



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**DISEÑO DE INVESTIGACIÓN PARA LA IMPLEMENTACIÓN DE HERRAMIENTA PARA LA
PREDICCIÓN DEL USO DE LOS RECURSOS DE LA PLATAFORMA DE DATOS MYSQL
DEL SISTEMA DE CONTROL ACADÉMICO DE LA FACULTAD DE HUMANIDADES DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Carlos Arturo López Taks

Asesorado por el Msc. Ing. Juan Fernando Cano Villatoro

Guatemala, noviembre de 2021

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DISEÑO DE INVESTIGACIÓN PARA LA IMPLEMENTACIÓN DE HERRAMIENTA PARA LA
PREDICCIÓN DEL USO DE LOS RECURSOS DE LA PLATAFORMA DE DATOS MYSQL
DEL SISTEMA DE CONTROL ACADÉMICO DE LA FACULTAD DE HUMANIDADES DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

CARLOS ARTURO LÓPEZ TAKS

ASESORADO POR EL MSC. ING. JUAN FERNANDO CANO VILLATORO

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, NOVIEMBRE DE 2021

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANA	Inga. Aurelia Anabela Cordova Estrada
VOCAL I	Ing. José Francisco Gómez Rivera
VOCAL II	Ing. Mario Renato Escobedo Martínez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Kevin Armando Cruz Lorente
VOCAL V	Br. Fernando José Paz González
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Pedro Antonio Aguilar Polanco
EXAMINADOR	Ing. Sergio Arnaldo Méndez Aguilar
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
EXAMINADOR	Ing. Miguel Ángel Cancinos Rendón
SECRETARIA	Inga. Lesbia Magalí Herrera López

HONORABLE TRIBUNAL EXAMINADOR

En cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**DISEÑO DE INVESTIGACIÓN PARA LA IMPLEMENTACIÓN DE HERRAMIENTA PARA LA
PREDICCIÓN DEL USO DE LOS RECURSOS DE LA PLATAFORMA DE DATOS MYSQL
DEL SISTEMA DE CONTROL ACADÉMICO DE LA FACULTAD DE HUMANIDADES DE LA
UNIVERSIDAD DE SAN CARLOS DE GUATEMALA**

Tema que me fuera asignado por la Dirección de la Escuela de Estudios de Postgrado, con fecha 30 de enero de 2021.

Carlos Arturo López Taks

ACTO QUE DEDICO A:

- Dios** Por haberme permitido completar esta meta y por la fuerza que me das cada día.
- Mis padres** Por ser mi guía y mis mentores. Mi eterno agradecimiento por su apoyo y sustento en mi formación.
- Mi esposa** Por ser mi compás, mi fuerza y mi apoyo en todos los proyectos que emprendemos juntos.
- Mi abuelita y mi tío** Elva Sánchez (q. d. e. p.) y Julio López por sus sabias enseñanzas, consejos y cariño durante toda mi vida.
- Mis hermanos** Carolina, Nicolas López y Michelle Garzaro Taks por su apoyo y compañía durante mi vida.

AGRADECIMIENTOS A:

Universidad de San Carlos de Guatemala	Por ser mi <i>alma mater</i> que me permitió adquirir mis conocimientos.
Facultad de Ingeniería	Por proporcionarme los conocimientos que me han formado profesionalmente.
Facultad de Humanidades	Por el apoyo que hace posible la realización de este trabajo de graduación.
Mi asesor	Mtro. Ing. Fernando Cano, por su guía y apoyo durante el proceso de la realización de este diseño de investigación.
Familia y amigos en general	Por compartir los momentos más importantes de mi vida gracias.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
LISTA DE SÍMBOLOS	VII
GLOSARIO.....	IX
RESUMEN	XIII
1. INTRODUCCIÓN.....	1
2. ANTECEDENTES.....	3
3. PLANTEAMIENTO DEL PROBLEMA	9
4. JUSTIFICACIÓN.....	13
5. OBJETIVOS	15
5.1. General	15
5.2. Específicos	15
6. NECESIDADES POR CUBRIR Y ESQUEMA DE SOLUCIÓN	17
6.1. Necesidades por cubrir	17
6.2. Esquema de la solución.....	19
7. MARCO TEÓRICO	23
7.1. Configuración alta disponibilidad en la plataforma de datos	23
7.1.1. Definición de alta disponibilidad	23
7.1.2. Arquitectura de clúster en Linux.....	24

7.1.3.	Tipos de clústeres.....	25
7.1.4.	Recursos de clúster	26
7.1.5.	Arquitectura de replicación de datos en Mysql	27
7.1.6.	Descripción de componentes.....	28
7.2.	Teorema de CAP.....	29
7.2.1.	Concurrencia.....	29
7.2.2.	Disponibilidad.....	30
7.2.3.	Tolerancia a la partición	30
7.3.	Plan de recuperación de desastres en la plataforma de datos	30
7.4.	Plan de continuidad de negocio en la plataforma de datos	32
7.5.	El uso de las nubes híbridas en plataforma de datos	35
7.6.	Los componentes clave en las nubes híbridas.....	36
7.7.	Costos de uso de la infraestructura sobre nube híbrida	37
7.8.	Integración de monitoreo en la nube híbrida	38
7.9.	Integración con el servicio en la nube	40
8.	PROPUESTA DE ÍNDICE DE CONTENIDOS.....	41
9.	METODOLOGÍA	43
9.1.	Tipo de estudio	43
9.2.	Diseño	43
9.3.	Alcance	44
9.4.	Variables	44
9.5.	Fases del estudio	45
9.5.1.	Fase 0: documentación inicial.....	45
9.5.2.	Fase 1: diseño y desarrollo de la herramienta	47
9.5.3.	Fase 2: migración a servidor de producción	48
9.5.4.	Fase 3: construcción de plataforma híbrida	50

9.5.5.	Fase 4: presentación de resultados	51
9.6.	Técnicas de recolección de información.....	52
10.	TÉCNICAS DE ANÁLISIS DE INFORMACIÓN	57
11.	CRONOGRAMA	61
12.	FACTIBILIDAD DEL ESTUDIO.....	65
12.1.	Factibilidad operativa	65
12.2.	Factibilidad técnica.....	68
12.3.	Factibilidad económica	70
13.	REFERENCIAS	73

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Plataforma de datos actual	20
2.	Plataforma de datos híbrida.....	21
3.	Cronograma de actividades	61
4.	Planificación general del proyecto.....	67

TABLAS

I.	Inventario de activos de base de datos.....	33
II.	Definición de variables	44
III.	Estimación de hardware.....	68
IV.	Estimación de licencias de software.....	69
V.	Especificaciones de software.....	69
VI.	Diagrama de flujo de caja	72

LISTA DE SÍMBOLOS

Símbolo	Significado
@	Arroba o At
S^2	Desviación standard
χ^2	Función chi cuadrado
%	Porcentaje, en ANSI SQL carácter de búsqueda
\$	Unidad de dólar
GB	Unidad de gigabytes
MB	Unidad de megabytes
Q	Unidad de quetzales
TB	Unidad de terabytes
σ^2	Varianza

GLOSARIO

<i>Backup</i>	Archivo de respaldo, generado por un sistema gestor, que es específicamente para gestores de bases de datos que puede ser tomado en línea o fuera de línea.
<i>Backup diferencial</i>	Es un tipo de archivo de resguardo que toma los cambios realizados desde el último punto de escritura de <i>log</i> con <i>backup</i> confirmado hasta la última marca confirmada.
<i>Backup full</i>	Archivo de resguardo que para los sistemas gestores de base de datos toma todas las transacciones totales hechas en el sistema para hacer un restablecimiento total de la base de datos.
<i>Backup incremental</i>	Es un tipo de archivo de resguardo que toma los cambios desde el último punto de resguardo incremental o total hasta la última marca.
BCP	Plan de continuidad de negocio, contiene los procesos de ejecución que comprende la institución o empresa en su totalidad, de manera que puede comprenderse la ruta crítica de los procesos que son fundamentales para la empresa.

Clúster	Es un arreglo de sistemas operativos en ambiente virtual que trabajan en conjunto compartiendo recursos.
CPU	Unidad central de procesamiento.
DDL	Lenguaje de definición que encapsula las transacciones en lenguaje ANSI-SQL que sirven para la creación de objetos contenedores sobre los cuales se escribirán los datos a consultar.
DML	Lenguaje de manipulación que encapsula las transacciones en lenguaje ANSI-SQL que sirven para realizar consultas sobre las tóuplas para obtener resultados de datos.
DRP	Plan de recuperación de desastres, comprende las actividades a realizarse de manera multidisciplinaria para llevar a cabo la recuperación total o parcial de los servicios que presta una institución.
Failover	Es un cambio de nodo dentro de una configuración de clúster, en donde todos los recursos son cambiados del nodo activo al pasivo.
NIC	<i>Network interface card</i> , o interface de tarjeta de red <i>hardware</i> de comunicación a través de protocolo Ethernet.

<i>On-premise</i>	Configuración local en sitio para un servicio o computador que se encuentra establecido y configurado en una instalación de manera local.
PaaS	Plataforma como servicio, es una clasificación de un tipo de servicio en la nube en el que se presta una plataforma para ser entregada como uso de un servicio, por ejemplo, base de datos como servicio es una subcategoría del mismo, así como las interfaces de desarrollo como servicio.
RPO	<i>Recovery point objective</i> , es el punto de recuperación objetivo, este refiere al punto de los datos en el que el servicio debe ser alcanzado.
RTO	<i>Recovery time objective</i> , es el punto objetivo de tiempo de recuperación, que determina el tiempo aceptable para la organización para recuperar el servicio.
<i>Run book</i>	Es un libro de procedimientos que detalla cada paso de las actividades realizadas por el personal, de manera que pueda reproducirse la tarea y exista documentación.
<i>Switch over</i>	Es un intercambio de recursos de modo esclavo y nodo primario.

RESUMEN

Con base en las necesidades del Departamento de Informática de la Facultad de Humanidades, y con la finalidad de mejorar la eficiencia de los servicios de la plataforma de datos para el apoyo de los procesos facultativos en línea, los cuales permitirá mejorar la realización de actividades de manera que mejore la comunicación entre estudiantes, catedráticos y auxiliares para la complementación de los cursos impartidos, para el manejo de los servicios virtuales que se provee.

Dentro de los contenidos se puede encontrar la información teórica, el esquema de solución y la metodología de la investigación que soporta el desarrollo del trabajo de tesis que permitirán evaluar la herramienta de predicción del desempeño de los recursos basado en la plataforma de datos del sistema de control académico de la Facultad de Humanidades de la Universidad de San Carlos de Guatemala.

La herramienta de predicción del desempeño de la plataforma de datos, la cual es un insumo construido para monitorear y evaluar las técnicas de ajustes e implementación de tecnologías de alta disponibilidad en la plataforma de datos de la Facultad de Humanidades, la cual permitirá cuantificar el impacto del cambio en la infraestructura para el mejoramiento de los servicios facultativos en línea.

1. INTRODUCCIÓN

La Facultad de Humanidades de la Universidad de San Carlos de Guatemala cuenta con varios servicios en línea que presta a los estudiantes, cuya infraestructura en su gran mayoría se encuentra configurada en el campus central de la Universidad. Los servicios se han visto impactados fuertemente debido a la fuerte demanda durante el tiempo de calamidad pública, siendo la infraestructura actual insuficiente para proveer el servicio en momentos de alta demanda.

El presente trabajo describe la implementación de una herramienta que provea información oportuna y predictiva en el uso de los recursos en el ambiente local de base de datos, para ser de uso eficiente a lo largo del proyecto de la resolución de la problemática en el que se efectúe el ajuste de la configuración, la escalabilidad de recursos de infraestructura, en la implementación de alta disponibilidad, e integración de replicación.

El siguiente trabajo se desarrollará describiendo en el capítulo 1, los antecedentes de la investigación que tratarán sobre los estudios previos que cubren los temas a tratar sobre en este trabajo.

El capítulo 2, describirá el planteamiento de la justificación que soporta las razones que validan la realización del estudio y el objeto del problema de investigación cubierto en el este trabajo. El capítulo 3, abordará los alcances que comprende el universo del estudio, describiendo los resultados, técnicos e investigativos del mismo.

En el capítulo 4, se detallará el marco teórico donde describe las teorías utilizadas para la implementación del proyecto, en el que describe las técnicas que soportan la creación y la configuración utilizada para la implementación de la solución.

El capítulo 5, tratará sobre la presentación de resultados, que comprende la implementación de alta disponibilidad en la configuración y migración del servicio de base de datos hacia los servidores de *failover cluster*. En el mismo capítulo se describirá la configuración de la nube híbrida, al implementar la replicación haciendo uso de plataforma como servicio y la integración con los servidores locales.

Asimismo, tratará el contenido del desarrollo e implementación de la herramienta de predicción y monitoreo de recursos de la base de datos en *Mysql*, donde se describirá el funcionamiento interno, los diagramas de datos y configuración que detallará el diseño de la solución de arquitectura. Dicha herramienta descrita será utilizada como insumo durante el proyecto de configuración de alta disponibilidad y adaptación de nube híbrida en la facultad de humanidades.

El capítulo 6, muestra la discusión de resultados, el cual buscará demostrar los datos obtenidos durante la implantación de nube híbrida para describir el comportamiento de la mejora de la fiabilidad en la plataforma de datos producto de la implementación de técnicas de alta disponibilidad y redundancia de datos, cuya obtención de datos será realizada a través de la herramienta de predicción de recursos diseñada y creada para dicho proyecto.

2. ANTECEDENTES

Los sistemas gestores de bases de datos han evolucionado convirtiéndose en parte vital de las arquitecturas de los sistemas de información. La búsqueda de información coherente y oportuna en todo momento se ha convertido la misión de toda plataforma. Las condiciones actuales orientan a diseñar arquitecturas de datos con capacidades de alta disponibilidad y desempeño que permitan al usuario tener el servicio de datos en todo momento.

Existen diversas teorías que demuestran la importancia de dichas técnicas y las ventajas que brindan dentro de la organización al ser implementadas conforme a las necesidades y conflictos que existan dentro de la misma. El presupuesto y las condiciones de infraestructura definen el camino a seguir y la selección de los recursos de arquitectura para lograr la implementación.

El concepto alta disponibilidad se aborda en distintos aspectos para ser logrado, según el capítulo titulado *high availability clustering and its architecture*, indica que existen tres tipos de configuración para el manejo de los recursos o nodos para trabajar en conjunto, con el fin de lograr alto desempeño, balanceo de carga o alta disponibilidad, asimismo menciona el autor, debe seleccionarse el ideal según los requerimientos, mencionando el típico caso de uso para alta disponibilidad, y los diferentes componentes para su implementación en ambientes Linux.

El objetivo de los clústeres de alta disponibilidad es para asegurarse que los recursos críticos logren la disponibilidad máxima posible. Este objetivo es cumplido al instalar software de clúster en múltiples servidores. Este software

monitorea la disponibilidad de los nodos del clúster, y monitorea que la disponibilidad de los servicios o recursos que son manejados por el clúster. Si un servidor se detiene o decae, la alta disponibilidad del clúster notará y se asegurará que el recurso sea reiniciado en otro lugar del clúster, de esta manera pueda ser utilizado nuevamente con una interrupción mínima (Vugt, 2014).

Otro punto importante a describir, es que la alta disponibilidad es un concepto ideal, sin embargo, no es posible mantener la disponibilidad, la consistencia y la tolerancia de particiones. Debe sacrificarse uno de estos atributos, esto es descrito por IBM en su publicación: *What is the CAP theorem?*, El teorema de CAP se aplica de la misma lógica similar, sabiendo que en un sistema distribuido puede únicamente proveer solo dos de tres características deseadas siendo estas: consistencia, disponibilidad, y tolerancia de particiones identificado por sus siglas en inglés (CAP). Una red de sistemas distribuidos que almacena datos en uno o más de un nodo (en máquinas físicas o virtuales) al mismo tiempo. Ya que todas las aplicaciones en la nube son sistemas distribuidas, es esencial entender el teorema de CAP cuando se diseña una aplicación de manera que pueda escoger el manejo de los sistemas de datos (Brewer, 2000).

Según mencionó *extending Mysql high availability*, se amplía el concepto de replicación en el motor de base de datos *Mysql*, en el que se menciona que el componente del servidor responsable para obtener alta disponibilidad es la replicación. La replicación en *Mysql* es la habilidad de duplicar cambios en los datos que ocurren en un servidor hacia otro. En lugar de copiar los datos directamente (lo cual podría ser lento y complicado cuando se trata de realizar transacciones), los cambios son transferidos en la forma de comandos de metadato, por tanto, eventos, que son copiados a un segundo servidor son

ejecutados allí esto provee la habilidad de duplicar exactamente los cambios en los datos del servidor original (Bell, 2012).

Al tratar sobre alta disponibilidad, el concepto del manejo de recuperación de desastres y un plan efectivo desde el punto de vista de las bases de datos se convierte en necesario ya que es parte del plan continuidad de negocio y de recuperación desastres. En el libro titulado *healthy SQL* hace las siguientes incógnitas: si ocurre un desastre, ya sea que el evento sea natural, técnico o hecho por el hombre, ¿cómo responderá y cómo se recuperará su organización?, ¿puede garantizar la continuidad del negocio ante un desastre? la alta disponibilidad y la recuperación ante desastres (*HA/DR*) abordan estas preocupaciones. Se discute sobre la estrategia, también sobre las características y la tecnología en torno al motor de base de datos para habilitar *HA/DR*. Como parte de lograr un motor de base de datos saludable, debe prepararse para lo peor. Un buen plan de recuperación ante desastres es su seguro (Pearl, 2015).

Con respecto a los métodos de resguardo de las bases de datos como parte de la estrategia se menciona. Existen algunos métodos típicos que las empresas utilizan para hacer una copia de seguridad de su infraestructura de base de datos. Usando el motor de respaldo nativo del motor, las bases de datos se respaldan en el sistema de archivos. Con este método, las copias de seguridad se volcarían localmente o al almacenamiento de red adjunto. La solución de copia de seguridad de red puede hacer una copia de seguridad de todos los archivos del servidor de archivos, además de los archivos de copia de seguridad de la base de datos, asegurando que los datos se conservan en caso de falla (Pearl, 2015).

La continuidad del negocio, como parte de los procesos a seguir dentro de la institución se convierte en fundamental para lograr la alta disponibilidad. Este concepto incluye los procedimientos y procesos promulgados para garantizar la

continuidad del negocio. A menudo se usa solo para cubrir lo que sucede en caso de desastre, pero un desastre no es lo único que afecta la continuidad del negocio. La continuidad de negocio debe cubrir cualquier evento que pueda impedir o detener los flujos de trabajo comerciales de sus sistemas. Todos los aspectos de la continuidad del negocio deben documentarse e incluirse en el libro de ejecución (Pearl, 2015).

Para garantizar la continuidad del negocio, debe llevarse a cabo una cierta cantidad de planificación que normalmente tendrá en cuenta cómo el negocio recuperará sus operaciones o trasladará las operaciones a otra ubicación esto se conoce como planificación de continuidad del negocio (BCP). Para diseñar un plan integral para la continuidad del negocio, los equipos internos responsables de su unidad de negocio deben evaluar y considerar los elementos del BCP el ciclo de vida, que consta de cinco fases: análisis, diseño, implementación, prueba y mantenimiento (Pearl, 2015).

Como parte del diseño de arquitectura que provea los atributos mencionados, el entorno de infraestructura nube híbrida se hace viable, por lo que es importante conocer el costo de su utilización y los métodos para lograrlo, el artículo científico *Costs of using hybrid cloud infrastructure: towards a general framework*, versa sobre las consideraciones en el manejo de costos.

Uno de los factores importantes que afectan las decisiones de las organizaciones de adoptar una nube híbrida es el costo total de adquirir y administrar la infraestructura. En un marco general para la evaluación de costos de infraestructura en la nube, en el que, para varios tipos de recursos de infraestructura en la nube, se consideran los componentes de costos asociados y los factores que determinan estos componentes (Mazhelis, 2012).

Los antecedentes previamente descritos son una descripción breve sobre los puntos a tratar en el presente documento, la finalidad es abordar las herramientas que nos permiten resolver de manera conceptual el soporte del desarrollo de la infraestructura de plataforma de la datos para lograr, alta disponibilidad, alto desempeño e integración hacia una plataforma de nube, convirtiéndose en una nube híbrida; a la vez diseñar los procesos de automatización de resguardo y recuperación para la continuidad de la organización, definiendo buenas practicas en los procesos humanos e integración de un plan para recuperación de desastres.

El contenido anteriormente descrito es la base de las teorías y técnicas a implementarse durante la ejecución del proyecto en el que se describe la utilización que comprende tecnologías de alta disponibilidad en failover cluster, replicación y utilización de plataforma como servicio dedicado para base de datos y el ajuste del mismo.

El siguiente detalle es abordado en el marco teórico del presente documento.

3. PLANTEAMIENTO DEL PROBLEMA

La Facultad de Humanidades de la Universidad de San Carlos de Guatemala cuenta con varios servicios en línea que presta a los estudiantes cuya infraestructura está configurada en el campus central de la Universidad y cuentan con servicios externos de aprovisionamiento de infraestructura en la nube para albergar parte de la infraestructura que poseen. Los servicios en línea han mostrado un fuerte degrado de desempeño afectando los procesos del personal facultativo y académico.

Los principales servicios en línea que presta la Facultad de Humanidades son: el sistema de educación virtual y el sistema de control académico, ambos han sido impactados fuertemente debido al incremento en la demanda durante el tiempo de calamidad pública, ya que tanto personal facultativo, docentes y estudiantes han utilizado exclusivamente los recursos en línea para las actividades académicas, de cátedra y administrativas.

Durante el tiempo de trabajo a distancia, las conexiones a los sistemas se han incrementado principalmente durante los fines de semana; siendo este tiempo el de mayor demanda por los usuarios. Se han tenido hasta 600 usuarios conectados en el sistema en menos de 3 horas, este punto se ha determinado que es el punto crítico para los recursos, ya que, al llegar a este umbral, los recursos del sistema llegan a utilizar mas de 85 % de los núcleos de CPU y memoria.

El sistema gestor de base de datos está configurado para permitir como máximo 600 conexiones y 8 GB de memoria compartida como limite, por lo que

al llegar a dicho número, impide nuevas conexiones de usuarios. El manejo del cache en tablas llega a más de 2000 objetos abiertos en menos de una hora teniendo un hit ratio de el 2 %, el 100 % del cache en uso, las tablas temporales y el buffer de operaciones en índices llegan a menos del 5 % libres por lo que crean cuellos de botella.

Debido a esta situación ha generado problemas serios en el tiempo de respuesta de ambos servicios. Como resultado de lo anterior el personal administrativo y educativo se ha visto afectado al retrasar y prolongar sus actividades con más horas de servicio para cubrir con las eventualidades del sistema.

En cuanto a la disponibilidad del servicio, se han reportado momentos de pérdida de servicio parcial y total, asimismo se ha registrado incidentes de caída de la base de datos en la que se ha requerido de técnicas de recuperación y que dicha actividad ha significado la pérdida de datos; por lo anterior planteado se encuentra la necesidad de verificar, ajustar y mejorar la configuración actual del recurso de la base de datos con el fin de orientar hacia el diseño de una plataforma de datos robusta, actualizada y moderna.

Durante el proceso de mejoramiento del desempeño del sistema de base de datos en sitio, es necesaria la evaluación del uso de los recursos, verificación de alertas, así como la capacidad de hacer estimaciones futuras para el aprovisionamiento de recursos de manera oportuna. Para lograr dicha meta una herramienta con características de predicción de recursos será desarrollada con el objeto de garantizar el aseguramiento del uso óptimo de los recursos para el ajuste, e implementación de la nueva infraestructura de plataforma de datos.

Como parte del mejoramiento de la continuidad de negocio, y derivado a la problemática de incidencias reportadas, se plantea la necesidad de implementar métodos de monitoreo automatizado en el sistema de base de datos (*on-premise*), a nivel de sistema operativo y del sistema gestor de datos. La Facultad de Humanidades cuenta con una incipiente implementación de infraestructura en la nube en donde el sistema está configurado en instalaciones del campus y una parte se encuentra desplegada utilizando un proveedor de servicios en la nube (*Customer Service Provider*).

La configuración actual en sitio no posee suficientes recursos para lograr la consulta y manejo de los datos de manera eficiente, por que es necesario realizar un análisis de brechas, con el objetivo de verificar, proponer y mejorar la infraestructura física; para lograr la realización de un escalamiento en los recursos asociados a la base de datos. Dicho escalamiento involucra la implementación de alta disponibilidad y redundancia de datos utilizando el *failover clústeres* y replicación.

En cuanto a la capa de datos que esta desplegada en infraestructura en la nube, se considera que actualmente dicho servidor únicamente hace uso de la capa de virtualización y que no aprovecha aún los recursos que provee la plataforma que pueden ser utilizados para el mejoramiento de la configuración de los recursos.

El Departamento de Informática cuenta con un proveedor externo local, el cual administra la plataforma de servicios en la nube (*AWS*); el departamento actualmente desaprovecha los recursos de infraestructura que involucra el servicio por lo que busca analizar, evaluar y ajustar la configuración, por lo que se hace necesario medir el uso eficiente de los recursos que permita la minimización de costos y garantizar el desempeño.

4. JUSTIFICACIÓN

El presente trabajo busca desarrollar sobre la línea de investigación enfocada a Tecnologías de la información y la comunicación para apoyo a la educación, para el fortalecimiento de la plataforma de datos de la Facultad de Humanidades en el sistema de control académico, mediante la implementación de técnicas de alta disponibilidad, redundancia las cuales debido a la situación de calamidad pública actual que aqueja el país, y considerando que el servicio educativo público que provee la Facultad de Humanidades de la Universidad de San Carlos debe continuar; surge la necesidad del mejoramiento de la plataforma de datos para garantizar la prestación de los servicios facultativos como parte indispensable de la continuidad de la academia.

En vista de las eventualidades como la degradación y disrupción en la prestación del servicio, el apoyo a la mejora de la plataforma de datos tendrá un impacto positivo directo en la calidad de entrega del servicio para la cátedra y actividades facultativas.

El Departamento de Informática de la Facultad de Humanidades de la Universidad de San Carlos de Guatemala será provisto con el mejoramiento de la infraestructura al integrar alta disponibilidad y redundancia de datos en la plataforma de datos para el sistema de control académico, así también con herramientas de monitoreo en tiempo real de los eventos a nivel de las capas de sistema operativo y plataforma de datos y de esta manera tener un medio de reacción inmediata y continua para el manejo de incidencias.

Dentro del manejo para el aseguramiento y resguardo de los datos, es la búsqueda de este trabajo implementar de un sistema de resguardo y transporte mediante el cual permita asegurar las copias de archivos en un destino provisto que permita la recuperación total y parcial de los datos hechos de manera continua e instantánea de manera que existan puntos de control de tiempo para recuperación de datos dada la importancia de los mismos. Con la existencia de métodos de resguardo continuos en lo plataforma de datos, cabe mencionar que abre la posibilidad para mejoras en el proceso de desarrollo y pruebas, ya que esto permitirá una forma eficiente para el manejo y transporte de copias que serán de utilidad para los ambientes de desarrollo, el cual es un punto focal para lograr la automatización en la construcción de escenarios para la implementación de codificación y pruebas en ambientes controlados.

Las condiciones actuales de infraestructura *on-premise* del sistema gestor de base de datos del sistema de control académico de la Facultad de Humanidades, requiere una actualización para convertirse en una arquitectura de plataforma de datos que se adapte a las condiciones demandadas por la nueva tecnología (dispositivos inteligentes), por lo cual la transformación hacia un diseño de nube híbrida cobra relevancia.

Por lo anterior mencionado se hace necesario a su vez definir y documentar los procesos para el reforzamiento en el manejo de recuperación de desastres, así como el del plan para la continuidad de las operaciones enfocadas a la plataforma de datos de la Facultad de Humanidades de la Universidad de San Carlos, para el uso de las operaciones del Departamento de Informática.

5. OBJETIVOS

5.1. General

Implementar un sistema de monitoreo, evaluación y predicción de recursos para la plataforma de datos *on-premise* como herramienta para el oportuno mejoramiento del desempeño de los recursos de infraestructura e integración de nube híbrida.

5.2. Específicos

- Implementar técnicas de alta disponibilidad y redundancia de datos a través de *failover clustering on-premise* y replicación circular hacia el recurso en la nube en la base de datos Mysql.
- Diseñar, Implementar un sistema de monitoreo, evaluación y predicción de recursos para la plataforma de datos *on-premise* para asesorar el mejoramiento del desempeño durante la implementación de mejoras a nivel de base de datos.
- Verificar y establecer políticas de (DRP) *disaster recovery plan* así como (BCP) *business continuity plan* concerniente a la capa nivel de base de datos, en las cuales establezca la administración de versiones de base de datos, manejo de parcheo, estrategia, políticas de *backup* y recuperación.

6. NECESIDADES POR CUBRIR Y ESQUEMA DE SOLUCIÓN

6.1. Necesidades por cubrir

En vista de la oportunidad planteada por la Facultad de Humanidades, este proyecto busca enfocarse en el mejoramiento del desempeño de la arquitectura de la plataforma de datos, específicamente en el sistema de control académico de la Facultad de Humanidades, considerando que este afecta tanto a estudiantes como al personal facultativo siendo este el eje central de la Facultad.

La búsqueda consiste en mejorar el desempeño del uso del sistema de control académico de la Facultad de Humanidades de la Universidad de San Carlos, enfocándose en implementar soluciones de alta disponibilidad y redundancia de datos a la configuración (*on-premise*) o en sitio en primera instancia. Dado que la base de datos central reside en servidores en sitio en instalaciones del campus central, y cuya configuración actualmente consiste en servidores virtualizados sobre sistema operativo Linux Centos 7 y Mysql 5.2 en modo standalone; es necesario instalar y configurar nuevos servidores que conformarán el arreglo de los recursos de *clúster*.

Para lograr este objetivo se desprende del problema el conocer el ajuste de las condiciones de los recursos actuales, así como la configuración apropiada para la asignación e instalación de la los recursos del nuevo *clúster*. Se requiere realizar una investigación y análisis de la carga de trabajo la cual involucra el monitoreo constante durante todas las fases de implementación y posteriores a ellas para las capas de sistema operativo y base de datos.

Para ello, el desarrollo de un sistema de monitoreo, evaluación y predicción de los recursos de la plataforma de datos *on-premise* como herramienta durante la ejecución del proyecto toma importancia, por lo que es como objetivo de este proyecto el desarrollar dicho sistema el cual será probado durante la ejecución del mismo, el cual a la vez servirá de herramienta para el futuro control del recurso en el Departamento de Informática de la Facultad de Humanidades.

En dicho análisis preliminar a la implementación permitirá dos objetivos:

- La preparación de la configuración de los recursos de clúster.
- La preparación de las recomendaciones de ajuste de mejores prácticas en la implementación de consultas y a nivel de la parametrización y configuración del motor de base de datos.

Una vez cubierto este proceso se requerirá realizar la migración, la cual involucra actividades de verificación previa de la copia y restauración de la base de datos y los objetos que la componen como seguridad y objetos de programación, así también de todos los casos de prueba funcionales de la aplicación.

Una vez cubierto este proceso se requerirá realizar la migración, la cual involucra actividades de verificación previa de la copia y restauración de la base de datos y los objetos que la componen como seguridad y objetos de programación, así también de todos los casos de prueba funcionales de la aplicación.

6.2. Esquema de la solución

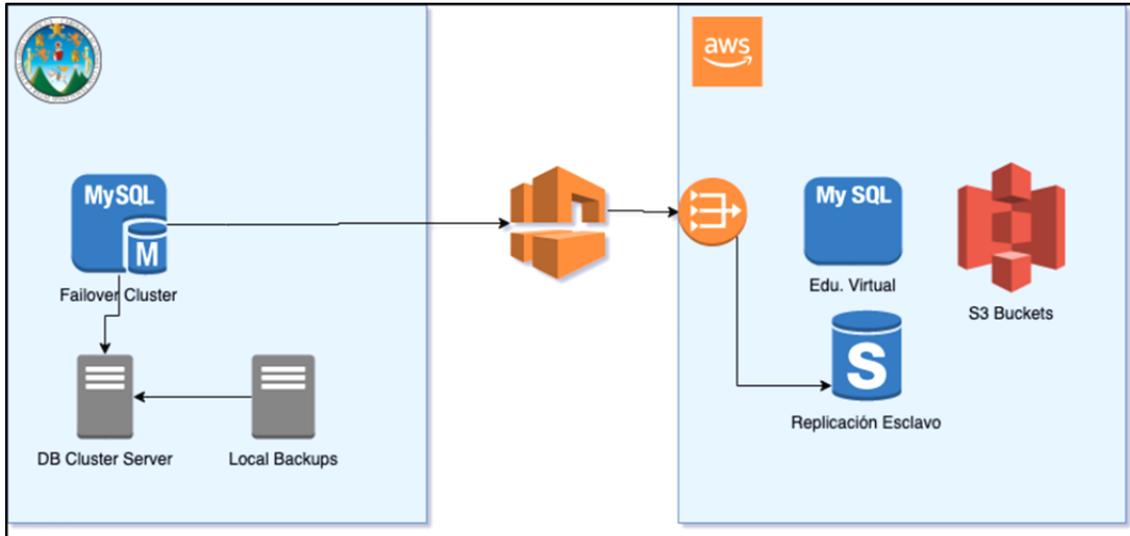
Con respecto a la configuración en la nube, la Facultad de Humanidades actualmente alberga un servidor virtualizado en *EC2* en la plataforma de *AWS*, sin embargo, este es utilizado únicamente para manejar una instancia stand-alone de los servicios de la aplicación de cursos en línea, lo cual no representa el aprovechamiento de los recursos provistos por el servicio.

Al lograr una implementación exitosa del nuevo ambiente de clúster, se procedería a implementar la integración con la infraestructura en la nube y aprovechar el uso de los recursos en la nube para configurar una replicación y tener una redundancia de datos y así lograr implementar características de configuración de nube híbrida.

Durante la ejecución de la implementación de la evaluación de los recursos, se realizará un monitoreo previo, se ajustará la configuración con los recursos existentes y se aprovisionará la configuración hacia el nuevo arreglo de servidor, en la segunda fase, al concluirse con éxito la implementación del clúster todo esto se hará a través del sistema de monitoreo y predicción de recursos que se llevará a cabo en este proyecto.

Dicha información se medirá en comparación con la integración de la replicación cíclica en *AWS RDS* la cual se evaluará y monitoreará la implementación con la nueva configuración de *clúster* existente *on-premise*.

Figura 1. **Plataforma de datos actual**

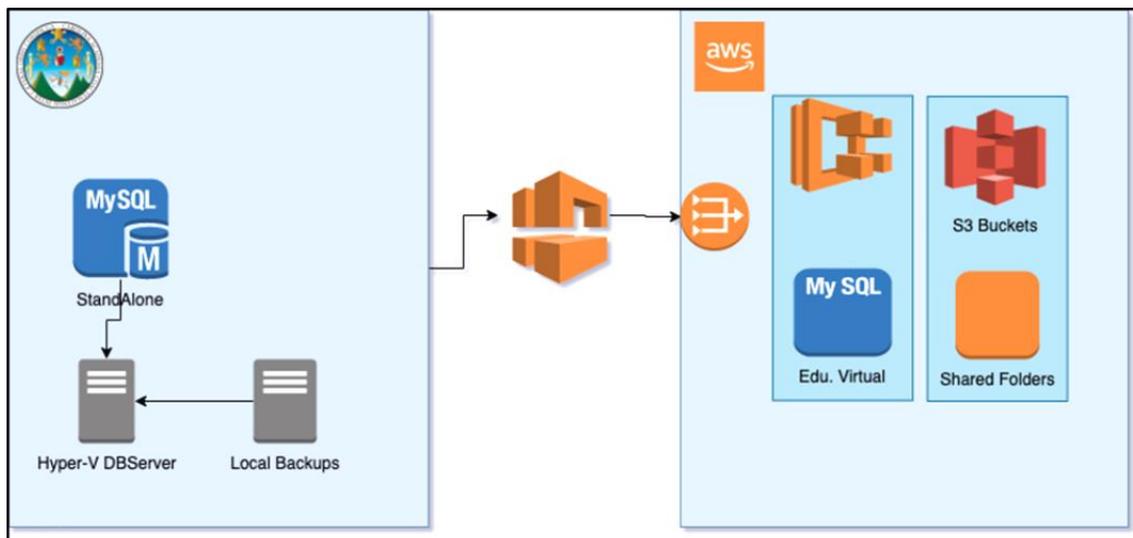


Fuente: elaboración propia.

En la figura 1, observamos la actual configuración de los servidores en la que nos encontramos con un servidor stand alone en el ambiente productivo, el cual se encuentra en el ambiente *on-premise* y no cuenta con respaldo de ninguna naturaleza.

Por otra parte la configuración AWS cuenta con la base de datos de educación virtual que se encuentra en el la configuración EC2, y que a su vez no cuenta con respaldo.

Figura 2. **Plataforma de datos híbrida**



Fuente: elaboración propia.

En la figura 2, observamos el esquemático que describe la solución al detalle de las necesidades a cubrir para mejorar el ambiente actual en donde vemos la integración entre el ambiente de *failover clúster* en la configuración de maestro-esclavo.

Asimismo, la integración del ambiente *on-premise* hacia los recursos AWS, el cual implica el crecimiento de recursos al integrar el modelo de replicación y que asimismo cuenta con una instancia esclavo cuyo propósito es ser testigo en las operaciones de replicación.

7. MARCO TEÓRICO

7.1. Configuración alta disponibilidad en la plataforma de datos

La creación de un clúster sirve para satisfacer necesidades de alta disponibilidad mediante la configuración de servicios de software de diferentes capas que trabajan entre sí. El componente clave para la gestión de servicio de alta disponibilidad en la adición de alta disponibilidad, implementa conmutación en frío para aplicaciones fuera de la plataforma este puede presentarse conmutación de servicio de alta disponibilidad, si se presenta error en un nodo de clúster o si el administrador de sistema de clúster traslada el servicio de un nodo del clúster a otro (por ejemplo, para un corte de energía planeado de un nodo del clúster).

7.1.1. Definición de alta disponibilidad

La búsqueda de la alta disponibilidad a nivel de infraestructura es la maximización de la disponibilidad de los recursos de infraestructura. Esta búsqueda es mantener el recurso prestando el servicio sin interrupción. En la necesidad de construir infraestructura con redundancia del servicio para mantener el recurso el mayor tiempo posible, a esto le conocemos como alta disponibilidad.

El concepto se asocia a la consideración con el sinónimo de confiabilidad, cuya búsqueda es hacer una solución accesible y a la vez tolerante a fallos, de manera planificada o no durante un periodo de tiempo. Un factor clave es establecer la robustez sin embargo lo importante es la calidad de esta robustez.

Particularmente en el producto Mysql se ha visto beneficiando en el sentido que Oracle ha invertido de manera de aplicar características en el producto para el producto de software libre. Para lograr dicha característica, se requiere un número superior a dos computadoras o nodos que compartan recursos de almacenamiento, red y servicios, en los que existe un nodo que trabajan en conjunto para presentar el recurso del servicio que prestan (Bell, 2018).

Esencialmente la búsqueda de esta técnica es lograr de manera confiable el uso del recurso, así lograr la mayor cantidad de tiempo, el tiempo de funcionamiento.

Para lograr estos principios debe seguirse los siguientes pasos:

- Elimine un único punto de falla en el diseño de la solución de arquitectura del servicio prestado.
- Agregue un punto de recuperación o redundancia de datos a través de la creación de múltiples particiones, las cuales pueden ser habilitadas a través de mecanismos de redundancia.
- Implemente tolerancia a fallos, a través de la implementación de mecanismos de detección de fallos y recuperación automática.

7.1.2. Arquitectura de clúster en Linux

La implementación del clúster en alta disponibilidad en Linux involucra varios elementos o recursos de clúster cooperando entre sí para presentarse como una unidad para el servicio prestado.

Los recursos necesarios para conformar el clúster son servidores o nodos, el almacenamiento compartido sobre una distribución *SAN* a través de particiones de datos lógicas o unidades lógicas de almacenamiento los cuales comparten una *IP* y un nombre el cual es manejado a través de un balanceador de carga o un monitoreo de servicio o *heartbeat* (Vugt, 2014).

7.1.3. Tipos de clústeres

Existen tres tipos de *clúster* según su función que son distinguibles al configurar en un ambiente Linux, los cuales son descritos de la siguiente manera:

- *Clúster* de alto desempeño: está diseñado para ambientes en el que varias computadoras trabajan en conjunto para albergar uno o más servicios de presentación como uno solo.
- *Clúster* de balanceo de carga: son utilizados para servicios *front-end* o de primera línea, en el que recibe la carga de los requerimientos de usuario, los cuales procesan las peticiones distribuidas en los diferentes nodos según se refiera el manejo de los recursos.
- *Clúster* de alta disponibilidad: Múltiples computadoras nodos trabajando en conjunto para mantener los recursos del servicio presentable con el menor tiempo de caída posible. Los diferentes tipos de configuración de alta disponibilidad puede ser en modo, activo-activo o activo-pasivo. Ambos modos significan la manera en que ambos recursos pueden existir para presentar el recurso. En el modo activo-activo, ambos servidores presentan o ejecutan simultáneamente los recursos, en comparación con activo-pasivo en el que únicamente uno de los recursos presenta y los demás son esclavos en espera de una caída (Vugt, 2014).

7.1.4. Recursos de clúster

Los elementos necesarios para conformar un recurso de *clúster*, como se mencionó con anterioridad de manera en que los recursos de sistema operativo actúan como uno solo en el que previamente se configuran varias máquinas sobre virtualización en el que se ejecutará los diferentes componentes de software que harán posible la comunicación para llevar a cabo el funcionamiento.

- Pacemaker, heartbeat 2.0 y RedHat cluster suite
- Almacenamiento compartido
- *NICS* asociadas
- Direcciones fijas de IP
- *Multipathing*
- *Fencing/dispositivos stonith*

El almacenamiento compartido está constituido por una configuración de sistema *clúster*, en ambos nodos comparten por medio de red un recurso de área de almacenamiento, dicho recurso es disponible y utilizable por ambos. La presentación de la red de área de almacenamiento hacia los nodos es una capa de datos es dada a través de direcciones o extensiones lógicas, las que cuales pertenecen a un arreglo de disco, a las que a nivel de la capa de sistema operativo son manejadas a través de un formato de tabla de particiones lógicas y a su vez un sistema de archivos. Dichos sistemas de archivos son manejados a través de puntos de montaje los cuales son visibles para ambos nodos de la configuración, sobre los que se encuentran alojados los archivos de datos y logs de la instancia(s) del sistema manejador de base de datos (Bell, 2016).

- Configuración de red asociada: para lograr integrar el tráfico de red asociado al clúster es necesario tener 2 direcciones *IP* asignadas a cada

nodo, una asociada a la interface de red física, la cual pertenece el servidor y una secundaria que será designada para el nombre del clúster asignado como conciliador o *pacemaker*, quien hará la conexión al nodo activo y manejará las transacciones entre los usuarios y el servicio.

- *Fencing* y *quorum*: para lograr evitar escenarios en donde parte de los recursos que conforman al clúster se asignen de manera dividida en un nodo u otro que pueden provocar corrupción de datos e inconsistencia en los mismos, se utiliza una técnica denominada *fencing*, que consiste en la aplicación de un *quorum* que significa crear un voto de mayoría, en donde el protocolo determina que si la mayoría de recursos que componen el *clúster* se encuentran disponibles en un mismo lugar físico, puede tomarse la acción de escritura o cambio requerido (Vugt, 2014).

7.1.5. Arquitectura de replicación de datos en Mysql

Es la habilidad de duplicar los cambios en los datos que puedan ocurrir de un servidor a otro. En vez de que los cambios sucedan en un único lugar, las operaciones son recibidas o centralizadas en el sistema gestor de base de datos primario disponible, en el que procesa la transacción en su *log* de operaciones y es aceptado en el estado de escritura, en este punto existe un proceso en el que un generador realiza una marca diferencial de todos los cambios escritos desde el último punto de escritura, el cual es transportado hacia el nodo secundario que se encuentra en modo de recuperación, lo cual indica que no está disponible o activo para recibir escrituras por parte del usuario, ya que se encuentra en un modo de espera y recuperación.

El proceso de generación, transporte y aplicación de los cambios es una configuración definida que determina el estado del proceso de replicación.

La escritura de los eventos que se aplicarán es almacenada en un archivo secuencial denominado *log* binario, y a la lectura y aplicación del archivo en el nodo secundario es denominado como *relay log*.

Debido a que este proceso es una captura y copia que logra replicar las transacciones hechas en el nodo primario, es que se conoce al proceso como replicación (Bell, 2012).

7.1.6. Descripción de componentes

La topología que compone a nivel de instancias de sistemas manejadores de base de datos según el rol que desempeñan, se componen de:

- **Nodo maestro:** este servidor es el que funge como el nodo primario, que en un determinado punto acepta y aplica las transacciones DML y DDL.
- **Nodo esclavo:** este servidor es el que mantiene una copia de todos los datos y copia o replica todas las transacciones que el nodo maestro realiza.
- **Nodo *relay* esclavo:** este servidor es capaz de ser esclavo y a su vez es maestro de uno o más esclavos, de esta manera la replicación puede ser reproducida en varios servidores copia. Como se mencionó con anterioridad los servidores esclavos solo pueden ser modo lectura, es decir no pueden aceptar transacciones de usuarios para escritura (Bell, 2012).

7.2. Teorema de CAP

El teorema de CAP aplica la misma lógica similar a la lógica que aplica en los sistemas distribuidos. El nombre es una composición de tres iniciales de tres atributos de fiabilidad alcanzables, como lo son la consistencia, la disponibilidad y la tolerancia de la partición. Un sistema distribuido es un conjunto de recursos de infraestructura (sistemas operativos, máquinas virtuales o físicas y de almacenamiento) que trabajan entre sí para conformar un sistema que se presenta como un todo, capaz de recuperarse o ejecutar de mejor manera para ser más eficiente en su proceso es entonces donde juega un papel importante los atributos de fiabilidad, ya que estos son los que tiene el objeto de cumplir.

El teorema de *CAP* también conocido como el teorema de Brewer, bautizado en nombre del profesor Eric A. Brewer, dicta que como regla fundamental un sistema distribuido trata de alcanzar o cumplir con los tres atributos mencionados, sin embargo, solo puede cumplir dos. Es entonces que el criterio del diseño de arquitectura entra en juego, ya que debe escogerse el producto de tecnología capaz de cumplir con los requerimientos del ambiente.

Así pues, los productos de tecnología se segmentan o catalogan según los atributos que cumplen, con base a este teorema, cual es el enfoque cual es el par de características que definen al producto de tecnología a evaluar (Brewer, 2000).

7.2.1. Concurrencia

Refiere al atributo de calidad en el que todos los datos pueden consultar los datos al mismo tiempo no importando en que nodo estén conectados. Inclusive si el usuario este realizando escritura en un nodo, esta debe ser observable en

otra conexión de nodo, con el entendido que esta es replicada mediante métodos de escritura colaborativa sincronizada (Gillbert y Lynch, 2002).

7.2.2. Disponibilidad

Refiere al atributo en el que cualquier cliente que realice una consulta de data obtiene una respuesta y no se deniega el servicio, aunque un nodo o más se encuentren fuera de servicio. De tal manera que todos los nodos conectados o activos procesan las solicitudes requeridas por cualquier cliente de manera centralizada (Gillbert y Lynch, 2002).

7.2.3. Tolerancia a la partición

Una partición es una ruptura en la comunicación dentro del sistema distribuido, que produce una pérdida parcial temporal entre dos nodos de comunicación. La tolerancia significa la capacidad de seguir prestando el servicio por medio de los demás nodos activos que conforman el sistema (Gillbert y Lynch, 2002).

7.3. Plan de recuperación de desastres en la plataforma de datos

El plan de recuperación de desastres involucra la construcción de un ambiente de *stand-by* pasivo de copia continua de transacciones de datos en un sitio alternativo el cual posea las mismas características de motor de base de datos en sincronía con el ambiente productivo, al cual se le denomina comúnmente como nodo *stand-by*. La solución a implementar con dichas características debe contar con la capacidad de copia y automatización de procesos de levantado de estrategias de archivos de respaldo. El proceso de copia, transporte y aplicación de archivos de respaldo es la esencia de la sincronía y réplica de base de datos.

El proceso de recuperación puede hacerse de manera sincrónica u asincrónica dependiendo sea el caso del ambiente de configuración, los factores a considerar son la velocidad del recurso de red, el tamaño de los *archive logs*; esto determina el tiempo de aplicación y refrescamiento de los datos.

Los métodos convencionales de generación y aplicación de archivos de respaldo, los cuales son útiles para la configuración del sistema gestor de base de datos esclavo, pueden realizarse a través de generación de archivos de respaldo en frío o en caliente sin los cuales no podría ser posible la implementación de alta disponibilidad, que es esencial para un ambiente de recuperación de desastres.

Con respecto a las bases de datos, un archivo de respaldo se refiere como en caliente o dinámico, si es generado mediante el sistema gestor de base de datos en su modo activo, de manera que los recursos son accesibles mientras este archivo se genera.

El transporte y almacenamiento de los archivos de respaldo son determinantes en el proceso de recuperación de desastres, ya que determinar el medio en donde se localicen es fundamental para garantizar el éxito de un plan para recuperación de desastres. Debido a la periodicidad de los mismos es necesario definir un plan de generación de archivos de respaldo, para lo cual definir un calendario, el tipo de archivo de recuperación es esencial.

El archivo de respaldo total, es el primero y más importante ya que este contiene la copia completa de base de datos, este debe ser el primero a ser aplicado y colocado en la copia secundaria y colocar la base de datos secundaria en modo de recuperación al implementarla.

La generación periódica del respaldo total dependerá del tamaño de la base de datos, así pues, si el generarlo toma más de 3 horas como ejemplo, es de considerar implementar un segundo tipo de respaldo adicional, tal como archivos de respaldo incrementales; con este tipo de archivos de respaldo se obtienen los cambios realizados acumulativos desde el último punto de archivo de respaldo total o full realizado. Adicionalmente se consideran un tercer tipo de archivos, como lo son los de tipo diferencial, los cuales guardan una copia de los cambios realizados desde el último punto de marca del *log* de transacciones. Teniendo en consideración los tipos de archivos, puede considerarse generar, transportar y almacenar de tipo full una vez por semana, Incrementales tres veces por semana y diferenciales cada cierto número de horas al día.

La verificación de todo proceso automático es un paso importante para garantizar el éxito de las operaciones futuras, por lo que añadir pasos de verificación de integridad del sistema gestor, de los archivos de respaldo, del transporte de los mismos y de la disponibilidad de los sistemas gestores ayuda a garantizar la fiabilidad del plan (Pearl, 2015).

7.4. Plan de continuidad de negocio en la plataforma de datos

A pesar de que los conceptos de alta disponibilidad, recuperación de desastres y continuidad de negociación son ampliamente utilizados, es de importancia recalcar sus diferencias. La continuidad del negocio incluye los procedimientos y procesos para asegurar la continuidad. Este concepto debe ser asociado con el plan de desastres ya que un desastre muchas veces ocurre en ambientes no planificados en donde no se contempló la continuidad del negocio.

Para contemplar dicho proceso es importante hacer una asesoría e inventario de una línea base, de los objetos que comprenden nuestra arquitectura de datos. Este inventario en un ambiente en crecimiento muchas veces es complejo, para lo que hacemos uso de herramientas que se conecten automáticamente, con el objeto de obtener la información de la instancia de servidor de base de datos y las configuraciones de cada una.

Un punto importante dentro de la descripción es obtener el nivel de prioridad que tiene en el inventario, y el ambiente al que pertenece, por ejemplo, determinar si es de tipo crítico, importante o pruebas, y su ambiente podría ser productivo, desarrollo y pruebas. Para determinar este punto podemos cuestionar lo siguiente: ¿cuántas personas se conectan a esta base de datos?, ¿si sucede un desastre cuantos servicios dependen de este?, ¿inclusive si es un ambiente de desarrollo, ¿cuánto cuesta el tiempo que el servicio este fuera de línea?

El siguiente es un ejemplo de una línea base de inventario que muestra gráficamente el escenario y sus prioridades.

Tabla I. **Inventario de activos de base de datos**

Business critical	Microsoft SQL Server 2008 R2	Production
Department Important	Microsoft SQL Server 2008	Test
Department Important	Microsoft SQL Server 2008 R2	Test
Business critical	Microsoft SQL Server 2008	Production
Business critical	Microsoft SQL Server 2000	Other
Business critical	Microsoft SQL Server 2000	Other
Business critical	Microsoft SQL Server 2000	Other
Business critical	Microsoft SQL Server 2000	Other
Business critical	Microsoft SQL Server 2000	Other
Business critical	Microsoft SQL Server 2000	Other
DR - Department important	Microsoft SQL Server 2000	DR
DR - Department important	Microsoft SQL Server 2000	DR
DR - Department important	Microsoft SQL Server 2000	DR
DR - Department important	Microsoft SQL Server 2000	DR

Continuación de la tabla I.

DR - Department important	Microsoft SQL Server 2000	DR
Business critical	Microsoft SQL Server 2005	Production
Business critical	Microsoft SQL Server 2005	Production
Business critical	Microsoft SQL Server 2005	Production
Business critical	Microsoft SQL Server 2005	Production
Business critical	Microsoft SQL Server 2005	Other
Business critical	Microsoft SQL Server 2005	Other
Business critical	Microsoft SQL Server 2005	Other
Business critical	Microsoft SQL Server 2000	Other
Department Important	Microsoft SQL Server 2005	Development

Fuente: elaboración propia.

Otro activo determinante en el proceso es la definición de un *run book*, o libro de ejecuciones, que significa y sirve para recolectar todos los procesos repetibles que describen los algoritmos de los pasos de las actividades que se realizan por los administradores de sistemas dentro de su rutina.

Esto sirve para garantizar que tenemos todos los artefactos necesarios para ejecutar las tareas que realiza cada rol en un momento de desastre (Pearl, 2015).

Para asegurar las fases del ciclo de vida del plan de continuidad de negocio, se describen en 5 fases al igual que cualquier diseño de proceso: análisis, diseño, implementación, pruebas y mantenimiento.

En la fase de análisis se cubre el impacto del negocio, las amenazas y los riesgos, los escenarios impactantes estos son vitales para cubrir los requerimientos necesarios para la recuperación.

La percepción de la aceptación de los riesgos es crítica para determinar las soluciones de alta disponibilidad y recuperación.

Por lo que la determinar la capacidad del riesgo de la pérdida de datos en un cierto periodo de tiempo es importante, así pues, por ejemplo, concluir ¿cuánto tiempo puedo estar fuera deservicio?, ¿cuál es el riesgo aceptable para conducir la pérdida de datos?

La construcción del plan se basa en la definición de los objetivos de punto de recuperación y de tiempos de recuperación, en donde el punto de recuperación determina cual es el mínimo punto aceptable de datos permitido por el negocio para perder datos, el máximo tiempo aceptable para recuperación del servicio (Pearl, 2015).

7.5. El uso de las nubes híbridas en plataforma de datos

La nube híbrida es un tipo de nube cubierta por el ambiente de computación de la nube que utiliza una mezcla de recursos locales como de nube privada y nube pública, cuyos servicios e interacción y su orquestación participan en diferentes plataformas.

Las organizaciones utilizan este paradigma de diseño de infraestructura para garantizar eficiencia y seguridad, mantienen sus aplicaciones en un ambiente local e implementan servicios a través de nube privada y es en esta combinación de paradigmas de infraestructura que se determina la nube híbrida (Kousalya, Balakrishnan, y Pethuru, 2017).

Según su necesidad y diseño las arquitecturas de nubes se catalogan en:

- Nube pública
- Nube privada
- Servidores dedicados

- Entre las características que ofrece la nube híbrida se observan:
 - Flexible
 - Eficiencia de costo
 - Seguridad
 - Escalabilidad

7.6. Los componentes clave en las nubes híbridas

Las nubes híbridas empoderan las infraestructuras de las instituciones para innovar rápidamente mientras logran superar el grado de eficiencia de su actual infraestructura, mejorando sustancialmente su habilidad de crecimiento sustentable y seguro en relación al riesgo de invertir de manera local. Al delegar parte de la infraestructura en una nube le permite al negocio ajustar la carga de recursos de manera óptima según su presupuesto y objetivos.

Las diferentes capacidades que tiene la nube híbrida son catalogadas según su uso, dichas funcionalidades son:

- Incremento en la potencia de servicios en la nube: la carga masiva es una característica usada en las nubes privadas, ya que se convierte en esencial el movimiento de datos digitales de la organización hacia la nube privada. Esta característica se reconoce como la capacidad del incremento en la carga y la acción recomendada para manejo de la situación, para determinar cuándo es oportuno realizar un incremento del uso de los recursos computacionales y la identificación de que recursos es necesario aumentar o disminuir.
- Migración y soporte de máquinas virtuales: una de las características esenciales es la migración de los recursos en sitio hacia la nube, esto es

el movimiento de las máquinas virtuales existentes hacia la nube, en donde el soporte de los principales hipervisores existe en la nube privada.

- Soporte de imágenes a medida: ya que el servicio esencial es mantener los servidores virtuales, existen diferentes licencias de productos de sistemas operativos en la nube para el manejo de las necesidades de los clientes para sus actividades de soporte como lo son: parcheo, actualización e instalación.
- Librería de imágenes: conjunto de imágenes disponibles para reproducir un sistema operativo virtual de manera instantánea para su configuración.
- Autoescalado de recursos: como se mencionó el manejo de carga de datos es fundamental, en donde el crecimiento y decrecimiento de recursos computacionales es fundamental para el manejo de la demanda del servicio (Raj, 2018).

7.7. Costos de uso de la infraestructura sobre nube híbrida

Las organizaciones pueden implementar la infraestructura de la nube en sus propios centros de datos, como una nube privada, o utilizar la infraestructura de la nube pública bajo demanda con cargo por pago por uso. Las organizaciones también pueden adoptar una solución híbrida, es decir, utilizar la capacidad de la nube pública para complementar los recursos en la nube privada, Uno de los factores importantes que afectan las decisiones de las organizaciones de adoptar una nube híbrida es el costo total de adquirir y administrar la infraestructura.

Los costos de la infraestructura híbrida se describen en componentes de costos basados en el recurso que incurre uso dentro de los factores identificados se encuentran:

- Dependencia de la demanda o uso: el costo de una capacidad de infraestructura se puede estimar como un producto del volumen de la capacidad cobrada y el costo unitario de esa capacidad (teniendo en cuenta el tiempo de uso esperado, así como los posibles cargos de reserva y el volumen descuentos). Estos factores dependen de si los recursos pertenecen a nubes privadas o públicas. Su costo es proporcional a la cantidad de recursos adquiridos y, por lo tanto, es proporcional a la demanda máxima que atiende.
- Dimensión de tiempo: el uso continuo del crecimiento del poder computacional de los proveedores de nubes ha hecho eficiente el uso de recursos, así como el crecimiento de los usuarios demandantes por lo que la reducción de la tarifa de servicio por parte de los proveedores es notoria.
- La tendencia de la baja en los precios es considerable para la implementación de nubes públicas (Mazhelis, 2012).

7.8. Integración de monitoreo en la nube híbrida

Al conectar el sistema central de comunicaciones y operaciones (*SCOM*) a la nube para monitorear, se puede lograr escalar la eficiencia y agregar valor ya que la nube provee funcionalidades complementarias al manejo del (*SCOM*) al coleccionar, analizar y almacenar los datos de sistema de comunicaciones y operaciones. Visto a gran nivel la integración permite:

- Realizar análisis configurable de los datos de *log* de los servicios en la nube.
- Agrega y habilitas soluciones adicionales en su espacio de trabajo que no sean las configuraciones de supervisión de agentes. Las soluciones de supervisión de gestión ayudan a recopilar datos y telemetría adicionales para recursos específicos. Puede examinar todas las soluciones de administración y TI disponibles (Chang y Chen, 2013).

Los puntos por considerar al integrar el sistema de comunicaciones son:

- Debe considerarse la implementación de una puerta de salida para el módulo de análisis de *log*, el cual estará configurado con el servicio local.
- El servidor de comunicaciones no envía datos a un almacén de datos.
- El servidor de administración almacena un caché de los datos localmente por lo que debe considerarse el espacio local de almacenamiento. Debe considerarse que servidor estará monitoreando en caso de caída del servicio o conexión a la nube (Chakraborty y Karthikeyan, 2019).
- Provisionarse autenticación a través de método seguro (OAuth).
- Un solo punto de acceso.
- Prevención de ataques de fuerza bruta.

7.9. Integración con el servicio en la nube

Para lograr la integración al servicio de nube es importante remarcar la selección y uso de la *API* de nube multiplataforma, que proporciona un nivel más alto de abstracción que la proporcionada por el proveedor de nube específico, la cual permite llamadas unificadas para aprovechar los recursos de uno o más nubes, considerando el cambio de proveedor u otra restricción de uso.

Esto a su vez reduce la complejidad del código dentro del *script* y reduce los costos de desarrollo al eliminar la necesidad de implementar el acceso de una variedad de servicios en la nube.

Existen diferentes proyectos de manejo de *API* multiplataforma como *Apache Libcloud*, cuyo *API* se basa en código *Python*, así como *Deltacloud* que se utiliza comúnmente para integrar la nube *Openstack*, ambos de tipo código abierto, así como un set de herramientas que provean el soporte. *Amazon web services* ofrece el servicio de virtualización *Vmware* en la nube que es un entorno basado en *Vmware Vsphere*, así también utiliza su servicio de migración de ambientes locales hacia la nube (Park, Kim, Yun, y Yeom, 2020).

8. PROPUESTA DE ÍNDICE DE CONTENIDOS

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES

LISTA DE SÍMBOLOS

GLOSARIO

RESUMEN

PLANTEAMIENTO DEL PROBLEMA

OBJETIVOS

RESUME DE MARCO METODOLÓGICO

INTRODUCCIÓN

1. ANTECEDENTES

2. MARCO TEÓRICO

3. PRESENTACIÓN DE RESULTADOS

3.1. Implementación de plataforma de datos híbrida

3.1.1. Implementación de alta disponibilidad a través de *failover cluster* en Mysql

3.1.2. Implementación de replicación en nube híbrida Mysql AWS RDS

3.1.3. Monitoreo y control de nube híbrida

3.1.4. Proceso general de obtención de datos

3.2. Análisis y diseño de la arquitectura de la herramienta de predicción de recursos

3.2.1. Casos de uso

- 3.2.2. Vista lógica
- 3.2.3. Vista de desarrollo
- 3.2.4. Vista física
- 3.2.5. Vista de escenarios

4. DISCUSIÓN DE RESULTADOS

- 4.1. Análisis de resultados de desempeño
- 4.2. Análisis de resultados de fiabilidad

CONCLUSIONES

RECOMENDACIONES

REFERENCIAS

9. METODOLOGÍA

9.1. Tipo de estudio

El enfoque del presente estudio será de tipo mixto. El estudio busca definir un marco de medición y comportamiento de los recursos durante la mejora del desempeño y migración de escalabilidad de recursos de la base de datos para el sistema de control académico de la Facultad de Humanidades.

Luego de contar con el componente tecnológico, se procede a realizar el análisis de la información generada, de manera que pueda verificarse que al implementar ajustes y técnicas de alta disponibilidad se produce un impacto positivo directo en el uso de los recursos y en los atributos de fiabilidad tales como disponibilidad, desempeño, eficiencia y tolerancia a fallos.

9.2. Diseño

La investigación propuesta para el presente trabajo, se realizará mediante el diseño no experimental, la investigación y la utilización de diferentes técnicas, que tienen como finalidad principal:

- Identificar mediante el proceso de análisis de métricas, las respuestas a las variables del estudio para la implementación para la prueba de la herramienta de predicción y monitoreo de recursos para base de datos *Mysql*, la cual será el insumo para la interpretación de los resultados de la implementación de alta disponibilidad y ajuste de la base de datos del sistema de control académico de la Facultad de Humanidades.

9.3. Alcance

La investigación tiene un alcance descriptivo y correlacional, por medio de la herramienta del análisis del rendimiento y predicción de los recursos del motor de la base de datos *Mysql* que ayudará a determinar el impacto en las operaciones en el sitio, para los administradores de base datos del Departamento de Informática de la Facultad de Humanidades.

9.4. Variables

En la tabla II, se detallan las variables, subvariables e indicadores que se consideran para desarrollar el proyecto.

Tabla II. **Definición de variables**

Variables	Definición	Subvariables	Indicadores
Uso de los recursos de base de datos de <i>Mysql</i> .	Variable que cuantifica el uso de los recursos de base de datos <i>Mysql</i> : <i>CPU</i> . Memoria, disco y red.	Indicadores de fiabilidad	Disponibilidad Desempeño Eficiencia Tolerancia a fallos. Capacidad

Fuente: elaboración propia.

El siguiente describe el detalle de las subvariables a estudiar:

- Disponibilidad: uso de recursos disponibles por usuario en un periodo de tiempo.
- Desempeño: uso de los recursos utilizados en proporción a los recursos totales del servicio.
- Eficiencia: tiempo de respuesta por transacciones despachadas.

- Tolerancia a fallos: cantidad de fallos, *failovers* y *switchovers* durante un periodo de tiempo.
- Capacidad: cantidad de usuarios soportados en un periodo de tiempo.

9.5. Fases del estudio

A continuación, se detallan las fases realizadas para llegar concretar el desarrollo de la investigación, las fases describen los métodos de recolección de información e instrumentos que se utilizarán para la investigación.

9.5.1. Fase 0: documentación inicial

Dentro de la fase de la documentación inicial se logrará visualizar las diferentes técnicas que servirán para la implementación de las bases del proyecto.

- Entrevistas documentales al personal del Departamento de Informática; las entrevistas respectivas serán enfocadas para la obtención de los procesos de administración del sistema de control académico y las técnicas que utilizan actualmente el departamento para resolver contingencias de recuperación de incidencias de fallos de la base de datos del sistema, conocer con cuantos ambientes (instancias) de bases de datos cuentan actualmente y el propósito de las mismas, identificar el nivel de importancia de los ambientes (instancias) de base de datos con que cuentan. Con las cuales se determinará el grado de procesos y estrategias con que cuenta el departamento para lograr definir un diagnóstico inicial que involucra el desempeño del sistema.

- Consultas de base datos (*queries*) de obtención de estadísticas de desempeño sobre los metadatos del motor; se construirán *scripts* de consultas sobre los metadatos de las estadísticas que servirán para obtener el desempeño del motor de la base de datos de los ambientes del sistema con el objetivo de obtener una recopilación inicial de datos y realizar un diagnóstico inicial sobre los ajustes y mejoras a la parametrización de la base de datos con la actual infraestructura.
- Extracción de *logs* de motor de base de datos, para la obtención de las alertas e información necesaria para la realización del diagnóstico inicial del desempeño de la base de datos.
- Extracción de *logs* de sistema operativo, para la obtención de las alertas a nivel de sistema operativo que tienen relevancia en el uso de recursos en el motor de base de datos con el objeto de realizar un diagnóstico inicial.
- En el desarrollo de la herramienta de predicción de recursos se utilizará un ambiente de pruebas utilizando una arquitectura de bases de datos orientado microservicios, se verificará documentación técnica sobre las siguientes tecnologías para los temas de:
 - Utilización de microservicios en base de datos *Mysql*.
 - Utilización de base de datos de series de tiempo.
 - Utilización de *Grafana* para la manipulación de base de datos de series de tiempo y construcción de *dashboards*.

9.5.2. Fase 1: diseño y desarrollo de la herramienta

En esta fase se realizará el diseño de la herramienta de predicción y los módulos que la comprenden considerando un ambiente de desarrollo y pruebas utilizando tecnología orientada a microservicios.

Este es un insumo necesario para las siguientes fases ya que dicho componente será de utilidad para la captura y obtención de información del desempeño del ambiente productivo para la evaluación eficaz del impacto que tiene el uso eficiente de los recursos en los atributos de fiabilidad.

El diseño de la herramienta estará constituido por los siguientes componentes o módulos:

Para el cumplimiento de dicho objetivo propuesto en la fase, se diseñará la arquitectura mediante el uso de la herramienta de modelado *UML* (Lenguaje unificado de modelado) para la elaboración del diseño contenida en los siguientes puntos:

- Representación de la arquitectura.
- Restricciones y metas de la arquitectura.
- Vista lógica.
- Elaboración del diagrama y representación de la arquitectura.
- Elaboración del diagrama de componentes.
- Diagrama de casos de uso.
- Casos de uso de los componentes.
- Diagrama y descripción de la seguridad asociada los componentes.
- Diagrama de clases de los componentes.
- Diagrama general de secuencia de la invocación de los componentes.

- Vista de la implementación.
- Diagrama de la implementación.
- Diagrama de puesta en marcha.
- Diagrama entidad y relación.
- Requerimientos de *hardware*, *software*.
- Infraestructura de servidores.
- Interoperabilidad con otros sistemas.

Para la construcción de la herramienta se utilizará una metodología de desarrollo de *software* ágil, en la que se realizarán ciclos de desarrollo en los cuales se realizaran las siguientes actividades generales que se deben incluir para cumplir el objetivo de esta fase:

- Instalación y configuración del ambiente de desarrollo.
- Instalación de herramientas.
- Instalación de servicios y servidores.
- Configuración de máquinas virtuales.
- Codificación de los componentes informáticos.
- Configuración del equipo en las instalaciones del ambiente productivo para colocación de la herramienta.
- Sesiones de pruebas y realimentación de los componentes desarrollados.
- Evaluación final del prototipo desarrollado con sus ajustes.

9.5.3. Fase 2: migración a servidor de producción

Durante esta fase se procederá a realizar la construcción y configuración del clúster de base de datos e iniciaran las técnicas de generación de archivos de respaldo, transporte, verificación y migración de la base de datos.

Para el cumplimiento de dicho objetivo se procederá a realizar las siguientes actividades:

- Construcción de máquinas virtuales.
- Creación de la red de área de almacenamiento compartido (*SAN*).
- Configuración de *clúster Linux*.
- Toma y transporte de archivos de respaldo en ambiente *clúster*.
- Restauración de base de datos en ambiente *clúster*.
- Redirección de aplicación hacia ambiente *clúster*.
- Durante este periodo se monitoreará y se ajustará haciendo uso de la herramienta de monitoreo la cual se utilizará para extracción y análisis de información. En este periodo se harán ajustes en los reportes y *dashboards* para hacer uso del diagnóstico y datos en tiempo real del uso de los recursos.

Paralelamente se obtendrá un listado de los recursos que comprende los objetos lógicos y físicos que componen la infraestructura, tales como:

- Procedimientos almacenados.
- Funciones, *logins*, roles.
- Asignación de roles.
- Vistas, contadores, índices y particiones.
- Configuraciones de instancias.
- Puertos, puntos de montaje.
- Distribuciones físicas y lógicas de almacenamiento.
- Configuraciones de servidor.

Para el aseguramiento de las actividades de migración se procederá a realizar y probar scripts a nivel de base de datos para cumplir con los siguientes puntos:

- Toma de archivos de respaldo.
- Verificación de prueba de integridad física de los archivos.
- Verificación de integridad en la restauración de la base de datos.

9.5.4. Fase 3: construcción de plataforma híbrida

Durante la fase 3 se procederá a la construcción de una instancia en el servicio *Amazon*, cuyo objetivo es la integración de la misma a la infraestructura en sitio de manera que exista un sitio alterno con redundancia de datos. Para el cumplimiento de dicho objetivo se efectuarán las siguientes actividades:

- Asignación de grupo de recursos de infraestructura en *Amazon* para la creación del servicio.
- Creación y configuración de base de datos *Mysql* sobre la plataforma *AWS RDS*.
- Creación y transporte de archivo de respaldo de base de datos del sistema de control académico.
- Restauración en modo de recuperación de base de datos.
- Configuración de replicación en modo esclavo entre base de datos secundaria y clúster primario en instalaciones locales.

Una vez concluida la integración de la replicación, se procederá a monitorear como punto de control para evaluar los resultados de desempeño para evaluar el comportamiento de la integración, el monitoreo será efectuado a través de:

- Herramienta de predicción de recursos construida en la fase 1.
- *Amazon web services cloudwatch.*

9.5.5. Fase 4: presentación de resultados

En esta fase se realizará un análisis estadístico tomando como base la información obtenida de las herramientas de monitoreo ejecutadas en la fase 2 y 3. Se utilizarán métodos estadísticos de análisis cuantitativo como regresión lineal, curva normal para las variables independientes de los recursos de desempeño y cualitativas utilizaciones la prueba χ^2 de Pearson.

El objetivo principal es evaluar los recursos de la base de datos durante el transcurso del mejoramiento de la infraestructura de datos al escalar los recursos e implementar las tecnologías que permitan cumplir con dicho objetivo.

Los recursos por ser monitoreados por las herramientas serán:

- CPU
- Almacenamiento
- Memoria
- Uso de red

El análisis de los resultados de fiabilidad será evaluado a través del método de la prueba χ^2 de Pearson, cuyo objetivo será medir los indicadores de fiabilidad del servicio frente a los parámetros de servicio esperados.

Los indicadores de fiabilidad a evaluar son los siguientes:

- Disponibilidad

- Desempeño
- Eficiencia
- Tolerancia a fallos.
- Capacidad

9.6. Técnicas de recolección de información

Las técnicas de recolección utilizadas para el desarrollo de la investigación de la identificación de la problemática en el Departamento de Informática de la Facultad de Humanidades sobre un espacio de la muestra que comprende 4 personas del Departamento de Informática, descritos por sus cargos comprende a: jefe del departamento, analista de sistemas, analista de infraestructura y analista de base de datos.

- Técnica de recolección de entrevistas de preguntas abiertas enfocadas a la infraestructura de base de datos, a la problemática y objetivos del departamento. Dentro de las preguntas realizadas se enfocó en la demografía de los usuarios del sistema.
- Técnica de entrevistas registradas a través de notas de registros anecdóticos, proporcionadas por los administradores del sistema de control académico.
- Detalle de los ambientes y el nivel de importancia utilizados en la infraestructura que comprenden el sistema.
- Los productos y servicios de sistema operativo y versiones utilizados en la infraestructura.

- Problemas identificados por el departamento enfocados a la base de datos y desde cuando suceden.
- Técnicas de recuperación utilizadas en los incidentes registrados.
- Configuración actual de la base de datos.
- Técnica de generación de consultas a la base datos (*queries*) para la obtención del diagnóstico de los indicadores de desempeño de recursos de base de datos mediante las vistas de metadatos.

Los metadatos por consultar son:

- Variables del esquema de desempeño.
- Consulta de parámetros de desempeño a través de la instrucción: *shows variables like 'perf%'*.
- Consulta de los eventos históricos del tiempo de espera de los procesos en la vista *performance_schema.events_waits_history*.
- Creación de un script que genere una tabla capaz de contener los datos de la información mantenida en memoria del esquema *performance_schema* y guarde los datos de manera permanente para su extracción futura.
- Técnica de extracciones del log binario del motor de base de datos y del sistema operativo para la obtención de los recursos a través de los programas *TOP* y *TOPAS*.

- Generación de un *daemon* calendarizado para ejecutarse cada 2 horas con el objeto de ejecutar un *bash script* para obtener la información de los programas *TOP*, *TOPAS*, *PS* y almacenarlo en un archivo plano, que contiene la información el detalle del uso de los recursos en ese periodo de tiempo.
- Generación de un *script* y calendarización en el sistema operativo para la copia del archivo binario del log del motor de base de datos *Mysql*.
- Técnica de documentación para el ambiente de desarrollo, en el que se obtendrán recursos técnicos para los temas.
- *Grafana* para la construcción de *dashboards* de monitoreo.
- *InfluxDB* para la construcción de base de datos de series de tiempo para almacenar los datos del uso de los recursos.
- *Docker* para la construcción de la base de datos *Mysql* de sujeto de pruebas.
- Técnica de documentación de la arquitectura de la herramienta de monitoreo basado en la técnica 4+1 Vistas *UML* que describen:
 - Vista lógica.
 - Vista física.
 - Vista de procesos.
 - Vista de datos.
 - Escenarios.

- Técnica de metodología de desarrollo a través de *Scrum* aplicando las fases de un *sprint*:
 - Inicio.
 - Planificación y estimación.
 - Implementación.
 - Revisión y retrospectiva.
 - Lanzamiento.

- Técnica de recolección de información en la construcción y configuración del ambiente de clúster.

- Obtención del *log* de la herramienta de virtualización *Proxmox* de los nodos virtuales, obteniendo los datos de configuración y *uptime*, versión y datos del sistema operativo.

- Obteniendo el *log* del punto de montaje y descripción del sistema de archivos distribuido en las particiones lógicas del sistema operativo indicando los puntos de distribución de la instalación binaria del motor, los datos y el *log* de la base de datos.

- Obteniendo los scripts ubicados en cada nodo del clúster en el árbol de sistema */var/opt/* y */etc/conf* relacionados a la base de datos.

- Obteniendo el *log* de respuesta del transporte y copia de archivos al ambiente clúster.

- Obteniendo el *log* del resultado de la restauración de base de datos en ambiente clúster.

- Obteniendo el *log* de las transacciones realizadas luego de la redirección de aplicación hacia ambiente *clúster*.
- Técnica de recolección de información en la construcción y configuración del ambiente de replicación.
- Obtención del reporte del grupo de recursos creados en la infraestructura de Amazon para la creación del servicio.
- Obtención del reporte de la plataforma como servicio en la creación de la base de datos *Mysql* sobre la plataforma *AWS RDS*.
- Obtención de *log* de restauración en modo de recuperación de base de datos en *AWS RDS*.

10. TÉCNICAS DE ANÁLISIS DE INFORMACIÓN

Las técnicas de análisis de información utilizadas para el desarrollo del proyecto que comprende el análisis de los datos obtenidos en el ambiente de base de datos de control académico de la Facultad de Humanidades sobre un espacio de la muestra que comprende:

- Ambiente productivo actual de la Facultad de Humanidades.
- Ambiente de pruebas de desarrollo de herramienta.
- Ambiente de *failover clúster* a migrar en la Facultad de Humanidades.
- Ambiente en la nube de replicación.

Las técnicas estadísticas por utilizar en el presente proyecto son de análisis descriptivo unidimensional que buscan detallar el comportamiento de la variable independiente que comprende la utilización de los recursos de la plataforma de base de datos de control académico. Mediante dicha técnica describirá la distribución de los datos, y a su vez entender la correlación de las variables.

Las técnicas de análisis de estadística descriptiva básica que se utilizarán en este proyecto serán:

- Media
- Desviación estándar
- Varianza
- Covarianza

El análisis de la variable dependiente a estudiar permitirá analizar los indicadores de fiabilidad del servicio de la base de datos. Para esta variable se utilizará el método de prueba χ^2 de Pearson, ya que se considera una prueba no paramétrica con el objeto de medir la discrepancia entre la distribución de datos observada y el parámetro de referencia del nivel de confianza, en el servicio de cada atributo de calidad. La medición se realizará a través del conteo del número de incidencias reportadas en un determinado tiempo de servicio contra el número esperado de veces toleradas que el servicio puede fallar. Para lograr demostrar la hipótesis de que el sistema cumple en mantener el servicio en un 95 % de fiabilidad en el tiempo de observación del proyecto.

La técnica de análisis de estadística que se aplicará sobre los atributos de fiabilidad es:

- Prueba χ^2 de Pearson

A través del uso de la herramienta desarrollada durante este proyecto, se ejecutará el proceso para obtener información de desempeño del uso de los recursos, la cual será insumo que serán utilizados para el análisis de la tendencia de uso de recursos que pueda describir y apoyar en el soporte del detalle de las variables de control es decir de la utilización del uso de los recursos.

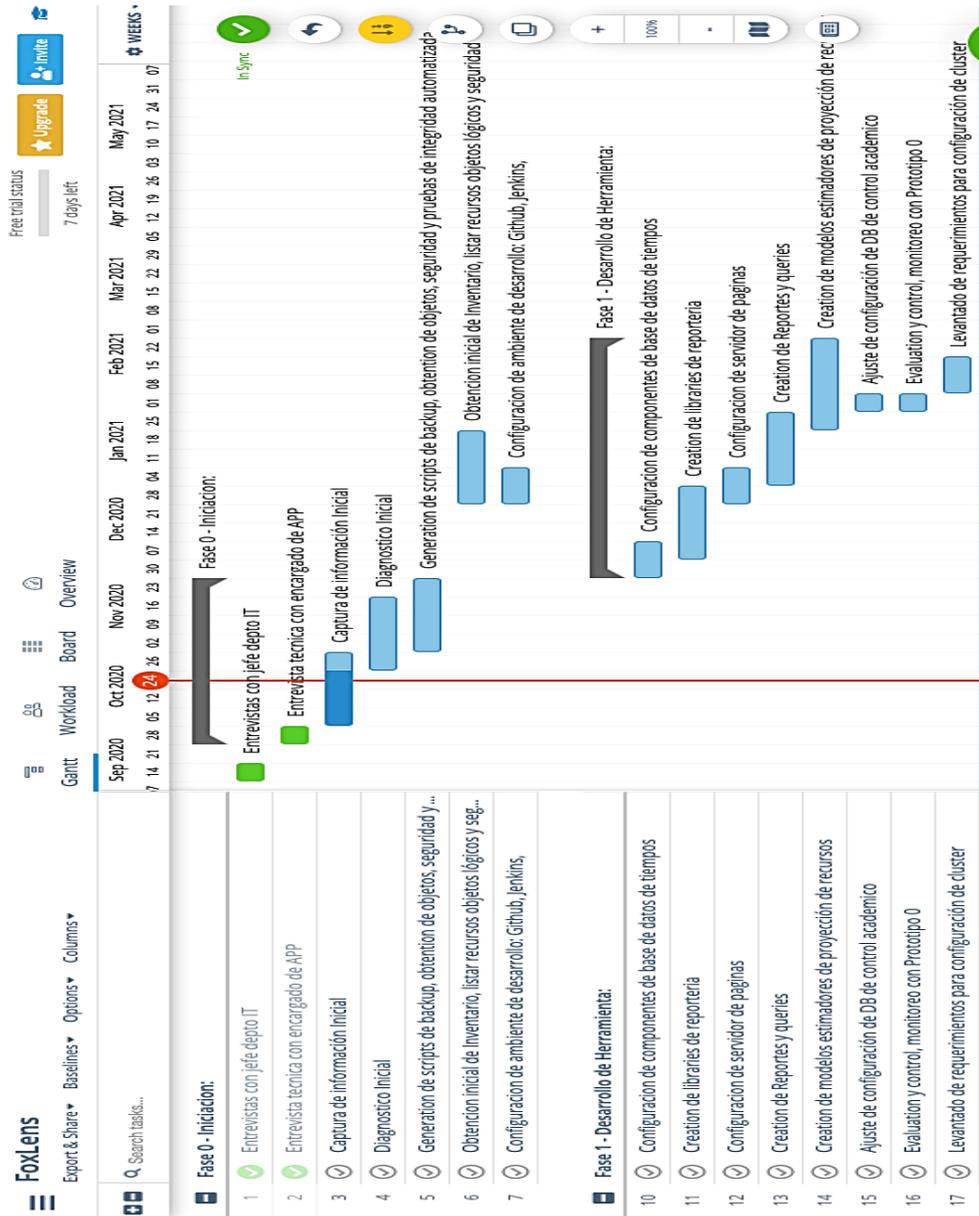
La técnica de análisis de diseño de la herramienta será descrita a través de los siguientes diagramas de arquitectura:

- Diagrama de entidad relación.
- Diagrama general de secuencia de la invocación de los componentes.
- Diagrama de componentes.
- Diagrama de paquetes.

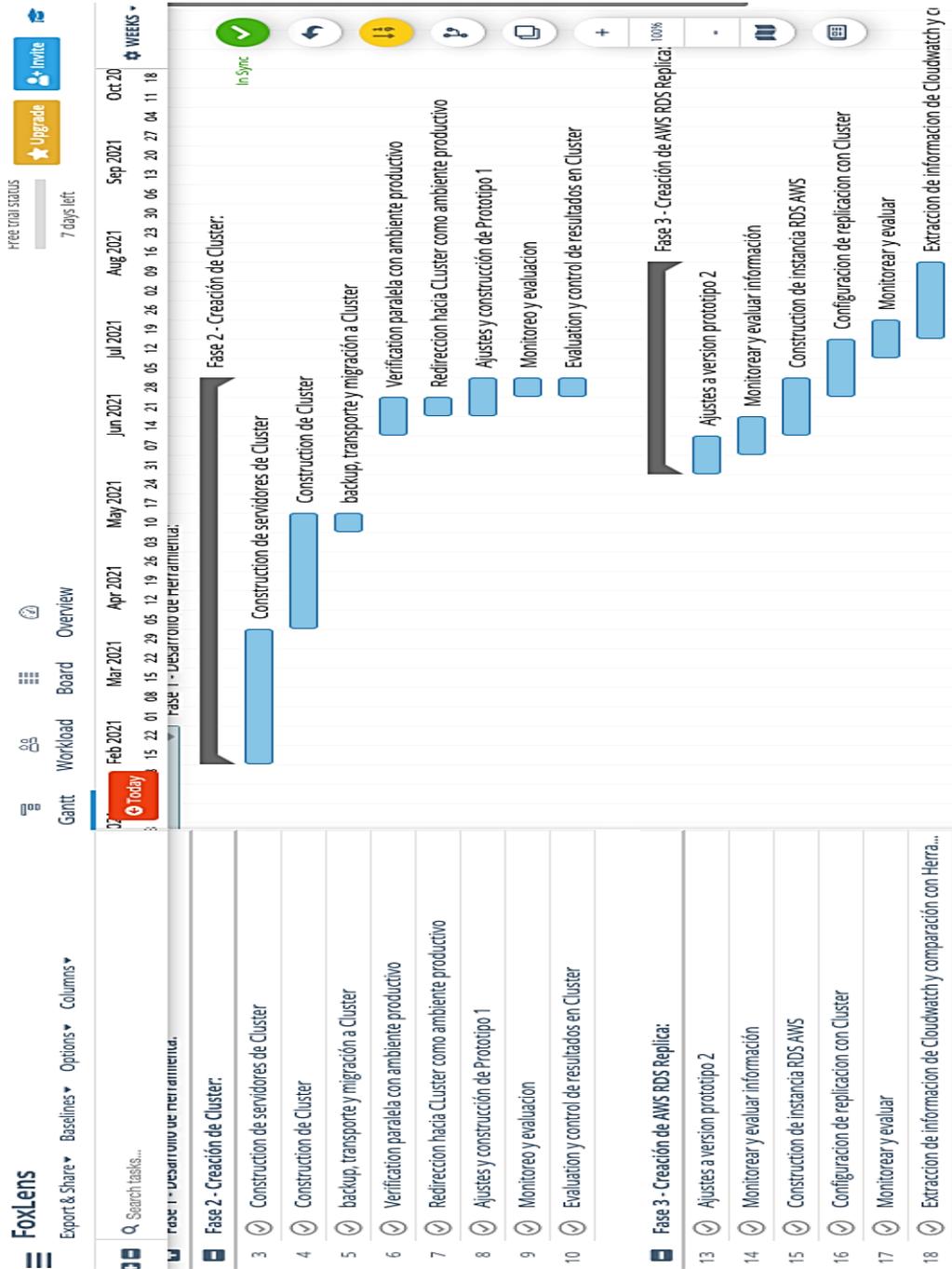
- Diagramas de casos de uso.

11. CRONOGRAMA

Figura 3. Cronograma de actividades



Continuación de la figura 3.



Continuación figura 3.



Fuente: elaboración propia.

12. FACTIBILIDAD DEL ESTUDIO

12.1. Factibilidad operativa

La factibilidad operacional de este proyecto se enfoca en el mejoramiento de la infraestructura con el objeto de lograr eficiencia en los procesos del entorno de los servicios de base de datos de control académico y así cumplir los objetivos estratégicos de la unidad facultativa.

Con el fin de proveer un servicio de calidad, en el que se garantiza un buen funcionamiento del sistema y una excelente percepción por parte del usuario, entregando un servicio fiable, la plataforma de datos permitirá mejorar el aseguramiento de los datos de una forma confiable y eficiente, asimismo con este mejoramiento el aprovisionamiento de herramientas previsoras para garantizar oportunamente el monitoreo y control del servicio para el soporte manejado por el Departamento de Informática de la Facultad de Humanidades, lo cual generará aspectos positivos que son descritos a continuación:

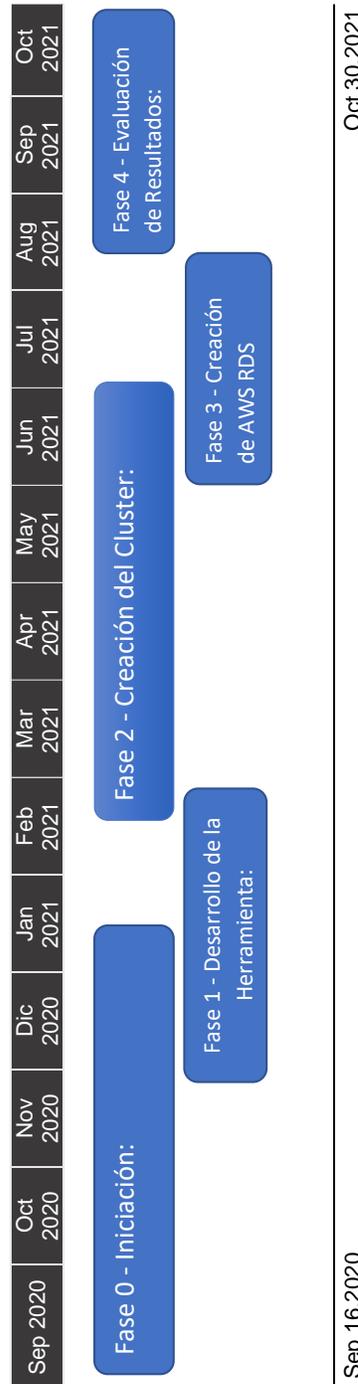
- Se disminuye el tiempo de procesamiento de datos por parte del personal administrativo.
- Mejoramiento del rendimiento del servicio para el estudiante.
- Aumenta la capacidad de servicio permitiendo un crecimiento la demanda de usuarios conectados en el sistema.
- Aumento en la disponibilidad de servicio al mejorar la capacidad a la tolerancia de fallos y reduciendo el riesgo en la caída del servicio.
- Mejoramiento de la fiabilidad al asegurar la redundancia de datos en sitios alternos.

Para lograr ejecutar los impactos positivos de la plataforma de datos híbrida, se llevará a efecto la siguiente programación de actividades operacionales para obtener la factibilidad del proyecto.

- Análisis y diagnóstico de la configuración actual.
- Generación de scripts de automatización.
- Ajuste y verificación.
- Adecuación de servidores.
- Acceso a la infraestructura.
- Implementación física de servidores.
- Implementación de red física de servidores.
- Verificación de puntos de red.
- Configuración de servidores virtuales.
- Implementación de servicio de nube.
- Configurar el saldo mensualmente del servicio de *AWS RDS*.

El recurso tiempo queda plasmado en el cronograma agrupado por fases que determinan los plazos del desarrollo de la implementación completa.

Figura 4. **Planificación general del proyecto**



Fuente: elaboración propia.

En función a la anterior definición de requerimientos necesarios para la implementación de infraestructura, y dado que el recurso humano que implementará el proyecto será donado, el financiamiento hace viable el proyecto, asimismo el apoyo del personal dedicado al soporte de la aplicación de control académico hace viable la implementación operativa del proyecto.

12.2. Factibilidad técnica

En esta sección del estudio de viabilidad, se explicarán las consideraciones de orden tecnológico que deba realizar la organización. Dada la naturaleza del proyecto, dependen de la tecnología informática ya que este servirá de insumo primario para el mejoramiento de la infraestructura.

Busca ser de enfoque en obtener un entendimiento de los recursos tecnológicos disponibles actualmente y su aplicabilidad a las necesidades que se espera tenga el proyecto.

El siguiente cuadro detalla la descripción de *hardware* estimado

- Estimación de descripción de hardware adecuado para implementación.

Tabla III. **Estimación de hardware**

Hardware	Descripción	Precio base (\$)
Servidor San Controlador	HPE ProLiant DL160 Gen10 Server	1127.99
Caja controladora	HP Q2R23A HPE MSA 1050 1GBE ISCSI DC SFF SAN	2000.00
Switching	TP-LIN 48 pts Gigabit Switch	100.00
		3227.99

Fuente: elaboración propia.

A continuación, se detalla el *software* considerado en función de licencias y servicios en la nube del proveedor configurando un servidor *standard*.

- Estimación de descripción de licencias de *software* para implementación.

Tabla IV. **Estimación de licencias de software**

Hardware	Descripción	Precio
Licencia de ProxMox	Licencia de virtualización	100.51 USD/Año
AWS RDS Instance	Standard Reserved Instances	155.97 USD/Mes

Fuente: elaboración propia.

Tabla V. **Especificaciones de software**

Hardware	Detalle	Precio (\$)	Periodo
Licencia de ProxMox	COMMUNITY Edition, Acceso al repositorio empresarial Conjunto completo de funciones Soporte comunitario	100.51	Anual
AWS RDS Mysql	Standard Reserved Instances		
Tipo de servidor	db.t3.xlarge, On-Demand hourly cost	139.72	Mensual
CPU	4 Cores		
Memoria	16 GB		
Storage	General Purpose (SSD) Storage US East	11.5	Mensual

Fuente: elaboración propia.

Considerando los factores técnicos detallados en los cuadros superiores, siendo la creación de servidores el impacto mas fuerte de inversión, y dado que parte de dicha configuración ya se posee en la infraestructura física, y que asimismo la renta de *software* en el proveedor en la nube ya se encuentra en uso, la extensión del servicio del mismo es razonable y considerable para lograr los objetivos del proyecto para beneficio de la calidad en la entrega del servicio.

Es de mencionar como un factor importante que existe el apoyo de la Facultad de Humanidades para la ejecución del proyecto dada la situación considerable del rendimiento del sistema.

Por lo anteriormente mencionado, el proyecto es técnicamente viable de ejecutar.

12.3. Factibilidad económica

El flujo de caja elaborado (véase tabla 5), detalla el movimiento del efectivo estimado para el inicio de proyecto, este ha sido elaborado tomando como base el cronograma de actividades previamente detallado.

Se estima de la implementación se llevará aproximadamente 12 meses iniciando con la compra de equipo, licencias y servicio de software en la nube en algún momento los saldos del efectivo se reflejan en negativo, pero se compensan con los ingresos programados mensualmente. Al final del periodo, se contempla que el proyecto es sostenible siempre y cuando se mantengan según lo presupuestado.

Adicionalmente, se refleja donaciones de la mano de obra del servicio técnico prestado para la ejecución del proyecto.

La propuesta no estima que se deba realizar un gasto adicional en costos de personal debido a que existe un presupuesto asignado para el personal de administración técnica.

En función a la descripción de ingresos y gastos considerables para el proyecto contando con el financiamiento de la creación de infraestructura y la

anualidad de servicio de *AWS RDS* para el servidor hace viable la implementación del mismo.

Basado en lo anterior descrito se concluye que el proyecto es factible de desarrollar desde el punto de vista operativo, técnico y económico.

- Presupuesto de proyecto para la implementación.

Tabla VI. Diagrama de flujo de caja

Descripción	Gastos de inversión											
	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto	Septiembre	Octubre	Noviembre	Diciembre
Saldo Inicial	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67	4,166.67
INGRESOS	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00
Donación de												
Salario Servicio Técnico	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00
EGRESOS	12,000.00	21,055.17	20,285.17	20,285.17	12,000.00	13,164.47	13,164.47	13,164.47	13,164.47	13,164.47	13,164.47	13,164.47
Equipo de Hardware		8,285.17	8,285.17	8,285.17								
Licencia de Software		770.00										
Servicio de Software en la Nube					1,164.47	1,164.47	1,164.47	1,164.47	1,164.47	1,164.47	1,164.47	1,164.47
Servicios Técnicos	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00	12,000.00
Servicio de Personal Local												
SALDO	4,166.67	(4,888.51)	(4,118.51)	(4,118.51)	4,166.67	3,002.20	3,002.20	3,002.20	3,002.20	3,002.20	3,002.20	3,002.20

Fuente: elaboración propia.

13. REFERENCIAS

1. Bell, C. (2012). *Extending MySQL High Availability*. California, United States of America: Apress.
2. Bell, C. (2016). *Demonstration of High Availability Techniques*. California, United States of America: Apress.
3. Bell, C. (2018). *Introducing InnoDB Cluster*. New York: Apress.
4. Brewer, E. (14 de noviembre de 2000). Towards robust distributed systems. [Mensaje de blog]. Recuperado de <https://www.ibm.com/cloud/learn/cap-theorem#toc-what-is-the-DXABpEgu>.
5. Chakraborty, B., y Karthikeyan, S. A. (2019). *Integration and Hybrid Monitoring*. Bangalore, India: Apress.
6. Chang, Y.-H., y Chen, J.-Y. (2013). *A Hybrid Cloud for Effective Retrieval from Public Cloud Services*. Malaysia: Springer.
7. Gllbert, S., y Lynch, N. (2002). Brewer's Conjecture and the Feasibility of Consistent Available Partition-Tolerant Web Services. *ACM SIGACT News*, 33(2), 51–59. Recuperado de <https://dl.acm.org/doi/abs/10.1145/564585.564601>.

8. Kousalya, G., Balakrishnan, P., y Pethuru Raj, C. (2017). *The Hybrid IT, the Characteristics and Capabilities*. Swinton, UK: Springer.
9. Mazhelis, O. (2012). *Costs of Using Hybrid Cloud Infrastructure: Towards a General Framework*. Cambridge, MA, USA: Springer.
10. Park, J., Kim, U., Yun, D., Yeom, K. (2020). *Approach for Selecting and Integrating Cloud Services to Construct Hybrid Cloud*. Busan, Korea del Sur: Springer.
11. Pearl, R. (2015). *High Availability and Disaster Recovery*. Berkeley,CA: Apress.
12. Raj, P. (2018). *The Hybrid Cloud: The Journey Toward Hybrid IT*. Bangalore: Springer.
13. Song , G., Xiaofei , L., Fangming , L., Yanmin , Z. (2015). *An ARM-Based Hadoop Performance Evaluation Platform: Design and Implementation*. Cham, Alemania: Springer.
14. Vugt, S. v. (2014). *Pro Linux High Availability Clustering*. Berkely, CA: Apress.