

Tipo de artículo: Artículo original
Temática: Soluciones informáticas
Recibido: 12/11/19 | Aceptado: 22/01/2020 | Publicado: 02/02/2020

Herramienta web para la creación de personalizaciones de Nova Servidores

Web tool for creating Nova Servers customizations

Javier Piñero Cárdenas^{1*}, Yileni Hechavarría González¹

¹ Departamento Sistemas Digitales, Facultad 1. Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños Km 2 ½, Torrens, Boyeros, La Habana, Cuba jpineiro@uci.cu

* Autor para correspondencia: jpineiro@uci.cu

Resumen

GNU/Linux es un sistema operativo tipo Unix, de código abierto, con grandes capacidades para el procesamiento y la administración de los recursos, que mantiene la seguridad como pilar importante. En la actualidad muchos países principalmente de América Latina están optando por esta alternativa, entre ellos se encuentra Cuba que está inmerso en un proceso de migración a software libre donde la distribución cubana de GNU/Linux Nova, es el sistema operativo seleccionado para ser desplegado en los Organismos de Administración Central del Estado. Este sistema operativo cuenta con una versión para servidores a la que frecuentemente se le realizan personalizaciones que consumen mucho tiempo debido a que se desarrolla de forma manual. El objetivo del presente trabajo de diploma es desarrollar una herramienta web que permita agilizar la construcción de personalizaciones de Nova Servidores en el proceso de desarrollo de soluciones a la medida. Para guiar el proceso de construcción de la propuesta de solución se utilizó la metodología de desarrollo de software Variación de AUP para la universidad y en la implementación se emplearon tecnologías y herramientas de software libre. La evaluación de la propuesta de solución se realizó a partir de la aplicación de técnicas, métricas y pruebas que garantizan el correcto funcionamiento de la aplicación y demostraron la satisfacción del cliente hacia el sistema desarrollado. Al culminar la presente investigación se obtuvo una herramienta web que permite la construcción de personalizaciones de Nova Servidores.

Palabras clave: *GNU/Linux, herramienta, personalizaciones, sistema operativo.*

Abstract

GNU / Linux is an open source Unix-type operating system, with great capabilities for processing and resource management, which maintains security as an important pillar. At present, many countries mainly in Latin America are opting for this alternative, among them is Cuba that is immersed in a process of migration to free software where the Cuban distribution of GNU / Linux Nova, is the operating system selected to be deployed in the Central State Administration Organizations. This operating system has a version for servers that frequently perform customizations that take a lot of time because it is developed manually. The objective of this diploma work is to develop a web tool that allows speeding up the construction of Nova Servidores customizations in the process of developing customized solutions. To guide the construction process of the solution proposal, the methodology of software development AUP variation for the university was used and in the implementation free software technologies and tools were used. The evaluation of the solution proposal was made from the application of techniques, metrics and tests that guarantee the correct operation of the application and demonstrated customer satisfaction towards the developed system. Upon completion of this research, a web tool was obtained that allows the construction of Nova Servidores customizations.

Keywords: *customizations, GNU / Linux, operating system, tool*

Introducción

Un sistema operativo es el software básico de una computadora que provee una interfaz entre el resto de los programas del ordenador, los dispositivos hardware y el usuario. Sus funciones básicas son administrar los recursos y las tareas de la máquina, coordinar el hardware y organizar archivos y directorios en dispositivos de almacenamiento (Carmona, 2012). La parte más importante de un sistema operativo es su núcleo. En un sistema GNU/Linux, Linux es el núcleo, el resto del sistema consiste en otros programas, algunos de los cuales fueron escritos por o para el proyecto

GNU¹, este en sí mismo no forma un sistema operativo funcional, se prefiere utilizar el término “GNU/Linux” para referirse a los sistemas que la mayor parte de las personas llaman de manera informal “Linux” (Debian, 2016).

GNU/Linux es un sistema operativo tipo Unix, diseñado para aprovechar al máximo las capacidades de las computadoras basadas en el microprocesador i386 y posteriores. Es un sistema con capacidades de multiprocesamiento, multitarea y multiusuario. Garantiza la protección de la memoria entre procesos, carga de ejecutables por demanda, utiliza memoria virtual de paginación (sin intercambio de procesos completos) y control de tareas POSIX (del inglés *Portable Operating System Interface*, Interfaz Portable de Sistema Operativo) (Linux-es, 2015).

Este sistema operativo se ha distinguido por su estabilidad de operación, su velocidad, su seguridad como pilar importante y su capacidad para la administración eficiente de los recursos de la computadora como la memoria, poder de la unidad central de procesamiento y espacio en disco. Todas estas características lo han llevado a diferenciarse del resto de los sistemas operativos del mundo, pero sin dudas, la más interesante de todas es que se compone de código abierto, es decir, su desarrollo se hace abiertamente y el código puede ser modificado y distribuido libremente. Esto ha propiciado que en la actualidad las distribuciones GNU/Linux hayan alcanzado una significativa popularidad ampliando su uso a varias esferas de la sociedad, convirtiéndose en una alternativa a nivel mundial (Linux-es, 2015).

GNU/Linux no ha conseguido introducirse en los ordenadores domésticos y solo tiene el 2,31% de uso en la cuota mundial (Market Share Reports, 2016), pero reina en móviles, empresas, centros de datos y en la infraestructura de la red. El 80% de las transacciones bursátiles se realizan en sus plataformas y 9 de cada 10 superordenadores llevan el símbolo del pingüino en su interior. También ha expandido su uso a televisiones y automóviles (Reventós, 2012).

En América Latina varios países están optando por esta alternativa; el Gobierno de Venezuela estableció por decreto presidencial el uso preferente del Software Libre (SL) y de GNU/Linux en toda la administración pública, incluso en los ministerios y oficinas gubernamentales (Chávez, 2004). En este país se desarrolla la distribución de GNU/Linux, Canaima, usada en el proyecto Canaima Educativo y en la producción de computadoras de escritorio que se ofrecen a bajo precio (Gavasa, 2014). En el 2008 el gobierno de Ecuador mediante el Decreto 1014 estableció la obligatoriedad del uso de tecnologías de código abierto en entes públicos, este marco legal permitió el desarrollo de eCURUL un programa que permite realizar votaciones electrónicas y controla los debates gubernamentales en las sesiones de la Asamblea Nacional (Correa, 2008).

¹ GNU es un acrónimo recursivo que significa GNU No es Unix (*GNU is Not Unix*).

Uruguay aprobó en diciembre de 2013 la Ley de Software Libre y Formatos Abiertos en el Estado. Esta ley establece la obligación de uso de formatos abiertos y estándares en las instituciones gubernamentales, además prioriza el desarrollo del software libre para las soluciones informáticas (El Senado y la Cámara de Representantes de la República Oriental del Uruguay, 2014).

La República Dominicana a través de distintas ONG (Organización No Gubernamental) apoya y promueve el uso del SL en la educación y en el desarrollo científico. En Argentina existen iniciativas estatales para promover el uso del SL y en Chile el Ministerio de Educación y la Universidad de la Frontera crearon EduLinux, programa que se aplica en las escuelas chilenas y en el 90% de las bibliotecas (Gavasa, 2014).

Cuba en aras de ganar en soberanía tecnológica y socio-adaptabilidad, así como garantizar la informatización de todas las esferas de la sociedad está inmersa en un proceso de migración a software libre donde la distribución cubana de GNU/Linux Nova, es el sistema operativo seleccionado para ser desplegado en los Organismos de Administración Central del Estado (OACE). Este sistema operativo desarrollado por el proyecto Nova, perteneciente al Centro de Software Libre (CESOL) de la Universidad de las Ciencias Informáticas (UCI), cuenta con una versión para servidores a la que frecuentemente se le realizan personalizaciones con el objetivo de facilitar la adaptabilidad del sistema a las necesidades del cliente. Este engorroso proceso actualmente consume mucho tiempo ya que se desarrolla de forma manual, haciéndose necesario la participación de varios especialistas para realizar la tarea, los cuales deben poseer amplios conocimientos sobre el tema debido a la cantidad y complejidad de los comandos que se utilizan en él. Todo esto trae consigo la pérdida oportuna de trabajo para CESOL.

Dada la situación problemática anteriormente planteada, se define como **problema de la investigación**: ¿cómo agilizar el proceso de construcción de personalizaciones de Nova Servidores en el desarrollo de soluciones a la medida?

Materiales y métodos

- **Histórico lógico:** este método permitió conocer la evolución de las herramientas para las personalizaciones de software, así como los elementos que las caracterizan.
- **Análisis y síntesis:** este método se utilizó para el estudio y análisis de diferentes fuentes bibliográficas con el objetivo de obtener un amplio conocimiento acerca las características y peculiaridades del proceso de creación de personalizaciones para distribuciones GNU/Linux.
- **Modelación:** se utilizó para modelar los diagramas de requisitos, análisis y diseño e implementación en la construcción de la herramienta para la creación de personalizaciones para Nova Servidores.

- **Entrevista:** este método se utilizó para conocer las características del proceso de desarrollo de personalizaciones de Nova Servidores en la actualidad.

Personalización de un sistema operativo

La Real Academia de la Lengua Española define la palabra personalizar como: “Dar carácter personal a algo”.

El sitio web Gesoft plantea que la personalización del software es el desarrollo de sistemas, aplicaciones, programas, sitios web, páginas web o cualquier software, de acuerdo con las especificaciones del cliente, es decir, se construye y forma a las necesidades únicas y gustos del cliente (Gesoft, 2015).

De lo anteriormente planteado el autor establece como personalización de un sistema operativo, la modificación del mismo a través de acciones realizadas sobre el conjunto de aplicaciones y configuraciones que lo conforman, con el objetivo de agregar características al sistema y adaptarlo a las necesidades y gustos de sus usuarios finales.

Herramientas de personalización de distribuciones

Instalinux

Instalinux es un servicio en línea que permite elegir entre varias distribuciones de GNU/Linux, configurarlas y personalizarlas. Para ello posee un asistente en inglés, en el que se ingresan los datos de configuración de la instalación y al finalizar se crea la imagen ISO² para descargar según las especificaciones del usuario. Instalinux cuenta con dos versiones, una simple para la mayoría de usuarios y una avanzada para usuarios más experimentados. Permite elegir entre siete distribuciones de Linux como base: CentOS, Debian, Fedora, Mint, OpenSUSE, Scientific y Ubuntu. De cada una, dispone la versión más reciente y algunas anteriores, opción muy útil para aprovechar equipos antiguos de bajas prestaciones de hardware (Linux en español, 2017).

En cuanto a la arquitectura, básicamente se puede optar por procesadores de 32 y de 64 bits. Respecto a la instalación, permite configurar una instalación vía internet o una más clásica con CD (del inglés *Compact Disc*, Disco Compacto), así como aspectos más locales como la hora, el idioma de la distribución, además de definir una contraseña de administrador. El proceso es relativamente rápido, más configurable cuanto más conocimiento se posea (Linux en español, 2017).

² ISO archivo donde se almacena una copia o imagen exacta de un sistema de ficheros.

SUSE Studio

SUSE Studio es una aplicación web creada por SUSE (del inglés *Software und Systementwicklung*, Desarrollo de Sistemas y de Software) para el diseño de versiones personalizadas de esta conocida distribución de GNU/Linux. Para usarla es necesario crear una cuenta o usar credenciales de *Google, Twitter, Facebook o Yahoo*. Una vez iniciada la sesión, cuenta con un asistente que guía el proceso paso a paso. Permite elegir qué versión de SUSE utilizar, la versión gratuita OpenSUSE o las versiones profesionales de SUSE en sus distintas variantes. También se puede optar por versiones más personalizadas de la galería SUSE, donde hay propuestas interesantes pensadas para públicos muy concretos con necesidades muy específicas. El siguiente paso será seleccionar los paquetes a instalar y las dependencias de estos se añaden automáticamente (Linux en español, 2017).

Otras opciones a tener en cuenta son las relacionadas con la configuración del sistema operativo, pudiendo configurar prácticamente cualquier aspecto. Además, se tiene la opción de incluir archivos personales, como manuales o documentos, en directorios concretos y finalmente, guardar el instalador en una imagen ISO normal o en otros formatos VMDK³ (del inglés Virtual Machine Disk, Disco de Máquina Virtual) o VHD⁴ (del inglés Virtual Hard Disk, Disco Duro Virtual) propios de máquinas virtuales (Linux en español, 2017).

1.2.3 Reconstructor

Reconstructor está presente como una aplicación de escritorio desde hace algún tiempo, pero ahora hace su aparición como aplicación web que permite crear y personalizar un LiveCD de Debian o Ubuntu, al más puro estilo de SUSE Studio. Primeramente, se debe seleccionar la versión base a modificar y los paquetes que se pretenden usar, esto último desde repositorios (incluidos universe y multiverse), luego se puede personalizar la “nueva” distribución en la sección “módulos”. Una vez concluidos los pasos, se podrá crear y descargar la imagen (Infonucleo, 2017).

Otras características (Infonucleo, 2017):

- Posibilidad de personalizar la imagen de arranque, el *usplash* (aplicación que dibuja el logo de Ubuntu con su barra de progreso durante la carga del sistema operativo), fondos de escritorio, pantalla de GNOME (del inglés *GNU Network Object Model Environment*, Entorno de Modelo de Objetos de Red GNU).
- Tema GDM (del inglés *GNOME Display Manager*, Gestor de Visualización de GNOME).

³ VMDK (.vmdk) es un formato de archivo que describe los contenedores usados en discos duros virtuales para ser utilizados en máquinas virtuales.

⁴ VHD es un formato de archivo que representa una unidad de disco duro virtual.

- Permite configurar el repositorio para la personalización generada.

Análisis de las herramientas de personalización

A continuación, se hace un análisis de las herramientas de personalización estudiadas en el epígrafe 1.2 con el objetivo de conocer si cumplen con las necesidades existentes en la creación de personalizaciones para la distribución de GNU/Linux Nova en su versión para servidores. En la comparación se utilizaron cuatro criterios de análisis:

Criterios de análisis:

- Tecnología web.
- Selección de paquetes.
- Personalizaciones para servidores.
- Personalizaciones con distribución GNU/Linux Nova como base.

Tabla 1. Resultado del análisis de las herramientas de personalización de distribuciones

(Fuente: elaboración propia)

Criterios de análisis	Herramientas seleccionadas		
	Instalinux	SUSE Studio	Reconstructor
Tecnología web	Sí	Sí	Sí
Selección de paquetes	Sí	Sí	Sí
Personalizaciones para servidores	No	Sí	No
Personalizaciones con la distribución GNU/Linux Nova como base	No	No	No

El análisis anterior sobre las diferentes herramientas de personalización demuestra la necesidad de desarrollar una nueva solución ya que estas herramientas no resuelven las carencias que presenta el proyecto Nova Servidores para la personalización de su sistema operativo. El estudio realizado permitió definir un conjunto de aportes y limitaciones, visibles en la Tabla 2, que fueron utilizados como referencia para el diseño de la propuesta de solución.

Tabla 2. Aportes y limitaciones de las herramientas analizadas

(Fuente: elaboración propia)

Herramientas	Aportes	Limitaciones
Instalinux	Permite crear el usuario administrador y configurar las opciones de red.	No permite realizar una personalización de una distribución que no sean las que están

		predefinidas.
SUSE Studio	Las dependencias de los paquetes seleccionados se añaden automáticamente.	Solo permite hacer personalizaciones de OpenSuse.
Reconstructor	Permite configurar el repositorio para la personalización generada.	Solo permite realizar personalizaciones de Debian y Ubuntu.

Resultados y discusión

La Figura representa la arquitectura de la aplicación en el marco de trabajo según el patón arquitectónico MVT.



Figura 1. Arquitectura de la solución.

(Fuente: elaboración propia)

Estructura de la aplicación

models.py. Contiene cuatro clases:

- Operaciones, es la clase responsable del trabajo con los ficheros (crearlos, eliminarlos, modificarlos).
- Preseed, es la clase responsable del trabajo con el fichero de preconfiguración (escribir configuraciones para la instalación).
- LocalRepo, es la clase responsable de crear el repositorio local (descargar los paquetes y sus dependencias desde el repositorio externo).
- CrearISO, clase que extiende de *Thread* (Hilo) encargada de hacer la llamada a los métodos para la creación del ISO.

views.py: Contiene todas las *views* encargadas de acceder a las clases del *models.py* y llamar a las plantillas contenidas en la carpeta *templates*.

Templates: Es la carpeta que contiene todas las plantillas utilizadas en la aplicación y llamadas desde las *views*.

A continuación, se presenta el modelado del diseño de la propuesta de solución.

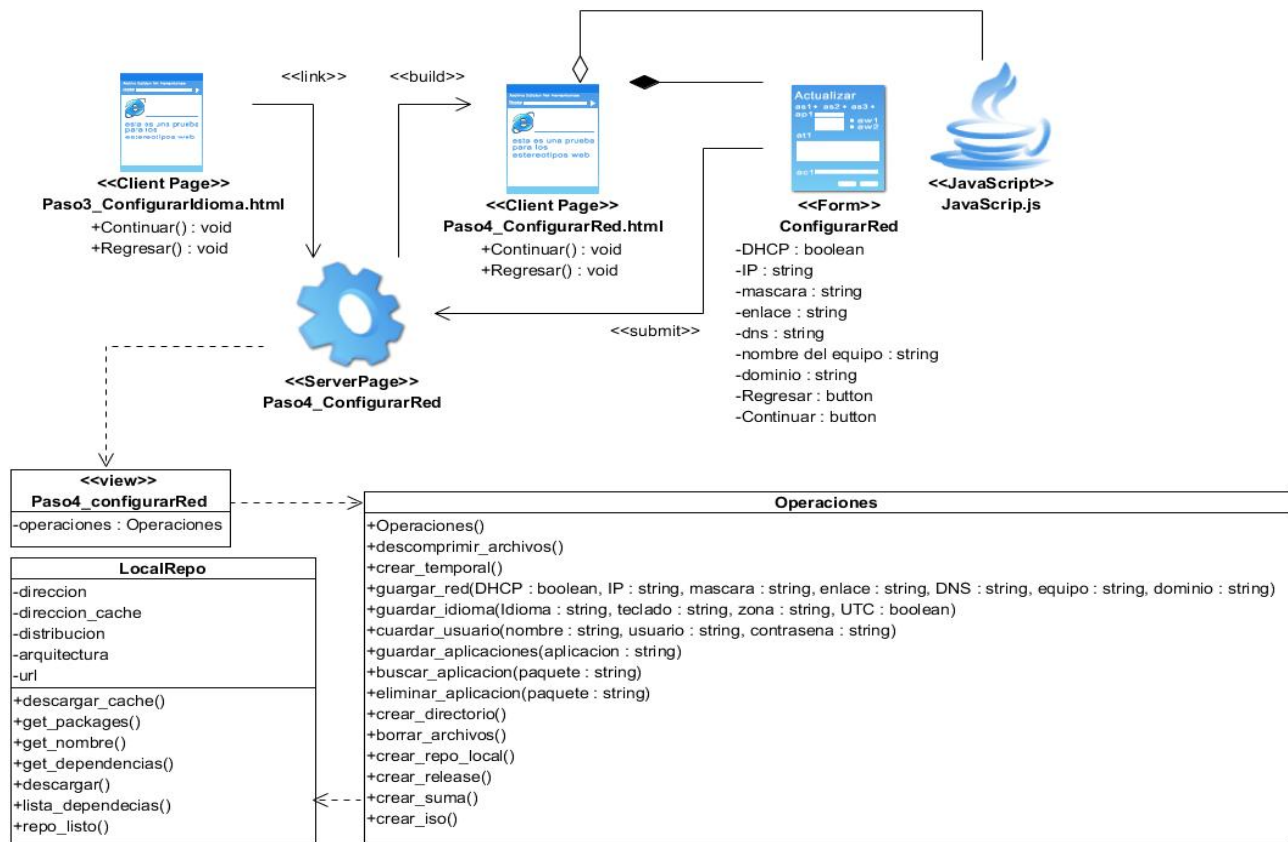


Figura 2. Diagrama de clases del diseño.
 (Fuente: elaboración propia)

El diagrama de clases del diseño permitió definir estructura estática de las clases en el sistema, su contenido y las relaciones que se establecen entre ellas.

Métrica tamaño operacional de clase (TOC)

Está dado por el número de métodos asignados a una clase y evalúa los siguientes atributos de calidad:

- Responsabilidad: mide la responsabilidad que tiene una clase determinada. Un aumento del TOC implica un aumento de la responsabilidad asignada a la clase.
- Complejidad de implementación: evalúa la complejidad de implementación que tiene una clase del diseño. Un aumento del TOC implica un aumento de la complejidad de implementación de la clase.

- Reutilización: indica el grado de reutilización que tiene una clase. Un aumento del TOC implica una disminución del grado de reutilización de la clase.

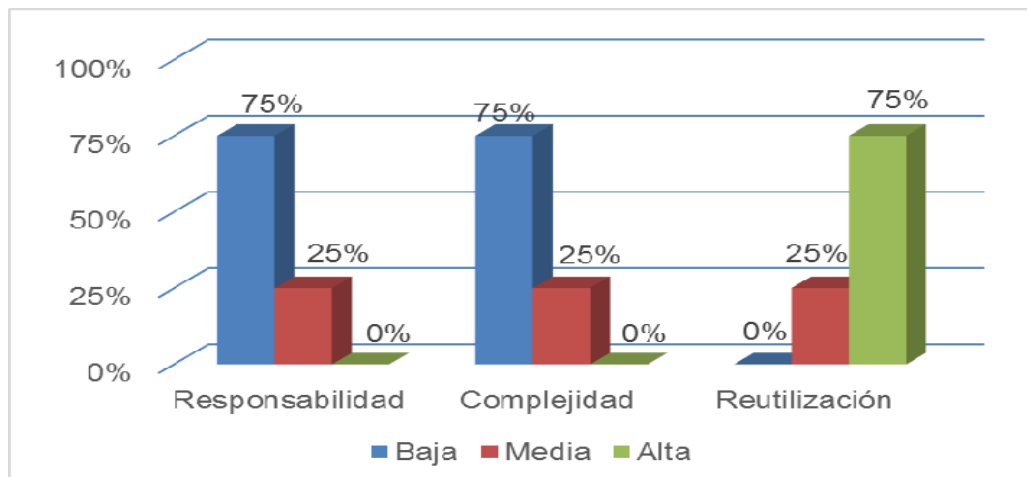


Figura 3. Resultados de la aplicación de la métrica TOC.

(Fuente: elaboración propia)

Métrica Relaciones entre Clases (RC)

Está dado por el número de relaciones de uso de una clase con otra y evalúa los siguientes atributos de calidad:

- Acoplamiento: representa las conexiones físicas (colaboraciones) entre las clases del diseño. Un aumento del RC implica un aumento del Acoplamiento de la clase.
- Cantidad de pruebas: evalúa la cantidad de pruebas necesarias para probar una clase. Un aumento del RC implica un aumento de la Cantidad de pruebas de unidad necesarias para probar la clase.
- Complejidad de mantenimiento: mide la complejidad de mantenimiento que tiene una clase determinada. Un aumento del RC implica un aumento de la complejidad del mantenimiento de la clase.
- Reutilización: proporciona una medida de la reutilización de una clase. Un aumento del RC implica una disminución en el grado de reutilización de la clase.

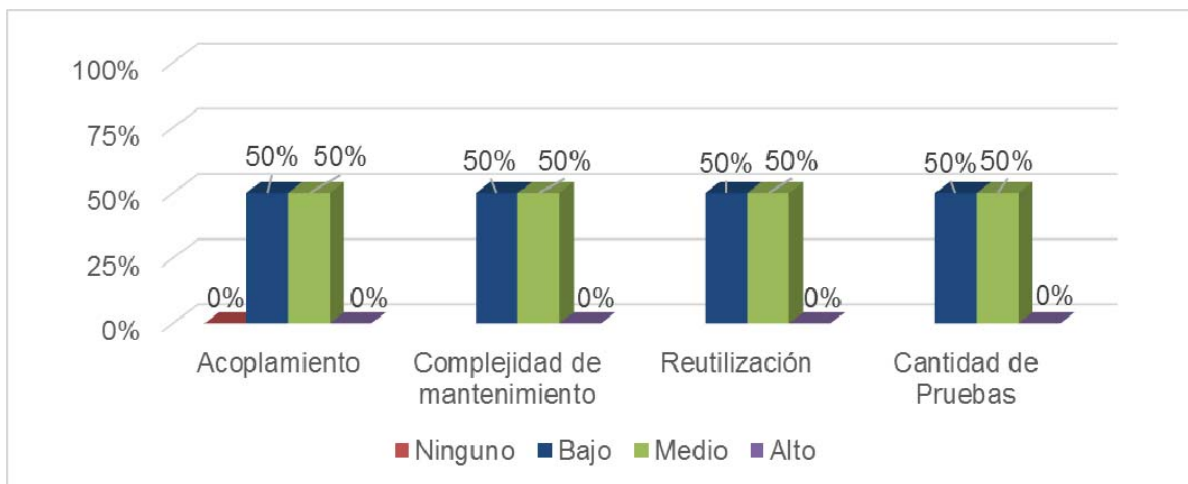


Figura 4. Resultado de la aplicación de la métrica RC.

(Fuente: elaboración propia)

El resultado de aplicar la métrica RC evidencia un 50% de bajo acoplamiento, un 50 % de la baja complejidad de mantenimiento, un 50% de la baja cantidad de pruebas y un 50% de alta reutilización que poseen las clases diseñadas para el sistema.

La aplicación de las métricas TOC y RC permitió tener una medida cuantitativa del grado en el que la aplicación cumple con los indicadores de calidad anteriormente descritos. Proporciona una visión del diseño realizado en el proceso de desarrollo de la propuesta de solución y demuestra que las clases del sistema poseen una baja carga de responsabilidades, un bajo acoplamiento y una alta reutilización permitiendo una fácil implementación, una baja complejidad de mantenimiento y baja la cantidad pruebas de unidad necesarias para probar las clases, pudiéndose afirmar que los resultados obtenidos por la aplicación y evaluación de las métricas son positivos.

A continuación, se presenta el diagrama de despliegue para la propuesta de solución.

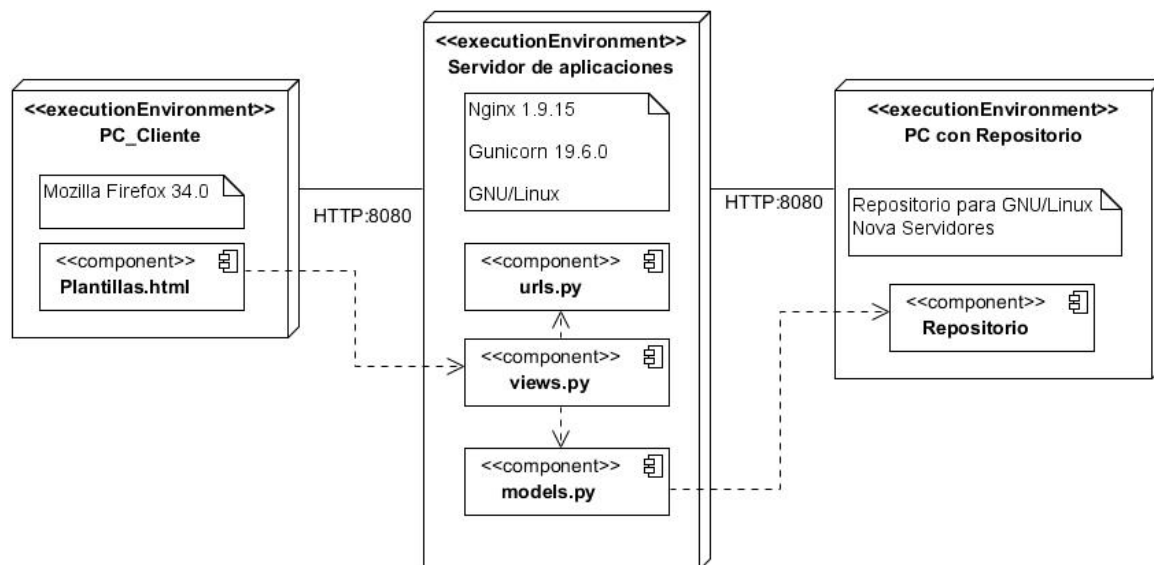


Figura 5. Diagrama de despliegue.
 (Fuente: elaboración propia)

Descripción de los nodos

PC_Cliente: contendrá el navegador web Mozilla Firefox 34.0 o superior mediante el cual los usuarios tendrán acceso al sistema y harán uso del mismo.

Servidor de aplicaciones: proporciona los servicios de la aplicación a las computadoras cliente y gestiona las funciones de la lógica de negocio y de acceso a los datos necesarios, los servidores de aplicaciones serán Nginx 1.9.15 y Gunicorn 19.6.0 y el sistema operativo una distribución de GNU/Linux.

PC_Repositorio: Repositorio de Nova Servidores y que proporciona los paquetes necesarios en el modelo.

Conclusiones

La investigación realizada cumple con los objetivos planteados mediante el desarrollo de la herramienta web para la creación de personalizaciones de Nova Servidores y se arriba a las siguientes conclusiones:

- El análisis de los referentes teóricos y de los sistemas informáticos estudiados evidenció la necesidad de desarrollar una herramienta para agilizar el proceso de construcción de personalizaciones de Nova Servidores.
- Se obtuvo una herramienta web que permite la construcción de personalizaciones de Nova Servidores cumpliendo con los requisitos definidos por el cliente.

- La evaluación de la investigación se realizó a partir de la aplicación de técnicas, métricas y pruebas que garantizan el correcto funcionamiento de la aplicación y demostraron la satisfacción del cliente hacia el sistema desarrollado.

Referencias

- 1&1 Internet España S.L.U. 2016. Servidor web: definición, historia y programas. [En línea] 25 de 10 de 2016. [Citado el: 28 de 2 de 2017.] <https://www.1and1.es/digitalguide/servidores/know-how/servidor-web-definicion-historia-y-programas/>.
- Acosta, Tomás y Hernández Rodríguez, Nodelvis. 2013. Módulo de reportes webmétricos para el motor de búsqueda Orión. Trabajo de Diploma para optar por el Título de Ingeniero en Ciencias Informáticas. [En línea] 2013. [Citado el: 22 de 4 de 2017.] https://repositorio_institucional.uci.cu/jspui/handle/ident/8296.
- Alvarez, Miguel Angel. 2014. DesarrolloWeb. Qué es MVC. [En línea] 2 de 1 de 2014. [Citado el: 5 de 12 de 2016.] <http://www.desarrolloweb.com/articulos/que-es-mvc.html>.
- BBVAOpen4U. 2016. BBVAOpen4U. Django: guía rápida para desarrollar páginas web con este framework. [En línea] 2016. [Citado el: 29 de 10 de 2016.] <https://bbvaopen4u.com/es/actualidad/django-guia-rapida-para-desarrollar-paginas-web-con-este-framework>.
- Campos, Cindy. 2015. Las pruebas en el desarrollo de software. Trabajo de Diploma para optar por el Título de Ingeniero en Computación. [En línea] 2015. [Citado el: 16 de 2 de 2017.] <http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/7627/Las%20pruebas%20en%20el%20desarrollo%20de%20software.pdf?sequence=1>.
- Carmona, Armando. 2012. Desarrollo de un sistema de información web para la automatización de los procesos de control académico, gestión administrativa, emisión de reportes y pagos. Caso: unidad educativa “Juan Bautista Arismendi”. Tesis de grado presentado. [En línea] 1 de 2012. [Citado el: 20 de 11 de 2016.] <http://www.miunespace.une.edu.ve/jspui/bitstream/123456789/1220/1/TG4700.pdf>.
- Chávez, Hugo. 2004. Decreto N° 3.390. [En línea] 28 de 12 de 2004. [Citado el: 29 de 5 de 2017.] <http://www.wipo.int/edocs/lexdocs/laws/es/ve/ve052es.pdf>.
- Correa, Rafael. 2008. Decreto Ejecutivo N° 1014. [En línea] 10 de 4 de 2008. https://softwarelibre.conocimiento.gob.ec/wpcontent/uploads/2016/04/Decreto_1014_software_libre_Ecuador_c2d0b.pdf.
- Dirección Técnica de la Producción. 2010. Programa de mejora. Estándar de codificación para Python. Univercidad de las Ciencias Informáticas. La Habana. Cuba : s.n., 2010.

- El Senado y la Cámara de Representantes de la República Oriental del Uruguay. 2014. Ley N° 19.179. Software Libre y Formatos Abiertos en el Estado. [En línea] 8 de 1 de 2014. [Citado el: 29 de 5 de 2017.] <http://antel.com.uy/wps/wcm/connect/67df2d0045ea0b53a081e68320768a44/03--ley19179.pdf?MOD=AJPERES.28879>.
- Gavasa, Juan. 2014. Panamericanworld. [En línea] 2014. [Citado el: 6 de 11 de 2016.] <http://www.panamericanworld.com/es/articulo/software-libre-cada-vez-mas-utilizado-latino-america>.
- Gesoft. 2015. Personalización de software. [En línea] 2015. [Citado el: 8 de 12 de 2016.] <http://www.gesoft.com.br/vista/desarrollo/personalizacion-de-software.jsp>.
- Guerra, César Arturo. 2017. SG Buzz. Obtención de Requerimientos. Técnicas y Estrategia. [En línea] 2017. [Citado el: 13 de 02 de 2017.] <https://sg.com.mx/revista/17/obtencion-requerimientos-tecnicas-y-estrategia>.
- Gunicorn.org. 2017. Gunicorn. [En línea] 2017. [Citado el: 7 de 5 de 2017.] <http://gunicorn.org>.
- Infonucleo. 2017. Crear CDs de Ubuntu personalizados con Reconstructor. [En línea] 2017. [Citado el: 21 de 1 de 2017.] <http://www.infonucleo.com/2010/05/13/crear-cds-de-ubuntu-personalizados-con-reconstructor/>.
- LibrosWeb. 2017. El patrón de diseño MTV. [En línea] 2017. [Citado el: 17 de 02 de 2017.] http://librosweb.es/libro/django_1_0/capitulo_5/el_patron_de_diseno_mtv.html.
- Linux en español. 2017. Cómo crear tu propia distribución Linux. [En línea] 2017. [Citado el: 21 de 1 de 2017.] <http://linuxespanol.arrayzone.com/2015/08/como-crear-tu-propia-distribucion-linux/>.
- Maikel Enrique Pernía Matos, Delio Gabriel Orozco González, Alberto Miguel Nuevo Rojo. 2016. Misox: Personalización de debian GNU/LINUX. [En línea] 2016. [Citado el: 4 de 12 de 2016.] <http://www.informaticahabana.cu/sites/default/files/ponencias/SWL15.pdf>.
- Market Share Reports. 2016. Market Share Reports. [En línea] 11 de 2016. <http://netmarketshare.com/>
- Maya, Esther. 2014. Métodos y técnicas de investigación. [En línea] 2014. [Citado el: 3 de 12 de 2016.] http://arquitectura.unam.mx/uploads/8/1/1/0/8110907/metodos_y_tecnicas.pdf.
- Nginx.org. 2017. nginx. [En línea] 2017. [Citado el: 4 de 4 de 2017.] <https://nginx.org/en/>.
- Pacheco, Lucy Gabriela. 2014. Desarrollo de una aplicación móvil en andorid de soporte para la prevención de recaídas en pacientes en proceso de recuperación del hospital psiquiátrico Humberto Uglade Camacho. Tesis previa a la obtención del título de Ingeniero de Sistemas. [En línea] 2014. [Citado el: 2017 de 5 de 2017.] <http://dspace.ups.edu.ec/bitstream/123456789/6294/1/UPS-CT002853.pdf>.
- Piña , Olga Lidia y Bringas Linare, José Antonio. 2006. La Modelación Teórica como método de la investigación científica. [En línea] 2006. [Citado el: 4 de 12 de 2016.] <http://www.redalyc.org/pdf/3606/360635561003.pdf>.

- Pressman, R.S. 2010. Software engineering: a practitioner's approach. 7th ed. New York : EUA, 2010. ISBN:978-0-07-337597.
- Ruiz, María de los Ángeles. 2016. MOPIGD: Modelo para la implementación de la gestión de documentos en el sistema empresarial cubano. [En línea] 2016. [Citado el: 12 de 3 de 2017.] <http://www.congreso-info.cu/index.php/info/2016/paper/viewFile/287/40>.
- Sánchez, Manuel Alejandro. 2015. Personalización de GNU/Linux Nova para el Sistema de Educación Primaria en Cuba. Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas. [En línea] 2015. [Citado el: 2017 de 5 de 23.] https://repositorio_institucional.uci.cu/jspui/handle/123456789/7101.
- Sánchez, Tamara Rodríguez. 2015. Metodología de desarrollo para la actividad productiva de la UCI. Universidad de las Ciencias Informáticas. La Habana. Cuba: s.n., 2015.
- Silega, Nemury. 2014. Método para la transformación automatizada del modelo de procesos de negocio a modelo de componentes para sistemas de gestión empresarial. Tesis presentada en opción al grado científico de Doctor en Ciencias Técnicas. [En línea] 2014. [Citado el: 20 de 1 de 2017.] https://repositorio_institucional.uci.cu/jspui/handle/ident/8584.
- Softwareoito. 2017. ¿Qué es la Interfaz Gráfica de Usuario? [En línea] 2017. [Citado el: 22 de 03 de 2017.] <https://www.softwareoito.es/definicion/definicion-interfaz-grafica-de-usuario.html>.
- Sommerville, I. 2011. Ingeniería de software. Novena ed. España : Pearson, 2011. ISBN:9786073206044.
- Sparxsystems. 2017. Sparxsystems. Diagrama de Componentes UML 2. [En línea] 2017. [Citado el: 22 de 03 de 2017.] http://www.sparxsystems.com.ar/resources/tutorial/uml2_componentdiagram.html.
- Ubuntu-es. 2016. Ubuntu-es. Sobre Ubuntu. [En línea] 2016. [Citado el: 8 de 12 de 2016.] http://www.ubuntu-es.org/sobre_ubuntu.
- Usemoslinux. Cómo personalizar una imagen de Ubuntu según tus necesidades | Desde Linux. [En línea] [Citado el: 29 de 10 de 2016.] <http://blog.desdelinux.net/como-personalizar-una-imagen-de-ubuntu-segun-tus-necesidades/>.
- Villafane, Agustin. 2017. Modelo de procesos de negocio. [En línea] 2017. [Citado el: 11 de 2 de 2017.] <http://strellis.com.ar/articulos/soa/modelo/>.
- Villafuerte, Deymor B. Centty. 2006. Manual metodológico para el investigador científico. [En línea] 2006. <http://www.eumed.net/libros-gratis/2010e/816/METODO%20DEL%20ANALISIS%20SINTESIS.htm>.
requerimientos/.