

SISTEMA DE MONITOREO PARA PYMES

Escuela de Ingeniería

Agosto, 2023

NOMBRE DEL ALUMNO

Patricio Valenzuela Baeza

NOMBRE DOCENTES

Oscar Pinto Garralaga

INDICE

1. Introducción	4
2. Antecedentes Generales	4
2.1 Presentación del Problema/Oportunidad.....	4
3. Fundamentación del Problema/Oportunidad.....	6
3.1 Descripción de causas	9
4. Gestión del Alcance del proyecto ³.....	10
4.1 Alcance del producto	10
4.2 Alcance del Proyecto	10
4.3 Requerimientos	11
4.4 Supuestos y limitaciones del alcance.....	11
4.5 Análisis de brechas	12
5. Objetivos del proyecto	12
5.1 Objetivo General.....	12
5.2 Objetivos específicos	12
5.3 Métricas de los objetivos	12
6. Propuesta de Solución	13
7. Plan de Proyecto	14
7.1 Metodología de proyecto y metodología de desarrollo de software.....	14
7.2 Plan de monitoreo y Control.....	15
7.3 Plan de Gestión de Riesgos.....	16
7.4 Plan de mitigación de riesgos	17
8. Materiales, métodos y herramientas	17
8.1 Patrones de diseño y arquitectura	17
8.2 Arquitectura de Alto Nivel.....	18
8.3 Herramientas de software	19
8.4 Plan de gestión de la configuración.....	20
9. Gestión del Proyecto	21
10. Descripción general de la propuesta de solución.....	23
10.1 Características clave de la propuesta de solución	23
10.2 Organización del desarrollo del producto	24
10.3 Arquitectura del Software a Implementar	25
10.4 Modelo de Base de Datos	26

11. Control y Seguimiento	27
11.1 Sprint 1	27
11.1.1 Riesgos Sprint 1	28
11.1.2 Matriz de trazabilidad de requerimientos Sprint 1	28
11.1.3 Propuesta de Controles y Evidencia	28
11.1.4 Sprint 1 - Review	33
11.1.5 Sprint 1 - Retrospective	33
11.2 Sprint 2	34
11.2.1 Riesgos Sprint 2	35
11.2.2 Matriz de trazabilidad de requerimientos Sprint 2	35
11.2.3 Propuesta de Controles y Evidencia	36
11.2.4 Sprint 2 - Review	38
11.2.5 Sprint 2 - Retrospective	38
11.3 Sprint Summary	39
12. Conclusión	39
13. Bibliografía	40

1. Introducción

Dentro del marco orgánico de las empresas existe la pequeña y mediana empresa la cual su acrónimo más conocido es Pyme. El cual hace referencia a una empresa compuesta por un número reducido de trabajadores y con un volumen de ingresos netos moderado.

Las pymes tienen libertad de desarrollar actividades comerciales en cualquier tipo de sector o actividad, bien sea de producción, comercialización o prestación de servicios. Las pymes comprenden las micros, pequeñas y medianas empresas operadas por una persona natural o jurídica bajo alguna forma de organización. Este tipo de empresas están compuestas por profesionales con competencias suficientes o mínimas para desempeñarse en las áreas de finanzas, marketing, producción o servicios, así como cualquier otra área.

Pese a ello, este tipo de empresas genera buena parte de los empleos formales en la región, con estas tasas de supervivencia es fácil entender por qué quienes se arriesgan a emprender encuentran los primeros años como los más difíciles.

En el presente documento presentaremos una solución que podrá abarcar algunas de las problemáticas que se pueden encontrar en estas empresas como por ejemplo las alertas tempranas, el estado de salud de sus servicios, el estado del o los enlaces de internet. A medida que avancemos en la construcción de esta solución se presentaran nuevas versiones dando una cobertura más ampliada para estas soluciones sin impactar en lo que significa los costos asociados de inversión por parte de estas empresas.

2. Antecedentes Generales

2.1 Presentación del Problema/Oportunidad

“Los incidentes informáticos suceden a diario y normalmente y toda nuestra red de trabajo debería operar sin contratiempos”¹, ya que gracias a ella es que se facilita nuestra vida diaria. Por ello, cada vez que alguno de nuestros equipos falla o funciona de manera anormal, se pueden generar graves problemas además de frustración.

Los incidentes o problemas informativos podríamos destacarlos como los siguientes:

- **Computadoras antiguas**

Es muy normal que en las empresas pequeñas y medianas no puedan definir o no puedan contar con una inversión dedicada a una renovación de equipamiento informático dada la creencia que mientras funcione no es necesario el cambio y se cuentan muchas veces con equipos que van quedando obsoletos.

Con esta forma de operar es normal que estos equipos pueden ocasionar problemas, fallos, incompatibilidad de software, funcionamiento lento, entre otros. Dado esto estas maquinas pueden generar incidentes a diario y reducen la productividad del resto de los trabajadores.

Adicionalmente al no renovar estos equipos se enfrentan a problemas como incompatibilidad en la compra de posibles repuestos cuando se requieran y otro factor importante es que el equipo ya no soporta las últimas versiones de su sistema operativo, ni las actualizaciones de programas o aplicaciones que se usan con frecuencia.

¹ (IONET, 2020)

- **Sin mantenimiento informático**

Los mantenimientos de equipos informáticos es un proceso que debería ejecutarse en forma periódica y los cuales normalmente corresponden a la eliminación de polvo en sus componentes, verificación de sus tarjetas de redes, chequeo y eliminación de virus si existen y por supuesto la verificación de espacio en los discos de almacenamiento. Es dado que la falta de estos mantenimientos se debe a la no rigurosidad o desconocimiento de estos procedimientos los cuales afectaran el funcionamiento de estos dispositivos.

- **Ataques informáticos**

En la época actual en la cual estamos viviendo y adicionalmente por el confinamiento que vivimos por el COVID19 los ataques informáticos han ido en escalada. Estos ataques informáticos normalmente corresponden a virus malware, ataques para denegación de servicios, robo de información sensible y normalmente estos ataques nos han dejado demostrado lo vulnerable que son los sistemas por un poco o nula protección frente a estos ataques.

- **Software pirata**

El software pirata o no licenciado aparte de ser una práctica ilegal puede generar varios problemas, de los cuales podríamos decir que pueden abrir las puertas a virus y malware (programas malignos). Adicionalmente no podrían contar con las actualizaciones y podrían sufrir incidentes que no se pueden resolver de forma sencilla y sumado a esto no se podría contar con el soporte del programa.

- **No generan respaldos**

Una de las prácticas que normalmente se encuentran es la no generación de respaldos la cual es esencial para proteger los datos de la empresa ante una posible falla. Esto debido al desconocimiento de este procedimiento o falta de recursos como almacenamiento definido para estas tareas.

- **Fallas de acceso a Internet**

Los enlaces de internet es una necesidad para todo tipo de empresa, dicho esto, siempre existirá el robo de internet o ancho de banda por Wifi y esto normalmente suele ser por los equipos y antenas o bien no está bien protegida la conexión por ejemplo con claves o contraseñas débiles o fáciles de descifrar. Mucho de estos problemas intentan solucionarlo sin resolver el problema de origen y que por ejemplo solicitan aumentar la velocidad de conexión lo cual sobrepasa las necesidades reales.

- **Contraseñas inseguras**

Otro de los problemas que es bastante común en toda clase de empresa es la utilización de claves demasiado sencillas o fáciles que pueden ser descifradas rápidamente por un hacker o alguien que baje un programa de internet para realizarlo.

- **Equipos descuidados**

La no mantención de las computadoras es algo que generalmente les puede generar mal funcionamiento como por ejemplo calentamiento (sobre todo en épocas calurosas) de sus componentes o en su peor caso que estos se descompongan. Otro de los problemas que se puede dar es que el ventilador interno suena muy fuerte y funciona a su máxima potencia. Otro de los problemas que suele suceder ya sea por desconocimiento o falta de rigurosidad es colocar los equipo en lugares que no sean los correctos como habitaciones húmedas o calurosas y sin ventilación adecuada.

- **Demasiados programas y archivos sin uso**

El incorrecto uso de los recursos de almacenamiento, procesamiento o memoria por los programas no utilizados ya sean licenciados o gratis nos puede generar problemas los cuales pueden significar por ejemplo minimizar la capacidad de procesamiento siendo que estos ya no se utilizan o la utilización de recursos de almacenamiento por la generación de archivos de logs los cuales no están siendo utilizados.

3. Fundamentación del Problema/Oportunidad

Para poder fundamentar y argumentar deberemos profundizar en los problemas que se le originan a las PYME por no tener un servicio que les permita monitorear el estado de salud de sus componentes y entreguen alertas tempranas y proactivas en vez de reactivas ya que al ser este caso se enfrentan al problema de perdida de algunas operaciones o en su mayor defecto quedarían sin servicios.

Algunas de las causas que podrían provocar estas situaciones de fallas podrían ser las siguientes:

Equipamiento

Una de las cosas con la cual normalmente nos podríamos encontrar y es uno de los riesgos mas grande es la utilización de servidores antiguos los cuales los proveedores ya no generan actualizaciones a nivel de software o sus componentes y serán lo que más afectara tanto a la empresa como a sus empleados.

Los servidores corporativos en estas empresas normalmente controlan permisos, usuarios, programas y datos y es por ello por lo que es uno de los elementos mas importante dentro de las empresas.

Al ocurrir algún tipo de inconveniente como por ejemplo el servidor se apagará o sufriera la perdida de comunicación por fallas en la tarjeta de red o los recursos de almacenamiento por cualquier motivo el daño a la empresa sería incuantificable. Sin embargo, la gran mayoría de las empresas pequeñas y medianas se reprimen a una renovación para ahorrar gastos lo cual aumenta las probabilidades de fallas de estos equipos.

- Los programas se ejecutarán muy lentamente.**

Un problema que es subsecuente de utilizar equipos antiguos es que los programas utilizados cada vez se ejecutarán de una forma más lenta o presentaran fallas en su procesamiento lo cual tendrá como resultado la frustración de todos aquellos que los utilicen y que signifique que no puedan alcanzar los objetivos definidos por la empresa.

- No podrás optar a actualizaciones por lo que las vulnerabilidades serán muy altas.**

Otro problema que puede ser muy grave es la no posibilidad de actualización del software a sus nuevas versiones o parchado en el caso de que estos estén licenciados, pero como se encuentran ejecutándose en equipos antiguos no cumplirán con los requisitos mínimos y para el caso de software no licenciado no contarán con el soporte para realizar los parchados o actualizaciones. Como resultado de todo esto tendremos mayores probabilidades de quedar vulnerables a los ataques informáticos los cuales encontraran vacíos en las restricciones existentes sin poder hacer nada para protegerlos.

- Si el software está obsoleto**

Al igual que el equipamiento, los programas o software quedarán obsoletos ya que no podrán ser actualizados por los requerimientos mínimos de instalación (hardware y/o sistema operativo) y no tendremos la posibilidad de actualizaciones de seguridad y estarán desprotegidos frente a ataques de terceros.

- Incompatibilidad**

Como ya lo hemos comentado con antelación que al contar con computadores antiguos u obsoletos podría ocurrir la incompatibilidad al momento de intentar instalar software actualizados o versiones más recientes estos presentaran incompatibilidades con los sistemas operativos mas antiguos. Ahora, producto de esto mismo las caídas de los programas podrían ser más frecuentes generando la pérdida de tiempo para toda la empresa, así como la posibilidad de perder información y no poder recuperar dicha información.

Ataques Informáticos

En la época actual estamos propensos a ser víctima de un ataque informático y esto dará como resultado la perdida de confianza de parte de los usuarios tanto internos como externos y posibles daños frente a terceros, alguno de estos ataques podría ser los siguientes:

- **Malware o Programa Maligno**

Una malware o programa maligno una vez que ingresa a los sistemas puede afectar de manera secreta y silenciosa a todos los sistemas. Los malware tienen la capacidad de interrumpir y perjudicarlos computadores, dicho en otras palabras, mediante estos programas malignos nos pueden robar y eliminar dato y espiar actividades sin ser notados.

- **Denegaciones de Servicio**

Otro de los problemas del cual ninguna empresa ha estado ajena corresponde a las denegaciones de servicio o retrasos en los procesos de producción, adicionalmente puede generar el secuestro de la información.

- **Robo de Datos sensibles**

Mediante la inyección SQL que corresponde a uno de los ataques más utilizados y que es un ataque a la web y que consiste en la inyección de un código malicioso que aprovecha los errores y todas las vulnerabilidades que pueda encontrar en una página web. Normalmente es utilizado para robar bases de datos, manipular o eliminar información.

Internet

Otro de los hechos mas comunes en esta época es el robo de ancho de banda inalámbrico o WIFI² y que básicamente corresponde a conectarse a internet mediante la utilización de un router inalámbrico sin autorización del dueño del enlace. Estas actividades pueden ser utilizadas para acceder a algún computador y podrían realizar robo de identidad, además estas actividades de robo de internet les genera una degradación en los servicios del dueño del enlace.

Utilización de Espacio

La falta de monitoreo puede provocar grandes estragos o problemas producto de programas y/o archivos sin utilización los cuales se van acumulando y posteriormente generando la falta de espacio en nuestros computadores. Esto adicionalmente nos puede generar que no podamos ejecutar las copias de seguridad necesarias para una posible recuperación.

Adicionalmente otro de los aspectos que puede provocar estragos en la operación es la falta de memoria y/o swap que tenemos disponibles en los servidores, esto puede darse por diversos motivos los cuales podrían ser programas que se estén ejecutando en segundo plano o una mala configuración de alguno de los aplicativos instalados en el servidor.

² (SUMMA, n.d.)

Dado que la cantidad de causas son bastantes y que entregan el origen al problema en investigación y lograr comprender cada una de las consecuencias que pueden acarrear costos asociados y perdidas de oportunidades, en la Figura 1 presentamos estas en forma grafica como un Diagrama de Ishikawa que nos permite resumir las causas mas significativas para el problema planteado.

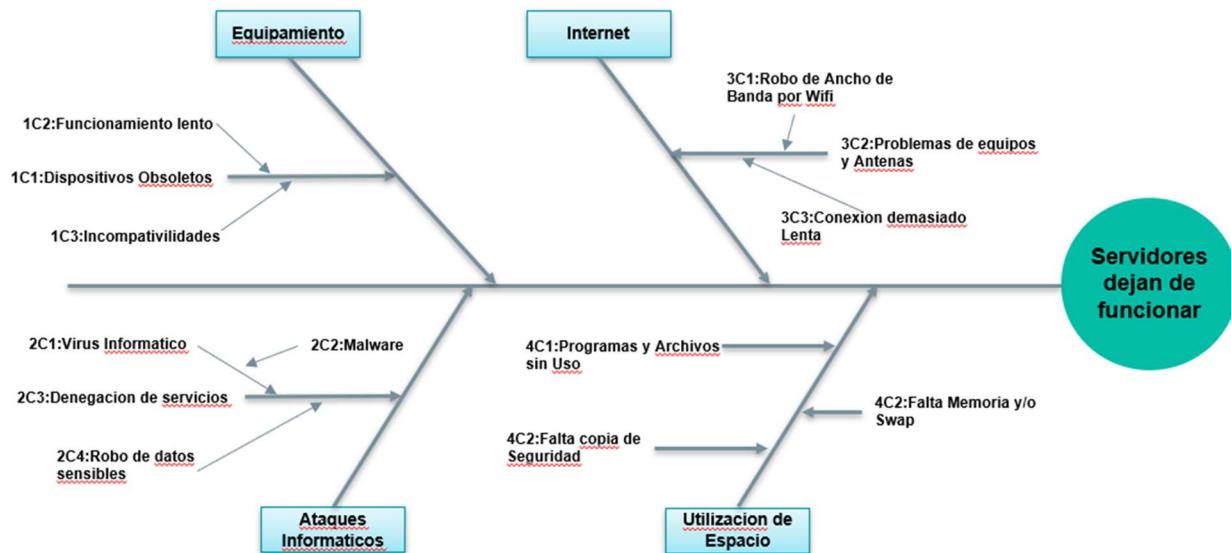


Figura 1 - Diagrama de Ishikawa (Fuente: Elaboración propia)

3.1 Descripción de causas

A continuación, definimos la Tabla 1 que contiene el detalle con su respectiva descripción de cada una de las causas que son mencionadas en el Diagrama de Ishikawa y están descritas en el capítulo 2 del presente documento (Fundamentación del Problema/Oportunidad).

Causa	Nombre de Causa	Descripción
1C	Equipamiento	
1C1	Dispositivos obsoletos	Al utilizar equipos antiguos los cuales no soportan las actualizaciones de seguridad para su sistema operativo obteniendo brechas a nivel de seguridad.
1C2	Funcionamiento lento	Consecuencia de la causa 1C, al utilizar equipos antiguos es que los programas utilizados cada vez se ejecutarán de una forma más lenta o presentarán fallas en su procesamiento.
1C3	Incompatibilidades	Otra de las consecuencias de la causa 1C es que los programas o software quedarán obsoletos ya que no podrán ser actualizados por los requerimientos mínimos de instalación (hardware y/o sistema operativo).
2C	Ataques Informáticos	
2C1	Virus Informático	Los virus informáticos tienen básicamente la función de propagarse a través de un software, son muy nocivos y algunos contienen además una carga dañina (payload).
2C2	Malware	Realiza acciones sin el consentimiento del usuario. Busca adentrarse de forma sigilosa al sistema objetivo. Su actividad en el sistema puede pasar desapercibida. Los ataques generados con malware son ilegales y, por tanto, penados por la ley.
2C3	Denegaciones de Servicio	Consiste en enviar de forma continuada un número elevado de paquetes ICMP Echo request (ping) de tamaño considerable a la víctima, de forma que esta ha de responder con paquetes ICMP Echo reply lo que supone una sobrecarga tanto en la red como en el sistema de la víctima.
2C4	Robo de Datos Sensibles	El robo de datos es el acto de robar información digital almacenada en equipos, servidores o dispositivos electrónicos para obtener información confidencial o afectar la privacidad.
3C	Internet	
3C1	Robo de ancho de banda por Wifi	Corresponde a conectarse a internet mediante la utilización de un router inalámbrico sin autorización del dueño del enlace
3C2	Problemas en equipos y antenas	Un mal funcionamiento de los equipos y/o antenas reducen la velocidad de internet utilizada.
3C3	Conexión demasiado lenta	Puede originarse tanto por la causa 3C1 o 3C2.
4C	Utilización de Espacio	
4C1	Programas y archivos sin Uso	Pueden originar grandes estragos o problemas producto de programas y/o archivos sin utilización los cuales se van acumulando y posteriormente generando la falta de espacio en nuestros servidores.
4C2	Falta de copias de seguridad	Es una de las tareas esenciales para proteger los datos de la empresa ante una posible falla y contar con un

		punto de restauración o recuperación de la información.
4C3	Falta de Memoria y/o Swap	Se puede estar dando que una mala configuración o ejecución de segundo plano este dejando sin recursos de memoria al servidor y estar impactando a los procesos que se requieran ejecutar.

Tabla 1 – Descripción de Causas (Fuente: Elaboración propia)

4. Gestión del Alcance del proyecto ³

Los sistemas de monitoreos actuales que se encuentran en el mercado actual si bien son bastante completos con una visión 360 tienen un altísimo costo y que los hacen poco accesibles para empresas de un tamaño pequeño o mediano (Pymes). Sin embargo, existen sistemas de monitoreo que son de código abierto o costo cero, pero tienen un altísimo costo de configuración para poder adecuarlos a las realidades de cada una de estas empresas. Por esta razón nace la necesidad de la implementación de un proyecto de sistema de monitoreo y de alerta temprana para empresas pequeñas y medianas, que sea de una ayuda de bajo costo para estas empresas y de fácil configuración además que sea capaz de enviar alertas a los correos de los administradores.

4.1 Alcance del producto

El producto tendrá un alcance final para su primera versión el monitoreo proactivo y de alertas tempranas de todos aquellos servidores basados en un sistema operativo Linux como por ejemplo Oracle Linux, CentOS, Ubuntu quedando excluidos cualquier otro tipo de sistemas operativo. Los monitoreos a realizar en esta primera versión para estos servidores corresponden al espacio físico de disco, utilización de memoria y del área de Swap.

No obstante, en versiones posteriores se evaluará la inclusión de servidores con sistemas operativos de otro vendor, así como motores de bases de datos o productos de tipo Middleware y funcionalidades que no se encuentran incluidas en la primera versión de este sistema.

4.2 Alcance del Proyecto

El proyecto tendrá un alcance para su primera versión de una aplicación que sirva para realizar las configuraciones requeridas para su funcionamiento y visualización de los datos recolectados. Esta aplicación cuenta con un Front End que nos permitirá visualizar la información recolectada y administrar la configuración necesaria para su funcionamiento, adicionalmente el Back End contendrá todo el desarrollo web y que se encuentra encargada de que toda la lógica de cada una de las páginas de la aplicación funcione.

³ (Asana, 2022)

4.3 Requerimientos

Los requerimientos son las necesidades, condiciones o características que un sistema de software que debe cumplir para satisfacer los objetivos y expectativas de los usuarios. Estos pueden incluir funcionalidades específicas, restricciones técnicas, requisitos de rendimiento, seguridad, usabilidad, entre otros aspectos.

Es importante tener en cuenta que los requerimientos pueden clasificarse en dos categorías que son las funcionales y no funcionales. Los requerimientos funcionales describen las acciones y funcionalidades específicas que el sistema debe realizar, mientras que los requerimientos no funcionales definen las características de calidad y restricciones técnicas que el sistema debe cumplir. Los requerimientos del proyecto están descritos en la tabla 2.

ID	Tipo	Requerimiento	Descripción
RQ1	Funcional	Registro y almacenamiento de datos	El sistema debe ser capaz de recopilar y almacenar las métricas del desempeño de los servidores a monitorear, permitiendo el análisis para evaluar el rendimiento y detectar áreas de mejora.
RQ2	Funcional	Monitoreo en tiempo real	El sistema debe proporcionar una visualización en tiempo real de los datos almacenados que corresponden a las métricas obtenidas desde los servidores.
RQ3	Funcional	Acceso remoto	El sistema debe permitir el acceso remoto a través de dispositivos móviles o computadoras, lo que permite a los usuarios monitorear y controlar el estado de salud de los servidores desde cualquier ubicación.
RQ4	Funcional	Configuración y personalización	El sistema debe ofrecer opciones de configuración y personalización para adaptarse a las necesidades específicas. Esto incluye la capacidad de definir parámetros de monitoreo, establecer límites de alerta.
RQ5	Funcional	Automatización	Debe contar con funciones de automatización, para reducir la carga de trabajo manual y mejorar la eficiencia operativa.
RQ6	Funcional	Compatibilidad	Debe ser compatible con diferentes sistemas operativos y plataformas, para maximizar la accesibilidad y usabilidad del sistema.
RQ7	Funcional	Economía de Costos	Reconocemos las limitaciones presupuestarias de las pequeñas y medianas empresas. Por lo tanto, nuestro sistema está diseñado para ser asequible en términos de costo inicial y mantenimiento continuo y con una arquitectura de bajo costo de implementación.
RQ8	Funcional	Gestión de alertas y notificaciones	El sistema debe ser capaz de generar alertas y notificaciones cuando se detecten problemas o situaciones anormales como fallas en los equipos entre otros.
RQ9	Funcional	Escalabilidad	A medida que las necesidades crecen, nuestro sistema es escalable. Puede integrar fácilmente más sensores y funcionalidades adicionales según sea necesario.
RQ10	Funcional	Respaldos y Recuperación	Debe contar con funciones de respaldo y recuperación de datos, para evitar la pérdida de información en caso de fallos o incidentes.
RQ11	Funcional	Usuarios y Roles	Debe permitir la gestión de usuarios y roles, para controlar el acceso a la información según los permisos establecidos.
RQ12	No	Disponibilidad	La aplicación debe estar disponible y funcionar correctamente 24 x 7 x 365.
RQ13	No	Performance	La aplicación debe tener una respuesta rápida, eficiente y de alta disponibilidad.
RQ14	No	Compatibilidad	La aplicación debe ser compatible con toda clase de dispositivos.
RQ15	No	Seguridad	La aplicación debe ser segura con controles de acceso.

Tabla 2 – Tabla de Requerimientos Funcionales y No Funcionales (Fuente: Elaboración propia)

4.4 Supuestos y limitaciones del alcance

En la Tabla 1 contiene una definición y descripción de las causas incluidas en la Figura 1 que deberán ser resueltas mediante las diferentes versiones de la utilización de la aplicación. No todos los aspectos descritos en la Tabla 1 alcanzarán a ser cubiertos en esta primera versión dado el poco tiempo para su desarrollo, los aspectos que serán cubiertos en esta primera versión son aquellos descritos en el capítulo “3.1 Alcance del producto” y lo que no se encuentre contenido en este capítulo queda fuera del alcance de esta primera versión.

Adicionalmente esta aplicación solo y solo obtendrá las métricas necesarias de los servidores de origen para evaluar su estado de salud y no tendrá intercambio de información con las aplicaciones residentes en estos servidores.

4.5 Análisis de brechas

Dado el análisis y la presentación de las causas que se encuentran descritas en la Tabla 1 y que originan el presente proyecto de monitoreo y que serán cubiertas en esta primera versión y se encuentran definidos en la Tabla 3 presentada a continuación con lo cual se generaran a posterior todos los requerimientos y/o las historias de usuario utilizando la metodología de proyecto que será definida a posterior.

Causa	AS IS	TO BE	Brecha
4C1	Falta de análisis de métricas en la utilización de recursos de espacio.	Analizar las métricas de los servidores de los recursos de espacio y calcular los umbrales de disponibilidad.	Proceso de análisis de umbrales para cada una de las métricas de espacio de los recursos de discos.
4C3	Falta de análisis de métricas en la utilización de recursos de Memoria y Swap.	Analizar métricas de disponibilidad de recursos de memoria y swap.	Proceso de análisis de umbrales para cada una de las métricas de los recursos de memoria y swap.

Tabla 3 – Análisis de brechas (Fuente: Elaboración propia)

5. Objetivos del proyecto

5.1 Objetivo General

El objetivo general es crear o generar una plataforma de bajo costo y accesible y que nos permita visualizar, analizar por intermedios de umbrales el estado de salud de diferentes servidores. Esta plataforma se encuentra definida en primera instancia para todas aquellas empresas pequeñas y/o medianas que no pueden invertir grandes sumas de dinero en sistemas de monitoreo.

5.2 Objetivos específicos

- **OE1:** Gestionar controles de acceso para dar un uso correcto a la aplicación.
- **OE2:** Gestionar los sistemas operativos que tendrán que ser monitoreados.
- **OE3:** Gestionar los scripts los cuales recolectaran y registrarán todas aquellas métricas obtenidas desde los servidores de origen.
- **OE4:** Gestionar los servidores y umbrales a utilizar en las mediciones de estado de salud de los servidores.
- **OE5:** Diseñar la visualización de las métricas obtenidas a partir de la ejecución del objetivo específico OE4.

5.3 Métricas de los objetivos

A continuación, definiremos las métricas con las cuales se podrá medir los cumplimientos de cada uno de los objetivos presentados en el punto “4.2 Objetivos específicos” y para registrar estas métricas utilizaremos la Tabla 4.

Objetivo Específico	ID Métrica	Nombre de la métrica	Descripción de la métrica
OE1	M-OE1-1	Control de acceso	Separar la visualización de métricas con respecto a la administración de configuración.
OE2	M-OE2-1	Configuración Sistemas Operativos	Se busca implementar la administración de los sistemas operativos a monitorear.
OE3	M-OE3-1	Configuración de scripts de monitoreo	Se busca implementar la administración de los scripts que serán utilizados en los monitoreos.
OE4	M-OE4-1	Configuración de servidores	Se busca implementar la administración de los servidores que serán monitoreados.

Tabla 4 – Métricas de objetivos específicos (Fuente: Elaboración propia)

6. Propuesta de Solución

La propuesta de solución consiste en una aplicación web, la cual cuenta con un formulario con el cual se podrán visualizar las métricas obtenidas desde los distintos servidores monitoreados como se puede visualizar en la Figura 2 que se muestra a continuación.

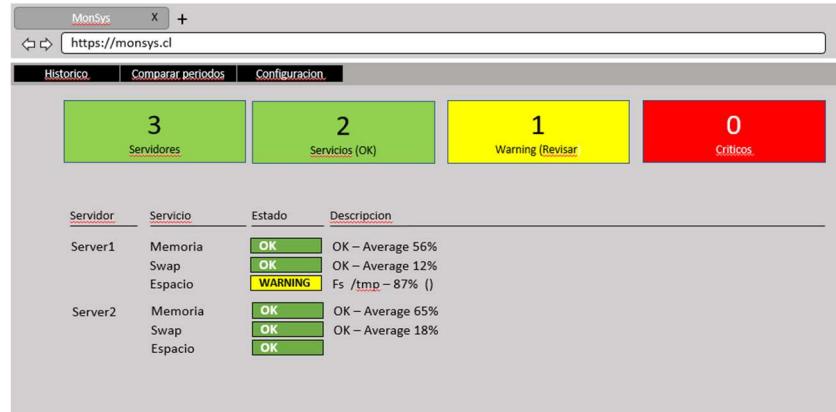


Figura 2 – Pagina web de visualización de métricas de monitoreo (Fuente: Elaboración propia)

Para una mejor compresión de la solución propuesta en la Figura 3 se detalla en un Diagrama de Contexto de alto nivel las interacciones del sistema.

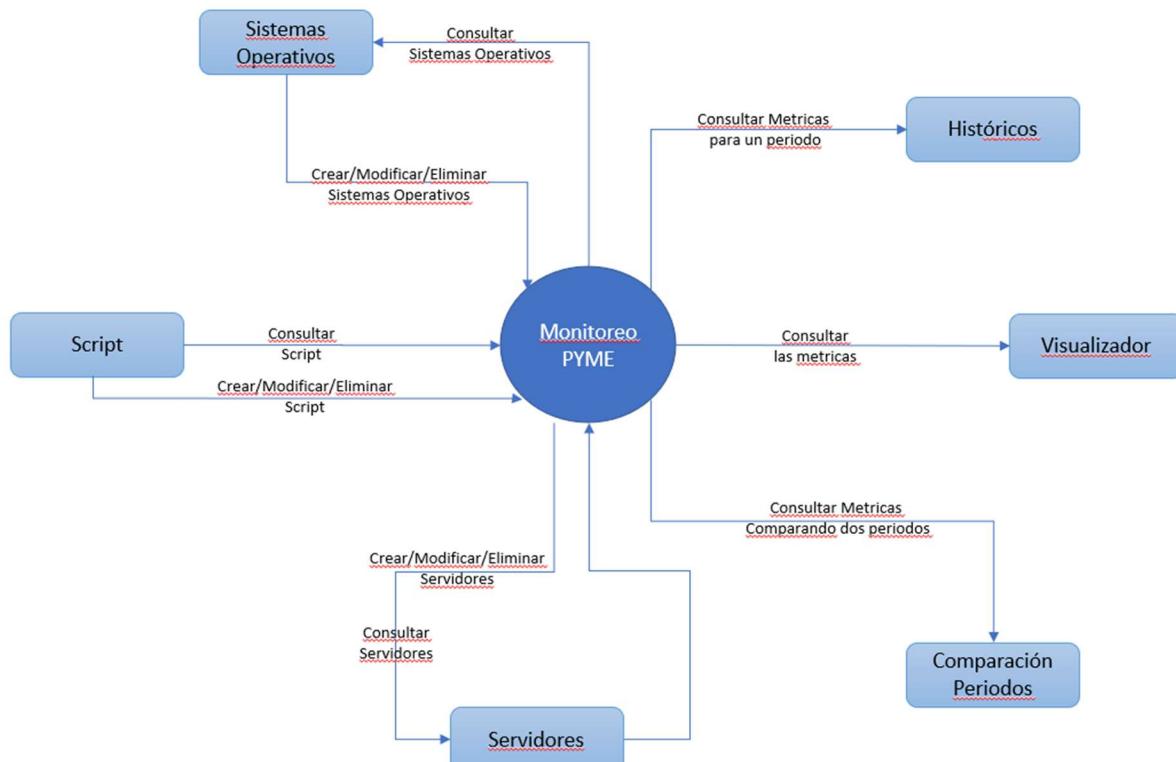


Figura 3 – Diagrama de Contexto de alto nivel (Fuente: Elaboración propia)

A partir del Diagrama de Contexto de Alto Nivel presentado en la Figura 3 se muestra a continuación en la Tabla 5 el mapeo de entradas, salidas y procesos con el cual el sistema de monitoreo debiera funcionar.

Entrada	Salida	Proceso
Crear/modificar/eliminar configuraciones	Administrar las configuraciones con la cual el sistema funciona.	Gestionar registrando las configuraciones requeridas para el correcto funcionamiento
Consultar Métricas	Visualización de las métricas registradas por el proceso de captura de información.	-

Tabla 5 – Tabla de Control (Fuente: Elaboración propia)

7. Plan de Proyecto

7.1 Metodología de proyecto y metodología de desarrollo de software

El desarrollo del proyecto se utilizará una metodología ágil, en este caso definimos la utilización de Scrum como “marco de trabajo” ⁴ como así lo cita la esta metodología. Los puntos por los cuales se definió scrum son los siguientes:

- El tiempo estimado para el desarrollo del producto es de 12 semanas tomando como inicio la semana del trimestre en curso.
- El tiempo para cada mínimo producto viable (MPV) no superara las 4 semanas, estimando 10 horas semanales dando como resultado que cada sprint no puede superar las 40 horas mensuales de trabajo efectivo.
- Se definen 3 sprint como máximo para la construcción del mínimo producto viable (MPV).
- Se realizarán entregas parciales por cada uno del sprint finalizado.
- Al finalizar el tercer sprint será la entrega del mínimo producto viable (MPV) ante la comisión de evaluación.

El desarrollo de software será de tipo iterativo incremental que es la metodología que estaremos utilizando y para lo cual adjuntamos la Figura 4 que nos muestra grafica el proceso a seguir.

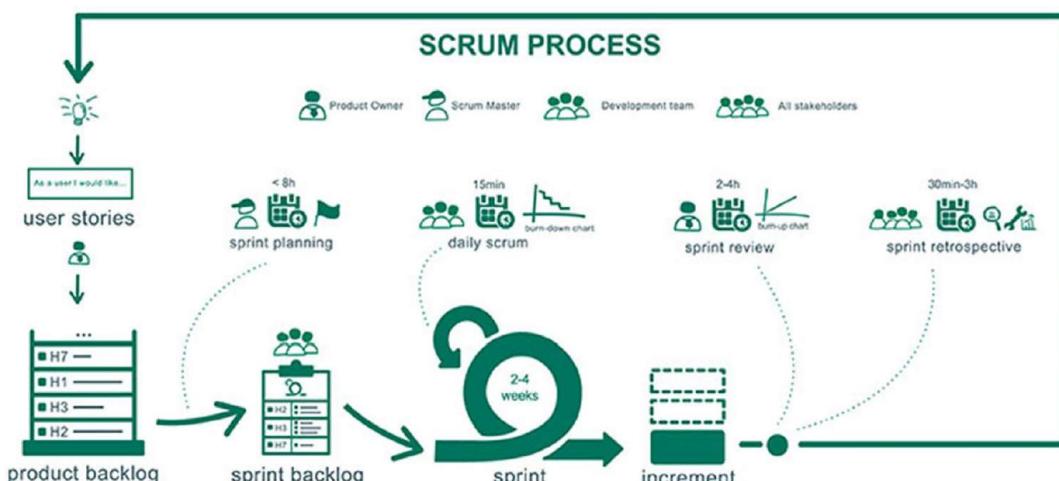


Figura 4 – Diagrama de Proceso Scrum (Fuente: www.scrum.org)

⁴ (Scrum, n.d.)

7.2 Plan de monitoreo y Control

En base a la metodología que definimos y que utilizaríamos en el punto anterior como “marco de trabajo” y corresponde a la metodología ágil como Scrum tiene una de las actividades principales que es monitorear el progreso del proyecto y verificar si el equipo está cumpliendo con sus objetivos. Para ello se basa en las “cinco ceremonias o eventos” de scrum y el principal objetivo de estas ceremonias es mantener los mínimos necesarios para facilitar el control empírico de funcionamiento de los procesos.

Estas ceremonias o eventos son los siguientes:

- Sprint Planning

Corresponde a la reunión que se realiza al comienzo de cada Sprint donde participa todo el equipo Scrum y sirve para inspeccionar el Product Backlog y que el equipo de desarrollo seleccione los Product Backlog Items en los que va a trabajar durante el siguiente Sprint. Adicionalmente estos Product Backlog Items son los que compondrán el Sprint Backlog.

- Daily Scrum

También conocido comúnmente sólo como “La Daily” y se ejecuta principalmente en las mañanas y corresponde a una reunión diaria de 15 minutos como máximo y en la que participa exclusivamente el equipo de desarrollo y se responden a las preguntas como:

¿Qué hice ayer para contribuir al Sprint Goal?

¿Qué voy a hacer hoy para contribuir al Sprint Goal?

¿Tengo algún impedimento que me impida entregar?

- Sprint Review

Es la reunión que ocurre al final de cada uno de los Sprint, generalmente el último viernes del Sprint, donde el product owner y el equipo de desarrollo y presentan a los stakeholders el incremento terminado para su inspección y adaptación correspondientes. En esta reunión organizada por el product owner se estudia cuál es la situación y se actualiza el Product Backlog con las nuevas condiciones que puedan afectar al proyecto.

- Sprint Retrospective

La retrospectiva ocurre al final de un Sprint, justo después del Sprint Review. El objetivo de la retrospectiva es hacer de reflexión sobre el último Sprint e identificar posibles mejoras para el próximo Sprint.

- Sprint Grooming o Refinement

El refinamiento del Product Backlog es una práctica recomendada para asegurar que éste siempre esté preparado. Esta ceremonia sigue un patrón similar al resto y tiene una agenda fija específica en cada Sprint. Se estima su duración en 2 horas máximo por semana del Sprint. Es responsabilidad del product owner agendar, gestionar y dirigir esta reunión.

Cabe señalar que durante la ejecución de un Sprint se pueden generar alguno de los siguientes eventos:

- Se podrían eliminar historias de usuario que sean poco o nada relevantes para el proyecto.
- Agregar nuevas historias de usuario basado en nuevas necesidades y que puedan ser un aporte al proyecto.
- En base a los avances en el sprint se podría modificar la prioridad de una Historia de Usuario.
- Cambiar o corregir las estimaciones de tiempo
- Ejecutar una reevaluación del backlog.

7.3 Plan de Gestión de Riesgos

Dentro del marco de la gestión de proyectos, la gestión del riesgo es uno de los aspectos claves y consiste en identificar y analizar los problemas a los cuales este proyecto podría enfrentarse durante su desarrollo para así poder evaluar los daños que supondría y poder diseñar una estrategia para prevenirlas y mitigarlas. Por lo tanto, es esencial conocer a la perfección el proyecto, así como en el entorno en el cual opera y los factores internos que afectan su funcionamiento y esto lo conseguiremos elaborando una matriz de “consecuencia/probabilidad” como la Tabla 6 que se encuentra adjunta.

Matriz de Criticidad		IMPACTO				
		Despreciable	Bajo	Medio	Alto	Crítico
PROBABILIDAD	86 - 100 %	5	10	15	20	25
	76 - 85 %	4	8	12	16	20
	51 - 75 %	3	6	9	12	15
	26 - 50 %	2	4	6	8	10
	0 - 25 %	1	2	3	4	5

Tabla 6 – Matriz de consecuencia/probabilidad (Fuente: IPLACEX Tecnológico Nacional)

Los riesgos identificados bajo la metodología anteriormente señalada se describen a continuación en la Tabla 7 que se detalla a continuación.

Id Riesgo	Descripción del Riesgo	Categoría	Probabilidad	Impacto	Magnitud	Riesgo Qualitativo
R01	No lograr disponibilizar el Tenant Free	Proyecto	3	3	9	Medio
R02	No poder desplegar la infraestructura requerida	Proyecto	2	5	10	Medio
R03	No disponer de cuotas de recursos para su despliegue	Proyecto	4	5	20	Alto
R04	No contar con la versión de base de datos requerida	Proyecto	2	2	4	Bajo
R05	Conocimientos de GitHub	Técnico	3	3	9	Medio
R06	Conocimientos de Korn Shell Scripting	Técnico	3	3	9	Medio
R07	Conocimientos de Ansible	Técnico	3	3	9	Medio
R08	Conocimientos de Python	Técnico	3	3	9	Medio
R09	Perdida de enlace local	Proyecto	4	5	20	Alto
R10	No cumplir con las evaluaciones de Testing	Proyecto	2	3	6	Medio
R11	No cumplir objetivos de alcance a servidores a ser parte del monitoreo	Técnico	2	2	4	Medio
R12	No cumplir con horas de trabajo definidas en los sprint	Proyecto	3	5	15	Alto

Tabla 7 – Matriz de Riesgo (Fuente: Elaboración propia)

7.4 Plan de mitigación de riesgos

Id Riesgo	Acciones preventivas/correctivas	Ocurrencia	Accion por Realizar	Riesgo Residual
R01	Intentar nuevamente o llamar a soporte de nube para poder crear el tenant free	Al inicio del primer Sprint	Mitigar	Alto
R02	Solicitar la disponibilidad de los componentes	Al inicio del primer Sprint	Mitigar	Alto
R03	Revisar los services Limits para disponibilizar los componentes	Al inicio del primer Sprint	Mitigar	Medio
R04	Verificar otra version compatible.	Al inicio del primer Sprint	Mitigar	Bajo
R05	Estudiar con un curso oficial o alternativos o videos en internet.	En todo el Proyecto	Mitigar	Medio
R06	Estudiar con un curso oficial o alternativos o videos en internet.	En todo el Proyecto	Mitigar	Medio
R07	Estudiar con un curso oficial o alternativos o videos en internet.	En todo el Proyecto	Mitigar	Medio
R08	Estudiar con un curso oficial o alternativos o videos en internet.	En todo el Proyecto	Mitigar	Medio
R09	Utilizar otro medio de enlace, contar con internet inalambrica (BAM).	En todo el Proyecto	Mitigar	Alto
R10	Revisar la definicion de la funcionalidad.	En todo el Proyecto	Mitigar	Medio
R11	Revisar configuracion de infraestructura	En todo el Proyecto	Mitigar	Medio
R12	Recuperacion de horas fuera de la definicion del Sprint	En todo el Proyecto	Aceptar	Alto

Tabla 8 – Plan de mitigación de riesgos (Fuente: Elaboración propia)

8. Materiales, métodos y herramientas

8.1 Patrones de diseño y arquitectura

Nuestro marco y el uso de metodologías de desarrollo ágiles ha cambiado la forma en que construimos software, de un enfoque rígido y lento a un proceso flexible y eficiente. En este marco, se propone un conjunto de mejores prácticas para analizar, diseñar y desarrollar proyectos de software de alta calidad en menos de 60 días, aplicado en un entorno medio limitado. personal o personal temporal.

Rad y Scrum son marcos ágiles que proponen un conjunto de prácticas y roles para desarrollar software de alta calidad, lograr los resultados deseados, brindar flexibilidad y adaptabilidad, lograr el retorno de la inversión, aumentar la productividad, entregar software efectivo con frecuencia y más.

En definición se presentan los dominios de los cuales deberán hacerse cargo las historias de usuarios que se presentarán más adelante.

- Definir los sistemas operativos a monitorear.
- Definir los scripts que serán utilizados para monitorear.
- Definir los servidores que serán monitoreados.
- Diseñar la extracción y registro de métricas de estados de salud de servidores monitoreados.
- Diseñar la visualización de las métricas obtenidas.

(ResearchGate, n.d.)

8.2 Arquitectura de Alto Nivel

Una de las fases principales es definir la arquitectura que será utilizada en el desarrollo del proyecto en la figura 5 se encuentra gráficamente la arquitectura en Alto Nivel que desplegaremos para el desarrollo y pruebas de nuestro proyecto.

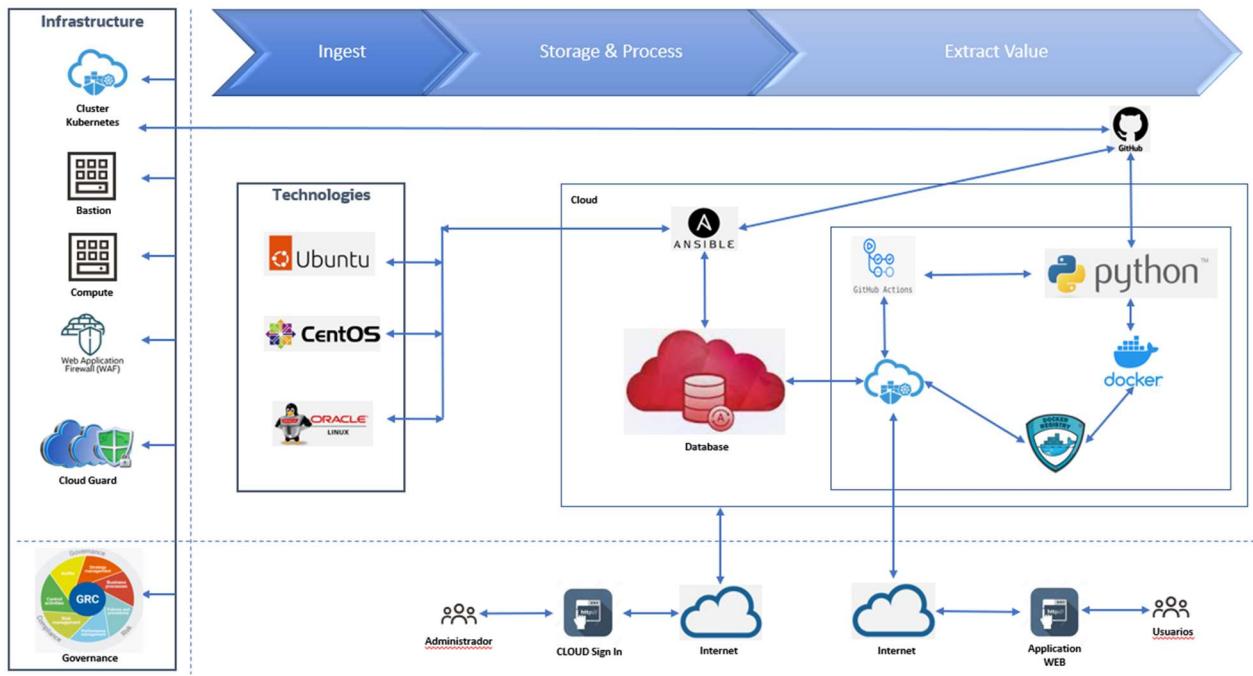


Figura 5 – Arquitectura de Alto Nivel (Fuente: Elaboración propia)

8.3 Herramientas de software

Para el desarrollo de nuestro proyecto utilizaremos las siguientes herramientas las cuales se encuentran detalladas en la siguiente Tabla 9.

Tecnología	Descripción / Función
 Cluster Kubernetes	Container Engine for Kubernetes es un servicio de Kubernetes gestionado para operar aplicaciones en contenedores a escala, al tiempo que se reduce el tiempo, el costo y la carga operativa.
 Bastión	Servidor de operaciones que es utilizado para comunicaciones entre distintas VCN (Virtual Connect Network) y adicionalmente On-Premise.
 Compute	Sera utilizado para albergar la instancia de Base de Datos.
 Ansible	Para ejecutar procesos repetitivos de extracción de información de los servidores a administrar y monitorear.
 Docker Registry	Almacenamiento de las imágenes de los contenedores de Kubernetes.
 Python	Lenguaje de Desarrollo el cual será utilizado para crear el front end de nuestro proyecto.
 Visual Code	Es un editor de código fuente que permite trabajar con diversos lenguajes de programación, admite gestionar tus propios atajos de teclado y refactorizar el código. Adicionalmente será integrado con GitHub.
 Git Hub	Sera utilizado como repositorio de versiones y diferentes reléase del desarrollo del proyecto.
 Git Hub Action	Plataforma de integración y despliegue continuos (IC/DC) que te permite automatizar mapas de compilación, pruebas y despliegue. Crear flujos de trabajo y crear y probar cada solicitud de cambios en tu repositorio o desplegar solicitudes de cambios fusionadas a producción.

Tabla 9 – Matriz de riesgos del proceso de construcción del proyecto de monitoreo (Fuente: Elaboración propia)

(Kubernetes, n.d.)

(Ansible, n.d.)

(GitHub, n.d.)

(Action, n.d.)

8.4 Plan de gestión de la configuración

La gestión de versiones y desarrollo del proyecto utilizaremos GitHub como administrador de versiones principal donde almacenaremos los avances del desarrollo del proyecto y sus diferentes versiones y reléase teniendo en consideración las generaciones de tag para generar las líneas base a medida de su avance. Adicionalmente a GitHub utilizaremos Git como medida de respaldo con almacenamiento local en caso de cualquier contingencia en nuestro repositorio principal que se accede por internet. A continuación, en la Figura 6 presentamos gráficamente la gestión de configuración que estaremos utilizando para el desarrollo de este proyecto.

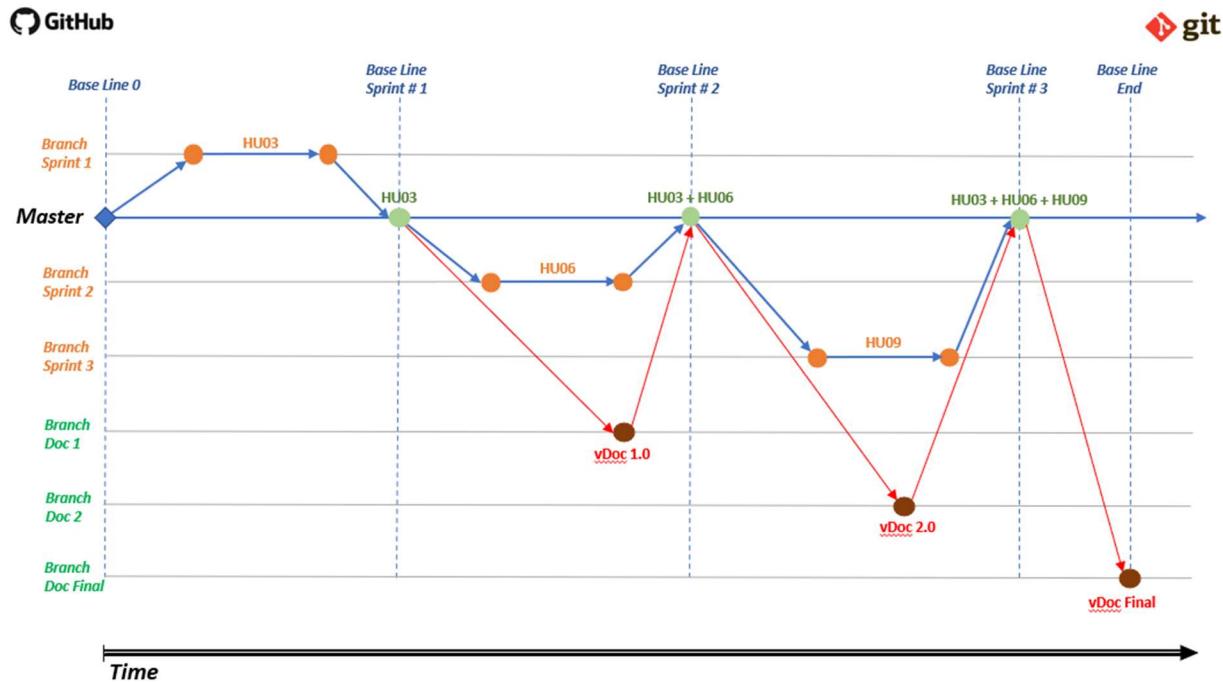


Figura 6 – Diagrama de Gestión de la Configuración (Fuente: Elaboración propia)

9. Gestión del Proyecto

La gestión de nuestro proyecto se enmarca por los siguientes roles que serán los que actuarán en nuestro proyecto.

- **Product Owner** o el propietario del producto es responsable de la gestión eficaz de la cartera de productos, que incluye: Desarrollar y comunicar explícitamente el objetivo del producto y en este caso es el autor del presente documento.
- **Scrum Master** o facilitador de proyectos, es la figura que lidera los equipos en la gestión ágil de proyectos. Su misión es que los equipos de trabajo alcancen sus objetivos hasta llegar a la fase de «sprint final», eliminando cualquier dificultad que puedan encontrar en el camino. El Scrum Master para este proyecto será la señorita Giannina Costa.
- **Scrum Team** es un pequeño equipo de personas que incluye un Scrum Master, un Product Owner y los desarrolladores. En este grupo, no hay sub-equipos ni jerarquías. Los profesionales son multifuncionales y tienen las habilidades necesarias para cumplir el objetivo del producto final. Este equipo se compone por el profesor Matías Vargas Marin como técnico y arquitecto de infraestructura y el autor del presente documento como el equipo de desarrollo.

El producto Backlog se encuentra en la Tabla 10 donde se encuentra detallado.

- El roadmap del proyecto con la trazabilidad basada en las historias de usuarios.
- La prioridad realizada con MosCow⁵.
- Para el calculo del peso utilizaremos T-Shirt⁶
- El peso relativo se estimado cuando priorizamos el Backlog.

HU	Descripcion	Alias	Prioridad	Peso
HU01	Yo como administrador quiero crear usuarios para que puedan administrar la configuración del sistema de monitoreo	Registrar configuración	Debe tener (Mo)	M
HU02	Yo como administrador quiero crear usuarios para que puedan solo visualizar el sistema de monitoreo	Registrar Métricas	Debe tener (Mo)	M
HU03	Yo como administrador, quiero registrar la configuración de los servidores UNIX para monitorear el estado de salud.	Ver Métricas	Debe tener (Mo)	M
HU04	Yo como administrador, quiero registrar la configuración de los servidores Windows para monitorear el estado de salud.	Usuarios	Debería Tener (S)	S
HU05	Yo como administrador, quiero registrar la configuración de los servidores de Bases de Datos para monitorear el estado de salud.	Usuarios	Debería Tener (S)	S
HU06	Yo como administrador quiero registrar las diferentes métricas de servicio de los servidores UNIX para que puedan ser consultadas.	Ver Métricas	Debe tener (Mo)	M
HU07	Yo como administrador quiero registrar las diferentes métricas de servicio de los servidores Windows para que puedan ser consultadas.	Ver Métricas	Debería Tener (S)	S
HU08	Yo como administrador quiero registrar las diferentes métricas de servicio de los servidores de Base de Datos para que puedan ser consultadas.	Registrar Métricas	Debería Tener (S)	S
HU09	Yo como usuario quiero ver dashboard de monitoreo para ver el estado de salud de los servidores.	Ver Métricas	Debe tener (Mo)	M
HU10	Yo como usuario quiero ver dashboard de monitoreo histórico para ver el estado de salud de los servidores en fechas pasadas.	Registrar Métricas	Debería Tener (S)	XS
HU11	Yo como usuario quiero poder comparar periodos de monitoreo para ver el estado actual con fechas predecesoras.	Ver Métricas	Debería Tener (S)	XS

Tabla 10 – Product Backlog (Fuente: Elaboración propia)

⁵ (Moscow, n.d.)

⁶ (Asana, Asana, n.d.)

10. Descripción general de la propuesta de solución

El proyecto “Monsys” surge como iniciativa a una necesidad de un problema crítico de la gestión de administración en sistemas computacionales de las empresas medianas y pequeñas denominadas PYME. La misión de este proyecto es poder ofrecer una solución tecnológica de bajo costo y fácil configuración y administración brindando la ayuda necesaria a poder mantener sus sistemas correctamente administrados.

Este sistema está diseñado para brindar a las empresas un control total sobre sus operaciones y activos, garantizando la seguridad y eficiencia en un entorno empresarial en constante cambio.

10.1 Características clave de la propuesta de solución

- **Facilidad de Configuración:** Nuestro sistema ha sido diseñado con una interfaz intuitiva y amigable que permite a los usuarios configurar y personalizar la plataforma según sus necesidades únicas, sin requerir conocimientos técnicos avanzados.
- **Monitoreo en Tiempo Real:** Proporcionamos una visión en tiempo real de los activos, operaciones y procesos críticos de la empresa. Esto permite tomar decisiones informadas y oportunas en función de la información más reciente.
- **Acceso Remoto:** El sistema permite el acceso remoto, lo que brinda la flexibilidad de monitorear las operaciones incluso cuando no se está en el lugar físico.
- **Economía de Costos:** Reconocemos las limitaciones presupuestarias de las pequeñas y medianas empresas. Por lo tanto, nuestro sistema está diseñado para ser asequible en términos de costo inicial y mantenimiento continuo y con una arquitectura de bajo costo de implementación.
- **Escalabilidad:** A medida que la empresa crece, nuestro sistema es escalable. Puede integrar fácilmente más sensores y funcionalidades adicionales según sea necesario.

En resumen, nuestro sistema de monitoreo proporciona una solución integral para empresas pequeñas y medianas que buscan un control efectivo sobre sus operaciones y activos a un costo accesible. La facilidad de configuración, las características de tiempo real y las notificaciones proactivas aseguran que la empresa esté siempre informada y preparada para tomar decisiones estratégicas y operativas fundamentadas.

10.2 Organización del desarrollo del producto

Los diagramas de desarrollo UML son representaciones gráficas que permiten visualizar y comunicar diferentes aspectos del proceso de desarrollo de software que estaremos utilizando tal como se observa en la figura 7.

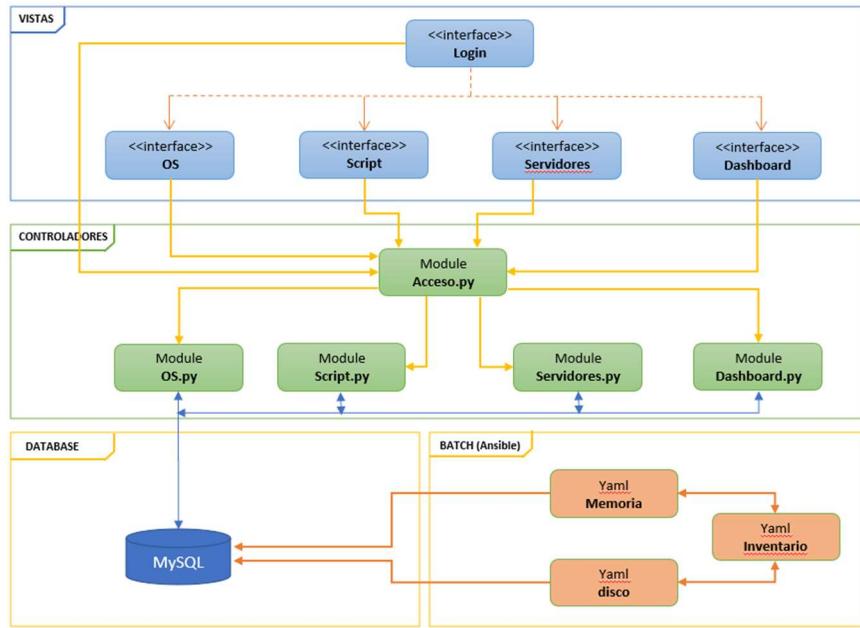


Figura 7 – Diagrama de Desarrollo (Fuente: Elaboración propia)

A continuación, se encuentra nuestro mapa de despliegue que se utiliza para representar la arquitectura física de un sistema y la distribución de los componentes de software en diferentes nodos o entornos de ejecución como se muestra en la figura 8.

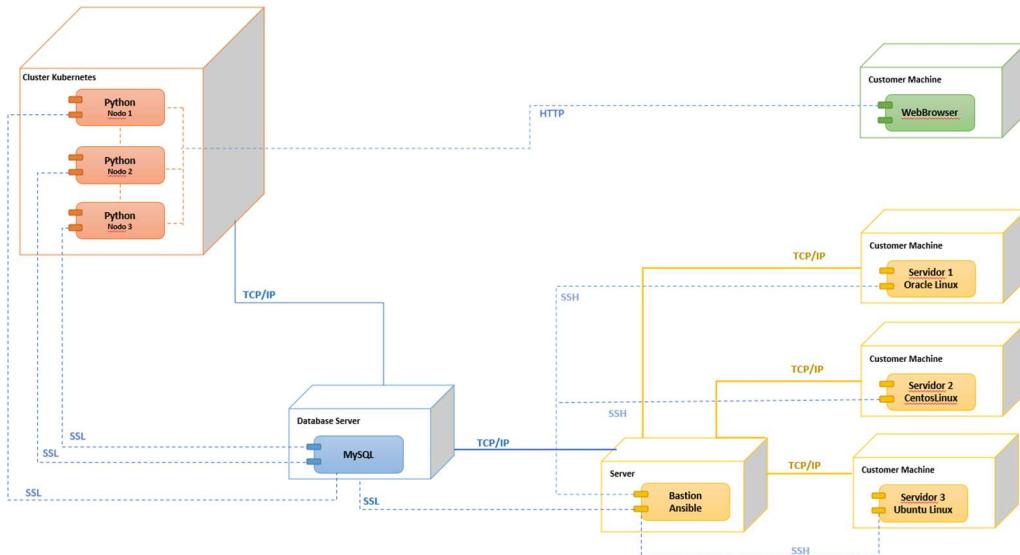


Figura 8 – Diagrama de Despliegue (Fuente: Elaboración propia)

10.3 Arquitectura del Software a Implementar

La arquitectura que utilizaremos para el desarrollo de nuestro proyecto este compuesto por distintos componentes los cuales fueron definidos al inicio de esta planificación, pero estos han sufrido algunos cambios y podrían ser reemplazados según como avancemos en los sprint. Estos cambios solo podrían ser producto a que no cumplan con lo deseado o la dificultad de implementación y serían reemplazados por algunos más sencillos y que presten las funcionalidades requeridas.

- **Backend** donde utilizaremos FLASK que es un marco micro web escrito en Python. Se clasifica como un micro framework porque no requiere herramientas o bibliotecas particulares. No tiene capa de abstracción de base de datos, validación de formularios ni ningún otro componente en el que las bibliotecas de terceros preexistentes proporcionen funciones comunes. Este despliegue será utilizado en el Sprint 3 el cual aún no ha comenzado por lo cual aún no sabemos ciertamente la versión que será utilizada.
- **Frontend** corresponde al desarrollo de elementos visuales e interactivos de un sitio web con los que los usuarios interactúan directamente. Es una combinación de HTML, CSS, donde HTML proporciona la estructura, CSS el estilo y el diseño.
- **Kubernetes** es un sistema de orquestación de contenedores de código abierto para automatizar la implementación, el escalado y la administración de software.
- **GitHub** es una plataforma y un servicio basado en la nube para el desarrollo de software y el control de versiones mediante Git, que permite a los desarrolladores almacenar y administrar su código.
- **MySQL** es un sistema de gestión de bases de datos relacionales de código abierto. Su nombre es una combinación de "My", el nombre de My y "SQL", el acrónimo de Structured Query Language.
- **Pruebas unitarias y de integración** serán ejecutadas a medida que avance cada uno de los sprint y cuando sean requeridas.
- **Gestión de riesgos:** Se identifican 12 riesgos en total (3 Alto, 6 Medio y 3 Bajos) de los cuales 8 de ellos ya han sido completamente mitigados en la ejecución del Sprint 1.
- **Historias de usuario:** Cumpliendo con los procedimientos de la metodología ágil que estamos utilizando que es Scrum se definen las Historias de Usuario (HU) para lo cual en este proyecto se definieron 11 de ellas.

10.4 Modelo de Base de Datos

La instancia de Base de Datos que se decidió utilizar corresponde a MySQL v 8.0 ya que esta base de datos corresponde a un código abierto y no incurre en costos de licenciamiento. Para dar soporte a nuestro desarrollo se definió un modelo de datos que puede escalar de forma sencilla e integrarse al modelo previamente establecido como se puede ver en la figura 9.

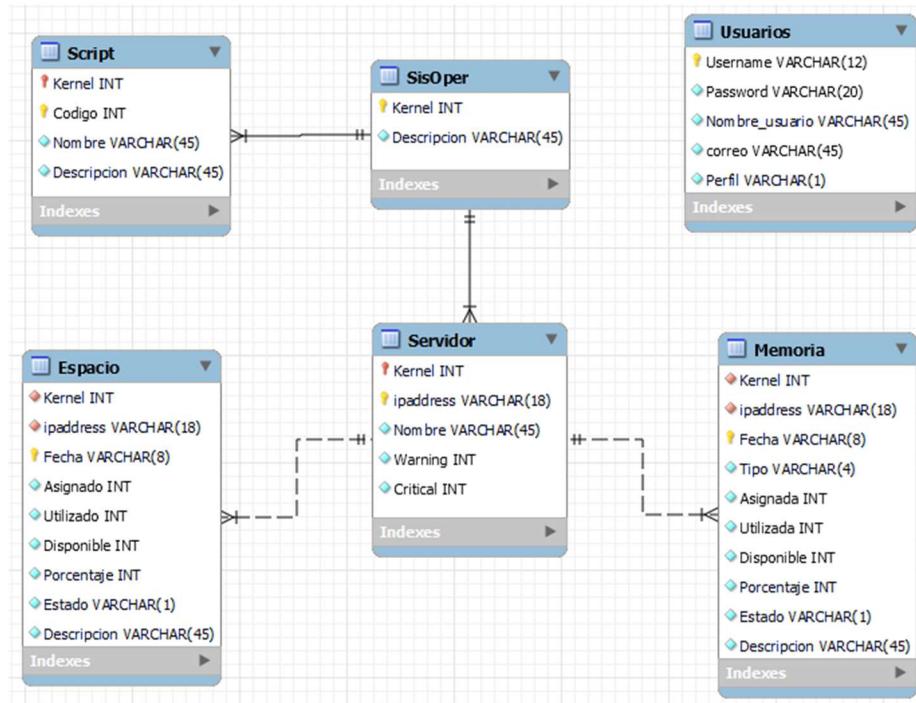


Figura 9 – Modelo de Datos (Fuente: MySQL Workbench)

11. Control y Seguimiento

El control de seguimiento de nuestro proyecto será realizado mediante el uso de los informes asociados con la metodología ágil Scrum para lo cual se presentarán los reportes de Sprint Review, Sprint Retrospective al final de cada uno de los sprint que se estén presentando en este documento.

11.1 Sprint 1

El proyecto para desarrollar como Mínimo Producto Viable (MPV) en el Sprint 1 tiene como meta realizar el despliegue de todos los recursos necesarios de infraestructura y una duración de 4 semanas lo cual hace factible el cumplimiento de lo que se encuentra planificado.

A continuación, en la Figura 10 detallamos las actividades desarrolladas para el Sprint Backlog 1.

Elemento de trabajo pendiente	Puntos de historia	Responsable	Estado	Estimado Original	Sprint Backlog #01														
					Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7	Día 8	Día 9	Día 10	Día 11	Día 12	Día 13	Día 14	Día 15
HU03	3																		
Analizar HU y sus riesgos	P.Valenzuela	Done		2	2	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Creacion de Tenant en Cloud (R1)	P.Valenzuela	Done		2	2	2	-	-	-	-	-	-	-	-	-	-	-	-	-
Creacion de Infraestructura base	P.Valenzuela	Done		2	2	2	2	-	-	-	-	-	-	-	-	-	-	-	-
Creacion de Cluster Kubernetes (R2) (R3)	P.Valenzuela	Done		4	4	4	4	4	-	-	-	-	-	-	-	-	-	-	-
Instalacion de podman	P.Valenzuela	Done		1	1	1	1	1	1	-	-	-	-	-	-	-	-	-	-
Creacion Repositorio y configuracion de GitHub (R5)	P.Valenzuela	Done		1	1	1	1	1	1	1	-	-	-	-	-	-	-	-	-
Creacion de Compute para Base de Datos	P.Valenzuela	Done		1	1	1	1	1	1	1	1	-	-	-	-	-	-	-	-
Descarga, Instalacion, Configuracion Ansible	P.Valenzuela	Done		1	1	1	1	1	1	1	1	-	-	-	-	-	-	-	-
Descarga, Instalacion, Configuracion MySQL (R4)	P.Valenzuela	Done		1	1	1	1	1	1	1	1	-	-	-	-	-	-	-	-
Creacion de Base de Datos MySQL	P.Valenzuela	Done		1	1	1	1	1	1	1	1	1	-	-	-	-	-	-	-
Creacion de Modelo de Base de Datos	P.Valenzuela	Done		2	2	2	2	2	2	2	2	2	2	2	2	-	-	-	-
Poblar Base de Datos con Datos de Prueba	P.Valenzuela	Done		2	2	2	2	2	2	2	2	2	2	2	2	2	-	-	-
Creacion de Registry	P.Valenzuela	Done		2	2	2	2	2	2	2	2	2	2	2	2	2	-	-	-
Crear secrets en Kubernetes para connect a Registry	P.Valenzuela	Done		1	1	1	1	1	1	1	1	1	1	1	1	1	-	-	-
Creacion de 3 VCN con distintos segmentos	P.Valenzuela	Done		1	1	1	1	1	1	1	1	1	1	1	1	1	1	-	-
Configuracion de LocalPeering	P.Valenzuela	Done		1	1	1	1	1	1	1	1	1	1	1	1	1	1	-	-
Creacion de 3 Compute con distinto OS Linux	P.Valenzuela	Done		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-
Pruebas de Integracion	P.Valenzuela	Done		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-
Total					27	27	25	23	21	17	16	15	14	12	11	10	8	6	4
																			-

Figura 10 – Sprint Backlog 1 (Fuente: Elaboración propia)

Adicionalmente adjuntamos el Burndown Chart en la Figura 11 del Sprint Backlog 1.

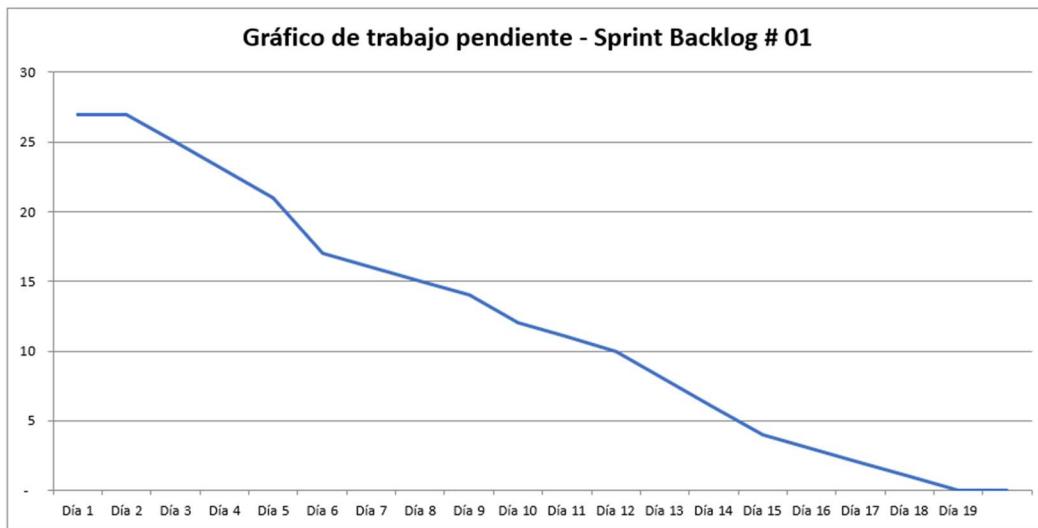


Figura 11 – Burndown Chart Sprint 1 (Fuente: Elaboración propia)

11.1.1 Riesgos Sprint 1

En este Sprint 1 no se gatillo ningún riesgo los cuales fueron identificados en el capítulo 7.3 Matriz de Riesgo y esto nos ayudó a finalizar el Sprint 1 antes de lo planificado ganando así 2 días para dar comienzo al sprint 2. A continuación, en la figura 12 se puede apreciar la reevaluación de los riesgos asociados al Sprint 1.

Id Riesgo	Descripción del Riesgo	Categoría	Probabilidad	Impacto	Magnitud	Riesgo Cualitativo
R01	No lograr disponibilizar el Tenant Free	Proyecto	1	1	1	Bajo
R02	No poder desplegar la infraestructura requerida	Proyecto	1	1	1	Bajo
R03	No disponer de cuotas de recursos para su despliegue	Proyecto	1	1	1	Bajo
R04	No contar con la versión de base de datos requerida	Proyecto	1	1	1	Bajo
R05	Conocimientos de GitHub	Técnico	2	2	4	Bajo
R07	Conocimientos de Ansible	Técnico	1	1	1	Bajo
R09	Perdida de enlace local	Proyecto	4	5	20	Alto
R10	No cumplir con las evaluaciones de Testing	Proyecto	1	1	1	Bajo

Figura 12 – Matriz de Riesgo reevaluada (Fuente: Elaboración propia)

11.1.2 Matriz de trazabilidad de requerimientos Sprint 1

A continuación, la tabla 11 nos muestra la matriz de trazabilidad de los requerimientos que se encuentran definidos en capítulo 4 y su relación con las Historias de Usuarios que se encuentra en el Capítulo 9 en relación con la ejecución de las tareas definidas en el Sprint 1.

HU	Requerimiento	Sprint	Requerimientos	Estado
HU03	Yo como administrador, quiero registrar la configuración de los servidores UNIX para monitorear el estado de salud.	Sprint 1	RQ3, RQ7, RQ12, RQ13, RQ14	Finalizado

Tabla 11 – Matriz de Trazabilidad de requerimientos Sprint 1 (Fuente: Elaboración propia)

11.1.3 Propuesta de Controles y Evidencia

Para este sprint se ejecutó dentro de los tiempos planificados con las correspondientes evidencias unitarias las cuales se encuentran a continuación. Adicionalmente existen evidencias de las instalaciones, configuraciones de los diferentes componentes, evidencias de las pruebas ejecutadas en GitHub en la URL adjunta.

- <https://github.com/pvalenzuelabaeza/licenciatura/tree/Sprint-1>

- Creación de Tenant en Cloud:** Se establece que una vez finalizada la creación del tenant se deberá evidenciar ingresando al tenant y validar el correcto acceso a la consola inicial del tenant. Como este es un proyecto solo de alcance académico y no comercial utilizaremos un tenant que actualmente estoy utilizando para las creaciones de demos y pruebas de concepto para los clientes de la organización en la cual trabajo tal como se evidencia en la figura 13.

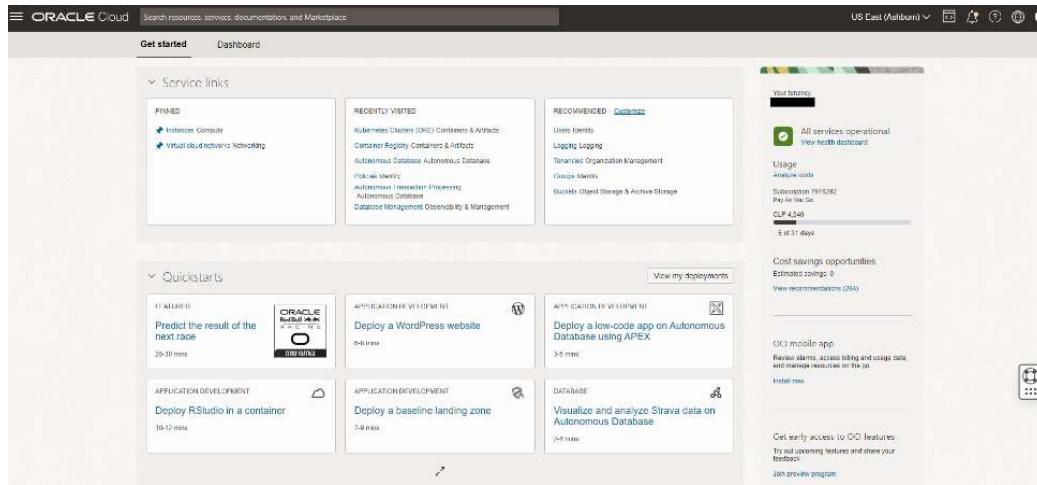


Figura 13 – Consola Oracle Cloud Infrastructure (Fuente: Oracle.com)

- Creación de Clúster Kubernetes:** Se establece que una vez finalizada la creación de la infraestructura de Clúster Kubernetes se deberá evidenciar ingresando al tenant y adjuntar evidencia del Clúster Kubernetes y los nodos asociados al clúster y que este correctamente desplegado tal como se evidencia en las figuras 14 y 15.

Clusters in Patricio.Valenzuela Compartment

Create cluster						
Name	Status	Node pools	VCN	Version	Created	
OKE1	● Active	1	oke-vcn-quick-OKE1-63d742893	✓ v1.26.2	Tue, Jul 11, 2023, 16:13:26 UTC	⋮

Showing 1 item < 1 of 1 >

Figura 14 – Clúster Kubernetes desplegado en consola OCI (Fuente: Consola OCI)

Create instance Table settings									
Name	State	Public IP	Private IP	Shape	OCPU count	Memory (GB)	Availability domain	Fault domain	Created
oke-czki2g5b6f0-n5jcbq6e3la-ses3ucn7oza-2	● Stopped	-	10.0.10.236	VM.Standard.E...	3	32	AD-1	FD-3	Tue, Jul 11, 202...
oke-czki2g5b6f0-n5jcbq6e3la-ses3ucn7oza-1	● Stopped	-	10.0.10.16	VM.Standard.E...	3	32	AD-2	FD-2	Tue, Jul 11, 202...
oke-czki2g5b6f0-n5jcbq6e3la-ses3ucn7oza-0	● Stopped	-	10.0.10.233	VM.Standard.E...	3	32	AD-3	FD-1	Tue, Jul 11, 202...

Figura 15 – Nodos asociados al Clúster Kubernetes (Fuente: Consola OCI)

- **instalación de podman:** Se establece que una vez finalizada la instalación de Podman que es un runtime para administrar los contenedores del Clúster Kubernetes y que se utiliza para ejecutar, crear, etcétera, y puede ser ejecutado por cualquier usuario donde sea instalado y la explicación de porque no utilizamos Dockers que es el runtime por default y más conocido ya que para ejecutarlo se requiere las credenciales de root y esto representa una brecha de seguridad, y para evidenciar que se encuentra instalado correctamente tal como se evidencia en la figura 16.

```
[opc@bastion ~]$ sudo yum install -y podman
Last metadata expiration check: 0:03:37 ago on Fri 21 Jul 2023 03:42:27 PM GMT.
Dependencies resolved.

Package           Arch      Version            Repository      Size
=====
podman           x86_64   3:4.4.1-12.module+el8.0.0+21056+d98a0860  ol8_appstream  15 M
common           x86_64   3:2.1.6-1.module+el8.0.0+21045+adcb6a64    ol8_appstream  57 k
container-selinux noarch   2:2.205.0-2.module+el8.8.0+21045+adcb6a64    ol8_appstream  64 k
containerNetworking-plugins x86_64   1:1.2.0-1.module+el8.8.0+21045+adcb6a64    ol8_appstream  20 M
container-common  x86_64   2:1.64.0.1.module+el8.8.0+21056+d98a0860  ol8_appstream  88 k
crs              x86_64   3.19-3.module+el8.8.0+21045+adcb6a64    ol8_appstream  519 k
fuse-overlayofs x86_64   1.11-1.module+el8.8.0+21056+d98a0860  ol8_appstream  74 k
libenet          x86_64   1.1.6-15.el8               ol8_appstream  67 k
libsskip         x86_64   4.4.0-1.module+el8.8.0+21045+adcb6a64    ol8_appstream  70 k
podman-cataonit x86_64   3:9.4.1-12.module+el8.8.0+21056+d98a0860  ol8_appstream  362 k
runc             x86_64   1:1.1.4-1.0.1.module+el8.8.0+21119+51f68e0d8  ol8_appstream  3.1 M
shadow-utils-subuid x86_64   2:4.6-17.el8               ol8_baseos_latest 119 k
slirp4netns       x86_64   1:2.0-2.module+el8.8.0+21045+adcb6a64    ol8_appstream  54 k
Enabling module streams:
container-tools   ol8

Transaction Summary

Install 13 Packages

Total download size: 39 M
Installed size: 119 M
Downloading Packages:
(1/13): common-2.1.6-1.module+el8.0.0+21045+adcb6a64.x86_64.rpm           1.9 MB/s | 57 kB     00:00
(2/13): container-selinux-2.205.0-2.module+el8.8.0+21045+adcb6a64.noarch.rpm  2.0 MB/s | 64 kB     00:00
```

Figura 16 – Instalación de Podman (Fuente: Local / OnPremise)

- **Creación Repositorio y configuración de GitHub:** Para cumplir con las propuesta y evidencia para este hito se deberá evidenciar con las imágenes del repositorio como se evidencia en la figura 17. Adicionalmente se puede visitar la URL <https://github.com/pvalenzuelabaeza/licenciatura> para visualizarlo, el repositorio fue creado como público y la única finalidad es cumplir con los requisitos presentados en la asignatura.

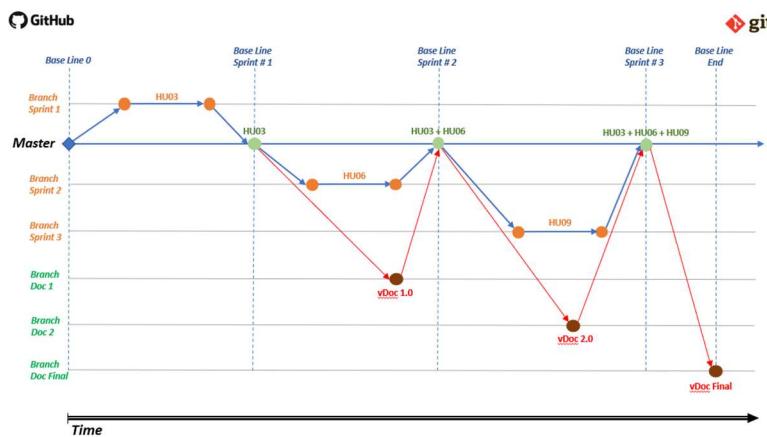
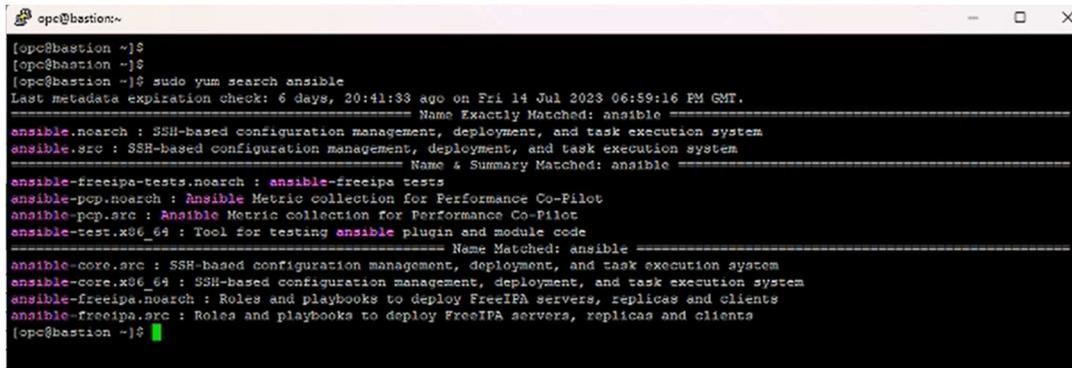


Figura 17 – Repositorio GitHub (Fuente: GitHub.com)

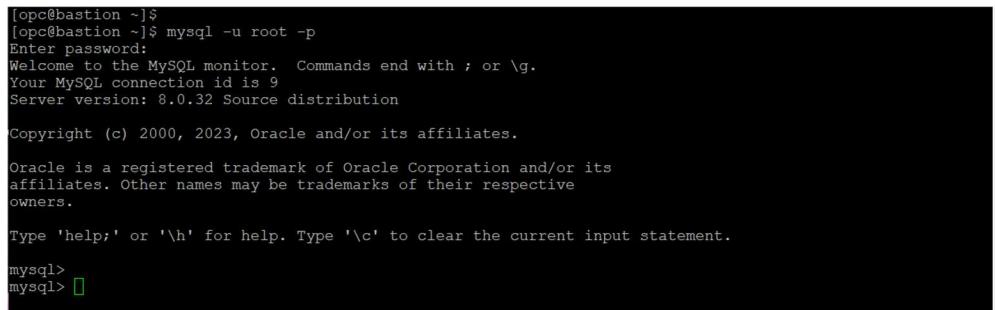
- **Instalación Ansible:** Ansible es un conjunto de herramientas de software que habilita la infraestructura como código. Es de código abierto y la suite incluye aprovisionamiento de software, gestión de configuración y funcionalidad de implementación de aplicaciones y ejecuciones de script. Este será utilizado con la ultima funcionalidad que corresponde a la extracción de las estadísticas de los distintos servidores y las evidencias de su correcta instalación se encuentra en la figura 18.



```
[opc@bastion ~]$ [opc@bastion ~]$ sudo yum search ansible
Last metadata expiration check: 6 days, 20:41:33 ago on Fri 14 Jul 2023 06:59:16 PM GMT.
=====
Name Exactly Matched: ansible
=====
ansible : SSH-based configuration management, deployment, and task execution system
ansible_src : SSH-based configuration management, deployment, and task execution system
=====
Name & Summary Matched: ansible
=====
ansible-freeipa-tests.noarch : ansible-freeipa tests
ansible-pcp.noarch : Ansible Metric collection for Performance Co-Pilot
ansible-pcp.src : Ansible Metric collection for Performance Co-Pilot
ansible-test.x86_64 : Tool for testing ansible plugin and module code
=====
Name Matched: ansible
=====
ansible-core.src : SSH-based configuration management, deployment, and task execution system
ansible-core.x86_64 : SSH-based configuration management, deployment, and task execution system
ansible-freeipa.noarch : Roles and playbooks to deploy FreeIPA servers, replicas and clients
ansible-freeipa.src : Roles and playbooks to deploy FreeIPA servers, replicas and clients
[opc@bastion ~]$
```

Figura 18 – Validación de instalación de Ansible (Fuente: Local)

- **Instalación MySQL:** La instalación de la base de datos se realiza para almacenar las configuraciones y las estadísticas obtenidas desde los distintos servidores que serán monitoreados y se puede observar su correcta instalación y configuración en la figura 19.



```
[opc@bastion ~]$ [opc@bastion ~]$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
mysql> █
```

Figura 19 – Prueba de conexión MySQL (Fuente: Servidor OCI)

- **Creación de Registry:** Contenedor para las imágenes de los Clúster Kubernetes y se puede observar su correcta configuración en la figura 20.

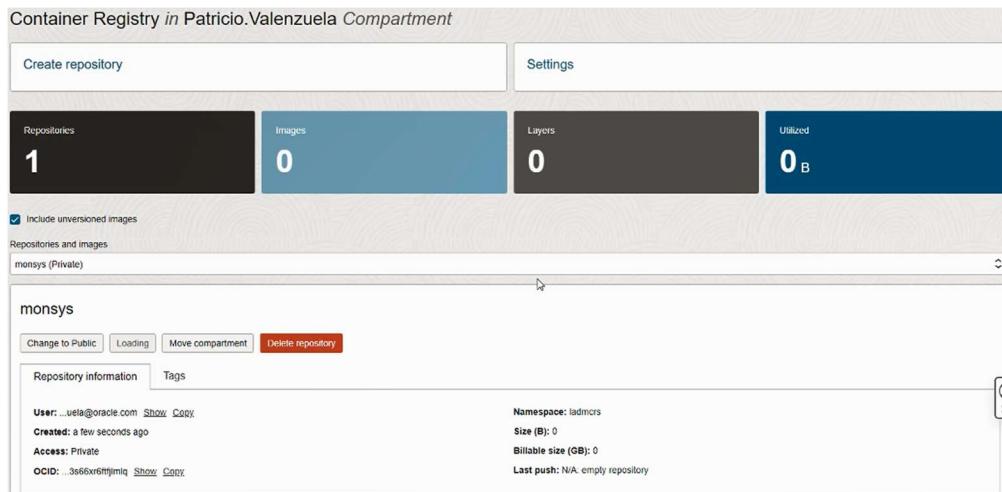


Figura 20 – Registry creado (Fuente: Consola OCI)

- **Secrets Kubernetes para conectar a Registry:** Es utilizado para poder conectar el Clúster Kubernetes con el Registry en OCI, la correcta configuración queda en evidencia en la figura 21.

```
[opz@bastion ~]# kubectl get secret ociregistry -n monsys -o yaml
apiVersion: v1
data:
  .dockerconfigjson: eyJhdXRocI6eyJpIWQub2Npcis5pbyl6eyJlc2VybmcFZS16ImlkGhbzZkcDNiZC9vcmFjBGVpZGVudGl0eNNsb3Vkc2VycmljZS9wYXRsawNpbys2YWxlbnplZWxhQG9yYWNsZS5jb
  30ILCwvXNx2d29yZC16IjBcYlpFTz1XyIsztzF03NE5V1zIwLYXVoC16im5XNnNhr3B2Tm1sd00ySmuMMj15WVdOclpXLGta7zUwYhSNVkyeH2kV1J6RlhKM0Jo2EhKcFkybH2MblpoYkdRdVVuVmxa
  iROZBYjNKAfkyeGxMbU52YlRvd2JKSmFYMDg1Y1Y4cGJGTXhYME56UKVaD013PT0ifX19
kind: Secret
metadata:
  creationTimestamp: "2023-07-21T19:18:06Z"
  name: ociregistry
  namespace: monsys
  resourceVersion: "3907968"
  uid: 3d697996-0588-4059-8e0a-e203b70f028e
type: kubernetes.io/dockerconfigjson
[opz@bastion ~]#
```

Figura 21 – Evidencia de Creación y configuración de Secret (Fuente: Consola OCI)

- **Computo con distribución OS Linux:** Maquinas definidas para efectuarles el monitoreo con las distribuciones que se encuentra enmarcadas en el alcance del proyecto como se muestra en la figura 22.



Figura 22 – Imágenes de sistema operativo utilizados para servidores a monitorear (Fuente: Consola OCI)

11.1.4 Sprint 1 - Review

A continuación, se comparte los comentarios y retroalimentación recibidos en 3 sesiones sincrónicas las cuales podemos resumir en la tabla 12.

Sprint 1 - Review				QUIEN
START	DURACION	ACTIVIDAD	DESCRIPCION	QUIEN
19:00	10 min	Objetivo del sprint	Presentacion de Objetivos del Sprint	Product Owner
			Revision de las metas del Sprint	
			Revision de la duracion del Sprint	
19:10	10 min	Revisión de los principales requisitos	Revision de despliegue de Cluster Kubernetes correctamente	Product Owner
			Revision de despliegue de Base de Datos correctamente	
			Revision de despliegue del resto de componentes correctamente	
19:20	10 min	Status sprint	El plan de ejecucion de las actividades se cumplen en tiempo y forma	Scrum Master
			No se encuentran issues en la ejecucion de las actividades	
			No se gatillaron ninguno de los riesgos establecidos para el sprint	
19:30	10 min	Demostracion	Se realiza una muestra en vivo de todos los recursos desplegados	Team
19:40	10 min	Feedback	Para este Sprint la retroalimentacion recibida es que se ha ejecutado en forma satisfactoria y sin desviaciones del plan original	Scrum Master
19:50	10 min	Cierre	Se revisara en la siguiente sesion el Sprint 2 ya que este finalizo en forma satisfactoria.	Scrum Master

Tabla 12 – Sprint Review (Fuente: Elaboración propia)

11.1.5 Sprint 1 - Retrospective

A continuación, un espacio para que el equipo comparta sus comentarios asociados a la ejecución del sprint, en la figura 23 se detalla la Información de la retrospective y en la figura 24 las preguntas asociadas a la ejecución del sprint.

Datos del Sprint		
Nº de Sprint	Sprint 1	
Periodo	Lunes 03 de Julio del 2023 al Viernes 28 de Julio de 2023	
Datos de la Retrospective		
Dia	Viernes 28 de Julio de 2023	
Hora	19:00	
Lugar	Sesion remota por Zoom	
Duracion Estimada	60 minutos	
Participantes	Scrum	Patricio Valenzuela
	Product Owner	Patricio Valenzuela
	Develop	Patricio Valenzuela

Figura 23 – Información de la retrospective (Fuente: Elaboración propia)

¿ Que se realizo bien?	¿ Que se realizo mal?
-- Se finalizo el sprint sin issues. -- La comunicacion siempre fue constante con PO y el cliente.	-- Se pudo finalizar el Sprint en menos tiempo.
Ideas	Acciones a Realizar
-- Acelerar la ejecucion de las tareas para disponibilizar mas tiempo frente a cualquier issue.	Priorizar los avances de documentacion

Figura 24 – Preguntas – Sprint retrospective (Fuente: Elaboración propia)

11.2 Sprint 2

El proyecto para desarrollar como Mínimo Producto Viable (MPV) en el Sprint 2 tiene como meta realizar el desarrollo de los programas para extraer las estadísticas de los servidores a monitorear con una duración de 4 semanas lo cual hace factible el cumplimiento de lo que se encuentra planificado.

A continuación, en la Figura 25 detallamos las actividades desarrolladas para el Sprint Backlog 2.

Elemento de trabajo pendiente	Puntos de historia	Responsable	Estado	Estimado Original	Sprint Backlog # 02														
					Día 1	Día 2	Día 3	Día 4	Día 5	Día 6	Día 7	Día 8	Día 9	Día 10	Día 11	Día 12	Día 13	Día 14	Día 15
HU06	3	P.Valenzuela	Done	2	2	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Analizar HU y sus riesgos		P.Valenzuela	Done	3	3	3	3	-	-	-	-	-	-	-	-	-	-	-	-
Creacion de Script Metricas de Espacio		P.Valenzuela	Done	2	2	2	3	-	-	-	-	-	-	-	-	-	-	-	-
Pruebas de Script de Metricas de Espacio		P.Valenzuela	Done	3	3	3	3	-	-	-	-	-	-	-	-	-	-	-	-
Configuracion GitHub Action para Script		P.Valenzuela	Done	3	3	3	3	-	-	-	-	-	-	-	-	-	-	-	-
Validacion de correcta ejecucion de Script		P.Valenzuela	En Ejecucion	2	2	2	2	2	2	2	2	-	-	-	-	-	-	-	-
Creacion de Script de Registro de Metricas		P.Valenzuela	En Ejecucion	2	2	2	2	2	2	2	2	-	-	-	-	-	-	-	-
Creacion de Script Metricas de Memoria/Swap		P.Valenzuela	En Ejecucion	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Pruebas de Script de Metricas de Memoria/Swap		P.Valenzuela	En Ejecucion	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Configuracion GitHub Action para Script		P.Valenzuela	Pendiente	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Validacion de correcta ejecucion de Script		P.Valenzuela	Pendiente	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Creacion de Script de Registro de Metricas		P.Valenzuela	Pendiente	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Integracion de Scripts en Ansible		P.Valenzuela	Pendiente	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Total				27	27	27	25	23	17	17	15	15	13	13	10	10	8	8	8

Figura 25 – Sprint Backlog 2 (Fuente: Elaboración propia)

Adicionalmente adjuntamos el Burndown Chart en la Figura 26 del Sprint Backlog 2.

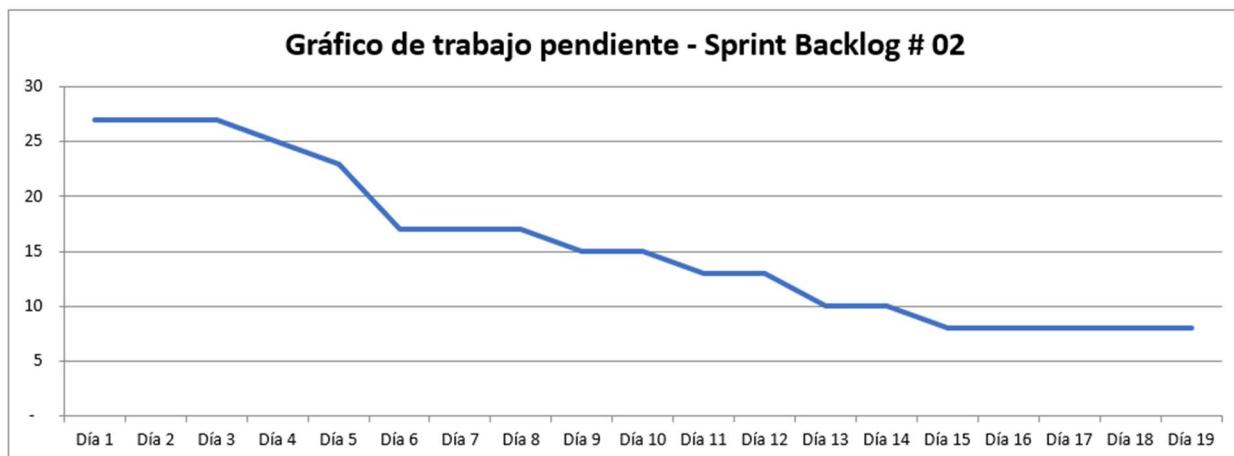


Figura 26 – Burndown Chart Sprint 2 (Fuente: Elaboración propia)

11.2.1 Riesgos Sprint 2

En este Sprint 2 que aún se encuentra en ejecución no han gatillado algún riesgo que fue identificado originalmente y que se encuentra documentado en el capítulo 7.3 Matriz de Riesgo, pero si han generado nuevos riesgos como el riesgo 13 el cual se suma a los ya previamente documentados. A continuación, en la figura XX se puede apreciar la actualización de la tabla de riesgos actualizada.

Id Riesgo	Descripción del Riesgo	Categoría	Probabilidad	Impacto	Magnitud	Riesgo Qualitativo
R01	No lograr disponibilizar el Tenant Free	Proyecto	1	1	1	Bajo
R02	No poder desplegar la infraestructura requerida	Proyecto	1	1	1	Bajo
R03	No disponer de cuotas de recursos para su despliegue	Proyecto	1	1	1	Bajo
R04	No contar con la versión de base de datos requerida	Proyecto	1	1	1	Bajo
R05	Conocimientos de GitHub	Tecnico	2	2	4	Bajo
R06	Conocimientos de Korn Shell Scripting	Tecnico	3	3	9	Medio
R07	Conocimientos de Ansible	Tecnico	1	1	1	Bajo
R08	Conocimientos de Python	Tecnico	3	3	9	Medio
R09	Perdida de enlace local	Proyecto	4	5	20	Alto
R10	No cumplir con las evaluaciones de Testing	Proyecto	1	1	1	Bajo
R11	No cumplir objetivos de alcance a servidores a ser parte del monitoreo	Tecnico	2	2	4	Medio
R12	No cumplir con horas de trabajo definidas en los sprint	Proyecto	3	5	15	Alto
R13	Los materiales de software definidos no cumplen para las definiciones de las funcionalidades necesarias.	Proyecto	5	5	25	Alto

Tabla 13 – Matriz de Riesgo actualizada (Fuente: Elaboración propia)

11.2.2 Matriz de trazabilidad de requerimientos Sprint 2

A continuación, la tabla 14 nos muestra la matriz de trazabilidad de los requerimientos que se encuentran definidos en capítulo 4 y su relación con las Historias de Usuarios que se encuentra en el Capítulo 9 en relación con la ejecución de las tareas definidas en el Sprint 2.

HU	Requerimiento	Sprint	Requerimientos	Estado
HU06	Yo como administrador quiero registrar las diferentes métricas de servicio de los servidores UNIX para que puedan ser consultadas.	Sprint 2	RQ1, RQ5	En Ejecución

Tabla 14 – Matriz de Trazabilidad de requerimientos Sprint 2 (Fuente: Elaboración propia)

11.2.3 Propuesta de Controles y Evidencia

Este sprint 2 aún se encuentra en ejecución, pero dentro de las fechas y estimaciones estipuladas y adicionalmente se adjuntan las evidencias de los avances de las actividades finalizadas y las que se encuentran en ejecución actualmente.

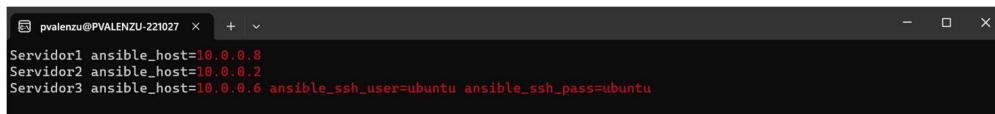
- **Configuración de Ansible:** Para poder ejecutar ansible se debe configurar el ambiente en forma correcta como se puede visualizar en la figura 27 del archivo de configuración ansible.cfg.



```
[defaults]
host_key_checking = false
remote_user = opc
private_key_file = /home/opc/.ssh/id_rsa_bastion
```

Figura 27 – Archivo ansible.cfg (Fuente: Elaboración propia)

- **Configuración archivo de Inventario:** Para las ejecuciones de ansible se debe contar con un archivo el cual indique a que servidores deberá acceder a monitorear para ello se crea el archivo de nombre inventario el cual se generara de forma dinámica cuando se ejecute como se puede visualizar en la figura 28 del archivo de inventario.



```
Servidor1 ansible_host=10.0.0.8
Servidor2 ansible_host=10.0.0.2
Servidor3 ansible_host=10.0.0.6 ansible_ssh_user=ubuntu ansible_ssh_pass=ubuntu
```

Figura 28 – Archivo inventario (Fuente: Elaboración propia)

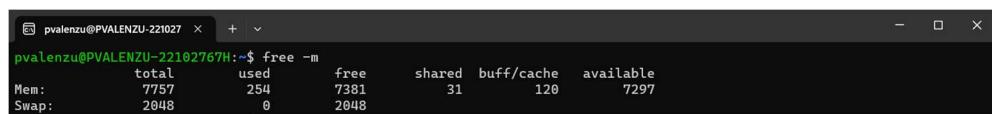
- **Configuración playbook memoria:** Para poder obtener las métricas desde los servidores que serán monitoreados se creó el archivo playbook memoria.yaml el cual posee las instrucciones suficientes para poder obtener la información de métricas de asignación y utilización de la memoria como se observa el figura 29.



```
- name: Valida Facts de Servidor
hosts: all
tasks:
- name: Imprime Memoria Asignada y Utilizada
ansible.builtin.shell: free -h
register: salida_mem
```

Figura 29 – Archivo memoria.yaml (Fuente: Elaboración propia)

Adicionalmente se adjunta la evidencia de ejecución del script como se observa en la figura 30.

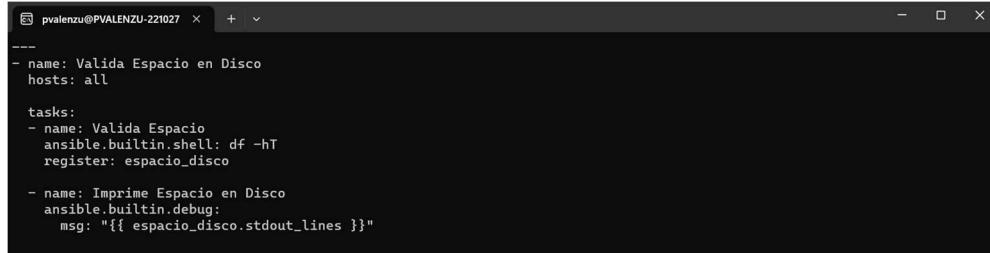


```
pvalenzu@PVALENZU-22102767H:~$ free -m
              total        used        free      shared  buff/cache   available
Mem:       7757         254       7381          31        120       7297
Swap:      2048           0       2048
```

Figura 30 – Métricas de Memoria de servidor monitoreado (Fuente: Elaboración propia)

Actualmente está en etapa de desarrollo el parseo de la información para poder ser registrada en la instancia de base de datos.

- **Configuración playbook disco:** Para poder obtener las métricas desde los servidores que serán monitoreados se creó el archivo playbook disco.yaml el cual posee las instrucciones suficientes para poder obtener la información de métricas de asignación y utilización de los discos o luns como se observa el figura 31.



```

pvaleNZU@PVALENZU-221027: ~
-----
- name: Valida Espacio en Disco
  hosts: all

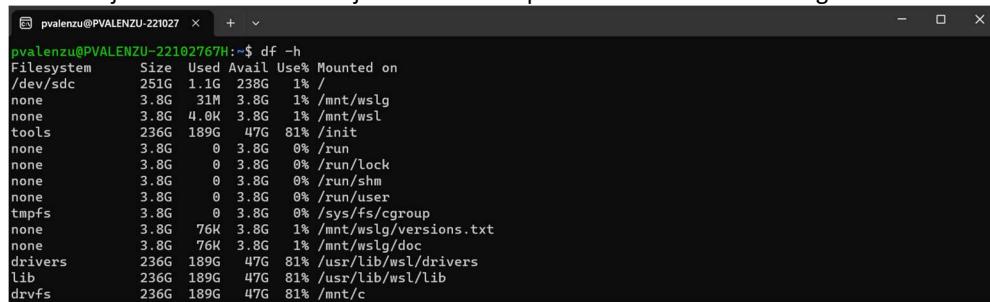
  tasks:
    - name: Valida Espacio
      ansible.builtin.shell: df -hT
      register: espacio_disco

    - name: Imprime Espacio en Disco
      ansible.builtin.debug:
        msg: "{{ espacio_disco.stdout_lines }}"

```

Figura 31 – Archivo disco.yaml (Fuente: Elaboración propia)

Adicionalmente se adjunta la evidencia de ejecución del script como se observa en la figura 32.



Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/sdc	251G	1.1G	238G	1%	/
none	3.8G	31M	3.8G	1%	/mnt/wsl
none	3.8G	4.0K	3.8G	1%	/mnt/wsl
tools	236G	189G	47G	81%	/init
none	3.8G	0	3.8G	0%	/run
none	3.8G	0	3.8G	0%	/run/lock
none	3.8G	0	3.8G	0%	/run/shm
none	3.8G	0	3.8G	0%	/run/user
tmpfs	3.8G	76K	3.8G	1%	/sys/fs/cgroup
none	3.8G	76K	3.8G	1%	/mnt/wsl/g/versions.txt
none	3.8G	76K	3.8G	1%	/mnt/wsl/doc
drivers	236G	189G	47G	81%	/usr/lib/wsl/drivers
lib	236G	189G	47G	81%	/usr/lib/wsl/lib
drvfs	236G	189G	47G	81%	/mnt/c

Figura 32 – Métricas de Espacios de servidor monitoreado (Fuente: Elaboración propia)

- **Ejecución de archivos yaml:** Para poder ejecutar los playbooks de ansible para lograr obtener las diferentes métricas de los servidores que serán monitoreados se deberá ejecutar las siguientes instrucciones las cuales serán almacenadas en un archivo de ejecución de Shell Unix.
 - **ansible-playbook -i inventario memoria.yaml**
 - **ansible-playbook -i inventario disco.yaml**
- **Funcionalidades Faltantes:** Todas las funcionalidades faltantes que son parte del sprint 2 están a la espera de su codificación que se encuentran en la planificación original.

11.2.4 Sprint 2 - Review

A continuación, se comparte los comentarios y retroalimentación recibidos en 2 sesiones sincrónicas las cuales podemos resumir en la tabla 15.

Sprint 2 - Review					
START	DURACION	ACTIVIDAD	DESCRIPCION	QUIEN	
19:00	10 min	Objetivo del sprint	Presentacion de Objetivos del Sprint	Product Owner	
			Revision de las metas del Sprint		
			Revision de la duracion del Sprint		
19:10	10 min	Revisión de los principales requisitos	Codificacion de la obtencion de las metricas	Product Owner	
			Revision y pruebas de la programacion		
			Revision de los pendientes de codificacion para completar el sprint		
19:20	10 min	Status sprint	El plan de ejecucion se encuentra en ejecucion dentro de los tiempos	Scrum Master	
			Hasta este punto aun no se encuentran issues.		
			Hasta este punto no han gatillado ninguno de los riesgos establecidos p		
19:30	10 min	Demostracion	Solo se muestran las ejecuciones de los scripts que han sido desarrolladas hasta este punto.	Team	
19:40	10 min	Feedback	Para este Sprint 2 se han tomado las recomendaciones realizadas y se han realizado los cambios sugeridos.	Scrum Master	
19:50	10 min	Cierre	En la siguiente sesion sincronica se revisara el avance del sprint 2	Scrum Master	

Tabla 15 – Sprint Review (Fuente: Elaboración propia)

11.2.5 Sprint 2 - Retrospective

A continuación, un espacio para que el equipo comparta sus comentarios asociados a la ejecución del sprint, en la figura 31 se detalla la Información de la retrospective y en la figura 32 las preguntas asociadas a la ejecución del sprint.

Datos del Sprint		
Nº de Sprint	Sprint 2	
Periodo	Lunes 31 de Julio del 2023 al Lunes 14 de Agosto de 2023	
Datos de la Retrospective		
Dia	Lunes 14 de Agosto de 2023	
Hora	19:00	
Lugar	Sesion remota por Zoom	
Duracion Estimada	60 minutos	
Participantes	Scrum	Patricio Valenzuela
	Product Owner	Patricio Valenzuela
	Develop	Patricio Valenzuela

Figura 31 – Información de la retrospective (Fuente: Elaboración propia)

¿ Que se realizo bien?	¿ Que se realizo mal?
-- Configuracion Ansible. -- La creacion de los archivos de playbooks para obtencion de las metricas.	-- Si no finalizan los scripts de extraccion de metricas podria generar un retraso de la planificacion.
Ideas	Acciones a Realizar
-- Cambiar estructuras de obtencion de metricas.	Priorizar los avances de documentacion

Figura 32 – Preguntas – Sprint retrospective (Fuente: Elaboración propia)

11.3 Sprint Summary

A continuación, en la figura 33 se presenta el resumen de los Sprint 1 que fue completado en su totalidad y el Sprint 2 que se encuentra en ejecución.

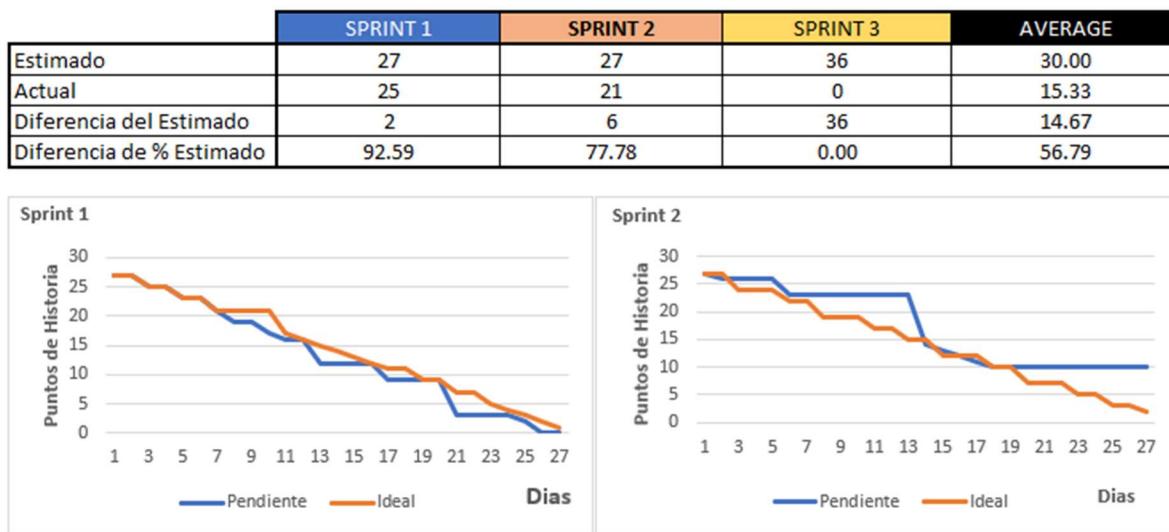


Figura 33 – Sprint Summary (Fuente: Elaboración propia)

12. Conclusión

En base a los avances obtenidos en la ejecución del Sprint 1 y la reducción del tiempo estimado, y adicionalmente la correcta utilización de la metodología de control aplicada “SCRUM” podemos determinar que el Sprint 2 también podremos reducir el tiempo planificado para así dejar estos deltas conseguidos para el Sprint 3 que es el más largo y así poder finalizar satisfactoriamente la entrega del Mínimo Producto Viable.

13. Bibliografía

- <https://www.ionet.cl/post/problemas-informaticos-comunes-en-las-pymes-los-vives>
- <https://revistasumma.com/cuales-son-los-paises-donde-mas-se-roba-el-wifi/>
- https://www.researchgate.net/publication/346060429_Implementando_srum_rad_para_la_gestion_y_desarrollo_de_proyectos_de_software_en_equipos_de_trabajo_con_personal_limitado_y_eventual
- <https://asana.com/es/resources/scope-management-plan>
- https://pirhua.udep.edu.pe/bitstream/handle/11042/3109/PYT-Capitulo_2-Gestion_del_Alcance_del_Proyecto-Juan_Quinde.pdf?sequence=1&isAllowed=y
- <https://blog.ganttpro.com/es/alcance-del-proyecto/>
- <https://ciberseguridad.blog/25-tipos-de-ataques-informaticos-y-como-prevenirlos/>
- <https://www.reuters.com/article/internet-tecnologia-mcafee-codigo-idLTASIE67919E20100810>
- <https://revistasumma.com/cuales-son-los-paises-donde-mas-se-roba-el-wifi/>
- <https://www.ionet.cl/post/problemas-informaticos-comunes-en-las-pymes-los-vives>
- <https://kubernetes.io/es/docs/home/>
- <https://docs.ansible.com/>
- <https://docs.github.com/es>
- <https://docs.github.com/es/actions>
- <https://www.scrum.org/resources/blog/scrum-no-es-una-metodologa-prescriptiva-es-un-marco-de-trabajo>
- <https://www.isotools.us/2020/12/15/como-analizar-los-riesgos-segun-la-iso-31010/>
- https://cursos.iplacex.cl/CED/GAR5005/S6/ME_6.pdf
- <https://www.cybermedian.com/es/agile-backlog-prioritization-technique-moscow/>
- <https://asana.com/es/resources/t-shirt-sizing>