FACULTAD 2

Sistema para automatizar la elaboración de documentos de consultorías a centros de datos que realiza el Departamento de Componentes del Centro de Telemática

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas

Autora: Brenda García Pérez

Tutores: Ing. Yenlys Guerra Dávila

Ing. Michael Brug Santrayll

La Habana, junio de 2018

"La posibilidad de realizar un sueño es lo que hace que la vida sea tan interesante."

Paulo Coelho.

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año ___.

_____

Brenda García Pérez

(Autora)

_____ _____

Ing. Yenlys Guerra Dávila Ing. Michael Brug Santrayll

(Tutora) (Tutor)

DATOS DE CONTACTO

Autora:

Brenda García Pérez

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: bgperez@estudiantes.uci.cu

Tutora:

Ing. Yenlys Guerra Dávila

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: yguerra@uci.cu

Tutor:

Ing. Michael Brug Santrayll

Universidad de las Ciencias Informáticas, La Habana, Cuba.

Email: mbrug@uci.cu

AGRADECIMIENTOS

Quisiera agradecerles:

En especial a mis padres, gracias a ellos hoy estoy en esta universidad graduándome como ingeniera en ciencias informáticas. Todo se lo debo a ellos. Los quiero mucho.

A mi hermana mayor, padrino y sobrinos que con amor y cariño me han apoyado en el transcurso de mi superación, sobre todo a mi hermana, que siempre me aconseja, me ayuda y me da ánimos y fuerzas para seguir avanzando, ella es mi ejemplo a seguir.

A mi pareja Reybel, que en todo momento estuvo junto a mi apoyándome y dándome ánimos para seguir adelante. Gracias por tu amor, confianza y ayuda todo este tiempo. Te quiero.

A la familia de mi pareja, que siempre estuvieron pendientes de mis resultados y me han dado su apoyo.

A mis compañeros de grupo, que estuvieron dispuestos a ayudarme desinteresadamente.

A mis compañeras de apartamento, por compartir todos estos años de universidad.

Desearía hacer un agradecimiento especial a mis tutores, los cuales me ayudaron mucho en el trascurso de mi tesis.

Y a todas aquellas personas que he conocido en esta universidad,

A todos, muchas gracias.

DEDICATORIA

A toda mi familia, en especial a mis padres y mi hermana, que sin ellos no hubiera podido ser nadie en esta vida, y les dedico esta tesis, con el objetivo de que se sientan orgullosos de mí.

A quienes en el transcurso de mi vida y formación como estudiante aportaron con modestia, amor y sinceridad todo aquello que hoy me ha permitido ser lo que soy.

RESUMEN

La consultoría es una de las asistencias más solicitadas por las empresas, debido a que proporcionan un servicio de asesoramiento profesional que ayuda a los empresarios y a las entidades a alcanzar sus objetivos, detectando posibles problemas o necesidades que se deben cubrir u optimizar, para elaborar y proponer una estrategia con el objetivo de analizar y diagnosticar aspectos mejorables en la empresa. En la Universidad de las Ciencias Informáticas se encuentra el Centro Telemática, el cual cuenta con un departamento con especialistas destinados a realizar servicios de consultorías a instituciones, con la finalidad de diagnosticar el estado actual de los centros de datos. Actualmente el proceso de elaboración y levantamiento de información de los documentos de las consultorías presenta problemas debido a que se realiza de forma manual, afectando así la calidad del proceso y que existan duplicados y pérdida de información. En la presente investigación se desarrolla un sistema que permite automatizar la elaboración de los documentos de consultorías a centros de datos que brinda el Departamento de Componentes del Centro de Telemática. Para el desarrollo del sistema se utilizó como metodología de desarrollo de software XP, como marco de trabajo el framework Xilema-Base-Web, lenguajes de programación como Python, HTML y JavaScript, además de la librería ReportLab para el trabajo de los informes en formato PDF, y como sistema gestor de base de datos PostgreSQL. Se obtuvo como resultado final la implantación del sistema de forma satisfactoria y las mejoras en el proceso de elaboración de los informes de consultorías a centros de datos.

Palabras claves: centros de datos, consultorías, sistema.

ABSTRACT

Consultancy is one of the most requested assistance by companies, because they provide a professional advice service that helps entrepreneurs and entities to achieve their objectives, detecting possible problems or needs that must be covered or optimized, to develop and propose a strategy with the objective of analyzing and diagnosing aspects that can be improved in the company. The Telematics Center is located at the University of Computer Sciences, which has a department with specialists who are dedicated to consulting services to institutions, in order to diagnose the current state of the data centers. Currently, the process of preparing and gathering information from consulting documents presents problems because it is done manually, thus affecting the quality of the process and duplication and loss of information. In the present investigation a system is developed that allows to automate the elaboration of consulting documents to data centers provided by the Components Department of the Telematics Center. For the development of the system, the Xilema-Base-Web framework, programming languages such as Python, HTML and JavaScript, as well as the ReportLab library for the work of the reports in format were used as a framework for the development of XP software. PDF, and as a PostgreSQL database management system. The final result was the implementation of the system in a satisfactory manner and the improvements in the process of preparing consultancy reports to data centers.

Keywords: data centers, consultancies, system.

Índice

INTRODUCCIÓN 1

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA 5

1.1 CONCEPTOS ASOCIADOS 5

1.2 NORMA POR LA CUAL SE RIGE EL DEPARTAMENTO DE COMPONENTES DEL CENTRO DE TELEMÁTICA PARA LA INFRAESTRUCTURA DE LOS CENTROS DE DATOS 6

1.3 SISTEMAS DE SERVICIOS DE CONSULTORÍAS. ESTADO DEL ARTE 10

1.3.1 Sistema de registro y control de actividades en servicios de consultoría para la empresa MASTERTICS S.A.S 10

1.3.2 Sunbird 11

1.3.3 Schneider Electric 11

1.3.4 CIGET 11

1.3.5 BIOMUNDI 12

1.4 CONCLUSIONES DEL ESTUDIO 12

1.5 METODOLOGÍAS DE DESARROLLO DE SOFTWARE 13

1.5.1 Fundamentación de la metodología de desarrollo seleccionada 13

1.6 TECNOLOGÍAS Y HERRAMIENTAS DE DESARROLLO 14

1.6.1 Marcos de trabajo 14

1.6.2 Lenguaje de programación 14

1.6.3 Bibliotecas utilizadas 15

1.6.4 Herramientas de desarrollo 16

1.6.5 Herramienta de modelado 17

1.6.6 Sistema gestor de base de datos 17

1.7 CONCLUSIONES DEL CAPÍTULO 17

CAPÍTULO 2: PROPUESTA DE SOLUCIÓN 18

2.1 ANÁLISIS DEL PROCESO DE SERVICIO DE CONSULTORÍAS A CENTROS DE DATOS QUE REALIZA EL DEPARTAMENTO DE COMPONENTES DEL CENTRO DE TELEMÁTICA 18

2.2 CARACTERÍSTICAS DE LA PROPUESTA DEL SISTEMA 18

2.3 CARACTERÍSTICAS NO FUNCIONALES DEL SISTEMA 21

2.4 CONCLUSIONES DEL CAPÍTULO 22

CAPÍTULO 3: EXPLORACIÓN, PLANIFICACIÓN Y DISEÑO 24

3.1 EXPLORACIÓN 24

3.1.1 Historias de usuario 24
3.2 PLANIFICACIÓN 27
3.2.1 Estimación de esfuerzo por Historias de Usuario 27
3.2.2 Plan de iteraciones 28
3.2.3 Plan de entregas 30
3.3 DISEÑO 31
3.4 ARQUITECTURA DE SOFTWARE 31
3.4.1 Arquitectura cliente- servidor 31
3.4.2 Patrón arquitectónico 31
3.4.3 Patrones de diseño utilizados en el desarrollo del sistema propuesto 32
3.4.4 Modelo de Datos 35
3.4.5 Tarjetas Clase- Responsabilidad- Colaboración (CRC) 36
3.5 CONCLUSIONES DEL CAPÍTULO 37
CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS 38
4.1 ESTÁNDARES DE CODIFICACIÓN 38
4.2 DESARROLLO POR ITERACIONES 39
4.2.1 Iteración 1 39
4.2.2 Iteración 2 42
4.2.3 Iteración 3 46
4.3 PRUEBAS 49
4.3.1 Pruebas Unitarias 49
4.3.2 Pruebas de Aceptación 51
4.4 CONCLUSIONES DEL CAPÍTULO 60
CONCLUSIONES 61
RECOMENDACIONES 62
REFERENCIAS BIBLIOGRÁFICAS 63
BIBLIOGRAFÍA 66

Índice de Figuras

Figura 1: Centro de datos que cumple con la TIA-942. 7
Figura 2: Propuesta de sistema. 21
Figura 3: Arquitectura Cliente- Servidor. 31
Figura 4: Funcionamiento de MTV en Django. 32
Figura 5: Diagrama Entidad Relación. 36
Figura 6: Resultados de las pruebas unitarias de la primera iteración. 50
Figura 7: Resultados de las pruebas unitarias de la segunda iteración. 50
Figura 8: Resultados de las pruebas unitarias de la tercera iteración. 51
Figura 9: Comportamiento de las No Conformidades por iteraciones. 60

Índice de Tablas

Tabla 1: HU - Gestionar nombre de institución. 26
Tabla 2: HU - Gestionar locales tecnológicos. 26
Tabla 3: Estimación del esfuerzo por HU. 28
Tabla 4: Plan de duración de las iteraciones. 29
Tabla 5: Plan de entrega de las iteraciones. 30
Tabla 6: Tarjeta CRC # 1. 37
Tabla 7: Tarjeta CRC # 2. 37
Tabla 8: Tarea de Ingeniería - Listar institución 39
Tabla 9: Tarea de Ingeniería - Añadir institución. 39
Tabla 10: Tarea de Ingeniería - Modificar institución. 40
Tabla 11: Tarea de Ingeniería - Eliminar institución. 40
Tabla 12: Tarea de Ingeniería - Listar locales tecnológicos. 41
Tabla 13: Tarea de Ingeniería - Añadir locales tecnológicos. 41
Tabla 14: Tarea de Ingeniería - Modificar locales tecnológicos. 41
Tabla 15: Tarea de Ingeniería - Eliminar locales tecnológicos. 42
Tabla 16: Tarea de Ingeniería - Listar sistema de detección y supresión de incendios. 42
Tabla 17: Tarea de Ingeniería - Añadir sistema de detección y supresión de incendios. 43
Tabla 18: Tarea de Ingeniería – Modificar sistema de detección y supresión de incendios. 43
Tabla 19: Tarea de Ingeniería - Eliminar sistema de detección y supresión de incendios. 44
Tabla 20: Tarea de Ingeniería - Listar riesgos tecnológicos. 44

Tabla 21: Tarea de Ingeniería - Añadir riesgos tecnológicos. 44
Tabla 22: Tarea de Ingeniería – Modificar riesgos tecnológicos. 45
Tabla 23: Tarea de Ingeniería - Eliminar riesgos tecnológicos. 45
Tabla 24: Tarea de Ingeniería - Listar topología lógica de la red. 46
Tabla 25: Tarea de Ingeniería - Añadir topología lógica de la red. 46
Tabla 26: Tarea de Ingeniería – Modificar topología lógica de la red. 46
Tabla 27: Tarea de Ingeniería - Eliminar topología lógica de la red. 47
Tabla 28: Tarea de Ingeniería - Listar topología física de la red. 47
Tabla 29: Tarea de Ingeniería - Añadir topología física de la red. 48
Tabla 30: Tarea de Ingeniería – Modificar topología física de la red. 48
Tabla 31: Tarea de Ingeniería - Eliminar topología física de la red. 48
Tabla 32: Caso de Prueba de Aceptación- Listar nombre de institución – Datos válidos. 51
Tabla 33: Caso de Prueba de Aceptación- Añadir nombre de institución – Datos válidos. 52
Tabla 34: Caso de Prueba de Aceptación- Añadir nombre de institución - Datos inválidos. 52
Tabla 35: Caso de Prueba de Aceptación- Modificar nombre de institución – Datos válidos. 53
Tabla 36: Caso de Prueba de Aceptación- Modificar nombre de institución - Datos inválidos. 54
Tabla 37: Caso de Prueba de Aceptación- Eliminar nombre de institución – Datos válidos. 54
Tabla 38: Caso de Prueba de Aceptación- Listar locales tecnológicos – Datos válidos. 55
Tabla 39: Caso de Prueba de Aceptación- Añadir locales tecnológicos – Datos válidos. 55
Tabla 40: Caso de Prueba de Aceptación- Añadir locales tecnológicos - Datos inválidos. 56
Tabla 41: Caso de Prueba de Aceptación- Modificar locales tecnológicos – Datos válidos. 57
Tabla 42: Caso de Prueba de Aceptación- Modificar locales tecnológicos - Datos inválidos. 57
Tabla 43: Caso de Prueba de Aceptación- Eliminar locales tecnológicos – Datos válidos. 58
Tabla 44: No Conformidades identificadas por iteraciones. 59

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) son cada vez más usadas para el apoyo y automatización de todas las actividades de las empresas. Gracias a ellas, las organizaciones han logrado obtener importantes beneficios, entre los que caben mencionar la optimización de sus recursos, la apertura a nuevos mercados y un conocimiento más profundo acerca de las necesidades de la clientela para brindarles un servicio de mejor calidad y una comunicación más fluida, no sólo con sus empleados sino también con sus clientes y proveedores (Aráuz 2016).

Los constantes avances que han experimentado las tecnologías en los últimos tiempos, han significado una verdadera revolución en la sociedad. En consecuencia, las organizaciones se han percatado de que la aplicación de las nuevas tecnologías en el ámbito de trabajo, posibilita el uso de las TIC en las diferentes áreas de las entidades, lo que ha propiciado un ahorro de costos y tiempo, ayudándoles a la vez con una mejor gestión de los flujos de información.

Hoy para cualquier empresa es importante la tecnología, en el entorno informático empresarial siempre se está en búsqueda de más capacidad de cómputo y más almacenamiento, debido a la gran cantidad de datos que son generados los cuales requieren de una infraestructura tecnológica para garantizar su utilización.

A partir de ello, las empresas, entidades e instituciones optan por contar con centros de datos que no son más que espacios que pueden abarcar desde una habitación hasta grandes extensiones y albergan en su interior múltiples servidores y unidades de almacenamiento, así como conexiones y cableado (Porto y Gardey 2018). Es donde se encuentra alojada toda la información que se maneja a diario que hace posible gran parte de los servicios online y las aplicaciones que se utilizan.

Los centros de datos disponen de un cúmulo de información sensible que debe ser protegida para garantizar todos los servicios que brindan, por consecuencia, muchos de estos centros se rigen por la norma TIA-942[1], en la cual se especifican los requerimientos mínimos en que debe estar conformada la infraestructura tecnológica, con el fin de optimizar sus servicios y explotar sus máximos potenciales. La norma TIA-942 debe ser cuidadosamente estudiada y aplicada para utilizar correctamente todas las potencialidades de cada uno de estos centros de datos (TIA-942 standard 2014).

[1] TIA-942: Concebido como una guía para los diseñadores e instaladores de centros de datos. Proporciona una serie de recomendaciones y directrices para la instalación de sus infraestructuras.

Actualmente, existen empresas que cuentan con expertos en diseño y estructuración de centros de datos que, a su vez, prestan servicios de contratación externa denominadas consultorías. Este servicio profesional constituye una ventaja de gran utilidad que tributa a que entidades identifiquen los principales problemas que afectan a los centros de datos. Además, permite analizar las causas que lo provocan en aras de tomar decisiones acertadas para su perfeccionamiento, alcanzando de esta forma sus propósitos fundamentales (Quintana 2017).

La Universidad de las Ciencias Informáticas (UCI) cuenta con centros de producción de software que están vinculados a varias esferas de la sociedad, uno de ellos es el Centro de Telemática (TLM), orientado al desarrollo de sistemas y servicios informáticos integrales para las Telecomunicaciones y la Seguridad Informática (UCI 2018). Dicho centro cuenta con el Departamento de Aplicaciones y el Departamento de Componentes. Este último, dispone de especialistas destinados a realizar servicios de consultorías a diversas instituciones nacionales, con la finalidad de diagnosticar el estado actual de los centros de datos.

El servicio de consultoría brindado por la UCI, se basa en los requerimientos establecidos por la norma TIA-942 y como parte de este proceso, se debe generar un informe llamado Diagnóstico que contenga un levantamiento de información, guiado por los parámetros a evaluar que propone dicha norma. En él, se evidencia la situación actual de la infraestructura tecnológica que conforma al centro de datos evaluado, sentando las bases para la confección de un segundo documento llamado Diseño, en el cual se propone mejorar aspectos en cuanto a la estructuración, tecnologías y servicios con que cuenta el centro de datos de acuerdo a los recursos que se dispone, del mismo modo, se recomienda adquirir nuevas tecnologías en caso de que la necesiten.

Actualmente, no se cuenta con un registro oficial de los informes y diseños de las consultorías realizadas, lo que significa que no se encuentran almacenadas de forma centralizada, así como tampoco se cuenta con un respaldo de datos capaz de asegurar toda esta información en caso de que exista algún daño en el equipo donde se encuentre almacenada. El proceso de elaboración y levantamiento de información de estos documentos, se realizan de forma manual por cada consultor, lo que contribuye a que sea confeccionado de acuerdo a la perspectiva del mismo, lo cual afecta la homogeneidad del documento final en cuanto al diseño, estructura, parámetros a evaluar y nombre de las secciones con que cuenta, afectando directamente la calidad de este proceso. Además, pueden existir duplicados innecesarios que conllevarían a una pérdida de tiempo y recursos.

Según la problemática anteriormente planteada se define como **problema a resolver:** ¿Cómo automatizar la elaboración de documentos de consultorías a centros de datos que brinda el Departamento de Componentes del Centro de Telemática?

El problema antes mencionado se centra en el **objeto de estudio:** Procesos de consultorías a centros de datos.

Teniéndose como **objetivo general:** Desarrollar un sistema para automatizar el proceso de elaboración de documentos de consultorías realizadas a centros de datos que brinda el Departamento de Componentes del Centro de Telemática.

Para ello se identifica como **campo de acción:** Sistemas que automatizan los procesos de consultorías a centros de datos.

Para dar cumplimiento al objetivo establecido se definen las siguientes **tareas de investigación**:

1. Análisis de los principales conceptos asociados para una mayor comprensión de la problemática planteada en esta investigación.
2. Análisis de la norma por la cual se rige el Departamento de Componentes para guiar y determinar los parámetros a seguir en la solución propuesta.
3. Examen de los sistemas que automatizan los procesos de consultorías a nivel nacional e internacional, para establecer semejanzas con la investigación y tomar buenas prácticas.
4. Selección de las herramientas y tecnologías que permitan una correcta implementación del sistema.
5. Exploración de los diferentes tipos de prueba que propone la metodología seleccionada para validar el correcto funcionamiento del sistema desarrollado.

Para dar cumplimiento a las tareas propuestas se tuvo en cuenta el estudio de varios métodos de investigación:

Métodos Teóricos:

- **Analítico-Sintético:** Permite realizar el estudio teórico de la investigación, facilitando el análisis de documentos y la extracción de los elementos más importantes acerca del proceso de consultorías a centros de datos.
- **Histórico-Lógico:** Este método permite analizar todo lo referente a la automatización de los procesos de consultorías en el ámbito nacional e internacional, para de esta forma obtener un conocimiento histórico sobre el tema.

Métodos Empíricos:

- **Observación:** Se refleja durante todo el transcurso de la investigación, fundamentalmente para observar los informes generados durante el proceso de consultoría.
- **Medición:** El sistema obtenido como resultado de la investigación será medido a través de las pruebas unitarias y de aceptación para garantizar el correcto funcionamiento del mismo.

- **Modelación:** Es utilizado para realizar las actividades encaminadas a la construcción de modelos los cuales se representa el objeto de estudio, se crean abstracciones con el objetivo de explicar la realidad. La condición principal de la modelación es la relación entre el modelo y el objeto que se modela.

En aras de estructurar el proceso de desarrollo del presente trabajo de una manera coherente y organizada, se dividió el mismo en un total de cuatro capítulos:

Capítulo 1. Fundamentación teórica: Se abordan los aspectos teóricos que soportan esta investigación, así como el estudio del ámbito nacional e internacional, metodología de desarrollo, las herramientas y tecnologías que se pudieran tener en cuenta para dar solución al problema planteado.

Capítulo 2. Propuesta de solución: Se detalla la propuesta de solución y se definen las funcionalidades y características que tendrá la propuesta del sistema, lo que permitirá un mejor entendimiento del mismo.

Capítulo 3. Exploración, Planificación y Diseño: Se detallan los artefactos generados durante la fase de exploración, planificación y diseño del proyecto.

Capítulo 4. Implementación y Pruebas: Se expone todo lo concerniente a los procesos de implementación y pruebas del sistema.

Capítulo 1: Fundamentación teórica

En el presente capítulo se realiza un análisis de las posibles soluciones existentes a nivel nacional e internacional para poder analizar las funcionalidades y características que se pudieran tener en cuenta para el desarrollo de la solución propuesta. Se hace una descripción de todo lo relacionado con los conceptos asociados al dominio del problema. Además, se fundamenta la selección de la metodología de desarrollo de software, tecnologías y herramientas a utilizar para el desarrollo de la aplicación.

1.1 Conceptos asociados

Consultorías

Existen numerosas definiciones del término “consultoría” y de su aplicación a situaciones y problemas empresariales, es decir, de la consultoría de empresas. Fritz Steele (1975) define: “Por consultoría se entiende cualquier forma de proporcionar ayuda sobre el contenido, proceso o estructura de una tarea o de un conjunto de tareas, en que el consultor no es efectivamente responsable de la ejecución de la tarea misma, sino que ayuda a los que lo son” (International Labour Office 2002).

Centros de datos

Se denomina Centro de Proceso de Datos a aquella ubicación donde se concentran todos los recursos necesarios para el procesamiento de la información de una organización. Dichos recursos consisten esencialmente en unas dependencias, debidamente acondicionadas, de computadoras y redes de comunicaciones (Galvan 2013).

Un centro de datos es un edificio o sala de gran tamaño usada para mantener en él una gran cantidad de equipamientos electrónicos. Suelen ser creados y mantenidos por grandes organizaciones con el objeto de tener acceso a la información necesaria para sus operaciones en todo momento. Prácticamente todas las compañías, ya sean medianas o grandes, tienen algún tipo de centro de datos, mientras que las más grandes llegan a tener varios (Galvan 2013).

Infraestructura tecnológica

La infraestructura tecnológica agrupa y organiza el conjunto de elementos tecnológicos que integran un proyecto, soportan las operaciones de una organización o sustentan una operación. Una infraestructura define el éxito de una empresa en la medida de que su robustez, calidad y sostenibilidad se traduce en incremento de la inversión en TI. Por este motivo es crucial conocer todos sus componentes o elementos a

nivel de software y de hardware. Una infraestructura sólida permite a un software operar de manera eficiente y eficaz durante el tiempo previsto con niveles altos de servicios y prestaciones (Infraestructura Tecnológica de Software 2018).

1.2 Norma por la cual se rige el Departamento de Componentes del Centro de Telemática para la infraestructura de los centros de datos

En abril de 2005, la *Telecomunication Industry Association* publicó su estándar TIA-942 con la intención de unificar criterios en el diseño de áreas de tecnologías y comunicaciones. Este estándar que en sus orígenes se basa en una serie de especificaciones para comunicaciones y cableado estructurado, avanza sobre los subsistemas de infraestructura, al generar los lineamientos que se deben seguir para clasificar estos subsistemas en función de los distintos grados de disponibilidad que se pretende alcanzar (Galvan 2013).

Espacio y diagrama de distribución

Según la norma, un centro de datos debe tener las siguientes áreas funcionales clave:

- Uno o más cuartos de entrada
- Un área de distribución principal (MDA, por sus siglas en inglés: main distribution area)
- Una o más áreas de distribución horizontal (HDA, por sus siglas en inglés: horizontal distribution areas)
- Un área de distribución de zona (ZDA, por sus siglas en inglés: zone distribution area)
- Un área de distribución de equipos

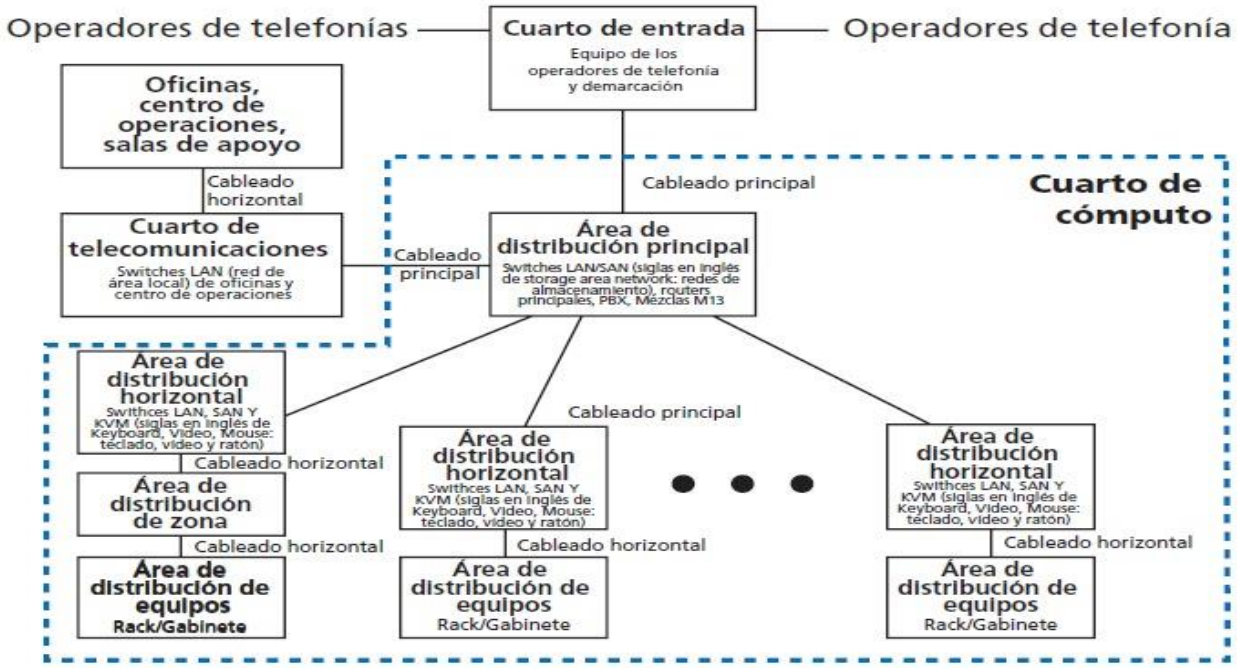


Figura 1: Centro de datos que cumple con la TIA-942.

Fuente: (ADC Telecomunications 2005)

Cuarto de Entrada: El cuarto de entrada alberga el equipo de los operadores de telefonía y el punto de demarcación. Puede estar dentro del cuarto de cómputo, pero la norma recomienda que esté en un cuarto aparte por razones de seguridad. Si está ubicado en el cuarto de cómputo, deberá estar consolidado dentro del área de distribución principal.

Área de distribución principal: El área de distribución principal alberga el punto de conexión cruzada central para el sistema de cableado estructurado del centro de datos. Esta área debe estar ubicada en una zona central para evitar superar las distancias del cableado recomendadas y puede contener una conexión cruzada horizontal para un área de distribución de un equipo adyacente. La norma especifica racks separados para los cables de fibra, UTP y coaxial.

Área de distribución horizontal: El área de distribución horizontal es la ubicación de las interconexiones horizontales, el punto de distribución para el cableado hacia las áreas de distribución de los equipos. Puede haber una o más áreas de distribución horizontal, según el tamaño del centro de datos y las necesidades de cableado. Una directriz para un área de distribución horizontal especifica un máximo de 2000 cables UTP de 4 pares o terminaciones coaxiales. Como en el caso del área de distribución principal, la norma especifica racks separados para cables de fibra, UTP y coaxiales.

Área de distribución de zonas: Es el área de cableado estructurado para los equipos que van en el suelo y no pueden aceptar paneles de parcheo. Como ejemplo, se puede citar a las computadoras centrales y los servidores.

Área de distribución de los equipos: Es la ubicación de los gabinetes y racks de equipos. La norma especifica que los gabinetes y racks se deben colocar en una configuración "hot aisle/cold aisle" ("pasillo caliente/pasillo frío") para que disipen de manera eficaz el calor de los equipos electrónicos.

Administración de cables

La clave para la administración de los cables en el centro de datos óptima es comprender que el sistema de cableado es permanente y genérico. Es como el sistema eléctrico, un servicio muy confiable y flexible al que se puede conectar cualquier aplicación nueva. Cuando está diseñado con este concepto en mente, no es difícil o perjudicial hacer adiciones o cambios.

Los sistemas de cableado altamente confiables y resistentes cumplen con los siguientes principios:

- Se usan racks comunes en toda la distribución principal y las áreas de distribución horizontal para simplificar el montaje del rack y brindar un control unificado de los cables.
- Se instala administradores de cables verticales y horizontales, comunes y extensos dentro de y entre los racks para garantizar una administración de cables eficaz y prever un crecimiento ordenado.
- Se instalan extensas trayectorias para cables (por arriba y por debajo del piso), para garantizar una administración de cables eficaz y prever un crecimiento ordenado.
- Los cables UTP y coaxiales se separan de la fibra en las trayectorias horizontales para evitar aplastarla. Los cables eléctricos van en bandejas de cables y la fibra, en canales montados en bandejas.
- El tendido de la fibra se hace en un sistema de canales para evitar que se dañe.

Racks y gabinetes

La administración de los cables comienza con los racks y gabinetes, que deben brindar un amplio control de cables horizontales y verticales. Una administración adecuada no sólo mantiene el cableado organizado, sino que también mantiene los equipos frescos al eliminar los obstáculos que impiden el movimiento del aire. Estas características deben proteger los cables, asegurar de que no se excedan los límites del radio de curvatura y manejar su holgura con eficacia.

Sistemas de tendido de cable

Una clave para lograr un tendido de cables óptimo es tener extensas trayectorias de cables superiores y por debajo del piso. Use el trayecto por debajo del piso para el cableado permanente y el trayecto superior para el cableado temporal. Separe la fibra de los cables UTP y coaxiales para garantizar que el peso de los otros cables no aplasta a la fibra que es más frágil.

Energía

La electricidad es la parte vital de un centro de datos. Un corte de energía de apenas una fracción de segundo es suficiente para ocasionar una falla en el servidor. Para satisfacer los exigentes requerimientos de disponibilidad de servicio, los centros de datos hacen todo lo posible para garantizar un suministro de energía confiable. Los procedimientos normales incluyen:

- Dos o más alimentaciones de energía de la empresa de servicio
- Suministro de Alimentación Ininterrumpible (UPS, por sus siglas en inglés: Uninterrupted power supplies)
- Circuitos múltiples para los sistemas de cómputo y comunicaciones y para equipos de enfriamiento
- Generadores en-sitio

Las medidas que se tomen para evitar disrupciones dependerá del nivel de fiabilidad requerido y, desde luego, de los costos.

Refrigeración

Los servidores, dispositivos de áreas de almacenamiento y los equipos de comunicación vienen cada vez más pequeños y potentes. La tendencia es usar más equipos en espacios más pequeños, y de esta forma se concentra una cantidad increíble de calor. Es un gran desafío ocuparse de este calor. Aunque sea una solución inicial, tener equipos de refrigeración adecuados es una buena forma para empezar a resolver el problema. La circulación de aire también es muy importante. Para favorecer la circulación de aire, la industria ha adoptado un procedimiento conocido como “hot aisle/cold aisle” (“pasillo caliente/pasillo frío”). En una configuración hot aisle/cold aisle, los racks de los equipos se disponen en filas alternas de pasillos calientes y fríos. En el pasillo frío, los racks de los equipos se disponen frente a frente. En el pasillo caliente, están dorso contra dorso. Las placas perforadas en el piso elevado de los pasillos fríos permiten que llegue aire frío al frente de los equipos. Este aire frío envuelve al equipo y se expulsa por la parte trasera hacia pasillo caliente. En el pasillo caliente, desde luego, no hay placas perforadas para evitar que se mezclen el

aire caliente con el frío. Para obtener los mejores resultados con este método, los pasillos deben tener dos azulejos de ancho para permitir el uso de placas perforadas en ambas filas, si fuera necesario (ADC Telecomunications 2005).

El propósito del estándar es proveer una serie de recomendaciones para el diseño e instalación de un centro de datos, con la intención de que sea utilizado por los diseñadores que necesitan un conocimiento acabado del sistema de cableado y el diseño de redes. Para cualquier tipo de negocio que necesite proyectar un centro de datos propio, por sus necesidades específicas, el estándar tratado hasta el momento es una buena guía con respecto a las buenas prácticas acerca de cómo arreglar las arquitecturas básicas de procesamiento de información (Galvan 2013).

Luego de conocer algunos conceptos como consultorías, centros de datos y realizar un resumen sobre la norma TIA- 942, se puede afirmar que con el crecimiento que existe en las empresas es indispensable tener una planificación de los centros de datos, pues el mal cálculo y la proyección pueden generar gastos dobles y pérdida de todo lo que se tiene a nivel de hardware y software.

1.3 Sistemas de servicios de consultorías. Estado del Arte

1.3.1 Sistema de registro y control de actividades en servicios de consultoría para la empresa MASTERTICS S.A.S

Es una aplicación web realizada para la empresa MASTERTICS S.A.S, la cual ofrece servicios de consultoría a diferentes plataformas tecnológicas y áreas de servicio de las tecnologías de la información y la comunicación. El sistema permite la gestión de actividades de consultoría y brinda flexibilidad a los consultores a la hora de registrar y llevar un control de las actividades realizadas en un proyecto. Permite registrar un contrato pactado con el cliente, registrar la información de las actividades realizadas por los consultores y generar reportes que contengan la información referente a los clientes de la empresa y los registros de las actividades realizadas por los consultores. A su vez, la aplicación permite interactuar con la información a través de computadoras de escritorio, equipos portátiles, así como también en Smartphone y tabletas, al permitir que el equipo de trabajo no dependa de su permanencia en su lugar de trabajo ni de un computador de escritorio o portátil para interactuar con el aplicativo (MEDINA 2013).

Es una aplicación enfocada a proyectos y todos los recursos tecnológicos que lo componen y no a centros de datos, por lo que no se rige por la norma TIA-942, por tanto, no se pueden valorar los parámetros asociados que se necesitan para realizar consultorías. Además, es una aplicación diseñada específicamente para cubrir las necesidades de la empresa MASTERTICS S.A.S.

1.3.2 Sunbird

Esta aplicación está destinada a la administración de centros de datos, supervisar problemas técnicos y de TI (Tecnologías de Información). Esto incluye la administración de las operaciones de las computadoras y el servidor, grandes cantidades de datos, servicios y aplicaciones.

Permite realizar diseños a centros de datos en cuanto a la administración de activos, para trazar con precisión todos los medios en los centros de datos globales. La capacidad, para ver rápidamente el espacio, la potencia y la conectividad de red. La energía, para aumentar la eficiencia energética, reducir los costos e impulsar el ahorro de energía. El ambiente, a través de un monitoreo inteligente para ayudarlo a mantener la salud del centro de datos. La conectividad, para administrar la conectividad de la red y los puertos. Es un producto privativo comercializado por la empresa que presenta el mismo nombre (Sunbird 2018). Este sistema brinda servicios para la administración y diseño de centros de datos, pero no realiza consultorías que se adhieran a la norma TIA- 942.

1.3.3 Schneider Electric

Software para centros de datos que permite realizar operaciones en cuanto a cómo planificar y optimizar la utilización de las capacidades reales de la infraestructura física mediante un modelo, de modo que permita facilitar la provisión eficiente de equipos y el dimensionamiento adecuado del centro de datos. Además, optimiza la TI, donde permite reducir el consumo de energía de los sistemas informáticos optimizando la utilización de los servidores para el aumento del rendimiento del centro de datos. Perfecciona el enfriamiento, donde elimina de forma dinámica las concentraciones de calor tributando al ahorro de energía. Permite el acceso a servidores, la capacidad total de acceso, apagado y re-encendido durante su ciclo de vida para gestión remota. Es un producto privativo distribuido por la empresa que presenta el mismo nombre (Schneider Electric 2018). No es un sistema orientado al diseño de consultorías a centros de datos y no se rige por la norma TIA- 942, en cambio, sí se toma en cuenta los aspectos fundamentales para la optimización de la infraestructura tecnológica, que es el objetivo fundamental de dichas consultorías.

1.3.4 CIGET

En nuestro país se encuentra el Centro de información y Gestión Tecnológica (CIGET) de Villa Clara. Es una entidad de interfaz entre las organizaciones de producción de bienes y servicios y de investigación-desarrollo. Está indisolublemente ligado a propiciar el desarrollo y la innovación tecnológica en el sector empresarial estatal del territorio. Brinda servicios de consultorías con enfoque de proceso y orientado al cliente, de manera flexible y proactiva para que garanticen rapidez de respuesta a las demandas del

mercado, así como la satisfacción con los resultados. Contribuye a posibilitar la planificación estratégica y la toma de decisiones de las empresas y sectores priorizados con una amplia gama de servicios y productos (Aranguiz y Cloris 2007).

1.3.5 BIOMUNDI

En nuestro país se encuentra el centro BIOMUNDI, la cual es la Dirección de Inteligencia Corporativa del Instituto de Información Científica y Tecnológica (IDICT/Cuba). Es un centro especializado en brindar servicios y productos de información e inteligencia empresarial que son de utilidad en las esferas científica, tecnológica, económica y comercial, para auxiliar a los empresarios en la toma de decisiones, tanto en el ambiente interno de la organización como en su entorno. Ofrece productos y servicios de información, consultoría e inteligencia empresarial que favorecen la toma de decisiones de los clientes, para un mejor desempeño de las entidades cubanas (BioMundi 2017).

Ambas empresas nacionales brindan servicios de consultorías a entidades del país y no a centros de datos en particular, por lo que no favorecen a la solución del problema planteado.

1.4 Conclusiones del estudio

El estudio realizado a los sistemas anteriores, evidenció la carencia de sistemas que automaticen la elaboración de documentos de consultorías a centros de datos que cumplan con las necesidades del Departamento de Componentes. Además, los sistemas estudiados a nivel internacional cuentan con una licencia de pago y fueron creados con fines totalmente diferentes a los requeridos para la aplicación a desarrollar. Sin embargo, se pudo constatar la existencia de sistemas para el diseño y administración de centros de datos que existen en el mercado, debido a la creciente demanda de las organizaciones y empresas.

Se analizaron diferentes centros que ofrecen servicios de consultorías a entidades para apoyar la toma de decisiones, pero no existen sistemas que propongan una solución viable a la problemática planteada en esta investigación. Sin embargo, el estudio arrojó resultados útiles que permitió tomar experiencias para una mayor comprensión de aspectos relacionados con el diseño de los centros de datos. A partir de lo anteriormente expuesto, se propone realizar un sistema que automatice la elaboración de documentos de consultorías a centros de datos, que cubra las necesidades específicas del Departamento de Componentes del Centro de Telemática.

1.5 Metodologías de desarrollo de software

El desarrollo de software no es una tarea fácil. Prueba de ello es que existen numerosas propuestas metodológicas que inciden en distintas dimensiones del proceso de desarrollo. Por una parte, se tienen aquellas propuestas más tradicionales que se centran especialmente en el control del proceso, que establecen rigurosamente las actividades involucradas, los artefactos que se deben producir, y las herramientas y notaciones que se usarán. Estas propuestas han demostrado ser efectivas y necesarias en un gran número de proyectos, pero también han presentado problemas en otros muchos. Una posible mejora es incluir en los procesos de desarrollo más actividades, más artefactos y más restricciones, basándose en los puntos débiles detectados. Sin embargo, el resultado final sería un proceso de desarrollo más complejo que puede incluso limitar la propia habilidad del equipo para llevar a cabo el proyecto. Otra aproximación es centrarse en otras dimensiones, como por ejemplo el factor humano o el producto software. Esta es la filosofía de las metodologías ágiles, las cuales dan mayor valor al individuo, a la colaboración con el cliente y al desarrollo incremental del software con iteraciones muy cortas.

Las metodologías ágiles están revolucionando la manera de producir software, y a la vez generando un amplio debate entre sus seguidores y quienes por escepticismo o convencimiento no las ven como alternativa para las metodologías tradicionales (Canós, Letelier y Penadés 2012).

Para la realización del sistema propuesto se decidió utilizar la metodología de desarrollo Extreme Programming (Programación Extrema, XP por sus siglas en inglés). XP es una metodología basada en valores de simplicidad, comunicación y retroalimentación. Funciona mediante la unión de todo el equipo y la aplicación de prácticas bien definidas, con suficiente retroalimentación para permitirle al equipo ver dónde están y adaptar sus prácticas a cada situación única (Beck 2000).

1.5.1 Fundamentación de la metodología de desarrollo seleccionada

Se decidió utilizar XP debido a que se adapta en gran medida tanto al tipo de proyecto a desarrollar como a las condiciones de trabajo y es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software. Se exponen varias de las razones que llevaron al uso de esta metodología.

- El proyecto es pequeño. XP está concebida para ser utilizada dentro de proyectos pequeños.
- Los requisitos del sistema pueden cambiar. El sistema debe cambiar y ampliar sus funcionalidades de forma que sea capaz de adaptarse a cada nueva situación. Uno de los principios básicos de XP

es que el cambio frecuente de los requerimientos es algo normal en el proceso de desarrollo. Esta metodología se adapta perfectamente a los proyectos cuyos requerimientos cambian a menudo.

- El cliente forma parte del equipo de desarrollo. Mediante la aplicación de XP se puede lograr una retroalimentación mayor y lograr un producto que satisfaga sus necesidades.
- El riesgo de desarrollo es elevado debido al corto tiempo de entrega planteado y a los continuos cambios de requerimientos. XP está diseñada a mitigar los riesgos en proyectos con estas características.
- Algunas prácticas de XP como las entregas pequeñas, el diseño simple y el Desarrollo Guiado por Pruebas (TDD, del inglés Test Driven Development) son parte de la filosofía de desarrollo del equipo.

1.6 Tecnologías y herramientas de desarrollo

Se describen las tecnologías y herramientas seleccionadas para el desarrollo del presente trabajo.

1.6.1 Marcos de trabajo

Django 1.8:

Framework web implementado sobre el lenguaje de programación Python, perteneciente a la licencia BSD (licencia de software libre permisiva). Django brinda estructura al código fuente, fomenta las buenas prácticas de desarrollo web, lo que promueve un código legible y fácil de mantener.

La implementación del patrón de diseño MTV (modelo-plantilla-vista), es una característica propia que contiene el framework, la cual contribuye a la organización de las distintas partes de la aplicación y a modificar éstas sin afectar cualquier otra pieza del software. Su alta escalabilidad le posibilita manejar el crecimiento continuo de trabajo de manera fluida sin perder calidad en los servicios (Django 2018).

Xilema-Base-Web 1.0:

Es un marco de trabajo desarrollado en el centro de Telemática (TLM) que está constituido por Django como framework base, librerías de Javascript como son JQuery y Backbone (TLM 2016).

Cumple con las pautas de diseño de la universidad y la estrategia marcaria a seguir por el Centro de Telemática y la Universidad de las Ciencias Informáticas.

1.6.2 Lenguaje de programación

Python 2.7.6:

Python es un lenguaje de programación creado por Guido van Rossum a principios de los años 90. Es un lenguaje con una sintaxis muy limpia lo que favorece un código legible. Se trata de un lenguaje interpretado o de script, con tipado dinámico, multiplataforma y orientado a objetos (Duque 2010).

Es un lenguaje de programación de código abierto para el desarrollo de aplicaciones independientes y de scripting en una gran cantidad de dominios de problemas. Es gratis, portable, y relativamente fácil de usar. Python es empleado tanto en proyectos grandes como pequeños, mejorando la productividad de los desarrolladores y la calidad del software (Lutz 2013).

Python ha sido seleccionado como lenguaje de programación por las características expuestas anteriormente, así como por la gran cantidad de librerías disponibles que pueden ser útiles en el presente trabajo.

HTML 5:

HTML, más que el lenguaje de marcado predominante usado para estructurar el contenido web, es realmente una tecnología web que interrelaciona varios estándares. En esta última revisión del lenguaje se incluyen varias características verdaderamente útiles, como son nuevos elementos semánticos y mejoras en los formularios (MacDonald 2013).

JavaScript:

JavaScript es un potente lenguaje de scripting basado en objetos. El código JavaScript puede ser embebido directamente en un documento HTML e interpretado por un navegador web en el lado del cliente (su uso más extendido). Cuando se combina con el Document Object Model (DOM) definido por un navegador web, JavaScript permite crear o manipular dinámicamente el contenido HTML (Flanagan 2011).

UML (lenguaje de modelado unificado):

Lenguaje de Modelado Unificado (Unified Modeling Language) es la sucesión de una serie de métodos de análisis y diseño orientados a objetos. Fue creado para forjar un lenguaje de modelado visual común y sintácticamente rico para la arquitectura, el diseño y la implementación de sistemas de software complejos, tanto en estructura como en comportamiento (Cornejo 2008).

1.6.3 Bibliotecas utilizadas

ReportLab 3.4.0:

Es una biblioteca de software que le permite crear documentos directamente en el formato PDF [2] usando el lenguaje de programación Python. También crea tablas y gráficos de datos en varios mapas de bits y formatos de vector.

La biblioteca ReportLab crea directamente documentos PDF en función de sus comandos gráficos. Sus aplicaciones pueden generar informes extremadamente rápidos, a veces de mayor magnitud que herramientas tradicionales de redacción de informes. Es una herramienta muy útil para generar dinámicamente archivos PDF en aplicaciones web utilizando altos volúmenes de información provenientes de bases de datos (ReportLab 2018).

Bootstrap 2:

Es un enfoque de diseño web destinado a la elaboración de sitios web para proporcionar una visualización óptima para una experiencia de navegación fácil y con un mínimo de cambio de tamaño, paneo, y desplazamiento a través de una amplia gama de dispositivos.

Un sitio diseñado con Bootstrap adapta su diseño a las condiciones de observación mediante el uso de fluidos, las proporciones basadas en cuadrículas, imágenes flexibles y CSS3 en los medios (Arias 2014).

1.6.4 Herramientas de desarrollo

Pycharm 2017.2.2:

Es un Entorno de Desarrollo Integrado (IDE) desarrollado por la compañía Jetbrain. Es multiplataforma, hay binarios para: Windows, Linux y Mac OS X. Proporciona análisis de código, depuración gráfica y soporte para el desarrollo web con Django, entre otras bondades (Islam 2015).

Tiene un excelente soporte para Python, además de facilitar los procesos de búsquedas de errores y ejecución de tareas a través de consola (JetBrains 2018).

pgAdmin III:

Es un completo sistema de gestión y diseño de base de datos de la plataforma PostgreSQL para sistemas Unix y Windows. Está diseñada para responder a las necesidades de todos los usuarios, desde escribir consultas SQL sencillas hasta el desarrollo de bases de datos complejas. Está disponible

[2] PDF: Sigla del inglés Portable Document Format (formato de documento portátil).

gratuitamente bajo los términos de la licencia PostgreSQL y puede redistribuirse siempre que se cumplan los términos de la licencia (pgAdmin 2018).

1.6.5 Herramienta de modelado

Visual Paradigm 8.0 Enterprise Edition:

Es una herramienta CASE que soporta el modelado mediante UML y proporciona asistencia a los analistas, ingenieros de software y desarrolladores durante todos los pasos del ciclo de vida de desarrollo de un software. Facilita el modelado de UML, ya que proporciona herramientas específicas para ello. Esto también permite la estandarización de la documentación, ya que la misma se ajusta al estándar soportado por la herramienta (Visual Paradigm 2014).

1.6.6 Sistema gestor de base de datos

PostgreSQL 9.4:

Es un sistema de gestión de bases de datos objeto-relacional. Se encuentra distribuido bajo licencia BSD y tiene su código fuente disponible libremente. Utiliza un modelo cliente/servidor y usa multiprocesos en vez de multihilos para garantizar la estabilidad del sistema. Esto ofrece como ventaja que un fallo en uno de los procesos no afecte al resto, garantizando así que el sistema continúe funcionando (PostgreSQL 2018).

1.7 Conclusiones del Capítulo

Con el desarrollo del marco teórico se logró organizar y guiar el trabajo, profundizando en el estudio de algunos conceptos fundamentales para el desarrollo de un sistema que permita automatizar la elaboración de documentos de consultorías que realiza el Departamento de Componentes del Centro de Telemática. Se realizó un análisis de sistemas similares en el ámbito internacional como nacional, donde se llega a la conclusión de que ninguno cubría las necesidades que requiere el problema planteado en esta investigación. Se proponen las tecnologías, herramientas y metodología de desarrollo de software que se utilizarán en la elaboración del sistema informático.

Capítulo 2: Propuesta de solución

En el presente capítulo se describen las características del sistema a desarrollar. Se hace un análisis del proceso de servicio de consultorías a centros de datos que realiza el Departamento de Componentes, se definen las tareas vinculadas a este proceso que necesitan ser automatizadas, lo cual posibilita que se tenga un mejor conocimiento del mismo. Además, se describen las principales características no funcionales que el sistema debe cumplir.

2.1 Análisis del proceso de servicio de consultorías a centros de datos que realiza el Departamento de Componentes del Centro de Telemática

El Departamento de Componentes del Centro Telemática, cuenta con especialistas encargados de realizar las consultorías a centros de datos de distintas empresas del país. Los principales procedimientos que ocurren durante este proceso son los siguientes:

1. Establecer contacto con el cliente por medio de citas, llamadas y correos electrónicos.
2. Programar reunión con el cliente para llegar a un acuerdo y firmar un contrato.
3. Análisis y estudio del centro de datos de la empresa a consultar.
4. Recoger y seleccionar la información del centro de datos.
5. Generar diagnóstico de la consultoría.
6. Generar diseño de la consultoría.

Este proceso es repetido cada vez que se realiza una consultoría a una empresa, y al no disponer de un sistema automatizado para este servicio la manipulación de esta información se hace lenta y engorrosa, lo que provoca pérdida de tiempo y de homogeneidad en los reportes realizados.

En la presente investigación se plantea automatizar los siguientes procesos:

1. Recoger y seleccionar la información del centro de datos.
2. Generar diagnóstico de la consultoría.
3. Generar diseño de la consultoría.

2.2 Características de la propuesta del sistema

En la presente investigación se propone el desarrollo de un sistema que permita automatizar la elaboración de documentos de consultorías a centros de datos.

El levantamiento de información realizado a los centros de datos podrá ser introducido en el sistema, el cual contará con una base de datos disponible en PostgreSQL, que le permitirá guardar toda la información de acuerdo a los parámetros que propone la norma TIA- 942. El sistema dispondrá de dos módulos para la gestión del diagnóstico y el diseño respectivamente. El uso de la biblioteca ReportLab permitirá que toda la información guardada en el sistema pueda ser exportada en un documento en formato PDF.

El documento generado tanto para el diagnóstico como para el diseño tendrá tamaño de página: 21 cm x 27 cm. En la portada aparecerá el logotipo de la universidad, el Centro de Telemática, el nombre de la institución a la que se le realizará la consultoría y la fecha en la que fue generado el documento. En la segunda página se mostrará una tabla acerca del control del documento en el que se mostrarán los campos (Redactado, Aprobado, Revisado). Además, el documento del diseño contará con las reglas de confidencialidad con el propósito de no reproducir, divulgar, difundir o hacer público el contenido de ese informe. Mediante el uso de la clase ParagraphStyle incorporada en la biblioteca ReportLab, los documentos se generarán con encabezado en Arial 14 y el resto del documento estará en formato de párrafo, tipo y tamaño de letra Arial 11, interlineado 1.5 y justificado. En las últimas páginas del diagnóstico se podrá visualizar las conclusiones del estudio realizado al centro de datos y recomendaciones que servirán como base para generar un diseño mejor estructurado.

El sistema contará con dos funciones llamadas, generar_diagnostico() y generar_diseño(), las cuales dispondrán de las clases proporcionadas por la herramienta ReportLab como SimpleDocTemplate, utilizada para definir el estilo en el documento en la cual se contemplan los márgenes y el tamaño de página. Dichas funciones serán ejecutadas a través de dos botones incorporados en el módulo Administración.

Según el orden del documento, debe quedar plasmado en el mismo la fecha en la cual fue generado. Para ello, se hace uso de la librería nativa de Python datetime, la cual es capaz de proporcionar de forma fiel, la fecha y hora, en la cual fue generado dicho documento. Para la confección de tablas en el documento, las funciones antes mencionadas, utilizarán la clase Table, propia de ReportLab, en la que se define las columnas y filas que contendrá, así como sus correspondientes tamaños. El uso de la clase TableStyle permitirá aplicar diferentes estilos a estas tablas.

Los métodos antes mencionados, deberán consultar en cada uno de los modelos de Django toda la información guardada por el usuario en el sistema, la cual será obtenida e introducida en el documento a exportar mediante el uso de la clase Paragraph, la cual permitirá crear objetos tipo párrafo especificando el

texto plano (obtenido de la base de datos) y los estilos que se deseen aplicar, quedando conformado como resultado final los documentos necesitados ajustados a la información introducida por el sistema.

Este sistema contará con cuatro módulos en total: Sistema, Administración, Diagnóstico y Diseño. A continuación, se muestra una explicación detallada de cada uno de los correspondientes módulos.

Módulo Sistema: Se encuentra integrado en el marco de trabajo Xilema- Base-Web y permite la gestión de usuarios, así como los permisos de cada uno de éstos y los grupos a los que pertenecen.

Módulo de Administración: Es donde se asientan las bases para realizar las consultorías. En él, se gestionan los datos básicos acerca de las instituciones y el documento como: nombre de la institución, control del documento y nomencladores. Además, contará con las opciones añadir sección, la que permitirá agregar un nuevo parámetro a evaluar en los documentos, y añadir introducción, que como su nombre lo indica, permitirá incorporar una breve introducción en ambos documentos. A partir de cada institución registrada en el sistema, el usuario podrá generar los documentos de diagnóstico y diseño en formato PDF.

Módulo de Diagnóstico: En el módulo se gestionará la situación actual del centro de datos y todas las configuraciones necesarias que le permitan optimizar su estado para el correcto cumplimiento de la norma TIA- 942 y sus recomendaciones futuras. Estará conformado por todos los parámetros a evaluar dentro del diagnóstico de la consultoría, en el que de forma predeterminada se introducirán todos los datos realizados en un levantamiento de información previo.

Módulo de Diseño: El módulo permitirá gestionar un diseño idóneo de acuerdo a las configuraciones establecidas en la norma por la cual se rigen los centros de datos. Permitirá valorar los posibles cambios y mejoras de acuerdo a todos los parámetros evaluados en el módulo anterior.

A continuación, se representará el funcionamiento de la propuesta de solución, donde se puede observar cómo el usuario introduce la información en el sistema, el cual cuenta con un servidor que dispone de los módulos de administración, diagnóstico y diseño, que gestionan y guardan la información en la base de datos. Estos módulos interactúan con la biblioteca ReportLab, lo que hace posible que esta herramienta genere reportes de acuerdo a los datos almacenados en el sistema de las consultorías realizadas.

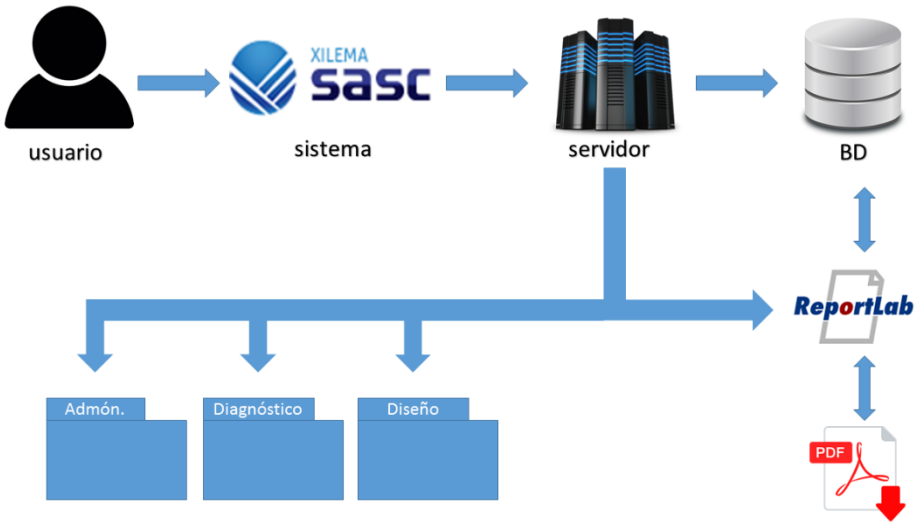


Figura 2: Propuesta de sistema.

2.3 Características no funcionales del sistema

Las características no funcionales son aquellas características o cualidades que debe cumplir el sistema. Además, constituyen las propiedades o cualidades que el producto debe poseer para su correcto funcionamiento. A continuación, deben tenerse en cuenta las siguientes características no funcionales:

Usabilidad:

- La aplicación deberá contar con una interfaz sencilla, descriptiva y fácil de usar por cualquier usuario con conocimientos básicos de computación.
- Debe estar adaptada a las pautas de diseño de la universidad y a la estrategia marcaria del centro.

Servidor

Software:

- Se recomienda hacer uso del navegador Mozilla Firefox (versión 35 o superior), Internet Explorer o Google Chrome.

Hardware:

- CPU Dual Core a 1.0 GHz o superior, 512 MB de RAM o superior, 20 GB de disco duro o superior.

Cliente

Software:

- Es necesario hacer uso del navegador Mozilla Firefox (versión 35 o superior), Internet Explorer o Google Chrome.

Hardware:

- CPU Pentium 300 MHz, 128 MB de RAM, 5 GB de disco duro.

Seguridad

Confidencialidad:

- La información manejada por el sistema está protegida de acceso no autorizado y divulgación, por lo cual se establecerá un nivel de acceso a la aplicación mediante usuario y contraseña, además, de los permisos asignados por usuarios.

Integridad:

- La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes. Esto se garantiza a través de la integridad referencial, la cual es una característica básica que proporciona PostgreSQL para garantizar la integridad de los datos almacenados. La integridad referencial es una función que está presente en las bases de datos relacionales que garantiza la coherencia de los datos entre relaciones aparejadas.

Disponibilidad:

- La aplicación deberá estar disponible en todo momento para aquellas personas que necesiten acceder y manejar la información.

2.4 Conclusiones del Capítulo

En el presente capítulo se realizó un análisis sobre el proceso de servicio de consultorías a centros de datos que realiza el Departamento de Componentes del Centro Telemática, el cual permitió definir los procesos que serán automatizados para dar solución al problema planteado en esta investigación. Además, se analizó el funcionamiento del sistema propuesto como solución, para así hacer énfasis en las

características de la propuesta del sistema y las características no funcionales que debe poseer el sistema para alcanzar el objetivo de esta investigación.

Capítulo 3: Exploración, planificación y diseño

En el presente capítulo se presentan los resultados obtenidos en las fases de exploración, planificación y diseño correspondientes a la metodología XP, y se generan los artefactos correspondientes a cada una de ellas.

3.1 Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario (HU) que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema para la construcción de un prototipo. La fase de exploración toma de pocas semanas a pocos meses, en dependencia del tamaño y familiaridad que tengan los programadores con la tecnología (Canós, Letelier y Penadés 2012).

3.1.1 Historias de usuario

Las historias de usuario son la técnica utilizada en XP para especificar las funcionalidades del software. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento las historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada HU es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (Canós, Letelier y Penadés 2012).

Durante la fase de exploración se identificaron 23 historias de usuario:

- **HU 1.** Gestionar nombre de institución
- **HU 2.** Gestionar locales tecnológicos
- **HU 3.** Gestionar enlaces
- **HU 4.** Gestionar sistema eléctrico
- **HU 5.** Gestionar clima
- **HU 6.** Gestionar local físico
- **HU 7.** Gestionar sistemas de monitoreo
- **HU 8.** Gestionar sistema de control de acceso
- **HU 9.** Gestionar sistema de detección y supresión de incendios
- **HU 10.** Gestionar riesgos tecnológicos

- **HU 11.** Gestionar tipo de equipamiento
- **HU 12.** Gestionar servicios
- **HU 13.** Gestionar red de almacenamiento
- **HU 14.** Gestionar recursos humanos
- **HU 15.** Gestionar certificaciones
- **HU 16.** Gestionar diagrama de comunicación entre servidores
- **HU 17.** Gestionar topología lógica de la red
- **HU 18.** Gestionar topología física de la red
- **HU 19.** Gestionar valores de entorno
- **HU 20.** Gestionar sistemas operativos
- **HU 21.** Exportar diagnóstico a formato PDF
- **HU 22.** Exportar diseño a formato PDF
- **HU 23.** Realizar Búsquedas

Cada historia de usuario es clasificada según:

Su prioridad para el negocio en:

- Alta: Se le otorga a las HU que son de funcionalidades fundamentales en el desarrollo del sistema.
- Media: Se le otorga a las HU que resultan necesarias, pero no imprescindibles para el funcionamiento del sistema.
- Baja: Se le otorga a las HU que son de funcionalidades a tener en cuenta, sin que estas tengan una afectación sobre el sistema que se esté desarrollando.

Riesgo en desarrollo:

- Alto: Cuando en la implementación de las HU pueda existir algún error y afecte el desarrollo de la funcionalidad correspondiente a esta HU.
- Medio: Cuando en la implementación de las HU puedan existir errores y retrasen el desarrollo de la funcionalidad correspondiente a esta HU.
- Bajo: Cuando en la implementación de las HU puedan existir errores, pero éstos son tratados fácilmente y no afectan el desarrollo de la funcionalidad correspondiente a esta HU.

Las HU son representadas mediante tablas, donde los puntos estimados representan días hábiles:

Tabla 1: HU - Gestionar nombre de institución.

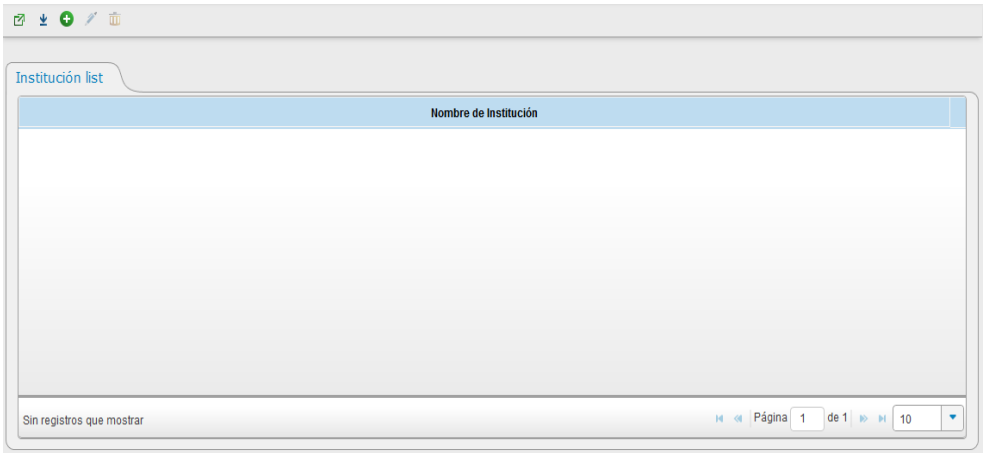
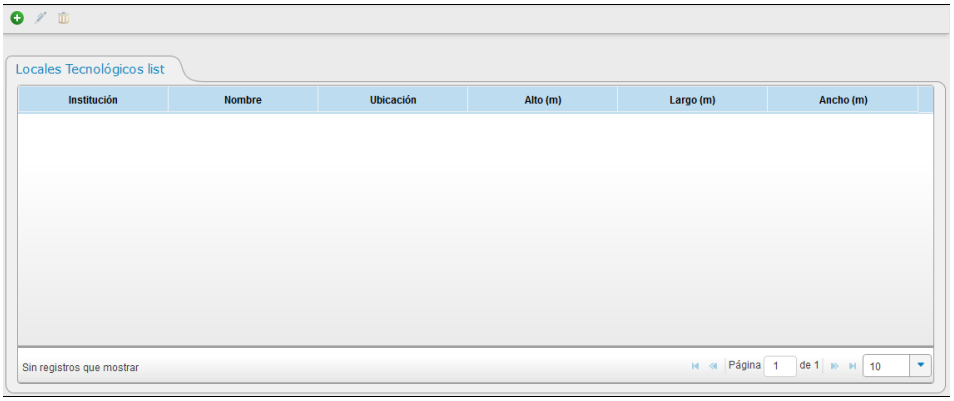
Número: 1	**Nombre:** Gestionar nombre de institución
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Alto
Puntos estimados: 0.2	**Iteración asignada:** 1
Programador responsable: Brenda García Pérez	
Descripción: Permite al usuario gestionar (crear, editar, visualizar, eliminar) el nombre de la institución a la que se realizará la consultoría en el sistema.	
Observaciones: El usuario puede acceder al sistema siempre y cuando se haya autenticado anteriormente.	
Prototipo de interfaz: 	

Tabla 2: HU - Gestionar locales tecnológicos.

Número: 2	**Nombre:** Gestionar locales tecnológicos
Prioridad en negocio: Alta	**Riesgo en desarrollo:** Alto
Puntos estimados: 0.4	**Iteración asignada:** 1
Programador responsable: Brenda García Pérez	

Descripción: Permite al usuario gestionar(crear, editar, visualizar, eliminar) en el sistema la información de los locales tecnológicos en cuanto a: nombre, ubicación y espacio que ocupan.
Observaciones: El usuario debe estar previamente autenticado.
Prototipo de interfaz: 

Para consultar el resto de las HU ver **Anexo I**.

3.2 Planificación

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Las estimaciones de esfuerzo asociado a la implementación de las HU la establecen los programadores utilizando como medida el punto. Un punto se considera como una semana ideal de trabajo donde los miembros de los equipos de desarrollo trabajan el tiempo planeado sin ningún tipo de interrupción (Beck 2000).

3.2.1 Estimación de esfuerzo por Historias de Usuario

Para el desarrollo del sistema propuesto en este trabajo se realizó una estimación del esfuerzo para cada una de las historias de usuario identificadas, llegándose a los resultados que se muestran en la siguiente tabla.

Tabla 3: Estimación del esfuerzo por HU.

No.	Nombre de historia de usuario	Puntos de estimación(semanas)
1	Gestionar nombre de la institución	0.2
2	Gestionar locales tecnológicos	0.4
3	Gestionar enlaces	0.4
4	Gestionar sistema eléctrico	0.4
5	Gestionar clima	0.4
6	Gestionar local físico	0.4
7	Gestionar sistemas de monitoreo	0.4
8	Gestionar sistema de control de acceso	0.4
9	Gestionar sistema de detección y supresión de incendios	0.4
10	Gestionar riesgos tecnológicos	0.4
11	Gestionar tipo de equipamiento	0.4
12	Gestionar servicios	0.4
13	Gestionar red de almacenamiento	0.4
14	Gestionar recursos humanos	0.4
15	Gestionar certificaciones	0.2
16	Gestionar diagrama de comunicación entre servidores	0.4
17	Gestionar topología lógica de la red	0.4
18	Gestionar topología física de la red	0.4
19	Gestionar valores de entorno	0.4
20	Gestionar sistemas operativos	0.4
21	Exportar diagnóstico a formato PDF	0.6
22	Exportar diseño a formato PDF	0.6
23	Realizar búsquedas	0.2

3.2.2 Plan de iteraciones

El plan de iteraciones incluye varias iteraciones sobre el sistema antes de ser entregado. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra al seleccionar las historias que fuercen la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide qué historias se implementarán

en cada iteración (para maximizar el valor de negocio). Al final de la última iteración el sistema estará listo para entrar en producción (Canós, Letelier y Penadés 2012).

Iteración 1

Esta iteración tiene como objetivo la implementación de las historias de usuario de mayor prioridad. En esta iteración se llevará a cabo las HU encargadas de Gestionar nombre de la institución, Gestionar locales tecnológicos, Gestionar enlaces, Gestionar sistema eléctrico, Gestionar clima, Gestionar local físico, Gestionar sistemas de monitoreo y Gestionar sistema de control de acceso. Durante el transcurso de la misma se creará la base de la arquitectura del sistema con una funcionalidad mínima. Al final de esta se contará con una primera versión de prueba, la cual será mostrada al cliente con el objetivo de obtener una retroalimentación para el equipo de desarrollo.

Iteración 2

El objetivo de esta iteración es la implementación de las historias de usuarios de prioridad media. En esta iteración se llevará a cabo las HU encargadas de Gestionar sistema de detección y supresión de incendios, Gestionar riesgos tecnológicos, Gestionar tipo de equipamiento, Gestionar servicios, Gestionar red de almacenamiento, Gestionar recursos humanos, Gestionar certificaciones y Gestionar diagrama de comunicación entre servidores. Al finalizar la misma se contará con una versión funcional del sistema donde se podrán probar las funcionalidades mencionadas anteriormente. Esta versión se mostrará al cliente con el objetivo de realizar cambios necesarios en base a la opinión del mismo.

Iteración 3

El objetivo de esta iteración es la implementación de las historias de usuarios de prioridad baja. En esta iteración se llevará a cabo las HU encargadas de Gestionar topología lógica de la red, Gestionar topología física de la red, Gestionar valores de entorno, Gestionar sistemas operativos, Exportar diagnóstico a formato PDF, Exportar diseño a formato PDF y Realizar búsquedas, que son las encargadas de complementar el resultado de lo implementado en las iteraciones anteriores. Al finalizar la misma se contará con la versión 1.0 del producto final y como resultado de ésta, el sistema estará listo para su despliegue.

Tabla 4: Plan de duración de las iteraciones.

Iteración	HU a implementar	Duración total de la iteración
	Gestionar nombre de la institución	
	Gestionar locales tecnológicos	

Iteración 1	Gestionar enlaces	3 semanas
	Gestionar sistema eléctrico	
	Gestionar clima	
	Gestionar local físico	
	Gestionar sistemas de monitoreo	
	Gestionar sistema de control de acceso	
Iteración 2	Gestionar sistema de detección y supresión de incendios	3 semanas
	Gestionar riesgos tecnológicos	
	Gestionar tipo de equipamiento	
	Gestionar servicios	
	Gestionar red de almacenamiento	
	Gestionar recursos humanos	
	Gestionar certificaciones	
	Gestionar diagrama de comunicación entre servidores	
Iteración3	Gestionar topología lógica de la red	3 semanas
	Gestionar topología física de la red	
	Gestionar valores de entorno	
	Gestionar sistemas operativos	
	Exportar diagnóstico a formato PDF	
	Exportar diseño a formato PDF	
	Realizar búsquedas	

3.2.3 Plan de entregas

A continuación, se presenta el plan de entregas elaborado para la fase de implementación, que comienza el lunes 5 de febrero de 2018. Como producto del mismo se harán entregas incrementales al finalizar cada iteración, en la fecha aproximada que se muestra en la siguiente tabla.

Tabla 5: Plan de entrega de las iteraciones.

Iteración	Fecha de entrega
Iteración 1	23/02/2018
Iteración 2	16/03/2018
Iteración 3	6/04/2018

3.3 Diseño

Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto. La complejidad innecesaria y el código extra debe ser removido inmediatamente. Kent Beck expresa que en cualquier momento el diseño adecuado para el software es aquel que: supera con éxito todas las pruebas, no tiene lógica duplicada, refleja claramente la intención de implementación de los programadores y tiene el menor número posible de clases y métodos (Canós, Letelier y Penadés 2012).

3.4 Arquitectura de software

“La arquitectura de software de un sistema o programa de computación es la estructura o estructuras del sistema, que comprenden elementos de software, las propiedades externamente visibles de estos elementos, y las relaciones entre ellos” (Durango 2015).

3.4.1 Arquitectura cliente- servidor

Esta arquitectura consiste básicamente en un cliente que realiza peticiones a otro programa (el servidor) que le da respuesta. Aunque esta idea se puede aplicar a programas que se ejecutan sobre una sola computadora es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras. La interacción cliente-servidor es el soporte de la mayor parte de la comunicación por redes (Areitio 2009).

Figura 3: Arquitectura Cliente- Servidor.

3.4.2 Patrón arquitectónico

La arquitectura de Django es una variación del patrón MVC (Modelo Vista Controlador), el cual es una guía de alto nivel para el diseño de arquitecturas de aplicaciones que ofrezcan una fuerte interactividad con usuarios.

Django es un framework MTV, del inglés “Model – Template -View” (Modelo Plantilla Vista), que intenta ser lo más funcional posible. En comparación con el patrón MVC, el Modelo en Django sigue como Modelo, a la Vista en Django se le denomina Plantilla y en el caso del Controlador se le denomina Vista.

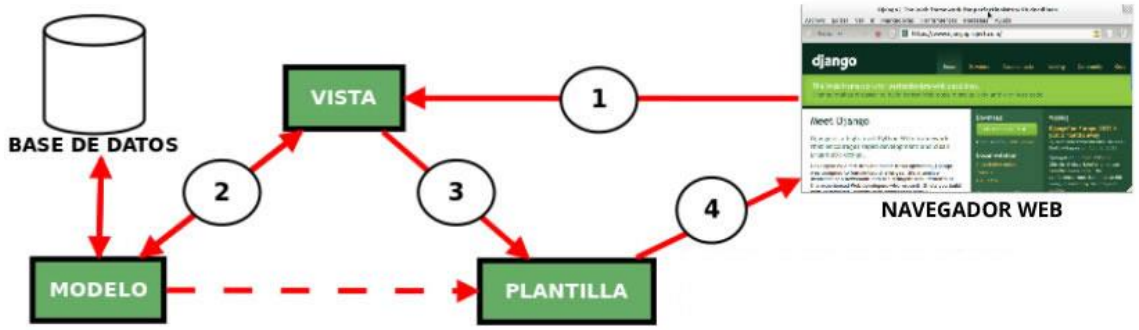


Figura 4: Funcionamiento de MTV en Django.

- **Modelo:** El modelo define los datos almacenados, se encuentra en forma de clase de Python, cada tipo de dato que debe ser almacenado se encuentra en una variable con ciertos parámetros; también posee métodos. Todo esto permite indicar y controlar el comportamiento de los datos.
- **Vista:** La vista se presenta en modo de funciones en Python y su propósito es determinar qué datos serán visualizados. El ORM[3] de Django permite escribir código Python en lugar de SQL para realizar las consultas que necesita la vista. La autentificación de servicios y la validación de datos a través de formularios son también tareas de las que se encarga la vista.
- **Plantilla:** La plantilla es básicamente una página HTML con algunas etiquetas extras propias de Django, en sí no solo crea contenido en HTML sino también XML, CSS y JavaScript. La plantilla recibe los datos de la vista y luego los organiza para la presentación al navegador web (Rios 2018).

3.4.3 Patrones de diseño utilizados en el desarrollo del sistema propuesto

El término de patrón fue dado por primera vez en el año 1977 por el arquitecto Christopher Alexander, quien dio en su libro A Pattern Language la siguiente definición: "Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la solución a ese problema, de tal manera que esa solución pueda ser usada más de un millón de veces sin hacerlo ni siquiera dos veces de la misma forma". En la ingeniería del software, un patrón constituye el apoyo para la solución a los

[3] ORM: El mapeo objeto relacional (más conocido por su nombre en inglés, Object-Relational mapping, o sus siglas ORM) es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación.

problemas más comunes que se presentan durante las diferentes etapas del ciclo de vida del software (Guerrero, Suárez y Gutiérrez 2013).

Para el diseño de la aplicación se hizo uso de los Patrones Generales de Software para Asignar Responsabilidades (GRASP), así como de los patrones de diseño.

Los Patrones (GRASP), describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones (Larman 2003). Los patrones GRASP utilizados fueron:

- **Alta cohesión:** Se aplica en la mayoría de las clases del diseño, ya que en cada una solo se implementan las funcionalidades que le corresponden.

```
class EnlacesSerializer(serializers.ModelSerializer):
    nombre_institucion =
serializers.SerializerMethodField(method_name='get_nombre_institucion')
    nombre_conexion =
serializers.SerializerMethodField(method_name='get_nombre_conexion')
    class Meta:
        model = enlaces
    @staticmethod
    def get_nombre_institucion(obj):
        local_tec = enlaces.objects.get(pk=obj.id)
        nombInst = Institucion.objects.get(pk=local_tec.institucion.id).nombre
        return nombInst
```

- **Bajo acoplamiento:** Es la idea de tener las clases lo menos ligadas entre sí como sea posible. De tal forma que, en caso de producirse una modificación en alguna de ellas, se tenga la mínima repercusión posible en el resto de clases, lo que potencia la reutilización, y disminuye la dependencia entre las clases.

  Ejemplo: Los modelos se relacionan con un número pequeño de clases, como es el caso de la clase institución, ya que la mayoría de las clases del sistema están relacionadas únicamente con la clase institución.
- **Creador:** El patrón creador ayuda a identificar quién debe ser el responsable de la creación o instanciación de nuevos objetos o clases, de forma tal que una instancia de un objeto solo pueda ser creada por el objeto que contiene la información necesaria para ello.

  Ejemplo: El patrón se encuentra evidenciado en la utilización del objeto obj, quien es una instancia de una clase y contiene los datos concurrentes de un objeto.

```
def get_nombre_institucion(obj):
    local_tec = Secciones.objects.get(pk=obj.id)
    nombInst = Institucion.objects.get(pk=local_tec.institucion.id).nombre
    return nombInst
```

- **Controlador:** El patrón controlador sirve como intermediario entre una determinada interfaz y el algoritmo que la implementa, este sugiere que la lógica de negocios debe estar separada de la capa de presentación, esto para aumentar la reutilización de código y a la vez tener un mayor control.
  Ejemplo: El patrón se evidencia en las clases serializers, las cuales determinan los datos que se mostrarán en la plantilla.
- **Experto:** Se mantiene el encapsulamiento, los objetos utilizan su propia información para llevar a cabo sus tareas. Se distribuye el comportamiento entre las clases que contienen la información requerida. Son más fáciles de entender y mantener.

```
x = datetime.datetime.now()
styles.add(ParagraphStyle(name='Fecha', fontSize=12, fontName="Helvetica",
spaceBefore=80, leading=20, spaceAfter=200))
fecha = Paragraph('Generado: '+x.strftime('%y/%m/%d %H:%M:%S'), styles['Fecha'])
```

Ahora bien, los patrones de diseño, describen soluciones simples y elegantes a problemas específicos en el diseño de software orientado a objetos (Larman 2003). Los utilizados en el diseño de la solución son:

- **Active Record:** El objetivo de este patrón es encapsular el acceso a una base de datos relacional y agrega lógica de dominio en esa información. Se evidencia en la implementación de los modelos de Django.
- **Identity Field:** Consiste en que las tablas de las bases de datos relacionales se encuentren identificadas. Se evidencia en las llaves primarias generadas a partir de la creación de los modelos de Django en la cual a cada tabla se le incorpora automáticamente la columna id (primary_key).
- **Command Pattern:** Encapsula una operación en un objeto, permitiendo ejecutar dicha operación sin necesidad de conocer el contenido de la misma. Se evidencia en el siguiente fragmento de código, en el cual se aprecia cómo se encapsula la información generada en un objeto tipo HttpResponse.

```
return HttpResponse ()
```

- **Front Controller (Controlador Frontal):** Django posee una implementación de Controlador Frontal que despacha las peticiones hacia métodos o clases, que en la práctica son páginas controladoras. Antes del despacho, la petición es procesada por varios filtros.

- **Decorator (Decorador):** Añade responsabilidades adicionales a un objeto dinámicamente, proporcionando una alternativa flexible a la especialización mediante herencia, cuando se trata de añadir funcionalidades. En el siguiente ejemplo de código, se evidencia cómo se agrega la condición de método estático a una función a través del decorador @staticmethod.

```
@staticmethod
    def get_nombre_institucion(obj):
        local_tec = enlaces.objects.get(pk=obj.id)
        nombInst = Institucion.objects.get(pk=local_tec.institucion.id).nombre
        return nombInst
```

3.4.4 Modelo de Datos

El origen del modelo entidad-relación se encuentra en los estudios que se efectuaron por Peter Chen en la década de 1970. Este modelo se basa en la percepción del mundo real por medio de un conjunto de objetos básicos que serán las entidades y las relaciones existentes entre estos objetos (Mariscal 2015).

El modelo de datos se desarrolló a partir de la base de datos del conjunto de clases persistentes y sus asociaciones en el modelo de diseño, para su representación se utilizó la herramienta Visual Paradigm 8.0. A continuación, se presenta el diagrama entidad-relación diseñado para el sistema desarrollado, en el cual se representan las tablas creadas que constituyen los parámetros a evaluar por los especialistas. Estas entidades se encuentran relacionadas con la entidad Institución, que contiene los datos básicos de cada una de las entidades a realizar las consultorías.

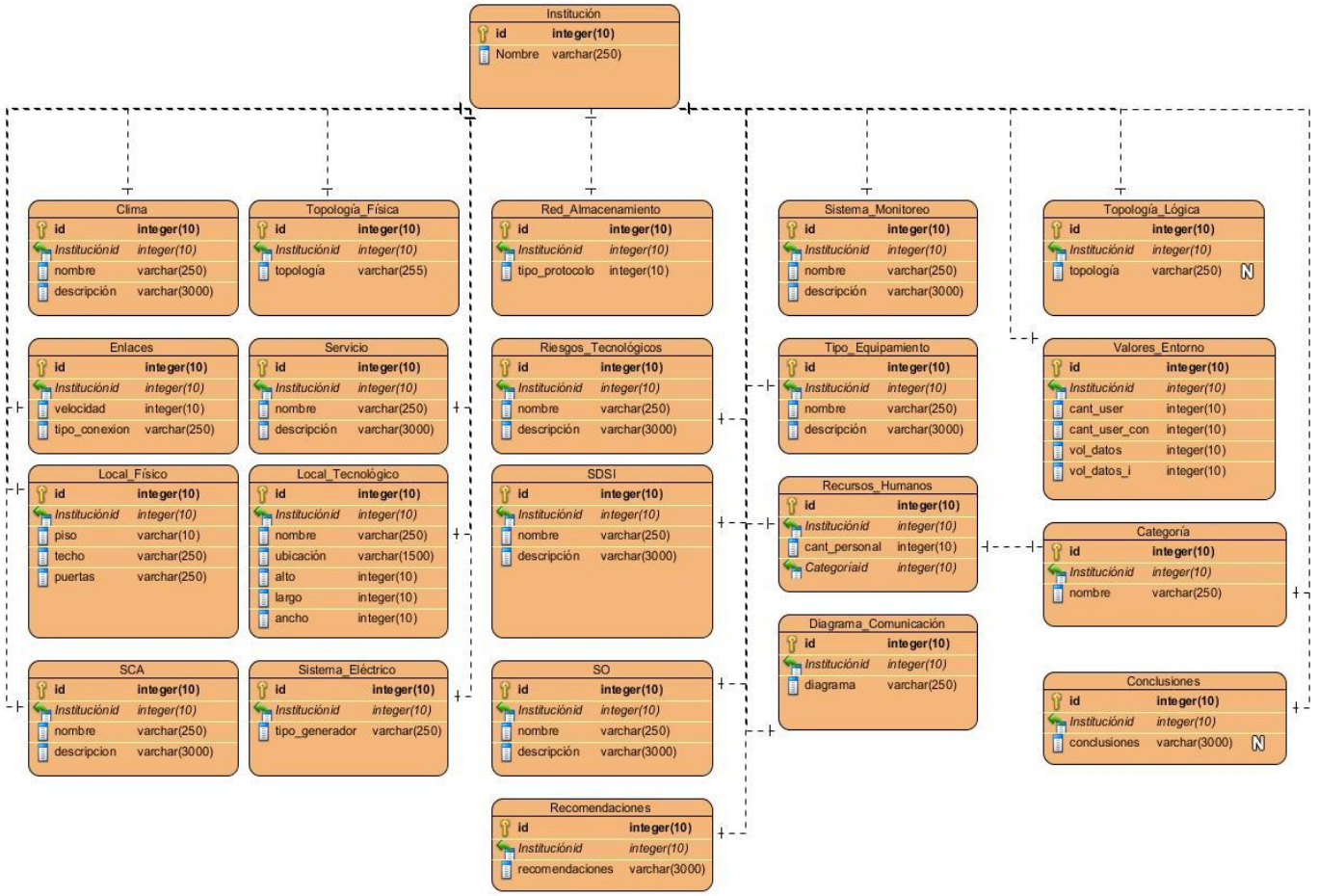


Figura 5: Diagrama Entidad Relación.

3.4.5 Tarjetas Clase- Responsabilidad- Colaboración (CRC)

En la fase de Diseño, la metodología XP no requiere la representación del sistema mediante diagramas de clases utilizando notación Lenguaje Unificado de Modelado (UML). En su lugar se usan otras técnicas como las tarjetas Clase Responsabilidad Colaboración (CRC) como una extensión informal a UML. La técnica de las tarjetas CRC se puede usar para guiar el sistema a través de análisis guiados por la responsabilidad. Las clases se examinan, se filtran y se refinan en base a sus responsabilidades con respecto al sistema, y las clases con las que necesitan colaborar para completar sus responsabilidades (Wallace, Raggett y Aufgang 2003).

A continuación, se muestran las dos primeras tarjetas CRC obtenidas durante la etapa de diseño de la presente investigación, las restantes son expuestas en el ¡Error! No se encuentra el origen de la referencia..

Tabla 6: Tarjeta CRC # 1.

Tarjetas CRC # 1	
Clase: models	
Responsabilidad	**Colaboración**
• Guardar y recuperar desde la Base de Datos la información asociada a cada uno de los modelos del sistema.	• django.db.models

Tabla 7: Tarjeta CRC # 2.

Tarjetas CRC # 2	
Clase: serializers	
Responsabilidad	**Colaboración**
• Definir la forma en que serán serializados los datos de cada uno de los modelos del sistema, tomando en cuenta el formato, los campos a serializar y las relaciones entre ellos.	• Models

3.5 Conclusiones del Capítulo

Con la realización de este capítulo, se especificaron los resultados obtenidos durante las etapas de Exploración, Planificación y Diseño al aplicar la metodología de desarrollo XP. Se identificaron 23 HU, de ellas ocho fueron catalogadas de prioridad alta, ocho de prioridad media y siete de prioridad baja. Se elaboraron los artefactos correspondientes a las etapas de Exploración, Planificación y Diseño, definiéndose así las funcionalidades que tendrá el sistema y la planificación del esfuerzo dedicado a la realización de cada una de estas en el orden en que se les dará cumplimiento según las necesidades del cliente. Además, permitió establecer una fecha estimada en la que se le entregará una primera versión del producto al cliente. Se definieron los patrones arquitectónicos y de diseño usados con el objetivo de lograr una mayor organización en los elementos que conforman la aplicación, así como las tarjetas CRC para obtener una representación de las principales clases y funcionalidades del sistema.

Capítulo 4: Implementación y pruebas

En el presente capítulo se describen los estándares de codificación utilizados para obtener un código limpio y legible. Además, se generan los artefactos correspondientes a las fases de implementación y pruebas. En la fase de implementación se exponen las tareas de ingeniería generadas por cada historia de usuario, con el propósito de facilitar su comprensión. En la fase de pruebas, se comprueba el correcto funcionamiento del software y se realizan las pruebas pertinentes para garantizar la calidad del sistema y se exponen los resultados obtenidos.

4.1 Estándares de codificación

XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación (del equipo, de la organización u otros estándares reconocidos para los lenguajes de programación utilizados). Los estándares de programación mantienen el código legible para los miembros del equipo, facilitando los cambios (Canós, Letelier y Penadés 2012).

Para lograr este objetivo, se utilizó la Guía de estilo para el código Python (PEP 8) («PEP 8 -- Style Guide for Python Code» 2018). Esta guía posee una gran cantidad de convenciones para escribir código legible, dentro de las cuales se destacan:

- Usar cuatro espacios por indentación.
- Nunca mezclar tabulaciones y espacios.
- Limitar todas las líneas a un máximo de caracteres (120 en este proyecto).
- Separar funciones de alto nivel y definiciones de clase con dos líneas en blanco, mientras que las definiciones de métodos dentro de una clase son separadas por una línea en blanco.
- Codificación UTF-8 en todos los módulos.
- Las importaciones deben estar en líneas separadas.
- Evitar espacios en blanco innecesarios.
- Utilizar el estilo CamelCase para nombrar clases, y el lower_case_with_underscores para funciones y métodos.

4.2 Desarrollo por iteraciones

Durante el transcurso de las iteraciones se realiza la implementación de las historias de usuario seleccionadas para cada una de estas y se llevan a cabo las pruebas unitarias y de aceptación asociadas a cada una de las mismas. Al iniciar cada iteración, se lleva a cabo una revisión del plan de iteraciones y se modifica de ser necesario. Como parte de este plan, se descomponen las HU en tareas de ingeniería. Estas tareas son para el uso de los programadores, pueden escribirse utilizando un lenguaje técnico y no necesariamente deben ser entendibles para el cliente (Beck 2000).

Ajustándose a la planificación realizada, se llevaron a cabo tres iteraciones de desarrollo sobre el sistema, obteniéndose al finalizar, un producto listo para su puesta en producción. A continuación, se detalla cada una de las iteraciones.

4.2.1 Iteración 1

En esta iteración se implementaron las tareas de ingeniería correspondientes a cada HU referentes a las principales funcionalidades del sistema.

HU- Gestionar el nombre de la institución.

Tabla 8: Tarea de Ingeniería - Listar institución

Tarea de Ingeniería	
Nombre: Listar institución	
HU: Gestionar el nombre de la institución	**Puntos estimados:** 0.05
Fecha de inicio: 5/02/2018	**Fecha de fin:** 5/02/2018
Descripción: El usuario luego de acceder en el módulo Administración al menú Institución, puede ver un listado con las instituciones que tiene guardada el sistema, en la que se registrará los datos (nombre), lo que permite gestionar(añadir, editar y eliminar) cada una de ellas.	

Tabla 9: Tarea de Ingeniería - Añadir institución.

Tarea de Ingeniería	
Nombre: Añadir institución	
HU: Gestionar el nombre de la institución	**Puntos estimados:** 0.05

Fecha de inicio: 5/02/2018	**Fecha de fin:** 5/02/2018
Descripción: El usuario después de acceder en el módulo Administración al menú Institución, selecciona la opción añadir. Luego el sistema muestra un formulario que le permite añadir el nombre de institución, por último, selecciona el botón “Aceptar”, el sistema lo guarda en la base de datos y muestra el mensaje: “La operación se realizó con éxito”.	

Tabla 10: Tarea de Ingeniería - Modificar institución.

Tarea de Ingeniería	
Nombre: Modificar institución	
HU: Gestionar el nombre de la institución	**Puntos estimados:** 0.05
Fecha de inicio: 5/02/2018	**Fecha de fin:** 5/02/2018
Descripción: El usuario luego de acceder en el módulo Administración al menú Institución selecciona un elemento y oprime el botón modificar. Luego el sistema muestra un formulario que permite editar el nombre de la institución. El usuario modifica los datos, el sistema los actualiza en la base de datos, y muestra el mensaje: “La operación se realizó con éxito”.	

Tabla 11: Tarea de Ingeniería - Eliminar institución.

Tarea de Ingeniería	
Nombre: Eliminar institución	
HU: Gestionar el nombre de la institución	**Puntos estimados:** 0.05
Fecha de inicio: 5/02/2018	**Fecha de fin:** 5/02/2018
Descripción: El usuario luego de acceder en el módulo Administración al menú Institución selecciona un elemento y oprime el botón eliminar. Se muestra un mensaje de confirmación ¿Está seguro que desea eliminar el elemento seleccionado? y se oprime el botón "Aceptar". El elemento se elimina de la base de datos y el sistema muestra el mensaje: “La operación se realizó con éxito”.	

HU- Gestionar locales tecnológicos.

Tabla 12: Tarea de Ingeniería - Listar locales tecnológicos.

Tarea de Ingeniería	
Nombre: Listar locales tecnológicos	
HU: Gestionar locales tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 6/02/2018	**Fecha de fin:** 6/02/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Locales Tecnológicos, puede ver un listado con los locales tecnológicos que tiene guardado el sistema, en la que se registrará los datos (nombre, ubicación, dimensión), lo que permite gestionar (añadir, editar y eliminar) cada uno de ellos.	

Tabla 13: Tarea de Ingeniería - Añadir locales tecnológicos.

Tarea de Ingeniería	
Nombre: Añadir locales tecnológicos	
HU: Gestionar locales tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 6/02/2018	**Fecha de fin:** 6/02/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Locales Tecnológicos, selecciona la opción añadir. Luego el sistema muestra un formulario que le permite añadir los elementos (nombre, ubicación y dimensión), por último, selecciona el botón “Aceptar”, el sistema lo guarda en la base de datos y muestra el mensaje: “La operación se realizó con éxito”.	

Tabla 14: Tarea de Ingeniería - Modificar locales tecnológicos.

Tarea de Ingeniería	
Nombre: Modificar locales tecnológicos	
HU: Gestionar locales tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 7/02/2018	**Fecha de fin:** 7/02/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Locales Tecnológicos, selecciona un elemento y oprime el botón modificar.	

Luego el sistema muestra un formulario que permite editar el elemento que desea modificar. El usuario modifica los datos, el sistema los actualiza en la base de datos y muestra el mensaje: “La operación se realizó con éxito”.

Tabla 15: Tarea de Ingeniería - Eliminar locales tecnológicos.

Tarea de Ingeniería	
Nombre: Eliminar locales tecnológicos	
HU: Gestionar locales tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 7/02/2018	**Fecha de fin:** 7/02/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Locales Tecnológicos, selecciona un elemento y oprime el botón eliminar. Se muestra un mensaje de confirmación ¿ Está seguro que desea eliminar el elemento seleccionado? y se oprime el botón "Aceptar". El elemento se elimina de la base de datos y el sistema muestra el mensaje: “La operación se realizó con éxito”.	

Las tareas de ingeniería pertenecientes a las HU con prioridad alta pueden verse en el **Anexo III**.

4.2.2 Iteración 2

En esta segunda iteración se desarrollaron las HU de prioridad media.

HU- Gestionar sistemas de detección y supresión de incendios.

Tabla 16: Tarea de Ingeniería - Listar sistema de detección y supresión de incendios.

Tarea de Ingeniería	
Nombre: Listar sistema de detección y supresión de incendios	
HU: Gestionar sistemas de detección y supresión de incendios	**Puntos estimados:** 0.1
Fecha de inicio: 26/02/2018	**Fecha de fin:** 26/02/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Sistemas de Detección y Supresión de Incendios puede ver un listado con los sistemas de detección que tiene guardado el sistema, en la que se registrará los	

datos (nombre, descripción), lo que permite gestionar (añadir, editar y eliminar) cada uno de ellos.

Tabla 17: Tarea de Ingeniería - Añadir sistema de detección y supresión de incendios.

Tarea de Ingeniería	
Nombre: Añadir sistema de detección y supresión de incendios	
HU: Gestionar sistemas de detección y supresión de incendios	**Puntos estimados:** 0.1
Fecha de inicio: 26/02/2018	**Fecha de fin:** 26/02/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Sistemas de Detección y Supresión de Incendios, selecciona la opción añadir. Luego el sistema muestra un formulario que le permite añadir los elementos (nombre, descripción), por último, selecciona el botón “Aceptar”, el sistema lo guarda en la base de datos y muestra el mensaje “La operación se realizó con éxito”.	

Tabla 18: Tarea de Ingeniería – Modificar sistema de detección y supresión de incendios.

Tarea de Ingeniería	
Nombre: Modificar sistema de detección y supresión de incendios	
HU: Gestionar sistemas de detección y supresión de incendios	**Puntos estimados:** 0.1
Fecha de inicio: 27/02/2018	**Fecha de fin:** 27/02/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Sistemas de Detección y Supresión de Incendios, selecciona un elemento y oprime el botón modificar. Luego el sistema muestra un formulario que permite editar el elemento que se desea modificar. El usuario modifica los datos, el sistema los actualiza en la base de datos y muestra el mensaje “La operación se realizó con éxito”.	

Tabla 19: Tarea de Ingeniería - Eliminar sistema de detección y supresión de incendios.

Tarea de Ingeniería	
Nombre: Eliminar sistema de detección y supresión de incendios	
HU: Gestionar sistemas de detección y supresión de incendios	**Puntos estimados:** 0.1
Fecha de inicio: 27/02/2018	**Fecha de fin:** 27/02/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Sistemas de Detección y Supresión de Incendios, selecciona un elemento y oprime el botón eliminar. Se muestra un mensaje de confirmación ¿Está seguro que desea eliminar el elemento seleccionado? y se oprime el botón "Aceptar". El elemento se elimina de la base de datos y el sistema muestra el mensaje: “La operación se realizó con éxito”.	

HU- Gestionar riesgos tecnológicos.

Tabla 20: Tarea de Ingeniería - Listar riesgos tecnológicos.

Tarea de Ingeniería	
Nombre: Listar riesgos tecnológicos	
HU: Gestionar riesgos tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 28/02/2018	**Fecha de fin:** 28/02/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Riesgos Tecnológicos, puede ver un listado con los riesgos que tiene guardado el sistema, en la que se registrará los datos (nombre, descripción), lo que permite gestionar (añadir, editar y eliminar) cada uno de ellos.	

Tabla 21: Tarea de Ingeniería - Añadir riesgos tecnológicos.

Tarea de Ingeniería	
Nombre: Añadir riesgos tecnológicos	
HU: Gestionar riesgos tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 28/02/2018	**Fecha de fin:** 28/02/2018

Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Riesgos Tecnológicos, selecciona la opción añadir. Luego el sistema muestra un formulario que le permite añadir los elementos (nombre, descripción), por último, selecciona el botón “Aceptar”, el sistema lo guarda en la base de datos y muestra el mensaje “La operación se realizó con éxito”.

Tabla 22: Tarea de Ingeniería – Modificar riesgos tecnológicos.

Tarea de Ingeniería	
Nombre: Modificar riesgos tecnológicos	
HU: Gestionar riesgos tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 1/03/2018	**Fecha de fin:** 1/03/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Riesgos Tecnológicos, selecciona un elemento y oprime el botón modificar. Luego el sistema muestra un formulario que permite editar el elemento que se desea modificar. El usuario modifica los datos, el sistema los actualiza en la base de datos y muestra el mensaje “La operación se realizó con éxito”.	

Tabla 23: Tarea de Ingeniería - Eliminar riesgos tecnológicos.

Tarea de Ingeniería	
Nombre: Eliminar riesgos tecnológicos	
HU: Gestionar riesgos tecnológicos	**Puntos estimados:** 0.1
Fecha de inicio: 1/03/2018	**Fecha de fin:** 1/03/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Riesgos Tecnológicos, selecciona un elemento y oprime el botón eliminar. Se muestra un mensaje de confirmación ¿Está seguro que desea eliminar el elemento seleccionado? y se oprime el botón "Aceptar". El elemento se elimina de la base de datos y el sistema muestra el mensaje: “La operación se realizó con éxito”.	

Las tareas de ingeniería pertenecientes a las HU con prioridad media pueden verse en el **Anexo IV**.

4.2.3 Iteración 3

En la tercera iteración se desarrollaron las HU de prioridad baja.

HU- Gestionar topología lógica de la red.

Tabla 24: Tarea de Ingeniería - Listar topología lógica de la red.

Tarea de Ingeniería	
Nombre: Listar topología lógica de la red	
HU: Gestionar topología lógica de la red	**Puntos estimados:** 0.1
Fecha de inicio: 19/03/2018	**Fecha de fin:** 19/03/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Topología Lógica de la Red, puede ver un listado con las topologías lógicas que están almacenadas en el sistema, en la que se registrará los datos (fichero), lo que permite gestionar (añadir, editar y eliminar) cada una de ellas.	

Tabla 25: Tarea de Ingeniería - Añadir topología lógica de la red.

Tarea de Ingeniería	
Nombre: Añadir topología lógica de la red	
HU: Gestionar topología lógica de la red	**Puntos estimados:** 0.1
Fecha de inicio: 19/03/2018	**Fecha de fin:** 19/03/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Topología Lógica de la Red, selecciona la opción añadir. Luego el sistema muestra un formulario que le permite añadir el elemento (fichero), por último, selecciona el botón “Aceptar”, el sistema lo guarda en la base de datos y muestra el mensaje “La operación se realizó con éxito”.	

Tabla 26: Tarea de Ingeniería – Modificar topología lógica de la red.

Tarea de Ingeniería	
Nombre: Modificar topología lógica de la red	
HU: Gestionar topología lógica de la red	**Puntos estimados:** 0.1
Fecha de inicio: 20/03/2018	**Fecha de fin:** 20/03/2018

Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Topología Lógica de la Red, selecciona un elemento y oprime el botón modificar. Luego el sistema muestra un formulario que permite editar el elemento que se desea modificar. El usuario modifica los datos, el sistema los actualiza en la base de datos y muestra el mensaje “La operación se realizó con éxito”.

Tabla 27: Tarea de Ingeniería - Eliminar topología lógica de la red.

Tarea de Ingeniería	
Nombre: Eliminar topología lógica de la red	
HU: Gestionar topología lógica de la red	**Puntos estimados:** 0.1
Fecha de inicio: 20/03/2018	**Fecha de fin:** 20/03/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Topología Lógica de la Red, selecciona un elemento y oprime el botón eliminar. Se muestra un mensaje de confirmación ¿Está seguro que desea eliminar el elemento seleccionado? y se oprime el botón "Aceptar". El elemento se elimina de la base de datos y el sistema muestra el mensaje: “La operación se realizó con éxito”.	

HU- Gestionar topología física de la red.

Tabla 28: Tarea de Ingeniería - Listar topología física de la red.

Tarea de Ingeniería	
Nombre: Listar topología física de la red	
HU: Gestionar topología física de la red	**Puntos estimados:** 0.1
Fecha de inicio: 21/03/2018	**Fecha de fin:** 21/03/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Topología Física de la Red, puede ver un listado con las topologías físicas de la red que están almacenadas en el sistema, en la que se registrará los datos (fichero), lo que permite gestionar (añadir, editar y eliminar) cada uno de ellos.	

Tabla 29: Tarea de Ingeniería - Añadir topología física de la red.

Tarea de Ingeniería	
Nombre: Añadir topología física de la red	
HU: Gestionar topología física de la red	**Puntos estimados:** 0.1
Fecha de inicio: 21/03/2018	**Fecha de fin:** 21/03/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Topología Física de la Red, selecciona la opción añadir. Luego el sistema muestra un formulario que le permite añadir el elemento (fichero), por último, selecciona el botón “Aceptar”, el sistema lo guarda en la base de datos y muestra el mensaje “La operación se realizó con éxito”.	

Tabla 30: Tarea de Ingeniería – Modificar topología física de la red.

Tarea de Ingeniería	
Nombre: Modificar topología física de la red	
HU: Gestionar topología física de la red	**Puntos estimados:** 0.1
Fecha de inicio: 22/03/2018	**Fecha de fin:** 22/03/2018
Descripción: El usuario después de acceder en el módulo Diagnóstico al menú Topología Física de la Red, selecciona un elemento y oprime el botón modificar. Luego el sistema muestra un formulario que permite editar el elemento que se desea modificar. El usuario modifica los datos, el sistema los actualiza en la base de datos y muestra el mensaje “La operación se realizó con éxito”.	

Tabla 31: Tarea de Ingeniería - Eliminar topología física de la red.

Tarea de Ingeniería	
Nombre: Eliminar topología física de la red	
HU: Gestionar topología física de la red	**Puntos estimados:** 0.1
Fecha de inicio: 22/03/2018	**Fecha de fin:** 22/03/2018
Descripción: El usuario luego de acceder en el módulo Diagnóstico al menú Topología Física de la Red, selecciona un elemento y oprime el botón eliminar. Se muestra un mensaje de confirmación ¿ Está seguro que desea eliminar el	

elemento seleccionado? y se oprime el botón "Aceptar". El elemento se elimina de la base de datos y el sistema muestra el mensaje: “La operación se realizó con éxito”.

Las tareas de ingeniería pertenecientes a las HU con prioridad baja pueden verse en el **Anexo V**.

4.3 Pruebas

Uno de los pilares de XP es el proceso de pruebas (Beck 2000). XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección (Crispin y House 2003).

XP divide las pruebas del sistema en dos grupos: pruebas unitarias, diseñadas por los programadores y encargadas de verificar el código, y pruebas funcionales o de aceptación destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final (Crispin y House 2003).

Durante el desarrollo de cada una de las iteraciones del presente proyecto se realizaron pruebas de caja blanca (pruebas unitarias) y de caja negra (pruebas de aceptación).

4.3.1 Pruebas Unitarias

Las pruebas unitarias son una de las piedras angulares de XP. Todos los módulos deben de pasar las pruebas unitarias antes de ser liberados o publicados. Por otra parte, las pruebas deben ser definidas antes de realizar el código. Que todo código liberado pase correctamente las pruebas unitarias es lo que habilita que funcione la propiedad colectiva del código. En este sentido, el sistema y el conjunto de pruebas debe ser guardado junto con el código, para que pueda ser utilizado por otros desarrolladores, en caso de tener que corregir, cambiar o recodificar parte del mismo (Joskowicz 2008).

Se utilizó el módulo doctest, que forma parte de la librería estándar de Python. Este módulo busca en los comentarios del código, fragmentos que se vean como sesiones del intérprete interactivo de Python, y procede a ejecutar dichos fragmentos, verificando que resulten como se les ha indicado. Esto significa, que importando el módulo doctest, éste, buscará en los comentarios de nuestro código, todo fragmento que represente al interprete interactivo, para luego ejecutarlo.

Este tipo de pruebas fue aplicada a las clases y métodos implementados durante el desarrollo de la solución propuesta. Se realizaron 22 pruebas unitarias, las cuales se ejecutaron satisfactoriamente.

En la primera iteración se realizaron ocho pruebas unitarias. Los resultados de su ejecución con el entorno integrado de desarrollo pueden observarse en la figura 4.

```
1 items had no tests:
    __main__
8 items passed all tests:
   1 tests in __main__.clima_test
   1 tests in __main__.enlaces_test
   1 tests in __main__.local_fis_test
   1 tests in __main__.locales_tec_test
   1 tests in __main__.nombre_inst_test
   1 tests in __main__.sistema_control_test
   1 tests in __main__.sistema_electrico_test
   1 tests in __main__.sistemas_monitoreo_test
8 tests in 9 items.
8 passed and 0 failed.
Test passed.

Process finished with exit code 0
```

Figura 6: Resultados de las pruebas unitarias de la primera iteración.

En la segunda iteración se realizaron ocho pruebas unitarias. Los resultados de su ejecución con el entorno integrado de desarrollo pueden observarse en la figura 5.

```
1 items had no tests:
    __main__
8 items passed all tests:
   1 tests in __main__.certificaciones_test
   1 tests in __main__.diagrama_comunicacion_test
   1 tests in __main__.red_almacenamiento_test
   1 tests in __main__.rh_test
   1 tests in __main__.riesgos_tecnologicos_test
   1 tests in __main__.servicios_test
   1 tests in __main__.sistema_deteccion_test
   1 tests in __main__.tipo_equipamiento_test
8 tests in 9 items.
8 passed and 0 failed.
Test passed.

Process finished with exit code 0
```

Figura 7: Resultados de las pruebas unitarias de la segunda iteración.

En la tercera y última iteración se realizaron seis pruebas unitarias. Los resultados de su ejecución con el entorno integrado de desarrollo pueden observarse en la figura 6.

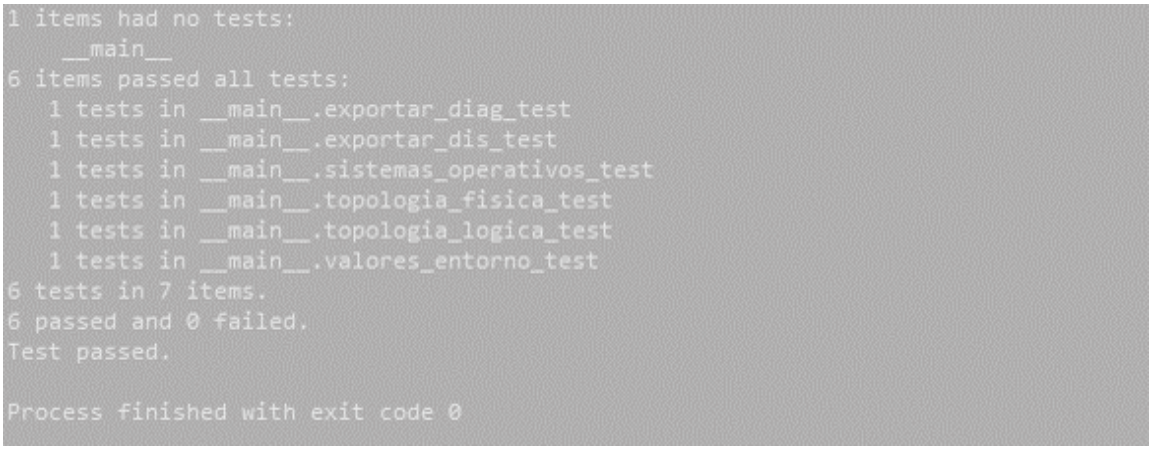


Figura 8: Resultados de las pruebas unitarias de la tercera iteración.

4.3.2 Pruebas de Aceptación

Las pruebas de aceptación son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Las pruebas de aceptación son consideradas como pruebas de caja negra. Los clientes son responsables de verificar que los resultados de éstas pruebas sean correctos. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. (Joskowicz 2008).

Para realizar las pruebas de aceptación correctamente se elaboraron casos de prueba para cada una de ellas. A continuación, se describen los casos de prueba de aceptación de las HU Gestionar nombre de institución y Gestionar locales tecnológicos, los restantes pueden ser consultados en el **Anexo VI**.

Tabla 32: Caso de Prueba de Aceptación- Listar nombre de institución – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Listar nombre de institución - Datos válidos
Historia de Usuario: Gestionar nombre de institución
Descripción: Se prueba que se listen correctamente las instituciones. Se comprueban los mensajes de notificación.

Precondiciones: El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario accede al módulo Administración, menú Institución. • Se muestra un listado de elementos. • El usuario será capaz de añadir, editar y eliminar elementos del listado.

Tabla 33: Caso de Prueba de Aceptación- Añadir nombre de institución – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Añadir nombre de institución - Datos válidos
Historia de Usuario: Gestionar nombre de institución
Descripción: Se prueba que se añadan correctamente las instituciones. Se comprueban los mensajes de notificación.
Precondiciones: El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción añadir. • Se muestra un formulario con el campo a llenar. • El usuario llena los campos y selecciona la opción “Aceptar”. • El sistema valida los datos y los guarda en la base de datos. • El sistema muestra un mensaje informando que la operación se realizó con éxito.

Tabla 34: Caso de Prueba de Aceptación- Añadir nombre de institución - Datos inválidos.

Caso de Prueba de Aceptación
Nombre: Añadir nombre de institución - Datos inválidos
Historia de Usuario: Gestionar nombre de institución

Descripción: El usuario añade datos incorrectos al sistema. Se comprueban los mensajes de notificación.
Precondiciones: El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción añadir. • Se muestra un formulario con el campo a llenar. • Se añade una nueva institución con datos incorrectos o dejando campos vacíos. • Selecciona la opción “Aceptar”. • Se muestra un mensaje de validación "Por favor inserte un valor para este campo".

Tabla 35: Caso de Prueba de Aceptación- Modificar nombre de institución – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Modificar nombre de institución - Datos válidos
Historia de Usuario: Gestionar nombre de institución
Descripción: Se prueba que se modifiquen correctamente las instituciones. Se comprueban los mensajes de notificación.
Precondiciones: El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción modificar. • Se muestra un formulario con los campos y datos a modificar. • El usuario modifica algunos o todos los campos y selecciona la opción “Aceptar”. • El sistema valida los datos y los guarda en la base de datos. • El sistema muestra un mensaje informando que la operación se realizó con éxito.

Tabla 36: Caso de Prueba de Aceptación- Modificar nombre de institución - Datos inválidos.

Caso de Prueba de Aceptación
Nombre: Modificar nombre de institución - Datos inválidos
Historia de Usuario: Gestionar nombre de institución
Descripción: El usuario modifica e introduce datos incorrectos. Se comprueban los mensajes de notificación.
Precondiciones: El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción modificar. • Se muestra un formulario con los campos y datos a modificar. • El usuario modifica algunos o todos los campos con datos incorrectos. • Selecciona la opción “Aceptar”. • Se muestra un mensaje de validación "Por favor inserte un valor para este campo".

Tabla 37: Caso de Prueba de Aceptación- Eliminar nombre de institución – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Eliminar nombre de institución - Datos válidos
Historia de Usuario: Gestionar nombre de institución
Descripción: Se prueba que se eliminen correctamente las instituciones. Se comprueban los mensajes de notificación.
Precondiciones: El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • Se selecciona la institución a eliminar.

• El usuario selecciona la opción eliminar. • Se muestra un mensaje de confirmación ¿ Está seguro que desea eliminar el elemento seleccionado? • El usuario selecciona la opción “Aceptar”. • El sistema elimina la institución de la base de datos. • El sistema muestra un mensaje informando que operación se realizó con éxito.

Tabla 38: Caso de Prueba de Aceptación- Listar locales tecnológicos – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Listar locales tecnológicos - Datos válidos
Historia de Usuario: Gestionar locales tecnológicos
Descripción: Se prueba que se listen correctamente los locales tecnológicos. Se comprueban los mensajes de notificación.
Precondiciones: Debe existir al menos una institución con la cual relacionar el local tecnológico. El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario accede al módulo Diagnóstico, menú Locales Tecnológicos. • Se muestra el listado de elementos. • El usuario será capaz de añadir, editar y eliminar elementos del listado.

Tabla 39: Caso de Prueba de Aceptación- Añadir locales tecnológicos – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Añadir locales tecnológicos - Datos válidos
Historia de Usuario: Gestionar locales tecnológicos
Descripción: Se prueba que se añadan correctamente los locales tecnológicos. Se comprueban los mensajes de notificación.

Precondiciones: Debe existir al menos una institución con la cual relacionar el local tecnológico. El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción añadir. • Se muestra un formulario con el campo a llenar. • El usuario llena los campos y selecciona la opción “Aceptar”. • El sistema valida los datos y los guarda en la base de datos. • El sistema muestra un mensaje informando que la operación se realizó con éxito.

Tabla 40: Caso de Prueba de Aceptación- Añadir locales tecnológicos - Datos inválidos.

Caso de Prueba de Aceptación
Nombre: Añadir locales tecnológicos - Datos inválidos
Historia de Usuario: Gestionar locales tecnológicos
Descripción: El usuario añade datos incorrectos al sistema. Se comprueban los mensajes de notificación.
Precondiciones: Debe existir al menos una institución con la cual relacionar el local tecnológico. El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción añadir. • Se muestra un formulario con el campo a llenar. • Se añade un nuevo local tecnológico con datos incorrectos o dejando campos vacíos. • Selecciona la opción “Aceptar”. • Se muestra un mensaje de validación "Por favor inserte un valor para este campo".

Tabla 41: Caso de Prueba de Aceptación- Modificar locales tecnológicos – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Modificar locales tecnológicos - Datos válidos
Historia de Usuario: Gestionar locales tecnológicos
Descripción: Se prueba que se modifiquen correctamente los locales tecnológicos. Se comprueban los mensajes de notificación.
Precondiciones: Debe existir al menos una institución con la cual relacionar el local tecnológico. El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción modificar. • Se muestra un formulario con los campos y datos a modificar. • El usuario modifica algunos o todos los campos y selecciona la opción “Aceptar”. • El sistema valida los datos y los guarda en la base de datos. • El sistema muestra un mensaje informando que la operación se realizó con éxito.

Tabla 42: Caso de Prueba de Aceptación- Modificar locales tecnológicos - Datos inválidos.

Caso de Prueba de Aceptación
Nombre: Modificar locales tecnológicos - Datos inválidos
Historia de Usuario: Gestionar locales tecnológicos
Descripción: El usuario modifica e introduce datos incorrectos. Se comprueban los mensajes de notificación.
Precondiciones: Debe existir al menos una institución con la cual relacionar el local tecnológico. El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • El usuario selecciona la opción modificar. • Se muestra un formulario con los campos y datos a modificar.

• El usuario modifica algunos o todos los campos con datos incorrectos. • Selecciona la opción “Aceptar”. • Se muestra un mensaje de validación "Por favor inserte un valor para este campo".

Tabla 43: Caso de Prueba de Aceptación- Eliminar locales tecnológicos – Datos válidos.

Caso de Prueba de Aceptación
Nombre: Eliminar locales tecnológicos - Datos válidos
Historia de Usuario: Gestionar locales tecnológicos
Descripción: Se prueba que se eliminen correctamente los locales tecnológicos. Se comprueban los mensajes de notificación.
Precondiciones: Debe existir al menos una institución con la cual relacionar el local tecnológico. El usuario debe estar autenticado. Se deben proveer credenciales correctas.
Pasos de ejecución: • Se selecciona el local tecnológico a eliminar. • El usuario selecciona la opción eliminar. • Se muestra un mensaje de confirmación ¿ Está seguro que desea eliminar el elemento seleccionado? • El usuario selecciona la opción “Aceptar”. • El sistema elimina el local tecnológico de la base de datos. • El sistema muestra un mensaje informando que operación se realizó con éxito.

Dado que una historia de usuario no se puede considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación y que la responsabilidad recae en todos los miembros del equipo de desarrollo, es recomendable publicar los resultados de las mismas, de manera que todo el equipo esté al tanto de esta información (Joskowicz 2008).

Las pruebas de aceptación se realizaron a las HU de la 1- 8 en una primera iteración, las que arrojaron 1 no conformidad la cual fue corregida en su totalidad. En la segunda iteración se elaboraron los casos de pruebas enfocados a las HU de la 9- 16, en estas se encontraron 2 no conformidades que fueron

resueltas. En la tercera y última iteración se realizaron los casos de pruebas correspondientes a las HU 17-22, en este caso se encontraron 6 no conformidades y todas fueron rectificadas.

Tabla 44: No Conformidades identificadas por iteraciones.

No.	No Conformidades	Número de iteración
1	No se valida el campo velocidad de la funcionalidad gestionar enlace.	1
2	El fichero de imagen para la funcionalidad diagrama de comunicación entre servidores no se carga correctamente en el sistema.	2
3	No se valida el campo cantidad de personal de la funcionalidad gestionar recursos humanos.	2
4	El fichero de imagen para la funcionalidad topología lógica de la red no se carga correctamente en el sistema.	3
5	El fichero de imagen para la funcionalidad topología física de la red no se carga correctamente en el sistema.	3
6	Al exportar el diagnóstico en formato PDF, los datos mostrados en la tabla, "Control del Documento", se encuentran desacoplados con respecto a los bordes de la tabla.	3
7	Al exportar el diseño en formato PDF, los datos mostrados en la tabla, "Control del Documento", se encuentran desacoplados con respecto a los bordes de la tabla.	3
8	No existe uniformidad en el tipo de letra de la introducción del Diagnóstico con el resto del documento.	3
9	La fecha en la que se genera el documento mostrado en ambos informes, no cuenta con un formato legible para los usuarios.	3

En el siguiente gráfico se resume los resultados de las pruebas de aceptación durante cada iteración del proceso de desarrollo de la solución propuesta.

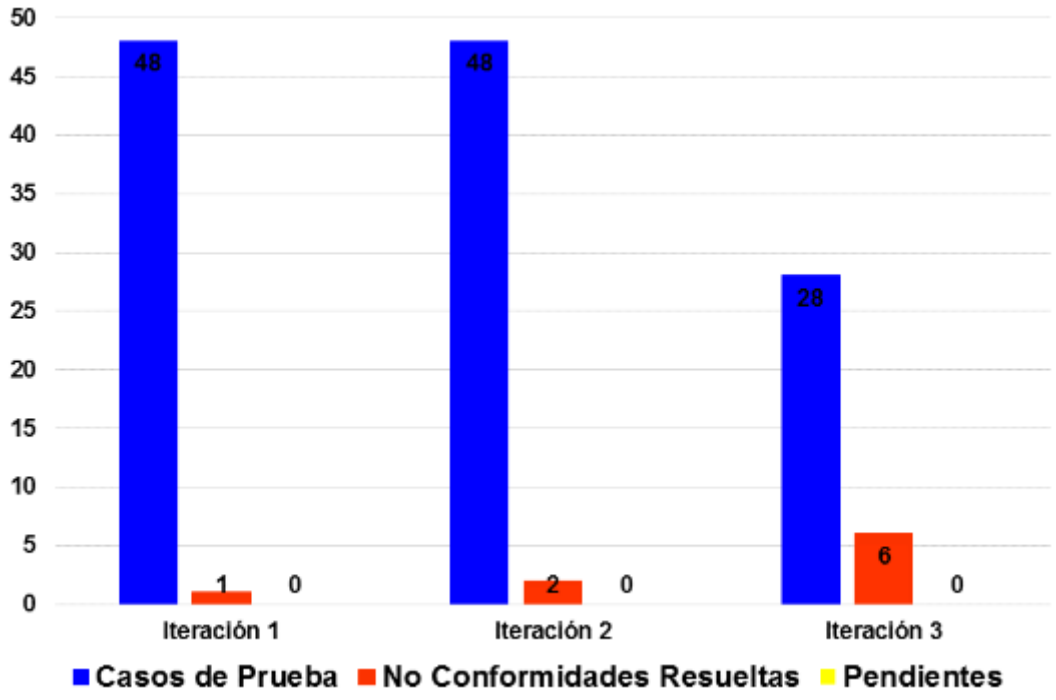


Figura 9: Comportamiento de las No Conformidades por iteraciones.

4.4 Conclusiones del capítulo

En este capítulo se llevaron a cabo las fases de implementación y pruebas planteadas por la metodología XP. Se implementaron y elaboraron las tareas de ingeniería que dieron solución a todas las HU logrando una mayor organización y rapidez en el desarrollo del sistema. Se realizaron las pruebas unitarias y de aceptación con resultados satisfactorios, lo que contribuyó a elevar la calidad final del producto, validando cada una de las funcionalidades para evitar que el sistema llegue con errores al cliente.

Conclusiones

El desarrollo del presente trabajo de diploma permitió que se llevaran a cabo todas las tareas de la investigación, a fin de dar cumplimiento al objetivo general, logrando arribar a las siguientes conclusiones:

- La elaboración del marco teórico de la investigación a través de la definición de conceptos fundamentales para su comprensión y el estudio del estado del arte realizado, permitió establecer un punto de partida de la posición actual del mundo respecto a la existencia de sistemas similares. Esto permitió al equipo de desarrollo validar la premisa inicial de la presente investigación.

- Para el desarrollo del sistema se seleccionaron herramientas y tecnologías que propiciaron el buen desempeño del equipo en el proceso de software.

- Se obtuvo un modelo guía para la implementación del sistema mediante la generación de los artefactos correspondientes a los flujos de trabajo propuestos por la metodología XP y se ha obtenido un sistema que permite automatizar la elaboración de documentos de consultorías a centros de datos, posibilitando que se cumpla satisfactoriamente el objetivo trazado para este trabajo de diploma, dando solución a la problemática planteada inicialmente por el cliente.

- La realización de las pruebas permitió validar el correcto funcionamiento del sistema, lográndose un producto final con resultados satisfactorios.

- Como resultado final se cuenta con un sistema informático que permite realizar la elaboración de documentos de consultorías a centros de datos con una mayor calidad y agilidad del proceso.

Recomendaciones

Se recomienda continuar con el desarrollo de la aplicación propuesta, agregando nuevas funcionalidades para mejorar el diseño de los documentos de diagnóstico y diseño generados a PDF, con el objetivo de establecer una mejor estructura del informe obtenido. Además, se propone desarrollar una aplicación para el sistema operativo Android para que el servicio de consultorías pueda ser accedido mediante dispositivos móviles.

Referencias Bibliográficas

1. ADC TELECOMMUNICATIONS, 2005. Norma ANSI EIA TIA 942. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: http://www.academia.edu/17915711/Norma_ANSI_EIA_TIA_942.

2. ARANGUIZ, D.V., Daisy y CLORIS, C.R., 2007. Desarrollo de un servicio de consultoría de gestión empresarial. [en línea], [Consulta: 5 marzo 2018]. ISSN 1665-8760. Disponible en: http://zaloamati.azc.uam.mx/handle/11191/287?locale-attribute=en.

3. ARÁUZ, A.S.A., 2016. *Estrategias para la implementación de TICs en la empresa Repuestos Mairena Flores, Estelí*. S.l.: s.n.

4. AREITIO, G.A. y A., 2009. *Información, Informática e Internet: del ordenador personal a la Empresa 2.0*. S.l.: Editorial Visión Libros. ISBN 9788499835259.

5. ARIAS, M.A., 2014. *Webs Responsivas. Responsive Design con Bootstrap*. S.l.: IT Campus Academy. ISBN 9781495492099.

6. BECK, K., 2000. *Extreme Programming Explained: Embrace Change*. S.l.: Addison-Wesley Professional. ISBN 9780201616415.

7. BIOMUNDI, 2017. BioMundi. [en línea]. [Consulta: 22 mayo 2018]. Disponible en: http://www.biomundi.inf.cu/.

8. CANÓS, J.H., LETELIER, P. y PENADÉS, M.C., 2012. Metodologías ágiles en el desarrollo de software. [en línea], [Consulta: 28 abril 2018]. Disponible en: https://scholar.google.es/citations?user=MIML7qMAAAAJ&hl=es.

9. CORNEJO, J.E.G., 2008. El Lenguaje de Modelado Unificado (UML). [en línea]. [Consulta: 3 marzo 2018]. Disponible en: http://www.docirs.com/uml.htm.

10. CRISPIN, L. y HOUSE, T., 2003. *Testing Extreme Programming*. S.l.: Addison-Wesley Professional. ISBN 9780321113559.

11. DJANGO, 2018. The Web framework for perfectionists with deadlines | Django. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: https://www.djangoproject.com/.

12. DUQUE, R.G., 2010. *Python para todos*. S.l.: s.n.

13. DURANGO, A., 2015. *Diseño de Software: 2ª Edición*. S.l.: IT Campus Academy. ISBN 9781519620736.

14. FLANAGAN, D., 2011. *JavaScript: The Definitive Guide, 6th Edition*. S.l.: O'Reilly Media, Inc. ISBN 9780596805524.

15. GALVAN, V.G., 2013. *DATACENTER - UNA MIRADA POR DENTRO*. S.l.: s.n. ISBN 9789873334672.

16. GUERRERO, C.A., SUÁREZ, J.M. y GUTIÉRREZ, L.E., 2013. Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web.

Información tecnológica, vol. 24, no. 3, pp. 103-114. ISSN 0718-0764. DOI 10.4067/S0718-07642013000300012.

17. INFRAESTRUCTURA TECNOLÓGICA DE SOFTWARE, 2018. Infraestructura Tecnológica de Software. *FUNIBER* [en línea]. [Consulta: 13 junio 2018]. Disponible en: https://www.funiber.org/infraestructura-tecnologica-de-software.

18. INTERNATIONAL LABOUR OFFICE, 2002. *Management Consulting: A Guide to the Profession*. S.l.: International Labour Organization. ISBN 9789221095194.

19. ISLAM, Q.N., 2015. *Mastering PyCharm*. S.l.: Packt Publishing Ltd. ISBN 9781783551323.

20. JETBRAINS, 2018. PyCharm: Python IDE for Professional Developers by JetBrains. *JetBrains* [en línea]. [Consulta: 10 abril 2018]. Disponible en: https://www.jetbrains.com/pycharm/.

21. JOSKOWICZ, J., 2008. Reglas y Prácticas en eXtreme Programming. [en línea], [Consulta: 18 marzo 2018]. Disponible en: https://www.researchgate.net/publication/267954657.

22. LARMAN, C., 2003. *UML y patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado*. S.l.: Pearson Educación. ISBN 9788420534381.

23. LUTZ, M., 2013. *Learning Python* [en línea]. S.l.: s.n. [Consulta: 10 abril 2018]. ISBN 9781449355739. Disponible en: http://shop.oreilly.com/product/0636920028154.do.

24. MACDONALD, M., 2013. *HTML5: The Missing Manual, 2nd Edition*. S.l.: O'Reilly Media. ISBN 9781449363260.

25. MARISCAL, A.B.G., 2015. *UF2405 - Modelo de programación web y bases de datos*. S.l.: Editorial Elearning, S.L.

26. MEDINA, J.D.T., 2013. *Sistema de registro y control de actividades en servicios de consultorías para la empresa MASTERTICS S.A.S*. SANTIAGO DE CALI: UNIVERSIDAD AUTÓNOMA DE OCCIDENTE.

27. PEP 8 -- Style Guide for Python Code. *Python.org* [en línea], 2018. [Consulta: 18 marzo 2018]. Disponible en: https://www.python.org/dev/peps/pep-0008/.

28. PGADMIN, 2018. Introduction — pgAdmin III 1.22.2 documentation. [en línea]. [Consulta: 6 abril 2018]. Disponible en: https://www.pgadmin.org/docs/pgadmin3/1.22/introduction.html.

29. PORTO, J.P. y GARDEY, A., 2018. Definición de data center — Definición.de. En: Diccionario Enciclopédico, *Definición.de* [en línea]. [Consulta: 11 junio 2018]. Disponible en: https://definicion.de/data-center/.

30. POSTGRESQL, 2018. PostgreSQL: The world's most advanced open source database. [en línea]. [Consulta: 4 marzo 2018]. Disponible en: https://www.postgresql.org/.

31. QUINTANA, L.L., 2017. *Procedimiento de consultoría para implementar los procesos contables en organizaciones empresariales.* [en línea]. S.l.: Departamento de Contabilidad y Finanzas. [Consulta: 8 abril 2018]. ISBN 9789591632944. Disponible en: http://rediuc.reduc.edu.cu:8080/jspui/handle/123456789/1121.

32. REPORTLAB, 2018. ReportLab PDF Generation User Guide - reportlab-userguide.pdf. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: https://www.reportlab.com/docs/reportlab-userguide.pdf.

33. RIOS, R.R., 2018. Capítulo 5: Diseño y Arquitectura del Sistema. [en línea], [Consulta: 4 marzo 2018]. Disponible en: http://www.academia.edu/28009487/Cap%C3%ADtulo_5_Dise%C3%B1o_y_Arquitectura_del_Sistema.

34. SCHNEIDER ELECTRIC, 2018. Software para Data Center | Schneider Electric. [en línea]. [Consulta: 5 marzo 2018]. Disponible en: https://www.schneider-electric.co.cr/es/product-category/88438-software-para-data-center/.

35. SUNBIRD, 2018. Sunbird | Data Center Management. [en línea]. [Consulta: 5 marzo 2018]. Disponible en: http://info.sunbirddcim.com/data-center-management.

36. TIA-942 STANDARD, 2014. TIA-942 Certified Data Centers | Consultants | Auditors. [en línea]. [Consulta: 15 junio 2018]. Disponible en: http://www.tia-942.org/.

37. TLM, 2016. *Xilema-Base-Web. Wiki TLM*. 2016. S.l.: s.n.

38. UCI, 2018. Centro de Telemática (TLM) | Universidad de las Ciencias Informáticas. [en línea]. [Consulta: 10 abril 2018]. Disponible en: https://www.uci.cu/investigacion-y-desarrollo/centros-de-desarrollo/centro-de-telematica-tlm.

39. VISUAL PARADIGM, 2014. Tutorial Visual Paradigm - PracticaVP.pdf. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: http://www.ie.inf.uc3m.es/grupo/docencia/reglada/Is1y2/PracticaVP.pdf.

40. WALLACE, D., RAGGETT, I. y AUFGANG, J., 2003. *Extreme Programming for Web Projects*. S.l.: Addison-Wesley Professional. ISBN 9780201794274.

Bibliografía

1. ADC TELECOMMUNICATIONS, 2005. Norma ANSI EIA TIA 942. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: http://www.academia.edu/17915711/Norma_ANSI_EIA_TIA_942.

2. ARANGUIZ, D.V., Daisy y CLORIS, C.R., 2007. Desarrollo de un servicio de consultoría de gestión empresarial. [en línea], [Consulta: 5 marzo 2018]. ISSN 1665-8760. Disponible en: http://zaloamati.azc.uam.mx/handle/11191/287?locale-attribute=en.

3. ARÁUZ, A.S.A., 2016. *Estrategias para la implementación de TICs en la empresa Repuestos Mairena Flores, Estelí*. S.l.: s.n.

4. AREITIO, G.A. y A., 2009. *Información, Informática e Internet: del ordenador personal a la Empresa 2.0*. S.l.: Editorial Visión Libros. ISBN 9788499835259.

5. ARIAS, M.A., 2014. *Webs Responsivas. Responsive Design con Bootstrap*. S.l.: IT Campus Academy. ISBN 9781495492099.

6. BECK, K., 2000. *Extreme Programming Explained: Embrace Change*. S.l.: Addison-Wesley Professional. ISBN 9780201616415.

7. BIOMUNDI, 2017. BioMundi. [en línea]. [Consulta: 22 mayo 2018]. Disponible en: http://www.biomundi.inf.cu/.

8. Book Review -- JavaScript: The Definitive Guide, 6th Edition - Slashdot. [en línea], [sin fecha]. [Consulta: 9 abril 2018]. Disponible en: https://books.slashdot.org/story/11/06/02/1511219/book-review----javascript-the-definitive-guide-6th-edition.

9. CANÓS, J.H., LETELIER, P. y PENADÉS, M.C., 2012. Metodologías ágiles en el desarrollo de software. [en línea], [Consulta: 28 abril 2018]. Disponible en: https://scholar.google.es/citations?user=MIML7qMAAAAJ&hl=es.

10. CORNEJO, J.E.G., 2008. El Lenguaje de Modelado Unificado (UML). [en línea]. [Consulta: 3 marzo 2018]. Disponible en: http://www.docirs.com/uml.htm.

11. CRISPIN, L. y HOUSE, T., 2003. *Testing Extreme Programming*. S.l.: Addison-Wesley Professional. ISBN 9780321113559.

12. DJANGO, 2018. The Web framework for perfectionists with deadlines | Django. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: https://www.djangoproject.com/.

13. DUQUE, R.G., 2010. *Python para todos*. S.l.: s.n.

14. DURANGO, A., 2015. *Diseño de Software: 2ª Edición*. S.l.: IT Campus Academy. ISBN 9781519620736.

15. FLANAGAN, D., 2011. *JavaScript: The Definitive Guide, 6th Edition*. S.l.: O'Reilly Media, Inc. ISBN 9780596805524.

16. GALVAN, V.G., 2013. *DATACENTER - UNA MIRADA POR DENTRO*. S.l.: s.n. ISBN 9789873334672.

17. GUERRERO, C.A., SUÁREZ, J.M. y GUTIÉRREZ, L.E., 2013. Patrones de Diseño GOF (The Gang of Four) en el contexto de Procesos de Desarrollo de Aplicaciones Orientadas a la Web. *Información tecnológica*, vol. 24, no. 3, pp. 103-114. ISSN 0718-0764. DOI 10.4067/S0718-07642013000300012.

18. GUIDO VAN ROSSUM, 2009. *El tutorial de Python* [en línea]. S.l.: s.n. [Consulta: 3 marzo 2018]. Disponible en: http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf.

19. INFRAESTRUCTURA TECNOLÓGICA DE SOFTWARE, 2018. Infraestructura Tecnológica de Software. *FUNIBER* [en línea]. [Consulta: 13 junio 2018]. Disponible en: https://www.funiber.org/infraestructura-tecnologica-de-software.

20. INTERNATIONAL LABOUR OFFICE, 2002. *Management Consulting: A Guide to the Profession*. S.l.: International Labour Organization. ISBN 9789221095194.

21. ISLAM, Q.N., 2015. *Mastering PyCharm*. S.l.: Packt Publishing Ltd. ISBN 9781783551323.

22. JETBRAINS, 2018. PyCharm: Python IDE for Professional Developers by JetBrains. *JetBrains* [en línea]. [Consulta: 10 abril 2018]. Disponible en: https://www.jetbrains.com/pycharm/.

23. JOSKOWICZ, J., 2008. Reglas y Prácticas en eXtreme Programming. [en línea], [Consulta: 18 marzo 2018]. Disponible en: https://www.researchgate.net/publication/267954657.

24. LARMAN, C., 2003. *UML y patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado*. S.l.: Pearson Educación. ISBN 9788420534381.

25. LUTZ, M., 2013. *Learning Python* [en línea]. S.l.: s.n. [Consulta: 10 abril 2018]. ISBN 9781449355739. Disponible en: http://shop.oreilly.com/product/0636920028154.do.

26. MACDONALD, M., 2013. *HTML5: The Missing Manual, 2nd Edition*. S.l.: O'Reilly Media. ISBN 9781449363260.

27. MARISCAL, A.B.G., 2015. *UF2405 - Modelo de programación web y bases de datos*. S.l.: Editorial Elearning, S.L.

28. MEDINA, J.D.T., 2013. *Sistema de registro y control de actividades en servicios de consultorías para la empresa MASTERTICS S.A.S*. SANTIAGO DE CALI: UNIVERSIDAD AUTÓNOMA DE OCCIDENTE.

29. PEP 8 -- Style Guide for Python Code. *Python.org* [en línea], 2018. [Consulta: 18 marzo 2018]. Disponible en: https://www.python.org/dev/peps/pep-0008/.

30. PGADMIN, 2018. Introduction — pgAdmin III 1.22.2 documentation. [en línea]. [Consulta: 6 abril 2018]. Disponible en: https://www.pgadmin.org/docs/pgadmin3/1.22/introduction.html.

31. PORTO, J.P. y GARDEY, A., 2018. Definición de data center — Definición.de. En: Diccionario Enciclopédico, *Definición.de* [en línea]. [Consulta: 11 junio 2018]. Disponible en: https://definicion.de/data-center/.

32. POSTGRESQL, 2018. PostgreSQL: The world's most advanced open source database. [en línea]. [Consulta: 4 marzo 2018]. Disponible en: https://www.postgresql.org/.

33. QUINTANA, L.L., 2017. *Procedimiento de consultoría para implementar los procesos contables en organizaciones empresariales.* [en línea]. S.l.: Departamento de Contabilidad y Finanzas. [Consulta: 8 abril 2018]. ISBN 9789591632944. Disponible en: http://rediuc.reduc.edu.cu:8080/jspui/handle/123456789/1121.

34. REPORTLAB, 2018. ReportLab PDF Generation User Guide - reportlab-userguide.pdf. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: https://www.reportlab.com/docs/reportlab-userguide.pdf.

35. RIOS, R.R., 2018. Capítulo 5: Diseño y Arquitectura del Sistema. [en línea], [Consulta: 4 marzo 2018]. Disponible en: http://www.academia.edu/28009487/Cap%C3%ADtulo_5_Dise%C3%B1o_y_Arquitectura_del_Sistema.

36. SCHNEIDER ELECTRIC, 2018. Software para Data Center | Schneider Electric. [en línea]. [Consulta: 5 marzo 2018]. Disponible en: https://www.schneider-electric.co.cr/es/product-category/88438-software-para-data-center/.

37. SUNBIRD, 2018. Sunbird | Data Center Management. [en línea]. [Consulta: 5 marzo 2018]. Disponible en: http://info.sunbirddcim.com/data-center-management.

38. TIA-942 STANDARD, 2014. TIA-942 Certified Data Centers | Consultants | Auditors. [en línea]. [Consulta: 15 junio 2018]. Disponible en: http://www.tia-942.org/.

39. TLM, 2016. *Xilema-Base-Web. Wiki TLM*. 2016. S.l.: s.n.

40. UCI, 2018. Centro de Telemática (TLM) | Universidad de las Ciencias Informáticas. [en línea]. [Consulta: 10 abril 2018]. Disponible en: https://www.uci.cu/investigacion-y-desarrollo/centros-de-desarrollo/centro-de-telematica-tlm.

41. VISUAL PARADIGM, 2014. Tutorial Visual Paradigm - PracticaVP.pdf. [en línea]. [Consulta: 3 marzo 2018]. Disponible en: http://www.ie.inf.uc3m.es/grupo/docencia/reglada/Is1y2/PracticaVP.pdf.

42. WALLACE, D., RAGGETT, I. y AUFGANG, J., 2003. *Extreme Programming for Web Projects*. S.l.: Addison-Wesley Professional. ISBN 9780201794274.