



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E S C I E N T I A H O M I N I S S A L U S "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**MIGRACIÓN DEL SOFTWARE DE PLANIFICACIÓN ESTRATÉGICA
DE LA EMPRESA GRUPO NOVATECH CÍA. LTDA. A UN
AMBIENTE CLOUD USANDO EL MODELO DE DISTRIBUCIÓN
SAAS**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN ELECTRÓNICA Y REDES DE INFORMACIÓN**

SEGUNDO GONZALO JIMÉNEZ AQUÍÑO

segundogjimenez@yahoo.com.mx

DIRECTORA: ANA MARÍA ZAMBRANO VIZUETE PhD.

ana.zambrano@epn.edu.ec

CODIRECTOR: XAVIER ALEXANDER CALDERÓN HINOJOSA MSc.

xavier.calderon@epn.edu.ec

Quito, agosto 2018

AVAL

Certificamos que el presente trabajo fue desarrollado por Segundo Gonzalo Jiménez Aquíño, bajo nuestra supervisión.

**ANA ZAMBRANO PhD.
DIRECTORA DEL TRABAJO DE TITULACIÓN**

**XAVIER CALDERÓN MSc.
CODIRECTOR DEL TRABAJO DE TITULACIÓN**

DECLARACIÓN DE AUTORÍA

Yo, Segundo Gonzalo Jiménez Aquino, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

SEGUNDO GONZALO JIMÉNEZ AQUINO

DEDICATORIA

Quiero dedicar este logro a aquellas personas que, por ese misterio que conlleva la vida, fueron parte de mí y ahora no están conmigo.

A mis familiares amados que partieron de este mundo, a mis grandes amigos que ya no están, a aquellas personas que compartieron su conocimiento desinteresadamente, a quienes supieron comprenderme y quererme, a aquellos que fueron tan admirados por mí pero que tomaron rumbos diferentes, a aquellos que encontraron un talento dentro de mí, incluso a aquellos fantasmas que fueron un obstáculo gigante y que más nunca querré volver a ver.

Es un poco confuso dedicarle a alguien que sabes que no lo leerá. Este pequeño trabajo queda como constancia para todas esas personas que fueron muy importantes en su momento y hoy son un bello recuerdo en mi memoria.

Segundo Gonzalo Jiménez Aquíño

“... es que este mundo sin amor es un mundo muerto, y que al fin llega un momento en que se cansa uno de la prisión, del trabajo y del valor, y no exige más que el rostro de un ser y el hechizo de la ternura en el corazón”.

Albert Camus

AGRADECIMIENTO

Todo lo que he logrado en mi vida, lo he podido realizar por tener tanta gente buena a mi lado.

La mujer que es mi motivación para el diario vivir, mi esposa Ánis, quien creyó en mí y supo darme las fuerzas y el amor para seguir adelante, incluso en los momentos en que estuve a punto de darme por vencido. Te amo con mi vida completa.

La familia tan hermosa que tengo: mi papá Manuel es un ejemplo de rectitud y honestidad, mi mamá Betty es tan sabia y paciente no solo conmigo sino con todas las personas, mi hermana Karla que siempre ha sabido cuidarme y darme valor en los momentos difíciles, mi hermano José que es mi alegría desde que me levanto hasta cuando cenamos juntos. Todo su amor ha sido un pilar fundamental en este logro alcanzado.

Los profesores que me han guiado con sus palabras y sobre todo, con su ejemplo: a la Doctora Anita Zambrano por haberme dado confianza en mi propio trabajo y por darme tantas lecciones académicas y también en el ámbito profesional; al Ing. Xavier Calderón por haberme dado una mano cuando no lograba encontrar un director, fue, a lo mejor sin saberlo, una gran ayuda en aquel momento de desesperanza; al Ing. Tarquino Sánchez que supo darme la oportunidad de finalizar este trabajo buscando constantemente la forma en cómo ayudar a todos los alumnos; al Ing. Pablo Hidalgo por las palabras alentadoras hacia mí incluso cuando solo le daba dolores de cabeza. Hay muchos profesores en la E.P.N. que son muy valiosos.

Los colegas de la empresa GRUPO NOVATECH CÍA. LTDA. quienes me apoyaron desde el primer momento sin dudarlo: al Ing. Jorge Barahona que puso a disposición las herramientas necesarias para desarrollar este trabajo; y al Ing. Mario Morales que me dio las pautas de cómo debería plantear el tema. Ambos son un apoyo para mi carrera profesional y, además, han sabido brindarme su amistad sincera.

Para mí es una bendición de Dios el contar con todos ustedes.

Segundo Gonzalo Jiménez Aquino

“Somos enanos – admitió Guillermo –, pero enanos subidos sobre los hombros de aquellos gigantes y, aunque pequeños, a veces logramos ver más allá de su horizonte”.

Umberto Eco

ÍNDICE DE CONTENIDO

AVAL.....	i
DECLARACIÓN DE AUTORÍA	ii
DEDICATORIA	iii
AGRADECIMIENTO	iv
ÍNDICE DE CONTENIDO.....	vi
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS	xii
LISTA DE SIGLAS	xiv
RESUMEN	xv
ABSTRACT.....	xvi
1. INTRODUCCIÓN	1
1.1 Objetivos.....	1
1.2 Alcance	1
1.3 Marco Teórico.....	3
1.4 Planificación Estratégica	4
1.4.1 Concepto	4
1.4.2 Objetivos Estratégicos	4
1.4.3 Indicadores y Cuadros de Mando Integrales.....	4
1.5 Empresa GRUPO NOVATECH CÍA. LTDA.	5
1.5.1 Situación Actual.....	5
1.6 Cloud Computing	6
1.6.1 Características Esenciales de Cloud Computing	6
1.6.2 Modelos de Despliegue	8
1.6.3 Modelos de Servicios.....	9
1.6.4 Riesgos en Cloud Computing	14
1.6.5 Proceso de Migración.....	16

1.6.6	Legislación en el Ecuador	17
1.7	Acuerdo de Nivel de Servicio	22
1.7.1	Lineamientos Principales de un SLA	22
1.8	Metodología SCRUM	23
1.9	Inteligencia de Negocios	24
1.9.1	Plataforma QLIK	24
2.	METODOLOGÍA.....	26
2.1	Diseño.....	26
2.1.1	Definición del Backlog	26
2.1.2	Sprint de Requerimientos	28
2.1.3	Sprint de Selección de Proveedor IaaS	31
2.1.4	Sprint del Flujo de Información	37
2.1.5	Sprint de Diseño de la Capa de Base de Datos	40
2.1.6	Sprint de Diseño de la Capa de Negocio	42
2.1.7	Sprint de Diseño de la Capa de Presentación.....	49
2.1.8	Componentes del Sistema.....	52
2.2	Implementación.....	54
2.2.1	Sprint de Instalación y Configuración de Plataforma IaaS	54
2.2.2	Capa de Base de Datos.....	56
2.2.3	Capa de Negocio.....	58
2.2.4	Capa de Presentación	61
2.2.5	Sprint de Redacción del SLA	66
3.	RESULTADOS Y DISCUSIÓN	67
3.1	Pruebas de Funcionamiento del IaaS Contratado	67
3.1.1	Pruebas de Acceso desde la Perspectiva Cliente.....	67
3.1.2	Pruebas de Acceso desde la Perspectiva Proveedor.....	68
3.2	Pruebas de Funcionamiento de la Capa de Base de Datos	69

3.3	Pruebas de Funcionamiento de la Capa de Negocio.....	70
3.3.1	Pruebas de Funcionamiento del Administrador.....	70
3.3.2	Pruebas de Funcionamiento del Facilitador	73
3.3.3	Pruebas de Funcionamiento del Colaborador	76
3.3.4	Pruebas de Funcionamiento del Gestor.....	76
3.4	Pruebas de Funcionamiento de la Capa de Presentación	77
3.5	Pruebas de Funcionamiento del Flujo de Datos	79
3.6	Resumen	79
4.	CONCLUSIONES.....	81
4.1	Conclusiones	83
4.2	Recomendaciones	85
5.	REFERENCIAS BIBLIOGRÁFICAS.....	87
6.	ANEXOS	93
	ORDEN DE EMPASTADO.....	94

ÍNDICE DE FIGURAS

Figura 1.1. Arquitectura actual Cliente - Servidor	2
Figura 1.2. Diagrama lógico del prototipo a diseñarse.....	3
Figura 1.3. Modelos de despliegue	8
Figura 1.4. Niveles de control en IaaS	9
Figura 1.5. Niveles de control en PaaS	11
Figura 1.6. Niveles de control en SaaS	12
Figura 1.7. Niveles de control en los servicios <i>cloud</i>	15
Figura 2.1. Diagrama de secuencia del flujo de datos	38
Figura 2.2. Diagrama de secuencia de ingreso y análisis de datos	38
Figura 2.3. Diagrama entidad – relación de la capa de base de datos	41
Figura 2.4. Caso de uso del Administrador	42
Figura 2.5. Caso de uso del Facilitador.....	43
Figura 2.6. Caso de uso del Colaborador.....	44
Figura 2.7. Caso de uso del perfil Gestor	44
Figura 2.8. Nivel de base de datos del diagrama de clases	45
Figura 2.9. Diagrama de Clases.....	46
Figura 2.10. Nivel de lógica de tablas del diagrama de clases	47
Figura 2.11. Nivel de interfaz web del diagrama de clases.....	48
Figura 2.12. Etapas de desarrollo del CMI	50
Figura 2.13. Nube de datos del CMI.....	52
Figura 2.14. Panel de control de Azure	54
Figura 2.15. Máquina virtual <i>D2S_V3</i> escogida.....	55
Figura 2.16. Máquina virtual creada	55
Figura 2.17. Creación del usuario administrador	56
Figura 2.18. Acceso a la base de datos	57

Figura 2.19. Codificación de la tabla <i>ARBOL_INDICADOR</i> en lenguaje <i>SQL</i>	57
Figura 2.20. Configuración del puerto de acceso 81 para el acceso	58
Figura 2.21. Habilitación de la extensión de <i>PHP</i> para <i>SQL</i> en la consola del servidor web	59
Figura 2.22. Reglas de acceso a la máquina virtual en el portal de Azure.....	59
Figura 2.23. Creación de la regla de ingreso para puerto 81 en <i>firewall</i> de máquina virtual	60
Figura 2.24. Codificación de la clase <i>Formula</i> en lenguaje <i>PHP</i>	60
Figura 2.25. Extracto de la codificación del menú de Fórmulas en la interfaz web.....	62
Figura 2.26. Configuración de puertos de acceso para el CMI	63
Figura 2.27. Asignación de direcciones IP permitidas	63
Figura 2.28. Creación de regla de acceso del <i>firewall</i> de máquina virtual	64
Figura 2.29. Habilitación de puertos de acceso al CMI en el panel de control de <i>Azure</i>	64
Figura 2.30. Conexión exitosa a la base de datos por ODBC.....	65
Figura 2.31. Creación de librería de conexión en <i>Qlik</i>	65
Figura 2.32. Codificación de la tabla <i>dm_Indicadores</i> en <i>Qlik</i>	66
Figura 3.1. Acceso al servidor web desde un navegador web externo	67
Figura 3.2. Acceso a la plataforma <i>Qlik</i> desde un navegador web externo	68
Figura 3.3. Acceso a la máquina virtual por medio de la herramienta de acceso remoto de <i>Microsoft</i>	69
Figura 3.4. Consulta a la tabla <i>Usuario</i>	69
Figura 3.5. Base de datos creada	70
Figura 3.6. Creación de nuevo usuario	71
Figura 3.7. Ingreso de parámetros iniciales de indicadores.....	71
Figura 3.8. Creación de un objetivo.....	72

Figura 3.9. Nuevo indicador añadido.....	72
Figura 3.10. Extensión de plazo para un indicador.....	72
Figura 3.11. Notificación de sistema	73
Figura 3.12. Ingreso de valores mensuales.....	73
Figura 3.13. Formulario para crear de un nuevo plan de acción.....	74
Figura 3.14. Formulario para crear de una nueva tarea	75
Figura 3.15. Formulario para crear de un nuevo avance de una tarea	76
Figura 3.16. Finalización de tarea	76
Figura 3.17. Aprobación de valores reportados.....	76
Figura 3.18. Etapas de desarrollo del CMI en <i>Qlik</i>	77
Figura 3.19. Tabla de cálculo del cumplimiento de la PE	78
Figura 3.20. Tareas de recarga de datos en la plataforma <i>Qlik</i>	79

ÍNDICE DE TABLAS

Tabla 1.1. Características esenciales de <i>Cloud Computing</i>	7
Tabla 1.2. Comparativo entre modelos de servicio.....	14
Tabla 1.3. Artículos citados de la Constitución	18
Tabla 1.4. Artículos citados de la Ley Orgánica de Transparencia y Acceso a la Información Pública.....	19
Tabla 1.5. Artículos citados de la Ley del Sistema Nacional de Registro de Datos Públicos.....	20
Tabla 1.6. Artículos citados de la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos	21
Tabla 1.7. Lineamientos principales de un SLA.....	22
Tabla 2.1. Valoración de los requerimientos	27
Tabla 2.2. Definición del <i>backlog</i>	27
Tabla 2.3. Historia de usuario de acceso ubicuo	28
Tabla 2.4. Historia de usuario de perfiles de usuario.....	29
Tabla 2.5. Historia de usuario de perfil Administrador	29
Tabla 2.6. Historia de usuario de perfil Facilitador.....	30
Tabla 2.7. Historia de usuario de perfil Colaborador	30
Tabla 2.8. Historia de usuario de perfil Gestor	31
Tabla 2.9. Historia de usuario de instalación única	32
Tabla 2.10. Historia de usuario de ambiente IaaS.....	32
Tabla 2.11. Historia de usuario de proyecto demostrativo	33
Tabla 2.12. Historia de usuario de administración del cliente	33
Tabla 2.13. Historia de usuario de actualización de capa presentación	34
Tabla 2.14. Recursos mínimos para la instancia virtual del proveedor IaaS.....	34
Tabla 2.15. Comparativo técnico entre proveedores IaaS.....	35
Tabla 2.16. Comparativo funcional entre los proveedores IaaS	36

Tabla 3.1. Tabla de resumen de cumplimiento de requerimientos80

Tabla 3.2. Tabla de resumen de los *Sprints*80

LISTA DE SIGLAS

PE:	Planificación Estratégica
CEPAL:	Comisión Económica para América Latina y el Caribe
OE:	Objetivos Estratégicos
KPI:	<i>Key Performance Indicators</i>
CMI:	Cuadro de Mando Integral
TI:	Tecnologías de la información
NIST:	<i>National Institute of Standards and Technology</i>
AWS:	<i>Amazon Web Services</i>
MA:	<i>Microsoft Azure</i>
GCE:	<i>Google Cloud Engine</i>
IaaS:	<i>Infrastructure as a Service</i>
PaaS:	<i>Platform as Service</i>
SaaS:	<i>Software as Service</i>
IJCSITCE:	<i>International Journal of Computational Science, Information Technology and Control Engineering</i>
SLA:	<i>Service Level Agreement</i>
ONG:	Organización No Gubernamental
BI:	<i>Business Intelligence</i>
QAM:	<i>Qlik Associative Model</i>
ID:	Identificador
EBS:	<i>Elastic Book Store</i>

RESUMEN

El presente trabajo tiene como objetivo el realizar una migración del software de planificación estratégica (PE) que ofrece la empresa GRUPO NOVATECH CÍA. LTDA., desde una arquitectura Cliente – Servidor hacia un modelo de distribución *Software as a Service* (SaaS).

La tecnología del *cloud computing* es una solución que provee ventajas significativas a los servicios de las empresas. Estas ventajas permiten una mejor gestión, lo que se refleja en un mejor servicio prestado a sus clientes.

El presente trabajo se lo va a dividir en tres capas: la capa de base de datos que será el repositorio central de toda la información del presente trabajo, la capa de negocio que tendrá toda la lógica de los servicios a usarse siendo la comunicación entre la capa de base de datos y la siguiente capa, la capa de presentación que será la interfaz con la que los usuarios (tanto cliente como proveedor) van a interactuar.

Para que la empresa GRUPO NOVATECH CÍA. LTDA. no haga una alta inversión en adquirir el hardware propietario para tener todo un *Data Center*, el presente trabajo se lo centralizará completamente sobre una arquitectura *Infrastructure as a Service* (IaaS) contratada. Para ello, también se realizará un análisis entre diferentes proveedores de servicios *cloud* públicos.

Este trabajo será realizado con información ficticia para no comprometer la privacidad de la empresa GRUPO NOVATECH CÍA. LTDA. ni a ninguno de sus clientes.

PALABRAS CLAVE: Computación en la nube, migración, SaaS, planificación estratégica, IaaS, software, cliente, proveedor.

ABSTRACT

The objective of this project is to migrate the strategic planning software (PE) offered by GRUPO NOVATECH CÍA. LTDA., from a Client – Server architecture to a Software as a Service (SaaS) distribution model.

The technology of cloud computing is a solution that provides significant advantages to the services of any company. These advantages allow a better software management and therefore costumers will have a better service.

This project will be divided into three layers: the database layer that will be the central repository of all the information, the business layer that will have all the logic of the services to be used, being the communication layer between the database layer and the next layer, called the presentation layer, that will be the interface with all users will interact (both client and provider).

In order to GRUPO NOVATECH CÍA. LTDA. do not make a high investment in acquiring the proprietary hardware to have an entire Data Center, this project will be completely centralized on an Infrastructure as a Service architecture (IaaS). For this, an analysis will be also carried out among different public cloud service providers.

This project will be used fictitious information to avoid compromising the privacy of GRUPO NOVATECH CÍA. LTDA. or that of its clients.

KEYWORDS: Cloud computing, migration, SaaS, strategic planning, IaaS, software, client, provider.

1. INTRODUCCIÓN

El presente capítulo detalla los objetivos, el alcance y el estado actual del software de planificación estratégica que lo comercializa la empresa GRUPO NOVATECH CÍA. LTDA. Se muestran los aspectos teóricos sobre la planificación estratégica y también los fundamentos teóricos del *Cloud Computing*, los niveles de servicio y de despliegue que se usan y finalmente lo relacionado con los Acuerdos de Niveles de Servicio.

1.1 Objetivos

El objetivo general de este Proyecto Integrador es desarrollar la migración del software de planificación estratégica de la empresa GRUPO NOVATECH CÍA. LTDA. a un ambiente *cloud* usando el modelo de distribución *Software as a Service* (SaaS).

Los objetivos específicos de este Proyecto Integrador son:

- Analizar la tecnología de Cloud Computing, los elementos que forman parte de un servicio SaaS y el proceso de migración de un software hacia la nube.
- Diseñar el software de planificación estratégica en su conjunto (capa de base de datos, capa de negocios y capa de presentación) para que se adapte al modelo de distribución SaaS.
- Implementar un prototipo del software diseñado en un ambiente *cloud*.
- Analizar los resultados de las pruebas de funcionamiento del software de planificación estratégica.

1.2 Alcance

Este trabajo tiene el alcance de realizar un prototipo SaaS del software de planificación estratégica para su implementación que sirva a los clientes de GRUPO NOVATECH CÍA. LTDA. de una manera más eficiente.

Para el diseño del software se usará la metodología SCRUM. Con base en esta metodología al prototipo se lo realizará como un modelo de distribución SaaS, en donde se contrate un ambiente IaaS. Previo a la contratación del IaaS se realizarán comparaciones en precio, escalabilidad, acceso de usuarios, tipo de soporte, recuperación de fallos, etc.; con el objetivo de cumplir con los requerimientos del prototipo a diseñarse.

Se diseñará la capa de base de datos, en base a un modelo relacional de las tablas de la base de datos.

Se diseñará la capa de negocio definiendo casos de uso, diagramas de secuencia y diagramas de clases del modelo de negocio. Esta capa está formada por el servidor web, el portal web para ingreso de datos y el módulo de administración del portal web.

Se diseñará la capa de presentación con la ayuda de los diagramas de visualización de los KPI (*Key Performance Indicators*) para los CMI (Cuadro de Mando Integral), es decir, se definirá la forma de visualización de los datos ingresados. Esta capa está formada por la plataforma de desarrollo de los CMI y la plataforma de administración. A este módulo de administración solamente tiene acceso el proveedor.

Al final, se tendrá una sola estructura para todos los clientes.

En la Figura 1.1 se muestra el diseño lógico de la arquitectura actual del software de planificación estratégica. Se observa que cada cliente debe tener una infraestructura dedicada para el uso de la herramienta.

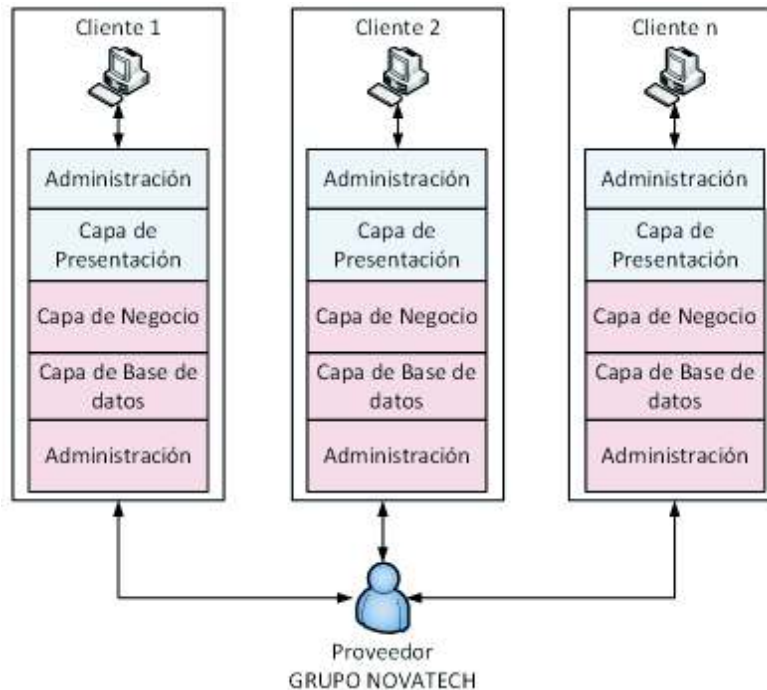


Figura 1.1. Arquitectura actual Cliente - Servidor

En la Figura 1.2 se muestra el diseño lógico del prototipo que se desea implementar. En el cual se centralizará la infraestructura en un ambiente SaaS pero sobre un ambiente IaaS contratado.

Con base en las características propias del *cloud computing*, se realizarán las pruebas de funcionamiento para el prototipo con datos de prueba de clientes ficticios.

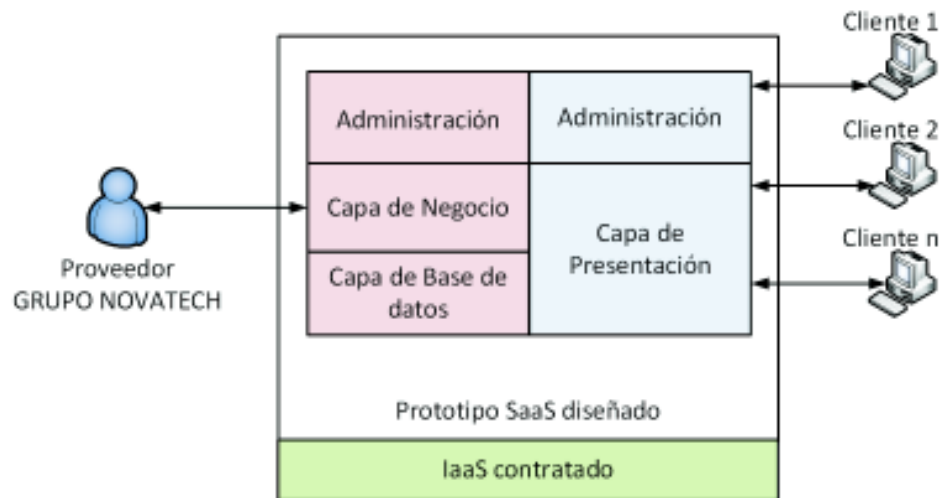


Figura 1.2. Diagrama lógico del prototipo a diseñarse

1.3 Marco Teórico

Durante muchos años, los datos de las empresas fueron almacenados dentro de sus instalaciones en lo que se conoce como un *Data Center*, que son espacios físicos reservados para las computadoras más potentes o servidores, cuyo fin es aislar los datos para que sean de uso exclusivo de las empresas.

Con el avance tecnológico de la Internet, ha surgido una modalidad en el uso y almacenamiento de datos conocida como servicios en la nube, también llamados servicios *cloud*, y que ofrecen ser algo muy similar a un *Data Center* pero en la Internet. Es decir que la empresa ya no necesita que sus servidores mantengan sus datos guardados en un espacio físico exclusivo, sino que ahora estarán en algún servidor o servidores dentro de la red más grande del mundo, la Internet.

Este cambio tecnológico supone muchas ventajas y varios riesgos. Existen muchas ventajas como el ahorro en costos al no tener que comprar máquinas nuevas sino solamente contratar el servicio mensualmente; la disponibilidad de los servicios a cualquier hora y desde cualquier parte; la facilidad en su uso ya que el soporte y mantenimiento las realiza el proveedor del servicio *cloud*, etc. Mientras que entre uno de los riesgos que más preocupan es la falta de control y seguridad en la información privada de la empresa.

Las empresas más grandes del mundo como *IBM*, *Google*, *Microsoft*, *Amazon*, entre otras; han ofrecido estos servicios desde hace mucho tiempo [1], y actualmente la tendencia

mundial es ir migrando no solo los datos, sino también los servicios hacia un modelo de prestación de servicios *cloud*, llamado *Cloud Computing*.

1.4 Planificación Estratégica

La Planificación Estratégica (PE) es un término muy usado en el ámbito militar desde el tiempo de la antigua Grecia que, llevado al contexto empresarial, hace referencia a las acciones que se han de llevar a cabo para cumplir uno o varios objetivos.

1.4.1 Concepto

Usando el concepto dado por la Comisión Económica para América Latina y el Caribe (CEPAL) [2] se puede decir que la PE es *“una herramienta de gestión que permite apoyar la toma de decisiones de las organizaciones en torno al quehacer actual y al camino que deben recorrer en el futuro para adecuarse a los cambios y a las demandas que les impone el entorno y lograr la mayor eficiencia, eficacia, calidad en los bienes y servicios que se proveen”*. Cabe clarificar que la eficiencia y la eficacia deben ser las metas a cumplir de forma simultánea, es decir, que se deben cumplir a cabalidad y en el menor tiempo posible los objetivos estratégicos definidos en la PE (eficacia) pero el cumplimiento de estos objetivos deberá ser realizado sacando el mayor provecho de los recursos disponibles (eficiencia).

1.4.2 Objetivos Estratégicos

Toda PE debe tener como resultado varios Objetivos Estratégicos (OE) definidos en función de la estrategia que se desee usar en la empresa.

Los OE son las metas que se han planteado en la PE y que permiten materializar la misión, la visión y los valores de cada una de las empresas clientes en un largo plazo. Como todos los objetivos, éstos deben ser claros, cuantificables, realistas y alcanzables en un tiempo limitado. Para poder cuantificar el avance del cumplimiento de un OE se hace uso de los Indicadores y los Cuadros de Mando Integrales.

1.4.3 Indicadores y Cuadros de Mando Integrales

Los indicadores o *Key Performance Indicators* (KPI) son las métricas con las cuales se evalúa el cumplimiento de un OE. Los KPI se agrupan y son presentados en lo que se conoce como un Cuadro de Mando Integral (CMI). No se puede hablar de Indicadores sin mencionar un Cuadro de Mando Integral.

Independientemente de la plataforma de diseño de los CMI, los KPI deben ser monitoreados constantemente por las empresas para constatar cual es el avance en el cumplimiento de los OE y, en función del tiempo de ejecución, tomar decisiones que puedan encausarlos de la mejor manera. Por eso es importante que los KPI que formen parte de los CMI sean los que aporten mayor relevancia en la información de la empresa ya que, finalmente, son los instrumentos de medición de toda la PE. El monitoreo de los KPI se los realiza con planes de acción asociados a ellos. Estos planes de acción constan de una lista de tareas a ser realizadas para poder llevarlos a cabo por medio de constantes registro de avances.

1.5 Empresa GRUPO NOVATECH CÍA. LTDA.

GRUPO NOVATECH CÍA. LTDA. es una empresa de consultoría y soluciones empresariales que lleva más de veinte años trabajando en el Ecuador y en otros países de América Latina. Tienen clientes de varias áreas comerciales de alimentos y bebidas, industriales, automotriz y acero, banco y financiero, sector público, farmacéutica y distribución, telecomunicaciones, etc. Es proveedor de plataformas empresariales mundialmente reconocidas como *Infor* [3], *Microsoft* [4], *Qlik* [5], *Corporate Planning* [6], *Demand Solutions* [7], entre otros.

En su página web [8] se puede leer que su misión es: *“Generamos valor para nuestros Socios-Clientes proveyendo soluciones empresariales integrales, innovadoras, flexibles, de clase mundial, válidas para nuestros mercados, para apoyar el logro de sus objetivos de negocio, creando relaciones de largo plazo, mutuamente rentables”*. Y su visión es: *“Novatech es la primera opción de asesoría empresarial para las organizaciones más exitosas. Su solidez, calidad de profesionales, presencia internacional y los resultados de sus clientes, contribuyen al desarrollo de la región”*.

1.5.1 Situación Actual

Actualmente el software de PE está siendo comercializado por la empresa GRUPO NOVATECH. CIA. LTDA. en una arquitectura Cliente – Servidor [9] siendo un requerimiento para las empresas que ya son sus clientes el adquirir componentes de hardware para su uso específico. Todas las herramientas comercializadas por esta empresa cuentan con un acuerdo de soporte y mantenimiento, sin embargo, es el departamento de Tecnologías de la Información (TI) en cada empresa cliente el responsable de la arquitectura y su correcto funcionamiento, una vez puesto en un ambiente de producción.

El problema que presenta el software de PE es la arquitectura con la cual se la comercializa debido a que dificulta su gestión. Desde la perspectiva de las empresas clientes de GRUPO NOVATECH CÍA. LTDA., el problema principal se da por dos razones: primero, por la inversión inicial considerable para el funcionamiento de la arquitectura del software y segundo, porque el departamento de TI de estas empresas no alcanza a atender aspectos más centrales del núcleo del negocio por tener a cargo una herramienta más a la lista de herramientas adquiridas y que residen bajo su completa responsabilidad. Es decir que se tiene que recurrir constantemente a la empresa GRUPO NOVATECH CÍA. LTDA. para realizar cualquier tipo de actualización y de mantenimiento, minimizando la funcionalidad del software hasta que se gestione la orden de soporte correspondiente, o en ciertos casos, hasta que la renovación del contrato de soporte sea aprobada por los directores de la empresa. Desde la perspectiva de proveedor, como lo es la empresa GRUPO NOVATECH CÍA. LTDA., el problema principal se deriva de las mismas razones anteriores. Primero, la inversión inicial que realizan sus clientes suele dificultar la proyección financiera ya que se convierte en un rubro que no representa un ingreso directo al proyecto. Segundo, el no gestionar personalmente la arquitectura, ralentiza el proceso de soporte, actualización y mantenimiento al tener varios clientes sin una administración centralizada de la herramienta.

Debido a que por cada cliente se debe realizar una instalación completa con su respectiva configuración, se tienen muchas versiones de la misma aplicación y esto dificulta al momento que se requiere realizar cualquier acción de gestión o actualización sobre el software de PE.

1.6 Cloud Computing

La definición de *Cloud Computing* dada por el Instituto Nacional de Estándares y Tecnología de los Estados Unidos (*National Institute of Standards and Technology* - NIST) dice que *Cloud Computing* es un modelo tecnológico de negocios “que permite un acceso a la red desde cualquier lugar, de forma conveniente y bajo demanda para compartir un conjunto de recursos computacionales configurables (red, servidores, almacenamiento, aplicaciones y servicios) que puedan ser rápidamente provisionados y ofertados con un mínimo esfuerzo de administración o de interacción con el proveedor del servicio” [10].

1.6.1 Características Esenciales de Cloud Computing

Para que un modelo tecnológico pueda ser considerado como *Cloud Computing* debe cumplir con las características que se detallan en la Tabla 1.1.

Tabla 1.1. Características esenciales de *Cloud Computing*

Característica	Definición	Recursos
Auto servicio bajo demanda	La capacidad de que el cliente aumente sus recursos de forma automática en función de su necesidad y sin la intervención de un proveedor de servicio.	CPU, RAM, disco duro, ancho de banda
Amplio acceso a la red	La disponibilidad de acceso al servicio en cualquier parte del mundo a través de diferentes plataformas y dispositivos con base en mecanismos estandarizados.	Navegador web
Disponibilidad de recursos	La oferta de recursos computacionales disponibles para ser asignados a uno o varios clientes de manera dinámica en función de sus requerimientos.	CPU, RAM, disco duro, ancho de banda
Rápida elasticidad	El rápido escalamiento de recursos proporcionalmente a la demanda, y preferiblemente de forma automática, que den al cliente la idea que tiene una oferta de recursos ilimitados.	CPU, RAM, disco duro, ancho de banda
Medición de los servicios	El monitoreo permanente de los recursos usados por el cliente acorde al tipo de servicio asegurando la transparencia tanto para el cliente como para el proveedor.	CPU, RAM, disco duro, ancho de banda, cuentas de usuarios activos por mes
Pago por uso	Pagar por un servicio en la nube en función de los recursos usados durante cierto periodo de tiempo establecido o pagar con antelación el potencial consumo que se piense usar, gracias a su característica de rápida elasticidad.	Pago mensual, pago por <i>tokens</i> , pre pago

1.6.2 Modelos de Despliegue

Un modelo de despliegue hace referencia al tipo de arquitectura en la cual el servicio *cloud* va a estar disponible. Los cuatro modelos de despliegue en *cloud computing* son:

1. **Nube Pública:** Es una arquitectura en la nube que está disponible para cualquier cliente. Esta arquitectura es administrada por el proveedor del servicio *cloud*. Los servicios *cloud* más reconocidos son *Amazon Web Services (AWS)* [11], *Microsoft Azure (MA)* [12] y *Google Cloud Engine (GCE)* [13].
2. **Nube Privada:** Es una arquitectura en la nube para uso exclusivo de una sola organización, puede ser administrada por la misma organización que la usa o por terceros. Este tipo de nube privada puede encontrarse dentro de la misma organización o puede ser arrendada a un proveedor *cloud* externo.
3. **Nube Comunitaria:** Es una arquitectura en la nube para el uso de varias organizaciones o clientes que posean intereses en común. Esta arquitectura puede ser administrada por una o varias organizaciones o por un proveedor de servicios *cloud*. Como ejemplo se puede citar a *GovCloud* [14], la nube comunitaria del gobierno de los Estados Unidos que funciona sobre AWS.
4. **Nube Híbrida:** Es básicamente cualquier combinación de las tres anteriores bajo la premisa de que sean entidades únicas pero que compartan datos o aplicaciones. Este tipo de nube se la puede usar por ejemplo cuando exista una aplicación en desarrollo en una nube privada que se la quiera comercializar en una nube pública.

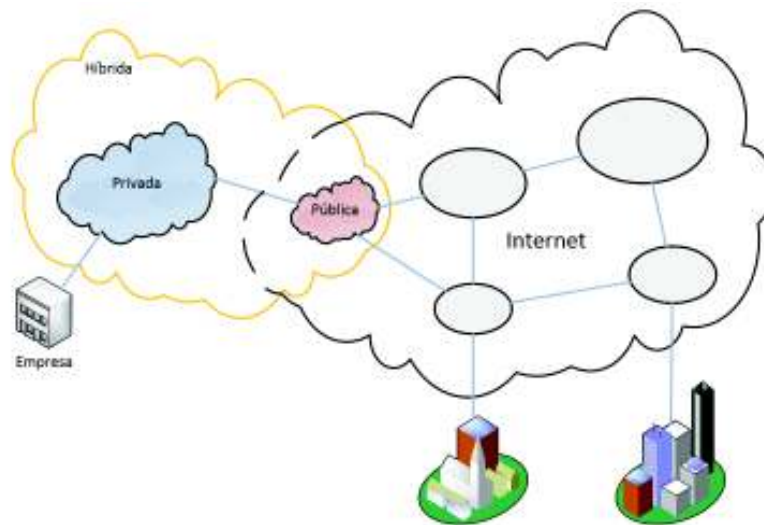


Figura 1.3. Modelos de despliegue

En la Figura 1.3 se muestra la interrelación que tienen los modelos de despliegue privado, público e híbrido.

1.6.3 Modelos de Servicios

a. Infrastructure as a Service (IaaS)

Con el modelo de servicio de infraestructura, o *Infrastructure as a Service* (IaaS), se puede hacer uso de recursos computacionales fundamentales sobre los cuales se desarrollará y ejecutará cualquier tipo de software, incluyendo sistemas operativos y aplicaciones.

En la Figura 1.4 [15] se muestran los niveles de control que tiene tanto un cliente como un proveedor con el modelo de servicio IaaS. Como se observa, el cliente tiene casi todo el control sobre la estructura, solamente no tiene acceso al hardware.

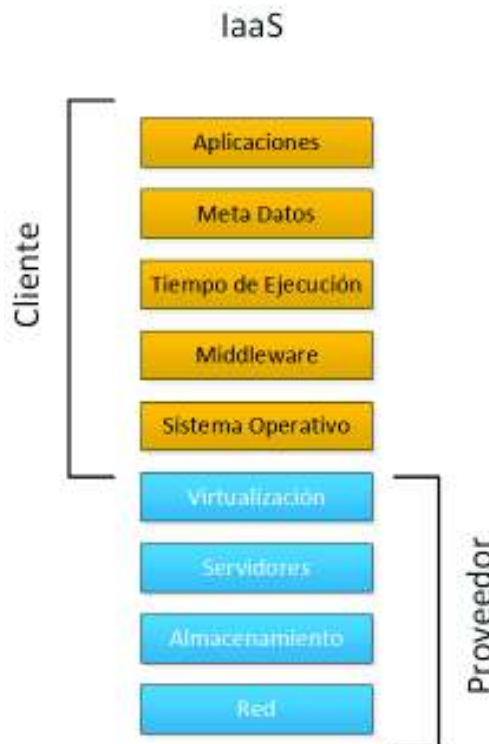


Figura 1.4. Niveles de control en IaaS

Este servicio da al cliente la idea de lo que se conoce como una máquina virtual porque lo que contrata a un proveedor es solamente los recursos computacionales que tendría una computadora sin sistema operativo. Un ejemplo de este servicio es *Amazon Web Services* (AWS) [11], que permite hacer uso de una capa gratuita con una combinación de recursos disponibles [16] como procesamiento, almacenamiento en bases de datos relacionales y no relacionales, flujos de trabajo, servicios de búsqueda, etc.

Dentro de las ventajas de este modelo de servicio se encuentran:

- **Escalabilidad:** Los recursos están a disposición de su uso de manera rápida y flexible, es decir, que se pueden aumentar o disminuir la infraestructura según las necesidades.
- **Rapidez:** El tiempo entre que se contrata el servicio y se obtiene acceso a la infraestructura es mínimo, casi se podría decir que los tiempos de espera desaparecen.
- **Costos directos:** El pago de los recursos solamente es en función a los recursos utilizados. Ya no existen los costos [17] en electricidad, refrigeración, espacio ocupado, control de acceso y mantenimiento físico por parte del cliente en la implementación de la infraestructura.
- **Seguridad:** Cada vez existen más medidas de seguridad en IaaS [18], no solo a nivel de autenticación en acceso sino también en prevención de ataques maliciosos, problemas de interoperabilidad de tecnología, pérdida de datos, etc.
- **Disponibilidad:** El tiempo de posibles fallas es reducido ya que, por los modelos de redundancia, se tiene una alta disponibilidad. Si falla un servidor, o incluso un conjunto de servidores, su afectación es mínima en el servicio ofrecido.
- **Acceso:** La posibilidad de acceder a cualquier momento y desde cualquier parte del mundo es otra ventaja significativa para tomar en cuenta.

b. Platform as a Service (PaaS)

Con el modelo de servicio de plataforma, o *Platform as a Service* (PaaS), se puede hacer uso de recursos de desarrollo de plataformas computacionales sobre una infraestructura establecida. Las herramientas de desarrollo están a disposición en el servicio y los usuarios acceden vía web. Los usuarios pueden además, realizar pruebas, compilar, ejecutar y documentar sin tener que instalar ninguna otra herramienta [19]. Un ejemplo de este servicio es *Google App Engine* [20] que es una plataforma de desarrollo para aplicaciones que se deseen escalar rápidamente.

En la Figura 1.5 [15] se muestran los niveles de control que tiene tanto un cliente como un proveedor con el modelo de servicio PaaS. Como se observa, el proveedor tiene casi todo el control sobre la infraestructura, solamente no tiene acceso a las aplicaciones ni a sus metadatos.

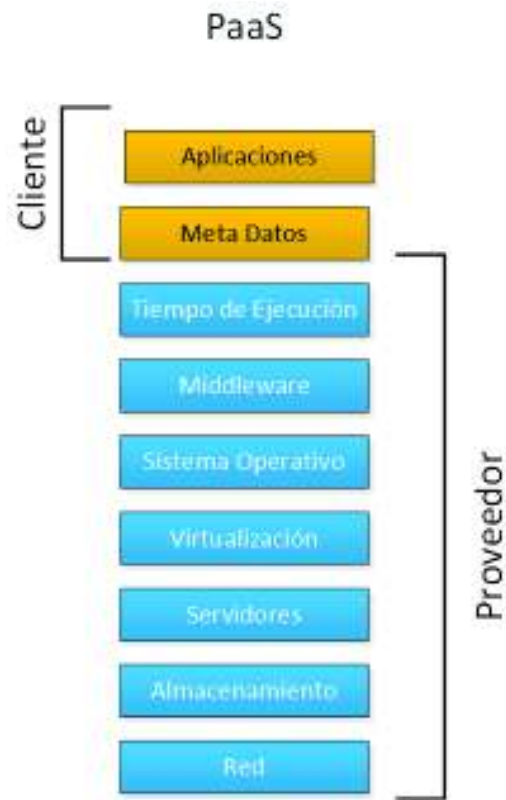


Figura 1.5. Niveles de control en PaaS

Hay varias ventajas en tener a disposición el servicio PaaS, como por ejemplo [21]:

- **Tiempo de programación:** Se reduce el tiempo en el que se pueden obtener los entregables de los desarrollos de software, ya que no hay que instalar y configurar las plataformas que van a ser usadas.
- **Equipos de desarrollo:** Se pueden agregar más funcionalidades en las plataformas de desarrollo sin aumentar el número del personal o contratar personal especializado.
- **Multiplataforma:** El servicio PaaS permite desarrollar independientemente de la plataforma que se vaya a usar. Esta opción permite una ventaja en los desarrollos y promueve la interoperabilidad.
- **Pago por uso:** Si el desarrollo requiere más recursos se podrá hacer uso de los mismos en tiempo real, y cuando ya no se los necesite no se tendrá que pagar. El pago se da en función de los recursos consumidos en el periodo acordado.
- **Procesos de desarrollo:** Al optar por un servicio PaaS, todo el proceso de desarrollo va a estar centralizado. Esto permite una mejor administración en la

programación, la compilación, las pruebas, la implementación, la actualización y la documentación de un proyecto.

c. **Software as a Service (SaaS)**

Con el modelo de aplicaciones como servicio, o *Software as a Service* (SaaS), el cliente puede usar las aplicaciones disponibles que están siendo ejecutadas sobre la infraestructura del proveedor. Esta infraestructura también se encuentra en la nube al ser administrada por el mismo proveedor. El acceso a las aplicaciones es por medio de un acceso web.

El cliente no administra ni controla ninguno de los elementos de la infraestructura en la nube del proveedor, solamente hace uso de las aplicaciones. Un cotidiano ejemplo de este tipo de servicio es el correo electrónico, al cual se accede por medio de un navegador web o de una aplicación instalada en la máquina del cliente, y se hace uso del servicio sin saber dónde está la infraestructura ni cómo se la administra.

En la Figura 1.6 [15] se muestran los niveles de control que tiene tanto un cliente como un proveedor con el modelo de servicio SaaS. Como se observa, el proveedor tiene control de toda la estructura. El cliente solamente usa los servicios disponibles.



Figura 1.6. Niveles de control en SaaS

A continuación se mencionan cinco ventajas específicas de uso de SaaS como modelo de servicio recopiladas por el NIST [22].

- **Acceso ubicuo al servicio:** La tendencia actual es usar navegadores para el acceso a los servicios SaaS porque son herramientas pre-configuradas, no exigen un proceso muy complicado de instalación, la mayoría son gratuitas, al ser muy ligeras no interfieren con otras aplicaciones que se estén ejecutando en paralelo, se actualizan constantemente y sus métodos de comunicación con los servidores son estandarizados.
- **Uso eficiente de licencias de software:** Por un lado, la licencia del software asignado a un cliente puede ser usada desde varios dispositivos, lo que implica una reducción en el costo al no tener que comprar una licencia para cada dispositivo. Por otro lado, las licencias de hardware que antes las gestionaba el cliente, ahora son administradas por el proveedor, lo que significa que, desde la perspectiva del cliente, no paga una licencia del hardware que vaya a usar.
- **Administración de datos centralizada:** Como las bases de datos son administradas por el proveedor, la información, desde la perspectiva del cliente, estará centralizada. Esto permite que el cliente tenga su información disponible a cualquier hora del día y desde cualquier lugar, que pueda recuperarla en caso de incidentes y que además pueda ser compartida fácilmente a otros usuarios sin necesidad de tenerla localmente en su computadora. Esto exige que el proveedor mantenga procesos de contingencia de los datos y también redundancia en su almacenamiento.
- **Responsabilidades de la plataforma de lado del proveedor:** Todos los procesos de actualizaciones, mantenimiento, configuraciones, versionamiento, pruebas de rendimiento, respaldos, parches de seguridad, soporte ante errores, etc.; recaen bajo la responsabilidad del proveedor. Lo que implica un uso directo de la herramienta del lado del cliente (excepto las opciones de configuración propias del usuario). El proveedor tendrá la responsabilidad sobre todos estos procedimientos, los tendrá que hacer una sola vez para todos los clientes, en lugar de realizar un procedimiento por cada uno de los servidores de cuantos clientes tenga.
- **Ahorro de Costos:** Los servicios SaaS permiten al cliente el uso de aplicaciones sin un costo inicial de adquisición del hardware requerido, pero, en cambio, con un rubro mensualizado. Un proveedor de servicios *cloud* podrá instalar y configurar

cualquier otro tipo de recurso bajo demanda de forma más eficiente de lo que haría el propio cliente, lo que significará un ahorro en los costos de escalamiento. Pero como en toda inversión, el cliente deberá analizar detalladamente los aspectos que puedan influir en el rendimiento deseado para el uso del aplicativo en un plazo de tiempo considerable.

d. Comparativo entre modelos de servicio

En la Tabla 1.2 se puede ver un comparativo de los modelos de servicio mencionados para posteriormente, en la Sección 2.1.3, definir el modelo de servicio necesario para el presente sistema.

Tabla 1.2. Comparativo entre modelos de servicio

	IaaS	PaaS	SaaS
Objetivo	Dónde migrar	Dónde desarrollar	Qué servicios consumir
Rol del Cliente	Alto	Medio	Bajo
Rol del Proveedor	Bajo	Medio	Alto
Usuarios	Administradores	Desarrolladores	Finales
Aplicaciones	Máquinas virtuales, red, servidores, bases de datos.	Librerías y plataformas de desarrollo.	Servicios e interfaces de aplicaciones.
Nivel de Abstracción	Bajo	Medio	Alto

En la Figura 1.7 [15] se muestran los niveles de control que tiene tanto el cliente como el proveedor en los diferentes servicios *cloud* comparados con un *Data Center* tradicional. Como se puede observar, el control que tiene el cliente va disminuyendo conforme se tenga uno u otro modelo de servicio.

1.6.4 Riesgos en Cloud Computing

Los servicios *cloud*, similar a cualquier otro *Data Center* de grandes prestaciones, tiene ciertos riesgos, los cuales hay que tenerlos presentes para una migración hacia la nube. A continuación, se presentan aquellos más relevantes:

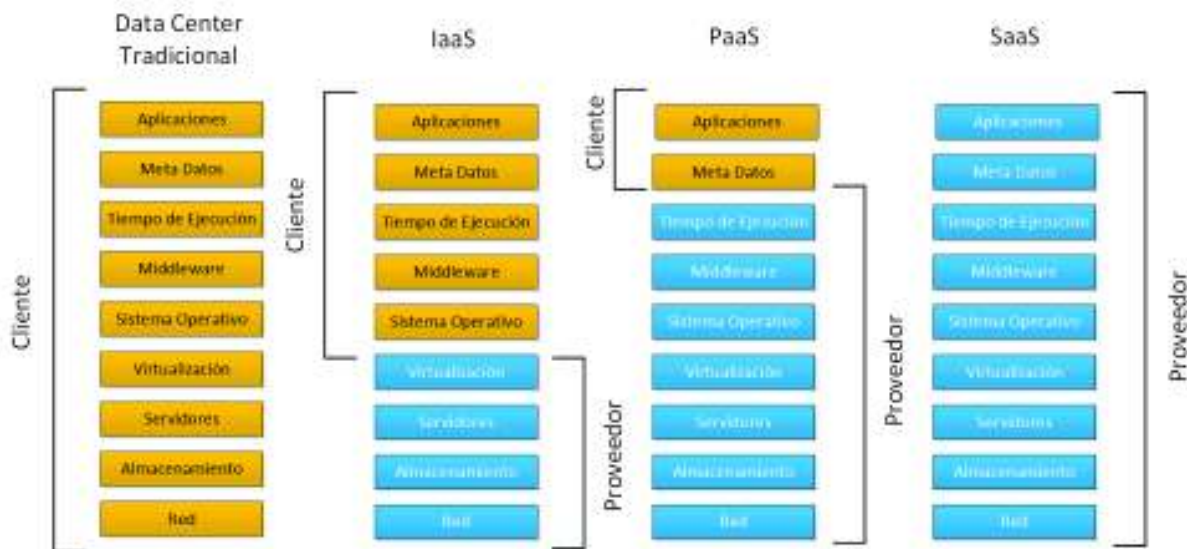


Figura 1.7. Niveles de control en los servicios *cloud*

- Rendimiento:** En un análisis realizado por la *International Journal of Computational Science, Information Technology and Control Engineering* (IJCSITCE) [23] en octubre del 2013, se observa que cada aplicación tiene sus propias exigencias a nivel del rendimiento, en otras palabras, aspectos tan disímiles como la vulnerabilidad de la seguridad, el tiempo de recuperación ante fallas, el ancho de banda por cada usuario, la capacidad de almacenamiento de la infraestructura *cloud*, el tamaño de los *buffers* en las aplicaciones con las que se accede al servicio, la tasa de disponibilidad acordada, el número de usuarios concurrentes, la ubicación de los *Data Center* distribuidos, entre otros; pueden afectar directamente al tiempo de respuesta percibido por el cliente.
- Confiabilidad:** Los proveedores de servicios *cloud* deben asegurar que los procesos críticos de sus clientes no se vean afectados por fallas a nivel de software o hardware. Existen muchas estrategias que son usadas actualmente para mitigar el riesgo frente a posibles caídas del sistema. Las recomendaciones básicas [24] se centran alrededor del monitoreo constante de los recursos de la infraestructura (especialmente de los niveles de energía en las fuentes de alimentación) con la ayuda de software integrado a sus plataformas de administración, la redundancia a nivel de aplicaciones que generen constantemente alarmas automáticas en función de la sincronización de los datos, la funcionalidad de respaldos de todo el sistema en servidores físicos de la forma más rápida posible hasta que se puedan resolver los problemas técnicos.

- **Conformidad:** En el Ecuador no existe una normativa que regule específicamente los servicios *cloud*. Las leyes vigentes contemplan la protección de datos y la confidencialidad, pero no existe legislación respecto de las limitaciones, responsabilidades e indemnizaciones. Como consecuencia de ello, se considera al Acuerdo de Nivel de Servicio o *Service Level Agreement* (SLA) el documento principal para el cumplimiento de los servicios *cloud*.
- **Dependencia:** Existe un riesgo asociado directamente con el proveedor del servicio *cloud* y es la dependencia que genera. Dependencia no solo de tipo de hardware y software que use en las capas de mayor abstracción (lo cual se reduce cada vez más con los estándares internacionales) sino y, sobre todo, de las configuraciones propias de cada servicio. Como resultado de esta dependencia, el proceso de migración a otro proveedor *cloud* puede volverse un punto de vulnerabilidad [25]. Especialmente cuando por cualquier razón el cliente desee cambiar de proveedor, o también cuando se desee integrar funcionalidades de un nuevo proveedor a la nube que está en producción. Este riesgo exige al área técnica conocer ampliamente los conceptos y definiciones del SLA de los diferentes proveedores *cloud*.

1.6.5 Proceso de Migración

El proceso de migración del software de PE que la empresa GRUPO NOVATECH CÍA. LTDA. comercializa desde una arquitectura Cliente – Servidor, y que ahora lo va hacer desde una solución SaaS, requiere los siguientes pasos:

- **Analizar la arquitectura actual del software:** El software que se desea migrar debe tener componentes independientes que puedan ser instalado y configurados de manera autónoma [26].
- **Evaluar los riesgos de un servicio en la nube:** Cada tipo de servicio en la nube tiene ventajas y desventajas, evaluar los posibles riesgos del proceso de migración podrá mitigar si alguno de ellos llegara a afectar al funcionamiento del software.
- **Escoger un proveedor de servicios *cloud* acorde a las necesidades:** Es importante hacer un estudio previo de los diferentes proveedores de servicios *cloud* y para contratarlo con base en un análisis técnico en función de los requerimientos.

- **Validar el funcionamiento integral:** Cuando se ha realizado la migración completa es de suma importancia verificar el correcto funcionamiento de los diferentes componentes del software pero ahora como un servicio *cloud*.
- **Planificar el proceso de monitoreo de recursos:** Un correcto monitoreo de recursos en la nube permitirá conocer el comportamiento del software bajo diferentes circunstancias y, con esta información, poder escalar rápidamente sin afectar al tiempo de respuesta de los usuarios.
- **Acoplar la gestión interna del software:** Al realizar la migración hacia la nube, la gestión interna va a cambiar, tanto el proveedor como los clientes tienen que estar claros en sus respectivas responsabilidades para no afectar al correcto funcionamiento del software [27].

1.6.6 Legislación en el Ecuador

La Red Iberoamericana de Protección de Datos [28] lista los documentos legales que regulan los servicios de *Cloud Computing* para el Ecuador. Como se observa en todos los artículos, la regulación es muy general y no está específicamente asociada al tema del *Cloud Computing*. Como consecuencia, la herramienta más usada para regular los servicios *cloud* es el SLA.

Un SLA es un documento en el que se detallan las características del servicio *cloud* que un cliente contrata con un proveedor determinado, y sus correspondientes responsabilidades. En la Sección 1.7 se tiene un detalle más amplio de un SLA en conjunto con sus componentes y sus lineamientos principales.

En la Tabla 1.3 se muestra el contenido de los artículos mencionados en la Constitución del Ecuador [29], en donde se mencionan aspectos como la protección de datos personales o corporativos, derechos de inviolabilidad de información, y las excepciones en las cuales se podría llegar a conocer estos datos protegidos.

Los servicios *cloud* manejan información que puede ser privada, tanto personal como corporativa, por eso es necesario conocer las normas legales que la rigen.

Tabla 1.3. Artículos citados de la Constitución

Constitución	
Art. 66 # 19	<p><i>“El derecho a la protección de datos de carácter personal, que incluye el acceso y la decisión sobre información y datos de este carácter, así como su correspondiente protección. La recolección, archivo, procesamiento, distribución o difusión de estos datos o información requerirán la autorización del titular o el mandato de la ley”.</i></p>
Art. 66 # 21	<p><i>“El derecho a la inviolabilidad y al secreto de la correspondencia física y virtual; ésta no podrá ser retenida, abierta ni examinada, excepto en los casos previstos en la ley, previa intervención judicial y con la obligación de guardar el secreto de los asuntos ajenos al hecho que motive su examen. Este derecho protege cualquier otro tipo o forma de comunicación”.</i></p>
Art. 66 # 92	<p><i>“Toda persona, por sus propios derechos o como representante legitimado para el efecto, tendrá derecho a conocer de la existencia y a acceder a los documentos, datos genéticos, bancos o archivos de datos personales e informes que sobre sí misma, o sobre sus bienes, consten en entidades públicas o privadas, en soporte material o electrónico. Asimismo, tendrá derecho a conocer el uso que se haga de ellos, su finalidad, el origen y destino de información personal y el tiempo de vigencia del archivo o banco de datos.</i></p> <p><i>Las personas responsables de los bancos o archivos de datos personales podrán difundir la información archivada con autorización de su titular o de la ley.</i></p> <p><i>La persona titular de los datos podrá solicitar al responsable el acceso sin costo al archivo, así como la actualización de los datos, su rectificación, eliminación o anulación. En el caso de datos sensibles, cuyo archivo deberá estar autorizado por la ley o por la persona titular, se exigirá la adopción de las medidas de seguridad necesarias. Si no se atendiera su solicitud, ésta podrá acudir a la jueza o juez. La persona afectada podrá demandar por los perjuicios ocasionados”.</i></p>

En la Tabla 1.4 se muestra el contenido de los artículos mencionados en la Ley Orgánica de Transparencia y Acceso a la Información Pública [30], en donde se mencionan aspectos como el acceso a la información pública y el correcto uso de información confidencial.

Tabla 1.4. Artículos citados de la Ley Orgánica de Transparencia y Acceso a la Información Pública

Ley Orgánica de Transparencia y Acceso a la Información Pública	
Art. 1	Principio de Publicidad de la Información Pública: <i>“El acceso a la información pública es un derecho de las personas que garantiza el Estado. Toda la información que emane o que esté en poder de las instituciones, organismos y entidades, personas jurídicas de derecho público o privado que, para el tema materia de la información tengan participación del Estado o sean concesionarios de éste, en cualquiera de sus modalidades, conforme lo dispone la Ley Orgánica de la Contraloría General del Estado; las organizaciones de trabajadores y servidores de las instituciones del Estado, instituciones de educación superior que perciban rentas del Estado, las denominadas organizaciones no gubernamentales (ONGs), están sometidas al principio de publicidad; por lo tanto, toda información que posean es pública, salvo las excepciones establecidas en esta Ley”.</i>
Art. 6	<p>Información Confidencial: <i>“Se considera información confidencial aquella información pública personal, que no está sujeta al principio de publicidad y comprende aquella derivada de sus derechos personalísimos y fundamentales, especialmente aquellos señalados en los artículos 23 y 24 de la Constitución Política de la República. El uso ilegal que se haga de la información personal o su divulgación dará lugar a las acciones legales pertinentes.</i></p> <p><i>No podrá invocarse reserva, cuando se trate de investigaciones que realicen las autoridades, públicas competentes, sobre violaciones a derechos de las personas que se encuentren establecidos en la Constitución Política de la República, en las declaraciones, pactos, convenios, instrumentos internacionales y el ordenamiento jurídico interno. Se exceptiona el procedimiento establecido en las indagaciones previas”.</i></p>

Los servicios *cloud* al ser contratados tanto por entidades privadas como por entidades públicas, deben cumplir las normas legales que las rigen para evitar el uso malintencionado de información confidencial a la par de mantener la transparencia de la información catalogada como pública.

En la Tabla 1.5 se muestra el contenido de los artículos mencionados en la Ley del Sistema Nacional de Registro de Datos Públicos [31] en donde se mencionan aspectos como lo que debe ser un sistema informático para entidades estatales y la seguridad con la que debe contar.

Tabla 1.5. Artículos citados de la Ley del Sistema Nacional de Registro de Datos Públicos

Ley del Sistema Nacional de Registro de Datos Públicos	
Art. 23	Sistema Informático: <i>“El sistema informático tiene como objetivo la tecnificación y modernización de los registros, empleando tecnologías de información, bases de datos y lenguajes informáticos estandarizados, protocolos de intercambio de datos seguros, que permitan un manejo de la información adecuado que reciba, capture, archive, codifique, proteja, intercambie, reproduzca, verifique, certifique o procese de manera tecnológica la información de los datos registrados. El sistema informático utilizado para el funcionamiento e interconexión de los registros y entidades es de propiedad estatal y del mismo se podrán conceder licencias de uso limitadas a las entidades públicas y privadas que correspondan, con las limitaciones previstas en la Ley y el Reglamento”.</i>
Art. 26	Seguridad: <i>“Toda base informática de datos debe contar con su respectivo archivo de respaldo, cumplir con los estándares técnicos y plan de contingencia que impidan la caída del sistema, robo de datos, modificación o cualquier otra circunstancia que pueda afectar la información pública”.</i>

En la Tabla 1.6 se muestra el contenido de los artículos mencionados en la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos [32] en donde se mencionan aspectos como la confidencialidad que deben tener los datos de comercio electrónico, la protección de datos que deben elaborarse por las entidades de certificación acreditadas.

Tabla 1.6. Artículos citados de la Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos

Ley de Comercio Electrónico, Firmas Electrónicas y Mensajes de Datos	
Art. 5	Confidencialidad y reserva: <i>“Se establecen los principios de confidencialidad y reserva para los mensajes de datos, cualquiera sea su forma, medio o intención. Toda violación a estos principios, principalmente aquellas referidas a la intrusión electrónica, transferencia ilegal de mensajes de datos o violación del secreto profesional, será sancionada conforme a lo dispuesto en esta Ley y demás normas que rigen la materia”.</i>
Art. 9	Protección de datos: <i>“Para la elaboración, transferencia o utilización de bases de datos, obtenidas directa o indirectamente del uso o transmisión de mensajes de datos, se requerirá el consentimiento expreso del titular de éstos, quien podrá seleccionar la información a compartirse con terceros. La recopilación y uso de datos personales responderá a los derechos de privacidad, intimidad y confidencialidad garantizados por la Constitución Política de la República y esta ley, los cuales podrán ser utilizados o transferidos únicamente con autorización del titular u orden de autoridad competente. No será preciso el consentimiento para recopilar datos personales de fuentes accesibles al público, cuando se recojan para el ejercicio de las funciones propias de la administración pública, en el ámbito de su competencia, y cuando se refieran a personas vinculadas por una relación de negocios, laboral, administrativa o contractual y sean necesarios para el mantenimiento de las relaciones o para el cumplimiento del contrato. El consentimiento a que se refiere este artículo podrá ser revocado a criterio del titular de los datos; la revocatoria no tendrá en ningún caso efecto retroactivo”.</i>
Art. 32	Protección de datos por parte de las entidades de certificación de información acreditadas: <i>“Las entidades de certificación de información garantizarán la protección de los datos personales obtenidos en función de sus actividades, de conformidad con lo establecido en el artículo 9 de esta ley”.</i>

1.7 Acuerdo de Nivel de Servicio

El Acuerdo de Nivel de Servicio o *Service Level Agreement* (SLA) es un documento en el que se detallan las características del servicio *cloud* que un cliente contrata con un proveedor determinado, y sus correspondientes responsabilidades. Aunque no existe un estándar respecto al formato del SLA o los elementos que no deben conformar, la *Cloud Standard Consumer Council* [33] ha desarrollado una guía que tiene los lineamientos principales de un SLA.

1.7.1 Lineamientos Principales de un SLA

Un SLA debe contener la mayor cantidad de información relevante alrededor del servicio *cloud* que un proveedor oferta y un cliente contrata. En la Tabla 1.7 se detallan los lineamientos principales que deberían formar parte de un SLA.

Tabla 1.7. Lineamientos principales de un SLA

Título	Lineamiento
Roles y responsabilidades	Se deben delimitar lo más claramente posible los deberes y derechos que tiene tanto el cliente como el proveedor.
Rendimiento	Los recursos usados por el cliente tienen que ser constantemente medidos para que el rendimiento sea lo más cercano a lo establecido.
Privacidad	Los datos del cliente no deberían ser manipulados por ninguna persona que no presente las credenciales correspondientes.
Disponibilidad	El servicio debe asegurar que, en la medida de lo posible, siempre esté activo y disponible para su acceso.
Recuperación ante fallos	Si existieran fallas en el servicio, por cualquier razón, se debe tener claro el proceso a llevar a cabo para una recuperación lo más rápida posible.
Gobernabilidad	Se debe tener un constante proceso de monitoreo de los recursos disponibles y de los datos respecto de la administración del servicio.
Terminación de Contrato	Se debe dejar en claro la forma de finalización del servicio y bajo qué circunstancias deberá producirse.

1.8 Metodología SCRUM

SCRUM es una metodología ágil de desarrollo de software que permite brindar soluciones de TI basadas en un proceso iterativo y que facilita la colaboración entre los desarrolladores y los clientes [34]. Esta metodología tiene varios componentes que forman parte de su proceso. A continuación, se van a detallar los componentes a usar en el sistema.

Historias de Usuario

La historia de usuario es un documento en el cual se describen los requerimientos que se desea del software a desarrollarse [35]. En su estructura debe estar explícitamente escrito el nombre de la funcionalidad, la descripción de la funcionalidad y el criterio de evaluación o aceptación. Al final del desarrollo del sistema se usarán estas historias de usuario para verificar que se hayan llevado a cabo todos los requerimientos solicitados.

Roles

En la metodología SCRUM es necesario definir los roles de las personas que van a participar en el desarrollo y para ello se definen tres roles [36]:

- El dueño del producto, o también conocido como el cliente. Es la persona o grupo de personas que requieren el desarrollo.
- El director del SCRUM es el que supervisa, verifica y lleva a cabo el proceso de desarrollo. Como su nombre lo indica, es el que dirige al equipo de desarrollo.
- Equipo de desarrollo, son todas las personas que intervienen en el desarrollo del software.

Para el sistema, y en el desarrollo de este escrito, cada vez que se mencione el “proveedor” se estará haciendo referencia a la empresa GRUPO NOVATECH CÍA. LTDA., cuando se mencione el “cliente” se estará haciendo referencia a cualquiera de las empresas clientes de GRUPO NOVATECH CÍA. LTDA. Para diferenciar cualquier proveedor o clientes de las anteriores empresas mencionadas, se lo expondrá explícitamente.

Backlog

El *backlog* es la recopilación de los requisitos en orden de prioridad que se han definido para el software. Este documento marcará el trabajo a realizar por parte del equipo de desarrollo. Este conjunto de requerimientos tiene que llevarse a cabo por completo, y

solamente será cambiado o editado, luego de una revisión y validación con dueño del producto.

Sprint

Un *sprint* se define como un ciclo en el proceso de la metodología SCRUM. Hace referencia a una reunión en la que intervienen todos los roles para revisar el avance del *backlog*. Solamente en los *sprints* está permitido cambiar el *backlog*. Y la frecuencia del próximo *sprint* se definirá en uno anterior [37].

Pruebas de Aceptación

Es la etapa de validación de los criterios de aceptación definidos en las historias de usuario, con este paso, se da por hecho el desarrollo de la funcionalidad que se está validando.

1.9 Inteligencia de Negocios

La inteligencia de negocios o BI (*Business Intelligence*) por sus siglas en inglés, es un conjunto de estrategias, infraestructura, buenas prácticas y herramientas que sirven para analizar datos de cualquier proceso empresarial y con base en el análisis de esta información poder tomar las mejores decisiones [38].

Para el presente sistema, los datos que van a ser recolectados se enmarcarán a aquellos respecto de la PE. Estos datos serán transformados, agrupados y sistematizados por medio de una de las plataformas de BI más conocidas, la plataforma *Qlik*.

1.9.1 Plataforma QLIK

La plataforma *Qlik* es un conjunto de herramientas enfocadas a BI que tienen un modelo particular de asociación de datos llamado *Qlik Associative Model* (QAM) [39]. Este modelo permite una rápida respuesta en las consultas dentro de los datos asociados debido a que todas las tablas de los datos extraídos se guardan en memoria RAM en el servidor principal, así los datos están disponibles siempre que se desee y además se están relacionando todas las tablas entre sí, sin necesidad de realizar una consulta jerárquica con en un modelo de base de datos tradicional. Este modelo de asociación de datos es exclusivo de la plataforma *Qlik* y permite obtener un buen rendimiento para volúmenes grandes de información.

En la conferencia anual para usuarios de la plataforma *Qlik* de todo el mundo del año 2017 llamada *Qonnections* [40], se presentó el siguiente caso de éxito [41]: la empresa *Whirlpool Corporation* es una de las más grandes fabricantes de electrodomésticos en el mundo, uno de sus productos estrella es el robot de cocina *KitchenAid* con más de 12 accesorios opcionales y hasta 20 colores para elegir y el alto nivel de personalización de este producto presentaba un enorme desafío de ventas en su portal de *Amazon*. Con la ayuda de la plataforma *Qlik*, en menos de un año, pudieron obtener un conjunto de reportes de complejidad creciente, con jerarquía de datos multinivel e independencia de datos. Es decir que esta plataforma de BI permitió estandarizar la alfabetización de datos en toda la organización, analizar e interpretar al instante las tendencias en venta de este producto y sus componentes.

Se usarán dos componentes principales de la plataforma *Qlik*, para el presente sistema: el panel de control y la consola de administración. En el primero, se desarrollan y se visualizan los CMI, mientras que en la segunda se gestionan accesos de usuarios, tareas de actualización de datos automáticas y agrupación de los diferentes CMI desarrollados.

2 METODOLOGÍA

2.1 Diseño

A continuación, se detalla el diseño a realizar para la migración del software de PE de la empresa GRUPO NOVATECH.CÍA. LTDA. a un ambiente *cloud*. Como punto inicial y según la metodología SCRUM, se debe definir un *backlog* que servirá como guía para todo el diseño. Como siguiente paso, se deberá definir los requerimientos, tanto como desde la perspectiva del cliente como de la perspectiva del proveedor, es decir, para la empresa GRUPO NOVATECH CÍA. LTDA. Con estos requerimientos se podrá realizar las historias de usuario que detallarán los requerimientos. A continuación, se realizará la selección del proveedor que se usará para la infraestructura en la nube. Con la infraestructura definida se realizará el diseño de la capa de base de datos. El siguiente paso es el diseño de la capa de negocio en donde se definirán todos los servicios del software de PE. Finalmente se deberá realizar el diseño de la capa de presentación.

2.1.1 Definición del Backlog

Según la metodología SCRUM, es necesario definir los requerimientos en un documento. Este documento se lo llama *backlog* y en él residirán todas las peticiones que el cliente desea del software a diseñarse. Previo a ello es necesario definir los roles de las personas que intervendrán en el sistema para que sean ellas las encargadas de definir los requerimientos que formarán parte del *backlog*.

- **Dueño del Producto:** Jorge Barahona, Gerente de la Unidad de *Corporate Planning Management*, GRUPO NOVATECH CÍA. LTDA.
- **Director de SCRUM:** Ana Zambrano PhD.
- **Desarrollador:** Gonzalo Jiménez

En la Tabla 2.1 se muestra la valoración de los requerimientos que formarán parte del *backlog*.

Una vez establecidos los roles y las valoraciones, se procede a definir el *backlog* con los requerimientos deseados en términos generales, los mismos que serán detallados en sus correspondientes historias de usuario. En la Tabla 2.2 se muestra el *backlog* en su totalidad.

Tabla 2.1. Valoración de los requerimientos

Número	Prioridad	Complejidad
1	Baja	Fácil
2	Media	Media
3	Alta	Difícil
4	Muy Alta	Muy Difícil

Tabla 2.2. Definición del *backlog*

ID	Requerimiento	Prioridad	Complejidad
01	Diseño de base de datos	4	4
02	Diseño de lógica del negocio	4	4
03	Diseño de interfaz de ingreso de datos	4	4
04	Diseño de CMI	4	2
05	Selección de proveedor IaaS	2	3
06	Configuración de ambiente IaaS	3	2
07	Instalación de base de datos	3	3
08	Instalación de servidor web	3	3
09	Publicación de interfaz de ingreso de datos	3	3
10	Instalación de consola de administración de CMI	3	3
11	Implementación de diseño de base de datos	4	4
12	Implementación de diseño de lógica de negocio	4	4
13	Implementación de diseño de interfaz de ingreso de datos	4	4
14	Implementación de diseño de CMI	4	4
15	Pruebas de funcionamiento	3	2

2.1.2 Sprint de Requerimientos

A partir del *backlog* se deben definir las historias de usuarios para los requerimientos desde la perspectiva del cliente y desde la perspectiva del proveedor.

Para el sistema, el director del SCRUM deberá recolectar de parte del dueño del producto los requerimientos tanto del cliente como del proveedor. En el primer caso, con base en las experiencias de comercialización anteriores y en el segundo caso como parte de la empresa proveedora de servicio. Según la metodología, esto se lo debe realizar por medio de un *sprint*.

a. Historias de Usuario desde la Perspectiva del Cliente

Desde la Tabla 2.3 hasta la Tabla 2.8, se describirán las diferentes historias de usuario desde la perspectiva del cliente que definirán de manera global los requerimientos del software a diseñarse. Cabe resaltar que cada una de las siguientes historias de usuario tendrá un identificador (ID) único.

Tabla 2.3. Historia de usuario de acceso ubicuo

Historia de Usuario	
ID: 01-C	Usuario: Cliente
Nombre de Historia: Acceso ubicuo	
Prioridad: Alta	Riesgo: Medio
Desarrollador: Gonzalo Jiménez	
Descripción: El acceso al software se lo podrá realizar desde cualquier lugar y a cualquier momento.	
Observaciones:	
Validación: Pruebas de acceso desde diferentes navegadores, horarios, lugares.	

Tabla 2.4. Historia de usuario de perfiles de usuario

Historia de Usuario	
ID: 02-C	Usuario: Cliente
Nombre de Historia: Perfiles de Usuario	
Prioridad: Media	Riesgo: Alto
Desarrollador: Gonzalo Jiménez	
Descripción: El acceso al software se lo podrá realizar con varios perfiles de usuarios.	
Observaciones: Los perfiles solicitados son: Administrador, Facilitador, Gestor, Colaborador.	
Validación: Pruebas de acceso con los diferentes perfiles de usuario. Cuando se cree un nuevo usuario, se podrá escoger qué perfil de usuario será posible usar.	

Tabla 2.5. Historia de usuario de perfil Administrador

Historia de Usuario	
ID: 02.1-C	Usuario: Cliente
Nombre de Historia: Perfil Administrador	
Prioridad: Alta	Riesgo: Alto
Desarrollador: Gonzalo Jiménez	
Descripción: El administrador del software podrá crear indicadores, ingresar los valores iniciales, configurar los parámetros iniciales, añadir nuevos parámetros, extender el tiempo de ingreso de los valores mensuales. Además, asignará los indicadores a su respectivo Facilitador y Gestor.	
Observaciones:	
Validación: Se deberá tener un formulario de creación de nuevos indicadores en donde se asignen los facilitadores y los gestores, un formulario de ingreso de valores iniciales, un menú de parámetros (fórmulas, unidades, días de plazo, frecuencias), un botón de extensión del plazo de ingreso de datos.	

Tabla 2.6. Historia de usuario de perfil Facilitador

Historia de Usuario	
ID: 02.2-C	Usuario: Cliente
Nombre de Historia: Perfil Facilitador	
Prioridad: Media	Riesgo: Medio
Desarrollador: Gonzalo Jiménez	
Descripción: El Facilitador ingresará los valores reales mensuales, los planes de acción, las tareas de cada plan y el avance de cada tarea. Además, asignará los planes de acción a su respectivo Colaborador.	
Observaciones: El Facilitador es designado a cada indicador por el Administrador.	
Validación: Se deberá tener un formulario de ingreso de valores mensuales, un formulario de creación de planes de acción, un formulario de creación de tareas de cada plan de acción, un formulario de registro de avances a las tareas de cada plan de acción, y un menú de asignación de un colaborador a un indicador.	

Tabla 2.7. Historia de usuario de perfil Colaborador

Historia de Usuario	
ID: 02.3-C	Usuario: Cliente
Nombre de Historia: Perfil Colaborador	
Prioridad: Media	Riesgo: Bajo
Desarrollador: Gonzalo Jiménez	
Descripción: El Colaborador da por finalizadas las tareas de cada plan de acción.	
Observaciones: El Colaborador es designado a cada plan de acción por el Facilitador.	
Validación: Se deberá tener un formulario de revisión de los avances registrados y un menú para dar por finalizada alguna tarea asociada.	

Tabla 2.8. Historia de usuario de perfil Gestor

Historia de Usuario	
ID: 02.4-C	Usuario: Cliente
Nombre de Historia: Perfil Gestor	
Prioridad: Baja	Riesgo: Bajo
Desarrollador: Gonzalo Jiménez	
Descripción: El Gestor aprueba el valor mensual de cada indicador ingresado.	
Observaciones: El Gestor es designado a cada indicador por el Administrador.	
Validación: Se deberá tener un formulario de aprobación de cada uno de los valores mensuales ingresados por el facilitador.	

Como se puede observar, a excepción de la primera historia de usuario, el resultado de este conjunto de historias de usuario desde la perspectiva del cliente, son requerimientos funcionales.

b. Historias de Usuario desde la Perspectiva del Proveedor

Desde la Tabla 2.9 hasta la Tabla 2.13, se describirán las diferentes historias de usuario desde la perspectiva del proveedor que definirán de manera global los requerimientos del software a diseñarse. Cabe resaltar que, de igual manera que las anteriores, cada una de las siguientes historias de usuario tendrá un ID único.

Como se puede observar, el resultado del conjunto de historias de usuario desde la perspectiva del proveedor, son requerimientos no funcionales.

2.1.3 Sprint de Selección de Proveedor IaaS

Se va a realizar el comparativo entre los diferentes proveedores de IaaS sobre el cual va a residir el software de PE. Para ello se ha tomado en cuenta a los proveedores de IaaS que son más reconocidos en el cuadrante de Gartner [42]. Para junio de 2015, se tienen tres empresas líderes en lo que respecta a proveedores IaaS: *Amazon Web Services (AWS)* [11], *Microsoft Azure (MA)* [12] y *Google Cloud Engine (GCE)* [13].

Tabla 2.9. Historia de usuario de instalación única

Historia de Usuario	
ID: 01-P	Usuario: Proveedor
Nombre de Historia: Realizar una única instalación del software en la arquitectura centralizada en la infraestructura <i>cloud</i> .	
Prioridad: Alta	Riesgo: Medio
Desarrollador: Gonzalo Jiménez	
Descripción: La instalación del software se lo realizará una única vez en la arquitectura central y a continuación se dará acceso solamente a los nuevos clientes.	
Observaciones:	
Validación: Verificar que la instalación y configuración se la realice en el ambiente IaaS contratado.	

Tabla 2.10. Historia de usuario de ambiente IaaS

Historia de Usuario	
ID: 02-P	Usuario: Proveedor
Nombre de Historia: Realizar la administración del software desde un solo servidor.	
Prioridad: Media	Riesgo: Medio
Desarrollador: Gonzalo Jiménez	
Descripción: La administración del software se la realizará desde el ambiente IaaS para todos los clientes.	
Observaciones:	
Validación: Verificar que la administración se la realice en el ambiente IaaS contratado sin limitar el número de clientes.	

Tabla 2.11. Historia de usuario de proyecto demostrativo

Historia de Usuario	
ID: 03-P	Usuario: Proveedor
Nombre de Historia: Proyecto demostrativo.	
Prioridad: Media	Riesgo: Bajo
Desarrollador: Gonzalo Jiménez	
Descripción: Tener un proyecto demostrativo para realizar las preventas a posibles nuevos clientes.	
Observaciones: El proyecto demostrativo tendrá las mismas características que un proyecto real pero con datos ficticios.	
Validación: Pruebas funcionales con el proyecto demostrativo.	

Tabla 2.12. Historia de usuario de administración del cliente

Historia de Usuario	
ID: 04-P	Usuario: Proveedor
Nombre de Historia: Administración por parte del cliente.	
Prioridad: Media	Riesgo: Bajo
Desarrollador: Gonzalo Jiménez	
Descripción: El cliente solamente administrará los parámetros propios de la interfaz de ingreso de datos, lo demás estará a cargo del proveedor.	
Observaciones:	
Validación: Usuario con perfil de administración hará uso de la interfaz de ingreso de datos en un menú de nuevos parámetros asociados solamente a la lógica del negocio.	

Tabla 2.13. Historia de usuario de actualización de capa presentación

Historia de Usuario	
ID: 05-P	Usuario: Proveedor
Nombre de Historia: Actualización de la capa de presentación por parte del proveedor.	
Prioridad: Alta	Riesgo: Medio
Desarrollador: Gonzalo Jiménez	
Descripción: Realizar la codificación de la capa de presentación de los indicadores con la versión más actual de su software.	
Observaciones:	
Validación: Se deberá ver que el CMI está desarrollado en la última versión de la plataforma <i>Qlik</i> .	

A continuación, se realizarán las comparaciones que nos permitirán elegir el proveedor IaaS a contratar.

a. Comparación entre Diferentes Proveedores

Para realizar un comparativo equitativo entre las diferentes alternativas primero se deben enumerar los requerimientos mínimos de la arquitectura y se debe escoger una instancia virtual que cumpla con estos requerimientos. En la Tabla 2.14 se muestran los recursos necesarios para la instancia virtual a escoger.

Tabla 2.14. Recursos mínimos para la instancia virtual del proveedor IaaS

Recursos	Requerimiento Mínimo
Sistema Operativo	<i>Windows Server 2008 R2</i>
Procesadores	2
Disco duro	500 GB
Memoria RAM	8 GB
Navegador web	<i>Internet Explorer 10</i> o similares
Base de datos	<i>SQL Server 2008 R2 Express</i>

Para cumplir con estos requerimientos mínimos se va a escoger las máquinas virtuales que están a disposición de cada uno de los proveedores IaaS. En la Tabla 2.15 se puede ver las opciones que cumplen al menos con dos de los requerimientos. Se ha resaltado con negrita aquella opción que cumple con la mayor cantidad de los requerimientos.

Tabla 2.15. Comparativo técnico entre proveedores IaaS

	<i>Amazon Web Services (AWS)</i>	<i>Microsoft Azure (MA)</i>	<i>Google Cloud Engine (GCE)</i>
Nombre	m3.xlarge	D2S v3	n1-standard-4
Sistema Operativo	<i>Windows Server 2008 R2</i>	<i>Windows Server 2008 R2</i>	<i>Windows Server 2008 R2</i>
CPU [unidades]	4	2	4
RAM [GB]	15	16	15
Almacenamiento [GB]	190	150	64000
Ampliación de almacenamiento	Solo con <i>Elastic Book Store (EBS)</i>	Instancia con mayores recursos	No necesario
Navegador web	<i>Internet Explorer 10</i> o superiores	<i>Internet Explorer 10</i> o similares	<i>Internet Explorer 10</i> o similares
Base de datos	<i>SQL Server 2008 R2 Express</i>	<i>SQL Server 2008 R2 Express</i>	<i>MySQL</i>
Disponibilidad [%]	99.95	99.95	99.95
Tiempo sin servicio [minutos]	04:23	04:23	04:23
Compensación [%]	10 - 30	10 – 25	10 - 50
Minutos consecutivos	1	1	5
Precio	Alto	Medio	Bajo
Costo de soporte	Alto	Bajo	Alto

Como se puede observar, el proveedor GCE queda descartado en primer lugar debido al hecho de que sus máquinas virtuales no permiten el uso de la base de datos SQL de *Microsoft*. Con la comparación entre AWS y MA se puede comprobar que ambas presentan casi el mismo nivel de funcionalidades, con la excepción de la limitación en almacenamiento. Y la principal diferencia está en que en MA se puede aumentar los

recursos de almacenamiento solamente subiendo de tamaño en la jerarquía de máquinas virtuales, mientras que, en AWS para aumentar el almacenamiento se tiene que contratar otro servicio de la misma empresa, llamado *Elastic Book Store* (EBS) [43], con un costo adicional . Al análisis previo se va a añadir la Tabla 2.16, en la cual se muestra un comparativo a nivel funcional [44].

Tabla 2.16. Comparativo funcional entre los proveedores IaaS

Proveedor	Pros [45]	Contras [45]	Adecuación Técnica
AWS	Alta experiencia, muchas funcionalidades, innovación constante, herramientas de integración externas, consultores y expertos	Demasiadas opciones de implementación, contratar servicios externos, actualizaciones continuas	Media
MA	Arquitectura estable, acuerdos con proveedor, integración con <i>Office 365</i> , posibilidad de nubes híbridas	Complejidad en su uso, pocos consultores y expertos, no tienen fuentes abiertas	Alta
GCE	Basada en arquitectura de <i>Google</i> , servicios de <i>Big Data</i> , compatibilidad con otros IaaS, descuentos en precio	Menos funciones, IaaS menos madura	Baja

Un concepto que cabe aclarar de la Tabla 2.15 es respecto a la “Compensación” [46]. Este concepto va de la mano del siguiente que se llama “Minutos consecutivos”, y el porcentaje de compensación que es el descuento mensual que aplica para la tarjeta de crédito en la siguiente facturación, cuando haya existido una caída del servicio de los minutos consecutivos mencionados. El mayor porcentaje lo tiene GCE.

Para el sistema se ha escogido como proveedor IaaS a *Microsoft Azure*. No solo porque es la que más se adecua de manera técnica sino porque también en el análisis funcional se ha determinado que sus puntos en contra no son tan cruciales [47]. Ya que, al ser el proveedor asociado con *Microsoft* en la comercialización de otras herramientas, se tiene un acceso gratuito y soporte general para su servicio con *Azure*. Y además, el no tener compatibilidad con sistemas abiertos no afecta debido a que la herramienta *Qlik*, que se necesita para la capa de presentación, solamente puede ser instalada en un servidor exclusivamente con un sistema operativo *Windows*. Un punto a favor es que el proceso de aumento de recursos en las máquinas de *Azure* es en vivo, con el correspondiente aumento del precio. Esto se verá más claramente en la Sección 2.2.1.

2.1.4 Sprint del Flujo de Información

Antes de realizar el diseño de las capas que forman parte del sistema, es necesario realizar el *sprint* del flujo de la información de manera global para tener claridad de los componentes y de cómo interactúan.

a. Diagrama de Secuencia

A continuación, se presentan los diagramas de secuencia, con los cuales se define el diseño de la interacción de los componentes que forman parte del sistema en función del tiempo [48].

En la Figura 2.1 se presenta el diagrama de secuencia del flujo de los datos. Aquí se observa las acciones que realizan tanto el cliente como el proveedor, en conjunto con la interfaz web, la base de datos y el CMI.

El cliente es el encargado de ingresar los datos a la base de datos por medio de la interfaz web previa su autenticación, esta información será guardada en la base de datos. El cliente visualizará los datos en el CMI. El CMI siempre tendrá los datos actualizados al menos con un día de anterioridad. Para visualizar el CMI también es necesario una autenticación previa.

El proveedor, por su parte, debe realizar la tarea automática de carga de datos en la consola de administración del CMI. Esto lo realizará una sola vez de forma asincrónica, es decir, a cualquier momento en el transcurso del tiempo del flujo de datos. Como resultado, se tendrá que el CMI contendrá los datos actualizados de forma diaria para que el cliente pueda analizarlos.

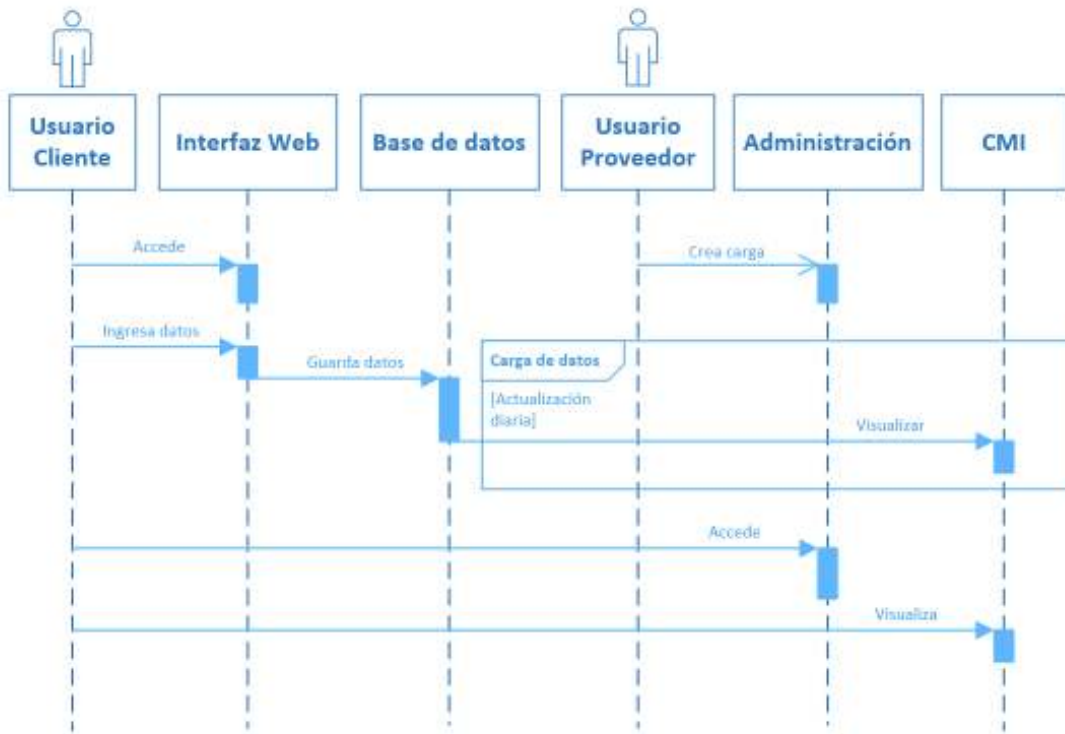


Figura 2.1. Diagrama de secuencia del flujo de datos

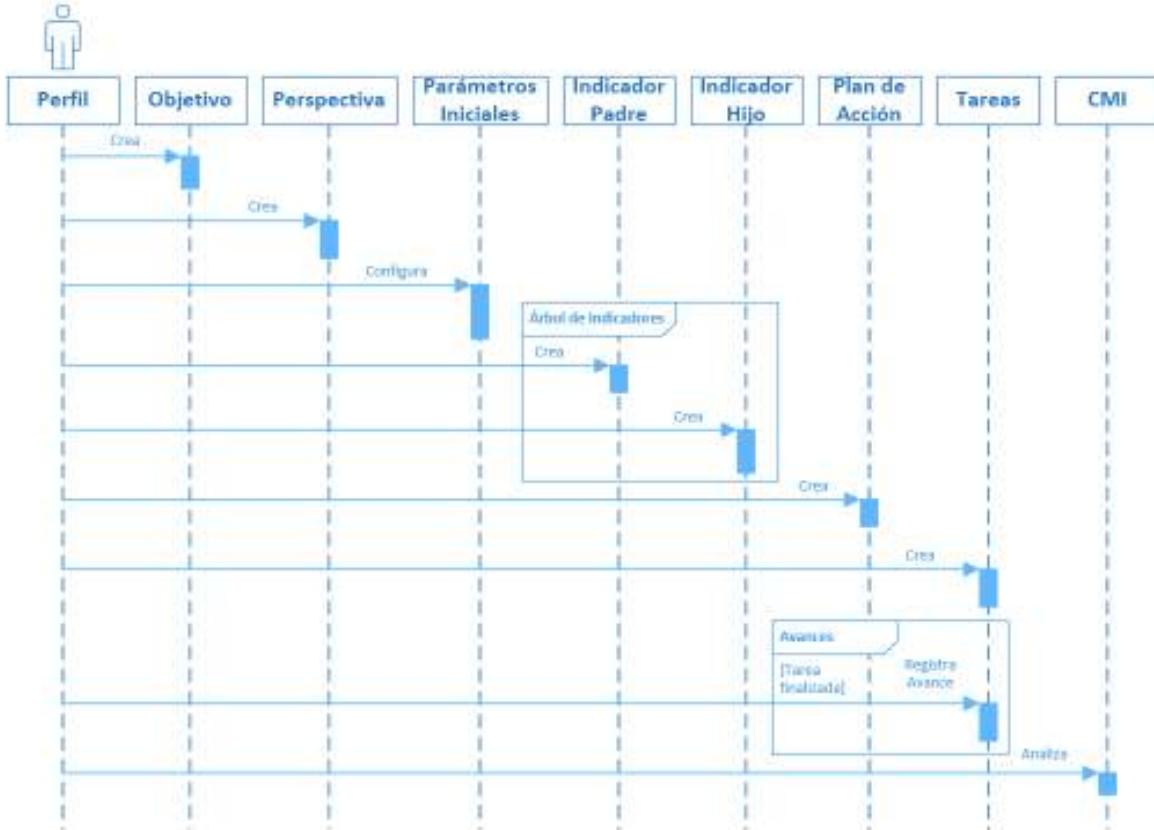


Figura 2.2. Diagrama de secuencia de ingreso y análisis de datos

En la Figura 2.2 se muestra el proceso de ingreso y análisis de datos que se tendrá en el sistema por parte de un usuario. Este usuario, podrá tener un perfil de Administrador, Facilitador, Gestor o Colaborador, dependiendo de las acciones a realizar.

De forma general, un usuario debe crear un objetivo. Cuando se habla de un objetivo, se hace referencia a los OE que forman parte de la PE. Como siguiente paso, el usuario debe crear una perspectiva. Cuando se habla de una perspectiva, se hace referencia a las perspectivas financieras que definen el esquema de la PE. Cuando han sido creadas las perspectivas, son agrupadas en uno o varios objetivos, creando una red sobre la que se van a desplegar el resto de componentes.

El siguiente paso es la configuración de los parámetros iniciales del software de PE. Para esto, es necesario crear algunos campos que serán necesitados más adelante. En la Sección 2.1.6a con los casos de uso del perfil de Administrador, se podrá observar con más detalle acerca de los parámetros iniciales. El siguiente paso se refiere a la creación de indicadores. Cuando se habla de indicadores se hace referencia a los diferentes KPI que conforman la PE y sobre los cuales se medirá el cumplimiento de las metas de toda la PE.

Es necesario crear los indicadores llamados *padre*, que pertenecerán a alguna de las perspectivas creadas anteriormente. Estos indicadores padres, no tienen ninguna diferencia del resto de indicadores, se los ha llamado así solamente para hacer notar que, a través de los indicadores hijos, se puede realizar una jerarquía de indicadores o, como se lo conocerá desde el diseño de la capa de base de datos, el árbol de indicadores. Por eso se define un bucle con este nombre. Los indicadores tienen muchos campos característicos que están definidos en la Sección 2.1.5a en el diagrama entidad – relación de la capa de base de datos.

El siguiente elemento por crear es el plan de acción, y como su nombre mismo lo indica, es un elemento que debe ser creado cuando se requiera una acción complementaria sobre un indicador con el objetivo de cumplir las metas propuestas. Es un elemento opcional pero la mayoría de los indicadores lo tiene. De igual manera que los indicadores, un plan de acción tiene varios campos característicos que están definidos en la Sección 2.1.5a.

Cuando se tiene un plan de acción, se debe crear tareas que van siendo realizadas conforme se va cumpliendo las metas del indicador. Las tareas son elementos más sencillos que a su vez están compuestas por registros de avances que los va ingresando el usuario. Esto se verá a detalle en la Sección 2.1.6a.

Finalmente, el usuario realizará el análisis de todos estos elementos, por medio del CMI. Todo lo que respecta al CMI está definido en la Sección 2.1.7a del diseño de la capa de presentación.

El diseño de los perfiles se verá en la Sección 2.1.6a con los casos de uso. El diseño de la base de datos será en *Microsoft SQL Server*, el diseño de la interfaz web será en lenguaje *PHP, HTML y CSS*; el diseño del CMI y su consola de administración será en *Qlik*.

2.1.5 Sprint de Diseño de la Capa de Base de Datos

Para el diseño de la capa de base de datos se ha realizado un diagrama entidad – relación en donde constan las tablas (entidad) con sus respectivos campos y cómo estas tablas se relacionan unas con otras por medio de campos en común (relaciones). Cada relación tiene una cardinalidad específica, es decir, el número de tablas con las cuales se pueden relacionar.

a. Diagrama Entidad - Relación

En el Figura 2.3 se puede ver el diagrama entidad – relación completo [49]. La base del diagrama son los indicadores, de los cuales parten el resto de tablas. Cada indicador está colocado en un árbol de indicadores, este árbol de indicadores forma la jerarquía de los indicadores, y cada árbol de indicadores pertenece a un objetivo. Los objetivos, a su vez, forman parte de un departamento específico. Tanto el árbol de indicadores como cada uno de los indicadores tienen asociado el campo despliegue, esta es una característica referente al alcance geográfico. Para ello, se tiene un grupo de niveles de despliegue que hace referencia a los niveles geográficos a los cuales cada indicador sea definido.

Cada indicador tiene características específicas que son: segmentos, fórmulas, productos, extensiones del ingreso de datos, unidades, frecuencias, usuarios, e incluso, se podrían crear catálogos personales, llamados campos adicionales. Existen controles de cambio que se hacen sobre un indicador y deben tener un usuario que las realice.

Cada indicador puede tener o no, un plan de acción. Este plan de acción, a su vez, puede tener o no tareas. Las tareas también pueden tener o no diferentes avances. Las tablas de planes de acción, tareas y avances tienen relación con la de usuarios y de cargos debido a las personas que serán responsables. Ya que como parte de los requerimientos se solicitó que existan diversos perfiles de usuario.

2.1.6 Sprint de Diseño de la Capa de Negocio

Sobre la capa de base de datos, se debe diseñar una capa que detalle los servicios a los cuales recurrirá la capa de presentación; a esta capa se la llama la capa de negocio.

Para poder definir la capa de negocio se ha escogido los diagramas UML [50] que ilustran de mejor manera los requerimientos solicitados debido a que “representa la funcionalidad que ofrece el sistema en lo que se refiere a su interacción externa” [51].

a. Casos de Uso

Para tener un detalle más preciso de los requerimientos funcionales, es necesario crear casos de uso en donde se puedan observar las diferentes interacciones que tendrán los componentes del sistema, con el objetivo de cumplir los requerimientos mencionados [52].

En la Figura 2.4 se muestra el caso de uso del Administrador. El administrador podrá crear usuarios nuevos para que puedan acceder al sistema, el identificador de cada usuario será su correo electrónico.

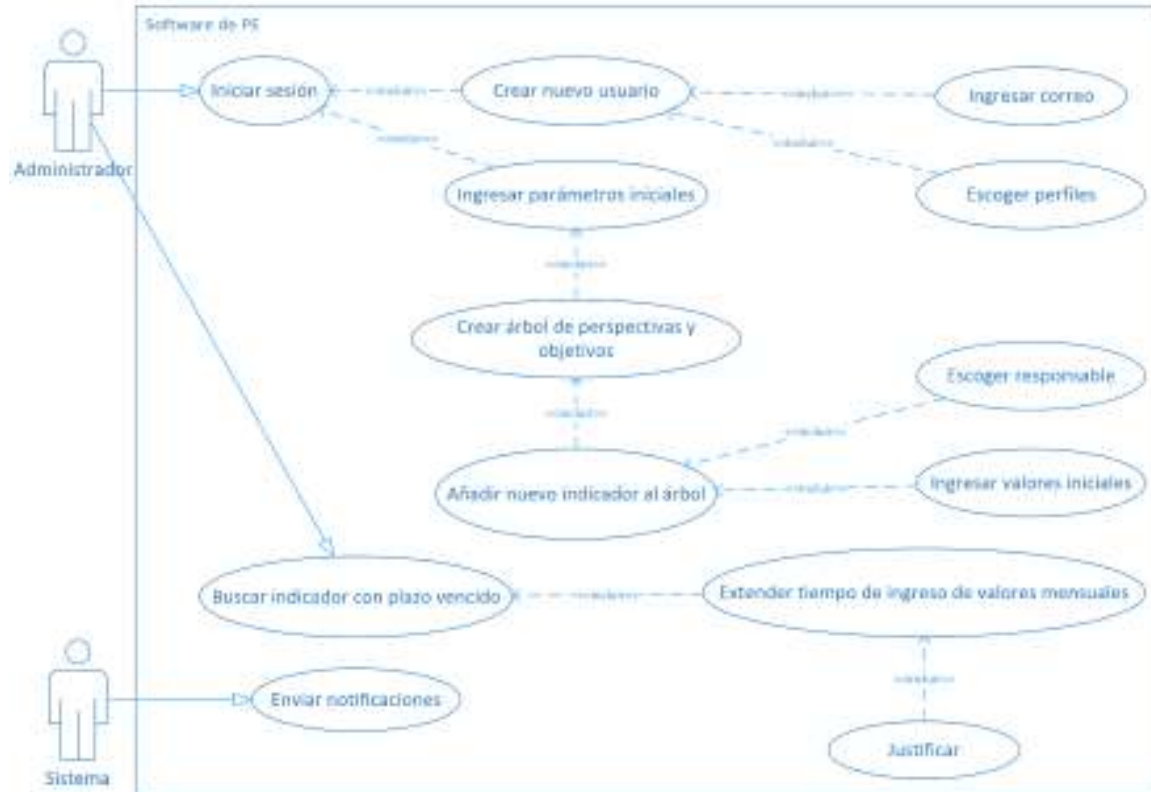


Figura 2.4. Caso de uso del Administrador

Un usuario podrá tener más de un perfil, de acuerdo con lo que decida el Administrador, y, una vez finalizado el proceso, el sistema enviará una notificación por correo electrónico al usuario asignado en el cual se mostrarán los perfiles a los que pertenece en conjunto con una contraseña temporal. En la misma figura se observa que el Administrador tiene que ingresar los parámetros iniciales del sistema. Muchos de estos parámetros iniciales no son obligatorios configurarlos la primera vez que se ingresa. Los únicos campos que sí se deben configurar, son los parámetros iniciales de indicadores. Esta parametrización de indicadores se refiere al rango de fechas en las cuales están habilitados todos los indicadores para permitir el ingreso de valores mensuales y también, el número de días extras que podrán ser extendidos una vez se haya cumplido el plazo.

A continuación, el Administrador puede añadir nuevos indicadores a un árbol de indicadores. Previamente se debería crear la jerarquía de perspectivas y objetivos, ya que es con base en esta jerarquía sobre la cual se irán añadiendo los diferentes indicadores, o árbol de indicadores. Luego de añadir el indicador o los indicadores, es necesario ingresar los valores iniciales del mismo, además de escoger el usuario responsable del mismo indicador. Esto generará una notificación al usuario asociado por parte del sistema, en donde se informará que se lo ha escogido como responsable de un indicador. Finalmente, el Administrador podrá extender el tiempo de ingreso de los valores mensuales de cada indicador, cuando las fechas se hayan cumplido, y el Facilitador aún tenga que ingresar los valores al indicador del que sea responsable. Esta extensión del tiempo se lo realiza por cada indicador debido a que es necesaria una justificación de la extensión solicitada.



Figura 2.5. Caso de uso del Facilitador

En la Figura 2.5 se muestra el caso de uso del Facilitador. En este caso de uso se ve el proceso de ingreso de los valores mensuales del indicador que sea responsable. Obviamente, este ingreso solamente se lo podrá realizar dentro del rango de fechas disponibles para todos los indicadores. Si no se realizó el ingreso de los valores mensuales en el momento adecuado, el Administrador podrá extender el plazo de los días para el ingreso de los valores mensuales. Además, el Facilitador es el encargado de la creación de un nuevo plan de acción. Cada plan de acción está asociado a un solo indicador cuyo responsable es el Facilitador. A continuación, el Facilitador debe realizar el ingreso de una nueva tarea. Cada tarea está asociada a un solo plan de acción, así como anteriormente se vio que cada plan de acción está asociado a un solo indicador. Finalmente, el Facilitador debe crear un nuevo avance. Cada avance está asociado a una sola tarea. El Facilitador podrá ingresar tantos avances como necesite.

En la Figura 2.6 se puede ver el caso de uso del Colaborador. Este actor tiene como función el dar por finalizadas las tareas creadas por el Facilitador. En este caso de uso se puede visualizar que el Colaborador puede ingresar en la jerarquía creada por el usuario Facilitador con el fin de revisar todos los avances y con esta información dar por finalizadas las tareas que conforman un plan de acción.



Figura 2.6. Caso de uso del Colaborador



Figura 2.7. Caso de uso del perfil Gestor

En la Figura 2.7 se ve el caso de uso del Gestor. El Gestor está encargado de aprobar los valores mensuales ingresados por el perfil Facilitador.

b. Diagrama de Clases

Un diagrama de clases presenta la estructura del sistema en un lenguaje orientado a objetos y de manera gráfica. Estas clases serán las que usará el sistema para todos los servicios que permitirán el intercambio de datos entre la capa de base de datos y la capa de presentación.

En la Figura 2.9 se muestra el diagrama de clases completo. Está estructurado en tres niveles: bases de datos, lógica de tablas e interfaz web.

En la Figura 2.8 se puede ver el detalle del nivel base de datos. En este nivel solamente se tiene una clase, la clase llamada *Ado*, y que es la encargada de comunicarse directamente con la base de datos a través de sentencias *SQL*.

Como se puede observar a través de las relaciones de dependencia [53], todas las clases del segundo nivel, es decir del nivel de lógica de tablas, tienen como referencia a la clase *Ado* para su funcionamiento.

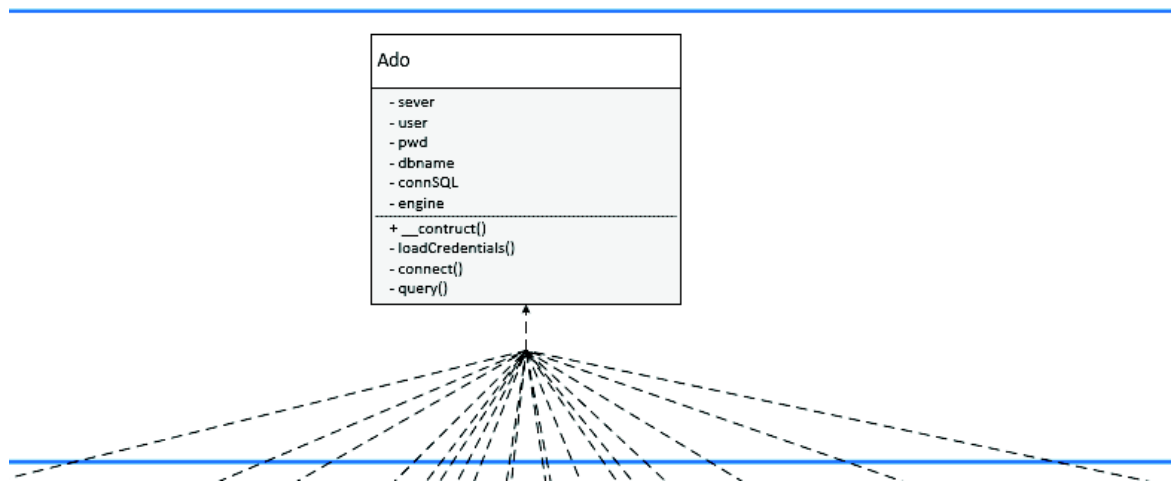


Figura 2.8. Nivel de base de datos del diagrama de clases

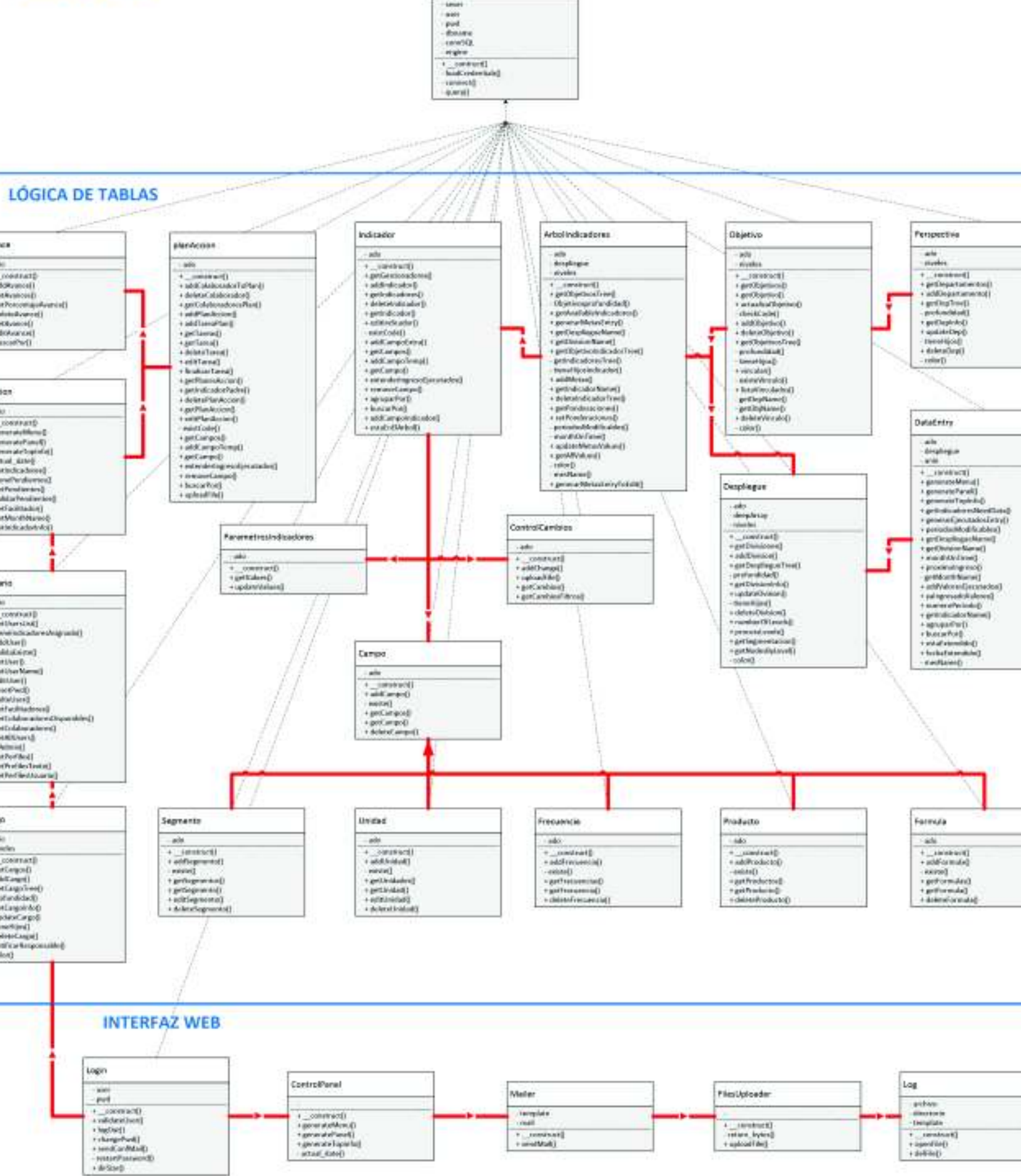


Figura 29. Diagrama de Clases

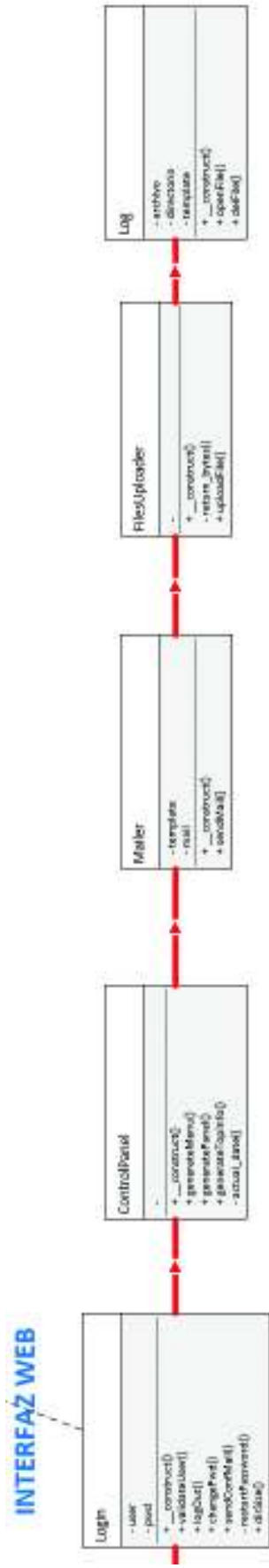


Figura 2.11. Nivel de interfaz web del diagrama de clases

En la Figura 2.10 se puede ver el detalle del nivel de la lógica de tablas. Aquí se muestran, las clases que definen el funcionamiento del software de PE. Y que están ligados a ciertas tablas de las bases de datos.

Por ejemplo, la jerarquía de los datos se mantiene, es decir: una perspectiva tiene un objetivo, un objetivo tiene a su vez un árbol de indicadores, cada uno de esos indicadores podría tener un plan de acción, y estos planes de acción pueden tener tareas con avances. Otra jerarquía que se puede observar es a nivel de los usuarios: los planes de acción son administrados por un usuario de gestión, que a su vez se relaciona con la clase usuarios, y esta clase usuario puede tener uno o varios cargos. Cada árbol de indicadores tiene a su vez relación con la clase despliegue geográfico.

Existe un conjunto de clases asociadas a la clase indicadores llamada parámetros de indicadores, que se refiere a los valores de control sobre las fechas de cumplimiento; control de cambios, que se refiere a los registros de cambios de datos por los usuarios; y la clase campo que tiene a su vez una relación de generalización con las clases segmento, unidad, frecuencia, producto y fórmula; todas ellas representan atributos propios de cada indicador.

Hay una clase que se usa para el ingreso de valores de indicadores por año que se llama *DataEntry* y permite ingresar un solo valor o muchos valores a la vez, por eso solamente se relaciona con la clase de despliegue. En la clase despliegue se definen si los indicadores se relacionan con uno o más elementos geográficos, como pueden ser: países, regiones, provincias, cantones, etc.

En la Figura 2.11 se puede ver el detalle del nivel de la interfaz web. En este grupo de clases a diferencia del grupo anterior, no tienen una dependencia de comunicación con la base de datos ya que se usan para generar funciones propias de la interfaz web, es decir que tienen una dependencia de comunicación con el usuario. Por ejemplo, la clase *Login* permite la correcta validación de acceso, la clase de *Control Panel* genera los menús desplegables según se necesita, la clase *Mailer* es la encargada del envío de correos, la clase *FilesUploader* es la que permite adjuntar documentos y la clase *Log* guarda los registros de actividades en un archivo plano.

2.1.7 Sprint de Diseño de la Capa de Presentación

La capa de presentación se la realizará con base en la herramienta llamada *Qlik*. *Qlik* es la plataforma de *Business Intelligence* (BI) que comercializa la empresa GRUPO NOVATECH CÍA. LTDA.

La plataforma *Qlik* tiene dos componentes principales: la consola de administración y el centro de control en donde el proveedor desarrollará los CMI de cada cliente. El centro de control permite la creación de los CMI a través de un lenguaje de programación propio de *Qlik* y también permite el diseño de los KPI que verán los clientes en el CMI. Este lenguaje propio solamente maneja funciones específicas para el desarrollo en *Qlik*, la lógica de programación es similar a cualquier otro lenguaje conocido. En la consola de administración se realiza la gestión del CMI como son: permisos de acceso a los usuarios, tareas automáticas de actualización de datos, gestión de respaldos, etc. A continuación, se diseñará la capa de presentación del CMI y de su administración usando la plataforma *Qlik*.

a. Cuadro de Mando Integral

En el presente sistema, el cuadro de mando integral o CMI, es el resultado de la programación realizada en el módulo de desarrollo de *Qlik*. En la Figura 2.12 se muestran las tres etapas de programación en el centro de control de *Qlik*.

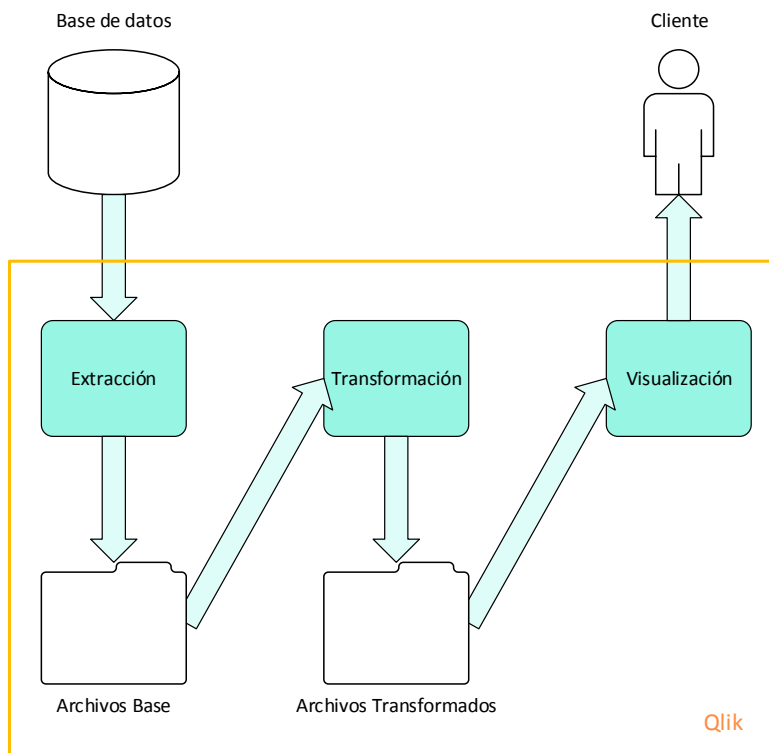


Figura 2.12. Etapas de desarrollo del CMI

No se debe confundir las tres etapas de codificación del CMI con las tres capas del diseño del presente proyecto. Se hace una diferencia explícita entre etapa de codificación y capa de diseño principalmente por el alcance de cada una. Una etapa de

codificación sencillamente es una separación en el proceso de desarrollo del CMI que se la podría hacer en una solo archivo *Qlik*, pero se los separa por un estándar propio usado en la empresa GRUPO NOVATECH CÍA. LTDA.

Cada una de estas etapas consta de dos tipos de archivos, los archivos en donde consta el *script* de la programación y los archivos propios de datos. Ambos archivos son exclusivos de la plataforma *Qlik* y solo podrán ser usados, abiertos o editados en la misma herramienta.

La primera etapa se llama Extracción y, como su nombre lo indica, carga la información que se encuentra en la base de datos hacia los archivos propios de *Qlik* para su posterior utilización. En la etapa de Extracción se deben leer las mismas tablas de las bases de datos sin cambio alguno.

La segunda etapa se llama Transformación y su función es modificar la información de la base y guardarla en los archivos finales cuyos datos estarán listos para usarse. La etapa de Transformación genera una nueva lógica de programación, a esta nueva asociación de tablas se la conoce como nube de datos.

La tercera y última etapa se llama Visualización y contiene toda la parte gráfica del CMI para que el cliente puede analizar los datos ingresados. En esta etapa se crean los indicadores de la PE.

En la Figura 2.13 se muestra el detalle de la nube de datos del CMI. Guarda mucha relación con las tablas originales, pero ahora ya no se tienen algunas de las tablas de la parametrización inicial. Según la necesidad se han creado o eliminado los campos que conforman las tablas iniciales.

En la plataforma *Qlik* la manera de relacionar las tablas de la nube de datos es solamente teniendo dos campos con el mismo nombre, por eso se pueden crear relaciones entre varias tablas como por ejemplo por el campo *Id_Objetivo*. Además, se ha creado una tabla completamente nueva que se llama *Periodo*, la cual sirve como un calendario.

Para la diferenciación de las diferentes tablas que conforman la nube de datos se ha usado dos prefijos. El prefijo *fc_* significa que se trata de una tabla transaccional, es decir, un conjunto de datos con fechas y con valores numéricos. El prefijo *dm_* significa que es una tabla que se conoce como catálogo, es decir, son datos anexos a una tabla transaccional. Generalmente las tablas *dm_* son mucho más pequeñas que las *fc_*.

- 4) Un conjunto de páginas web que serán publicadas en el servidor web y formarán la interfaz de ingreso y visualización de los datos de la PE de cada cliente.
- 5) La plataforma *Qlik* en donde se desarrollará y visualizará los CMI para cada cliente. Además, que se deberá configurar para que actualice automáticamente la información de la PE ingresada en los CMI.

2.1.9 Situación actual del software de PE

Con base en todo el diseño detallado anteriormente, es necesario aclarar la situación actual del software de PE y contrastarlo con el alcance del presente trabajo de titulación.

El cliente no tendrá una posibilidad de aumentar sus recursos directamente, este proceso tendrá la intervención del proveedor, debido a razones principalmente relacionadas con el manejo de la diversidad de información de cada cliente, el control de cada uno de los proyectos y temas relacionados a pago por uso, que la empresa GRUPO NOVATECH CÍA. LTDA. quiere mantener bajo su dominio. Al ser este una primera versión del prototipo SaaS la migración no es en su totalidad. Eso no significa que la implementación será un rehacer desde cero toda la base de datos, el software de PE y el CMI. Muchos de los componentes serán reusados para mantener el funcionamiento original de software. Sin embargo, en las historias de usuario se han podido ampliar estas funcionalidades.

Debido al alcance definido en la Sección 1.2 se debe tomar en cuenta que estos procesos que no se van a automatizar, se los manejará así por pedido exclusivo de la empresa GRUPO NOVATECH CÍA. LTDA. Aspectos como el hecho de tener un proyecto por cada cliente, la seguridad de su transmisión en los datos, el financiamiento del pago por uso a los clientes y al proveedor IaaS contratado, así como también, el licenciamiento de la plataforma *Qlik* no forman parte del alcance y por ende no se los verá al detalle.

Sin embargo, el proceso de migración ha sido validado por el equipo conformado según la metodología, quienes han revisado que los puntos mencionados en la Sección 1.6.5 acerca del proceso de migración hayan sido cumplidos.

2.2 Implementación

A continuación, se muestra el proceso que se llevará a cabo para la implementación del sistema. Se detallarán los pasos que se siguió para configurar el ambiente IaaS, se mostrará el proceso de la implementación de la capa de base de datos, el cual incluye la instalación de la base de datos, configuración de la misma y la codificación del diseño; se realizará la implementación de la capa de negocios, es decir, la instalación del servidor web, la configuración del mismo y la codificación de la interfaz web; finalmente, se mostrará el proceso de implementación de la capa de presentación, la que consta de la instalación de la plataforma *Qlik*, la configuración de la misma, la codificación del CMI y de la interfaz web.

2.2.1 Sprint de Instalación y Configuración de Plataforma IaaS

Como primer punto, se procede a crear una máquina virtual en la plataforma MA de *Microsoft*. Para esto, se ingresa al portal y se escogen las características necesarias de la máquina virtual que se desee. En la Figura 2.14 se puede observar el panel de control de *Azure* sin ninguna máquina virtual.

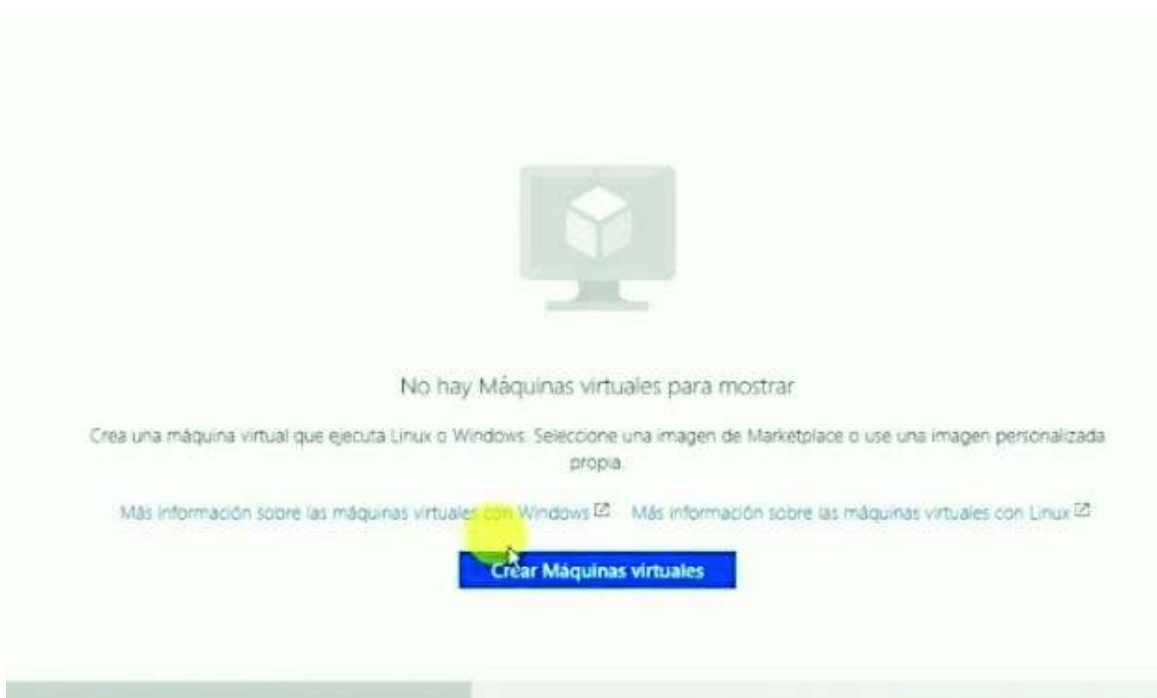


Figura 2.14. Panel de control de *Azure*

En la Figura 2.15 se muestra la máquina virtual de Azure que se definió en el diseño conforme a los requerimientos vistos en la Sección 2.1.2. En el recuadro rojo se enmarca la máquina virtual escogida.

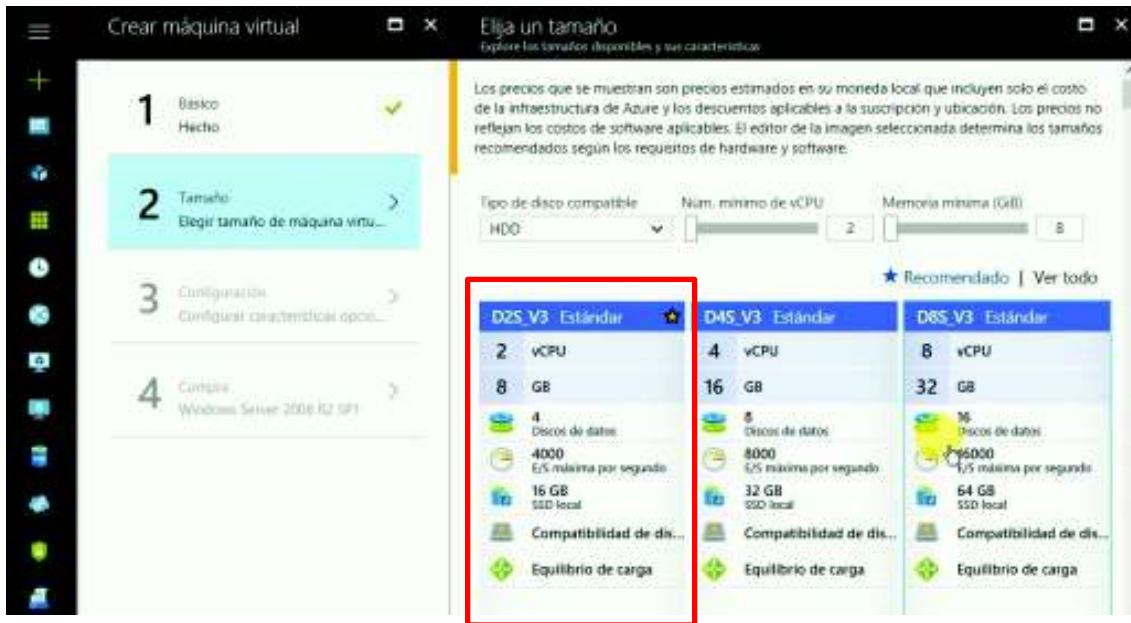


Figura 2.15. Máquina virtual D2S_V3 escogida

En la Figura 2.16 se puede observar la máquina virtual D2S_V3 creada en el mismo portal de Azure. Esta máquina virtual solamente tiene el sistema operativo instalado por lo cual es necesario ingresar de forma remota para poder configurarla.

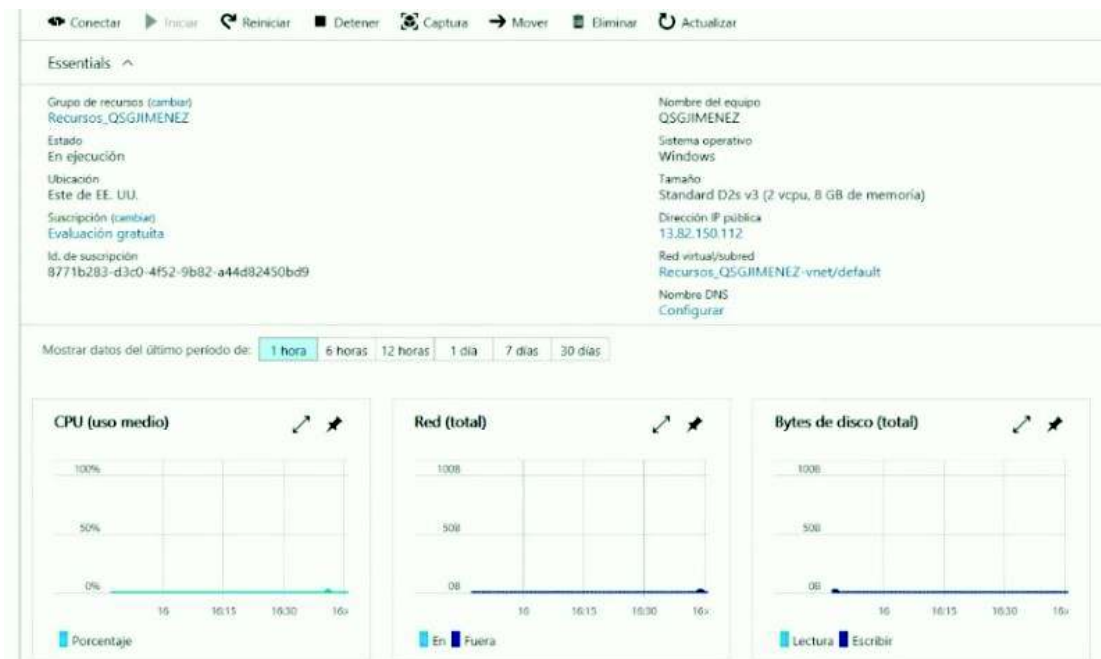


Figura 2.16. Máquina virtual creada

Cuando se accede a la máquina virtual, se pueden crear dos tipos de usuarios: un único usuario administrador y varios usuarios locales. El usuario administrador será el que usará el usuario de la empresa GRUPO NOVATECH CÍA. LTDA como proveedor; y los diferentes usuarios locales serán usados por los clientes de la empresa mencionada. Es necesaria la creación de usuarios locales para la autenticación del acceso al CMI. En la Figura 2.17 se ve la creación del usuario administrador con los permisos de acceso correspondientes.

2.2.2 Capa de Base de Datos

Para la implementación de capa de base de datos, se tiene que realizar la instalación y la configuración del software de base de datos, para el presente sistema se ha seleccionado *Microsoft SQL Server Express 2008 R2*. Para la codificación de la base de datos se mostrará solamente un ejemplo de código, ya que el *script* completo estará en el Anexo A.

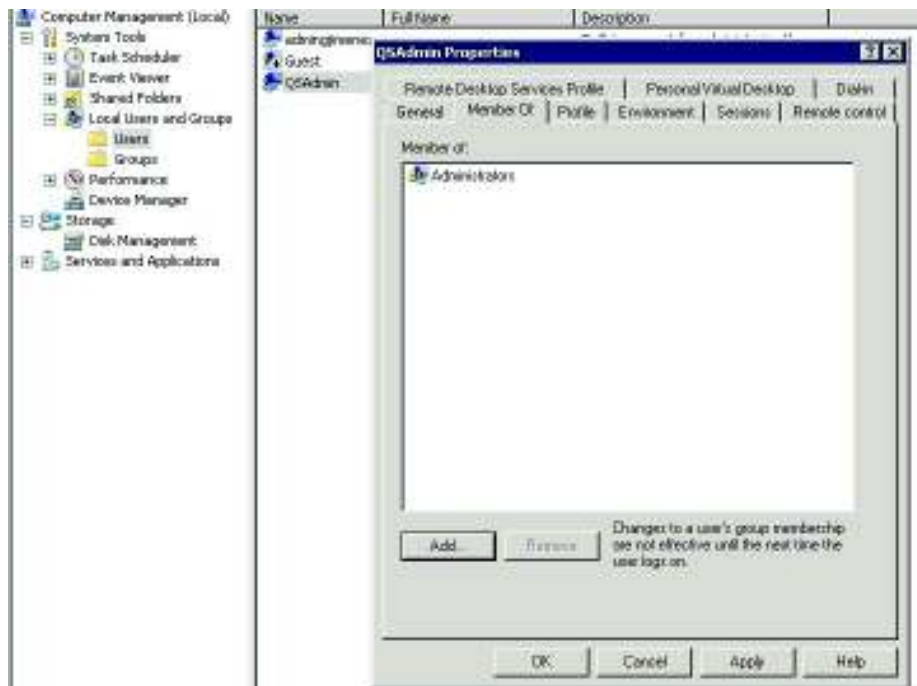


Figura 2.17. Creación del usuario administrador

a. Sprint de Instalación y Configuración de la Capa de Base de Datos

La instalación de la base de datos *Microsoft SQL Server Express 2008 R2* se realiza con el usuario administrador creado anteriormente. Esta instalación es sencilla, solamente hay que asignar una clave para la autenticación SQL, debido a requerimientos de configuración de la interfaz web.

A continuación, se debe instalar *Microsoft SQL Server Management Studio* que es el componente que permitirá administrar la base de datos. En la Figura 2.18 se muestra el acceso a la base de datos con la herramienta instalada.

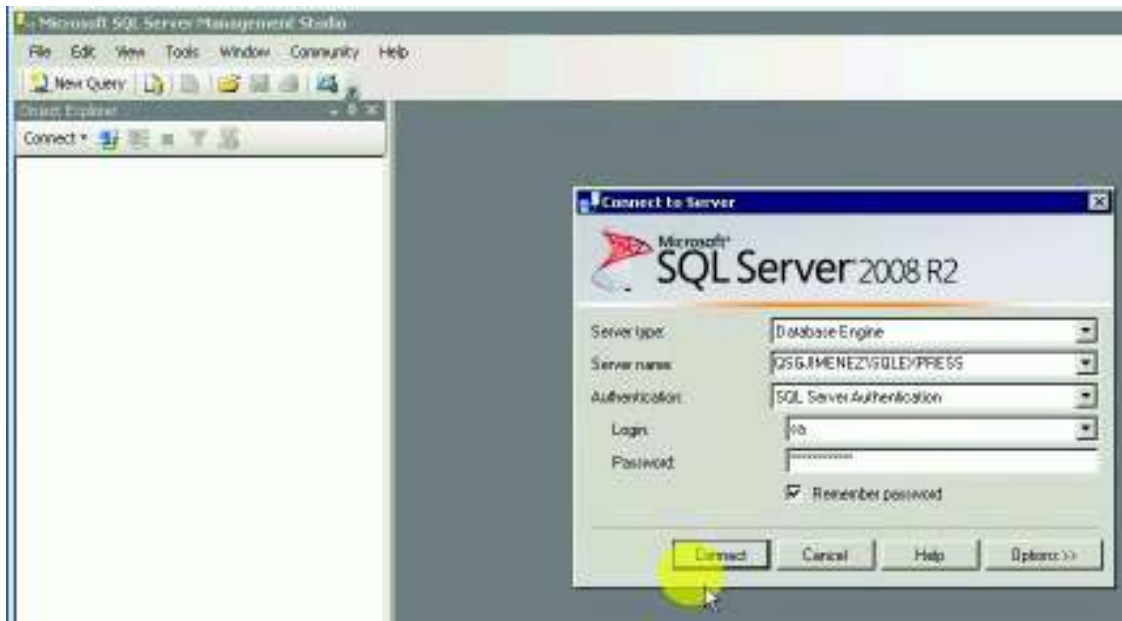


Figura 2.18. Acceso a la base de datos

b. Sprint de Codificación de Base de Datos

Luego de la instalación, se procede a la codificación de la base de datos según el diseño realizado en la Sección 2.1.5. En la Figura 2.19 se muestra un ejemplo del *script* de la programación en *SQL* de una de las tablas que conforman la base de datos. El *script* completo se muestra en el Anexo A. En el ejemplo, se muestra la tabla llamada *ARBOL_INDICADOR*, en donde se guarda la jerarquía de los indicadores asociados a un objetivo específico y cada indicador con su respectiva ponderación.

```
CREATE TABLE [dbo].[ARBOL_INDICADOR] (  
    [IDARBOL] int IDENTITY(1, 1) NOT NULL,  
    [IDOBJETIVO] int NULL,  
    [IDINDICADOR] int NULL,  
    [INDICADOR_PADRE] int NULL,  
    [PONDERACION] decimal(10, 3) NULL  
)  
ON [PRIMARY]  
GO
```

Figura 2.19. Codificación de la tabla *ARBOL_INDICADOR* en lenguaje *SQL*

2.2.3 Capa de Negocio

A continuación, se presenta el detalle de la implementación de la capa de negocio compuesta de dos partes: la configuración del servidor web y la codificación de la interfaz web. Para la codificación de la interfaz web se mostrará solamente un ejemplo de código, ya que el *script* completo está en el Anexo B.

a. Sprint de Instalación y Configuración del Servidor web

Para el presente sistema se ha usado el *Zend Server Express 6.1* que trabaja con un servidor web *Apache 2.2.22*. Un paso importante es la configuración del puerto al cual se accederá a la interfaz Web. Se ha decidido, por facilidad en la configuración del servidor web, que para el acceso a los CMI en *Qlik*, se use el puerto 80 y el 443, por esta razón, el acceso a las páginas publicadas en el servidor web se lo haga por medio del puerto 81. En la Figura 2.20 se puede observar la configuración del puerto de acceso.

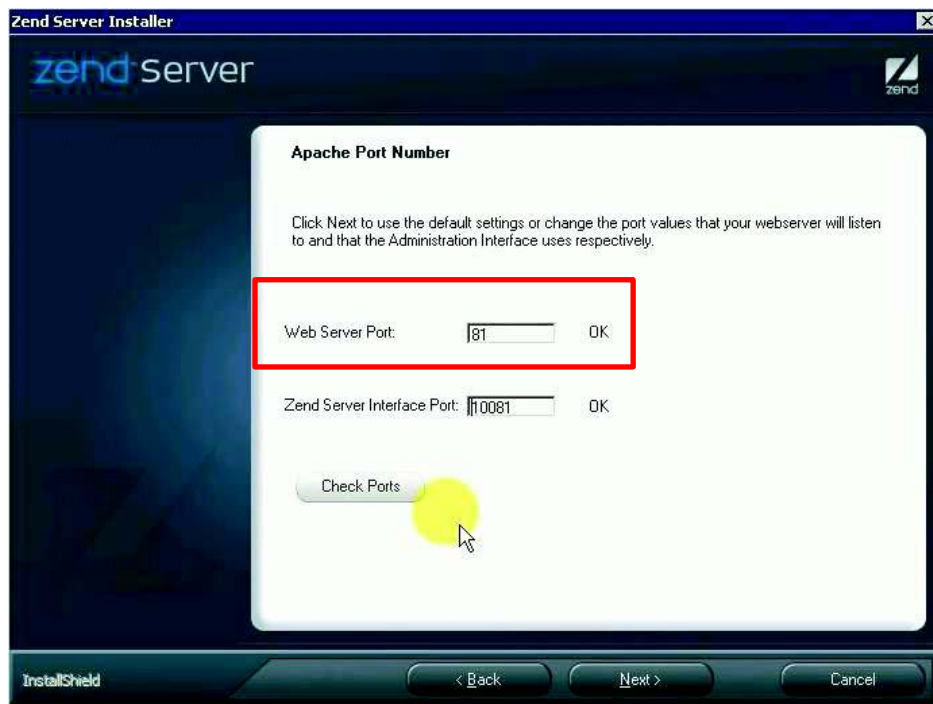


Figura 2.20. Configuración del puerto de acceso 81 para el acceso

Como siguiente paso, es necesario configurar la consola del servidor web instalado. Se debe habilitar la extensión de *PHP* para *SQL* en la configuración del servidor web. En la Figura 2.21 se observa la habilitación de la extensión *SQL* en la consola de configuración del servidor web. Esta opción hay que desactivarla para que no sea una fuente de seguridad en el servidor web.



Figura 2.21. Habilitación de la extensión de *PHP* para *SQL* en la consola del servidor web

Como se asignó al puerto 81 para acceso a la interfaz web, se debe abrir este mismo puerto en las reglas de acceso del portal de *Azure* de la máquina virtual. En la Figura 2.22 se ve la regla de acceso creada para el puerto 81. Los puertos 4244 y 4248 son puertos de uso interno de la plataforma *Qlik*.



Figura 2.22. Reglas de acceso a la máquina virtual en el portal de *Azure*

Y de la misma forma se debe abrir el puerto dentro del *firewall* de la máquina virtual. En la Figura 2.23 se observa la creación de la regla de acceso en el *firewall*.

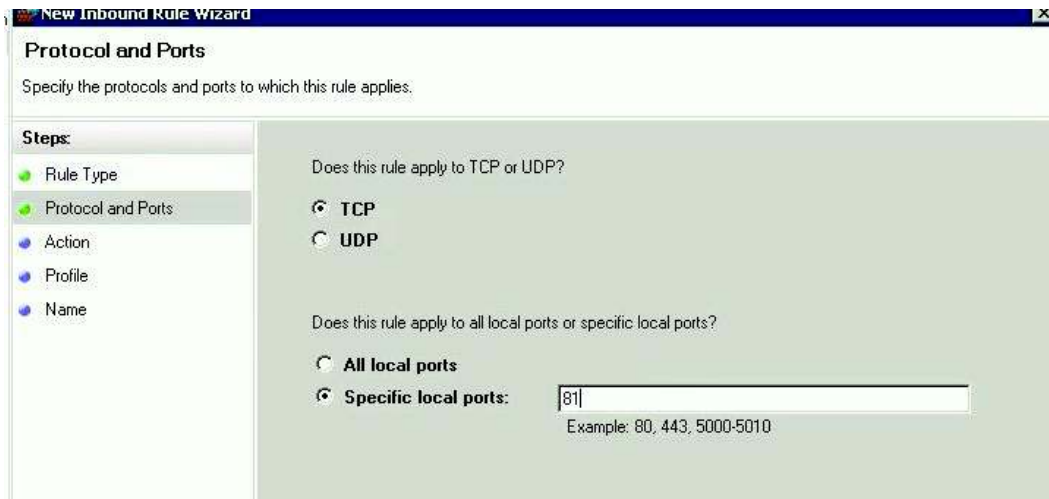


Figura 2.23. Creación de la regla de ingreso para puerto 81 en *firewall* de máquina virtual

b. Sprint de Codificación de la Lógica del Negocio

Luego de la instalación, se procede a la codificación de la lógica del negocio que usará la interfaz web según el diseño realizado en la Sección 2.1.6. En la Figura 2.24 se muestra un ejemplo del *script* de la programación en *PHP* de una de las clases que conforman la interfaz web. El *script* completo se muestra en el Anexo B.

```

class Formula{

    private $ado;

    public function __construct(){

    }

    public function addFormula($data){

    }

    private function existe($nombre){
        $nombre = trim($nombre);
        $sql = "SELECT * FROM FORMULA WHERE NOMBRE = '$nombre'";
        $result = $this->ado->query($sql);

        if(count($result))
            return TRUE;
        else
            return FALSE;
    }

    public function getFormulas(){

    }

    public function getFormula($id){

    }

    public function deleteFormula($id){

    }
}

```

Figura 2.24. Codificación de la clase Formula en lenguaje *PHP*

En el ejemplo, se muestra la clase llamada *Formula*, esta clase define las diferentes fórmulas que puede tener un indicador, está compuesta por un atributo y seis métodos, incluido el constructor. De igual manera, se muestra uno de los métodos, llamado *existe*, en el cual se busca en la base de datos para verificar que esta fórmula aún no exista y pueda ser guardada. Toda la comunicación a la base de datos la hace la clase *Ado* (enmarcado). La clase *Ado* se puede revisar en el diseño de la capa de negocio dentro de la Sección 2.1.6b.

2.2.4 Capa de Presentación

A continuación, se presenta el detalle de la implementación de la capa de presentación. Está compuesta de dos partes: la interfaz web y la plataforma *Qlik*. Ambas partes son consideradas como la capa de presentación porque ambas funcionan como una interfaz directa con el cliente. La codificación realizada para la interfaz web, servirá para el ingreso de la información de la PE de cada cliente hacia la capa de presentación; mientras que la codificación realizada para la plataforma *Qlik*, servirá para la visualización de la información ingresada por medio del CMI correspondiente.

Para la codificación del CMI en *Qlik* y también para la interfaz web, se mostrará solamente un ejemplo de código, ya que el *script* completo está en el Anexo C. Para la plataforma *Qlik* se tendrá la instalación del módulo de administración, la configuración del mismo y la codificación del CMI.

a. Sprint de Codificación de la Interfaz web

A continuación, se presenta la codificación que se realizó para la interfaz web que usará el cliente para ingresar los datos a la base de datos. Para ello se usó una combinación de lenguaje PHP con HTML. En la Figura 2.25 se muestra el extracto de la codificación del menú de Fórmulas en la interfaz web, aquí se puede ver cómo la interfaz usa las funciones creadas en la Sección 2.2.3b.

b. Sprint de Instalación y Configuración del Módulo de Administración Plataforma Qlik

Una vez que se ha ingresado en la máquina virtual con el usuario administrador, se procede a la instalación de la plataforma *Qlik*.

```

<?php
/*
 * Vista del modulo de formulas
 */
function start():
include($_SESSION['path'].'./modules/parametros/formula/launch.php');

$formula = new Formula();
if(isset($_GET['add'])){
}
if(isset($_GET['edit'])){

}

$lista = $formula->getFormulas();
?>
<h2>Fórmulas</h2>

<div id="listaFrecuencia">
<h3>Listado de fórmulas</h3>
<table>
<tr class="tableTitle">
<?php
$i=0;
foreach($lista as $item){
    if($i%2)
        $clase = "trImpar";
    else
        $clase = "trPar";
    $i++;

    echo "<tr class='".$clase."'>";
    echo "<td><p>{$item['NOMBRE']}</p></td>";
    echo "<td><a href='\"#$\"' onClick='\"if(confirm(\"¿Desea eliminar esta fórmula?\"))\"'\">";
    echo "<img src='\"".$_SESSION['url']."/images/eliminar.gif\"></a></td>";
    echo "</tr>";
}
?>
</table>
</div>

```

Figura 2.25. Extracto de la codificación del menú de Fórmulas en la interfaz web

Para realizar la configuración del módulo de administración, es necesario ingresar con el usuario administrador (usuario local de la máquina virtual) creado en la Sección 2.2.1. La primera vez se debe ingresar el usuario y la contraseña, en los siguientes accesos, ya podría guardarse la contraseña. Posteriormente, se deben configurar los permisos de acceso a los usuarios. En primer lugar, se dará acceso al usuario administrador, que es el usuario con el que se realizará la codificación del CMI. Esta sección de configuración del CMI será llevada a cabo exclusivamente por el administrador.

El siguiente paso es habilitar los puertos de acceso de los navegadores web [54] dentro de la configuración del módulo de administración. En la Figura 2.26 se visualiza la configuración de los puertos de acceso para los diferentes navegadores web que se

vayan a usar para acceder a la plataforma *Qlik*. Como se indicó en la Sección 2.2.3a los puertos a usarse para el acceso al CMI, son los puertos 80 y 443 (enmarcados).



Figura 2.26. Configuración de puertos de acceso para el CMI

A continuación, se debe realizar la configuración de las direcciones IP permitidas para acceder al CMI. En la Figura 2.27 se ve la asignación de las IP correspondientes. Las direcciones IP que se colocan en esta configuración, son las direcciones a las cuales deberán apuntar los clientes para poder entrar al CMI ya que, al ser una solución SaaS, el acceso de los clientes es siempre vía web.

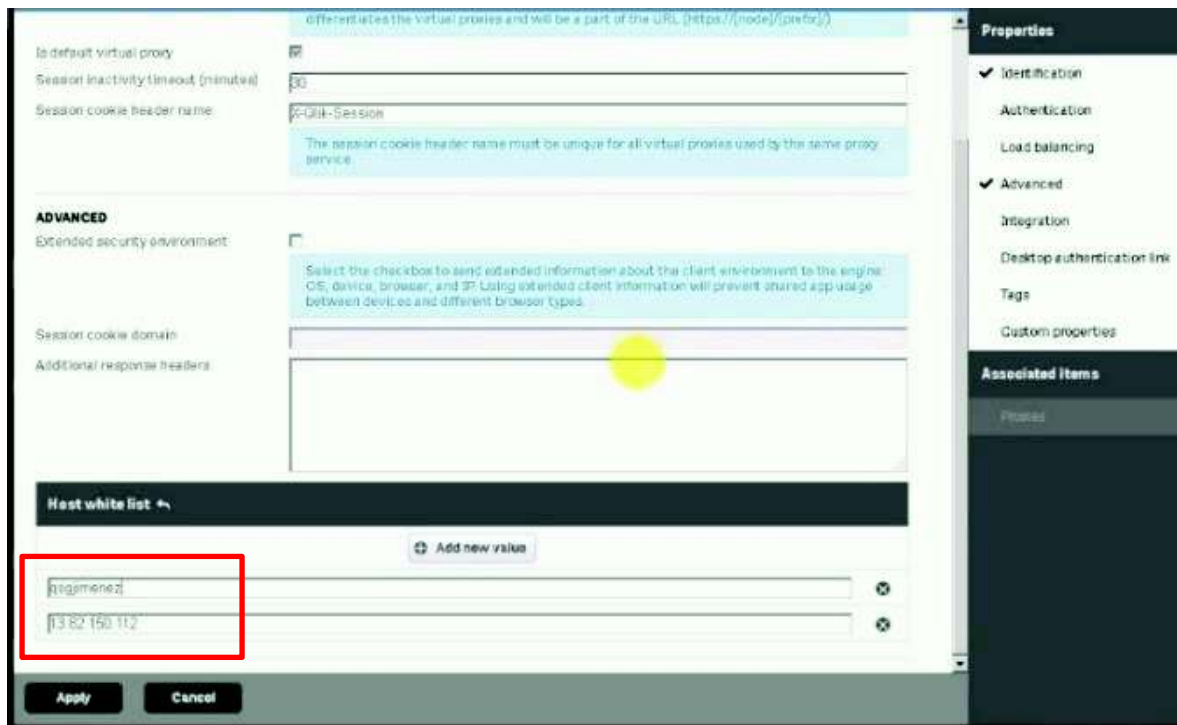


Figura 2.27. Asignación de direcciones IP permitidas

A continuación, lo que se requiere, es habilitar los puertos en el *firewall* de la máquina virtual para que puedan acceder remotamente los usuarios. En la Figura 2.28 se puede observar la regla de acceso del *firewall*.

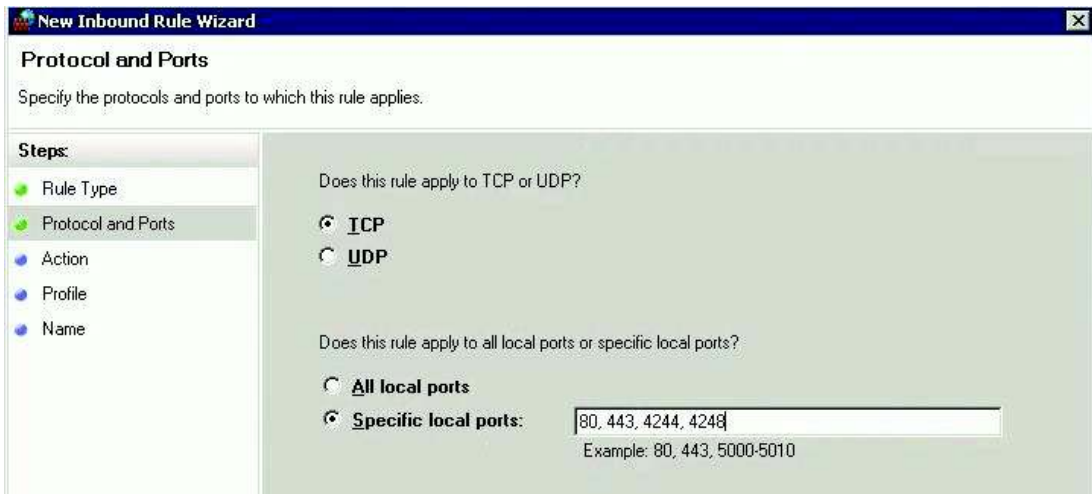


Figura 2.28. Creación de regla de acceso del *firewall* de máquina virtual

Finalmente, hay que crear la regla de acceso con la habilitación de los puertos en la máquina virtual dentro del panel de control de *Azure*. En la Figura 2.29 se puede observar la habilitación de los puertos en el panel de control de *Azure*.

PRIORIDAD	NOMBRE	PUERTO	PROTOCOLO	ORIGEN	DESTINO
100	HTTP	80	TCP	Cualquiera	Cualquiera
101	HTTPS	443	TCP	Cualquiera	Cualquiera
102	Port_4244	4244	Cualquiera	Cualquiera	Cualquiera
103	Port_4248	4248	Cualquiera	Cualquiera	Cualquiera
1000	default-allow-rdp	3389	TCP	Cualquiera	Cualquiera

Figura 2.29. Habilitación de puertos de acceso al CMI en el panel de control de *Azure*

c. Sprint de Codificación del CMI Plataforma Qlik

A continuación, se procede a la codificación del CMI según el diseño realizado en la Sección 2.1.7. Para realizar la codificación del CMI primero es necesario crear una conexión abierta de *Microsoft*, también conocida como ODBC (*Microsoft Open Database Connectivity*) [55]. Esta conexión debe apuntar a la base de datos. En la Figura 2.30 se muestra la prueba exitosa de conexión a la base de datos por medio de un ODBC.

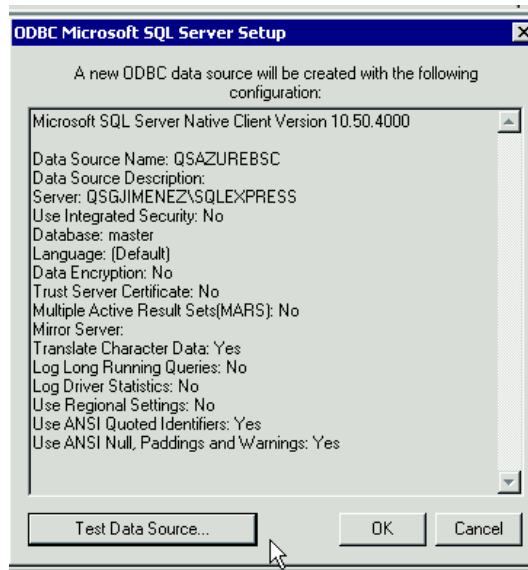


Figura 2.30. Conexión exitosa a la base de datos por ODBC

Con la ODBC creada, es necesario crear una librería en *Qlik* para extraer los datos. En la Figura 2.31 se puede observar la creación de la librería *QSAZUREBSC* en *Qlik* la cual será usada para comunicarse con la capa de base de datos. El proceso de extracción de datos es la lectura de las tablas que se necesitan para la creación del CMI desde la base de datos hacia la plataforma *Qlik*. Dentro de la consola de comandos de *Qlik* se colocan las sentencias SQL para leer los datos y, en conjunto con las funciones propias de *Qlik*, se irán guardando los datos en los archivos propios del CMI llamados archivos QVD.

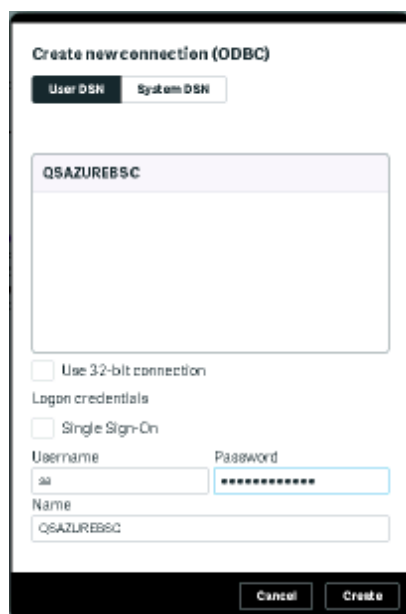


Figura 2.31. Creación de librería de conexión en *Qlik*

En la Figura 2.32 se muestra un ejemplo del *script* de la programación en el lenguaje propio de la plataforma *Qlik* de la nube de datos. El *script* completo se muestra en el Anexo C. En el ejemplo, se muestra la cadena de conexión llamada *QSAZUREBSC* la cual permite tomar los datos. Se extraen la información de la tabla llamada *dm_Indicadores* y al final esta información se guarda en un archivo QVD con el mismo nombre.

```
LIB CONNECT TO 'QSAZUREBSC';

dm_Indicadores:
LOAD IDINDICADOR as Id_Indicador,
    CODIGO asCodigo_Indicador,
    "NIVEL ESTRATEGICO" as Nivel_Estrategico_Indicador,
    mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Indicador,
    UNIDAD as Id Unidad_Indicador,
    mapsubstring('temp_Caracteres',DEFINICION) as Definicion_Indicador,
    FRECUENCIA as Id_Frecuencia,
    MANUAL as Manual,
    FORMULA as Id_Formula,
    SIGNO as Signo_Indicador,
    [FACILITADOR] as Id Usuario_Facilitador_Indicador,
    [RESPONSABLE GERENCIA] as Id Usuario_Gestor_Indicador,
    [DESPLIEGUE] as Id_Despliegue,
    [SEGMENTO] as Id_Segmento
;

SQL SELECT *
FROM UTE.dbo.INDICADORES;

STORE dm_Indicadores into [lib://qvdBase/dm_Indicadores.qvd];
DROP Table dm_Indicadores;
```

Figura 2.32. Codificación de la tabla *dm_Indicadores* en *Qlik*

2.2.5 Sprint de Redacción del SLA

Como parte final del presente capítulo, se va a detallar el SLA que será entregado por parte de la empresa GRUPO NOVATECH CÍA. LTDA. a sus respectivos clientes, al momento de realizar la comercialización de su software de PE como un servicio SaaS. La redacción completa del SLA [56] se encuentra en el Anexo D.

En la Sección 1.7 se definieron los parámetros que debería constar en el SLA con el objetivo de que tanto la parte del proveedor como la parte de los clientes tengan claros los acuerdos de servicio, las circunstancias sobre las que se va a llevar a cabo el servicio y las responsabilidades de ambas partes.

3 RESULTADOS Y DISCUSIÓN

El presente capítulo muestra las pruebas realizadas sobre el sistema implementado. Se comenzará con las pruebas de funcionamiento del laaS contratado, le seguirán las pruebas de funcionamiento de la capa de base de datos, a continuación, se realizarán las pruebas de funcionamiento de la capa de negocios, y finalmente, se realizarán las pruebas de funcionamiento de la capa de presentación. Como un paso final de la metodología SCRUM, dentro de las pruebas anteriormente mencionadas, constarán también las pruebas de aceptación de las historias de usuario y de los todos los *sprints* realizados en el diseño.

3.1 Pruebas de Funcionamiento del laaS Contratado

Para las pruebas de funcionamiento del laaS contratado se deberá acceder desde la perspectiva del cliente como la del proveedor. El primer conjunto de pruebas será desde diferentes navegadores web, tanto al servidor web como a la plataforma *Qlik*. Y el segundo conjunto de pruebas, el acceso como proveedor, que se lo realizará con el acceso remoto por medio de la propia herramienta de *Microsoft*.

3.1.1 Pruebas de Acceso desde la Perspectiva Cliente

El acceso a la máquina virtual de parte de los clientes solo será posible cuando el proveedor haya dado los permisos correspondientes, estos permisos de accesibilidad han sido configurados en las Secciones 2.2.3 y 2.2.4.

En la Figura 3.1 se puede observar el acceso desde un navegador web externo hacia la máquina virtual por medio del puerto 81, es decir, a la página de inicio del servidor web.

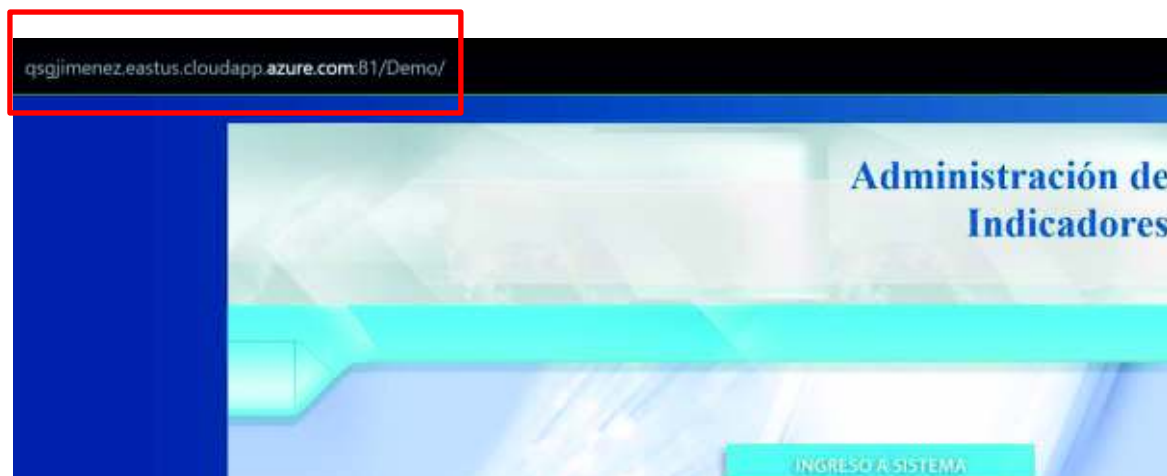


Figura 3.1. Acceso al servidor web desde un navegador web externo

Como se configuró en la Sección 2.2.3a. Con estas pruebas se está dando por aprobada la historia de usuario 01-C diseñada en la Sección 2.1.2a.

En la Figura 3.2 se puede observar el acceso a la máquina virtual por medio de un navegador web externo usando el puerto 80 (por defecto), es decir, se accede a la plataforma *Qlik* que ha sido instalada y configurada en la Sección 2.2.4b.

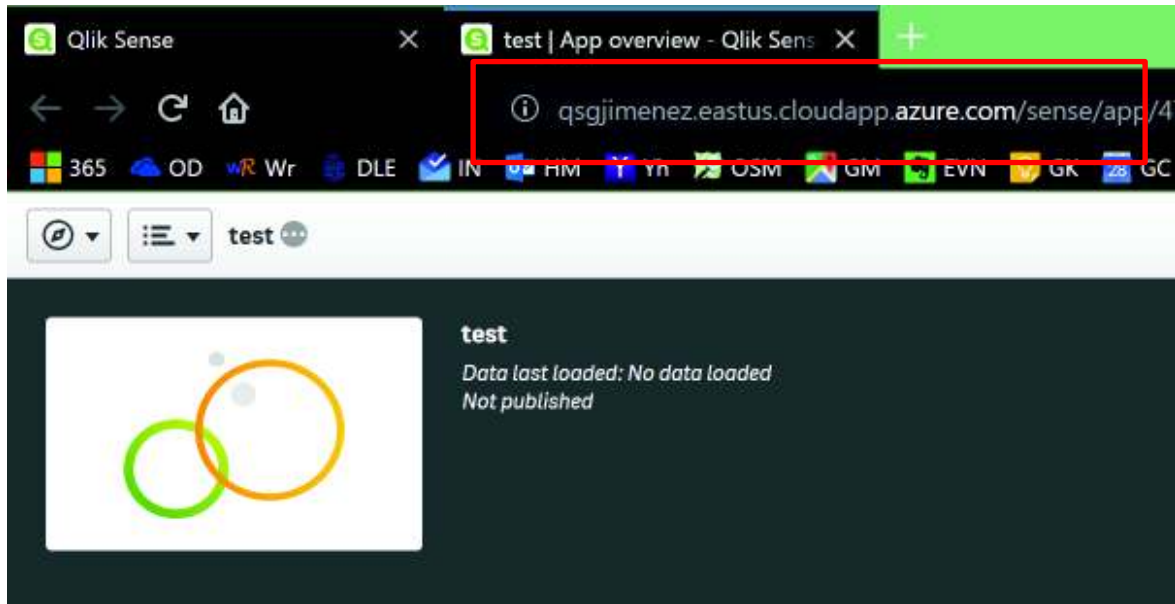


Figura 3.2. Acceso a la plataforma *Qlik* desde un navegador web externo

Como se pudo observar, el cliente solamente tendrá acceso a la interfaz web y al CMI, a los cuales deberá ingresar con unas credenciales asignadas.

3.1.2 Pruebas de Acceso desde la Perspectiva Proveedor

El acceso a la máquina virtual de parte del proveedor se lo realiza por medio de la propia herramienta de acceso remoto de *Microsoft*. La dirección IP y el puerto al cual se debe apuntar se lo obtiene desde el panel de control de *Azure*, como se observó en la Sección 2.2.1.

En la Figura 3.3 se puede observar el acceso del proveedor a la máquina virtual por medio de la herramienta de acceso remoto que viene por defecto en cualquier sistema operativo de *Microsoft*. Solamente el proveedor podrá acceder a la máquina virtual completa y como se puede observar, el usuario y la contraseña a usarse en la IP del ejemplo (enmarcada) fueron configuradas en la Sección 2.2.2. Con estas pruebas se está dando por aprobadas las historias de usuario 01-P y 02-P diseñadas en la Sección 2.1.2b.

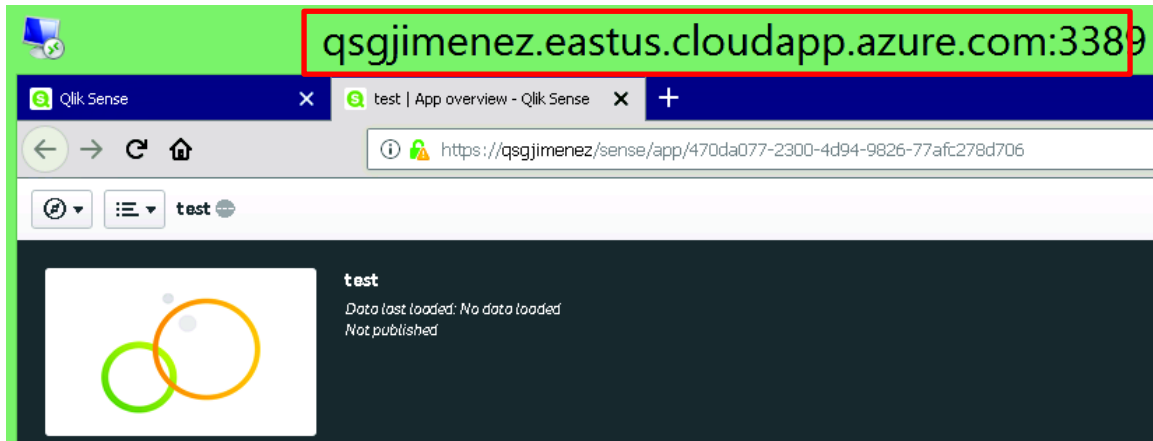


Figura 3.3. Acceso a la máquina virtual por medio de la herramienta de acceso remoto de Microsoft

3.2 Pruebas de Funcionamiento de la Capa de Base de Datos

Para las pruebas de funcionamiento de la capa de base de datos se deberá comprobar que la base de datos puede leer, guardar y editar la información que se encuentra en sus tablas en base en el diagrama entidad – relación diseñado en la Sección 2.1.5a.

En la Figura 3.4 se ha realizado una consulta de lectura de datos de la tabla *Perfil*. El resultado muestra los tipos de usuario definidos en la historia de usuario 02-C, Sección 2.1.2a.

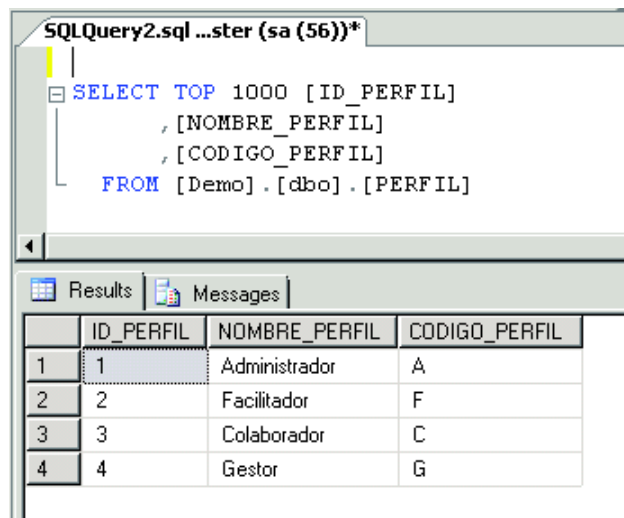


Figura 3.4. Consulta a la tabla *Usuario*

Además, en la Figura 3.5 se puede observar a la base de datos creada en su totalidad en donde constan las mismas tablas que fueron diseñadas en el modelo entidad - relación.

La base de datos creada se la ha llamado *Demo*, ya que como se especificó en la historia de usuario 03-P, Sección 2.1.2b, se realizará un proyecto demostrativo con datos ficticios.

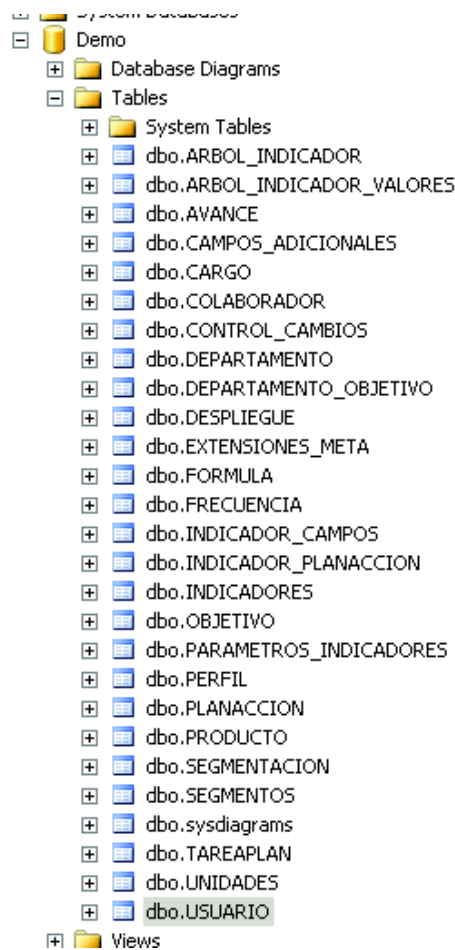


Figura 3.5. Base de datos creada

3.3 Pruebas de Funcionamiento de la Capa de Negocio

Las pruebas de la capa de negocio se las va a realizar en función de los perfiles creados, es decir, pruebas de funcionamiento del perfil Administrador, del perfil Facilitador, del perfil Colaborador y del perfil Gestor en función de los casos de uso diseñados en el Sección 2.1.6a.

3.3.1 Pruebas de Funcionamiento del Administrador

El primer conjunto de pruebas de funcionamiento de la capa de negocio es respecto al perfil Administrador, estas pruebas se basan en la historia de usuario 02.1-C del Sección 2.1.2a. En la Figura 3.6 se puede observar que el Administrador puede crear nuevos usuarios con la posibilidad de cualquiera de los perfiles. Con estas pruebas de

funcionamiento también se da por aprobada la historia de usuario 04-P diseñada en el Sección 2.1.2b.

Añadir / Editar usuarios del sistema

Nombre:	<input type="text" value="Gonzalo"/>
Apellido:	<input type="text" value="Jiménez"/>
E-mail:	<input type="text" value="gjimenez@grupo-novatech.com"/>
Estado:	<input checked="" type="radio"/> Activo - <input type="radio"/> Inactivo
Tipo:	<input checked="" type="checkbox"/> Administrador
	<input checked="" type="checkbox"/> Facilitador
	<input checked="" type="checkbox"/> Colaborador
	<input checked="" type="checkbox"/> Gestor
<input type="button" value="Aplicar"/>	

Figura 3.6. Creación de nuevo usuario

En la Figura 3.7 se puede ver los diferentes parámetros iniciales que el Administrador puede crear, se ha escogido la parametrización de indicadores, en donde se ven el día de fecha de inicio de ingreso de valores ingresados, el número de días para que estos valores puedan ser ingresados por el facilitador y también, el número de días extra que se pueden extender en caso de una ampliación del plazo.

Parametrización de indicadores

Día del mes que inicia el ingreso de valores ejecutados:	<input type="text" value="20"/> de cada mes.
Número de día para el ingreso de valores ejecutados:	<input type="text" value="10"/> días.
Número máximo de día extra para el ingreso de valores ejecutados:	<input type="text" value="4"/> días.
<input type="button" value="Aplicar"/>	

Figura 3.7. Ingreso de parámetros iniciales de indicadores

En la Figura 3.8 se puede observar la posibilidad que tiene el Administrador para crear un objetivo. Dependiendo de la estructura de PE de cada cliente, se podría anidar uno o más objetivos para crear un árbol de objetivos (enmarcada).

- ▶ Objetivos
- ▶ Arbol de Objetivos
- ▶ Crear Objetivo
- ▶ Vincular Objetivo

Agregar / Modificar Objetivos

Código:

Nombre:

Objetivo Padre:

Signo:

Operación:

Figura 3.8. Creación de un objetivo

En la Figura 3.9 se puede ver un nuevo indicador añadido a un objetivo. El indicador tiene el nombre de *Número de Magias*, y ha sido añadido al objetivo *Incrementar la producción de investigación científica*. Como ya existían dos indicadores anteriores atados a este mismo objetivo, la ponderación (en paréntesis) es de 33. Entre los tres indicadores suman el 100 % de la ponderación.

[4] INCREMENTAR LA PRODUCCIÓN DE INVESTIGACIÓN CIENTÍFICA +

- [0] Número de Magias (33) +
- [1] Mecanismos comunitarios de protección (33) +
- [4.7] Ejecutar plan anual de investigación (34) +

Figura 3.9. Nuevo indicador añadido

En la Figura 3.10 se puede observar el botón amarillo que permite extender el plazo de tiempo para ingresar los valores de algún indicador. Este plazo de tiempo debería ser definido en los parámetros iniciales de indicador, como se vio en la Figura 3.7.

DEFINICIÓN	FORMULA	DESPLIEGUE	FRECUENCIA	VIGENTE	ESTADO	FACILITADOR		
Magias con los Checklist completos	Suma	Prov	Trimestre	Vigente	Activo	Facilitador DEMO		
Número de mecanismos	Suma	País	Trimestre	Vigente	Activo	Facilitado	Extender plazo de ingreso de valores meta.	

Figura 3.10. Extensión de plazo para un indicador

En la Figura 3.11 se puede ver la notificación del sistema en el momento de haber creado un nuevo usuario para el acceso al software de PE.



Figura 3.11. Notificación de sistema

3.3.2 Pruebas de Funcionamiento del Facilitador

El segundo conjunto de pruebas de funcionamiento de la capa de negocio es respecto al perfil Facilitador, estas pruebas se basan en la historia de usuario 02.2-C de la Sección 2.1.2a. En la Figura 3.12 se puede observar que el Facilitador puede ingresar los valores mensuales a los indicadores a los cuales haya sido asociado.



Figura 3.12. Ingreso de valores mensuales

En la Figura 3.13 se observa la creación de un nuevo plan de acción del indicador llamado *Porcentaje de docentes que participan en los proyectos de investigación*.

Crear / Editar Plan de Acción

Indicador:	Número de Magias
* Definición	<input type="text" value="Recopilación de información"/>
* Justificación	<input type="text" value="Evaluación interna"/>
Acción a Tomar	<input type="text" value="Centralizar la documentación ingresada al departamento"/>
Valor Meta	<input type="text" value="2000"/>
Presupuesto	<input type="text" value="1000"/>
Responsable	<input type="text" value="DEMO Facilitador"/>
Fecha Objetivo 1	<input type="text" value="2018-04-30"/> <input type="text" value="Cargo"/>
Documento	Click aquí para cargar un archivo.
Otros Recursos	<input type="text" value="NA"/>
Fecha de Creación	<input type="text" value="2018-04-22"/>
Observaciones	<input type="text"/>

Figura 3.13. Formulario para crear de un nuevo plan de acción

En la Figura 3.14 se puede observar la creación de una nueva tarea por parte del Facilitador. Esta nueva tarea debe estar asociada a un plan de acción específico.

Agregar / Editar tarea para plan de acción

Nombre de la tarea	<input type="text" value="Optimizar proceso de gestión documental"/>
Alcance de la tarea	<input type="text" value="Reducir tiempo de registro y recopilación de documentos ingresados al departamento."/>
Responsable	<input type="text" value="DEMO Facilitador"/>
Producto u entregable	<input type="text" value="Plan de optimización"/>
Documento	Click aquí para cargar un archivo.
Plazo para inicio	<input type="text" value="2018-04-22"/>
Plazo para fin	<input type="text" value="2019-04-30"/>
Estado	<input type="text" value="No iniciado"/>
Presupuesto	<input type="text" value="1000"/>
Justificación	<input type="text" value="Evaluación interna"/>
Avance	<input type="text" value="A tiempo"/>
Porcentaje de avance	<input type="text" value="0"/> %
Fecha Objetivo 1	<input type="text" value="2018-04-22"/> <input type="text" value="Cargo"/>
Fecha Objetivo 2	<input type="text" value="2018-04-30"/> <input type="text" value="Cargo"/>
Fecha Objetivo 3	<input type="text" value="2018-05-31"/> <input type="text" value="Cargo"/>
	<input type="button" value="Crear"/>

Figura 3.14. Formulario para crear de una nueva tarea

Siguiendo el esquema de elementos habilitados para el Facilitador, en la Figura 3.15 se puede observar la pantalla de creación de un nuevo avance de una tarea pre existente.

Registrar Nuevo Avance

* Avance 15 %

* Descripción

Entrega de documentación

Documento [Click aquí para cargar un archivo.](#)

Fecha de Creación

Figura 3.15. Formulario para crear de un nuevo avance de una tarea

3.3.3 Pruebas de Funcionamiento del Colaborador

El tercer conjunto de pruebas de funcionamiento de la capa de negocio es respecto al perfil Colaborador, estas pruebas se basan en la historia de usuario 02.3-C de la Sección 2.1.2a. En la Figura 3.16 se puede observar la forma de finalización de la tarea de uno de los planes de acción del cual sea responsable. En la opción de Reportes (enmarcado) se podrá listar los diferentes avances de esta tarea que hayan sido registrados.

ESTADO	AVANCE	REPORTES	CUMPLIDO
No iniciado	Adelantada		✘ SIN FINALIZAR Cambiar a FINALIZADA

Figura 3.16. Finalización de tarea

3.3.4 Pruebas de Funcionamiento del Gestor

El cuarto conjunto de pruebas de funcionamiento de la capa de negocio es respecto al perfil Colaborador, estas pruebas se basan en la historia de usuario 02.4-C de la Sección 2.1.2a. En la Figura 3.17 se puede observar la forma de aprobar un valor que haya sido ingresado por el Facilitador para un indicador. En el ejemplo se aprueba el valor de septiembre del indicador *Porcentaje de docentes que participan en los proyectos de investigación* al habilitar el recuadro de aprobación y guardando los cambios.

AÑO	PERIODO	DESPLIEGUE	VALOR META	VALOR EJECUTADO	<input type="checkbox"/>
2017	September	Prov	100.000	10.000	<input type="checkbox"/>

Figura 3.17. Aprobación de valores reportados

3.4 Pruebas de Funcionamiento de la Capa de Presentación

Las pruebas de funcionamiento de la capa de presentación son las de la plataforma *Qlik*. Según la Sección 2.1.7, se realizaron tres tipos de etapas para el desarrollo del CMI en la plataforma *Qlik*: la de extracción, la de transformación y la de visualización. Con estas pruebas, además, se está dando por aprobada la historia de usuario 05-P definida en la Sección 2.1.2b.

En la Figura 3.18 se puede observar las tres etapas mencionadas. Para la extracción se ha realizado el *crg_Demo*, para la transformación se ha desarrollado el *prg_Demo* y para la visualización el *CMI Demo*; esta nomenclatura es una sintaxis definida por GRUPO NOVATECH CÍA. LTDA. para todos sus desarrollos en *Qlik*. El cliente podrá tener acceso solamente al *CMI Demo*.

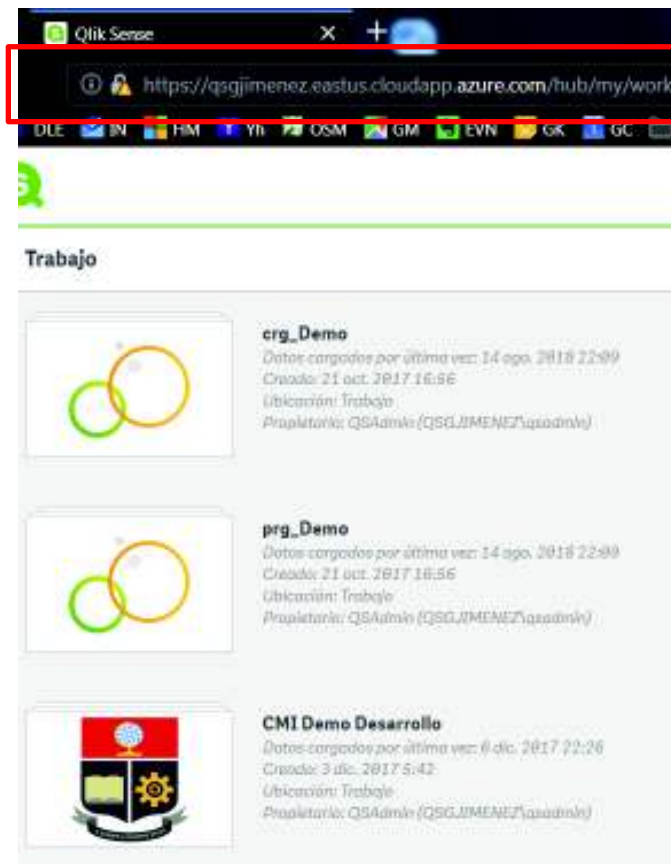


Figura 3.18. Etapas de desarrollo del CMI en *Qlik*

En la Figura 3.19 se observa una tabla en la que se está calculando el cumplimiento de los indicadores y de los objetivos que forman parte de PE en el *CMI Demo*. Estos valores están agrupados por periodos y se respeta la jerarquía creada tanto en el árbol de objetivos como en el árbol de indicadores.

Objetivo Padre	Objetivo Hijo	Objetivo Nieto	Indicador	Periodo			Medidas						
				1	2	3	4	5	6				
				% Cump.	% Relativo	Peso	% Cump.	% Relativo	Peso	% Cump.	% Relativo	Peso	% Relativo
	5.1.2			-	-	15,00	0,00%	-	15,00	0,00%	-	15,00	-
	5.1.3			-	-	25,00	0,00%	-	25,00	0,00%	-	25,00	-
	5.1.4			-	-	10,00	0,00%	-	10,00	0,00%	-	10,00	-
				7,79%	23,38%	300,00	11,44%	34,33%	200,00	19,05%	38,10%	200,00	46,26%
+	Incrementar la producción de investigación científica												
	4.71			11,18%	11,18%	100,00	9,02%	9,02%					
				90,00%	4,50%	5,00	76,67%	3,83%	5,00				
	4.73			0,00%	-	5,00	0,00%	-	5,00				
	4.74			0,00%	-	20,00	0,00%	-	20,00				
	4.75			0,00%	-	10,00	0,00%	-	10,00				
	4.76			0,00%	-	60,00	0,00%	-	60,00				
				6,52%	6,52%	100,00	13,04%	13,04%	100,00	16,30%	16,30%	100,00	19,57%
	4.9			6,52%	6,52%	100,00	13,04%	13,04%	100,00	16,30%	16,30%	100,00	19,57%
				0,00%	0,00%	100,00	20,00%	20,00%	100,00	23,64%	23,64%	100,00	29,09%
	4.10			0,00%	-	100,00	20,00%	20,00%	100,00	23,64%	23,64%	100,00	29,09%
+	Publicar artículos en revistas regionales												
				6,52%	6,52%	100,00	13,04%	13,04%	100,00	16,30%	16,30%	100,00	19,57%
	4.9			6,52%	6,52%	100,00	13,04%	13,04%	100,00	16,30%	16,30%	100,00	19,57%
				0,00%	0,00%	100,00	20,00%	20,00%	100,00	23,64%	23,64%	100,00	29,09%
+	Publicar libros revisados por pares												
	4.10			0,00%	-	100,00	20,00%	20,00%	100,00	23,64%	23,64%	100,00	29,09%

Figura 3.19. Tabla de cálculo del cumplimiento de la PE

3.5 Pruebas de Funcionamiento del Flujo de Datos

Para las pruebas de funcionamiento del flujo de datos solamente se mostrará la forma de actualización de información por medio de las tareas de recarga en la plataforma *Qlik*, debido a que el resto de elementos que conforman el *sprint* diseñado en la Sección 2.1.4 ya se los ha visto en detalle a partir de la Sección 3.3 en adelante.

En la Figura 3.20 se puede observar las tareas de recarga de información de las tres etapas de desarrollo dentro de la plataforma *Qlik*. Todas las tareas de actualización se ejecutarán una vez al día, todos los días, para asegurar que el CMI pueda tener datos recientes.

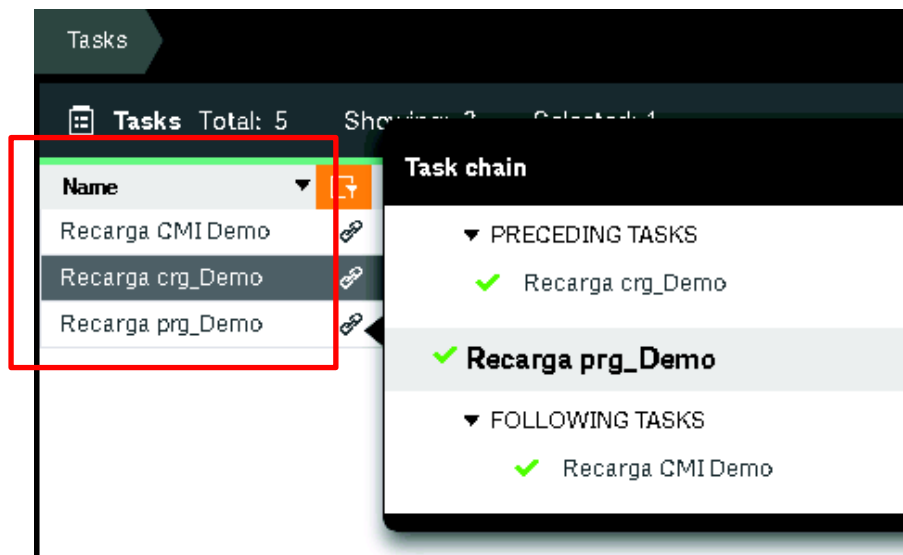


Figura 3.20. Tareas de recarga de datos en la plataforma *Qlik*

El orden en el que han sido configuradas (enmarcado) es el siguiente: primero se actualizan los datos de *crg_Demo*, si esta tarea finaliza con éxito, se actualiza automáticamente el *prg_Demo*. Y de igual manera a la anterior tarea, si el *prg_Demo* se actualiza con éxito, se procede a actualizar en *CMI Demo*. De esta manera, se asegura que, si hay alguna interrupción en el proceso de actualización de datos, al menos se tenga información en el CMI de un día atrás.

3.6 Resumen

Una vez finalizadas las pruebas, se va a realizar un resumen del cumplimiento de las historias de usuario para comprobar que todas hayan sido ejecutadas. En la Tabla 3.1 se puede observar la historia de usuario y la correspondiente sección en el que se cumple.

Tabla 3.1. Tabla de resumen de cumplimiento de requerimientos

Historia de Usuario	Código	Sección
Acceso Ubicuo	01-C	3.1.1
Perfiles de Usuario	02-C	3.2
Perfil Administrador	02.1-C	3.3.1
Perfil Facilitador	02.2-C	3.3.2
Perfil Colaborador	02.3-C	3.3.3
Perfil Gestor	02.4-C	3.3.4
Realizar la instalación del software en la infraestructura <i>cloud</i> .	01-P	3.1.2
Realizar la administración del software desde un solo servidor.	02-P	3.1.2
Proyecto demostrativo.	03-P	3.2
Administración por parte del cliente.	04-P	3.3.1
Actualización de la capa de presentación por parte del proveedor.	05-P	3.4

Tabla 3.2. Tabla de resumen de los *sprints*

<i>Sprint</i>	Sección
Requerimientos	2.1.2
Selección de Proveedor de IaaS	2.1.3
Flujo de Información	2.1.4
Diseño de la capa de base de datos	2.1.5
Diseño de la capa de negocio	2.1.6
Diseño de la capa de presentación	2.1.7
Instalación y configuración de plataforma IaaS	2.2.1
Instalación y configuración de la capa de base de datos	2.2.2-a
Codificación de base de datos	2.2.2-b
Instalación y configuración del servidor web	2.2.3-a
Codificación de la lógica del negocio	2.2.3-b
Codificación de la interfaz web	2.2.4-a
Instalación y configuración del módulo de administración plataforma <i>Qlik</i>	2.2.4-b
Codificación del CMI plataforma <i>Qlik</i>	2.2.4-c
Redacción de SLA	2.2.5

En la Tabla 3.2 se puede observar la lista de los *sprints*, la sección en los que se encuentran y las horas reales que se usaron.

La historia de usuario 01-C ha sido validada debido a que durante la configuración y uso del presenta trabajo se ha accedido desde diferentes navegadoras, a diferentes horas del día y desde diferentes lugares con los cuales se ha tenido acceso a internet.

La historia de usuario 02-C ha sido validada debido a que en el formulario de creación de un nuevo usuario se puede observar los diferentes perfiles de usuario que puede escoger al crear un nuevo usuario.

La historia de usuario 02.1-C ha sido validada debido a que se han comprobado las funciones del perfil administrador: creación de nuevos indicadores, ingreso de valores iniciales, adición de nuevos parámetros, extensión del plazo de indicadores vencidos, y asignación de facilitadores y gestores a los indicadores.

La historia de usuario 02.2-C ha sido validada debido a que se han comprobado las funciones del perfil facilitador: ingreso de valores mensuales en un indicador, creación de nuevos planes de acción para un indicador, creación de tareas para un plan de acción, registro de avances para cada tarea y asignación de colaboradores.

La historia de usuario 02.3-C ha sido validada debido a que se han comprobado las funciones del perfil colaborador: revisión de avances registrados a una tarea, dar por finalizada una tarea de un plan de acción.

La historia de usuario 02.4-C ha sido validada debido a que se han comprobado las funciones del perfil de gestor: aprobación de los datos mensuales ingresados por el facilitador.

La historia de usuario 01-P ha sido validada debido a que se ha realizado una única instalación y configuración del ambiente IaaS contratado, en la máquina virtual de *Azure*.

La historia de usuario 02-P ha sido validada debido a que durante la presentación del presenta trabajo se demostró con varios usuarios que la administración se la realizar exclusivamente en el ambiente IaaS contratado y que esta administración solamente la realiza el proveedor.

La historia de usuario 03-P ha sido validada debido a que el demo creado para el presente trabajo tiene información ficticia, que incluso, no estuvo relacionada entre sí.

La historia de usuario 04-P ha sido validada en conjunto con la historia de usuario 02.1-C.

La historia de usuario 05-P ha sido validada debido a que el CMI ha sido desarrollado el *QlikSense* que es la nueva versión de la plataforma *Qlik*.

4 CONCLUSIONES

4.1 Conclusiones

- La Planificación Estratégica (PE) es un pilar fundamental en toda empresa para tener un mapa cuantitativo y cualitativo de cómo se están llevando a cabo las metas propuestas, y si se están logrando o no, los resultados esperados. Para ello, los indicadores o *Key Performance Indicators* (KPI) son los elementos de medición apropiados para un el análisis del cumplimiento dentro de un Cuadro de Mando Integral (CMI).
- Se ha dado solución al problema de gestión del software de PE para la empresa GRUPO NOVATECH CÍA. LTDA. porque se logró centralizar la implementación, desarrollo y presentación del software de PE con base en el modelo de servicio *Software as a Service* (SaaS). La empresa GRUPO NOVATECH.CÍA. LTDA. ya no tendrá que realizar un nuevo proyecto desde cero cada vez que tenga un nuevo cliente, ahora solamente se crearán nuevos usuarios y nuevos CMI. Los clientes de la empresa GRUPO NOVATECH CÍA. LTDA. perciben un beneficio al acceder directamente a un conjunto de servicios sin necesidad de intervenir en sus propias infraestructuras de datos, además que el departamento de Tecnologías de la información tendrá una carga significativamente menor en comparación con otros sistemas que trabajan bajo el modelo Cliente – Servidor.
- Se pudo observar que los componentes de la solución SaaS para el presente sistema son: una máquina virtual de un proveedor IaaS público sobre la cual se ha instalado una base de datos, un servidor web y una plataforma de *Business Intelligence* (BI). Un conjunto de páginas web publicadas en el servidor web se comunica con la base de datos y permiten que los usuarios ingresen, actualicen y eliminen indicadores de su PE. Esta información es extraída, transformada y presentada en un CMI por la plataforma de BI. Un usuario solamente necesita un navegador web para acceder al sistema SaaS.
- La tecnología de *cloud computing* es de gran ayuda para una solución SaaS ya que si la demanda de clientes aumentara de forma exponencial, solamente escogiendo una máquina virtual de *Azure* de mayores recursos se podría mantener el rendimiento. Caso contrario, si el número de clientes se redujera, sería una buena opción el reducir los recursos de la máquina virtual en vivo para disminuir el pago por uso que haría el proveedor.

- La nube pública del presente sistema es un tipo de solución que permite a las empresas pequeñas y medianas, el gestionar un software de PE sin una mayor inversión tanto en infraestructura como en recursos humanos. El esfuerzo que se requiere está enfocado al diseño, implementación y codificación de nuevos servicios.
- Una solución SaaS es una buena opción para tener recursos bajo demanda en un tiempo mínimo de producción, pero la contraparte es que es un servicio muy personalizado, es decir, que funciona bajo un diseño predeterminado y puede ser que haya empresas que no se puedan acoger al tipo de uso que se le da en el presente sistema. Si se deseara una nueva funcionalidad, será necesario un nuevo alcance para un nuevo desarrollo.
- Es de suma importancia el definir un *Service Level Agreement* (SLA) que permita dar seguridades en el servicio tanto para el proveedor como para el cliente ya que al no existir todavía un marco legal explícitamente enfocado a los servicios de *cloud computing*, pueden darse confusiones en los términos bajo los cuales se ofrece un servicio determinado.
- Actualmente existe demasiada información acerca de las herramientas y servicios de los diferentes proveedores de *Infrastructure as a Service* (IaaS), incluso muchos de ellos tienen paquetes de servicios muy parecidos. Es importante informarse bien, comparar al detalle de las diversas propuestas y evaluarlos contra los requerimientos del sistema con el objetivo de escoger el que sea más beneficioso.
- Para el presente sistema, la plataforma *Qlik*, como herramienta de BI, permitió crear KPI dentro del CMI para el análisis de datos de la PE. Es necesario el ingreso previo de los datos por parte de los usuarios con sus respectivos perfiles para que el flujo de datos sea acorde al diseño y pueda crear un proceso constante de información útil que sirva como una base para la toma de decisiones.
- El presente trabajo podrá servir como base para futuros trabajos de titulación basados en la migración de cualquier tipo de software hacia un modelo de distribución SaaS haciendo uso de todas las ventajas que provee la tecnología *cloud computing*. Tomando en cuenta el alcance del presente trabajo respecto a la automatización de algunos procedimientos, seguridades y financiamiento que no formaban parte del alcance de este prototipo.

4.2 Recomendaciones

- Al usar la plataforma *Qlik* para la extracción de la capa de base de datos en el presente sistema, es posible prever un servicio de extracción de datos de otras fuentes. Es decir, que se podría crear una conexión desde la máquina virtual hacia cualquier otra base de datos y con las mismas etapas de desarrollo, crear nuevos CMI, incluso de otro tipo de datos que no sean relacionados con la PE.
- Durante el monitoreo del uso de recursos del presente sistema, se observó que no se exige mucho a la máquina virtual del proveedor IaaS contratado. Pero es necesario tomar en cuenta un aumento en los recursos para la máquina virtual en caso de que se desee realizar otras soluciones SaaS bajo este mismo esquema.
- El proceso de migración de la solución SaaS no ha contemplado la automatización de escalamiento por parte del cliente debido a las restricciones propias de la empresa GRUPO NOVATECH CÍA. LTDA. asociadas a la diversidad de información, al control de los proyectos y al coste del pago por uso. Para nuevas actualizaciones de la solución se deberá tomar en cuenta estas posibilidades de mejora.
- Durante el desarrollo del presente sistema se ha constatado que el proveedor IaaS *Azure* de *Microsoft* ha cambiado constantemente. Las configuraciones predeterminadas, es decir, las que se han usado para la fase de implementación, no han cambiado. Pero se ha podido observar que se están agregando cada vez más nuevas funcionalidades para los servicios *cloud*. Por eso, es importante estar siempre actualizado con los servicios ofertados para sacarles mayor provecho.
- En el presente sistema, se ha procurado usar software de licenciamiento gratuito aunque la plataforma *Qlik* tiene un esquema de licenciamiento específico. Existe una versión gratuita de la plataforma *Qlik*, llamada *Qlik Cloud* [57] que se podría usar con el correspondiente cambio del diseño e implementación de toda la capa de presentación. Todos los temas de licenciamiento no han sido parte del alcance del presente trabajo, aunque es una variable muy importante a tomar en cuenta, cuando se quiera comercializar una solución SaaS porque estos costos afectan directamente al proveedor del servicio.
- La implementación del presente trabajo ha sido enfocada a los clientes del mercado ecuatoriano para la empresa GRUPO NOVATECH CÍA. LTDA. Como esta empresa

también tiene clientes en otros países, a futuro, se podría usar esta solución para clientes fuera del Ecuador.

- Al ser este un trabajo de prototipo, muchas de las configuraciones se han heredado de la forma en que se realizaba anteriormente, como por ejemplo la habilitación del puerto 81 para la interfaz web, o el uso tanto del puerto 80 como del 443 para el acceso al CMI. Estas opciones podrían irse automatizando o mejorando con nuevas actualizaciones de la solución.
- Aunque es cierto que las máquinas virtuales de *Azure* pueden aumentar los recursos en vivo, habrá que definir el proceso de escalamiento de las bases de datos, debido a que este recurso no lo haría de forma automática. Para ello *Microsoft* tiene varias soluciones que podrían ser de ayuda, aunque con un costo extra.

5 REFERENCIAS BIBLIOGRÁFICAS

- [1] Expansión.com , “Amazon, Microsoft, Google e IBM libran la gran batalla del ‘cloud computing’”, *Expansión.com*, 01-nov-2016. [Online]. Disponible en: <http://www.expansion.com/economia-digital/companias/2016/11/01/581381d8e5fdea8e3e8b4587.html>. [Consultado: 20-mar-2017]
- [2] M. Armijo y G. Pública, “Manual de planificación estratégica e indicadores de desempeño en el sector público”, *Santiago ILPESCEPAL*, 2009 [Online]. Disponible en: http://seieg.iplaneg.net/pmd/doc/santa_catarina/i.insumos/2.planeacion/manual_planificacion_estrategica.pdf. [Consultado: 01-nov-2016]
- [3] Infor, “infor”. [Online]. Disponible en: <http://es.infor.com/>. [Consultado: 25-mar-2017]
- [4] Microsoft, “Microsoft: página de inicio oficial”. [Online]. Disponible en: <https://www.microsoft.com/es-es/>. [Consultado: 25-mar-2017]
- [5] Qlik, “Business Intelligence | Data Visualization Tools | Qlik”. [Online]. Disponible en: <http://www.qlik.com/us>. [Consultado: 25-mar-2017]
- [6] Corporate Planning, “CP Corporate Planning AG”. [Online]. Disponible en: <http://www.corporate-planning.com/es/>. [Consultado: 25-mar-2017]
- [7] Demand Solutions, “Demand Solutions”. [Online]. Disponible en: <http://es.demandsolutions.com/>. [Consultado: 25-mar-2017]
- [8] Grupo Novatech, “Grupo Novatech Ecuador | Grupo Novatech – Quienes Somos?” [Online]. Disponible en: <http://www.grupo-novatech.com/index.php/grupo-novatech/>. [Consultado: 11-ago-2016]
- [9] División Consultoría Evaluando ERP, *SaaS versus Implementación On Premise*. 09-abr-2013 [Online]. Disponible en: <http://www.evaluandoerp.com/implementacion-saas-versus-implementacion-on-premise/>. [Consultado: 21-feb-2017]
- [10] P. Mell y T. Grance, “The NIST definition of cloud computing”, 2011 [Online]. Disponible en: <http://faculty.winthrop.edu/domanm/csci411/Handouts/NIST.pdf>. [Consultado: 12-oct-2016]

- [11] Amazon Web Services, Inc., “Tipos de instancias de Amazon EC2 – Amazon Web Services (AWS)”, *Amazon Web Services, Inc.* [Online]. Disponible en: [//aws.amazon.com/es/ec2/instance-types/](https://aws.amazon.com/es/ec2/instance-types/). [Consultado: 14-ago-2017]
- [12] Microsoft Azure, “Precios - Máquinas virtuales Windows | Microsoft Azure”. [Online]. Disponible en: <https://azure.microsoft.com/es-es/pricing/details/virtual-machines/windows/>. [Consultado: 27-ago-2017]
- [13] Clutch.co, “Best Cloud Service Providers - 2017 Reviews | Clutch.co”. [Online]. Disponible en: <https://clutch.co/cloud#leaders-matrix>. [Consultado: 20-mar-2017]
- [14] Amazon Web Services, Inc., “AWS GovCloud (US) Region Overview – Government Cloud Computing”, *Amazon Web Services, Inc.* [Online]. Disponible en: [//aws.amazon.com/govcloud-us/](https://aws.amazon.com/govcloud-us/). [Consultado: 22-oct-2016]
- [15] IBM, “Cloud computing defined: Characteristics & service levels”, *Cloud computing news*, 31-ene-2014. [Online]. Disponible en: <https://www.ibm.com/blogs/cloud-computing/2014/01/cloud-computing-defined-characteristics-service-levels/>. [Consultado: 24-oct-2016]
- [16] Amazon Web Services, Inc., “Servicios gratuitos de la nube – Capa gratuita de AWS”, *Amazon Web Services, Inc.* [Online]. Disponible en: [//aws.amazon.com/es/free/](https://aws.amazon.com/es/free/). [Consultado: 17-oct-2016]
- [17] D. Mataró, “Costes del Cloud vs Datacenter Propio”. [Online]. Disponible en: <http://www.davidmataro.com/costes-del-cloud-vs-datacenter-propio>. [Consultado: 27-mar-2017]
- [18] S. Rana y P. K. Joshi, “Risk analysis in web applications by using cloud computing”, *Int. J. Multidiscip. Res.*, vol. 2, núm. 1, pp. 386–394, 2012.
- [19] Licencias On Line, “Modelos de Servicio Cloud - SaaS, IaaS, PaaS - LOL Cloud”. [Online]. Disponible en: <http://www.licenciasonline.com/ec/es/cloud/modelos-de-servicio>. [Consultado: 23-abr-2017]
- [20] Google Cloud Platform, “App Engine - Plataforma como servicio”, *Google Cloud Platform*. [Online]. Disponible en: <https://cloud.google.com/appengine/?hl=es>. [Consultado: 24-abr-2017]

- [21] Microsoft Azure, “¿Qué es PaaS? Plataforma como servicio | Microsoft Azure”. [Online]. Disponible en: <https://azure.microsoft.com/es-es/overview/what-is-paas/>. [Consultado: 23-abr-2017]
- [22] L. Badger, T. Grance, R. Patt-Corner, y J. Voas, “Draft cloud computing synopsis and recommendations”, *NIST Spec. Publ.*, vol. 800, p. 146, 2011.
- [23] N. Khanghahi y R. Ravanmehr, “Cloud Computing Performance Evaluation: Issues and Challenges”, *Int. J. Cloud Comput. Serv. Archit.*, vol. 3, núm. 5, pp. 29–41, oct. 2013.
- [24] J. Tessier, “Strategies for Improving Cloud Reliability”, *WIRED*. [Online]. Disponible en: <https://www.wired.com/insights/2012/10/improve-cloud-reliability/>. [Consultado: 30-oct-2016]
- [25] J. Opara-Martins, R. Sahandi, y F. Tian, “Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective”, *J. Cloud Comput.*, vol. 5, núm. 1, dic. 2016 [Online]. Disponible en: <http://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-016-0054-z>. [Consultado: 31-oct-2016]
- [26] Amazon Web Services, Inc., “4 Key Steps Before Migrating Your SaaS Application To The Cloud”, *G2techgroup.com*. 15-dic-2015 [Online]. Disponible en: <https://www.g2techgroup.com/4-key-steps-before-migrating-your-saas-application-to-the-cloud/>. [Consultado: 09-nov-2017]
- [27] Cebit Australia, “Migrating to SaaS: 8 best practices”. [Online]. Disponible en: <http://blog.cebit.com.au/migrating-to-saas-eight-best-practices>. [Consultado: 09-nov-2017]
- [28] Red Iberoamericana de Protección de datos, “Legislación. Ecuador. Red Iberoamericana de Protección de datos”. [Online]. Disponible en: <http://www.redipd.es/legislacion/ecuador-ides-idphp.php>. [Consultado: 01-dic-2016]
- [29] Asamblea Constituyente, “Constitución de la República del Ecuador”, *Quito-Ecuad. Regist. Of.*, vol. 449, pp. 20–10, 2008.
- [30] A. Cabrera Robles, “Transparencia y acceso a la información pública, sujeto a la vigilancia de la defensoría del pueblo en el cumplimiento de la LOTAIP, en las instituciones públicas de Cuenca durante el 2012”, B.S. thesis, 2013 [Online]. Disponible

en: https://educacion.gob.ec/wp-content/uploads/downloads/2015/07/LOTAIP/SNAP/LEY_ORGANICA_DE_TRANSPARENCIA_Y_ACCESO_A_LA_INFORMACION_PUBLICA.pdf. [Consultado: 21-mar-2017]

[31] Ministerio de Telecomunicaciones, “LEY DEL SISTEMA NACIONAL DE REGISTRO DE DATOS PÚBLICOS” [Online]. Disponible en: <https://www.telecomunicaciones.gob.ec/wp-content/uploads/downloads/2012/11/LEY-DEL-SISTEMA-NACIONAL-DE-REGISTRO-DE-DATOS-PUBLICOS.pdf>. [Consultado: 21-mar-2017]

[32] A. Carrasco Guzmán, “ESTUDIO DE FACTIBILIDAD PARA LA COMERCIALIZACIÓN ON-LINE DE EQUIPOS IMPORTADOS DESDE EE. UU.”, 2010 [Online]. Disponible en: http://www.redipd.org/legislacion/common/legislacion/ecuador/ecuador_ley_2002-67_17042002_comelectronico.pdf

[33] Cloud Standards Customer Council, “Practical Guide to Cloud Service Agreements - CSCC-Practical-Guide-to-Cloud-Service-Agreements.pdf”. [Online]. Disponible en: <http://www.cloud-council.org/deliverables/CSCC-Practical-Guide-to-Cloud-Service-Agreements.pdf>. [Consultado: 02-dic-2016]

[34] L. Arana, M. Ruiz, y N. La Serna, “Análisis de aplicaciones empleando la computación en la nube de tipo PaaS y la metodología ágil Scrum”, *Rev. Fac. Ing. Ind. UNMSM*, vol. 18, núm. 1, feb. 2015 [Online]. Disponible en: revistasinvestigacion.unmsm.edu.pe/index.php/idata/article/download/12077/10792

[35] Scrum Manager, “Historia de usuario - Scrum Manager BoK”. [Online]. Disponible en: http://www.scrummanager.net/bok/index.php?title=Historia_de_usuario. [Consultado: 25-may-2017]

[36] K. Schwaber y M. Beedle, *Agile Software Development with Scrum*. USA: Prentice Hall, 2001.

[37] K. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. USA: Addison-Wesley, 2012.

[38] Gartner, “Business Intelligence - BI - Gartner IT Glossary”. [Online]. Disponible en: <http://www.gartner.com/it-glossary/business-intelligence-bi/>. [Consultado: 12-oct-2017]

[39] C. Mabardy, "Qlik's Associative Model". 03-ene-2017 [Online]. Disponible en: <https://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwjr3ePE7erWAhWqxVQKHSQyBRsQFggIMAA&url=http%3A%2F%2Fglobal.qlik.com%2Fuk%2F~%2Fmedia%2FFiles%2Fresource-library%2Fglobal-us%2Fregister%2Fwhitepapers%2FWP-Qlik-Associative-Model-EN&usg=AOvVaw2rbvrosTYt8-rQcQ6QDrSY>

[40] Qlik Qonnections 2018, *Qlik Qonnections 2018*. [Online]. Disponible en: <https://www.qlikqonnections.com/>. [Consultado: 13-nov-2017]

[41] Data IQ, "Conocé los casos de éxito de Qonnections 2017", *Blog Data IQ*. 31-jul-2017 [Online]. Disponible en: <https://dataiq.com.ar/blog/conoce-los-casos-exito-qonnections-2017-whirlpool/>. [Consultado: 13-nov-2017]

[42] Gartner, "Gartner Reprint". [Online]. Disponible en: <https://www.gartner.com/doc/reprints?id=1-2G2O5FC&ct=150519>. [Consultado: 13-ago-2017]

[43] Amazon Web Services, Inc., "AWS | Elastic block store (EBS) para almacenamiento persistente", *Amazon Web Services, Inc.* [Online]. Disponible en: <https://aws.amazon.com/es/ebs/>. [Consultado: 14-ago-2017]

[44] Cloud Endure, "7 Crucial Points on Cloud Computing Providers", *CloudEndure*, 21-sep-2016. [Online]. Disponible en: <https://www.cloudendure.com/blog/7-things-to-watch-out-for-cloud-provider-sla/>. [Consultado: 12-ago-2017]

[45] Computer World Venezuela, "La batalla de las nubes AWS vs. Microsoft Azure vs. Google Cloud Platform | Computerworld Venezuela". [Online]. Disponible en: <http://www.cww.com.ve/la-batalla-de-las-nubes-aws-vs-microsoft-azure-vs-google-cloud-platform/>. [Consultado: 03-ago-2017]

[46] K. Weins, "AWS vs Azure vs Google Cloud Pricing: Compute Instances". [Online]. Disponible en: <https://www.rightscale.com/blog/cloud-cost-analysis/aws-vs-azure-vs-google-cloud-pricing-compute-instances>. [Consultado: 12-ago-2017]

[47] Cloud Berry Lab, "Azure VM vs Amazon EC2 vs Google CE: Cloud Computing Comparison", *CloudBerry Lab Blog*. [Online]. Disponible en: <https://www.cloudberrylab.com/blog/azure-vm-vs-amazon-ec2-vs-google-ce-cloud-computing-comparison/>. [Consultado: 12-ago-2017]

- [48] K. Cevallos, "UML: Diagrama de Secuencia", *INGENIERÍA DEL SOFTWARE*. 07-jul-2015 [Online]. Disponible en: <https://ingsoftwarekarlacevallos.wordpress.com/2015/07/07/uml-diagrama-de-secuencia/>. [Consultado: 23-ago-2017]
- [49] P. Gutiérrez, "Fundamento de las bases de datos: Modelo entidad-relación", *Genbeta Dev*, 05-nov-2013. [Online]. Disponible en: <https://www.genbetadev.com/bases-de-datos/fundamento-de-las-bases-de-datos-modelo-entidad-relacion>. [Consultado: 23-sep-2017]
- [50] Microsoft, "Diagramas de casos de uso de UML: Instrucciones". [Online]. Disponible en: <https://msdn.microsoft.com/es-es/library/dd409432.aspx>. [Consultado: 21-sep-2017]
- [51] F. Grau y S. Segura, "Desarrollo orientado a objetos con UML", *Fac. Informática-UPM Artículo*, vol. 3, 2001 [Online]. Disponible en: http://moodle2.unid.edu.mx/dts_cursos_md/lic/IEL/SI/AM/11/UML.pdf. [Consultado: 18-sep-2017]
- [52] A. Holub, "Allen Holub's UML Quick Reference – Allen Holub". [Online]. Disponible en: <http://holub.com/uml/>. [Consultado: 18-sep-2017]
- [53] NativeHex, "Relaciones entre clases [UML]", *NativeHex*, 09-jun-2014. [Online]. Disponible en: <https://blog.nativehex.com/relaciones-entre-clases-uml>. [Consultado: 20-ago-2017]
- [54] Qlik Sense, "Ports in a default Qlik Sense installation – Qlik Sense". [Online]. Disponible en: https://help.qlik.com/en-us/sense/1.0/Subsystems/Installation_Guide/Content/Deployments/SRM_Deployment_Ports.htm. [Consultado: 11-sep-2017]
- [55] Microsoft, "Microsoft Open Database Connectivity (ODBC)". [Online]. Disponible en: <https://docs.microsoft.com/en-us/sql/odbc/microsoft-open-database-connectivity-odbc>. [Consultado: 11-sep-2017]
- [56] Microsoft, "Appendix B: Sample SLA". [Online]. Disponible en: <https://technet.microsoft.com/en-us/library/cc543293.aspx>. [Consultado: 16-nov-2017]
- [57] Qlik Sense Cloud, "Qlik Sense Cloud". [Online]. Disponible en: <https://www.qlikcloud.com/>. [Consultado: 23-oct-2017]

6 ANEXOS

Anexo A. SCRIPT DE LA CAPA DE BASE DE DATOS

Anexo B. SCRIPT DE LA CAPA DE NEGOCIOS

Anexo C. SCRIPT DE LA CAPA DE PRESENTACIÓN

Anexo D. REDACCIÓN DEL SLA

ANEXO A

SCRIPT DE LA CAPA DE BASE DE DATOS

```
CREATE TABLE [dbo].[ARBOL_INDICADOR] (  
    [IDARBOL] int IDENTITY(1, 1) NOT NULL,  
    [IDOBJETIVO] int NULL,  
    [IDINDICADOR] int NULL,  
    [INDICADOR_PADRE] int NULL,  
    [PONDERACION] decimal(10, 3) NULL  
)  
ON [PRIMARY]  
GO  
  
CREATE TABLE [dbo].[ARBOL_INDICADOR_VALORES] (  
    [ID] int IDENTITY(1, 1) NOT NULL,  
    [VALOR_META] decimal(10, 3) NULL,  
    [VALOR_EJECUTADO] decimal(10, 3) NULL,  
    [FECHA_META] datetime DEFAULT getdate() NULL,  
    [FECHA_EJECUTADO] datetime NULL,  
    [IDINDICADOR] int NULL,  
    [IDDESPLIEGUE] int NULL,  
    [IDARBOL] int NULL,  
    [PERIODO] int NULL,  
    [ANIO] int NULL  
)  
ON [PRIMARY]  
GO  
  
CREATE TABLE [dbo].[AVANCE] (  
    [IDAVANCE] int IDENTITY(1, 1) NOT NULL,  
    [DESCRIPCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,  
    [AVANCE] int NOT NULL,  
    [IDTAREA] int NOT NULL,  
    [DOCUMENTO] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL,  
    [FECHACREACION] datetime NULL,  
    [USUARIO] int NULL  
)  
ON [PRIMARY]  
GO  
  
CREATE TABLE [dbo].[CAMPOS_ADICIONALES] (  
    [IDCAMPO] int IDENTITY(1, 1) NOT NULL,  
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL  
)  
ON [PRIMARY]  
GO  
  
CREATE TABLE [dbo].[CARGO] (  
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,  
    [DESCRIPCION] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL,  
    [IDPADRE] int DEFAULT 0 NULL,  
    [ESTADO] int DEFAULT 1 NULL,  
    [RESPONSABLE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,  
    [EMAIL] varchar(128) COLLATE Modern_Spanish_CI_AS NULL,  
    [IDCARGO] int IDENTITY(1, 1) NOT NULL  
)  
)
```



```
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[COLABORADOR] (
  [IDCOLABORADOR] int IDENTITY(1, 1) NOT NULL,
  [IDUSUARIO] int NOT NULL,
  [IDPLAN] int NOT NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[CONTROL_CAMBIOS] (
  [ID] int IDENTITY(1, 1) NOT NULL,
  [IDUSUARIO] int NULL,
  [ACCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [FECHA] datetime DEFAULT getdate() NULL,
  [VALOR_ANTERIOR] nvarchar(16) COLLATE Modern_Spanish_CI_AS NULL,
  [VALOR_NUEVO] nvarchar(16) COLLATE Modern_Spanish_CI_AS NULL,
  [ARCHIVO] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [COMENTARIO] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [IDINDICADOR] int NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[DEPARTAMENTO] (
  [IDDEPARTAMENTO] int IDENTITY(1, 1) NOT NULL,
  [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
  [DESCRIPCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [IDPADRE] int NULL,
  [ESTADO] int DEFAULT 1 NOT NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[DEPARTAMENTO_OBJETIVO] (
  [ID] int IDENTITY(1, 1) NOT NULL,
  [IDDEPARTAMENTO] int NOT NULL,
  [IDOBJETIVO] int NOT NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[DESPLIEGUE] (
  [IDDESPLIEGUE] int IDENTITY(1, 1) NOT NULL,
  [IDPADRE] int NULL,
  [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NOT NULL,
  [CODIGO] nvarchar(16) COLLATE Modern_Spanish_CI_AS NULL,
  [DESCRIPCION] varchar(512) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[EXTENSIONES_META] (
  [ID] int IDENTITY(1, 1) NOT NULL,
  [IDINDICADOR] int NULL,
  [FECHA] datetime NULL,
  [FECHAINGRESO] datetime NULL
)
```

```

)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[FORMULA] (
  [IDFORMULA] int IDENTITY(1, 1) NOT NULL,
  [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NOT NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[FRECUENCIA] (
  [IDFRECUENCIA] int IDENTITY(1, 1) NOT NULL,
  [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
  [MESES] int NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[INDICADOR_CAMPOS] (
  [ID] int IDENTITY(1, 1) NOT NULL,
  [IDCAMPO] int NULL,
  [IDINDICADOR] int NULL,
  [VALOR] nvarchar(253) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[INDICADOR_PLANACCION] (
  [IDINDPLAN] int NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[INDICADORES] (
  [IDINDICADOR] int IDENTITY(1, 1) NOT NULL,
  [CODIGO] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,
  [NIVEL ESTRATEGICO] int NULL,
  [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
  [UNIDAD] int NULL,
  [DEFINICION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [FORMULA] varchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [SEGMENTO] int NULL,
  [PRODUCTO] int NULL,
  [TIPO] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,
  [DESPLIEGUE] int NULL,
  [FRECUENCIA] int NULL,
  [RESPONSABLE] int NULL,
  [FACILITADOR] int NULL,
  [VIGENTE] int NULL,
  [ESTADO] int NULL,
  [COMENTARIO] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
  [VIGENTE_DESDE] datetime DEFAULT getdate() NULL,
  [VIGENTE_HASTA] datetime NULL,
  [MANUAL] int NULL,
  [RESPONSABLE_GERENCIA] int NULL,
  [SIGNO] char(1) COLLATE Modern_Spanish_CI_AS DEFAULT '+' NULL
)

```

```
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[OBJETIVO] (
    [IDOBJETIVO] int IDENTITY(1, 1) NOT NULL,
    [CODIGO] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
    [OBJETIVO_PADRE] int NULL,
    [SIGNO] char(1) COLLATE Modern_Spanish_CI_AS NULL,
    [OPERACION] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[PARAMETROS_INDICADORES] (
    [DIA_INICIO] int NULL,
    [DIAS_INGRESO] int NULL,
    [MAXIMO_INGRESO] int NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[PERFIL] (
    [ID_PERFIL] int IDENTITY(1, 1) NOT NULL,
    [NOMBRE_PERFIL] varchar(64) COLLATE Modern_Spanish_CI_AS NOT NULL,
    [CODIGO_PERFIL] char(1) COLLATE Modern_Spanish_CI_AS NOT NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[PLANACCION] (
    [IDPLANACCION] int IDENTITY(1, 1) NOT NULL,
    [DEFINICION] nvarchar(128) COLLATE Modern_Spanish_CI_AS NOT NULL,
    [JUSTIFICACION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NOT NULL,
    [ACCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
    [VALORMETA] int NULL,
    [PRESUPUESTO] int NULL,
    [IDINDICADOR] int NOT NULL,
    [RESPONSABLE] int NOT NULL,
    [FECHA1] datetime NULL,
    [CARGO1] int NULL,
    [DOCUMENTO] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL,
    [OTROSRECURSOS] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL,
    [FECHACREACION] date NULL,
    [OBSERVACIONES] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[PRODUCTO] (
    [IDPRODUCTO] int IDENTITY(1, 1) NOT NULL,
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
    [DESCRIPCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO
```

```
CREATE TABLE [dbo].[SEGMENTACION] (
```

```

    [NIVEL] int NULL,
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[SEGMENTOS] (
    [IDSEGMENTO] int IDENTITY(1, 1) NOT NULL,
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
    [DESCRIPCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[TAREAPLAN] (
    [IDTAREAPLAN] int IDENTITY(1, 1) NOT NULL,
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NOT NULL,
    [ALCANCE] nvarchar(512) COLLATE Modern_Spanish_CI_AS NOT NULL,
    [ENTREGABLE] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
    [FECHAINICIO] date NULL,
    [FECHAFIN] date NULL,
    [PRESUPUESTO] int NULL,
    [FECHA1] date NULL,
    [CARGO1] int NULL,
    [FECHA2] date NULL,
    [CARGO2] int NULL,
    [FECHA3] date NULL,
    [CARGO3] int NULL,
    [RESPONSABLE] int NOT NULL,
    [DOCUMENTO] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL,
    [CUMPLIDO] int DEFAULT 0 NULL,
    [AVANCE] nvarchar(256) COLLATE Modern_Spanish_CI_AS NULL,
    [JUSTIFICACION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL,
    [ESTADO] nvarchar(65) COLLATE Modern_Spanish_CI_AS NULL,
    [ESTADOAVANCE] nvarchar(65) COLLATE Modern_Spanish_CI_AS NULL,
    [IDPLAN] int NOT NULL,
    [FECHACUMPLIDO] date NULL,
    [USUARIOCUMPLIDO] int NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[UNIDADES] (
    [IDUNIDAD] int IDENTITY(1, 1) NOT NULL,
    [NOMBRE] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
    [SIMBOLO] varchar(16) COLLATE Modern_Spanish_CI_AS NULL,
    [DESCRIPCION] nvarchar(512) COLLATE Modern_Spanish_CI_AS NULL
)
ON [PRIMARY]
GO

CREATE TABLE [dbo].[USUARIO] (
    [IDUSUARIO] int IDENTITY(1, 1) NOT NULL,
    [NOMBRE] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,
    [APELLIDO] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,
    [EMAIL] nvarchar(128) COLLATE Modern_Spanish_CI_AS NULL,
    [PWD] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,
    [PWD_INIT] nvarchar(64) COLLATE Modern_Spanish_CI_AS NULL,

```

```
[ESTADO] char(1) COLLATE Modern_Spanish_CI_AS NULL,  
[TIPO] varchar(64) COLLATE Modern_Spanish_CI_AS,  
[ELIMINADO] int NULL,  
[FECHA] datetime DEFAULT getdate() NULL  
)  
ON [PRIMARY]  
GO
```

ANEXO B

SCRIPT DE LA CAPA DE NEGOCIOS

Por su extensión, el Anexo B se encuentra en el Disco anexo al presente trabajo de titulación.

ANEXO C

SCRIPT DE LA CAPA DE PRESENTACIÓN

Extracción

```
LIB CONNECT TO 'Demo';

MAPPING
temp_Caracteres:
LOAD * INLINE [
    from,to
    Ã-,í
    Ã@,é
    Ã³,ó
    Ã±,ñ
    Ã¡,á
    Ãº,ú
    Ã,Á
    Á“,Ó
    Ã-,í
    Â°,°
    Á‘,Ñ
    Ã, Í
    Á☐, Í
    Ã☐, Í
    Ãš, Ú
    Ã-, í
    Ãº, ú
    Ã³, ó
    Ã¡, á
    Ãº, ú
    Ã@, é
];

dm_Arbol_Indicadores:
LOAD
    IDOBJETIVO as Id_Objetivo,
    IDINDICADOR as Id_Indicador,
    "INDICADOR_PADRE" as Id_Padre_Indicador,
    PONDERACION as Ponderacion_Indicador;
SQL SELECT *
FROM UTE.dbo."ARBOL_INDICADOR";

STORE dm_Arbol_Indicadores into [lib://qvdBase/dm_Arbol_Indicadores.qvd];
DROP Table dm_Arbol_Indicadores;

fc_Arbol_Indicadores_Valores:
LOAD
    "VALOR_META" as Valor_Meta_Indicador,
    VALOR_EJECUTADO as Valor_Ejecutado_Indicador,
    Date("FECHA_META") as Fecha_Meta_Indicador,
    Date("FECHA_EJECUTADO") as Fecha_Ejecutado_Indicador,
    IDINDICADOR as Id_Indicador,
```

```

PERIODO as Periodo_Indicador,
ANIO as Año_Indicador,
[VALIDADO_GESTION] as Validado_Indicador,
Date([FECHA_VALIDADO_GESTION]) as Fecha_Validado_Indicador,
[USUARIO_GESTION_VALIDA] as Id_Usuario_Valida_Indicador,
[IDDESPLIEGUE] as Id_Despliegue
;
SQL SELECT *
FROM UTE.dbo."ARBOL_INDICADOR_VALORES";

STORE fc_Arbol_Indicadores_Valores into
[lib://qvdBase/fc_Arbol_Indicadores_Valores.qvd];
DROP Table fc_Arbol_Indicadores_Valores;

dm_Avance_Tareas:
LOAD IDAVANCE as Id_Avance_Tarea,
     mapsubstring('temp_Caracteres',DESCRIPCION) as Descripcion_Avance_Tarea,
     AVANCE as Valor_Avance_Tarea,
     IDTAREA as Id_Tarea,
     Date(FECHACREACION) as Fecha_Creacion_Avance_Tarea,
     [USUARIO] as Id_Usuario_Avance_Tarea
;
SQL SELECT *
FROM UTE.dbo.AVANCE;

STORE dm_Avance_Tareas into [lib://qvdBase/dm_Avance_Tareas.qvd];
DROP Table dm_Avance_Tareas;

dm_Frecuencia:
LOAD IDFRECUENCIA as Id_Frecuencia,
     mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Frecuencia;
SQL SELECT *
FROM UTE.dbo.FRECUENCIA;

STORE dm_Frecuencia into [lib://qvdBase/dm_Frecuencia.qvd];
DROP Table dm_Frecuencia;

dm_Indicadores:
LOAD IDINDICADOR as Id_Indicador,
     CODIGO asCodigo_Indicador,
     "NIVEL ESTRATEGICO" as Nivel_Estrategico_Indicador,
     mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Indicador,
     UNIDAD as Id_Unidad_Indicador,
     mapsubstring('temp_Caracteres',DEFINICION) as Definicion_Indicador,
     FRECUENCIA as Id_Frecuencia,
     MANUAL as Manual,
     FORMULA as Id_Formula,
     SIGNO as Signo_Indicador,
     [FACILITADOR] as Id_Usuario_Facilitador_Indicador,
     [RESPONSABLE_GERENCIA] as Id_Usuario_Gestor_Indicador
;
SQL SELECT *
FROM UTE.dbo.INDICADORES;

STORE dm_Indicadores into [lib://qvdBase/dm_Indicadores.qvd];
DROP Table dm_Indicadores;

dm_Objetivos:

```



```

LOAD
    IDOBJETIVO as Id_Objetivo,
    CODIGO asCodigo_Objetivo,
    [OBJETIVO_PADRE] as Objetivo_Padre,
    mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Objetivo;
SQL SELECT *
FROM UTE.dbo.OBJETIVO;

```

```

STORE dm_Objeticos into [lib://qvdBase/dm_Objeticos.qvd];
DROP Table dm_Objeticos;

```

```

dm_Plan:
LOAD IDPLANACCION as Id_Plan,
    mapsubstring('temp_Caracteres',DEFINICION) as Definicion_Plan,
    mapsubstring('temp_Caracteres',JUSTIFICACION) as Justificacion_Plan,
    mapsubstring('temp_Caracteres',ACCION) as Accion_Plan,
    IDINDICADOR as Id_Indicador,
    Date(FECHACREACION) as Fecha_Creacion_Plan,
    [VALORMETA] as Meta_Plan,
    [PRESUPUESTO] as Presupuesto_Plan,
    [RESPONSABLE] as Id_Usuario_Plan,
    Date([FECHA1]) as Fecha_Fin_Plan,
    mapsubstring('temp_Caracteres',[OBSERVACIONES]) as Observacion_Plan;
SQL SELECT *
FROM UTE.dbo.PLANACCION;

```

```

STORE dm_Plan into [lib://qvdBase/dm_Plan.qvd];
DROP Table dm_Plan;

```

```

dm_Tarea:
LOAD IDTAREAPLAN as Id_Tarea,
    mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Tarea,
    mapsubstring('temp_Caracteres',ALCANCE) as Alcance_Tarea,
    mapsubstring('temp_Caracteres',ENTREGABLE) as Entregable_Tarea,
    Date(FECHAINICIO) as Fecha_Inicio_Tarea,
    Date(FECHAFIN) as Fecha_Fin_Tarea,
    PRESUPUESTO as Presupuesto_Tarea,
    mapsubstring('temp_Caracteres',ESTADO) as Estado_Tarea,
    IDPLAN as Id_Plan,
    Date(FECHACUMPLIDO) as Fecha_Cumplido_Tarea,
    [RESPONSABLE] as Id_Usuario_Tarea,
    mapsubstring('temp_Caracteres',[JUSTIFICACION]) as Justificacion_Tarea,
    [USUARIOCUMPLIDO] as Id_Usuario_Cumplido_Tarea;
SQL SELECT *
FROM UTE.dbo.TAREAPLAN;

```

```

STORE dm_Tarea into [lib://qvdBase/dm_Tarea.qvd];
DROP Table dm_Tarea;

```

```

dm_Unidad:
LOAD IDUNIDAD as Id_Unidad_Indicador,
    mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Unidad,
    SIMBOLO as Simbolo_Unidad;
SQL SELECT *
FROM UTE.dbo.UNIDADES;

```

```

STORE dm_Unidad into [lib://qvdBase/dm_Unidad.qvd];
DROP Table dm_Unidad;

```

```

dm_Departamentos:
LOAD
    IDDEPARTAMENTO as Id_Departamento,
    mapsubstring('temp_Caracteres',NOMBRE) as Nombre_Departamento,
    mapsubstring('temp_Caracteres',DESCRIPCION) as Descripcion_Departamento,
    IDPADRE as Id_Padre_Departamento,
    ESTADO as Estado_Departamento;
SQL SELECT *
FROM UTE.dbo.DEPARTAMENTO;

STORE dm_Departamentos into [lib://qvdBase/dm_Departamentos.qvd];
DROP Table dm_Departamentos;

dm_Vinculados:
LOAD
    IDDEPARTAMENTO as Id_Departamento,
    IDOBJETIVO as Id_Objetivo;
SQL SELECT *
FROM UTE.dbo."DEPARTAMENTO_OBJETIVO";

STORE dm_Vinculados into [lib://qvdBase/dm_Vinculados.qvd];
DROP Table dm_Departamentos;

dm_Despliegue:
LOAD IDDESPLIEGUE as Id_Despliegue,
    IDPADRE as Id_Padre_Despliegue,
    NOMBRE as Nombre_Despliegue,
    CODIGO as Cod_Despliegue,
    DESCRIPCION as Descripcion_Despliegue;
SQL SELECT *
FROM Demo.dbo.DESPLIEGUE;

STORE dm_Despliegue into [lib://qvdBase/dm_Despliegue.qvd];
DROP Table dm_Departamentos;

```

Transformación

```

Hierarchy(Id_Objetivo,Objetivo_Padre,Nombre_Objetivo)
LOAD
    Id_Objetivo,
    Codigo_Objetivo,
    Nombre_Objetivo,
    Objetivo_Padre
FROM [lib://qvdBase/dm_Objetivos.qvd]
(qvd);

Hierarchy(Id_Departamento,Id_Padre_Departamento,Nombre_Departamento)
LOAD
    Id_Departamento,
    Nombre_Departamento,
    Descripcion_Departamento,
    Id_Padre_Departamento,
    Estado_Departamento
FROM [lib://qvdBase/dm_Departamentos.qvd](qvd);

```

ANEXO D

REDACCIÓN DEL SLA

A continuación, se presenta el texto que formará parte del documento que deberán firmar ambas partes luego de haber cerrado el proceso de negociación para la adquisición del servicio SaaS del software de PE de la empresa GRUPO NOVATECH CÍA. LTDA.:

- 1) El cliente tendrá a su disposición el completo funcionamiento del software de PE bajo los términos definidos en este presente documento. Cualquier tipo de interferencia en el funcionamiento deberá ser reportado al proveedor para su presente proceso de soporte. El cliente se compromete a pagar una cuota mensual al proveedor por el presente servicio.
- 2) Los tipos de interferencias que podrán ser soportados por el proveedor son los siguientes: caída de servicio por fallas explícitas de infraestructura interna, pérdidas de credenciales de acceso por parte del cliente, pérdida de información por fallas explícitas de la base de datos interna, errores explícitos del software en cualquiera de los procesos. Las anteriores causas de interferencias serán corregidas con base en una validación previa del proveedor. Cualquier otro tipo de interferencia deberá ser justificada por parte del cliente para iniciar un proceso de corrección.
- 3) El servicio contratado tiene el siguiente funcionamiento estándar: acceso a cualquier hora y desde cualquier lugar, acceso desde cualquier navegador web, envío de notificaciones por correo electrónico a una cuenta definida por el cliente, una interfaz web para ingreso de datos, una interfaz web para visualización de los datos ingresados, actualización diaria de los datos para la interfaz de visualización a una hora específica definida previamente con el cliente, ingreso y visualización de datos específicamente de la PE del cliente.
- 4) El uso del software será monitoreado diariamente por parte del proveedor para asegurar un rendimiento estable. El proveedor asegura el funcionamiento estándar con los recursos computacionales con los que se contrata actualmente el servicio, si el cliente requiere más recursos computacionales, se deberá realizar un estudio previo porque esto significará un aumento en el pago mensual.
- 5) El proveedor podrá asegurar la privacidad de los datos del cliente en el manejo dentro del servidor exclusivamente porque ninguna otra persona que no sea el

mismo proveedor podrá acceder a la infraestructura interna sobre la cual funciona el servicio. El cliente tiene libre acceso por medio de las interfaces del software, tanto de ingreso de datos como de su visualización. El cliente acepta los riesgos de usar el servicio con las configuraciones por defecto. Si el cliente desea una mayor seguridad en la transmisión de los datos, se deberá diseñar un plan de seguridad personalizado con un costo asociado al nivel de seguridad deseado.

- 6) El servicio está a disposición de los clientes para su uso constante, a cualquier hora y desde cualquier locación. El proveedor solamente podrá interrumpir el funcionamiento estándar del servicio con base en procesos internos programados con anticipación, agendando un tiempo limitado para estas actividades y notificando al cliente con 15 días de anticipación. El proveedor no se hace responsable de cualquier tipo de interrupción del servicio de internet del cliente que obstaculice el acceso a la solución.
- 7) El proveedor tiene la responsabilidad de respaldar la información del cliente cada semana como punto de partida para una recuperación ante fallos. Todos los procesos de aseguramiento de la información los realizará exclusivamente el proveedor. El proceso de respaldo consta de realizar copias del proyecto de cada cliente, tanto de la base de datos, de los componentes de la interfaz web y de CMI.
- 8) Si por alguna razón el cliente da por terminado el servicio contratado, se deberá acoger al proceso de terminación de servicio realizado por el proveedor. La información permanecerá activa por un plazo de un mes desde la fecha de culminación. Si no existiera una reactivación del servicio, toda la información relacionada con el cliente será eliminada permanentemente del software. El proceso de dada de baja de un cliente constará en eliminar toda la información relacionada a su proyecto, es decir, eliminar permanentemente todos los respaldos de la base de datos, de los componentes de la interfaz web y del CMI.

El presente documento será revisado anualmente y podrá ser cambiado previo a un acuerdo explícito del cliente y del proveedor.

ORDEN DE EMPASTADO