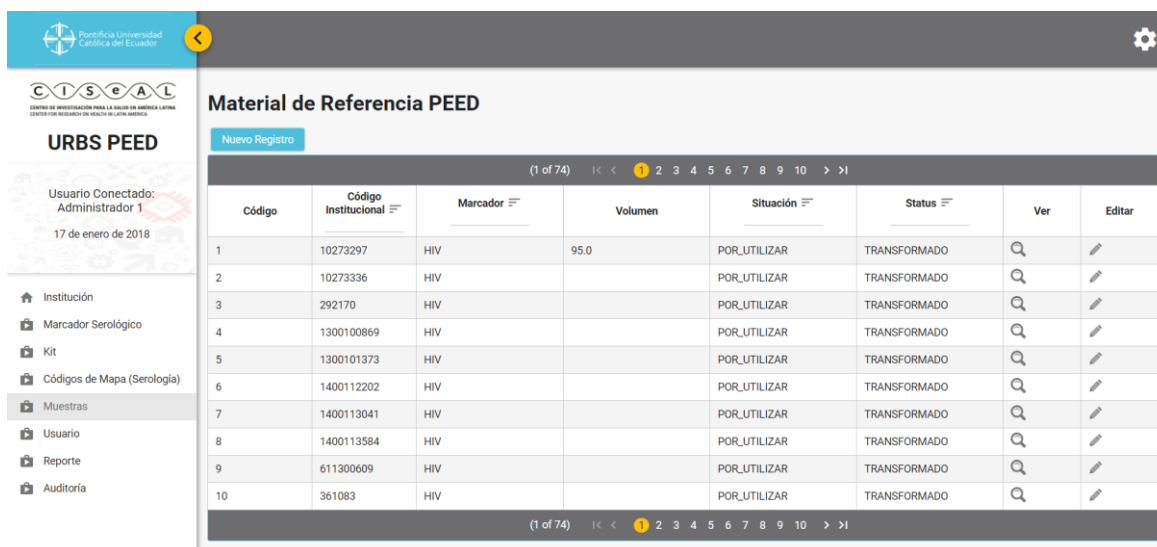
- Ingresar un nuevo código de mapa de trabajo de serología en el sistema.

6.12 Administración y procesos del material de referencia de URBS-PEED

La centralización, organización y seguimiento de la información del material de referencia en URBS-PEED es la parte principal del aplicativo, creando una trazabilidad completa para los procesos de pre-análisis, análisis y post-análisis, las muestras pasan por varias etapas dentro del laboratorio, y toda esta información correctamente relacionada debe

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

plasmarse en el aplicativo para que conserven esta relación entre los datos.



The screenshot displays the 'Material de Referencia PEED' interface. On the left, there is a sidebar with the CISEAL logo and navigation options: 'URBS PEED', 'Institución', 'Marcador Serológico', 'Kit', 'Códigos de Mapa (Serología)', 'Muestras', 'Usuario', 'Reporte', and 'Auditoría'. The main area shows a table with 10 records. The table has columns for 'Código', 'Código Institucional', 'Marcador', 'Volumen', 'Situación', 'Status', 'Ver', and 'Editar'. The records are numbered 1 through 10, with the first record having a volume of 95.0. The status for all records is 'TRANSFORMADO' and the situation is 'POR UTILIZAR'. There are search and edit icons for each row.

Código	Código Institucional	Marcador	Volumen	Situación	Status	Ver	Editar
1	10273297	HIV	95.0	POR_UTILIZAR	TRANSFORMADO	🔍	✎
2	10273336	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
3	292170	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
4	1300100869	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
5	1300101373	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
6	1400112202	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
7	1400113041	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
8	1400113584	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
9	611300609	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎
10	361083	HIV		POR_UTILIZAR	TRANSFORMADO	🔍	✎

Ilustración 29 Aplicación Web - Material de Referencia Lista

Filtro

- Código institucional, marcador serológico, situación, status.

Botón

- Nuevo Registro: Direcciona a la pantalla de ingreso de nuevo material de referencia.
- Editar
- Ver

Funcionalidad

- Visualizar en una lista todos los códigos de mapas de trabajo de serología activos o inactivos en el sistema.

En ingreso del material de referencia, se compone de distintas etapas que conforman los procesos de URBS-PEED, a continuación.

6.12.1 Recepción de Material de Referencia

La recepción del material de referencia consiste en hacer el ingreso del plasma reactivo o no reactivo en URBS-PEED, este plasma viene de algún banco de sangres del país, y trae consigo una serie de datos que son matriculados en el aplicativo.

Recepción

Analista Recepción: Administrador 1

Fecha recepción:

Institución*:

Código Institucional:

Marcador Serológico*: Seleccione

Ubicación:

Ubicación Plasmoteca:

RESULTADOS INSTITUCIÓN:

Abs/COI:

Kit: Seleccione

Casa Comercial:

Ilustración 30 Aplicación Web - Material de referencia - Recepción

Etiquetas

- Analista Recepción: Corresponde al analista que hace el ingreso de la información en el sistema, el nombre de este concuerda con el usuario que se encuentra dentro del sistema en ese momento.
- Casa comercial

Campos

- Fecha recepción, código institucional, ubicación, ubicación plasmoteca, Abs/COI.
- Institución: Se selecciona una de las instituciones o bancos de sangre creados previamente en la administración de instituciones.
- Marcador Serológico: Se selecciona uno de los marcadores serológicos creados previamente en la administración de marcadores serológicos.
- Kit: Se selecciona uno de los Kits de reactivos creados previamente en la administración de kits de reactivos.

Botón

- Guardar, Atrás, seleccionar institución.
- Eliminar: Solo los roles de administración pueden eliminar una de las muestras.

6.12.2 Caracterización de Materia Prima

La caracterización de materia prima consiste en realizar diferentes pruebas de la materia prima y obtener su información con respecto a los reactivos utilizados, esta información es ingresada en el sistema para su posterior análisis, en la tabla desplegada en el final de la pantalla se almacenan cada una de las pruebas realizadas en la caracterización.

Marcador Serológico	Kit	Casa Comercial	Prueba	Valor	Lote	Fecha Caducidad	Usuario	Código Mapa	Valor	Eliminar
No existen registros										

Ilustración 31 Aplicación Web - Material de referencia - Caracterización de Materia Prima

Etiquetas

- **Analista:** Corresponde al analista que hace el ingreso de la información en el sistema, el nombre de este concuerda con el usuario que se encuentra dentro del sistema en ese momento.

Campos

- **Prueba, valor prueba, lote, fecha caducidad, valor código mapa de serología, fecha caracterización.**
- **Marcador Serológico:** Se selecciona uno de los marcadores serológicos creados previamente en la administración de marcadores serológicos.
- **Kit:** Se selecciona uno de los Kits de reactivos creados previamente en la administración de kits de reactivos.
- **Código mapa:** Se selecciona uno de los códigos de mapa de serología creados previamente en la administración de códigos de mapa de serología.

Botón

- **Guardar, Atrás, agregar prueba.**

- Eliminar: Solo los roles de administración pueden eliminar una de las muestras.

6.12.3 Caracterización de Material de Referencia

La caracterización de material de referencia consiste en realizar múltiples pruebas de la materia con todos los reactivos correspondientes a cada uno de los marcadores serológicos, para verificar su resultado, esta información es ingresada en el sistema para su posterior análisis, en la tabla desplegada en el final de la pantalla se almacenan cada una de las pruebas realizadas en la caracterización.

The screenshot shows the 'Caracterización de Material de Referencia' web application. The header includes navigation tabs: 'Recepción', 'Caracterización Materia Prima', 'Transformación', 'Caracterización de Material de Referencia' (active), 'Caracterización Laboratorio de Referencia', and 'Situación'. The left sidebar contains a menu with items: 'Institución', 'Marcador Serológico', 'Kit', 'Códigos de Mapa (Serología)', 'Muestras', 'Usuario', 'Reporte', and 'Auditoría'. The main content area features a form for 'Caracterización' with fields for 'Analista: Administrador 1', 'Fecha Caracterización', 'Marcador Serológico*', 'Kit', 'Prueba #', 'Valor Prueba', 'Lote', 'Fecha caducidad', 'Código de Mapa', and 'Valor Código de Mapa'. A blue button labeled 'Agregar Caracterización' is positioned to the right of the 'Valor Prueba' field. Below the form is a table with 11 columns: 'Marcador Serológico', 'Kit', 'Casa Comercial', 'Prueba', 'Valor', 'Lote', 'Fecha Caducidad', 'Usuario', 'Código Mapa', 'Valor', and 'Eliminar'. The table currently displays 'No existen registros'.

Ilustración 32 Aplicación Web - Material de referencia - Caracterización de Material de Referencia

Etiquetas

- Analista: Corresponde al analista que hace el ingreso de la información en el sistema, el nombre de este concuerda con el usuario que se encuentra dentro del sistema en ese momento.

Campos

- Prueba, valor prueba, lote, fecha caducidad, valor código mapa de serología, fecha caracterización.
- Marcador Serológico: Se selecciona uno de los marcadores serológicos creados previamente en la administración de marcadores serológicos.
- Kit: Se selecciona uno de los Kits de reactivos creados previamente en la administración de kits de reactivos.

- Código mapa: Se selecciona uno de los códigos de mapa de serología creados previamente en la administración de códigos de mapa de serología.

Botón

- Guardar, Atrás, agregar caracterización.
- Eliminar: Solo los roles de administración pueden eliminar una de las muestras.

6.12.4 Caracterización de Laboratorio de Referencia

La caracterización de laboratorio de referencia consiste en realizar múltiples pruebas de la materia con todos los reactivos correspondientes a cada uno de los marcadores serológicos para verificar su resultado, esta información es ingresada en el sistema para su posterior análisis, en la tabla desplegada en el final de la pantalla se almacenan cada una de las pruebas realizadas en la caracterización.

The screenshot displays the 'Caracterización Laboratorio de Referencia' web application. The interface includes a sidebar with navigation options such as 'Institución', 'Marcador Serológico', 'Kit', 'Códigos de Mapa (Serología)', 'Muestras', 'Usuario', 'Reporte', and 'Auditoría'. The main content area shows a form for entering test data, including fields for 'Marcador Serológico', 'Kit', 'Lote', 'Valor', 'Fecha caducidad', and 'Interpretación'. Below the form is a table with columns: 'Marcador Serológico', 'Kit', 'Casa Comercial', 'Lote', 'Valor', 'Fecha Caducidad', 'Usuario', 'Resultado', and 'Eliminar'. The table currently shows 'No existen registros'.

Ilustración 33 Aplicación Web - Material de referencia - Caracterización Laboratorio de Referencia

Etiquetas

- Analista: Corresponde al analista que hace el ingreso de la información en el sistema, el nombre de este concuerda con el usuario que se encuentra dentro del sistema en ese momento.

Campos

- Valor, lote, fecha caducidad, fecha caracterización.

- Marcador Serológico: Se selecciona uno de los marcadores serológicos creados previamente en la administración de marcadores serológicos.
- Kit: Se selecciona uno de los Kits de reactivos creados previamente en la administración de kits de reactivos.
- Interpretación: reactivo, no reactivo.

Botón

- Guardar, Atrás, agregar caracterización.
- Eliminar: Solo los roles de administración pueden eliminar una de las muestras.

6.12.5 Transformación

La transformación de la materia prima consiste en la información del status del plasma, esta información es ingresada en el sistema para su posterior análisis.

Ilustración 34 Aplicación Web - Material de referencia - Transformación

Etiquetas

- Analista: Corresponde al analista que hace el ingreso de la información en el sistema, el nombre de este concuerda con el usuario que se encuentra dentro del sistema en ese momento.

Campos

- Fecha transformación, localización transformación, ubicación seroteca.
- Volumen: Valida números y está en medida de mililitros (ml).

- Concentración de calcio: Valida números y está en medida de miligramos sobre decilitros (mg/dl).

Botón

- Guardar, Atrás.
- Eliminar: Solo los roles de administración pueden eliminar una de las muestras.
- Status: Pendiente, desechado, transformado.

6.12.6 Situación

La situación de la materia prima consiste en la información de la cantidad de plasma que está disponible y localizado, esta información es ingresada en el sistema para su posterior análisis, en la tabla desplegada en el final de la pantalla se almacenan cada uno de los usos y propósitos que se le dio al plasma.

Observación

Observaciones:

Situación: Por Utilizar Terminado

Volumen

Volumen 2,323.00 ml

Fecha uso:

Propósito Agregar Uso

Volumen a Usar

Propósito	Fecha Uso	Volumen Usado	Usuario
No existen registros			

Fecha Ingreso: 17-01-18
Fecha Última Modificación: 17-01-18

Ilustración 35 Aplicación Web - Material de referencia - Situación

Etiquetas

- Volumen: Corresponde al volumen del plasma que se encuentra disponible en ese momento.

Campos

- Fecha uso, propósito, observaciones.
- Volumen por usar: Valida números y está en medida de mililitros (ml).

Botón

- Guardar, Atrás, agregar uso.
- Eliminar: Solo los roles de administración pueden eliminar una de las muestras.
- Situación: Por utilizar, terminado.

6.13 Auditoría

La auditoría consiste en el seguimiento a todo nivel de un proceso, en este caso toda operación llevada al cabo en el sistema guarda un historial de cambios, en este se visualiza, quien hizo el cambio, que tipo de cambio, la IP desde el computador que se conectó, la hora y fecha de la modificación, así se controla en alto nivel cualquier modificación en los datos.

Código	Acción	Fecha	IP	Nombre	Registro
1	crear	2017-10-23 12:08:49.912	192.168.132.84	andres	Muestra [d=1, nombre=null, fechaRecepcion=Thu Sep 18 00:00:00 COT 2014, fechaIngreso=Mon Oct 23 12:07:41 COT 2017, fechaModificacion=Mon Oct 23 12:08:49 COT 2017, institucion=7, codigoInstitucional=10273297, marcadorSero=1, Marcador=null, usuarioRecepcion=Usuario[identificacion=andres, nombre=Andres, apellido=Gonzalez, estado=ACT, direccion=Las Casas, email=ragonzalez@puce.edu.ec, registroNuevo=false], roles=[], usuarioTranf=null, marcador2=, ubicacion=, ubicacionPlasmoteca=D1-2/2, fechaTransformacion=null, localizacionTransformacion=null, ubicacionSeroteca=null, concentracionCalcio=null, volumen=null, resAbsCoi=, fechacaracterizacion=null, fechacaracterizacion2=null, fechacaracterizacion3=null, muestraLabRef=null, muestraVolumen=null, observacion=, situacion=POR_UTILIZAR, status=PENDIENTE, muestraPruebaKit=null, muestraPruebaKitMaterialRef=null, editable=true, registroNuevoTransformacion=false, registroVol=false, kitInstitucional=null]
1	actualizar	2017-10-23 12:10:42.862	192.168.132.84	andres	Muestra [d=1, nombre=null, fechaRecepcion=Thu Sep 18 00:00:00 COT 2014, fechaIngreso=2017-10-23 12:07:41.944, fechaModificacion=Mon Oct 23 12:10:42 COT 2017, institucion=7, codigoInstitucional=10273297, marcadorSero=1, Marcador=ec.edu.puce.ciseal.modelo.Marcador@624677, usuarioRecepcion=Usuario[identificacion=andres, nombre=Andres, apellido=Gonzalez, estado=ACT, direccion=Las Casas, email=ragonzalez@puce.edu.ec, registroNuevo=false], roles=[], usuarioTranf=null, marcador2=, ubicacion=, ubicacionPlasmoteca=D1-2/2, fechaTransformacion=null, localizacionTransformacion=null, ubicacionSeroteca=null, concentracionCalcio=null, volumen=null, resAbsCoi=, fechacaracterizacion=Thu Oct 17 00:00:00 COT 2013, fechacaracterizacion2=null, fechacaracterizacion3=null, muestraLabRef=null, muestraVolumen=null, observacion=, situacion=POR_UTILIZAR, status=PENDIENTE, muestraPruebaKit=null, muestraPruebaKitMaterialRef=null, editable=true, registroNuevoTransformacion=false, registroVol=false, kitInstitucional=null]

Ilustración 36 Aplicación Web - Auditoría

Filtro

- Código, acción, fecha, IP, nombre.

Funcionalidad

- Visualizar en una lista de todos los cambios hechos sobre los plasmas.

6.14 Reporte (Estadísticas)

Los reportes ofrecen una visión general de toda la información ingresada por las muestras de plasma reactivos y no reactivos, en esto reportes se

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

puede visualizar las estadísticas necesarias para la trazabilidad requerida.

REPORTE ESTADISTICO

Nombre Institucion

Nombre Apellido

Andres Gonzalez

Diego Montalvo

Veronica Aguaiza

situacion

POR_UTILIZAR

Nombre Marcador

fecha_recepcion

9/25/2004 8/25/2017

status

PENDIENTE

TRANSFORMADO

codigo institucional	fecha_recepcion	situacion	status	volumen
101503630	10/15/2015 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	
101503637	10/15/2015 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	
101600244	7/7/2016 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	145.00
10267118		POR_UTILIZAR	PENDIENTE	
10267192		POR_UTILIZAR	PENDIENTE	
10267257		POR_UTILIZAR	PENDIENTE	
10267336		POR_UTILIZAR	PENDIENTE	
10268316	7/22/2013 12:00:00 AM	POR_UTILIZAR	PENDIENTE	
10270180	7/22/2013 12:00:00 AM	POR_UTILIZAR	PENDIENTE	
10270835	7/22/2013 12:00:00 AM	POR_UTILIZAR	PENDIENTE	
10272566		POR_UTILIZAR	PENDIENTE	
10273297	9/18/2014 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	15.00
10273336	9/18/2014 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	
12434		POR_UTILIZAR	TRANSFORMADO	
1300100869	9/18/2014 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	
1300101373	9/18/2014 12:00:00 AM	POR_UTILIZAR	TRANSFORMADO	

Ilustración 37 Aplicación Web - Reporte Pag 1

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

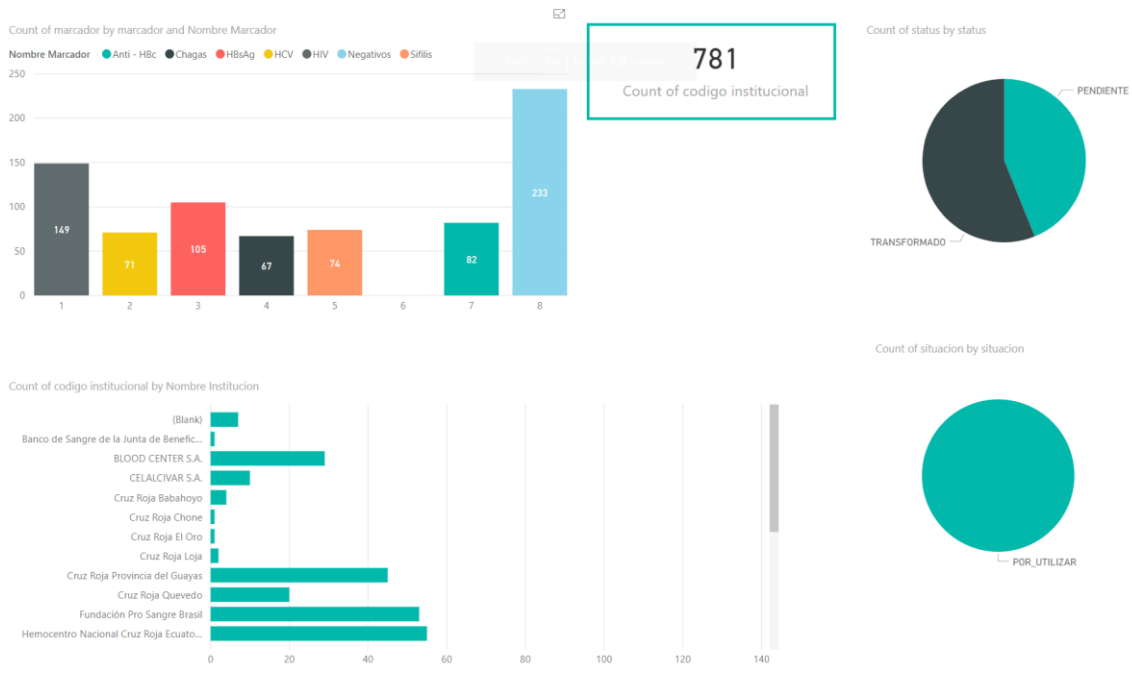


Ilustración 38 Aplicación Web - Reporte Pag 12

Filtro

- Código, acción, fecha, IP, nombre.

Funcionalidad

- Visualizar en una lista de todos los cambios hechos sobre los plasmas.

El reporte ofrece características dinámicas de interacción con los datos con el objetivo de que las estadísticas no sean confusas, sino que se puedan trabajar individualmente dependiendo de lo requerido obteniendo una verdadera trazabilidad.

CAPÍTULO VII

7. Instalación de la aplicación web

Previamente a la instalación de la aplicación web se prepara el ambiente entregado para su verificación.

La instalación de los servicio y software necesario para el funcionamiento de la aplicación se realiza cuidadosamente para que no se tenga contratiempos posteriores.

7.1 Verificación del ambiente de ejecución para la aplicación web

El ambiente para la instalación y ejecución de la aplicación web es entregado por URBS-PEED, este equipo debe tener unos requerimientos mínimos para el correcto funcionamiento.

A continuación, el detalle del ambiente donde el aplicativo y su ecosistema de ejecución va a ser instalado:

- CPU: AMD dual core de 2.80Ghz
- RAM: 4Gb
- Sistema operativo: Windows 10 Pro Licenciado.
- Red: CISEAL ubicada en la Pontificia Universidad Católica del Ecuador sede Nayón.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)



Ilustración 39 Características del equipo servidor

7.2 Instalación de Java

La aplicación web está desarrollada en lenguaje de programación Java, para que esta funcione correctamente es necesario que se tenga instalada la versión oficial con la que se programó el aplicativo, en este caso la versión a utilizar es Java 7 precisamente `jdk1.7.0_79`.

Esta versión se puede descargar desde la página oficial de Oracle.

<http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html>

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

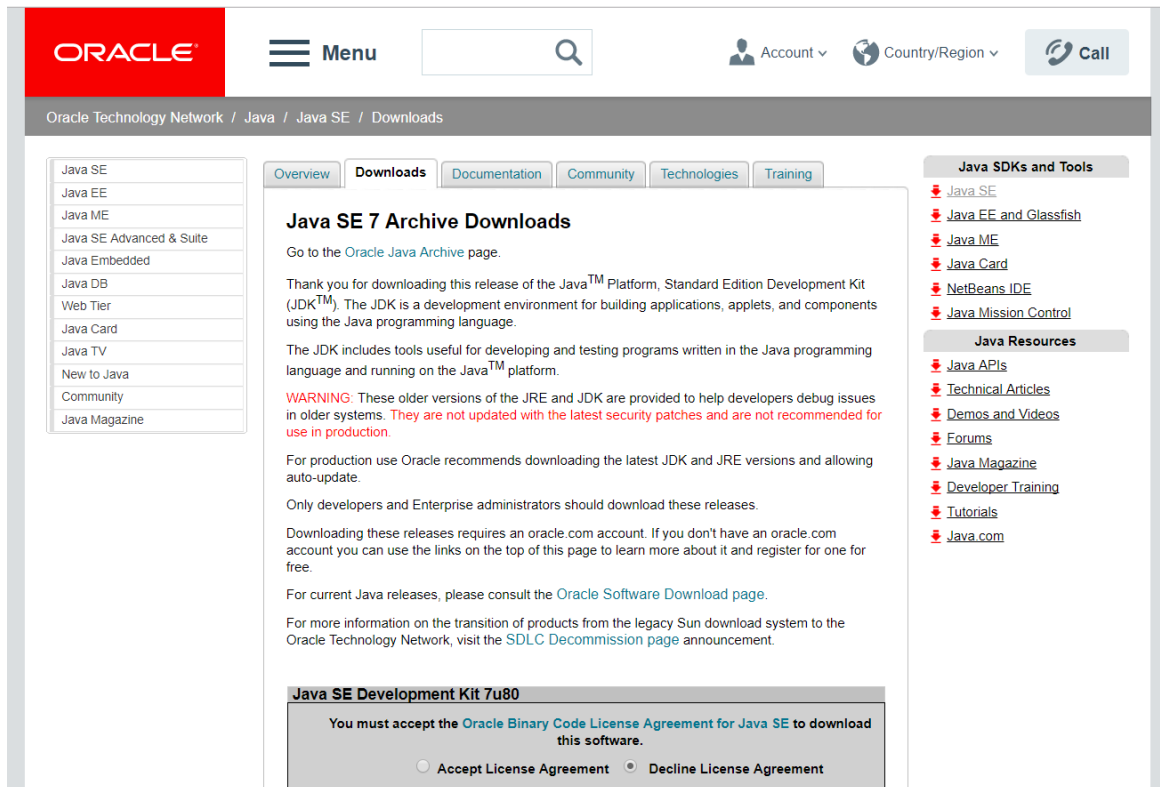


Ilustración 40 Descarga de Java Fuente: <http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html>

Para descargar Java de la página oficial de Oracle es necesario tener una cuenta que se puede crear de manera gratuita, una vez descargado se procede a la instalación de Java Jdk (Java Development kit).

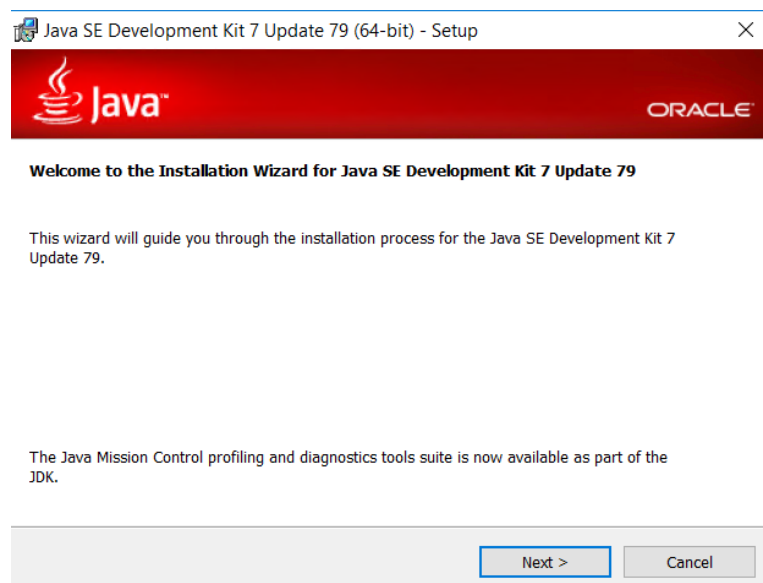


Ilustración 41 Instalación de Java

Se selecciona la ruta en donde se desea instalar Java, en este caso se deja el directorio por defecto de instalación, en CISEAL de la Pontificia

Universidad Católica del Ecuador sede de Nayón, se solicitó permiso para todas las instalaciones sobre el sistema operativo.

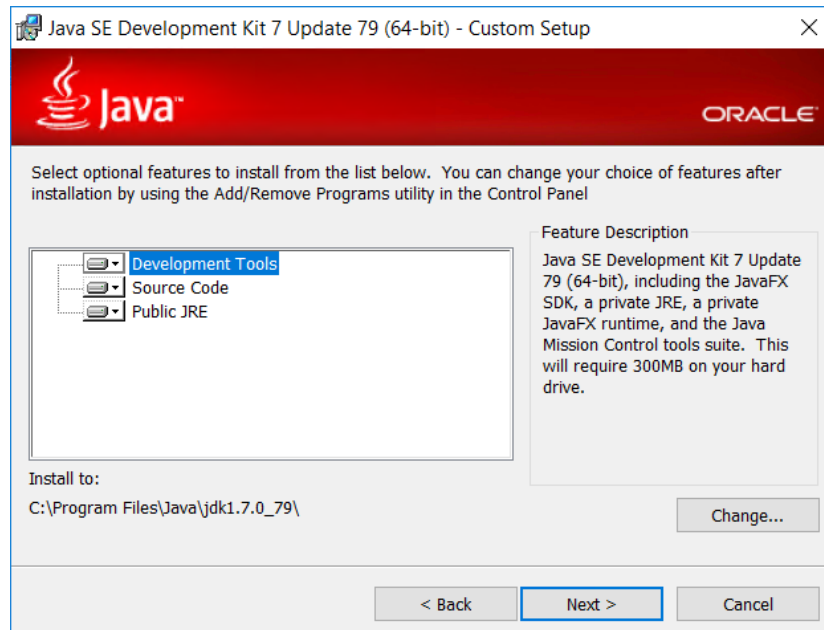


Ilustración 42 Instalación de Java - Selección de Directorio

Finalizada la instalación, se procede a la configuración de la variable de entorno, para que la Java Virtual Machine JVM pueda ser ejecutada desde cualquier parte del sistema operativo.

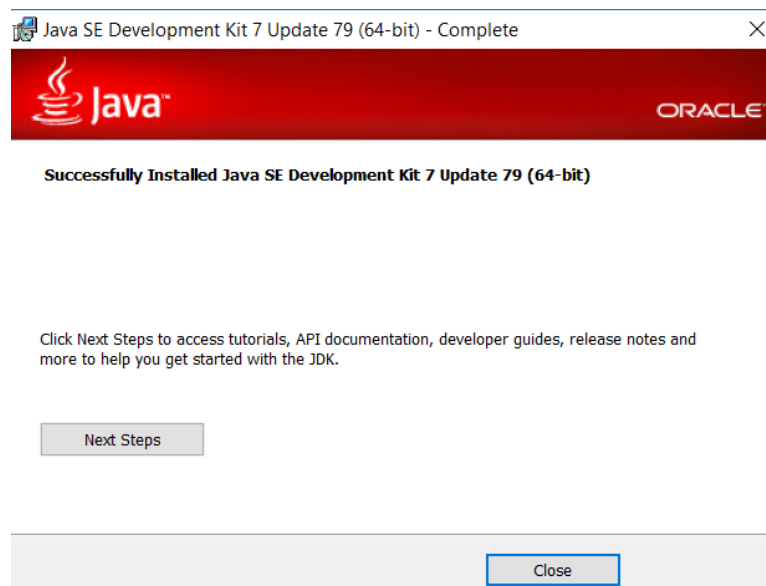


Ilustración 43 Instalación de Java - Instalación Exitosa

Configuración de la variable de ambiente Java para el sistema operativo, para esto se ingresa a las opciones avanzadas del equipo, en la pestaña de avanzado se sitúa el botón de variables de ambiente.

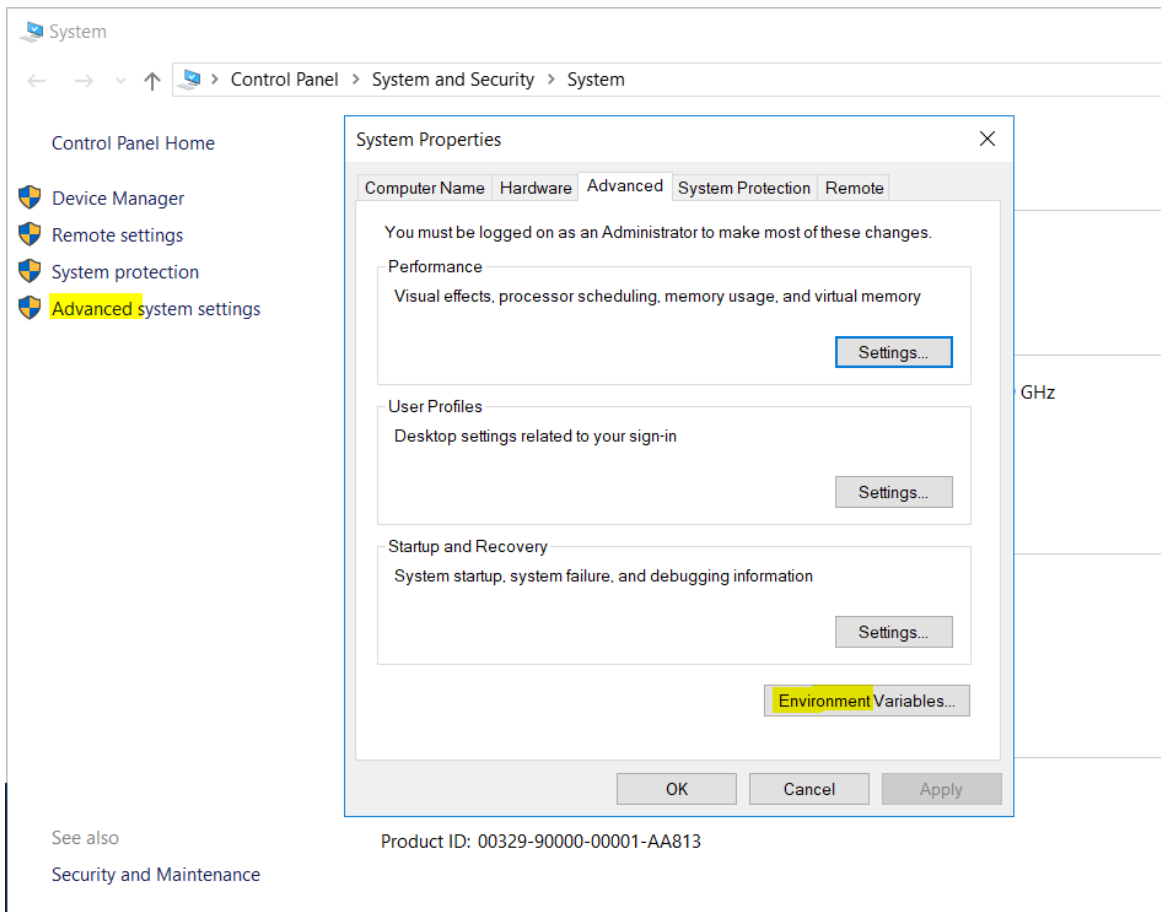


Ilustración 44 Configuración de Variables de ambiente de Java

Una vez en la configuración, se ingresa los siguientes registros:

- Variable: JAVA_HOME
- Valor: Directorio donde está instalado Java, en este caso: C:\Program Files\Java\jdk1.7.0_79

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

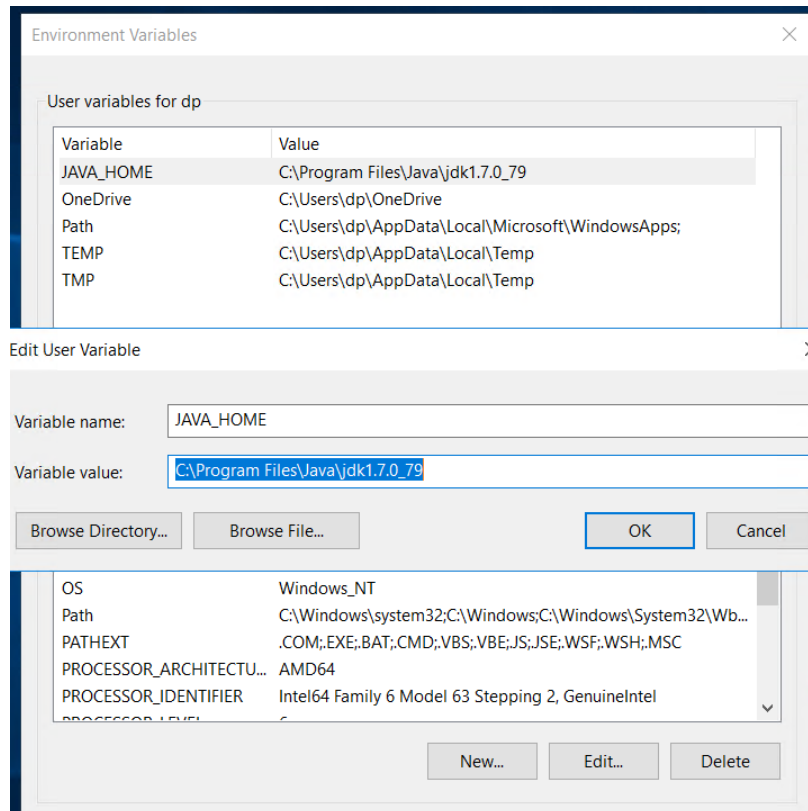


Ilustración 45 Creación de variable de ambiente JAVA_HOME

Adicional se configura un registro más en el Path de variables de ambiente con el valor de: %JAVA_HOME%\bin

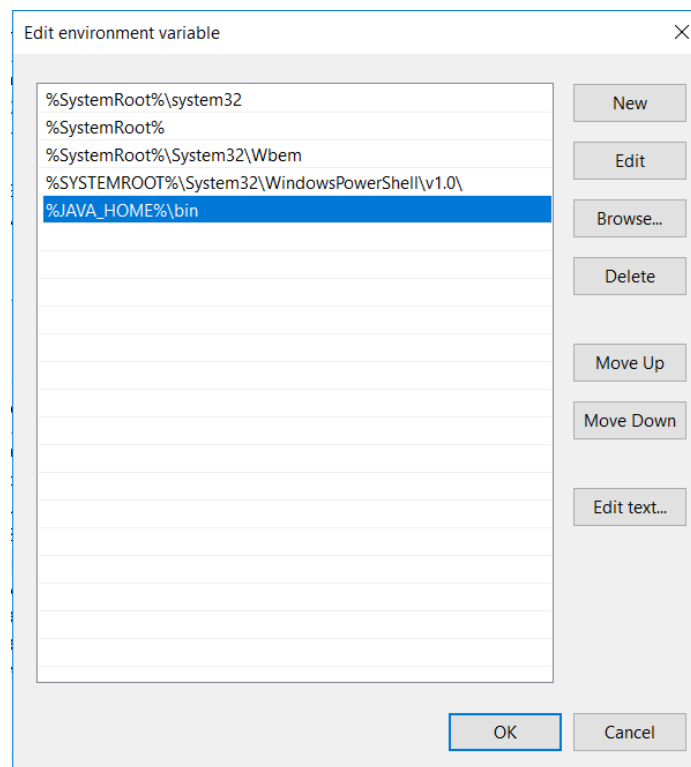
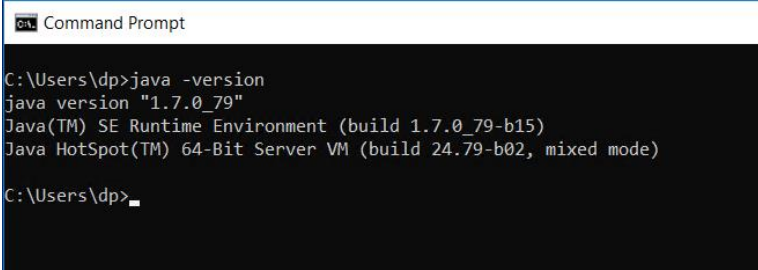


Ilustración 46 Creación de variable JAVA_HOME en el Path del sistema operativo

Una vez creadas las variables de ambiente se puede reconocer a Java desde cualquier programa del sistema operativo.



```
Command Prompt
C:\Users\dp>java -version
java version "1.7.0_79"
Java(TM) SE Runtime Environment (build 1.7.0_79-b15)
Java HotSpot(TM) 64-Bit Server VM (build 24.79-b02, mixed mode)
C:\Users\dp>
```

Ilustración 47 Verificación de Variables de ambiente Java

7.3 Instalación de la Base de datos Postgresql

Para la instalación y administración de la base de datos es necesario dos aspectos fundamentales, el sistema de base de datos y el cliente para la administración de Postgres llamada PGAdmin.

7.3.1 Instalación de la Base de datos Postgresql

Construyendo la aplicación web se define que versión de base de datos se va a trabajar, para el aplicativo la versión de Postgres a instalar es: PostgreSQL 9.4

Se descarga PostgreSQL de la página oficial de la base de datos: <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads#windows>

Y se selecciona la versión que se necesita.

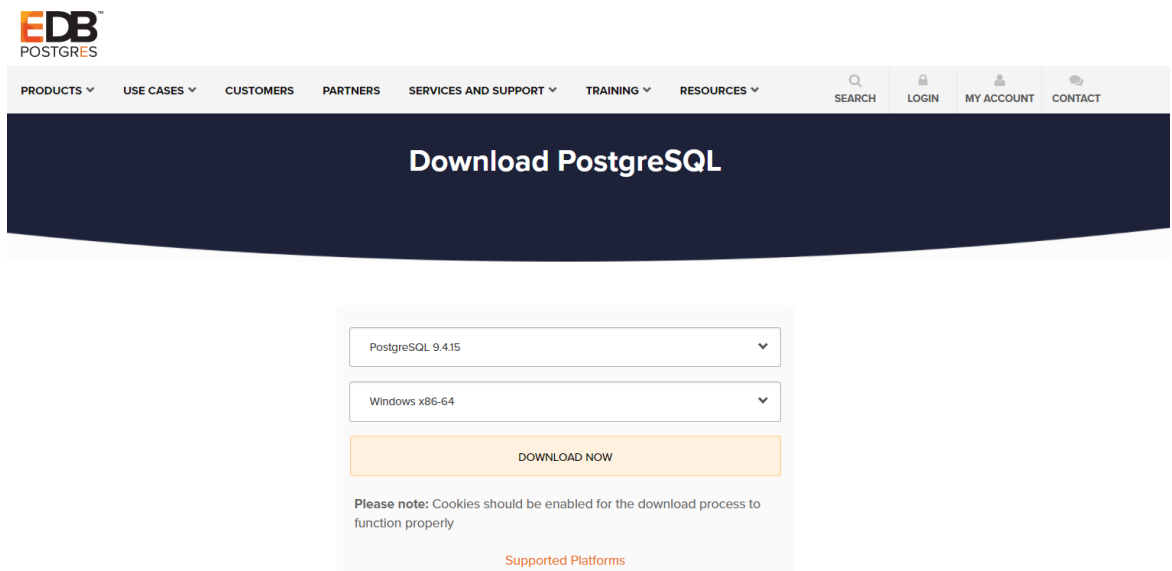


Ilustración 48 Descarga de PostgreSQL Fuente: <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads#windows>

Una vez descargado el instalador se procede a la instalación de la base de datos PostgreSQL, la configuración que se hace durante la instalación debe ser ejecutada correctamente para que no haya inconvenientes con la integración del servidor de aplicaciones, PostgreSQL exige una versión de .NET específica, en caso de no tenerla, se debe instalar.



Ilustración 49 Instalación de PostgreSQL

Se selecciona el directorio donde se va a instalar PostgreSQL, este directorio es preferible que se encuentre en la carpeta Archivos de Programa por temas de código .NET necesario para su ejecución.

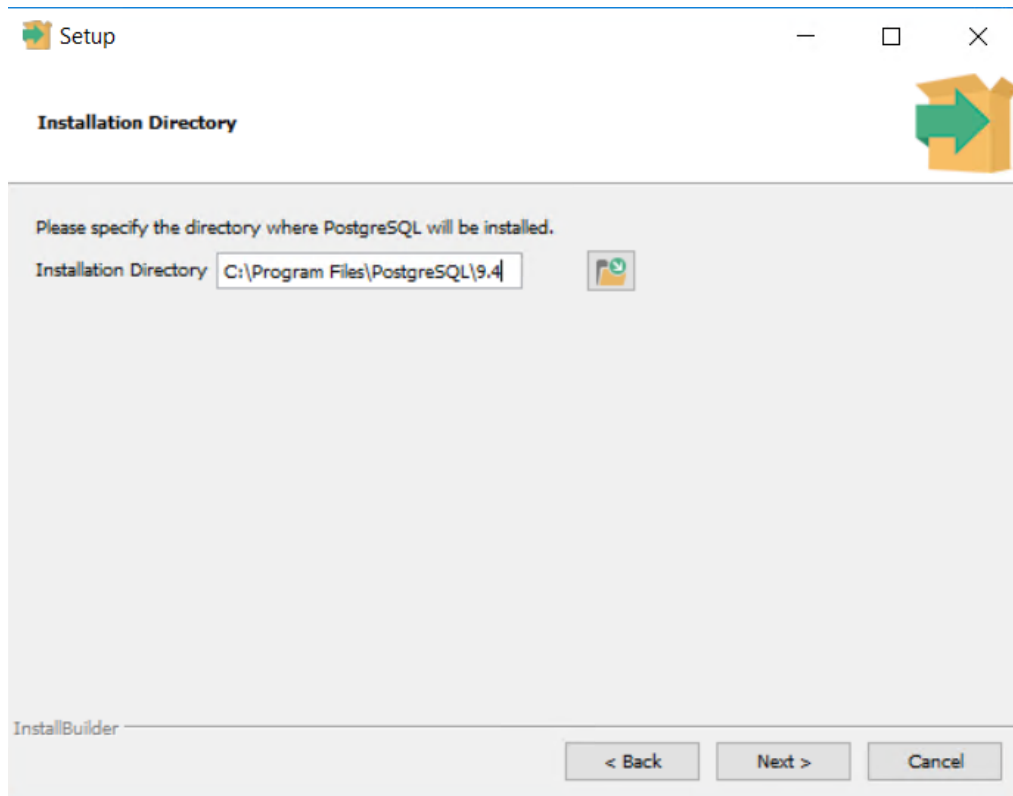


Ilustración 50 Instalación de PostgreSQL - Selección de directorio de instalación

Se selecciona el directorio donde se va a guardar la información de la base de datos PostgreSQL, este directorio guarda la información que se ingrese y en muchos de los casos es preferible que este en un disco diferente a donde está instalado el aplicativo web.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

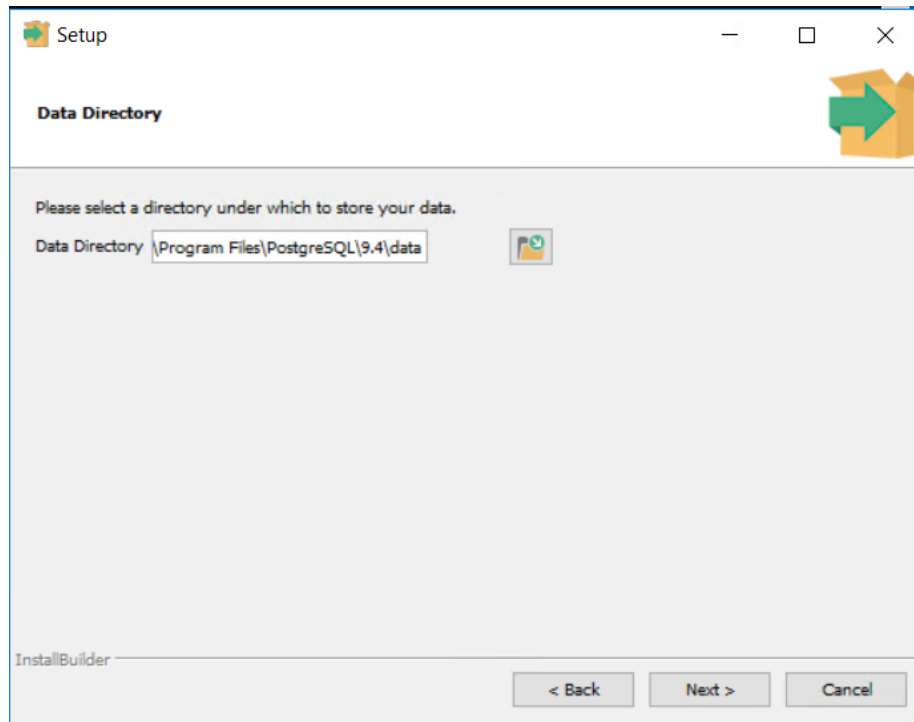


Ilustración 51 Instalación de PostgreSQL - Selección de directorio de datos

Se ingresa la clave que va a servir para el super usuario postgres, este super usuario tiene acceso a todas las funcionalidades de la base de datos, se recomienda no perder esta clave porque el proceso de reiniciarla es complicado.

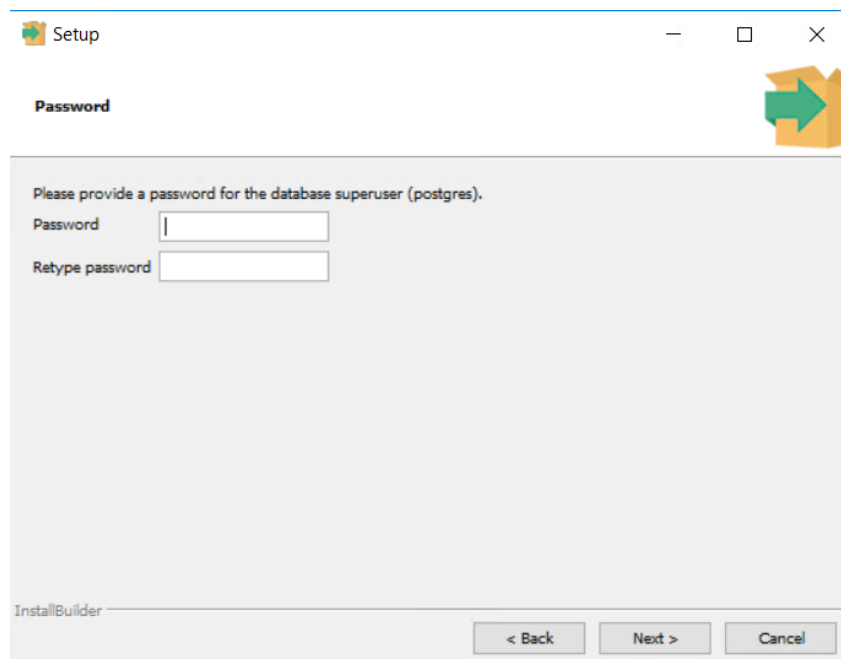


Ilustración 52 Instalación de PostgreSQL - clave del usuario postgres

Se registra el puerto por el que se va acceder a la base de datos en el sistema operativo, este por defecto es 5432 pero puede ser modificado a cualquiera

mientras se tenga disponibilidad, ósea que no esté siendo usado por otro servicio.

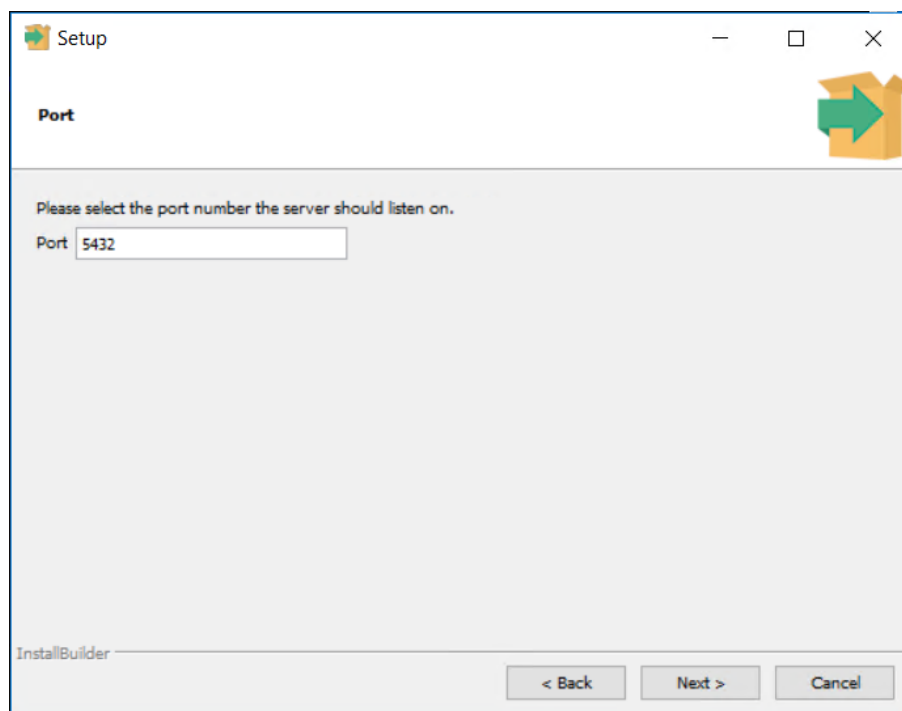


Ilustración 53 Instalación de PostgreSQL - Selección de puerto

La instalación de la base de datos PostgreSQL va a trabajar solo en ambiente local, lo cual no requiere configuración de clusters, se deja por defecto esta opción.

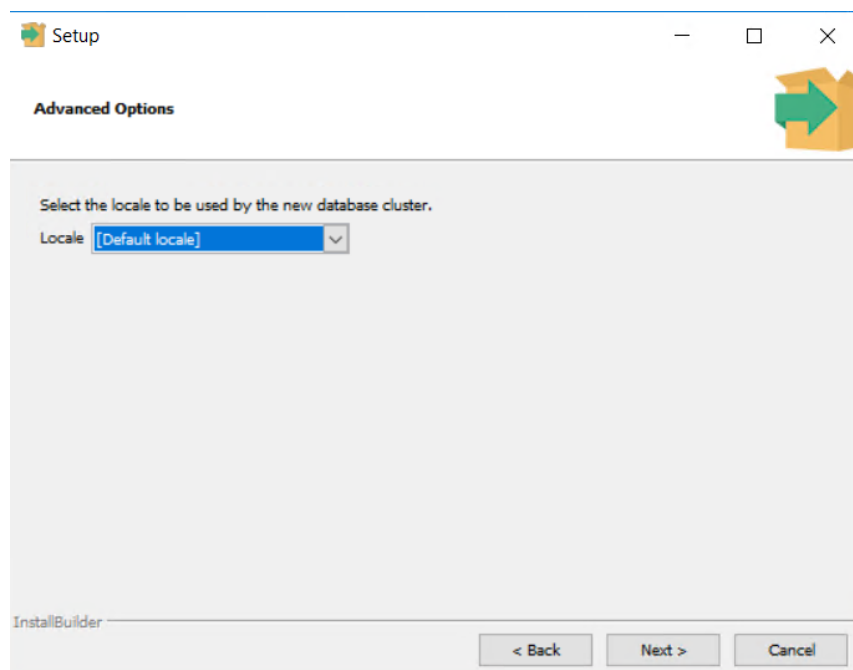


Ilustración 54 Instalación de PostgreSQL - Opciones Avanzadas para Cluster

Se confirma la instalación del motor de datos PostgreSQL en el servidor local formando parte de la capa de acceso a datos del aplicativo web.

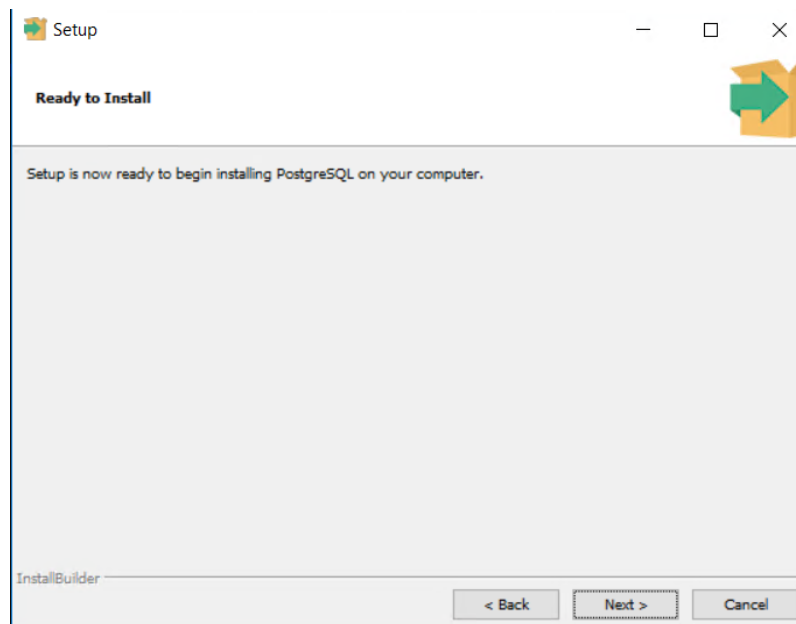


Ilustración 55 Instalación de PostgreSQL - Confirmación de Instalación

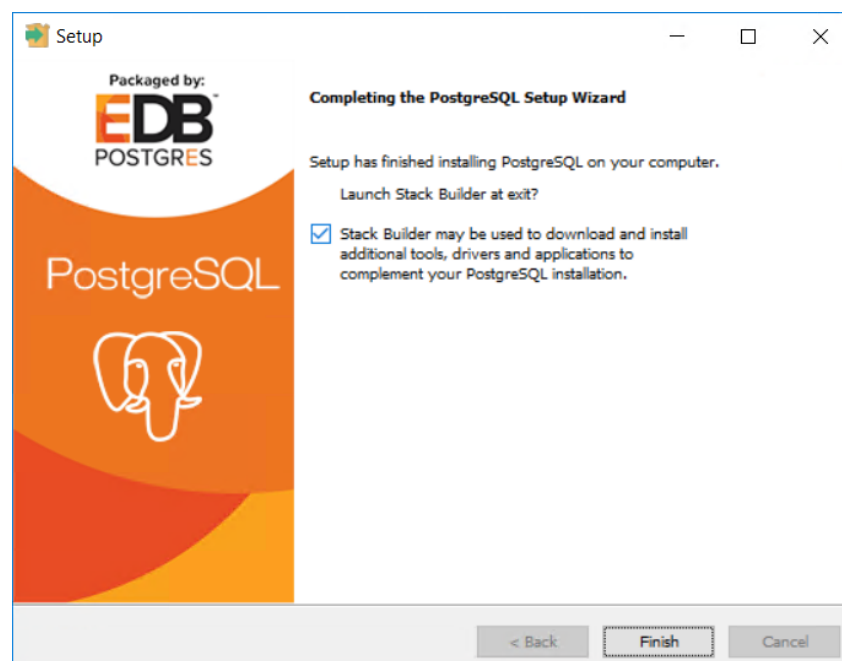


Ilustración 56 Instalación de PostgreSQL - Instalación Exitosa

7.3.2 Instalación de cliente PgAdmin

El motor de datos PostgreSQL por sí solo no sirve para administrar las bases ni los esquemas que en ella maneja, para esto se necesita instalar un cliente de base de datos, este cliente permite el desarrollo y administración de la información, pgAdmin es la plataforma open source oficial y puede ser

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

utilizada en cualquier sistema operativo como Windows, Mac o Linux.

Se descarga pgAdmin de la página repositorio del software:

<https://www.pgadmin.org/download/pgadmin-4-windows/>

pgAdmin 4 (Windows)

Download

Maintainer: Dave Page

pgAdmin is available for Windows™ 7 (desktop) or 2008R2 (server) and above. The packages below include both the Desktop Runtime and Web Application:

- pgAdmin 4 v2.1 (released Jan. 11, 2018)
- pgAdmin 4 v2.0 (released Oct. 5, 2017)
- pgAdmin 4 v1.6 (released July 13, 2017)
- pgAdmin 4 v1.5 (released May 19, 2017)
- pgAdmin 4 v1.4 (released April 13, 2017)
- pgAdmin 4 v1.3 (released March 10, 2017)
- pgAdmin 4 v1.2 (released Feb. 9, 2017)
- pgAdmin 4 v1.1 (released Oct. 27, 2016)
- pgAdmin 4 v1.0 (released Sept. 29, 2016)

Ilustración 57 - Instalación de pgAdmin – Descarga Fuente: <https://www.pgadmin.org/download/pgadmin-4-windows/>

Al ser una plataforma para administrar la base de datos, no se necesita una versión específica del programa.

Donate | Contact | Search

PostgreSQL The world's most advanced open source database.

Home | About | Download | Documentation | Community | Developers | Support | Your account

» Downloads
Binary
Source

» Software Catalogue

» File Browser

File Browser

Top → pgadmin → pgadmin4 → v2.1 → windows

Directories

- [Parent Directory]

Files

	CURRENT_MAINTAINER	2018-01-11 13:33:57	31 bytes
	pgadmin4-2.1-x86.exe	2018-01-11 13:37:25	154.5 MB
	pgadmin4-2.1-x86.exe.sig	2018-01-11 13:37:25	543 bytes

Current Maintainer

Dave Page
dpage@postgresql.org

Ilustración 58 Instalación de pgAdmin - Descarga de repositorio Fuente: <https://www.postgresql.org/ftp/pgadmin/pgadmin4/v2.1/windows/>

para la administración de base de datos pgAdmin.

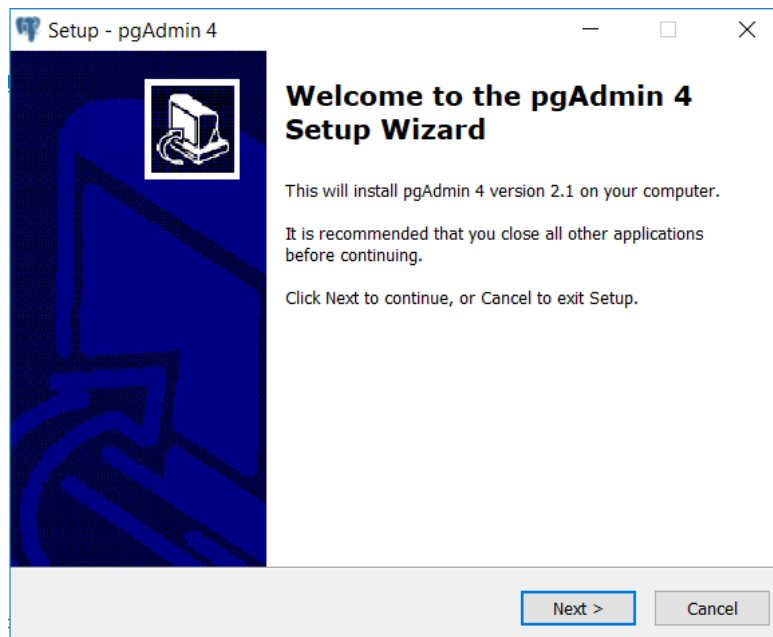


Ilustración 59 Instalación de pgAdmin

Es necesario aceptar el licenciamiento de tipo open source de pgAdmin.

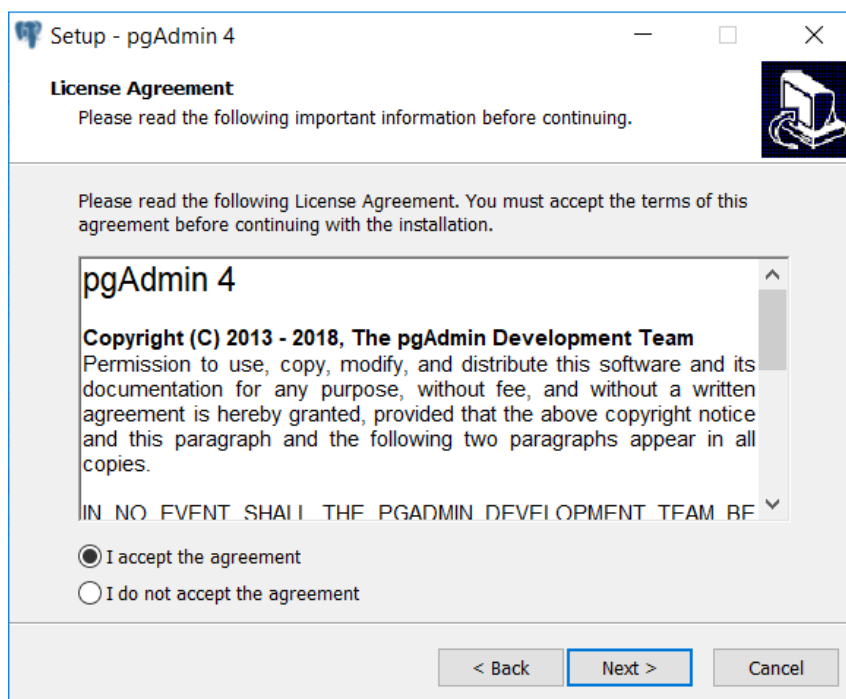


Ilustración 60 Instalación de pgAdmin - Licenciamiento y condiciones

Se selecciona el directorio raíz donde pgAdmin va a ser instalado.

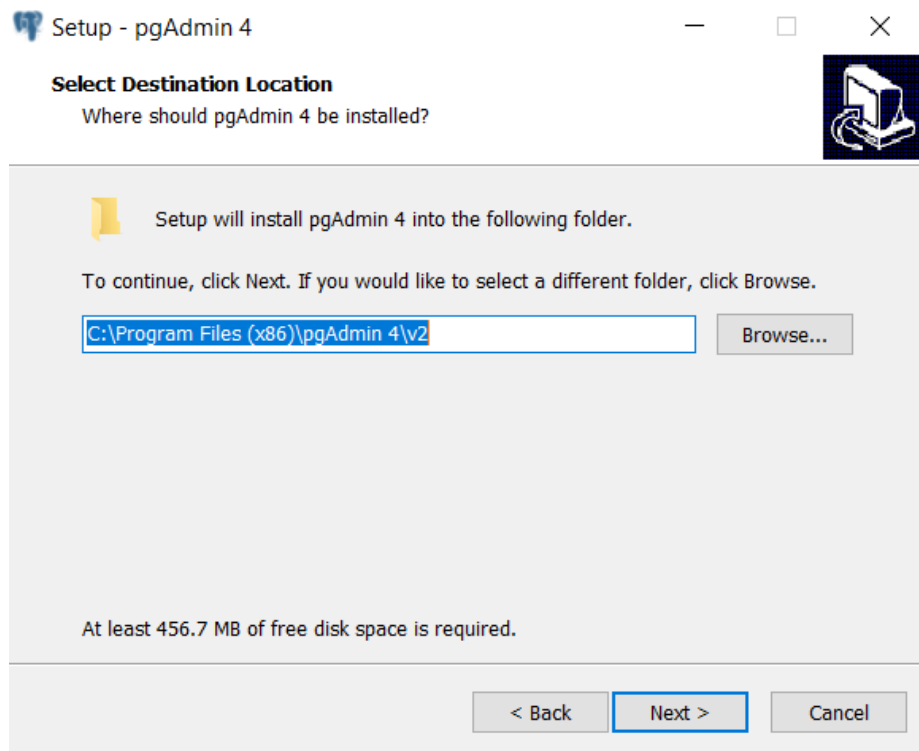


Ilustración 61 Instalación de pgAdmin - Directorio de Instalación

Se confirma y se acepta la instalación.

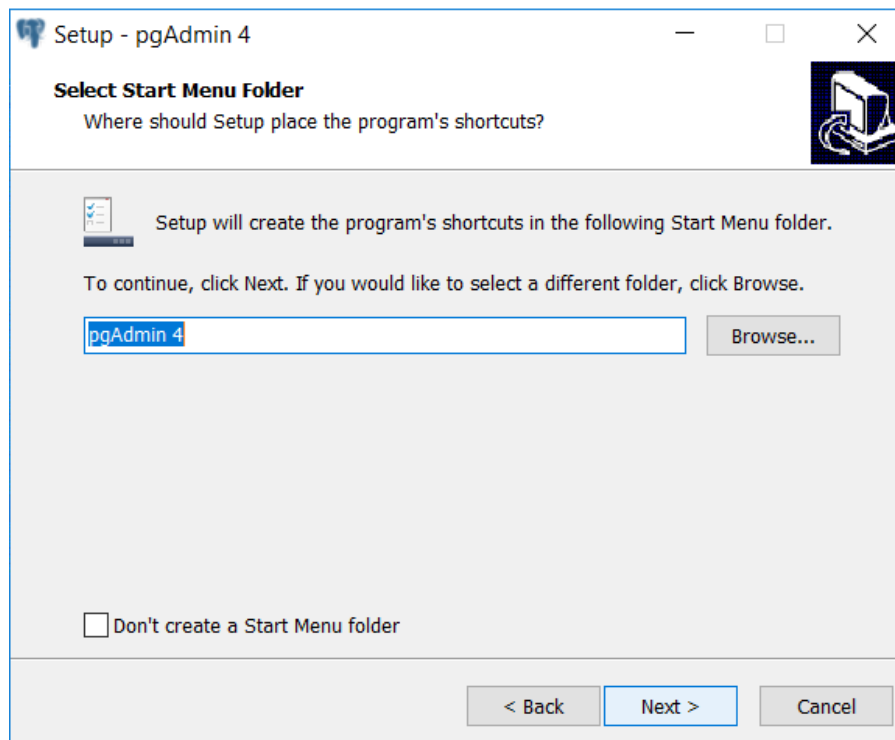


Ilustración 62 Instalación de pgAdmin - Configuración Adicional

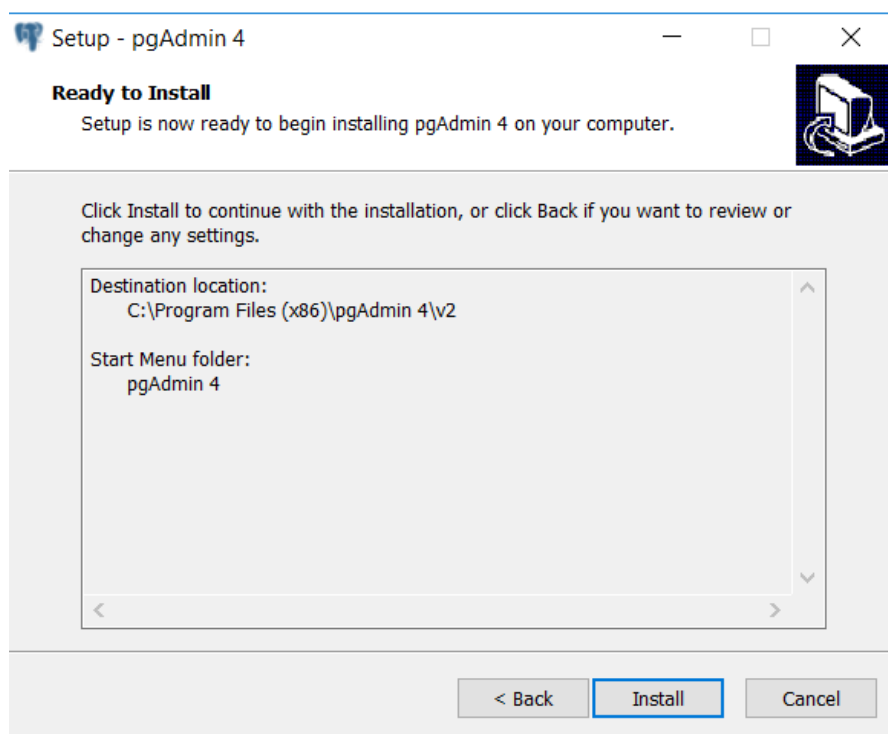


Ilustración 63 Instalación de pgAdmin - Confirmación de Instalación

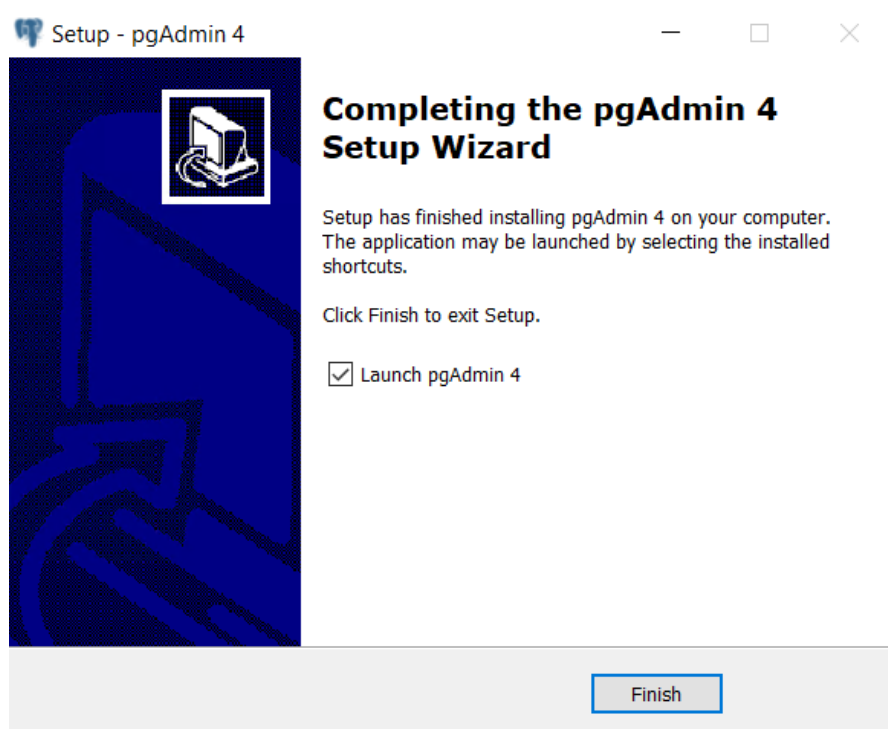


Ilustración 64 Instalación de pgAdmin - Finalización exitosa de Instalación

7.3.3 Creación de la base de datos

Después de la instalación de la base de datos PostgreSQL y de la plataforma para administrarla pgAdmin, se procede a utilizar este programa para crear la base de datos en la donde se va a alojar la información que URBS-PEED

Diego Paúl Montalvo Páez

ingrese usando el aplicativo web desarrollado.

Se crea una nueva conexión al servidor o equipo en la para trabajar con la base de datos PostgreSQL.

En la pestaña de General, se coloca el nombre que se le quiere dar a la conexión, el resto de parámetros de los deja por defecto.

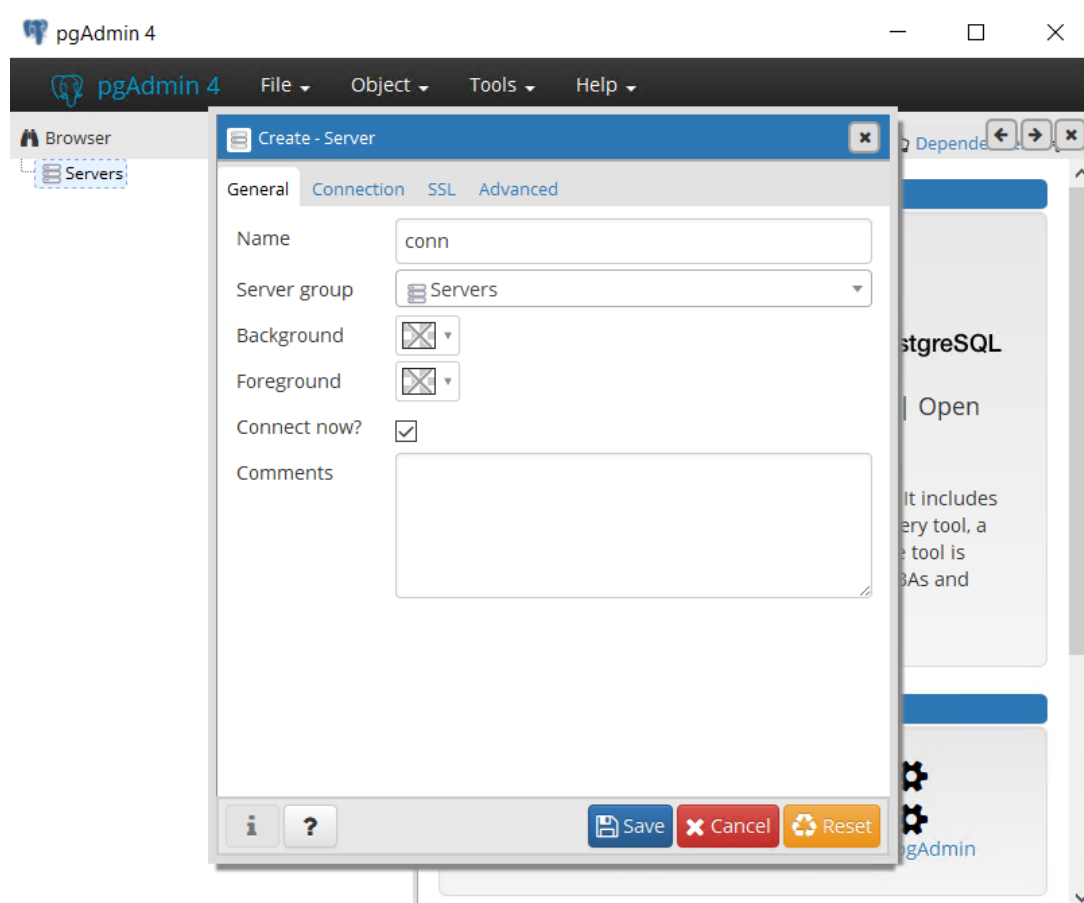


Ilustración 65 Creación de base de datos - Nueva conexión a servidor

En la pestaña de Conexión, se coloca el nombre del servidor en el cual se encuentra el servicio de base de datos, en este caso el servidor es el local, además se coloca el puerto por el cual esta publica el servicio y la contraseña del super usuario postgres que se colocó en el instante de la estación.

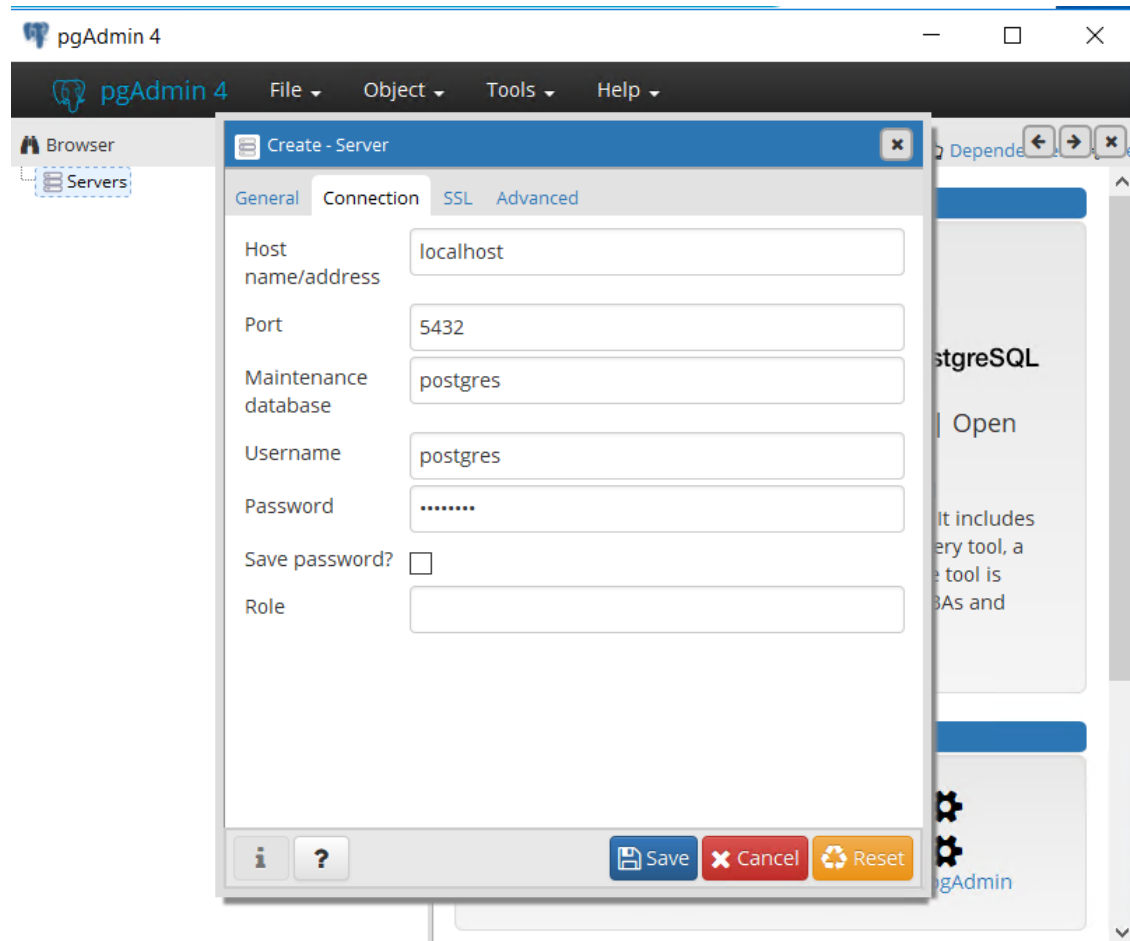


Ilustración 66 Creación de base de datos - Pestaña Conexión

Se crea la conexión al motor de base de datos PostgreSQL, en el cual se puede verificar las bases creadas.

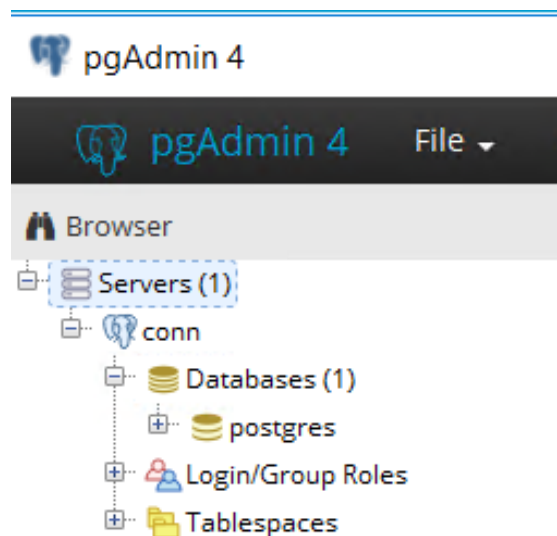


Ilustración 67 Creación de base de datos - Conexión a servidor exitosa

Se procede a crear una nueva base de datos la cual va a almacenar la información del aplicativo web.

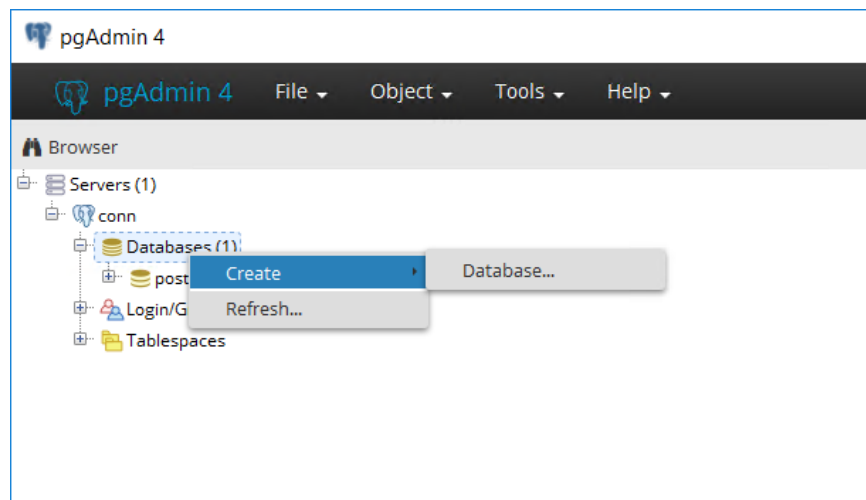


Ilustración 68 Creación de base de datos - nueva base de datos

Se procede a crear una nueva base de datos la cual va a almacenar la información del aplicativo web, se coloca el nombre y el super usuario que va a ser dueño de la base.

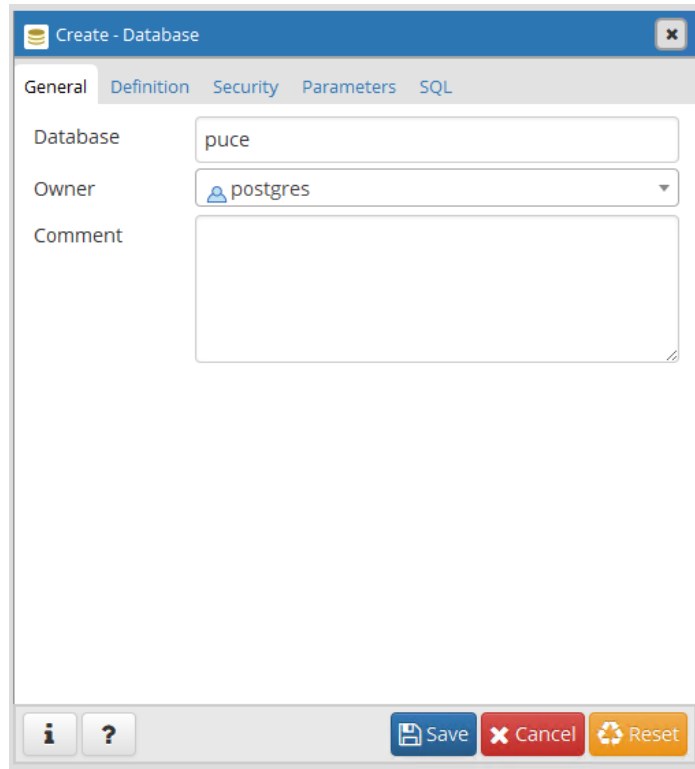


Ilustración 69 Creación de base de datos

Una vez creada la base de datos, se crea la estructura esquema correspondiente con la sección tablas que es donde se van a ver reflejadas

las entidades que se creen detalladas en el modelo de entidad-relación.

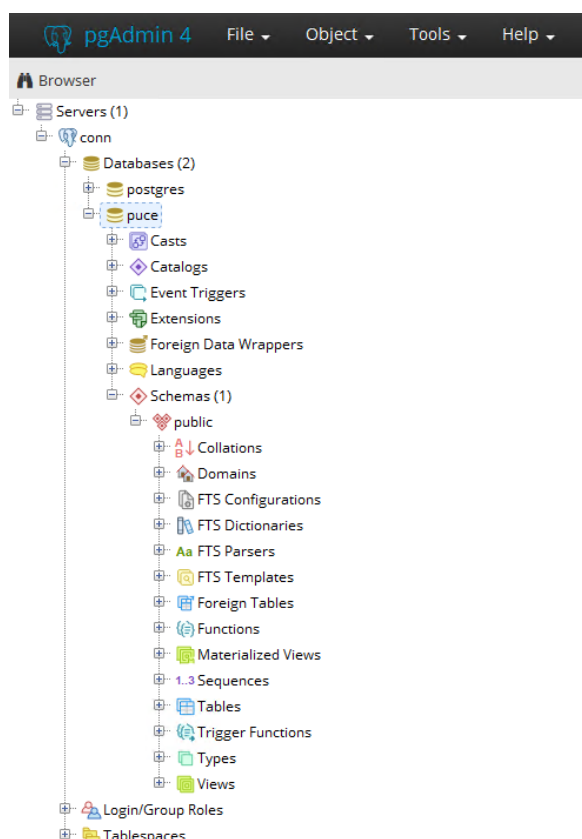


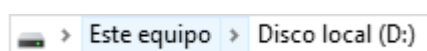
Ilustración 70 Creación de base de datos - Estructura de la Base de datos

En caso de que ya se tenga información ingresada en la base de datos, se debe cargar el respaldo de la información en la sección de restaurar base, así el sistema puede levantarse con la opción de solo lectura al esquema.

7.4 Instalación del Servidor de Aplicaciones WildFly

Con base a lo entregado como fuentes del código del desarrollo y ambientes de instalación, se encuentra el servidor de aplicaciones WildFly con las configuraciones necesarias para el correcto despliegue y funcionamiento del aplicativo.

El directorio que se debe copiar es “wildfly-8.2.1.Final”, este se ubica directamente en el disco D:



Como las variables de ambiente de Java han sido creadas exitosamente, el servidor de aplicaciones va a reconocer sobre que Java Virtual Machine debe ejecutarse y sobre el mismo correrá la aplicación.

Para que el servidor de aplicaciones este integrado con el aplicativo móvil se debe realizar una serie de configuraciones.

7.4.1 Configuración del Pool Base de datos

Se debe configurar el pool de conexiones que va a tener el servidor de aplicación Wildfly con la base de datos PostgreSQL que ya se instaló previamente, estas modificaciones deben ser realizadas en el archivo “standalone” que se encuentra en el directorio:

\wildfly-8.2.1.Final\standalone\configuration

Standalone es un archivo de configuración escrito en XML que permite manejar la modularidad del servidor de aplicaciones agregando o modificando subsystems de tal manera que haya una integración entre el servidor y los aplicativos que se manejen.

Para modificar la configuración de la base de datos se debe cambiar la etiqueta de tipo Datasource de la siguiente manera:

```
<subsystem xmlns="urn:jboss:domain:datasources:2.0">
  <datasources>
    <datasource jta="true" jndi-name="java:jboss/datasources/modulosPlusDS" pool-name="modulosPlusDS" enabled="true" use-ccm="true">
      <connection-url>jdbc:postgresql://localhost/puce</connection-url>
      <driver-class>org.postgresql.Driver</driver-class>
      <driver>postgresql-9.4.1212.jre6.jar</driver>
      <security>
        <user-name>postgres</user-name>
        <password>postgres</password>
      </security>
      <validation>
        <validate-on-match>>false</validate-on-match>
        <background-validation>>false</background-validation>
      </validation>
      <timeout>
        <set-tx-query-timeout>>false</set-tx-query-timeout>
        <blocking-timeout-millis>0</blocking-timeout-millis>
        <idle-timeout-minutes>0</idle-timeout-minutes>
        <query-timeout>0</query-timeout>
        <use-try-lock>0</use-try-lock>
        <allocation-retry>0</allocation-retry>
        <allocation-retry-wait-millis>0</allocation-retry-wait-millis>
      </timeout>
      <statement>
        <share-prepared-statements>>false</share-prepared-statements>
      </statement>
    </datasource>
  </datasources>
</subsystem>
```

Ilustración 71 Standalone - subsystem Datasource

En el datasource creado del subsystem se manejan las siguientes etiquetas:

- Datasource: Hace relación al datasource creado en la aplicación web a continuación.

```
<?xml version="1.0" encoding="UTF-8"?>
<persistence version="2.1"
  xmlns="http://xmlns.jcp.org/xml/ns/persistence" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence_
  <persistence-unit name="ciseal" transaction-type="JTA">
  <provider>org.hibernate.ejb.HibernatePersistence</provider>
  <jta-data-source>java:jboss/datasources/modulosPlusDS</jta-data-source>
  <properties>
  <property name="hibernate.archive.autodetection" value="class" />
  <property name="hibernate.dialect" value="org.hibernate.dialect.PostgreSQL9Dialect" />
  <property name="hibernate.connection.driver_class" value="org.postgresql.Driver" />
  <property name="hibernate.connection.url" value="jdbc:postgresql://localhost:5432/puce" />
  <property name="hibernate.connection.username" value="postgres" />
  <property name="hibernate.connection.password" value="1234" />
  <property name="hibernate.show_sql" value="false" />
  <property name="hibernate.flushMode" value="FLUSH_AUTO" />
  <property name="hibernate.hbm2ddl.auto" value="update" />
  </properties>
  </persistence-unit>
</persistence>
```

Ilustración 72 aplicativo web - persistence

Este es un tipo de archivo de tipo XML en el desarrollo que hace referencia a la conexión que va a existir entre la base de datos PostgreSQL y la capa de negocios de la aplicación web, el nombre de la etiqueta:

```
<jta-data-source>java:jboss/datasources/modulosPlusDS</jta-data-source>
```

Hace referencia al subsystem en el servidor de aplicaciones,

- Conexión URL: Hace referencia a la cadena de conexión a la base de datos.

jdbc:postgresql://localhost/puce

- Driver: Hace referencia al Api JDBC o Java Database Connectivity que permite hacer la conexión a la base de datos, este se lo puede encontrar en la localización.

\\wildfly-8.2.1.Final\\standalone\\deployments\\postgresql-9.4.1212.jre6

- Adicional se coloca el usuario y contraseña de la base de datos donde están las tablas a consumir por el aplicativo web

7.4.2 Configuración de validación de credenciales (Login)

El servidor de aplicaciones Wildfly también permite crear un subsystem para realizar la validación de credenciales de usuario y contraseña en la pantalla de Login de la aplicación web, esto se realiza modificando:

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACION EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

```
<subsystem xmlns="urn:jboss:domain:security:1.2">
  <security-domains>
    <security-domain name="modulosPlus-realm">
      <authentication>
        <login-module code="Database" flag="required">
          <module-option name="dsJndiName" value="java:jboss/datasources/modulosPlusDS"/>
          <module-option name="principalsQuery" value="select password from usuario where identificacion=?"/>
          <module-option name="rolesQuery" value="select u.rol_Id,'Roles' from usuario u where u.identificacion = ?"/>
        </login-module>
      </authentication>
    </security-domain>
    <security-domain name="jboss-web-policy" cache-type="default">
      <authorization>
        <policy-module code="Delegating" flag="required"/>
      </authorization>
    </security-domain>
    <security-domain name="jboss-ejb-policy" cache-type="default">
      <authorization>
        <policy-module code="Delegating" flag="required"/>
      </authorization>
    </security-domain>
  </security-domains>
</subsystem>
```

Ilustración 73 Standalone - subsystem security domain

En el módulo de login `<login-module code="Database" flag="required">` se configura:

- Datasource: Hace relación al datasource creado en la capa de persistencia del aplicativo web.
- PrincipalQuery: Aquí se coloca el query de base de datos que obtiene la contraseña, en este case esta consulta es:

```
select password from usuario where identificacion=?
```

En donde el “?” corresponde al usuario que se ingresa en la pantalla de Login.

- RolesQuery: Aquí se coloca el query de base de datos que obtiene los roles del usuario, en este case esta consulta es:

```
select u.rol_Id,'Roles' from usuario u where u.identificacion = ?
```

En donde el “?” corresponde al usuario que se ingresa en la pantalla de Login.

De esta manera la configuración se integra con el login del aplicativo web y permite utilizarlo como medio de validación de credenciales.

7.4.3 Configuración de puertos

Para la configuración de puertos en los que el servidor de aplicaciones va a ser publicado para el consumo de sus servicios mediante la red, se cambia la configuración en la etiqueta `socket-binding-group` del archivo `standalone`.

```
<socket-binding-group name="standard-sockets" default-interface="public" port-offset="${jboss.socket.binding.port-offset:0}">
  <socket-binding name="management-http" interface="management" port="${jboss.management.http.port:9990}"/>
  <socket-binding name="management-https" interface="management" port="${jboss.management.https.port:9993}"/>
  <socket-binding name="ajp" port="${jboss.ajp.port:8009}"/>
  <socket-binding name="http" port="${jboss.http.port:8080}"/>
  <socket-binding name="https" port="${jboss.https.port:8443}"/>
  <socket-binding name="txn-recovery-environment" port="4712"/>
  <socket-binding name="txn-status-manager" port="4713"/>
  <outbound-socket-binding name="mail-smtp">
    <remote-destination host="localhost" port="25"/>
  </outbound-socket-binding>
</socket-binding-group>
```

Ilustración 74 Standalone - Configuración de puertos

Aquí se modifica el puerto HTTP o HTTPS para los valores requeridos, el puerto por defecto es el 8080 y ese es con el que se va a trabajar.

7.5 Ejecución del Servidor de Aplicaciones WildFly

Previo a la ejecución del servidor de aplicaciones es necesario colocar el aplicativo web en un directorio perteneciente al servidor, la aplicación web se obtiene después de terminado el desarrollo, compilado y empaquetado, este se coloca en la localización:

`\wildfly-8.2.1.Final\standalone\deployments`

El empaquetado del aplicativo web se encuentra en las fuentes entregadas, a este empaquetado se le conoce como el archivo WAR.

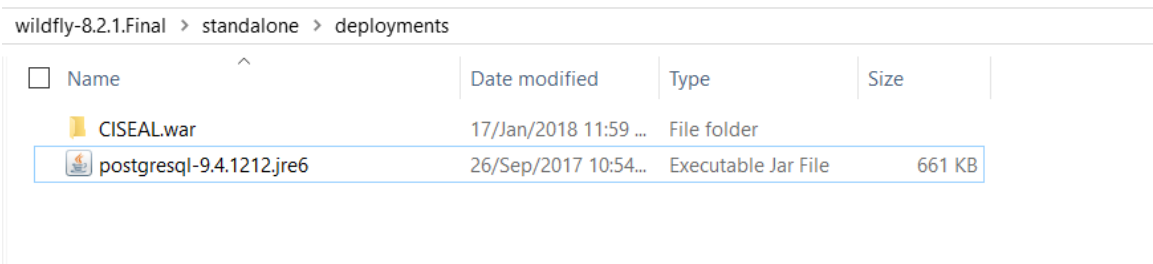


Ilustración 75 pre despliegue de la aplicación

Para iniciar los servicios del servidor de aplicaciones es necesario ejecutar el archivo de inicio:

```
\wildfly-8.2.1.Final\bin\standalone.bat
```

Este archivo inicia una Java Virtual Machine con las configuraciones de memoria por defecto en donde se desplegarán las aplicaciones web que se tenga instaladas.

Este archivo inicia el servicio de la aplicación solo para trabajo local o solo para uso dentro del ambiente del servidor, para que el aplicativo pueda ser utilizada por todo equipo que este dentro de la misma red se tiene que colocar los siguientes parámetros:

```
D:  
cd wildfly-8.2.1.Final\bin  
standalone.bat -b=0.0.0.0
```

Con los parámetros de ejecución del archivo standalone.bat -b=0.0.0.0 se publica y despliega la aplicación para todos los usuarios de la red.

El inicio del servidor de aplicación tiene varias etapas, las cuales se detallan:

1. Inicio: Arranca el servidor de aplicaciones y encuentra los WAR a desplegar en su directorio predeterminado.

```
12:08:30,407 INFO [org.jboss.modules] (main) JBoss Modules version 1.3.3.Final  
12:08:30,757 INFO [org.jboss.msc] (main) JBoss MSC version 1.2.2.Final  
12:08:30,831 INFO [org.jboss.as] (MSC service thread 1-6) JBAS015899: WildFly 8.2.1.Final "Tweek" starting  
12:08:32,305 INFO [org.jboss.as.server.deployment.scanner] (DeploymentScanner-threads - 1) JBAS015003: Found CISEAL.war in deployment directory.
```

Ilustración 76 Log WildFly - Inicio de servidor de aplicaciones

2. Base de datos: Despliega el driver de base de datos y se realiza la conexión a esta.

```
INFO [org.jboss.ws.common.management] (MSC service thread 1-4) JBWS022052: Starting JBoss Web Services - Stack CXF Server 4.3.2.Final  
INFO [org.jboss.as.connector.deployers.jdbc] (MSC service thread 1-4) JBAS010404: Deploying Non-JDBC-compliant driver class org.postgresql.Driver (version 9.4)  
INFO [org.jboss.as.connector.deployers.jdbc] (MSC service thread 1-4) JBAS010417: Started Driver service with driver-name = postgresql-9.4.1212.jre6.jar  
INFO [org.jboss.as.connector.subsystems.datasources] (MSC service thread 1-6) JBAS010400: Bound data source [java:jboss/datasources/modulosPlusDS]
```

Ilustración 77 Log WildFly - Conexión a base de datos

3. Servicios: Inicia los servicios del aplicativo web que van a ser consumidos.

DESARROLLO DE UN SISTEMA INFORMÁTICO PARA LA TRAZABILIDAD DE LOS RESULTADOS DE LA MATERIA PRIMA (PLASMAS REACTIVOS Y NO REACTIVOS) EN EL PROGRAMA DE EVALUACIÓN EXTERNO DEL DESEMPEÑO PARA EL CENTRO DE INVESTIGACIÓN PARA LA SALUD EN AMÉRICA LATINA (CISEAL)

```
12:08:38,651 INFO [org.jboss.as.ejb3.deployment.processors.EjbJndiBindingsDeploymentUnitProcessor] (MSC service thread 1-1)
java:global/CISEAL/ServicioCrud!ec.edu.puce.ciseal.crud.ServicioCrud
java:app/CISEAL/ServicioCrud!ec.edu.puce.ciseal.crud.ServicioCrud
java:module/ServicioCrud!ec.edu.puce.ciseal.crud.ServicioCrud
java:global/CISEAL/ServicioCrud
java:app/CISEAL/ServicioCrud
java:module/ServicioCrud

12:08:38,651 INFO [org.jboss.as.ejb3.deployment.processors.EjbJndiBindingsDeploymentUnitProcessor] (MSC service thread 1-1)
java:global/CISEAL/ServicioMateria!ec.edu.puce.ciseal.servicio.ServicioMateria
java:app/CISEAL/ServicioMateria!ec.edu.puce.ciseal.servicio.ServicioMateria
java:module/ServicioMateria!ec.edu.puce.ciseal.servicio.ServicioMateria
java:global/CISEAL/ServicioMateria
java:app/CISEAL/ServicioMateria
java:module/ServicioMateria

12:08:38,651 INFO [org.jboss.as.ejb3.deployment.processors.EjbJndiBindingsDeploymentUnitProcessor] (MSC service thread 1-1)
java:global/CISEAL/RolServicio!ec.edu.puce.ciseal.servicio.ServicioRol
java:app/CISEAL/RolServicio!ec.edu.puce.ciseal.servicio.ServicioRol
java:module/RolServicio!ec.edu.puce.ciseal.servicio.ServicioRol
java:global/CISEAL/RolServicio
java:app/CISEAL/RolServicio
java:module/RolServicio
```

Ilustración 78 Log WildFly - Servicios

4. Validación de tablas: Valida las tablas que se encuentran en la base de datos y todas sus relaciones.

```
INFO [org.hibernate.tool.hbm2ddl.SchemaUpdate] (ServerService Thread Pool -- 50) HHH000228: Running hbm2ddl schema update
INFO [org.hibernate.tool.hbm2ddl.SchemaUpdate] (ServerService Thread Pool -- 50) HHH000102: Fetching database metadata
INFO [org.hibernate.tool.hbm2ddl.SchemaUpdate] (ServerService Thread Pool -- 50) HHH000396: Updating schema
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000261: Table found: public.auditoria
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000037: Columns: [id, fecha, usuario, registro, muestra, accion, ip]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000108: Foreign keys: []
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000126: Indexes: [auditoria_pkey]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000261: Table found: public.casa_comercial
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000037: Columns: [nombre, id, estado, tipo]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000108: Foreign keys: []
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000126: Indexes: [casa_comercial_pkey]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000261: Table found: public.codigos_mapa
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000037: Columns: [nombre, id, estado]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000108: Foreign keys: []
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000126: Indexes: [codigos_mapa_pkey]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000261: Table found: public.institucion
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000037: Columns: [nombre, codigo, id, estado, tipo]
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000108: Foreign keys: []
INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000126: Indexes: [institucion_pkey]
```

Ilustración 79 Log WildFly - Validación de tablas

5. Finalización inicio: El inicio del servidor de aplicaciones se inició correctamente y la aplicación web ha sido correctamente desplegada.

```
10:40:27,095 INFO [org.hibernate.tool.hbm2ddl.TableMetadata] (ServerService Thread Pool -- 50) HHH000126:
Indexes: [secuencias_pkey]
10:40:27,095 INFO [org.hibernate.tool.hbm2ddl.SchemaUpdate] (ServerService Thread Pool -- 50) HHH000232:
Schema update complete
10:40:28,704 INFO [javax.enterprise.resource.webcontainer.jsf.config] (MSC service thread 1-2) Inicializa
ndo Mojarra 2.2.8-jbossorg-1 20140822-1131 para el contexto '/CISEAL'
10:40:34,719 INFO [javax.enterprise.resource.webcontainer.jsf.config] (MSC service thread 1-2) Monitoring
file:/D:/SistemaMuestras/wildfly-8.2.1.Final/standalone/deployments/CISEAL.war/WEB-INF/faces-config.xml f
or modifications
10:40:34,797 INFO [org.primefaces.webapp.PostConstructApplicationEventListener] (MSC service thread 1-2)
Running on PrimeFaces 6.0
10:40:34,844 INFO [org.wildfly.extension.undertow] (MSC service thread 1-2) JBAS017534: Registered web co
ntext: /CISEAL
10:40:34,953 INFO [org.jboss.as.server] (ServerService Thread Pool -- 28) JBAS018559: Deployed "postgresq
l-9.4.1212.jre6.jar" (runtime-name : "postgresql-9.4.1212.jre6.jar")
10:40:34,969 INFO [org.jboss.as.server] (ServerService Thread Pool -- 28) JBAS018559: Deployed "CISEAL.wa
r" (runtime-name : "CISEAL.war")
10:40:35,045 INFO [org.jboss.as] (Controller Boot Thread) JBAS015961: Http management interface listening
on http://127.0.0.1:9990/management
10:40:35,047 INFO [org.jboss.as] (Controller Boot Thread) JBAS015951: Admin console listening on http://1
27.0.0.1:9990
10:40:35,047 INFO [org.jboss.as] (Controller Boot Thread) JBAS015874: WildFly 8.2.1.Final "Tweek" started
in 25324ms - Started 538 of 597 services (100 services are lazy, passive or on-demand)
```

Ilustración 80 Log WildFly - Finalización de inicio

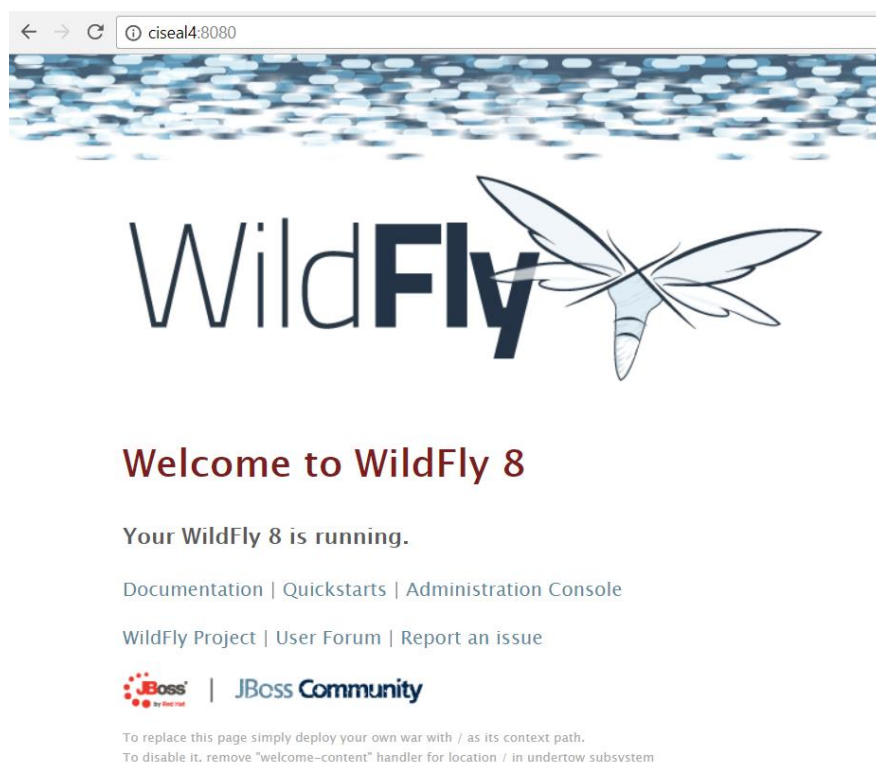
6. Verificación de correcto despliegue: Se verifica revisando el directorio donde se instaló el WAR, este debe tener su similar, pero en esta de desplegado.

wildfly-8.2.1.Final > standalone > deployments

<input type="checkbox"/>	Name	Date modified	Type	Size
	CISEAL.war	17/Jan/2018 12:08 ...	File folder	
	CISEAL.war	17/Jan/2018 12:08 ...	DEPLOYED File	1 KB
	postgresql-9.4.1212.jre6	26/Sep/2017 10:54...	Executable Jar File	661 KB
	postgresql-9.4.1212.jre6.jar	26/Sep/2017 10:54...	DEPLOYED File	1 KB

Ilustración 81 Verificación de despliegue de aplicación

7. Verificación de servicio: Se verifica utilizando un navegador, apuntando al servidor donde se aloja el servidor de aplicación y utilizando el puerto configurado.



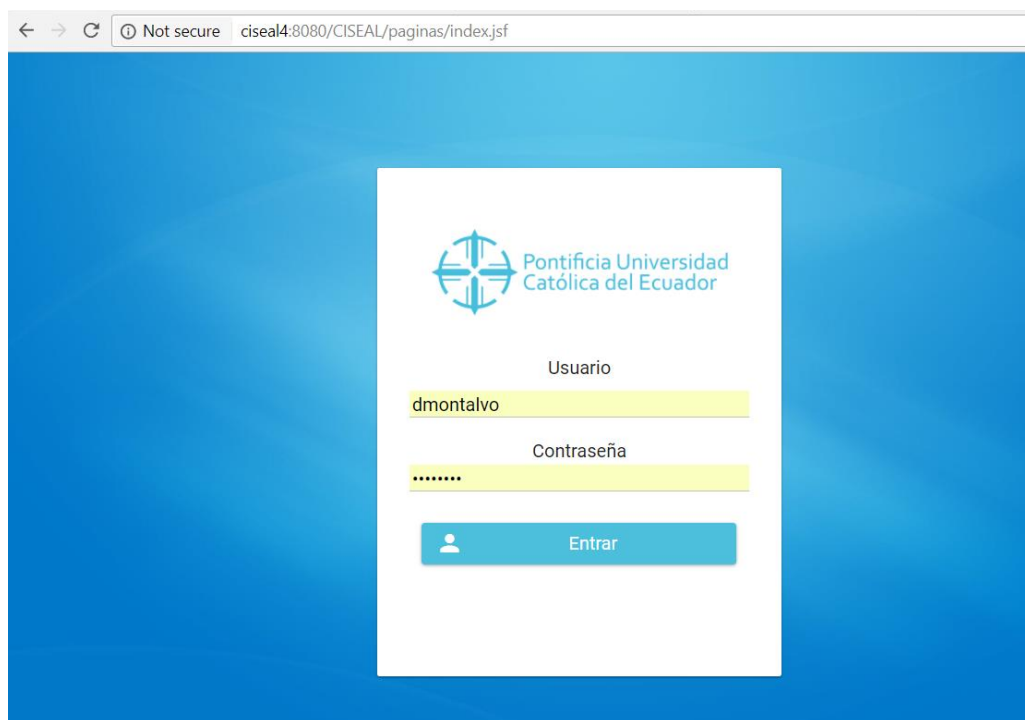


Ilustración 83 Verificación de la aplicación web

Una vez se visualiza la página de inicio y el servidor de aplicaciones está corriendo efectivamente por el puerto configurado, ya se pueden consumir todos los procesos detallados por URBS-PEED en el navegador.

7.6 Interacción del personal de URBS-PEED con la aplicación web

Los procesos se han vuelto más dinámicos gracias al sistema, este ofrece la real posibilidad de tener una trazabilidad sobre los temas críticos en URBS-PEED como los es el pre-análisis, el análisis y el post-análisis, además que ofrecen garantía de monitorizar todos los cambios que se realicen sobre los plasmas reactivos y no reactivos, con el fin de que tengan la consistencia directa con la documentación.

La centralización de la información y los reportes dinámicos ayudan a entender más sobre lo que sucede en URBS-PEED, con el fin de dar calidad a los resultados enviados a los bancos de sangre con la rapidez y eficacia que esta institución necesita.

CAPÍTULO VIII

8. Conclusiones y Recomendaciones

8.1 Conclusiones

- Es de suma importancia sistematizar los procesos de URBS-PEED y CISEAL, ya que se trata de un centro de investigación con certificaciones internacionales y esto va de la mano con el nivel de organización que tengan de la información manejada, los sistemas aportan con esta característica fundamental.
- Las herramientas open source para el desarrollo de software tienen una enorme potencia además de grado de madurez tecnológica, aprovechar estas herramientas y software puede ahorrar muchos costos en temas de sistematización de procesos.
- La centralización de la información es necesaria para el análisis de situación actual de un centro de investigación, si no se tiene bien definida esta característica, no es disponer de indicadores para poder mejorar.
- Una de las características esenciales de URBS-PEED es la calidad, disponer de un control de calidad en los resultados de cada proceso es la fortaleza de los aplicativos web, por lo tanto, es necesario que la aplicación sea desarrollada y probada bajo niveles altos de control y metodologías que aseguren su correcto funcionamiento.
- Scrum es una metodología ágil de desarrollo de software que ayuda a la correcta ejecución de un proyecto de desarrollo, pero si el desarrollador no conoce a profundidad de los procesos a desarrollar, se crean falsas expectativas del sistema que pueden desfavorecer la confianza del cliente.
- La metodología por aplicar en el desarrollo de un proyecto no tiene que ser siempre la misma, sino que tiene que ser seleccionada dependiendo del proyecto y los requerimientos.
- La documentación de los requerimientos, como de cada una de las fases del proceso de desarrollo de software son clave, esta tiene que

estar aprobada por ambas partes para evitar futuras controversias.

- Siempre van a existir cambios sobre el desarrollo del aplicativo web por temas como, reestructuración de procesos, disponer de un código ordenado, correctamente distribuido y reutilizable facilita las modificaciones que se puedan solicitar.
- Cuando se desarrolla un aplicativo es necesario que se tenga como objetivo todos los dispositivos disponibles, como móviles, tabletas, laptops, ya que el fin de una aplicación es ser utilizada sobre cualquier plataforma sin necesidad de alguna dependencia de equipo.
- El aplicativo web para trazabilidad en los procesos de URBS-PEED mejorará enormemente todos los ámbitos de calidad y eliminará el error humano involuntario en el manejo de la información.

8.2 Recomendaciones

- Es importante que cualquier aplicativo que se desarrolle para CISEAL sea alojado en un servidor que tenga mantenimiento del equipo con personal técnico, esto evita que el aplicativo este fuera de línea por razones que el personal de URBS-PEED no pueda solventar.
- Es necesario tener un alcance claro del proyecto para seleccionar la metodología de desarrollo de software a trabajar, así como también establecer los tiempos de entrega.
- Se recomienda hacer un comparativo de usuarios versus recursos que utiliza la aplicación, este análisis nos ayuda a calcular el nivel de recursos que debe consumir un aplicativo para que un número establecido de usuario no genere tráfico y se vea reflejado en una lentitud del sistema.
- Es necesario que el equipo que vaya a trabajar con el sistema este correctamente capacitado tanto en los procesos de URBS-PEED, como en el uso de la aplicación para que la información que se ingrese en el sistema tenga la validez que se necesita.
- Se recomienda tener un mantenimiento frecuente de los respaldos de la información en el aplicativo, para que en caso de algún evento que dañe al equipo donde están alojados los datos, se puedan recuperar

con su respectiva ventana de tiempo.

- CISEAL y URBS-PEED debe automatizar más los procesos que maneja además de centralizar la información en sistemas y bases de datos, para que se consiga una verdadera integración entre los datos que se manejan, con el fin de conseguir indicadores que puedan ayudar a mejorar la calidad.
- Es factor clave en los aplicativos web la interfaz con la cual va a interactuar el usuario, esta debe cumplir con todas las normas de experiencia del usuario, además de tener un diseño amigable a prueba de errores.

Bibliografía

Gnu.org. (2017). ¿Qué es el software libre? - Proyecto GNU - Free Software Foundation. [online] Obtenido de: <https://www.gnu.org/philosophy/free-sw.es.html>

Technet.microsoft.com. (2017). Introducción a Servidor de aplicaciones. [online] Obtenido de: [https://technet.microsoft.com/es-es/library/cc725986\(v=ws.11\).aspx](https://technet.microsoft.com/es-es/library/cc725986(v=ws.11).aspx)

Instructivo para tutorías en la PUCE-SI. (2016). Versión 2.0, p.3. Obtenido de: http://www.pucesi.edu.ec/web/wp-content/uploads/2016/09/INSTRUCTIVO_PARA_TUTORIAS_V_2_0_2016.pdf

PUCE - Historia del Centro de Enfermedades Infecciosas y Crónicas. Obtenido de: <http://www.puce.edu.ec/portal/content/Centro%20de%20Investigaci%C3%B3n%20en%20Enfermedades%20Infecciosas/386?link=oln30.redirect>

Proyectos Agiles ORG. Qué es Scrum. obtenido de: <https://proyectosagiles.org/que-es-scrum/>

Programa de Evaluación Externo de Desempeño (PEED). Informe de resultados generales. (2017 06 15).

Aplicaciones web Características, funcionamiento y estructura. Nacho Iborra. (2012).

Metodología para el Análisis de Requerimientos de Software. López, M. d. (2001). Requerimientos de Software conducente al Reúso.

Sitio PrimeFaces (2017) Why PrimeFaces. A Different Point of View. obtenido de: <https://www.primefaces.org/whyprimefaces/>

Java EE 7 Development with WildFly, Michal Cmil, Francesco Marchioni. (2014).

PostgreSQL 9 Administration Cookbook. Simon Riggs, Gianni Ciolli, Hannu Krosing y Gabriele Bartolin. (2015).