



UNIVERSIDAD ESTADAL DE MILAGRO
FACULTAD CIENCIAS E INGENIERÍA

PLANTEAMIENTO DE LA PROPUESTA DE TRABAJO DE
INTEGRACIÓN CURRICULAR
PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN
SISTEMAS COMPUTACIONALES

TEMA: DESARROLLO DE UNA APLICACIÓN WEB PARA
EVALUAR LA ADOPCIÓN DE SOFTWARE LIBRE EN
INSTITUCIONES PÚBLICAS Y PRIVADAS.

Autores:

Sr. Ramirez Arreaga Daniel Bryan

Sr. Silva Acosta Angel Ismael

Profesor de la asignatura:

Mgr. Rea Sánchez Víctor Hugo

Milagro, Febrero 2020

ECUADOR

DERECHOS DE AUTOR

Ingeniero.
Fabricio Guevara Viejó, PhD.
RECTOR
Universidad Estatal de Milagro
Presente.

Yo, Ramirez Arreaga Bryan Daniel, en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de integración curricular, modalidad presencial, mediante el presente documento, libre y voluntariamente procedo a hacer entrega de la Cesión de Derecho del Autor, como requisito previo para la obtención de mi Título de Grado, como aporte a la Línea de Investigación "1S2019 UIC TECNOLOGÍAS DE LA INFORMACIÓN Y DE LA COMUNICACIÓN", de conformidad con el Art. 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, concedo a favor de la Universidad Estatal de Milagro una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservo a mi favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo a la Universidad Estatal de Milagro para que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Milagro, 20 de febrero de 2020



Ramirez Arreaga Bryan Daniel
Autor 1
C.I: 094036266-8

DERECHOS DE AUTOR

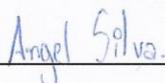
Ingeniero.
Fabricio Guevara Viejó, PhD.
RECTOR
Universidad Estatal de Milagro
Presente.

Yo, Silva Acosta Angel Ismael, en calidad de autor y titular de los derechos morales y patrimoniales del trabajo de integración curricular, modalidad presencial, mediante el presente documento, libre y voluntariamente procedo a hacer entrega de la Cesión de Derecho del Autor, como requisito previo para la obtención de mi Título de Grado, como aporte a la Línea de Investigación "1S2019 UIC TECNOLOGÍAS DE LA INFORMACIÓN Y DE LA COMUNICACIÓN", de conformidad con el Art. 114 del Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación, concedo a favor de la Universidad Estatal de Milagro una licencia gratuita, intransferible y no exclusiva para el uso no comercial de la obra, con fines estrictamente académicos. Conservo a mi favor todos los derechos de autor sobre la obra, establecidos en la normativa citada.

Así mismo, autorizo a la Universidad Estatal de Milagro para que realice la digitalización y publicación de este trabajo de integración curricular en el repositorio virtual, de conformidad a lo dispuesto en el Art. 144 de la Ley Orgánica de Educación Superior.

El autor declara que la obra objeto de la presente autorización es original en su forma de expresión y no infringe el derecho de autor de terceros, asumiendo la responsabilidad por cualquier reclamación que pudiera presentarse por esta causa y liberando a la Universidad de toda responsabilidad.

Milagro, 20 de febrero de 2020

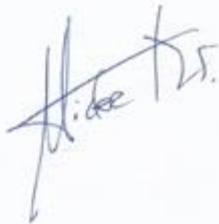


Silva Acosta Angel Ismael
Autor 2
C.I: 095798325-7

APROBACIÓN DEL TUTOR DEL TRABAJO DE INTEGRACIÓN CURRICULAR

Yo, Rea Sánchez Víctor Hugo en mi calidad de tutor del trabajo de integración curricular, elaborado por los estudiantes Ramirez Arreaga Bryan Daniel y Silva Acosta Angel Ismael, cuyo título es "DESARROLLO DE UNA APLICACIÓN WEB PARA EVALUAR LA ADOPCIÓN DE SOFTWARE LIBRE EN INSTITUCIONES PÚBLICAS Y PRIVADAS", que aporta a la Línea de Investigación "1S2019 UIC TECNOLOGÍAS DE LA INFORMACIÓN Y DE LA COMUNICACIÓN", previo a la obtención del Título de Grado INGENIERO EN SISTEMAS COMPUTACIONALES; considero que el mismo reúne los requisitos y méritos necesarios en el campo metodológico y epistemológico, para ser sometido a la evaluación por parte del tribunal calificador que se designe, por lo que lo APRUEBO, a fin de que el trabajo sea habilitado para continuar con el proceso previa culminación de Trabajo de Integración Curricular de la Universidad Estatal de Milagro.

Milagro, 20 de febrero de 2020



Rea Sánchez Víctor Hugo

Tutor

C.I: 091744052-1

APROBACIÓN DEL TRIBUNAL CALIFICADOR

El tribunal calificador constituido por:

Mgr. Rea Sanchez Victor Hugo.

Mgr. Arevalo Gamboa Lissett Margarita.

Mgr. Mendoza Cabrera Denis Dario.

Luego de realizar la revisión del Trabajo de Integración Curricular, previo a la obtención del título (o grado académico) de INGENIERO EN SISTEMAS COMPUTACIONALES presentado por el estudiante Ramirez Arreaga Bryan Daniel

Con el tema de trabajo de Integración Curricular: "DESARROLLO DE UNA APLICACIÓN WEB PARA EVALUAR LA ADOPCIÓN DE SOFTWARE LIBRE EN INSTITUCIONES PÚBLICAS Y PRIVADAS".

Otorga al presente Trabajo de Integración Curricular, las siguientes calificaciones:

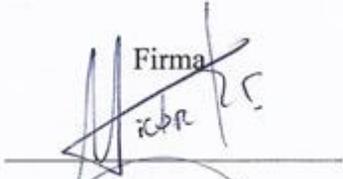
Trabajo Curricular	Integración	[60]
Defensa oral		[40]
Total		[100]

Emite el siguiente veredicto: (aprobado/reprobado)

Aprobado

Fecha: 20 de febrero de 2020

Para constancia de lo actuado firman:

	Nombres y Apellidos	Firma
Presidente	Rea Sanchez Victor Hugo.	
Secretario /a	Arevalo Gamboa Lissett Margarita.	
Integrante	Mendoza Cabrera Denis Dario.	

APROBACIÓN DEL TRIBUNAL CALIFICADOR

El tribunal calificador constituido por:

Mgtr. Rea Sanchez Victor Hugo.

Mgtr. Arevalo Gamboa Lissett Margarita.

Mgtr. Mendoza Cabrera Denis Dario.

Luego de realizar la revisión del Trabajo de Integración Curricular, previo a la obtención del título (o grado académico) de INGENIERO EN SISTEMAS COMPUTACIONALES presentado por el estudiante Silva Acosta Angel Ismael

Con el tema de trabajo de Integración Curricular: "DESARROLLO DE UNA APLICACIÓN WEB PARA EVALUAR LA ADOPCIÓN DE SOFTWARE LIBRE EN INSTITUCIONES PÚBLICAS Y PRIVADAS".

Otorga al presente Trabajo de Integración Curricular, las siguientes calificaciones:

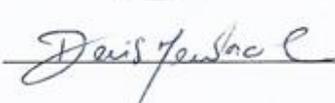
Trabajo Curricular	Integración	[60]
Defensa oral		[40]
Total		[100]

Emite el siguiente veredicto: (aprobado/reprobado)

Aprobado

Fecha: 20 de febrero de 2020

Para constancia de lo actuado firman:

	Nombres y Apellidos	Firma
Presidente	Rea Sanchez Victor Hugo.	
Secretario /a	Arevalo Gamboa Lissett Margarita.	
Integrante	Mendoza Cabrera Denis Dario.	

DEDICATORIA

Estudiante 1

Yo, Daniel Ramirez A.

Dedico este proyecto a quienes me han apoyado incondicionalmente a lo largo de este trayecto. Mi etapa universitaria no ha sido sencilla, pero tuve la oportunidad de vivirla junto a personas especiales que han formado parte de mi vida desde que tengo uso de razón y otras a quienes no me arrepiento de haber conocido.

A Mis amigas Lissette y Jazmín, las cuales me brindaron su apoyo incondicional en este camino, siendo mis compañeras desde primer semestre y reforzando ese vínculo durante este tiempo, enseñándome valores importantes como el compañerismo, la amistad y el trabajo en equipo.

A mi hermana Eli, por apoyarme en los momentos más difíciles, por escucharme y entenderme, motivándome a seguir adelante y ayudándome a crear lindos recuerdos que quedarán en mi memoria por el resto de mi vida.

A mi tía Katy, por ser como mi segunda madre, preocupándose por mí desde el momento en que llegue a este mundo y por enseñarme a ser una mejor persona al mostrarme el valor e importancia de la familia.

A mis abuelos Gustavo y Clara, por enseñarme a ser agradecido con la vida y por creer en mí.

A Dios por darme vida y por bendecirme con una familia maravillosa.

Por último me gustaría dedicar este trabajo a la persona más especial.

A mi madre Ruth, el ser al que más amo en este mundo, por creer en mí y no dejarme caer, por enseñarme a ser fuerte ante la vida y no rendirme, por inculcarme los valores que hacen grande a un ser humano, como la humildad, la superación y perseverancia, ella es mi razón de estar aquí.

DEDICATORIA

Estudiante 2

Yo, Angel Silva A.

Dedico este proyecto a cada persona que ha marcado en mi vida y me han mostrado su apoyo de manera brindándome ánimos y confianza en mis capacidades y méritos.

A mi padre Jacobo y a mi madre Katty, los cuales son la fuerza y el motor en mi vida y quienes me han guiado e inculcado mediante valores que han servido en mi formación como ser humano.

A mis hermanos Anabel, Pamela y Ronald, por ser ejemplo en mi vida y motivarme a ser mejor.

A mis amigos Alexis, Álvaro, Bryan e Isaac, con quienes he formado lazos de confianza y apoyo incondicional en objetivos y metas en la vida.

A mi amiga Leslie, por darme su apoyo en mis buenos y malos momentos y por haber compartido de sus consejos que me han ayudado a superar mis debilidades.

A la pareja conformada por Jimmy y Erika, quienes fueron las primeras personas, con excepción de mi familia, en apoyarme y aconsejarme al iniciar mis estudios y de quienes he aprendido y me he fortalecido moralmente.

A Dios, por el regalo de la vida y por sostener a mi familia en los malos y buenos momentos.

Y por último y no menos importante.

A los lectores, ya que son ellos quienes dan valor a todo este trabajo hecho con entrega, amor y dedicación, y que por ende estoy satisfecho de brindarles todo este conocimiento que me ha costado obtener durante mi experiencia académica en colaboración con mi compañero de tesis.

AGRADECIMIENTO

Estudiante 1

Yo, Daniel Ramirez A.

Son muchas las personas que han contribuido en mi formación universitaria a lo largo de estos años, he aquí donde resaltare a aquellos docentes que semestre a semestre se dedicaron a compartir sus ideales, valores, tiempo y conocimiento. Dedicando cada clase a impartir aquellos valores que deben formar parte de un profesional, en donde sobresale la responsabilidad, el compromiso, la ética y la moral, también el ser visionarios, emprendedores y capaces de cumplir con nuestras metas.

A mis maestros, agradezco a Víctor Hugo Rea S, mi tutor de tesis, quien fue el primero en creer en este proyecto, brindándonos su apoyo como profesional desde el principio, hasta alcanzar el punto de culminación de este trabajo y a Lissett Margarita Arévalo G, docente de la materia “Diseño de Integración Curricular”, quien se encargó de gestionar las revisiones de avances de mi proyecto a la par con mi tutor, a lo largo del periodo, demostrando así su apoyo en este trayecto de forma profesional.

A la institución, agradezco sinceramente a la Universidad Estatal de Milagro por la oportunidad de formar parte de ella y brindarme los recursos para alcanzar la obtención de mi título como Ingeniero en Sistemas Computacionales y convertirme en un profesional.

A mí, agradezco el haber tenido la fuerza de voluntad para terminar este capítulo en mi vida, sé que aún me queda mucho por recorrer, pero es un pequeño gran comienzo, una meta alcanzada y me siento muy orgulloso de aquello, esto no hubiera sido posible sin las personas a quienes más amo, mi familia, estoy muy seguro de que ellos también lo estarán.

Gracias.

AGRADECIMIENTO

Estudiante 2

Yo, Angel Silva A.

Quiero agradecer a toda persona que han sido parte en mi formación universitaria y que ha influido en mí en todo este largo trayecto de aprendizaje y experiencia.

A mis docentes, que debido a su vasta experiencia y sacrificio he aprendido lo útil e importante de cada materia, valorando sus esfuerzos impartidos que mediante su preparación y desvelo han sido responsables de llevar su cátedra hacia sus estudiantes.

A la Universidad Estatal de Milagro, que como institución me ha acogido y ha sido como un hogar en donde pude desenvolverme social y académicamente con la tecnología que está a su disposición.

A mi tutor de tesis, Víctor Hugo Rea S, el cual mediante su guía y experiencia este trabajo de tesis ha sido culminado de manera satisfactoria.

Gracias.

ÍNDICE GENERAL

DEDICATORIA	7
AGRADECIMIENTO.....	9
ÍNDICE GENERAL.....	11
RESUMEN.....	12
ABSTRACT.....	13
CAPÍTULO 1	14
1. INTRODUCCIÓN	14
1.1. Planteamiento del problema	16
1.2. Objetivos	17
1.3. Justificación.....	17
1.4. Marco Teórico	18
1.4.1. Software Libre.....	18
1.4.2. Diferencia entre FLOSS y código abierto	20
1.4.3. FLOSS y los modos de clasificar las empresas de software libre	20
1.4.4. Ventajas del software libre en las empresas	23
CAPÍTULO 2.....	25
2. METODOLOGÍA	25
2.1. Proceso de desarrollo	25
2.1.1. Fases de desarrollo	25
CAPÍTULO 3.....	30
3. RESULTADOS (ANÁLISIS O PROPUESTA)	30
CONCLUSIONES	32
RECOMENDACIONES	33
REFERENCIAS BIBLIOGRÁFICAS	34
ANEXOS.....	39

Título de Trabajo Integración Curricular: “DESARROLLO DE UNA APLICACIÓN WEB PARA EVALUAR LA ADOPCIÓN DE SOFTWARE LIBRE EN INSTITUCIONES PÚBLICAS Y PRIVADAS”

RESUMEN

Las ventajas de Free Libre Open Source Software (FLOSS) han sido a menudo discutidas en la literatura reciente de sistemas de información; personalización, acceso libre al código fuente y los ahorros de costos son los principales aspectos positivos reportados por varias fuentes de investigación. Los beneficios que FLOSS puede traer a las organizaciones ya sean estas del sector público o privado son: reducir el fenómeno de la brecha digital, promover las economías locales, evitar dependencia del vendedor y, en una visión más amplia, atacar el dominio monopolístico en el mercado.

El desarrollo del sistema se considera una de las fases más complejas y exigentes dentro de todo proyecto, ya que conlleva una larga y ardua tarea, la cual es condensar todos los conocimientos adquiridos, y ponerlos a prueba para conseguir el producto final. Nuestro sistema se basa en un proceso de evaluación de soluciones informáticas de software libre que valora aspectos técnicos, organizacionales y económicos previo a la selección de la misma a ser implementado en una institución pública o privada.

Los resultados obtenidos nos demuestran que el proyecto desarrollado ha sido favorable para el desarrollo de ciertas destrezas y habilidades, como el desarrollo de destrezas para el trabajo en equipo y habilidades de comunicación, como también de búsqueda de información.

Pudimos vencer barreras existentes en el desarrollo de nuestro proyecto, siendo el mismo la base primordial para alcanzar la meta propuesta, permitiéndonos de este modo, el recolectar e indagar la mayor cantidad de información, con el fin de nutrir dicho estudio y obtener los resultados esperados.

PALABRAS CLAVE: Desarrollo aplicaciones, FLOSS, Adopción, Software Libre.

Título de Trabajo Integración Curricular: “DESARROLLO DE UNA APLICACIÓN WEB PARA EVALUAR LA ADOPCIÓN DE SOFTWARE LIBRE EN INSTITUCIONES PÚBLICAS Y PRIVADAS”

ABSTRACT

The advantages of Free Open Source Software (FLOSS) have often been discussed in recent information systems literature; customization, free access to source code and cost savings are the main positive aspects reported by various research sources. The benefits that FLOSS can bring to organizations in both the public and private sectors are: reducing the phenomenon of the digital divide, promoting local economies, avoiding vendor dependence and, in a broader view, attacking monopolistic dominance in the market.

The development of the system is considered one of the most complex and demanding phases within any project, as it involves a long and arduous task, which is to condense all the knowledge acquired, and put it to the test to achieve the final product. Our system is based on an evaluation process that assesses technical, organizational and economic aspects prior to the selection of free software to be implemented in a public or private institution.

The results obtained show us that the project developed has been favorable for the development of certain skills and abilities, such as the development of skills for teamwork and communication skills, as well as information search.

We were able to overcome existing barriers in the development of our project, being the same the primordial base to reach the proposed goal, allowing us in this way, to gather and to investigate the greater amount of information, in order to nourish this study and to obtain the expected results.

KEY WORDS: Development, Application, Adoption, Free Software, Institutions.

CAPÍTULO 1

1. INTRODUCCIÓN

Las ventajas de Free Libre Open Source Software (FLOSS) han sido a menudo discutidas en la literatura reciente de sistemas de información; personalización, acceso libre al código fuente y los ahorros de costos son los principales aspectos positivos reportados por varias fuentes de investigación. Los beneficios que FLOSS puede traer a las organizaciones ya sean estas del sector público o privado son: reducir el fenómeno de la brecha digital, promover las economías locales, evitar dependencia del vendedor y, en una visión más amplia, atacar el dominio monopolístico en el mercado. Estos son los aspectos que en efecto aumentan la importancia de FLOSS, sin embargo, no podemos descartar el hecho de que las organizaciones en el sector público no se centran en adoptar por completo el software libre, ya que los factores para la adopción en el sector público son diferentes a los privados.

En algunos casos, las migraciones a FLOSS han logrado ser exitosas, pero esto se basa netamente en que las necesidades y requerimientos de la organización estén bien definidos previo a su evaluación.

En este contexto, se genera la pregunta, ¿Por qué el sector público no está completamente de acuerdo en adoptar una nueva solución FLOSS?

La primera razón se refiere a un aspecto endógeno de la innovación, ya que incluso las tecnologías que son beneficiosas para una comunidad pueden tardar varios años en ser ampliamente adoptadas, desplegadas y utilizadas en otra.

Pese a dicho aspecto existe otra razón. FLOSS ha sido descrito como un trastorno en la tecnología. Mientras que la innovación sostenida mejora los productos ya existentes agregándole valor, las tecnologías disruptivas revolucionan la manera de pensar acerca de un producto.

En este caso, tres características comunes de FLOSS son:

- Software como una mercancía;
- Colaboración habilitada en red; y
- Personalización del software (software como servicio).

En primer lugar, la introducción de FLOSS plantea a la organización un radical cambio potencial, que considera a FLOSS una mercancía como un bien público impulsado por estándares abiertos y plataformas abiertas. Entonces, una clase dada de licencias FLOSS facilita la colaboración entre organizaciones que les dan la oportunidad de redistribuir software personalizado a través de una red. Finalmente, FLOSS permite un alto grado de personalización y abre la puerta a nuevas fronteras empresariales. Como tal, la adopción de FLOSS puede determinar un cambio significativo en los procesos de negocio de la organización, para adaptarlos al nuevo software.

Esto hace que la idea de adoptar FLOSS sea convincente, pero también problemática, ya que una migración a FLOSS puede implicar una gran desviación de la infraestructura organizativa tecnología de la información preexistente. Este cambio radical y la desconfianza intrínseca hacia la innovación típicamente retrasan cualquier adopción de FLOSS.

Definimos un marco teórico para la adopción de FLOSS dentro de las organizaciones, refinamos el marco teórico con los comentarios de las partes interesadas y luego construimos la herramienta para la evaluación de adopción de FLOSS. Contribuimos a la investigación en la adopción de FLOSS con:

- La definición de un marco compuesto por dimensiones, factores y subfactores para la caracterización de la adopción de FLOSS;
- La evaluación en los estudios de caso de los factores de adopción del FLOSS mediante un conjunto de métricas identificadas, y la aplicación de un protocolo de

estudio empírico para evaluar los resultados del estudio intra-caso y los estudios inter-caso.

1.1. Planteamiento del problema

Las instituciones públicas y privadas del cantón Milagro tienden a adquirir software sin una previa evaluación o consultoría con respecto a las necesidades que se presentan a nivel organizacional, estas decisiones tienen como consecuencia la adopción del software incorrecto, el cual puede presentar incompatibilidad con el sistema de infraestructura tecnológica de la organización, poca flexibilidad al momento de implementarlo e interfaz poco amigable con el usuario. En este contexto, se podría evidenciar de que el software seleccionado no ha sido la mejor decisión, ya que no cumple con ciertos aspectos técnicos, organizacionales y económicos.

1.2. Objetivos

1.2.1. Objetivo General

Desarrollar una aplicación web que permita evaluar la adopción de FLOSS en instituciones públicas y privadas.

1.2.2. Objetivos Específicos

Para alcanzar el objetivo propuesto es necesario cumplir con los siguientes puntos:

- Identificar dimensiones, factores y subfactores relevantes para el proceso de evaluación de adopción de FLOSS.
- Diseñar la estructura junto al modelo conceptual de la base de datos y la aplicación.
- Desarrollar la estructura de la aplicación mediante el lenguaje de programación Python, usando como motor de base de datos a PostgreSQL e integrando el sistema con Django.

1.3. Justificación

En la actualidad, las instituciones han empezado a apostar por nuevas tecnologías las cuales se adaptan a las necesidades de cada sector; dichas tecnologías abarcan una línea de elementos definidos como hardware y software, aquí es donde interviene FLOSS, ya que basándose en varios estudios se ha llegado a la conclusión de que posee más ventajas ante los sistemas adquiridos bajo licencias. FLOSS resulta ser de mucho provecho para aquellas instituciones que buscan expandirse y posicionarse en el mercado, ya que al ser de uso libre se produce un ahorro de inversión en comparación a los sistemas con licencias, generando así una mayor productividad dentro de la institución, junto con la oportunidad de expansión a mejoras del software sin restricciones y a su vez aprovechando los recursos de su infraestructura TI al máximo.

1.4. Marco Teórico

1.4.1. Software Libre

Hablamos de software libre cuando nos queremos referir a la libertad que tiene un usuario para copiar, distribuir y modificar un software sin que ninguna compañía o individual pueda tomar acciones legales contra él (González, 2004, p. 9).

El error más frecuente de concepción sobre el software libre deriva de la ambigüedad del vocablo anglosajón free, que significa tanto “libre” como “gratis”. Así pues, es usual (y comprensiva) la confusión que la expresión free software puede causar. Entendamos; el software libre es una cuestión de libertad y no de precio (González, 2004, p. 9).

“Es cierto que es muy habitual que el software libre sea también gratuito, pero no necesariamente tiene que ser así” (González, 2004, p. 10).

Los sistemas Open Source se definen como software libre y de código abierto que se han establecido con gran fuerza en el mercado y en el sector tecnológico, desde la fundación de Free Software Foundation (FSF) en 1984 queriendo hacerse un pequeño espacio en el ámbito de software para ordenadores; surgieron debido a que la principal función de los gobiernos era encontrar soluciones que sean eficientes y sostenibles para tecnologías de la información en el ámbito del sector público para incrementar el crecimiento económico y apoyar el bienestar social (Rossi et al., 2012, p. 135).

A pesar de ser más utilizado, FLOSS en los últimos años aún no logra posicionarse como el imperio de Microsoft, las grandes empresas no confían por completo en los sistemas abiertos, pero a pesar de la poca confianza que se le otorgue, la comunidad de software libre cree firmemente en que llegara un punto en que las empresas optaran por recursos de licencia libre (Rossi et al., 2012).

La selección del software en el sector público tiene políticas que involucran el ámbito de seguridad como también el ámbito económico. Se deben considerar aspectos de política que garanticen la seguridad, para que así la entidad pueda abrirse paso a usarla con fines de medios comerciales y sus actividades sean consideradas adecuadas dentro de los márgenes legales.

La industria del software es una de las empresas que más dinamismo tiene a nivel internacional ya que la convergencia entre la informática y las comunicaciones se ha expandido a la aplicación de software, logrando abarcar todas las actividades ya sea sociales o productivas, haciendo que las organizaciones logren tener más ventajas para poder promocionar sus productos, llegando de manera eficiente a los usuarios (Macho et al., 2017). El auge de automatización y desarrollo han contribuido a que se modifiquen las formas de trabajar de cada entidad y sobre todo la forma en la que producen sus bienes o servicios, además las organizaciones hay adoptado nuevos métodos de innovación que han mejorado sus utilidades, ya que les ha permitido llegar a más usuarios que a la final optan por adquirir sus productos o servicios a través de los medios de comunicación (Pessagno et al., 2008, p. 148).

Los usuarios de software libre van aumentando día a día, claro que no como se lo espera, ya que el incremento de estos usuarios es menor a los que usan software de licencias pagadas, las pequeñas empresas se han visto desafiadas y han tenido que idear estrategias de negocios para poder expandir sus actividades FLOSS, haciendo que se idealicen estrategias creativas mejorando el plan de negocio para poder realizar las actividades diarias de la misma manera, pero con recursos que sean más eficientes y que permitan obtener mejores utilidades, dando paso a sistemas automatizados con códigos libres. Las estadísticas de FLOSS no permiten que se distinga el ámbito económico en las diversas empresas de software libre (Macho et al., 2017).

1.4.2. Diferencia entre FLOSS y código abierto

Los dos campos políticos en la comunidad de “Free Software” son el movimiento de Software libre y el Código abierto. El movimiento del software libre es una campaña para la libertad de los usuarios de computadoras; Decimos que un programa no libre es una injusticia para sus usuarios. El campo de código abierto se niega a ver el tema como una cuestión de justicia para los usuarios, y basa sus argumentos solo en beneficios prácticos (Stallman, 2016, párr.1).

Un investigador que estudia las prácticas y los métodos utilizados por los desarrolladores de la comunidad de software libre decidió que estas preguntas eran independientes de las opiniones políticas de los desarrolladores, por lo que utilizó el término "FLOSS", que significa "Free / libre Open Source Software", para evitar explícitamente una preferencia entre los dos campos políticos. Si desea ser neutral, esta es una buena manera de hacerlo, ya que esto hace que los nombres de los dos campos sean igualmente prominentes (Stallman, 2016, párr.3).

Otros usan el término "FOSS", que significa " Free Open Source Software”, muy parecido a "FLOSS", pero en comparación es menos claro, ya que no explica que "Free" se refiere a la libertad y no a la gratuidad (Stallman, 2016, párr.4).

1.4.3. FLOSS y los modos de clasificar las empresas de software libre

La producción de software se encuentra vinculada entre los científicos y los ingenieros que laboran en centros académicos o gubernamentales para que se desarrolle como práctica de la cultura de investigación, el intercambio libre de software, la producción de estos se privatizó en la década de 1970 cuando se empezó a comercializar el pago de estos sistemas.

En gran medida, la diferencia que existe entre el Open Source y versiones de pago se establecieron por medio de organizaciones que buscaron que se privatice ciertas características de los sistemas para entidades privadas, el software en si se utiliza para corporaciones en diversos ámbitos y finalidades, desde los sistemas operativos más básicos hasta los sistemas de gestión ERP; para todos los usos empresariales existen soluciones FLOSS con funcionalidades parecidas a las soluciones propietarias (Marsan & Paré, 2013, p. 735).

Con el software propietario se crean vínculos de dependencia con proveedores de software, a partir de esto se quieren cambios y modificaciones de código, pero con FLOSS no ocurre esta dependencia ya que el código es libre y se puede contratar a cualquier persona para que se pueda desarrollar nuevas modificaciones o funcionalidades (Macho et al., 2017).

El software propietario no dispone de código fuente en las aplicaciones, por lo que se tendrá que confiar que sus desarrolladores hicieron un trabajo confiable y seguro, incluso las brechas de seguridad no se las puede conocer si el proveedor no se las especifica, esto no sucede con FLOSS ya que como se mencionó la característica de código libre hace que sea más probable detectar las vulnerabilidades y corregirlas haciendo que los sistemas sean más seguros ya que son verificados por las mismas entidades (Pessagno et al., 2008, p. 150).

A nivel de producción en las empresas y a nivel económico los términos se pueden usar indistintamente o bien como FLOSS, las ventajas de adopción pueden destacar su potencial ya que permite la difusión de software en la economía, las posibilidades que FLOSS brinda para acceder a los programas gratuitos a un costo menor que los productos que son propietarios y esto hace que se logre reducir la estructura de costos y también la dependencia tecnológica de un proveedor único (Marsan & Paré, 2013, p. 736).

FLOSS puede ser usado en varios sectores sea público o privado, aunque quienes por el momento más interactúan con él son emprendimientos pequeños y medianos con equipos de trabajo que son reducidos pero con capacidades altas, el nivel de trabajo de estas empresas es local, y su objetivo es fortalecer las redes de mercadeo que comparten con otras grandes empresas (Marsan & Paré, 2013, p. 732).

FLOSS ha ganado buena aceptación entre el área de la tecnología, y poco a poco más organizaciones lo están acogiendo como modelo de negocios. La mayoría de los proyectos que involucran a FLOSS se enmarcan en un modelo de desarrollo abierto, donde se toman datos públicos de proyectos que ofrecen la posibilidad de realizar trabajos estructurados con consideraciones de los requerimientos propios de la organización (Pessagno et al., 2008, p. 140).

Para crear una referencia hacia el avance que ha tenido el software libre en la actualidad tenemos a Moodle¹, esta es una herramienta que se basa en la web, desarrollada sobre el lenguaje PHP², con conexión a MySQL³ como base de datos. Posee un diseño sencillo de interpretar, que permite desarrollar modelos para agregar a los proyectos. Mayormente es usado en el ámbito educativo para lograr interacción con los estudiantes y hoy en día es una de las plataformas más usadas.

En diversos países del mundo las empresas publicas iniciaron la migración de su información del software propietario al software libre FLOSS, en Europa se promovió este incentivo bajo una ley establecida por la comisión europea y que a medida que pasan los años se va estableciendo con mayor fuerza (Marsan & Paré, 2013).

Debido al éxito de Apache y luego la expansión de Linux generaron más interés y confiabilidad en estos tipos de software llegando ser considerados alternativas para cambio empresarial, y ahora un modelo de negocios rentable que permitirá obtener mayor utilidades y control propio de la administración organizacional (Bouras et al., 2014, p. 238).

1.4.4. Ventajas del software libre en las empresas

La versatilidad de las herramientas de software que sirven como apoyo para los modelos de negocios que se basan en software libre poseen varias funcionalidades, por lo que existen varios softwares que dan apoyo a lenguajes de programación que intervienen en los modelos de negocios de muchas empresas, pero sobre todo lo que FLOSS ofrecerá a las organizaciones que decidan adoptarlos como suyos será:

- **Reducción de costos:** por ser la mayoría de los programas gratuitos, se convierte en gran aliado de todas aquellas organizaciones que deseen reducir costos en el ámbito de administración e informática, aplicando sistemas que le permitan hacer todas sus actividades, pero con costos menores que los que se obtendrían con Software propietario.
- **Compatibilidad con otros programas:** por lo general esta es una de las inquietudes que más se producen entre las organizaciones, saber si el software libre tendrá la capacidad de poder acoplarse con diferentes programas, por lo general se los diseña para que funcionen con las últimas mejoras de los programas que estén al despuente en la informática.
- **Escalabilidad:** permite realizar modificaciones y además obtener actualizaciones del software, esto permite que se desarrollen nuevos productos con base en el mismo y adaptarlo a las necesidades de cada organización.
- **El código es abierto, no existen secretos:** por lo cual se podrá hacer lo que sea para personalizarlo y una parte importante es que se conocerá exactamente cuáles son las vulnerabilidades que posee el sistema al momento de la inspección.

- **Se pueden hacer cambios con rapidez:** para un conocer del lenguaje en el que funciona el sistema no habrá inconvenientes para poder hacer cambios o mejoras en el sistema ya que posee un lenguaje estandarizado que todo programador podrá entender.
- **No se necesita de grandes requerimientos de hardware para funcionar:** por lo cual puede acoplarse en cualquier organización sin inconvenientes, haciendo que sea fácil y atractivo de implementar en cada empresa.

[1]: <https://moodle.org/?lang=es> [2]: <https://www.php.net> [3]: <https://www.mysql.com>

CAPÍTULO 2

2. METODOLOGÍA

La metodología forma parte del proceso de desarrollo de todo sistema o proyecto que se pretende realizar, ya que nos describe la serie de métodos y técnicas que seguiremos para alcanzar los objetivos propuestos, logrando así aumentar la eficacia al ejecutarlos y consigo, brindándonos un mejor resultado al momento de su entrega final.

2.1. Proceso de desarrollo

Según afirma (Jacobson, 2000), el modelo de proceso más conveniente para el desarrollo de software es un proceso iterativo e incremental, puesto que, en contraste de otros modelos de proceso, como por ejemplo el modelo de cascada, permite obtener diversas versiones del producto software antes de la entrega final del mismo y la depuración y validación progresiva del mismo, lo que sin duda redundará en un software más complaciente para usuarios y cliente.

Dentro del proceso de desarrollo de sistemas se encuentran fases, las cuales detallan los pasos a seguir, priorizando sus actividades en función a los objetivos.

2.1.1. Fases de desarrollo

- **Fase de inicio**

El principal objetivo de esta fase es dar una visión aproximada del proyecto esclareciendo así los verdaderos objetivos del proyecto y el alcance que tendrá. “En esta fase se identifican y priorizan los riesgos más importantes” (Torossi, 2018).

En la fase de “Inicio” encontramos una variable esencial, la cual es el equipo con el que trabajaremos a lo largo del desarrollo, aquí es donde se detallan las especificaciones del sistema.

Personalmente nosotros hemos utilizado un equipo HP con las siguientes características:

- Sistema operativo: Windows 10
- Procesador: Intel(R) Celeron(R) CPU N3060 @ 1.60GHz
- Memoria RAM: 4,00 GB
- Tipo de sistema: Sistema operativo de 64 bits, procesador x64

La fase de inicio involucra una planeación que representa una idea general de las actividades y tiempos estimados para llevar a cabo el desarrollo del proyecto, por tal motivo, hemos creado un **Diagrama de Gantt** con la ayuda de Project para detallar tareas, tiempos y recursos que se ven involucrados en este proceso. Ver Anexo 1.

Nota: Este archivo cuenta con fechas reales de los avances presentados dentro del acompañamiento de tutorías y se apega a la realidad con respecto a la fecha de entrega final del proyecto.

- **Fase de elaboración**

En esta fase se realiza un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta transformarse en el sistema final. Esto debe incluir los casos de uso críticos identificados en la fase de inicio. También se debe demostrar que se han evitado los riesgos más graves, bien con este prototipo, bien con otros de usar y tirar. (Martinez, 2017).

Definimos a la fase de elaboración mediante un **Caso de Uso**, el cual hace mención a los actores que intervienen o participan en el sistema, siendo estos el administrador y el usuario encuestado, donde cada uno es capaz de ejercer funciones distintas dependiendo su rol. Ver Anexo 2.

La creación de la base de datos es uno de los pasos principales dentro del desarrollo de la aplicación. Su modelo conceptual lo obtenemos con la ayuda del software DBWrench (Database Design & Synchronization Software)¹, ya que al trabajar con PostgreSQL no es posible visualizar ni manipular las tablas de forma gráfica. Ver Anexo 7.

- **Fase de construcción**

Esta fase se centra en la construcción del producto, el desarrollo físico del software que se inicia cuando todos los casos de uso han sido implementados, los cuales aún pueden sufrir modificaciones de manera que el sistema está en pleno crecimiento y refinamiento, terminado ello se procede a realizar los códigos de programación, pruebas alfa que dan una visión del producto final. (Torossi, 2018).

Para el desarrollo del sistema utilizamos PyCharm¹ en su versión 5.0 como IDE (Entorno de Desarrollo Integrado), para posteriormente hacer uso de Python² en su versión 3.2.5 como lenguaje de programación, ya que permite la programación orientada a objetos, lo que nos ayudara a crear una conexión con Django³ de manera fácil y eficiente. Hacemos uso de Django en su versión 2.1 como base para el desarrollo de nuestra aplicación web, ya que siendo este un sistema de código abierto, resulta ser uno de los mejores frameworks si de conectividad y extensibilidad se trata.

Como sistema gestor de base de datos empleamos PostgreSQL⁴ en su versión 9.1, el cual nos ayudará a crear una conexión con PyCharm/Python para la creación de modelos y tablas que formarán parte de nuestro sistema.

Al momento de almacenar y manipular los mantenimientos hacemos uso de consultas personalizadas de Django, los cuales nos permitirán ejecutar esta tarea de la manera más óptima y eficiente.

Si hablamos de estética dentro de nuestro proyecto, debemos mencionar el uso de Bootstrap⁵ como biblioteca para los diseños de pantalla de nuestra aplicación web.

Cabe recalcar que optamos por seleccionar estas herramientas de desarrollo, por el hecho de sentirnos mucho más familiarizados al momento de trabajar con ellas.

El desarrollo del sistema se considera una de las fases más complejas y exigentes dentro de todo proyecto, ya que conlleva una larga y ardua tarea, la cual es condensar todos los conocimientos adquiridos, y ponerlos a prueba para conseguir el producto final.

Al ser un sistema de encuestas e interpretación de resultados hemos tomado como referencia el artículo científico; “A framework for evaluating organization's motivation and preparation for adopting product lines” (Rincón et al., 2018), el cual contiene información relevante para nuestro trabajo, ya que mediante métricas estadísticas analiza los factores más importantes que influyen dentro de la evaluación para la adopción de líneas de productos.

El artículo nos brinda un modelo de tablas con una matriz que nos permitirá calcular el valor de importancia de un factor y otros valores decisivos para el proceso de evaluación. Para apreciar la tabla propuesta por los autores. Ver Anexo 4.

Según afirman (Rincón et al., 2018), la escala de respuesta para la valoración de cada factor se basa en una escala tipo Likert con un formato de cinco puntos, en donde: (5) es "completamente de acuerdo", (4) “de acuerdo”, (3) “no saber”, (2) “estar en desacuerdo” y (-1) “completamente en desacuerdo”.

- **Fase de transición**

En esta fase es la entrega (“despliegue”) de software, donde el producto se encamina a la versión beta, las iteraciones en esta fase agregan características al software, que nacen del uso supervisado del usuario (Torossi, 2018).

El usuario puede acceder al dominio o dirección del sistema mediante el servidor de Django para iniciar sesión dependiendo su rol, ya sea este usuario o administrador.

El usuario será capaz de proceder a la evaluación y si en dado momento desea cambiar de idea en cuanto a ponderación, se le brindará la opción de re-evaluación.

El administrador posee todos los derechos y permisos del sistema, es el único que puede crear usuarios y asignar roles, como también visualizar datos estadísticos sobre las encuestas mediante reportes, analizar los movimientos que realizan los usuarios desde el módulo de auditoria, manteniendo así un mejor control; será quien tenga acceso al módulo de mantenimiento, en donde se encuentra la creación y modificación de posteriores evaluaciones, el administrador también puede formar parte de la evaluación.

- **Integración y validación del sistema**

Una vez finalizada la fase de transición se verificará que todos los componentes del sistema funcionen correctamente y que cumplan con los requerimientos establecidos, como también el correcto diseño de la base de datos y el sistema como tal, se harán las respectivas validaciones de los modelos y se comprobará el correcto funcionamiento del sistema que ayudará de manera positiva al momento de ejercer la toma de decisiones dentro de las instituciones.

[1]: <https://www.jetbrains.com/pycharm/>

[2]: <https://www.python.org>

[3]: <https://www.djangoproject.com>

[4]: <https://www.postgresql.org>

[5]: <https://getbootstrap.com>

CAPÍTULO 3

3. RESULTADOS (ANÁLISIS O PROPUESTA)

Como propuesta hacia la solución del problema, se plantea el desarrollo de una aplicación web de autoría propia cuyo nombre ha sido definido como: “Floss System Evaluator”, la cual tiene como finalidad ayudar a despejar las dudas que se presentan ante la suposición de incluir un nuevo sistema abierto dentro de una organización, esto se logrará mediante el uso de una pre-evaluación de elementos, tales como: las dimensiones, factores y subfactores que posee el sistema en cuestión. Una vez obtenida la puntuación general con respecto a los criterios de los encuestados se podrá analizar la factibilidad de aceptación y manejo del mismo, dando así lugar al proceso de adopción de FLOSS adecuado, el cual estará dirigido hacia instituciones tanto públicas como privadas.

Para determinar el grado de cumplimiento de los objetivos propuestos en este proyecto, hemos planteado un análisis de carácter descriptivo el cual nos brindará la información necesaria para conocer si dichos objetivos fueron alcanzados.

- Uno de los primeros objetivos específicos que se pretendieron alcanzar es el de Identificar dimensiones, factores y subfactores relevantes para el proceso de evaluación de adopción de FLOSS. Para alcanzar el objetivo de la identificación de dimensiones y factores se realizó la búsqueda en varias fuentes de información con temas relacionados con la adopción de FLOSS en instituciones públicas y privadas. Según la frecuencia encontrada acerca de las dimensiones y factores en artículos científicos, se realizó una matriz para codificar las mismas y construir los subfactores que son un nivel más detallado de los factores descritos, con la finalidad de proporcionarle al usuario del sistema criterios específicos al momento de evaluar. Ver Anexo 6.

- El segundo objetivo específico se enfoca en diseñar la estructura junto al modelo conceptual de la base de datos y la aplicación. Claramente se puede señalar como un objetivo cumplido, ya que el diseño y creación de la base de datos es uno de los principales pasos dentro del desarrollo de la aplicación. El modelo conceptual de la base de datos lo obtuvimos con la ayuda del software DBWrench (Database Design & Synchronization Software)¹, ya que al trabajar con PostgreSQL no es posible visualizar ni manipular las tablas de forma gráfica. Ver Anexo 7.
- El tercer y último objetivo específico se refiere netamente al desarrollo de la estructura de la aplicación mediante el lenguaje de programación Python, usando como motor de base de datos a PostgreSQL e integrando el sistema con Django. Esto quiere decir, la culminación y entrega final del sistema, en donde podemos afirmar que si se cumplió de la manera que se esperaba. Ver Anexo 3.
- Definimos como objetivo general del proyecto el “Desarrollar una aplicación web que permita evaluar la adopción de FLOSS en instituciones públicas y privadas”. Se considera como un objetivo alcanzado, ya que al cumplir con los objetivos específicos obtendremos como producto final nuestro sistema “System Floss Evaluator”, el cual nos propusimos desarrollar desde un comienzo. Ver Anexo 5.

Nota: En el Anexo 5 podemos visualizar la pantalla de inicio de sesión, que a su vez se encuentra contenida en el segmento de Funcionalidad del sistema, en donde también se exponen las demás capturas de pantalla del sistema junto a una breve descripción de su funcionalidad y uso.

- Los resultados obtenidos nos demuestran que el proceso escogido ha sido favorable para el desarrollo de destrezas y habilidades, tales como el desempeño de técnicas para el trabajo en equipo, junto con habilidades de comunicación y búsqueda de información.

[1]: <http://www.dbwrench.com/index.shtml>

CONCLUSIONES

La importancia de soluciones informáticas para evaluar distintos factores que forman parte de próximos sistemas a implementarse desempeñan un papel importante dentro de las organizaciones y crece en gran medida con el paso del tiempo, ya que resulta de mucho provecho para el área gerencial y personal responsable del departamento de tecnología de la información conocer las ventajas o desventajas que dicha solución informática pueda ofrecer, lo cual representa ser una ventaja al momento de prevenir problemas a futuro, ya que dichas áreas son las responsables de una correcta selección del software a implementarse en las instituciones..

Empleando datos reales para el análisis de evaluación y adopción de sistemas abiertos, nuestra App, “System Floss Evaluator” hace posible el ingreso de información como punto de partida (dimensiones, factores y subfactores) para generar una evaluación detallada, crear reportes y gráficos estadísticos con facilidad, contribuyendo de esta manera a una mejor toma de decisión de la solución informática a implementar.

El presente estudio puede servir para trabajos futuros y ser usado como referencia para ampliar el tema de migración y adopción de FLOSS dentro de instituciones públicas y privadas.

RECOMENDACIONES

En primer lugar, motivamos a todos los estudiantes e investigadores interesados en el proyecto a continuar mejorando las bases de este sistema con respecto a la tecnología moderna, corrección y optimización del mismo.

Como recomendación principal, sugerimos adaptar el sistema a teléfonos móviles, con el fin de dar soporte a las pantallas de baja resolución, esto se debe a que el sistema fue creado en base a resoluciones mayores de 540 x 529.

Otra recomendación sería incluir mensajes de estado, chat de contacto para aclarar las posibles dudas o preguntas que el evaluador posea y mejorar la pantalla de ingreso de datos a una de mejor comodidad al usuario.

REFERENCIAS BIBLIOGRÁFICAS

- Alduraywish, Y., Xu, Y., & Salonitis, K. (2017). Evaluating state of information systems failure in developing countries using ITPOSMO model. *ICAC 2017 - 2017 23rd IEEE International Conference on Automation and Computing: Addressing Global Challenges through Automation and Computing*, (September), 7–8. <https://doi.org/10.23919/IconAC.2017.8082032>
- Aversano, L., & Tortorella, M. (2013). Quality evaluation of floss projects: Application to ERP systems. *Information and Software Technology*, 55(7), 1260–1276. <https://doi.org/10.1016/j.infsof.2013.01.007>
- Ayala, C., Cruzes, D. S., Franch, X., & Conradi, R. (2011). Towards Improving OSS Products Selection – Matching Selectors and OSS Communities Perspectives, 244–258. https://doi.org/10.1007/978-3-642-24418-6_17
- Badampudi, D., Wohlin, C., & Petersen, K. (2016). Software component decision-making: In-house, OSS, COTS or outsourcing - A systematic literature review. *Journal of Systems and Software*, 121, 105–124.
- Beck K, Addison W. (2000). "Extreme Programming Explained. Embrace Change", Pearson Education, 1999. Traducido al español como: "Una explicación de la programación extrema. Aceptar el cambio".
- Belle, J. Van, & Reed, M. (2012). OSS Adoption in South Africa: Applying the TOE Model to a Case Study, 304–309.
- Benlian, A., & Hess, T. (2011). Comparing the relative importance of evaluation criteria in proprietary and open-source enterprise application software selection - a conjoint study of ERP and Office systems. *Information Systems Journal*, 21(6), 503–525. <https://doi.org/10.1111/j.1365-2575.2010.00357.x>
- Bouras, C., Filopoulos, A., Kokkinos, V., Michalopoulos, S., Papadopoulos, D., & Tseliou, G. (2014). Policy recommendations for public administrators on free and open source software usage. *Telematics and Informatics*, 31(2), 237–252. <https://doi.org/10.1016/j.tele.2013.06.003>
- Bouras, C., Kokkinos, V., & Tseliou, G. (2013). Methodology for Public Administrators for selecting between open source and proprietary software. *Telematics and Informatics*, 30(2), 100–110. <https://doi.org/10.1016/j.tele.2012.03.001>
- Damiani, E., Riehle, D., Frati, F., & Wasserman, A. I. (2015). Open source systems: Adoption and impact. *IFIP Advances in Information and Communication Technology*, 451, iii–iv. <https://doi.org/10.1007/978-3-319-17837-0>
- Damiani, E., Riehle, D., Frati, F., & Wasserman, A. I. (2015). Open source systems: Adoption and impact. *IFIP Advances in Information and Communication Technology*, 451, iii–iv. <https://doi.org/10.1007/978-3-319-17837-0>

- Daneshgar, F., Low, G. C., & Worasinchai, L. (2013). An investigation of “ build vs . buy ” decision for software acquisition by small to medium enterprises. *Information and Software Technology*, 55(10), 1741–1750.
<https://doi.org/10.1016/j.infsof.2013.03.009>
- Del Bianco, V., Lavazza, L., Morasca, S., & Taibi, D. (2011). A survey on open source software trustworthiness. *IEEE Software*, 28(5), 67–75.
<https://doi.org/10.1109/MS.2011.93>
- Del Bianco, V., Lavazza, L., Morasca, S., Taibi, D., & Tosi, D. (2010). An investigation of the users’ perception of OSS quality. *IFIP Advances in Information and Communication Technology*, 319 AICT, 15–28.
- EcuRed. (2018). Metodologías de desarrollo de Software. Obtenido de EcuRed:
https://www.ecured.cu/Metodologias_de_desarrollo_de_Software
- Gangadharan, G. R., & Butler, M. (2012). Free and Open Source Software Adoption in Emerging Markets: An Empirical Study in the Education Sector Factors Favoring the Adoption of FOSS, 244–249.
- Gangadharan, G. R. (2017). Open source solutions for cloud computing. *Computer*, 50(1), 66–70.
- González, D. (2004). Software libre en los institutos. Vol (1), 9-10.
https://www.cs.upc.edu/~tonis/daniel_gonzalez_pinyero.pdf
- Grifol, D. (2018). Metodologías ágiles de desarrollo de software. Obtenido de
<http://danielgrifol.es/metodologias-agiles-de-desarrollo-de-software/>
- Hauge, Ø., Cruzes, D. S., Conradi, R., Velle, K. S., & Skarpenes, T. A. (2010). Risks and risk mitigation in open source software adoption: Bridging the gap between literature and practice. *IFIP Advances in Information and Communication Technology*, 319 AICT(1), 105–118.
- Jacobson, I. B. (2000). El proceso unificado de desarrollo de software. Addison-Wesley Professional
- Jeffries, R., Anderson, A., Hendrickson, C. (2001), "Extreme Programming Installed". Addison-Wesley.
- Joia, L. A., & dos Santos Vinhais, J. C. (2017). From closed source to open source software: Analysis of the migration process to Open Office. *Journal of High Technology Management Research*, 28(2), 261–272.
<https://doi.org/10.1016/j.hitech.2017.10.008>
- Kuechler, V., Jensen, C., & Bryant, D. (2013). Misconceptions and Barriers to Adoption of FOSS in the U . S . Energy Industry, 232–244.

- Li, Y., Tan, C. H., & Yang, X. (2013). It is all about what we have: A discriminant analysis of organizations' decision to adopt open source software. *Decision Support Systems*, 56(1), 56–62. <https://doi.org/10.1016/j.dss.2013.05.006>
- López, D., De Pablos, C., & Santos, R. (2010). Profiling F/OSS adoption modes: An interpretive approach. *IFIP Advances in Information and Communication Technology*, 319 AICT, 354–360. https://doi.org/10.1007/978-3-642-13244-5_31
- Lundell, B., Lings, B., & Syberfeldt, A. (2011). Practitioner perceptions of Open Source software in the embedded systems area. *Journal of Systems and Software*, 84(9), 1540–1549. <https://doi.org/10.1016/j.jss.2011.03.020>
- Macho, H., Robles, G., & Gonzalez, J. (2017). Evaluation of FLOSS by Analyzing Its Software Evolution:: An Example using the Moodle Platform. *Journal of Information Technology Research*. https://www.researchgate.net/publication/276461811_Evaluation_of_FLOSS_by_Analyzing_Its_Software_Evolution_An_Example_using_the_Moodle_Platform
- Marsan, J., & Paré, G. (2013). Antecedents of open source software adoption in health care organizations: A qualitative survey of experts in Canada. *International Journal of Medical Informatics*, 82(8), 731–741.
- Martinez, R. (2017). Guía a Rational Unified Process. Obtenido de researchgate.net: https://www.researchgate.net/publication/268005509_Guia_a_Rational_Unified_Process
- Munoz-Cornejo, G., Seaman, C. B., & Koru, A. G. (2011). An Empirical Investigation into the Adoption of Open Source Software in Hospitals. *International Journal of Healthcare Information Systems and Informatics*, 3(3), 16–37. <https://doi.org/10.4018/jhisi.2008070102>
- Nagy, D., Yassin, J. M., & Bhattacharjee, A. (2010). Organizational adoption of open source software: barriers and remedies. *Communications of the ACM*, 53(3), 148–151.
- OK Hosting. (2016). Metodologías de Desarrollo de Software. Obtenido de Ok Hosting: https://okhosting.com/blog/metodologias-del-desarrollo-desoftware/#En_que_consisten_las_Metodologias_de_Desarrollo_de_Software
- Paschali, M. E., Ampatzoglou, A., Bibi, S., Chatzigeorgiou, A., & Stamelos, I. (2017). Reusability of open source software across domains: A case study. *Journal of Systems and Software*, 134, 211–227. <https://doi.org/10.1016/j.jss.2017.09.009>
- Pessagno, L., Domínguez, K., Rivas, L., Perez, M., Mendoza, L., & Méndez, Ed. (2008). Modelo de calidad para herramientas FLOSS que dan apoyo al modelado de procesos del negocio. *Revista Española de Innovación, Calidad e Ingeniería del Software*, 4, 1. https://www.researchgate.net/publication/255632144_Modelo_de_calidad_para_herramientas_FLOSS_que_dan_apoyo_al_modelado_de_procesos_del_negocio

- Petrinja, E., Sillitti, A., & Succi, G. (2011). Adoption of OSS Development Practices by the Software Industry: A Survey, 233–243. https://doi.org/10.1007/978-3-642-24418-6_16
- Rincón, L., Mazo, R., & Salinesi, C. APPLIES: A framework for evaluating organization's motivation and preparation for adopting product lines, 12th International Conference on Research Challenges in Information Science (RCIS), Nantes, 2018, pp. 1-12.
- Rossi, B., Russo, B., & Succi, G. (2012). Adoption of free/libre open source software in public organizations: Factors of impact. *Information Technology and People*, 25(2), 156–187. <https://doi.org/10.1108/09593841211232677>
- Roumani, Y., Nwankpa, J. K., & Roumani, Y. F. (2017). Adopters' trust in enterprise open source vendors: An empirical examination. *Journal of Systems and Software*, 125, 256–270.
- Torossi, G. (18 de marzo de 2018). Diseño de Sistemas. Obtenido de Instituto Tecnológico de Morelia: <http://dsc.itmorelia.edu.mx/~jcolivares/courses/pm10a/rup.pdf>
- Saghafi, F., Noorzad Moghaddam, E., & Aslani, A. (2017). Examining effective factors in initial acceptance of high-tech localized technologies: Xamin, Iranian localized operating system. *Technological Forecasting and Social Change*, 122, 275–288. <https://doi.org/10.1016/j.techfore.2016.04.010>
- Shaikh, M., & Cornford, T. (2011). IFIP AICT 365 - Framing the {itshape Conundrum} of Total Cost of Ownership of Open Source Software, 208–219. Retrieved from <http://personal.lse.ac.uk/shaikh/%0Ahttp://personal.lse.ac.uk/cornford>
- Silic, M., & Back, A. (2017). Open Source Software Adoption: Lessons from Linux in Munich. *IT Professional*, 19(1), 42–47. <https://doi.org/10.1109/MITP.2017.7>
- Sowe, S. K., & Mcnaughton, M. (n.d.). Using Multiple Case Studies to Analyse Open Source Software Business Sustainability in Sub-Saharan Africa, 160–177.
- Stallman, R. (18 de noviembre 2016). FLOSS and FOSS. GNU Operating System. Recuperado de: <https://www.gnu.org/philosophy/floss-and-foss.html>
- Steinmacher, I., Silva, M. A. G., & Gerosa, M. A. (2014). Barriers Faced by Newcomers to Open Source Projects: A Systematic Review, 153–163. https://doi.org/10.1007/978-3-642-55128-4_21
- Stol, K. J., & Fitzgerald, B. (2015). Inner Source - Adopting Open Source Development Practices in Organizations: A Tutorial. *IEEE Software*, 32(4), 60–67. <https://doi.org/10.1109/MS.2014.77>
- Stol, K.-J., Ali Babar, M., & Avgeriou, P. (2011). The Importance of Architectural Knowledge in Integrating Open Source Software, 142–158. https://doi.org/10.1007/978-3-642-24418-6_10

Ven, K., & Verelst, J. (2010). A field study on the barriers in the assimilation of open source server software. *IFIP Advances in Information and Communication Technology*, 319 AICT, 281–293. https://doi.org/10.1007/978-3-642-13244-5_2

ANEXOS

Anexo 1: Proyección de fases y periodos de Desarrollo

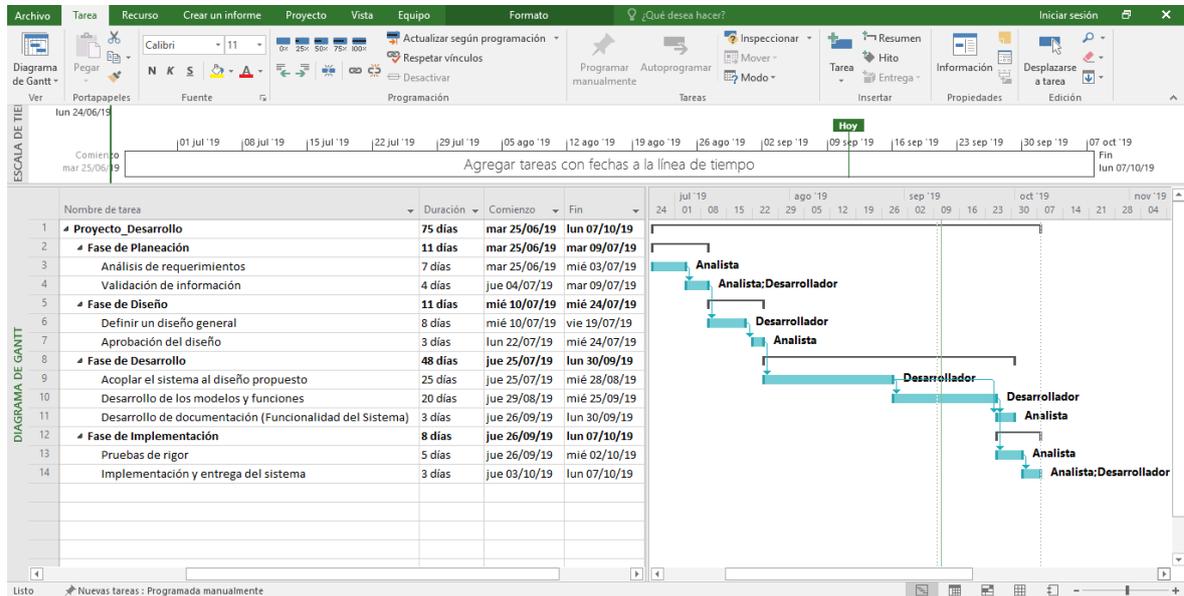


Ilustración 1: Diagrama de Gantt, diseñado en Project, mostrando fases y periodos de desarrollo con estimación de tiempos de inicio y finalización de tareas, incluyendo recursos, como lo son: el desarrollador y el analista.

Anexo 2: Caso de Uso

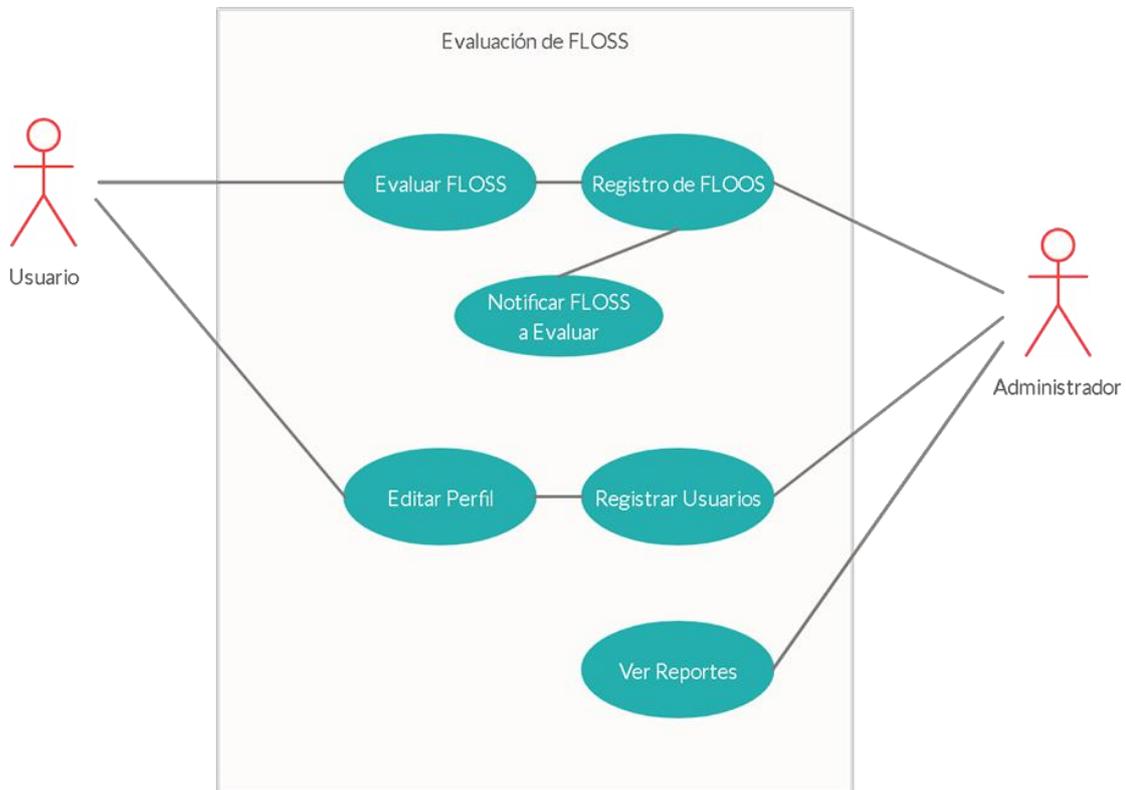


Ilustración 2: Caso de uso que identifica a los actores que intervienen en el sistema, representados como usuario y administrador, mostrando las acciones que cada uno puede ejercer dentro del sistema.

Anexo 3: Conexión a base de datos PostgreSQL

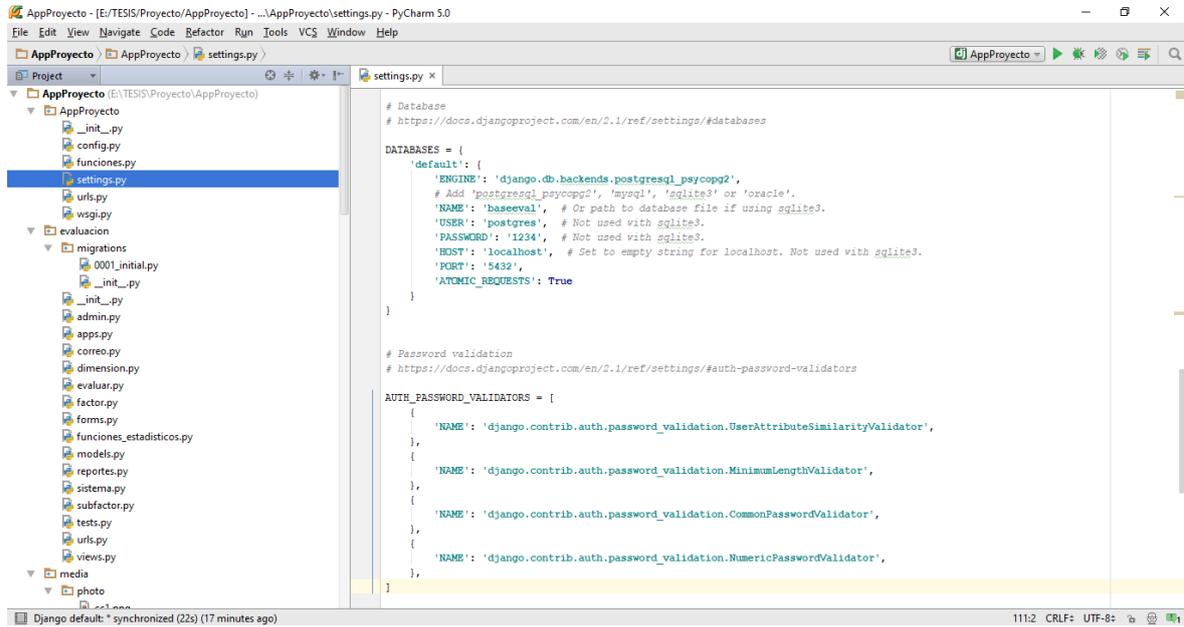


Ilustración 3: Pantalla de codificación en PyCharm que muestra el bloque de conexión entre la base de datos de PostgreSQL y Django.

Anexo 4: Matriz para determinar la importancia relativa

Valor de importancia personalizado	Valor de importancia sugerido		
	1	2	3
3	Importante	Importante	Muy Importante
2	Conveniente	Importante	Importante
1	Conveniente	Conveniente	Importante
0	No aplica	Conveniente	Conveniente

Ilustración 4: Tabla representada en forma de matriz para determinar la importancia relativa, combinando la importancia sugerida y la importancia personalizada.

Anexo 5: Funcionalidad del sistema

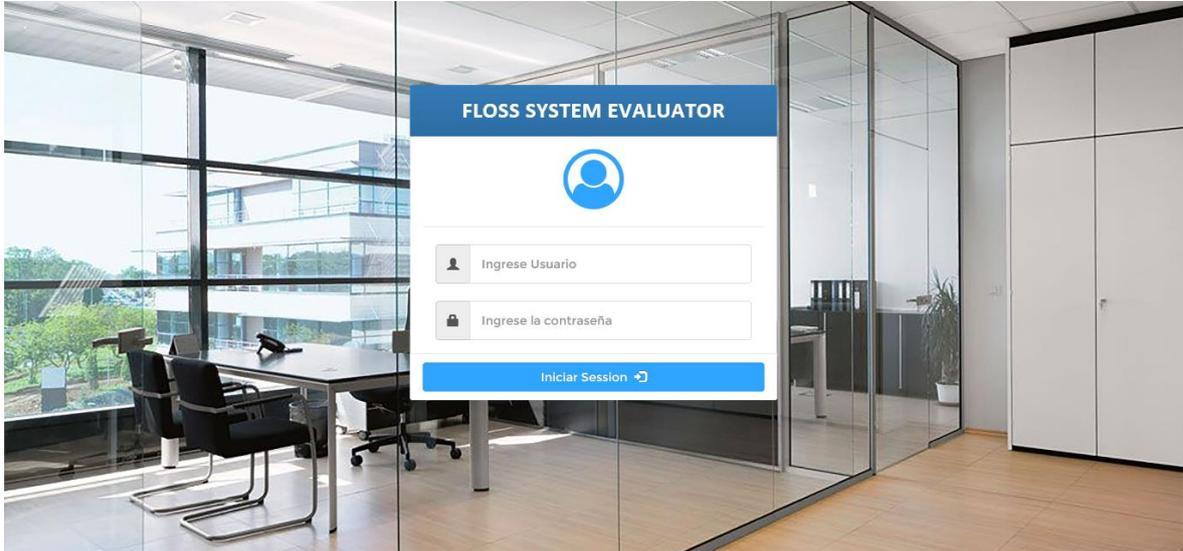


Ilustración 5: Pantalla principal que representa el inicio de sesión del sistema.

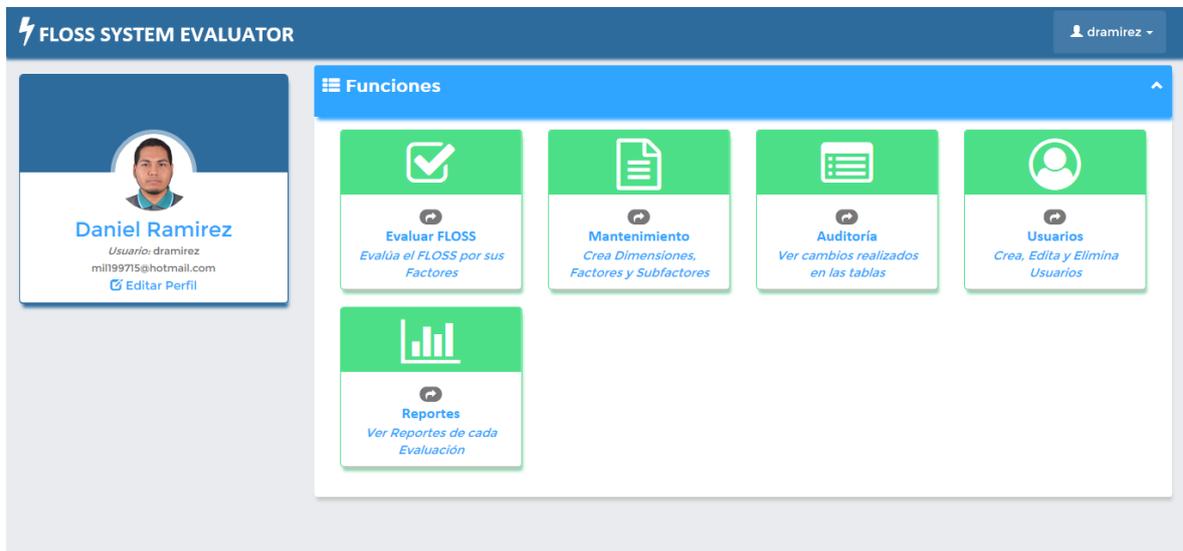


Ilustración 6: Menú principal del sistema el cual permite el acceso a los diferentes módulos pertinentes, dependiendo el rol del usuario, ya sea este administrador o encuestado. Visualización en modo usuario con permisos de administrador.

The screenshot shows the 'Evaluación' module interface. At the top, there is a header with the logo and the text 'FLOSS SYSTEM EVALUATOR' and a user profile 'dramirez'. Below the header, there is a breadcrumb trail 'Inicio / Sistema'. The main content area features a search bar with the placeholder text 'Ponga una descripción válida..' and a magnifying glass icon. Below the search bar is a table with the following data:

Sistema	Valor Sistema	Estado	Acción
LINUX CENTOS SERVER	0.55	Pendiente	Ver Dimensiones
SOFTWARE INTELIGENTE	0.0	Pendiente	Ver Dimensiones

At the bottom of the table area, there is a small blue square button with the number '1'.

Ilustración 7: Módulo de “Evaluación”, dentro es posible visualizar las evaluaciones de los diferentes sistemas, junto con la puntuación que se le ha dado con respecto a los criterios de evaluación y su estado, el cual dependerá de las dimensiones, si estas ya se encuentran evaluadas o no. Se podrá acceder a dicho formulario desde el botón “Ver Dimensiones”.

The screenshot shows the 'Dimensiones' module interface. At the top, there is a header with the logo and the text 'FLOSS SYSTEM EVALUATOR' and a user profile 'dramirez'. Below the header, there is a breadcrumb trail 'Inicio / Sistema / Dimension'. The main content area features a search bar with the placeholder text 'Ponga una descripción válida..' and a magnifying glass icon. Below the search bar, there is a sub-header 'SISTEMA: LINUX CENTOS SERVER'. Below this is a table with the following data:

Dimensión	Valor Dimensión	Estado	Acción
TECNOLÓGICA	0.76	Pendiente	Ver Factores
ORGANIZACIONAL	0.33	Pendiente	Ver Factores

At the bottom of the table area, there is a small blue square button with the number '1'.

Ilustración 8: La pantalla “Dimensiones” nos muestra el nombre de la dimensión que vamos a analizar junto con el valor de evaluación que indica el puntaje ponderado y su estado, al igual que en la pantalla principal del módulo evaluación este dependerá de los factores, si estos ya se encuentran evaluados o no. Se podrá acceder a dicho formulario desde el botón “Ver Factores”.

Factor	SubFactores	Valor Sugerido Factor	Asignar Valor Personalizado	Valor Personalizado Factor	Resultado de Importancia Factor	Valor Factor	Evaluar Subfactores
COMPATIBILIDAD	<ul style="list-style-type: none"> La compañía proporciona infraestructura en la nube para que la solución FLOSS sea adoptada. El software puede exportar formatos propietarios. El software interactúa con solución propietaria (integración fácil). El software está certificado para el funcionamiento (sector de la 	2	Editar Valor	2	Importante	3,1	Reevaluar

Ilustración 9: Al ingresar a la ventana “Factores” se podrá visualizar una tabla de contenidos, la cual muestra el nombre del factor, junto con una breve descripción de cada uno, el puntaje o valor de evaluación, (esto si en tal caso ya se evaluó previamente), como también los subfactores que forman parte de él. El valor de importancia sugerida, importancia personalizada y el resultado de importancia se encuentran en este formulario, como también el botón de acción de evaluar los subfactores y los valores de importancia.

Subfactor	Evaluar
El software tiene un buen registro en términos de errores de seguridad (el software es seguro).	Muy de Acuerdo
Software es más flexible / eficiente que la solución propietaria.	De Acuerdo
Software es más confiable que la solución propietaria.	En Desacuerdo
Software proporciona funciones de control de acceso.	De Acuerdo

[Guardar](#) [Cancelar](#)

Ilustración 10: Accediendo al botón de acción “Evaluar” se nos mostrará una tabla de contenidos, la cual presenta los subfactores que pertenecen a un factor en específico y se llenará mediante las opciones que se encuentran en el “ComboBox”, los cuales también son conocidos como criterios de evaluación y al momento de finalizar y guardar la evaluación en la tabla de “Factores” se mostrara el puntaje ponderado.

The screenshot shows the 'Consulta de Sistema' (System Query) module. At the top, there is a search bar with the placeholder text 'Ponga una descripción válida..' and a search icon. Below the search bar is a table with the following data:

Sistema	Fecha	Valor Sistema	Estado	Usuario	Acción
LINUX CENTOS SERVER	15 de Septiembre de 2019	0,98	Activo	admin	Acción
SOFTWARE INTELIGENTE	15 de Septiembre de 2019	1,25	Activo	admin	Acción

Below the table, there is a pagination control showing the number '1'.

Ilustración 11: Visualización del módulo “Mantenimiento” con información de un sistema que esta por evaluar, mostrando el nombre del sistema, fecha de creación, puntaje de evaluación, estado, el usuario que lo creo y las acciones que se pueden ejecutar en él.

The screenshot shows the 'Consulta de Sistema' (System Query) module with a form for creating a new system record. The form has two input fields: 'Nombre' (Name) and 'Estado' (Status). Below the form, there are two buttons: 'Guardar' (Save) and 'Cancelar' (Cancel).

Ilustración 12: Formulario de nuevo registro de sistema perteneciente al módulo de “Mantenimiento”.

FLOSS SYSTEM EVALUATOR dramirez

Inicio / Sistema

Consulta de Sistema

Nuevo Registro

Buscar Ponga una descripción válida..

Sistema	Fecha	Valor Sistema	Estado	Usuario	Acción
LINUX CENTOS SERVER	15 de Septiembre de 2019	0,98	Activo	admin	<ul style="list-style-type: none"> Ver Dimensión Editar Eliminar
SOFTWARE INTELIGENTE	15 de Septiembre de 2019	1,25	Activo	admin	

1

Ilustración 13: Dentro de las acciones que se pueden ejecutar en el módulo "Mantenimiento" tenemos las siguientes: crear nuevos registros de "Dimensiones", "Factores" y "Subfactores", como también hacer uso de las funciones de buscar, editar y eliminar. "Cabe recalcar que el único que puede manipular las evaluaciones es el administrador".

FLOSS SYSTEM EVALUATOR dramirez

Inicio / Sistema / Dimension

Consulta de Dimension

SISTEMA: LINUX CENTOS SERVER

Nuevo Registro

Buscar Ponga una descripción válida..

Dimensión	Valor Dimensión	Acción
TECNOLÓGICA	0,76	<ul style="list-style-type: none"> Ver Factores Editar Eliminar
ORGANIZACIONAL	1,21	

1

Ilustración 14: La presente imagen muestra el mantenimiento de dimensiones.

FLOSS SYSTEM EVALUATOR dramirez

[Nuevo Registro](#)

Buscar

Factor	Descripción	SubFactores	Valor Sugerido Factor	Valor Personalizado Factor	Resultado de Importancia Factor	Valor Factor	Acción
COMPATIBILIDAD	Compatibilidad de la solución a ser adoptada con respecto a los formatos de los datos.	<ul style="list-style-type: none"> La compañía proporciona infraestructura en la nube para que la solución FLOSS sea adoptada. El software puede exportar formatos propietarios. El software interactúa con solución propietaria (integración fácil). El software está certificado para el funcionamiento (sector de la salud). El software es confiable / seguro (probado y comúnmente adoptado / fácil de usar / usabilidad / se recupera bien de fallos). El software está bien documentado. El software es compatible con los... 	2	1	Conveniente	3,1	Acción Ver Subfactores Editar Eliminar

Ilustración 15: Formulario de mantenimiento de factores.

FLOSS SYSTEM EVALUATOR dramirez

[Consulta de Subfactores](#)

SISTEMA: LINUX CENTOS SERVER / DIMENSIÓN: TECNOLÓGICA / FACTOR: COMPATIBILIDAD

[Nuevo Registro](#)

Buscar

SubFactor	Acción
La compañía proporciona infraestructura en la nube para que la solución FLOSS sea adoptada.	Acción
El software puede exportar formatos propietarios.	Editar Eliminar
El software interactúa con solución propietaria (integración fácil).	Editar Eliminar
El software está certificado para el funcionamiento (sector de la salud).	Acción

1 2 3

Ilustración 16: Cada elemento posee sus funciones para crear nuevos registros, buscarlos, modificarlos y eliminarlos, esta ventana representa el mantenimiento de los Subfactores.



Ilustración 17: Modificación de la descripción de un Subfactor.

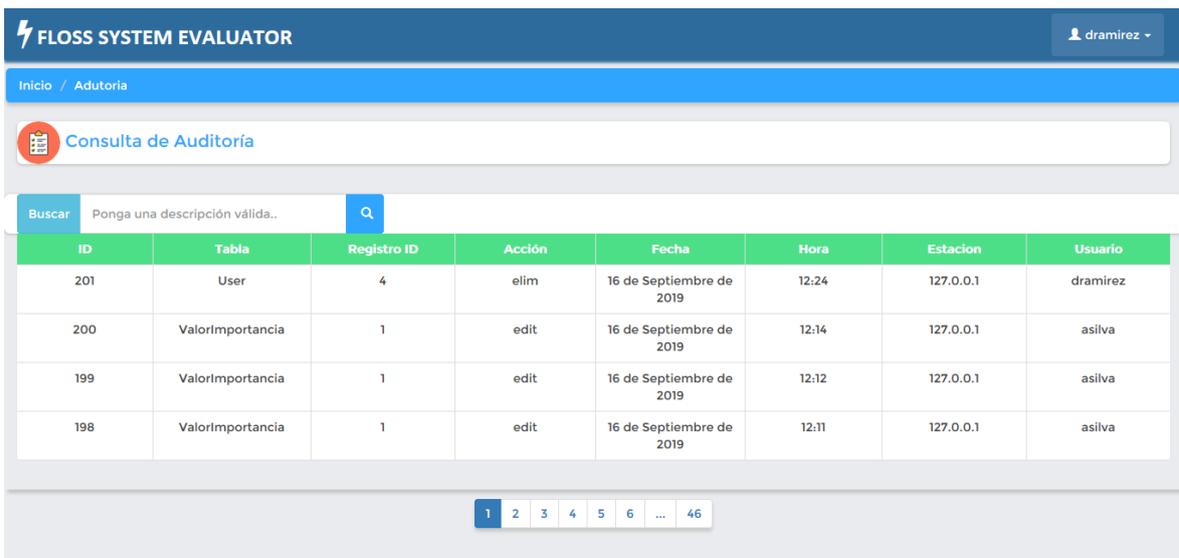


Ilustración 18: El modulo "Auditoría" nos presenta una tabla con registro de las acciones realizadas a lo largo del uso del sistema, indicando la tabla en la cual se ejerció algún cambio, el tipo de acción, fecha y hora, estación y nombre del usuario.

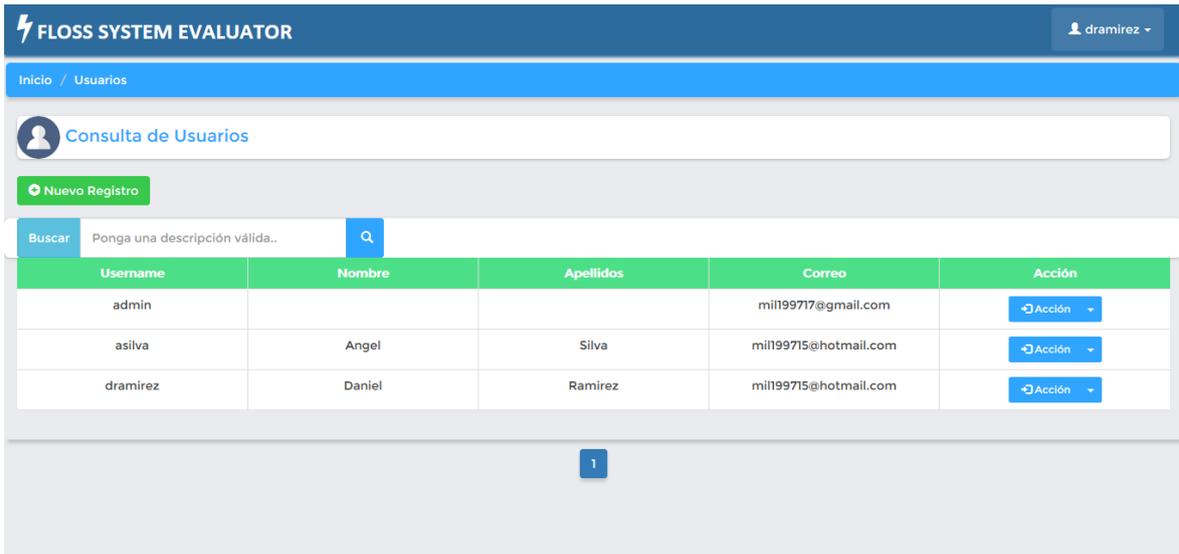


Ilustración 19: En el módulo “Usuarios” es posible registrar un nuevo usuario en el sistema asignándole la capacidad de visualizar y ser parte de la evaluación, como también se puede modificar su información, entre ellos su nombre de usuario, nombres y apellidos, correo, permisos de administrador y contraseña, como también buscar usuarios o eliminarlos.

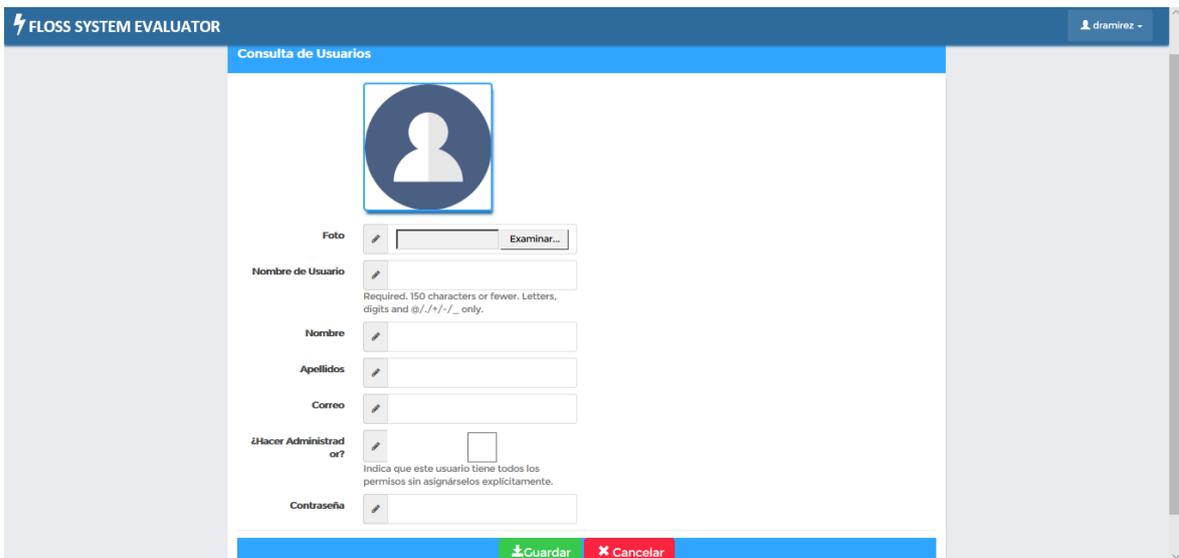


Ilustración 20: Formulario general para el registro de nuevos usuarios.

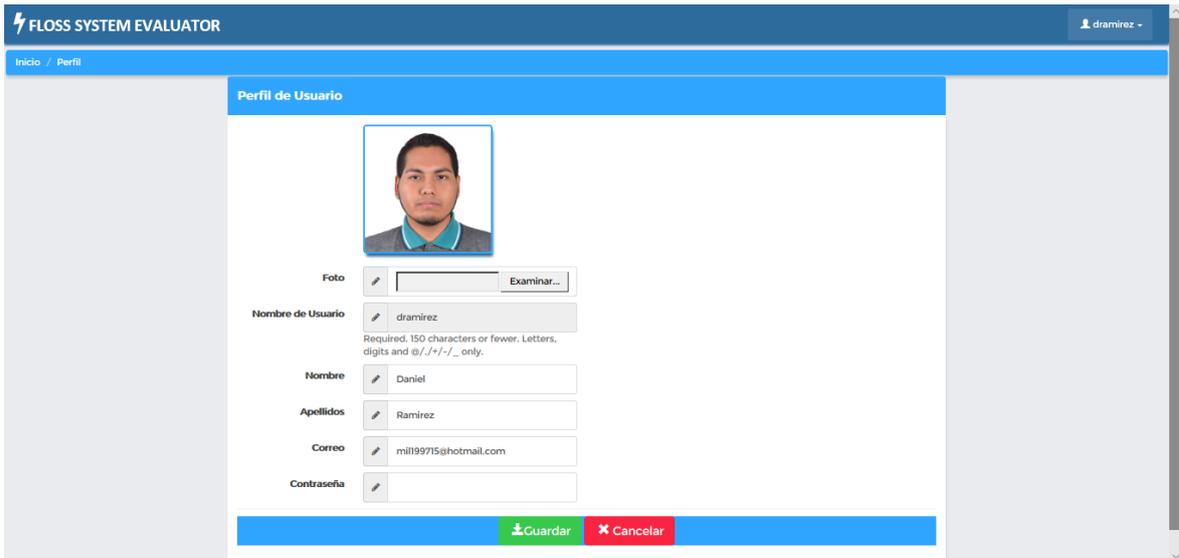


Ilustración 21: En la pantalla del menú principal, específicamente en la parte superior izquierda, justo donde se encuentra el nombre del usuario activo, se puede visualizar un acceso hacia la opción “editar perfil”, la cual nos dirigirá a la ventana que se presenta a continuación, en donde podremos modificar la información de un perfil existente, en este caso del usuario que se encuentra en sesión.

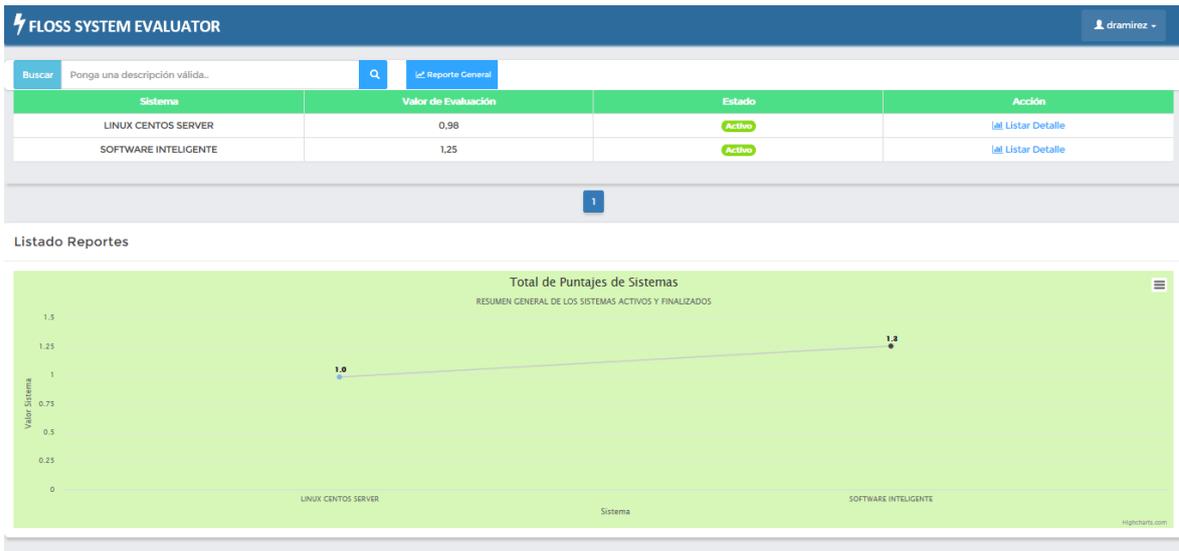


Ilustración 22: Al ingresar al módulo de “Reportes” se nos permitirá visualizar las estadísticas de los puntajes obtenidos de las evaluaciones de manera general y detallada de cada factor y Subfactor que integra cada sistema. En esta pantalla podemos apreciar una tabla que contiene el nombre del sistema, la puntuación obtenida, su estado y las acciones de búsqueda y listar detalles.

Listado Reportes

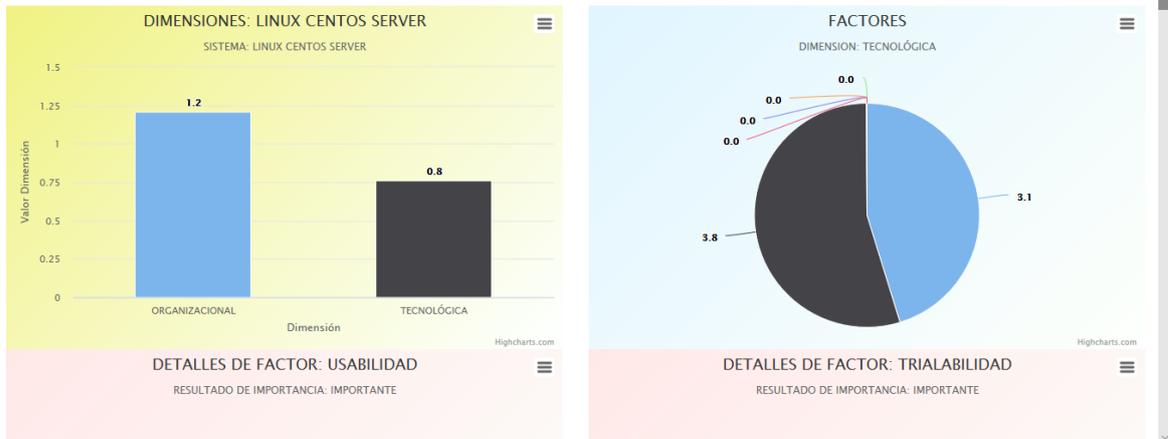


Ilustración 23: En la presente imagen podemos observar una pantalla que nos muestra gráficos estadísticos detallados de las evaluaciones divididas por Dimensiones, Factores y Subfactores que integran el sistema.

Anexo 6: Tabla de Dimensiones, Factores y Subfactores del marco de trabajo propuesto para la adopción de FLOSS

<i>Dimensión</i>	<i>Factor</i>	<i>Descripción</i>	<i>Subfactor</i>
<u><i>Tecnológica</i></u>	Compatibilidad (SI = 2)	Compatibilidad de la solución a ser adoptada con respecto a los formatos de los datos.	<p>La compañía proporciona infraestructura en la nube para que la solución FLOSS sea adoptada.</p> <p>El software puede exportar formatos propietarios.</p> <p>El software interactúa con solución propietaria (integración fácil).</p> <p>El software está certificado para el funcionamiento (sector de la salud).</p> <p>El software es confiable / seguro (probado y comúnmente adoptado / fácil de usar / usabilidad / se recupera bien de fallas).</p> <p>El software está bien documentado.</p> <p>El software es compatible con los casos de uso más comunes funcionalidades.</p> <p>El software es compatible con varios equipos de hardware.</p> <p>El software usa formatos estándar.</p> <p>El software es multiplataforma.</p>
	Personalización (SI = 2)	El grado de personalización permitido por el software a través de cambios en su configuración predeterminada.	<p>El acceso al código fuente es un incentivo para la organización.</p> <p>La funcionalidad para personalizar el software está disponible.</p> <p>El software se puede modificar fácilmente para satisfacer mis necesidades (extensibilidad).</p> <p>El software innova a un ritmo rápido.</p> <p>Software es ad-hoc (no genérico), el software necesita ser generalizado.</p> <p>Software es fácil de personalizar.</p> <p>El software admite nuevas funciones a través de módulos (el software es modular).</p> <p>El usuario puede crear extensiones fácilmente / disponibilidad de funcionalidad de crear extensiones.</p> <p>El usuario puede instalar extensiones muy fácilmente / disponibilidad de funcionalidad para instalar extensiones.</p> <p>La interfaz de usuario es personalizable.</p>

<i>Dimensión</i>	<i>Factor</i>	<i>Descripción</i>	<i>Subfactor</i>
	Confiabilidad (SI = 2)	Este factor se refiere a la forma en que se utiliza una solución que debe adoptar en las condiciones durante un periodo de tiempo determinado (Se refiere al comportamiento que demuestra el sistema, si es confiable).	El software tiene un buen registro en términos de errores de seguridad (el software es seguro). Software es más flexible / eficiente que la solución propietaria. Software es más confiable que la solución propietaria. Software proporciona funciones de control de acceso.
	Reutilización (SI = 2)	Cantidad de código que se puede reutilizar a través de bibliotecas y paquetes.	Formatos de datos están bien documentados. La licencia permite extensiones de propiedad (grado de reutilización). Los archivos de software se pueden reutilizar fácilmente. El software se ofrece como una biblioteca / framework.
	Usabilidad (SI = 2)	Cuan intuitiva y satisfactoria es la interfaz de usuario del software.	El software provee una interfaz gráfica de usuario (GUI). El software es más fácil de usar. El software es fácil de aprender. El software representa la información de una manera intuitiva y bien estructurada. El usuario está satisfecho con el software propietario. El usuario tiene altas / bajas expectativas en software.
	Documentación (SI = 1)	La cantidad y calidad de la documentación disponible de la solución que se adoptara.	La documentación del desarrollador cubre todas las características. La documentación está disponible en múltiples formatos / plataformas. La documentación es fácil de entender / comprensible. La documentación está actualizada. La documentación está escrita por escritores especializados (no desarrolladores). Calidad de la documentación. Una organización similar ha adoptado con éxito el software. El software viene con la documentación de desarrollo. El software viene con la documentación del usuario. La documentación del usuario cubre todas las características.

<i>Dimensión</i>	<i>Factor</i>	<i>Descripción</i>	<i>Subfactor</i>
	Mantenimiento (SI = 1)	Cantidad de recursos que deben dedicarse para mantener la solución FLOSS.	El software es mantenido activamente por los desarrolladores.
	Portabilidad (SI = 1)	La posibilidad de implementar una solución en más de una plataforma, como diferentes sistemas operativos.	El cliente de la aplicación móvil está disponible (app). El software es DBMS independiente. El cliente web está disponible.
	Trialabilidad (SI = 1)	La facilidad para probar la solución para adoptar.	El software es fácil de probar de forma gratuita. El usuario no está obligado a instalar actualizaciones.
<u>Organizacional</u>	Soporte (SI = 3)	La disponibilidad de soporte técnico interno y externo.	Soporte comercial disponible (expertos consultores de TI). El soporte de la comunidad está disponible. Disponibilidad de personal experto externo de apoyo para consultas específicas. El soporte para soluciones FLOSS está disponible en todo momento (24/7). FLOSS ofrece soporte a través de diferentes canales: blogs, Google-Groups, call center, email, chat, wiki, comunidades expertas, etc. FLOSS ofrece soporte de calidad a través de personal experto de TI generando satisfacción y seguridad en los usuarios. La organización cuenta con desarrolladores que pueden personalizar el software (saben cómo desarrollar código fuente / tecnología). Personal experto en diferentes plataformas: web services, soluciones de escritorio, así como en diferentes dominios. El soporte para personalizaciones está disponible.

<i>Dimensión</i>	<i>Factor</i>	<i>Descripción</i>	<i>Subfactor</i>
	Tiempo de adopción (SI = 3)	El tiempo que requiere una organización para implementar la solución FLOSS.	Recursos necesarios para el despliegue de FLOSS. Tiempo requerido para adoptar soluciones FLOSS. Tiempo requerido para desplegar soluciones FLOSS.
	Formación (SI = 3)	Acciones para mejorar las habilidades y el conocimiento de los usuarios en la organización.	Adoptar soluciones FLOSS permite a los usuarios adquirir o mejorar las habilidades técnicas de TI. Entrenamiento autónomo en herramientas FLOSS para aplicarlo en la institución. Personal capacitado para resolver problemas tecnológicos. Planificación de capacitaciones para personal de la institución.
	Casos de estudio de la adopción de FLOSS (SI = 2)	Historias exitosas existentes de la adopción de FLOSS por organizaciones similares.	Información disponible acerca de casos de éxitos de adopción de FLOSS. Uso de redes de cooperación para compartir casos de éxito de FLOSS nacional e internacionalmente.
	Centralidad de TI (SI = 2)	El grado de dependencia de la organización en su propia infraestructura de TI.	Cambio de la infraestructura TI crea un buen ambiente de trabajo entre los usuarios. Centralización de la infraestructura TI mejora el proceso de migración de sistemas privativos a FLOSS. Dependencia del IT manager sobre la decisión de cambio de la infraestructura TI.
	Soporte de alta dirección (SI = 2)	Grado de respaldo de la gerencia a la decisión de adoptar la solución FLOSS.	El apoyo de alto nivel de la dirección es esencial para el éxito de adopción de soluciones FLOSS en una organización. Liderazgo en los administradores de TI ayudan a tomar correctas decisiones de adopción de FLOSS.

<i>Dimensión</i>	<i>Factor</i>	<i>Descripción</i>	<i>Subfactor</i>
	Bloqueos de proveedores (SI = 3)	Dependencias en una solución de software específica de manera que sea costoso cambiar a una alternativa.	Menos dependencia de proveedores con soluciones FLOSS versus soluciones privativas. Reducir el riesgo de dependencia con proveedores.
	Actitud hacia el cambio (SI = 1)	Comportamiento de los empleados relacionado con los cambios tecnológicos que se adoptaran.	Cambio de tecnología sin previo aviso a usuarios finales. Resistencia al cambio de tecnología por parte de los usuarios. Usuarios motivados por aprender nuevas herramientas.
<i><u>Económica</u></i>	Costo de la licencia (SI = 3)	El costo de adquirir una licencia de la solución de software.	Ahorro de coste de licencias FLOSS. Regulación de costes de licencias y actualizaciones. Uso libre y distribución de licencias FLOSS.
	TCO (SI = 2)	Los costos de licencia, operación y soporte.	Bajo coste en licencias FLOSS. Costes ocultos de migración y mantenimiento de FLOSS. Uso libre de soluciones FLOSS.
	Costo de soporte (SI = 1)	Soporte externo y acceso a actualizaciones.	Bajo coste de soporte en soluciones FLOSS.
	Costo operacional (SI = 1)	El costo de mantenimiento, desarrollo, despliegue y migración.	Bajo coste de desarrollo de soluciones FLOSS. Bajo coste de mantenimiento de soluciones FLOSS.

Ilustración 24: Tabla de contenido con dimensiones, factores y subfactores empleados dentro del marco de trabajo propuesto para la adopción de Free Libre Open Source Software (FLOSS).

Anexo 7: Diagrama de Base de Datos del Sistema

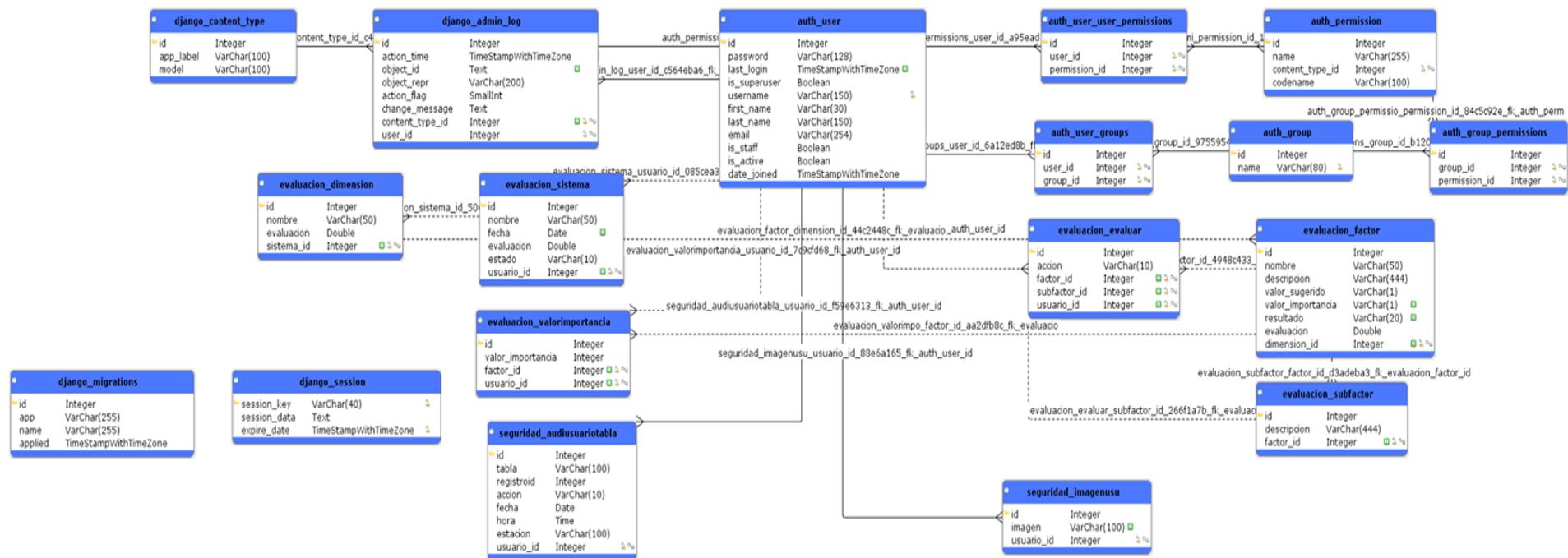
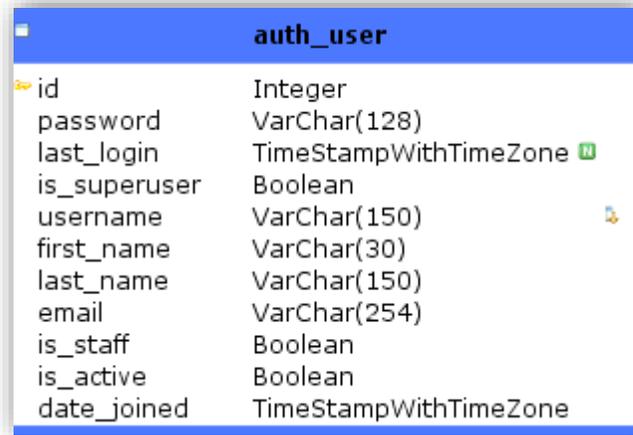


Ilustración 25: Esquema de la base de datos del sistema elaborada en DBWrench (Database Design & Synchronization Software).

TABLAS FUERTES

AUTH_USER



auth_user	
id	Integer
password	VarChar(128)
last_login	TimeStampWithTimeZone
is_superuser	Boolean
username	VarChar(150)
first_name	VarChar(30)
last_name	VarChar(150)
email	VarChar(254)
is_staff	Boolean
is_active	Boolean
date_joined	TimeStampWithTimeZone

Ilustración 26: Tabla AUTH_USER, lugar en donde se almacena los datos del usuario.

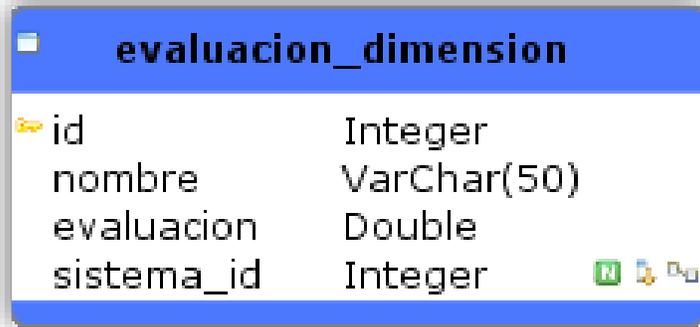
EVALUACION_SISTEMA



evaluacion_sistema	
id	Integer
nombre	VarChar(50)
fecha	Date
evaluacion	Double
estado	VarChar(10)
usuario_id	Integer

Ilustración 27: Tabla EVALUACION_SISTEMA, lugar en donde se almacena los datos del sistema.

EVALUACION_DIMENSION

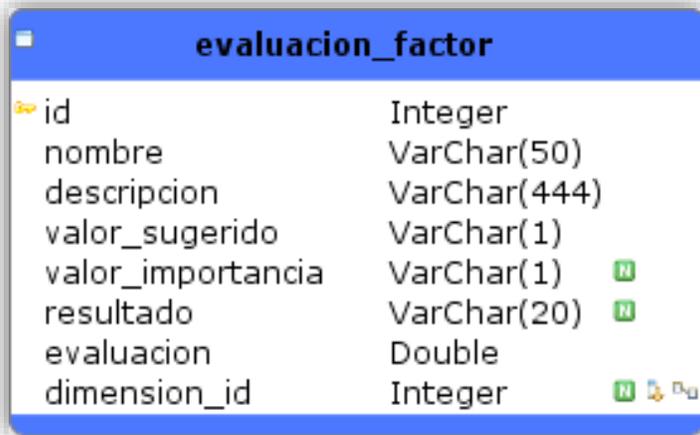


The screenshot shows a table definition window for 'evaluacion_dimension'. The table has four columns: 'id' (Integer, primary key), 'nombre' (VarChar(50)), 'evaluacion' (Double), and 'sistema_id' (Integer). There are icons for 'N', a key, and a refresh symbol at the bottom right.

evaluacion_dimension	
id	Integer
nombre	VarChar(50)
evaluacion	Double
sistema_id	Integer

Ilustración 28: Tabla EVALUACION_DIMENSION, lugar en donde se almacena la dimensión del sistema.

EVALUACION_FACTOR

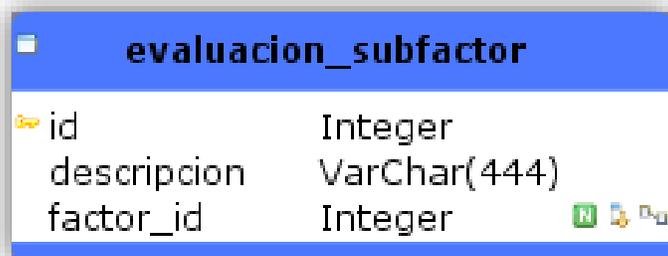


The screenshot shows a table definition window for 'evaluacion_factor'. The table has eight columns: 'id' (Integer, primary key), 'nombre' (VarChar(50)), 'descripcion' (VarChar(444)), 'valor_sugerido' (VarChar(1)), 'valor_importancia' (VarChar(1)), 'resultado' (VarChar(20)), 'evaluacion' (Double), and 'dimension_id' (Integer). There are 'N' icons next to 'valor_importancia', 'resultado', and 'dimension_id', and a key icon next to 'id'. There are also icons for 'N', a key, and a refresh symbol at the bottom right.

evaluacion_factor	
id	Integer
nombre	VarChar(50)
descripcion	VarChar(444)
valor_sugerido	VarChar(1)
valor_importancia	VarChar(1)
resultado	VarChar(20)
evaluacion	Double
dimension_id	Integer

Ilustración 29: Tabla EVALUACION_FACTOR, lugar en donde se almacena los factores correspondientes a la dimensión del sistema.

EVALUACION_SUBFACTOR

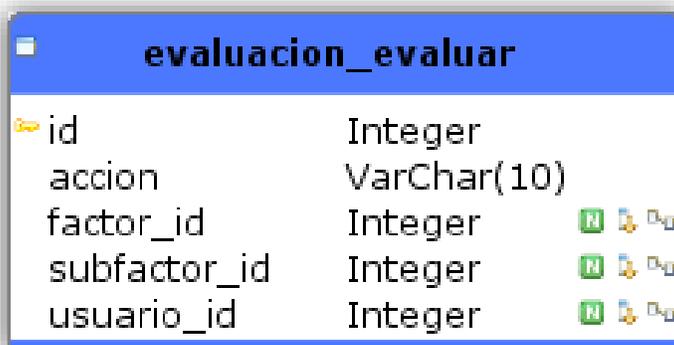


evaluacion_subfactor		
id	Integer	
descripcion	VarChar(444)	
factor_id	Integer	N

Ilustración 30: Tabla EVALUACION_SUBFACTOR, lugar en donde se almacena los subfactores correspondientes a los factores y respectivamente a la dimensión del sistema.

TABLAS TRANSACCIONALES

EVALUACION_EVALUAR



evaluacion_evaluar		
id	Integer	
accion	VarChar(10)	
factor_id	Integer	N
subfactor_id	Integer	N
usuario_id	Integer	N

Ilustración 31: EVALUACION_EVALUAR, tabla transaccional que permite evaluar los subfactores mediante una caja de selección y dar una puntuación a las tablas predecesoras relacionadas: factores, dimensión y sistema.

EVALUACION_VALORIMPORTANCIA



The image shows a screenshot of a database table definition for 'evaluacion_valorimportancia'. The table has four columns: 'id', 'valor_importancia', 'factor_id', and 'usuario_id'. The 'id' column is the primary key. The 'factor_id' and 'usuario_id' columns have foreign key relationships with other tables, indicated by icons.

evaluacion_valorimportancia	
id	Integer
valor_importancia	Integer
factor_id	Integer   
usuario_id	Integer   

Ilustración 32: EVALUACION_VALORIMPORTANCIA, tabla transaccional que evalúa opcionalmente en un rango de 0 a 3 al factor con el fin de medir su valor de importancia.



REGISTRO DE ACOMPAÑAMIENTOS

Inicio: 31-05-2019 Fin 15-02-2020

FACULTAD CIENCIAS E INGENIERÍA

CARRERA: INGENIERÍA EN SISTEMAS COMPUTACIONALES 2019

Línea de investigación: 1S2019 UIC TECNOLOGÍAS DE LA INFORMACIÓN Y DE LA COMUNICACIÓN

TEMA: None

ACOMPAÑANTE: REA SANCHEZ VICTOR HUGO

DATOS DEL ESTUDIANTE			
Nº	APELLIDOS Y NOMBRES	CÉDULA	CARRERA
1	RAMIREZ ARREAGA DANIEL BRYAN	0940362668	INGENIERÍA EN SISTEMAS COMPUTACIONALES 2019
2	SILVA ACOSTA ANGEL ISMAEL	0957983257	INGENIERÍA EN SISTEMAS COMPUTACIONALES 2019

Nº	FECHA	HORA		Nº HORAS	DETALLE
1	04-07-2019	Inicio: 11:00 a.m.	Fin: 12:00 p.m.	1	REVISIÓN CAPÍTULO 1: OBJETIVOS Y JUSTIFICACIÓN. SE REALIZARON OBSERVACIONES A SER PRESENTADAS EN LA SIGUIENTE SESIÓN
2	23-07-2019	Inicio: 11:00 a.m.	Fin: 12:00 p.m.	1	REVISIÓN DEL CAPÍTULO 1: MARCO TEÓRICO. SE REALIZARON OBSERVACIONES Y CORRECCIONES EN EL DOCUMENTO.
3	01-08-2019	Inicio: 11:00 a.m.	Fin: 12:00 p.m.	1	REVISIÓN DEL CAPÍTULO 1: REVISIÓN FINAL DEL CAPÍTULO 1 Y REVISIÓN DEL AVANCE DE DESARROLLO DEL SISTEMA
4	15-08-2019	Inicio: 12:00 p.m.	Fin: 13:00 p.m.	1	REVISIÓN DEL CAPÍTULO 2: METODOLOGÍA. SE CORRIGIERON IDEAS PRINCIPALES PARA LA CONSTRUCCIÓN CORRECTA DEL CAPÍTULO.
5	27-08-2019	Inicio: 12:00 p.m.	Fin: 13:00 p.m.	1	REVISIÓN DEL CAPÍTULO 2: METODOLOGÍA (CAMBIOS EN EL FORMATO Y CORRECCIONES) REVISIÓN DEL AVANCE DE MODIFICACIONES DEL SISTEMA
6	25-06-2019	Inicio: 11:00 a.m.	Fin: 12:00 p.m.	1	REVISIÓN DEL CAPÍTULO 1: INTRODUCCIÓN Y PLANTAMIENTO DEL PROBLEMA (PRIMER BORRADOR)
7	18-07-2019	Inicio: 11:00 a.m.	Fin: 12:00 p.m.	1	REVISIÓN DEL CAPÍTULO 1: AVANCE DE DISEÑO DEL SISTEMA
8	03-09-2019	Inicio: 12:00 p.m.	Fin: 13:00 p.m.	1	REVISIÓN FINAL DEL CAPÍTULO 2

REA SANCHEZ VICTOR HUGO
PROFESOR(A)

REA SANCHEZ VICTOR HUGO
DIRECTOR(A)

RAMIREZ ARREAGA DANIEL BRYAN
ESTUDIANTE

SILVA ACOSTA ANGEL ISMAEL
ESTUDIANTE

Dirección: Cda. Universitaria Km. 1 1/2 vía km. 26
 Consultador: (04) 2715081 - 2715079 Ext. 3107
 Telefax: (04) 2715187
 Milagro • Guayas • Ecuador

VISIÓN
 Ser una universidad de docencia e investigación

MISIÓN
 La UNEMI forma profesionales competentes con actitud proactiva y valores éticos, asumiendo investigación relevante y oferta servicios que demanda el sector externo contribuyendo al desarrollo de la sociedad.