TECNOLÓGICO DE ESTUDIOS SUPERIORES DE COACALCO

Clúster de servidores con Linux Fedora y Firebase para la gestión de aplicaciones web

Informe Técnico de Residencias Profesionales

INGENIERÍA EN SISTEMAS COMPUTACIONALES

Especialidad
Ingeniería para el Desarrollo de Software

Presenta(n):
Campos Hernández Arturo Adrián
Rojo López Edson Bryan

Coacalco de Berriozábal, Noviembre/2024

AGRADECIMIENTOS

Edson Bryan Rojo López.

A mis queridos padres y hermana, gracias por su amor incondicional y apoyo constante para poder lograr este maravilloso proyecto, sin ustedes no lo hubiera logrado. Al profesor Oscar Escobar, mi gratitud por su dedicación hacia este proyecto, enseñanzas valiosas y apoyo durante mi estancia en la universidad. A mi compañero de equipo y amigo Arturo, agradezco tu colaboración, dedicación e impulso para lograr este resultado final. Al Tecnológico de Estudios Superiores de Coacalco, gracias por proporcionar el ambiente propicio para mi desarrollo académico.

Arturo Adrián Campos Hernández.

Agradezco a mis padres, hermana y abuela por el apoyo en el proceso de la elaboración del proyecto, demostrando su cariño y amor incondicional, también dando ánimos con el camino loco y complicado que tome. A su vez al profesor Oscar Escobar por recibirnos y asesorarnos en el proyecto y llenarnos de nuevos aprendizajes en esta tarea tan complicada. A mi compañero, amigo y colega de trabajo Edson por recibirme como compañero de trabajo mostrando toda su dedicación para llegar hasta el final. Por último, al Tecnológico de Estudios Superiores de Coacalco por proporcionarme a profesores que pude extraer todo conocimiento de ellos.

RESUMEN

Al desarrollar contables sistemas anteriores, surge la necesidad de la creación de un clúster de servidores, aunque primeramente se necesita probar de forma progresiva con otros sistemas más livianos, partiendo desde los servicios de Apache (HTPPD), Base de datos interna y externa (Firebase y sus APIs), además de un balanceador para evitar las cargas a los servidores web.

Para montar el clúster y evaluar todos los sistemas operativos posibles se opta por Fedora dando apertura a usar sus ambas distribuciones que son Workstation y Server usando los servicios que ofrece, además de cargar un pequeño sistema hosteado con el mismo servidor y usando un servicio de DNS externo para que este en línea, los cuales son Noip para ocultar la dirección pública y CloudFlare.

El sistema utilizado para pruebas es una versión incompleta diseñada para facilitar los procedimientos de trámites relacionados con los deudores del Tecnológico de Estudios Superiores de Coacalco, con el propósito de gestionar la titulación, cambios de carrera o la baja de la institución.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO I
GENERALIDADES DEL PROYECTO

INTRODUCCIÓN

El presente proyecto se centra en la implementación de un clúster de servidores con Linux Fedora y Firebase con el propósito de mejorar la gestión de aplicaciones web. Este abarca la adopción de una infraestructura tecnológica que permitirá una mayor eficiencia operativa, seguridad de datos y escalabilidad, aspectos cruciales en un entorno digital altamente competitivo. El interés reside en proporcionar un servicio de alta calidad y personalización propia, para cumplir con regulaciones de seguridad y privacidad de datos en constante cambio, y mantener nuestra competitividad en el mercado.

Este proyecto se llevará a cabo mediante el enfoque ágil SCRUM, que permite una gestión iterativa e incremental del desarrollo y la implementación de la infraestructura. La metodología involucrará la definición de un backlog de producto, la planificación de iteraciones (sprints), desarrollo, revisión y retroalimentación continua para garantizar que las necesidades de la organización se aborden de manera efectiva y oportuna.

Se reconoce que la implementación de un clúster de servidores es un proyecto complejo que puede enfrentar desafíos técnicos y de recursos. Las limitaciones pueden incluir restricciones presupuestarias, limitaciones de personal técnico y posibles obstáculos en la migración de datos y aplicaciones existentes.

El trabajo se organiza en varios capítulos que abordan aspectos específicos del proyecto:

- En el primer capítulo, se plantea el problema y se justifica la necesidad de la implementación del clúster de servidores.
- El segundo capítulo se centra en la revisión de la literatura y la revisión de las tecnologías clave involucradas en el proyecto.
- En el tercer capítulo, se presenta la metodología de trabajo y el enfoque SCRUM utilizado.
- El cuarto capítulo describe detalladamente la implementación del clúster de servidores con Linux Fedora y Firebase.
- El quinto capítulo aborda la evaluación de los resultados y el rendimiento del sistema. Finalmente, el sexto capítulo presenta las conclusiones y recomendaciones finales.

UBICACIÓN

Av. 16 de septiembre # 54, C.P. 55700, Col. Cabecera municipal, Coacalco de Berriozábal, México, México.

ORGANIGRAMA

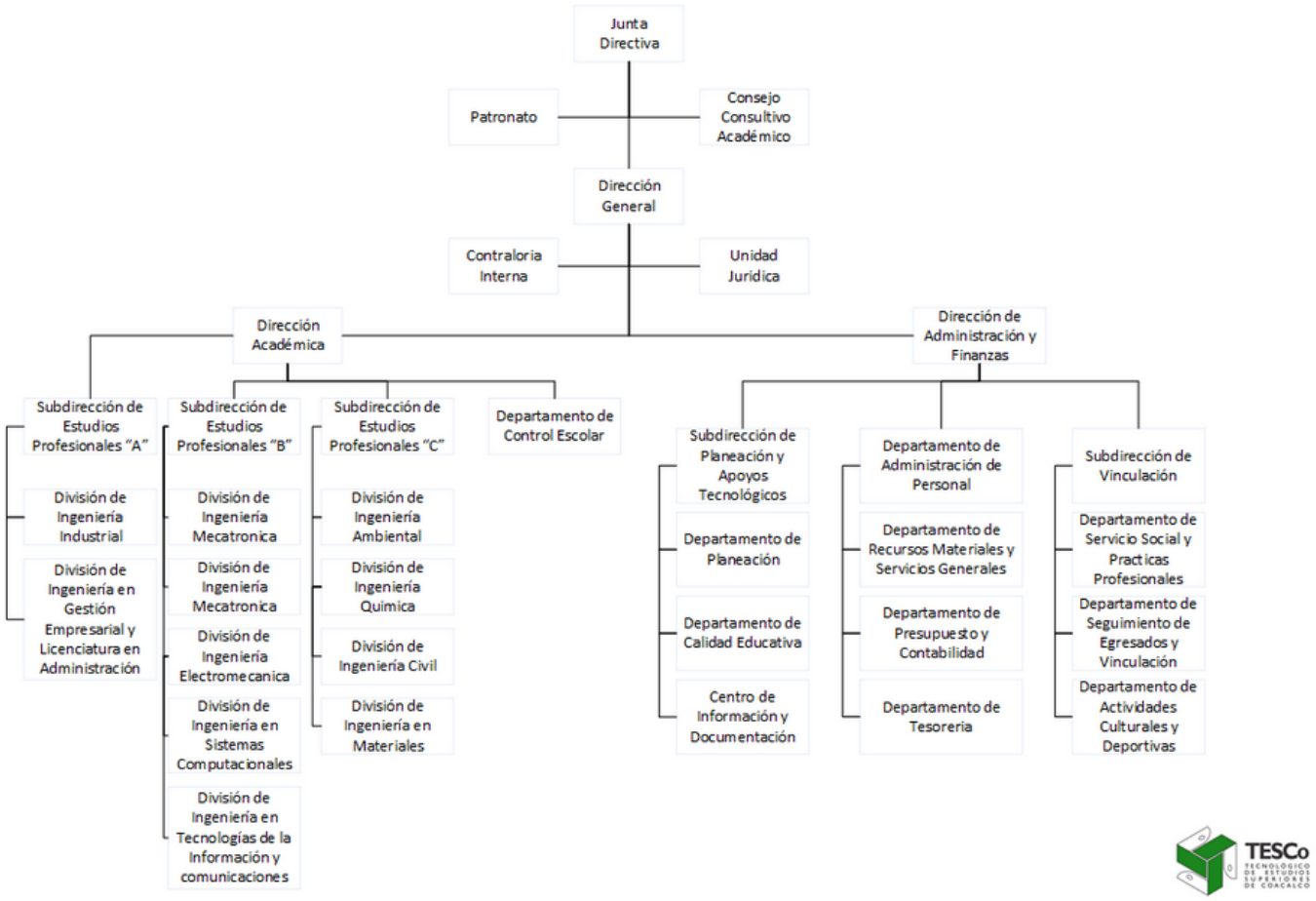

Figura 1: Organigrama del Tecnológico de Estudios Superiores de Coacalco (TESCo, s.f.).

SECTOR PRODUCTIVO DE LA EMPRESA

Terciario

MISIÓN, VISIÓN, FILOSOFÍA Y/O VALORES

MISIÓN

Somos una institución pública de Educación Superior Tecnológica que forma de manera integral profesionistas e investigadoras e investigadores, con creatividad e innovación, capaces de resolver problemas de su ámbito laboral y de apoyar el desarrollo tecnológico; a través de la oferta de programas educativos de licenciatura y posgrado reconocidos por su calidad, con un enfoque de responsabilidad social basado en valores y el respeto al medio ambiente.

VISIÓN

Consolidarse como una institución pública de Educación Superior Tecnológica que fomenta la investigación básica, aplicada y el desarrollo tecnológico vinculado con las empresas e instituciones nacionales e internacionales a través de su alumnado, egresadas y egresados, profesoras, profesores, investigadoras e investigadores con enfoque en las áreas de automatización y control, manufactura avanzada y desarrollo sustentable.

BREVE HISTORIA DEL TECNOLÓGICO DE ESTUDIOS SUPERIORES DE COACALCO.

El Tecnológico de Estudios Superiores de Coacalco, TESCo fue creado el 4 de Septiembre de 1996, como Organismo Público Descentralizado de Carácter Estatal, con Personalidad Jurídica y Patrimonio Propios; con objeto de formar profesionales e investigadores aptos para la aplicación y generación de conocimientos, con capacidad crítica y analítica en la solución de problemas, con sentido innovador para incorporar los avances científicos y tecnológicos al ejercicio responsable de la profesión, de acuerdo a los requerimientos del entorno, el Estado y el País.

El Tecnológico de Estudios Superiores de Coacalco forma parte del Sistema Nacional de Institutos Tecnológicos, SNIT, que está constituido por 131 Institutos Tecnológicos Descentralizados, 126 Institutos Tecnológicos Federales y seis Centros de Investigación, que suman un total de 263 instituciones, en las que se atiende a una población escolar de 491,165 estudiantes en Licenciatura y Posgrado. (TESCo, s.f.).

PROBLEMA A RESOLVER

Durante el proceso de desarrollo de diversas aplicaciones bajo el nombre de “COMPRESSOFT SYSTEMS”; en la programación e implementación surge la necesidad de encontrar servidores para las bases de datos, APIs y otras funciones, ya que la demanda cada día aumenta de forma considerable y los servicios aumentan de costo.

De acuerdo con el contexto anterior es de suma importancia priorizar la implementación de un clúster de servidores propio y de renta utilizando Linux Fedora y Firebase para la gestión de aplicaciones web. Esta propuesta se basa en la necesidad de abordar los problemas actuales, como el rendimiento, las solicitudes de información en bases de datos, ancho de banda y a futuro un respaldo de seguridad. Esto permite la mejora del rendimiento de carga y procesamiento de solicitudes además de la optimización de recursos facilitando la gestión de servidores físicos y virtuales, reduciendo los costos operativos, y operación propia.

Este enfoque proporcionará beneficios significativos, como un rendimiento mejorado, una gestión eficiente de recursos, y una reducción de costos operativos. Al adoptar este modelo, los sistemas implementados no solo optimizarán la carga y el procesamiento de solicitudes, sino que también establecerán una base sólida para el futuro crecimiento y desarrollo de otros sistemas. En última instancia, el clúster de servidores se presenta como una solución integral para garantizar la estabilidad, escalabilidad y eficiencia continua de los sistemas que pueden implementar este modelo.

Las consecuencias de que no se aplique este clúster para su uso:

- Costos Operativos Elevados: La gestión ineficiente de recursos y la falta de escalabilidad seguirán generando costos operativos innecesarios.
- Riesgos de Seguridad: La vulnerabilidad a amenazas cibernéticas aumentará el riesgo de brechas de seguridad y pérdida de datos críticos.
- Incapacidad para Cumplir Normativas: La falta de cumplimiento normativo puede dar lugar a sanciones legales y dañar nuestra reputación.

OBJETIVOS

OBJETIVO GENERAL

Diseñar una estructura de servidores (clúster), mediante la implementación de un servidor Linux Fedora y una integración del servicio Firestore de Firebase, para el almacenamiento y gestión de información de aplicaciones web.

OBJETIVOS ESPECÍFICOS

1. Realizar un análisis de las diferentes alternativas sobre clústeres
    1.1. Comparar los diferentes métodos de conexión de clúster
    1.2. Elegir el estándar de calidad a usar para la calidad del clúster
2. Configurar un servidor maestro para gestionar la distribución de tareas entre los nodos del clúster.
    2.1. Configurar el servidor con Linux Fedora.
    2.2. Definir los nodos del clúster.
3. Configurar Firebase para el almacenamiento y sincronización de datos.
    3.1. Implementar la conexión a Firebase mediante los recursos proporcionados por la plataforma.
    3.2. Configurar el servicio Auth de Firebase.
4. Evaluar el rendimiento, así como la capacidad del clúster para reducir tiempos de respuesta.
    4.1. Utilizar Benchmarks para evaluar el rendimiento del clúster.
    4.2. Supervisar la red y el tráfico de datos con herramientas de supervisión.
5. Implementar la seguridad en las capas de red del servidor maestro.
    5.1. Analizar las diferentes herramientas de seguridad adecuadas para las capas de red.
    5.2. Configurar Listas de control de acceso utilizando los estándares marcados por Cisco.
    5.3. Utilizar protocolos de cifrados seguros, como HTTPS y SSH, para proteger las comunicaciones entre servidores y entre clientes.
6. Conectar Firebase con servidor maestro.
    6.1. Configurar el proveedor de internet mediante el modem y el firewall para la salida del servidor maestro.

6.2. Definir los puertos de entrada y salida de la conexión.

6.3. Realizar la conexión de Firebase con el servidor maestro.

6.4. Verificar el funcionamiento de la herramienta con las aplicaciones web y el servidor.

7. Evaluar el servidor maestro para la necesidad del cliente o usuario.

7.1. Verificar con la ISO/IEC 27001 el funcionamiento del servidor maestro.

7.2. Documentar el proceso bajo los estándares de la ISO/IEC 27001.

7.3. Implementar los nuevos cambios al servidor maestro en base al resultado de la evaluación.

JUSTIFICACIÓN

Hoy en día las empresas buscan la rapidez y eficiencia en sus servicios; es de suma importancia buscar costos de operación bajos y servicios para uso propio. Las soluciones actuales manejan modelos de rápida implementación y despliegue de aplicaciones web y móviles permitiendo un veloz inicio de soluciones para resolver problemas de índole especifica, con poca personalización y ajustándose a limitaciones de hardware debido a los planes de renta para el uso de equipo “rentado”.

La solución para obtener un modelo con estructura, con un menor número de limitaciones y personalizable se basa en construir un clúster de servidores físico y virtual, esto permitirá optimizar los recursos y mejorar el rendimiento de las aplicaciones; contando con seguridad y monitorización física.

CAPÍTULO II
MARCO TEÓRICO

2.1 MARCO CONCEPTUAL

Redes de Computadora: Una red de ordenadores se refiere a dispositivos de computación interconectados que pueden intercambiar datos y compartir recursos entre sí. Los dispositivos de la red utilizan un sistema de reglas, llamados protocolos de comunicaciones, para transmitir información a través de tecnologías físicas o inalámbricas. (AWS, s.f.). Los nodos y los enlaces son los componentes básicos de las redes de ordenadores. Un nodo de red puede ser un equipo de comunicación de datos (DCE), como un módem, un concentrador o un conmutador, o un equipo terminal de datos (DTE), como dos o más ordenadores e impresoras. Un enlace se refiere a los medios de transmisión que conectan dos nodos. Los enlaces pueden ser físicos, como cables o fibras ópticas, o espacio libre utilizado por redes inalámbricas. Los nodos siguen un conjunto de reglas o protocolos que definen cómo enviar y recibir datos electrónicos a través de los enlaces. La arquitectura de la red de computadoras define el diseño de estos componentes físicos y lógicos. Proporciona las especificaciones para los componentes físicos de la red, la organización funcional, los protocolos y los procedimientos. (AWS, s.f.).

Fedora: Fedora es un sistema operativo libre basado en Linux que además ofrece una plataforma de desarrollo. Es libre para su uso, modificación y distribución, siendo además gratuito. Fedora se construye utilizando las últimas tecnologías del mundo open source y se desarrolla a través de una comunidad de apasionados usuarios y desarrolladores empeñados en proporcionar y mantener la mejor distribución de software libre del mundo. (Fedora, s.f.).

Firebase: Es una plataforma de apps para dispositivos móviles con bibliotecas cliente integradas y unificadas en varios lenguajes de programación para dispositivos móviles. Las funciones de backend como servicio (BaaS) de Firebase te ayudan a desarrollar apps de alta calidad, aumentar la base de usuarios y ganar más dinero. Cada función es independiente, pero juntas funcionan aún mejor. (Firebase, s.f.).

Clústeres de Kubernetes: Un clúster de Kubernetes es un grupo de nodos (máquinas que ejecutan aplicaciones). Cada nodo puede ser una máquina física o una máquina virtual. La capacidad del nodo (su número de CPU y la cantidad de memoria) se define al crear el nodo. Un clúster comprende:

- Nodos de plano de control ("nodos maestros"). Normalmente, habrá tres nodos de plano de control para una alta disponibilidad.
- Nodos de trabajador, organizados en pools de nodos. (Oracle, s.f.).

iSCSI: El protocolo SCSI (Small Computer System Interface) original se diseñó para compartir datos a través de una red de área local (LAN). El protocolo iSCSI se desarrolló a finales de la década de 1990 para permitir el protocolo SCSI a través de una red TCP/IP. Es un protocolo de capa de transporte diseñado para proporcionar un acceso sin interrupciones a los dispositivos de almacenamiento a través una red. El nombre iSCSI se utilizó para indicar que el protocolo original se modificó y encapsula los comandos SCSI en paquetes TCP/IP. La arquitectura iSCSI es de la categoría cliente-servidor. El cliente se conoce como iniciador y el servidor se conoce como destino iSCSI. El dispositivo de almacenamiento en bloques se conoce como una unidad lógica y un destino iSCSI puede tener muchas unidades lógicas. Cada uno tiene un número de unidad lógica (LUN) designado. (Amazon Web Services [AWS], s. f.).

HAProxy: HAProxy es un equilibrador de carga TCP o http de código fuente abierto. Añada este componente para añadir prestaciones de equilibrio de carga a la arquitectura de recopilación de datos. También puede utilizarlo para proteger LFA frente a cambios que se realicen en el clúster receptor. (IBM Operations Analytics Log Analysis [IBM], 2021).

Red Hat Enterprise Linux Server: Se trata de un sistema operativo muy sencillo de controlar y gestionar que puede implementar en el entorno que necesite, ya sea en los sistemas físicos de hardware, las máquinas virtuales o en la nube. Funciona como base para la infraestructura de TI y es compatible con las plataformas de hardware más importantes (x86, ARM, POWER y Z), así como con miles de aplicaciones personalizadas y comerciales. (Red Hat® Enterprise Linux® [RHEL], s. f.).

ABC GNU/Linux: La distribución ABC GNU/Linux basada en Ubuntu y está especializada en la construcción automática de clústeres Beowulf de alto rendimiento con tan solo arrancar el sistema en modo "live" en el frontend o siendo instalado en su disco duro. Los nodos arrancan diskless a través de PXE. Usa como gestor de ventanas Gnome. Integra

el monitor de recursos Ganglia. Se trata de la primera distribución que integra todas estas características. (Castaños, 2011).

CentOS: Es un proyecto open source que ofrece dos distribuciones de Linux® diferentes: CentOS Stream y CentOS Linux. CentOS Stream es una plataforma de desarrollo upstream para las próximas versiones de Red Hat Enterprise Linux. CentOS viene de Community ENTerprise Operating System, que en español significa "sistema operativo para empresas basado en los aportes de la comunidad". (Red Hat® Enterprise Linux® [RHEL], 2023).

VMware Workstation: es una línea de productos de Hipervisor de escritorio que permiten a los usuarios ejecutar máquinas virtuales, contenedores y clústeres de Kubernetes. (WMware, INC, 2023).

Oracle VM VirtualBox: Oracle VM VirtualBox es una virtualización multiplataforma de código abierto software que permite ejecutar múltiples sistemas operativos simultáneamente en un solo dispositivo. Los desarrolladores utilizan VirtualBox para entregar código más rápido ejecutando y probando diferentes sistemas operativos en su computadora portátil. (Oracle, 2020).

Gnome Boxes: Boxes es una aplicación que le da acceso a máquinas virtuales, ejecutándose local o remotamente. También le permite conectarse a la pantalla de un equipo remoto. (The GNOME Project, 2014).

Apache HTTP: Es un esfuerzo para desarrollar y mantener un servidor HTTP de código abierto para sistemas operativos modernos, incluidos UNIX y Windows. El objetivo de este proyecto es proporcionar un servidor seguro, eficiente y extensible que proporcione servicios HTTP en sincronía con los estándares HTTP actuales. (The Apache Software Foundation, s. f.)

2.2 ANTECEDENTES HISTÓRICOS

Hosting web: Remotamente esta tecnología se populariza después del desarrollo el primer sitio web el CERN de Suiza, GeoCities fue fundada en 1994, seguido de Tripod en 1995, ambas pioneras en este sector, estas ofrecían servicios de alojamiento web gratuito a usuarios que deseaban subir páginas web, actualmente es muy común encontrar servicios tanto gratuitos como de pago entre los que destacan Hostinger, HostGator o GoDaddy. Usualmente brindan paneles de control y sistemas de gestión que utilizan tecnologías como Apache, NGIX, PHP y MySQL para alojar y administrar sitios web.

Compartición de medios: Inicia su historia en 2004 con el lanzamiento de la famosa plataforma de video Vimeo, seguido de YouTube en 2006 y SoundCloud en 2007. La compartición de multimedia utiliza tecnologías de transmisión de video y audio, así como sistemas de almacenamiento en la nube para alojar contenido multimedia en línea.

Almacenamiento en la nube: En los años 2000's se dio lugar al termino almacenamiento en la nube, servicios como Amazon Web Services, Dropbox y Google Drive fueron las primeras empresas en brindar este servicio. Emplean tecnologías de almacenamiento distribuido y sincronización de archivos en servidores en la nube para ofrecer acceso desde cualquier lugar.

Transmisión en vivo: En la década de los 2000 se hizo popular este servicio de transmisión, posteriormente Justin Kan uno de los fundadores de Justin.tv servicio que más adelante cambiaria su nombre a Twitch es conocido por popularizar el servicio. Estas plataformas usan tecnologías de transmisión de video en tiempo real y servidores en la nube para permitir la transmisión y la interacción con los espectadores.

CDN (Content Delivery Network): Los servicios de CDN comenzaron a surgir en la década de los 90s, Akamai Technologies fundada en 1998 fue una de las primeras compañías en ofrecer CND, en 2009 la plataforma Cloudflare famosa por ofrecer CDN y otros servicios se estableció. Los sistemas se basan en tecnologías de redes de distribución de contenido para acelerar su entrega en los servicios web, almacenan copias caché de los sitios y recursos multimedia en servidores globales.

Streaming de música: A principios de los años 2000 es cuando este servicio inicia, con la incorporación de Pandora al mercado y fundación de Spotify en 2006 se dio un giro completo a la industria musical incorporando tecnología de vanguardia. Las plataformas actuales de música utilizan tecnologías de transmisión de audio y servidores en la nube para permitir el streaming de música y video a través de sus aplicaciones web y móviles.

Cloud Gaming: El CloudGaming se dio a conocer cerca del año 2010. NVIDA GeForce NOW y Google Stadia son ejemplos de plataformas reales de juego en la nube actualmente. Emplean tecnologías de virtualización de GPU y servidores en la nube para permitir a los usuarios jugar videojuegos en línea sin necesidad de hardware de juego local potente.

Diseño y despliegue de una infraestructura de un clúster que ofrece un servicio web basado en Apache en alta disponibilidad y con balanceo de carga en Fedora. En este proyecto, se investiga el potencial de Fedora Workstation 35 como sustituto de Linux CentOS 8 para crear una infraestructura de clúster que proporcione servicios web con alta disponibilidad y balanceo de carga a través de Apache y una base de datos. El servicio web ofrecido es WordPress. El clúster también incluye almacenamiento compartido basado en iSCSI. Se utiliza software de distribución libre en el desarrollo del proyecto. Esto se realiza en respuesta a la discontinuidad de la distribución de Linux CentOS 8. (Del Pino Domínguez y ULPGC, 2022)

2.3 INFORMACIÓN TEÓRICA

En informática, el término clúster (“grupo” o “racimo”) hace referencia a conjuntos o conglomerados de computadoras construidos mediante el uso de hardware común y que se comportan como si fueran una única computadora. (López Fuentes, 2015).

También IBM (2022), dice que cualquier aplicación que instale en un clúster debe poder ejecutarse en cualquier servidor de aplicaciones que sea miembro de dicho clúster. Puesto que un grupo de nodos forma los límites de un clúster, todos los miembros de un clúster deben ser miembros del mismo grupo de nodos. Por lo tanto, para que la aplicación que despliegue se ejecute correctamente, todos los miembros de un clúster deben estar ubicados en nodos que cumplan los requisitos de dicha aplicación.

En una célula que tiene muchas configuraciones de servidor diferentes, puede ser difícil determinar qué nodos tienen las prestaciones para alojar la aplicación. Un grupo de nodos se puede utilizar para definir grupos de nodos que tienen suficientes en común para alojar miembros de un clúster determinado. Todos los miembros de clúster de un clúster deben estar en el mismo grupo de nodos.

Según AWS (s.f.), una red de ordenadores se refiere a dispositivos de computación interconectados que pueden intercambiar datos y compartir recursos entre sí. Los dispositivos de la red utilizan un sistema de reglas, llamados protocolos de comunicaciones, para transmitir información a través de tecnologías físicas o inalámbricas.

También AWS (s.f.), menciona que funcionan mediante nodos y los enlaces son los componentes básicos de las redes de ordenadores. Un nodo de red puede ser un equipo de comunicación de datos (DCE), como un módem, un concentrador o un conmutador, o un equipo terminal de datos (DTE), como dos o más ordenadores e impresoras. Un enlace se refiere a los medios de transmisión que conectan dos nodos. Los enlaces pueden ser físicos, como cables o fibras ópticas, o espacio libre utilizado por redes inalámbricas.

Fedora es un sistema operativo desarrollado por Red Hat Project, el cual es un sistema operativo de código abierto en base a Unix, el cual este SO tiene diversas funciones como utilización para un servidor o incluso usarlo para tareas cotidianas en sus dos distribuciones las cuales son Fedora Workstation (versión de escritorio) y Fedora Server.

CAPÍTULO III
DESARROLLO

3.1 ANÁLISIS DEL CLÚSTER

Considerando que el clúster está destinado para cubrir ciertas necesidades de un sistema previamente realizado, se debe de tomar en cuenta las prestaciones que ofrece, en este caso surgió para un sistema de gestión de envíos, pero como prueba para el clúster se implementará una página para diferentes servicios escolares previamente programado y documentado.

A la bifurcación de CentOS con otra distribución de Linux específicamente la que está a cargo de Red Hat Enterprise Linux, se puede obtener otras opciones más viables como como Fedora para Servidores (Fedora Server) y para el usuario promedio (Fedora Workstatation). Otra alternativa es ABC GNU basada en Ubuntu 9.0.1.

Fedora es la opción más viable para el proyecto ya que se puede implementar sus dos distribuciones dependiendo del servicio y/o característica que requiere el nodo especifico los cuales son virtualizados

Para crear los enlaces entre los nodos del clúster ya sea con el nodo maestro y los nodos virtualizados se proporciona diferentes opciones como:

- Clúster por Kubernetes
- Protocolo NFS
- Protocolo iSCSI

A continuación, se presenta la Tabla 1 la cual compara los diferentes métodos de conexión para los nodos del clúster

Nota: todos los datos son ficticios, así como las direcciones ips han sido censuradas por privacidad a excepción de las direcciones de la red LAN

Tabla 1: Tabla comparativa de los diferentes métodos de clúster

Característica	Kubernetes	Protocolo NFS	Protocolo iSCSI
Tipo de Clúster	Orquestación de contenedores y aplicaciones en contenedores.	Compartición de sistemas de archivos en red.	Acceso directo a bloques de datos en dispositivos de almacenamiento.
Arquitectura	Distribuido.	Cliente-Servidor.	Cliente-Servidor.
Escalabilidad	Muy alta.	Alta.	Alta.
Compatibilidad	Ideal para aplicaciones en contenedores y microservicios.	Utilizado para compartir sistemas de archivos en red.	Utilizado para acceder a bloques de datos en almacenamiento.
Tolerancia a fallos	Alta.	Depende de la configuración y redundancia de servidores NFS.	Depende de la configuración y redundancia de almacenamiento iSCSI.
Rendimiento	Bueno.	Rendimiento depende de la red y hardware. Adecuado para aplicaciones que no son intensivas en E/S.	Bueno. Ofrece acceso de alto rendimiento a bloques de almacenamiento.
Administración	Requiere el uso de herramientas específicas de Kubernetes para el despliegue y gestión de contenedores y aplicaciones.	Configuración y mantenimiento de servidores NFS.	Configuración y mantenimiento de servidores iSCSI.
Ventajas	Escalabilidad, orquestación y gestión de contenedores, automatización, despliegue rápido.	Facilidad para compartir archivos en red, compatible con una variedad de sistemas operativos.	Acceso rápido y de alto rendimiento a bloques de almacenamiento.
Desventajas	Requiere aprendizaje y experiencia en Kubernetes, más complejo de configurar y administrar.	Rendimiento puede verse afectado por la red, menos adecuado para aplicaciones intensivas en E/S.	Configuración y mantenimiento más complejos, menos adecuado para compartir archivos en red.

Al realizar esta comparativa se optó por el método de clúster por el protocolo iSCSI el cual es un método fácil y controlable en este caso pudiendo usar una red LAN o una red WAN dependiendo del proveedor de internet y la dirección IP que proporcione.

La ISO 27001 se elige ya que es un estándar para el Sistema de Gestión de la Seguridad de la Información, en donde entra en juego la información personal, donde la información almacenada será en Firebase y una base de datos local, aplicando todas las reglas de la ISO para el resguardo de los datos como nombres, contraseñas, imágenes, videos, etc.

3.2 INSTALACIÓN DE Y CONFIGURACIÓN DEL SERVIDOR MAESTRO.

Primeramente, se instala el sistema operativo en el servidor, para este caso la computadora cuenta con un hardware de:

- Procesador Intel Core i5-8500 6 núcleos
- 16 GB de RAM DDR4 SDRAM hasta 2666 MHz
- Unidad de almacenamiento M.2 SSD de 512 GB (Opción a 4 expansiones más como máximo de 2 TB).
- Gráficos Intel UHD 630
- 2 puertos USB 3.1
- 1 puerto tipo C

Antes de iniciar la instalación se deberá crear una memoria flash extraíble USB para iniciar la instalación de la imagen ISO del sistema operativo, en este caso es "Fedora Workstation 38". Conectar la memoria en el servidor para que bootee como dispositivo principal y haga inicio a la instalación de Fedora Workstation. Se debe de elegir en que unidad se va a instalar el sistema y crear un usuario para el uso del servidor/maquina.

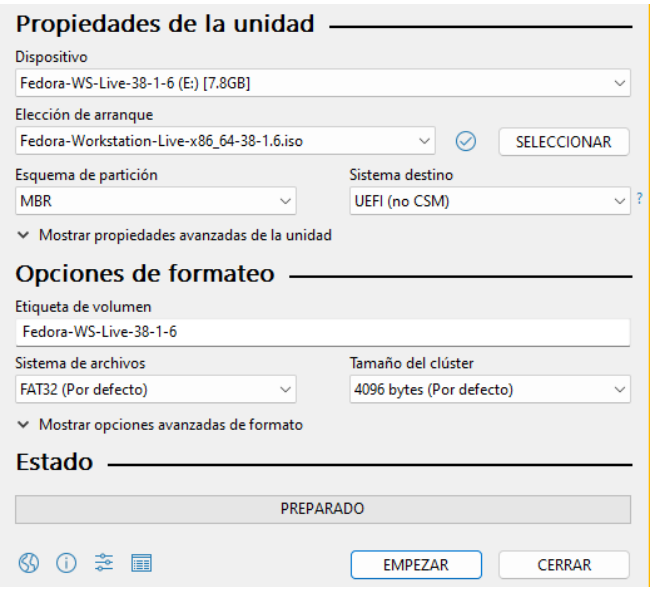

Figura 2:Preparación de la memoria para el boot del sistema operativo

Al finalizar la instalación se debe de configurar una contraseña para el usuario root que es el usuario administrador del sistema operativo por defecto.

$ sudo passwd root

Nota: cuando el comando se usa un carácter de dólares (“$”) está en modo usuario normal, cuando se pone un signo de gato (“#”) está en usuario root ósea que se tiene permiso total de hacer cambios en el sistema.

Con este proceso se asigna una contraseña para el usuario root y permitir cambios en el sistema se debe de iniciar una nueva terminal para actualizar el sistema y los diferentes drivers y/o aplicaciones que tiene el sistema operativo en usuario root.

$ su

Para hacer esta opción se debe de acceder a modo root o super usuario, se debe de ingresar la contraseña del usuario.

 # dnf update

Al hacer este proceso da apertura a instalar los programas, complementos y/o herramientas que se deben de ocupas. Ya actualizado el sistema se instalan las herramientas, como primera herramienta es “Wireshark” el cual analiza los diferentes protocolos y los paquetes de redes que hay en el entorno y en la máquina.

 # dnf install wireshark

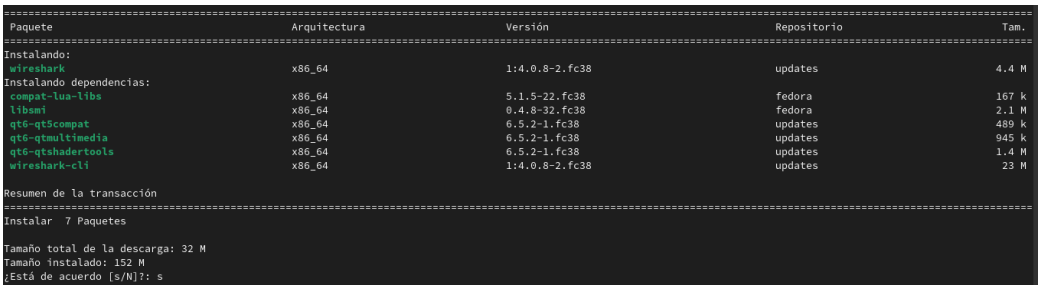

Figura 3: Instalación de Wireshark en el servidor maestro

A continuación, se debe de dar los permisos necesarios a todos los usuarios, ya que el programa debe tener el acceso necesario para obtener el monitoreo de los paquetes, Wireshark mostrara todos los puertos/interfaces activos, ya sean físicos o virtuales.

#usermod -aG wireshark [usuario]

```
[root@fedora compressoftsys]# usermod -aG wireshark compressoftsys
[root@fedora compressoftsys]# usermod -aG wireshark root
```

Figura 4: Ejecución de los comandos para dar permiso a Wireshark en todos los usuarios existentes

3.3 CONFIGURACIÓN DE LOS PUERTOS VIRTUALES PARA LA CONEXIÓN ENTRE LOS NODOS VIRTUALES Y EL NODO MAESTRO

Al elegir que los nodos van a hacer virtualizados se tienen diferentes alternativas para ejecutarlos, haciendo uso de las máquinas virtuales (VM por sus siglas en ingles) y entre los diferentes entornos para hacer uso de una VM como, por ejemplo: VMware Workstation, Oracle VM VirtualBox y Gnome Boxes. Al tomar en cuenta las soluciones se decide por usar Boxes, ya que es un entorno desarrollado para Linux, así evitando algunos temas de compatibilidad.

Para instalar Boxes, se puede obtener desde la tienda de aplicaciones o mediante terminal, para este caso se obtiene desde la tienda ya que se usa Fedora Workstation y por términos de practicidad se hace así.

Figura 5: Gnome Boxes en la tienda de aplicaciones de Linux Fedora.

Antes de iniciar el proceso de instalación se debe de crear dos redes virtuales que es la de control y almacenamiento, las cuales son necesarias para el protocolo iSCSI, con el administrador de máquinas y buscar la opción de “Crear una red virtual”, al entrar ahí aparece una ventana donde se configura de forma manual o mediante código XML, la configuración se hace de forma manual, se asigna el nombre de “Control”, con el modo “Isolado[1]”, seguido de eso se habilita la configuración IPv4 y se asigna una dirección IP previamente subneteada, es necesario que esta interfaz virtual se deshabilite el DHCP ya que se corre el riesgo de que al conectar un proveedor de internet asigne una IP dinámica o caso contrario el servidor

[1] Isolado: Una red aislada que no está conectada a una física a propósito por motivos de seguridad

maestro le asigne una IP, la red que usara esta interfaz nueva es la 172.16.100.0, con mascara de subred /24 ósea 255.255.255.0

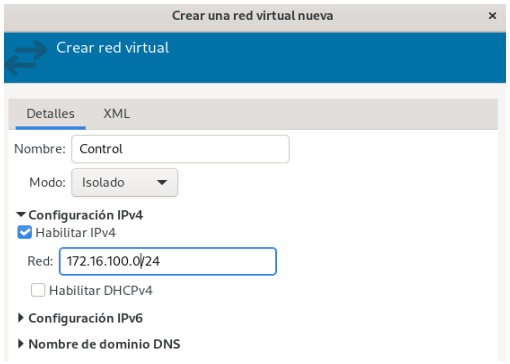

Figura 6: Creación de la red virtual "Control"

Se repite el proceso para crear la red de almacenamiento, el cual por nombre se le asigna "MVAlmacen" de igual forma es en modo Isolado y habilitando la configuración IPv4 asignándole la red 172.16.200.0, con la máscara de subred /24 ósea 255.255.255.0

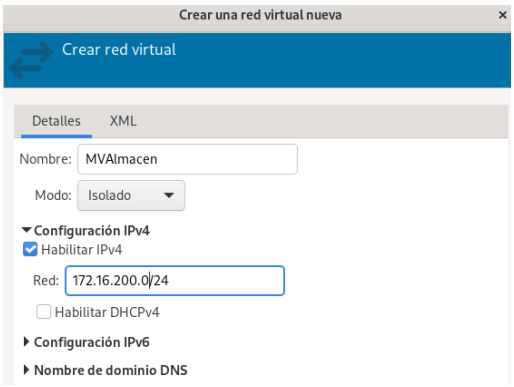

Figura 7: Creación de la red virtual "MVAlamcen"

Se verifica que las redes o interfaces virtuales sea hayan creado, en la ventana de "Detalles de Conexión", debe de aparecer ambas interfaces con su "Mac Address[2]".

[2] Mac Address: Dirección o identificador de 48 bits de algún dispositivo.

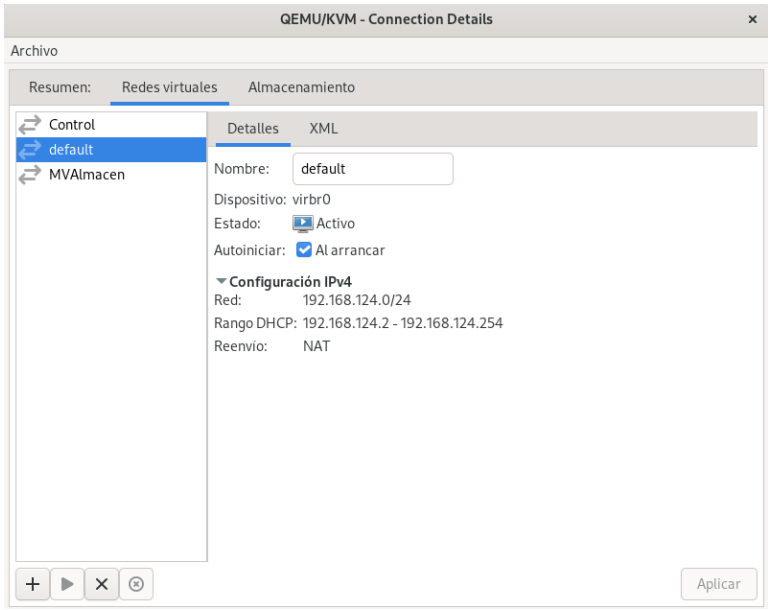

Figura 8: Comprobación de las redes virtuales creadas en el sistema.

Tabla 2: Subneteo de direcciones IPs para los puertos virtuales.

Nombre	Red	Subneteo	Mascara de subred	Rango
MVAlmacen	172.16.0.0	172.16.100.0	255.255.255.0	172.16.100.1 – 172.16.100.254
Control	172.16.0.0	172.16.200.0	255.255.255.0	172.16.200.1 – 172.16.200.254

Para la distribución de los nodos se contempla la siguiente distribución en base a un diagrama de despliegue o también nombrado diagrama de redes, mostrando como se ve el clúster de manera gráfica, además de una idea más clara de cómo debe de ir acomodado el todo, aunque siendo un clúster este llegara siempre al mismo punto deseado

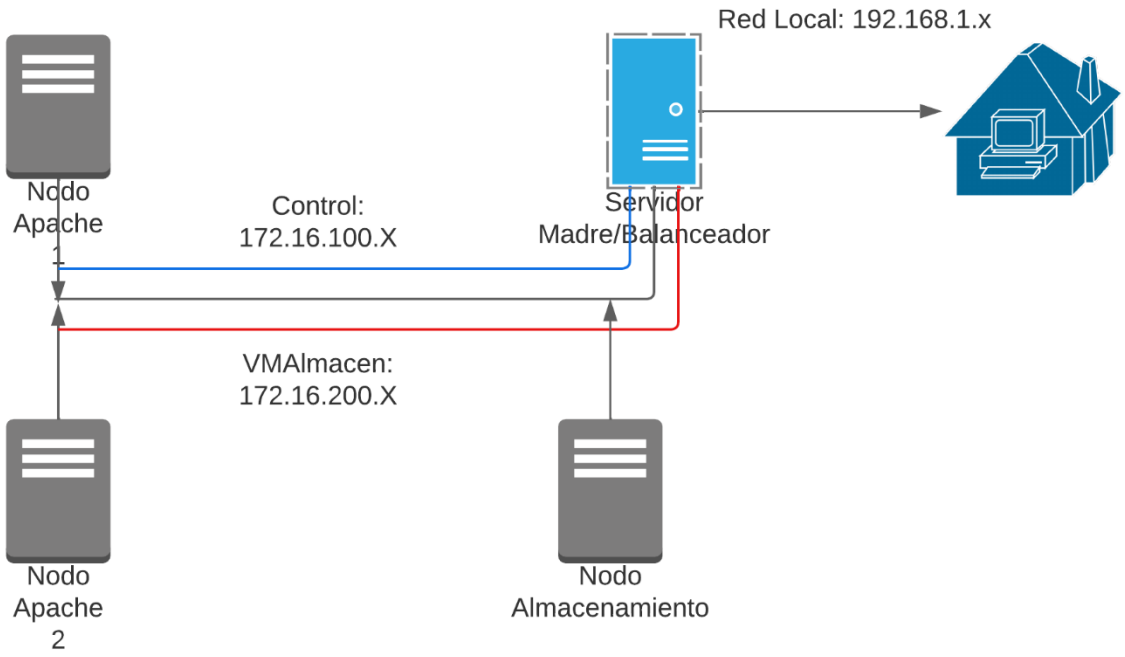

Figura 9: Diagrama de despliegue o de red del clúster propuesto

Para la creación de la máquina virtual se debe de tener previamente la imagen ISO[3] del sistema operativo que se va a usar en este caso es Fedora Server 38, la máquina virtual debería de detectar el archivo, de caso contrario el archivo debe de estar dañado o es el incorrecto.

[3] Imagen ISO: Archivo de una copia exacta de un sistema operativo

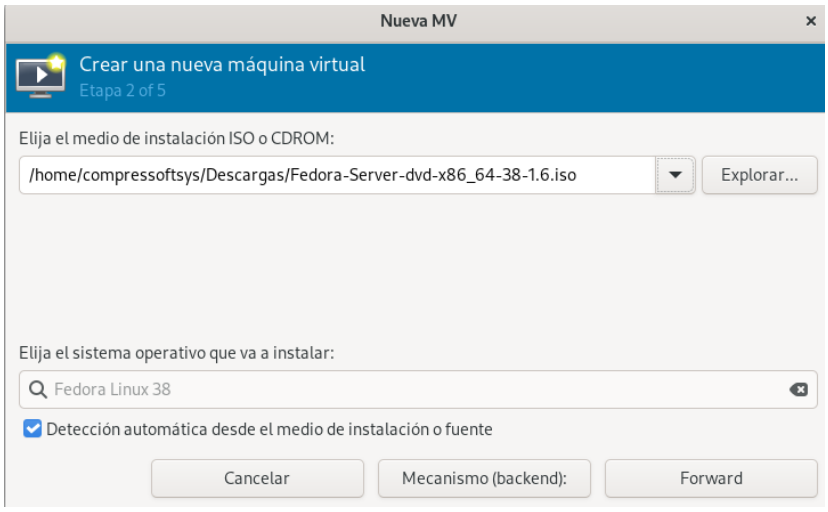

Figura 10: Ventana de creación de una nueva máquina virtual

Seguido de esto, se asigna la cantidad de memoria RAM a virtualizar y la cantidad de núcleos de CPU virtualizados para la máquina virtual durante su ejecución, importante, esto es el consumo máximo de la VM, puede ser que el consumo de esta sea mínimo a menos que se ejecuten tareas pesadas, por norma general para este clúster, los nodos usaran 2 hasta dos núcleos y 2 GB de RAM (2048 MB).

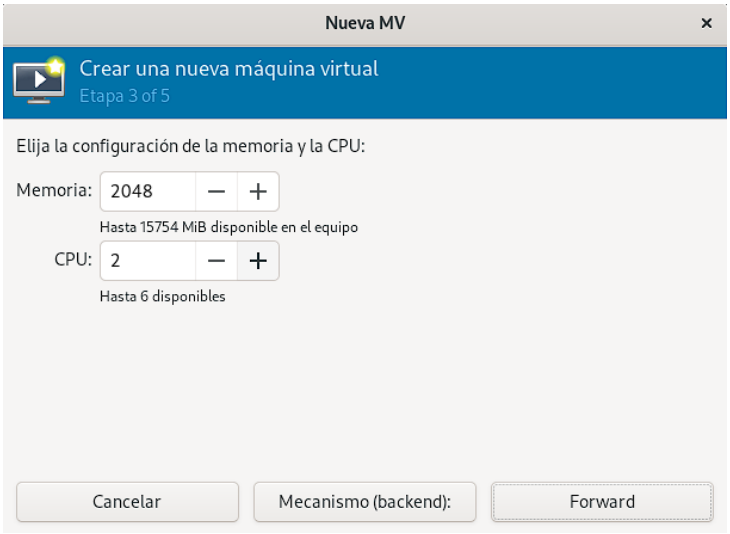

Figura 11: Asignación de los núcleos y memoria RAM de la máquina virtual

Para que la máquina virtual se ejecute al arranque del servidor físico se debe de activar la función de encendido automático. De manera que se tiene que hacer el proceso ingresando desde la configuración de la máquina virtual, se despliega el menú de opciones, de ahí se selecciona “Opciones de arranque” y habrá una zona nombrada “Autoiniciar”, el cual contiene un checkbox desmarcado que dice “Iniciar la máquina virtual cuando arranque el equipo” se marca esa propiedad y se guardan cambios. Cabe mencionar que este proceso se debe de hacer con todas las máquinas virtuales en uso para el clúster.

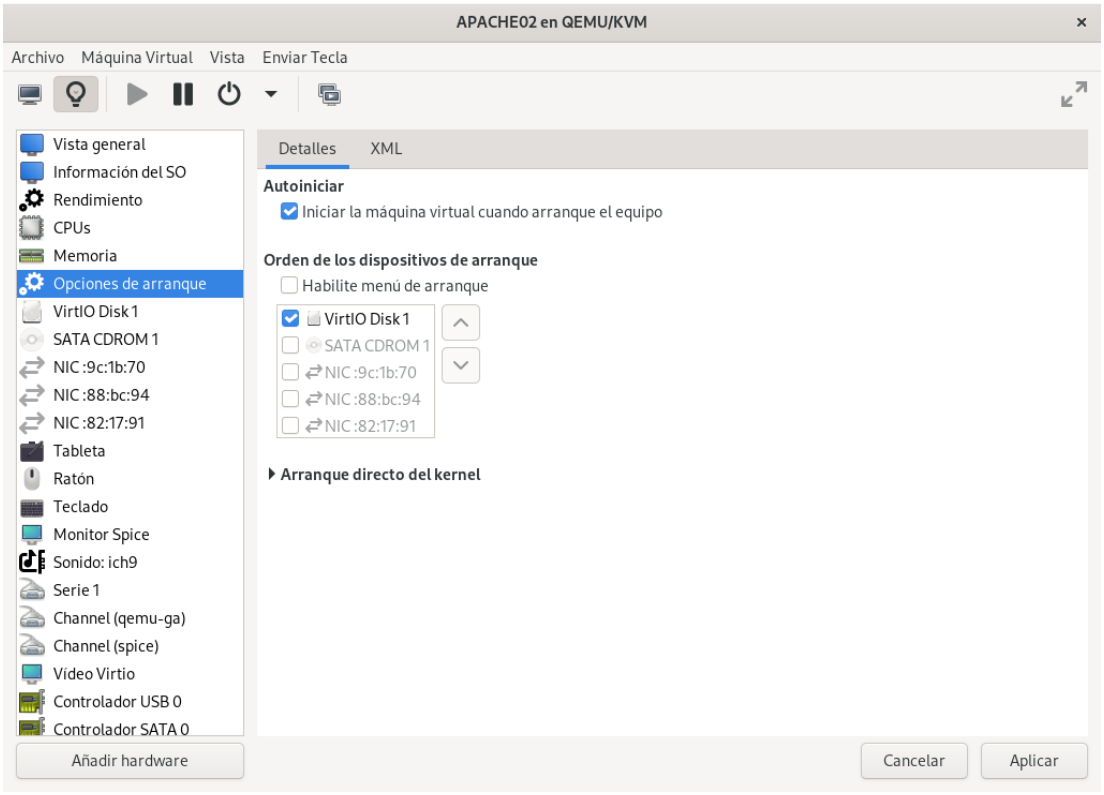

Figura 12: Activación del arranque automático

A continuación, se ejecuta la máquina virtual y esta leerá la imagen ISO del sistema operativo, seguido de la configuración de varios aspectos para la MV: La unidad de almacenamiento, usuarios, idioma, y otros aspectos básicos.

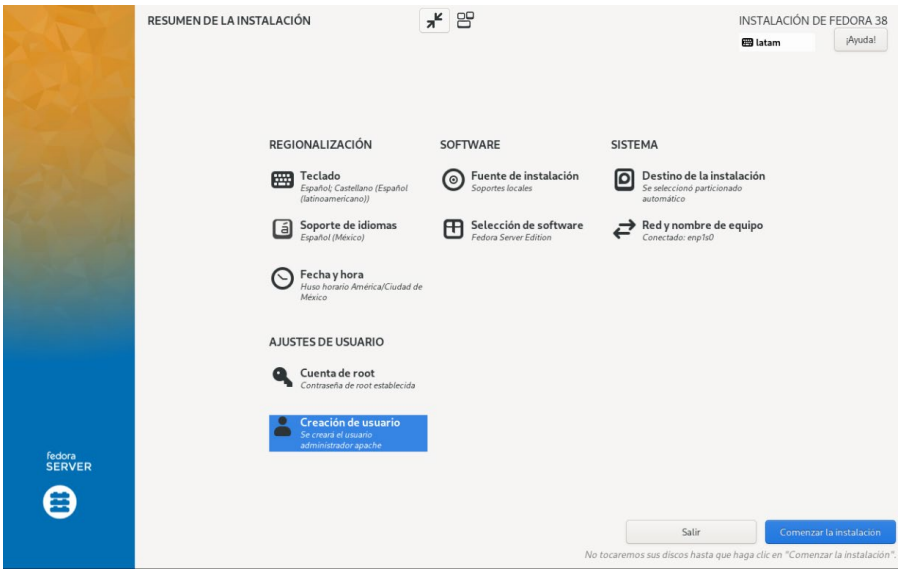

Figura 13: Interfaz de inicio de la instalación de Fedora Server.

Asignar un nombre de usuario y contraseña para manipular el sistema operativo, en este caso se hace referencia al uso que se le dará a este nodo que es para uso de Apache HTTP entonces el nombre de usuario es “Apache” y creando una contraseña para el acceso del usuario en Fedora Server, el nombre dependerá de la máquina virtual que se use o se configure en el momento o que el usuario requiera.

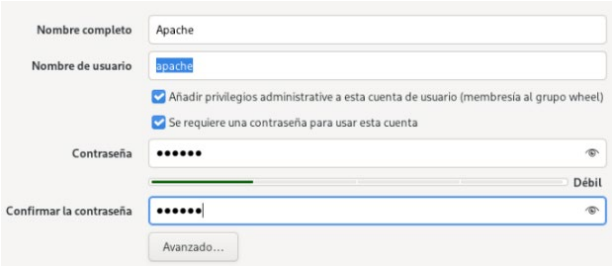

Figura 14: Asignación de nombre de usuario y contraseña del usuario.

Configurar el usuario root, el cual es quien tiene todos los privilegios para hacer cambios en el sistema operativo, ya sea para instalar herramientas, actualizar el sistema, crear una regla nueva en el Firewall[4], entre otras funciones diversas.

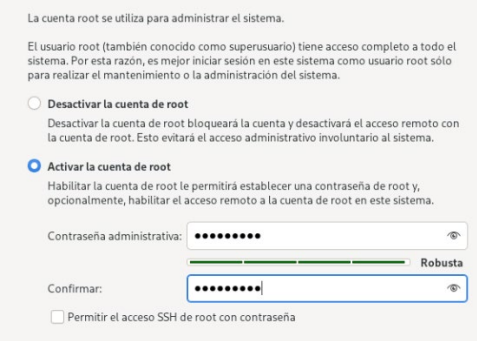

Figura 15: Configuración del usuario root con su contraseña y activando el SSH con root.

Al configurar todas estas opciones empieza el proceso de instalación del sistema operativo en la máquina virtual, este proceso requiere de una conexión a internet ya que baja diferentes paquetes necesarios.

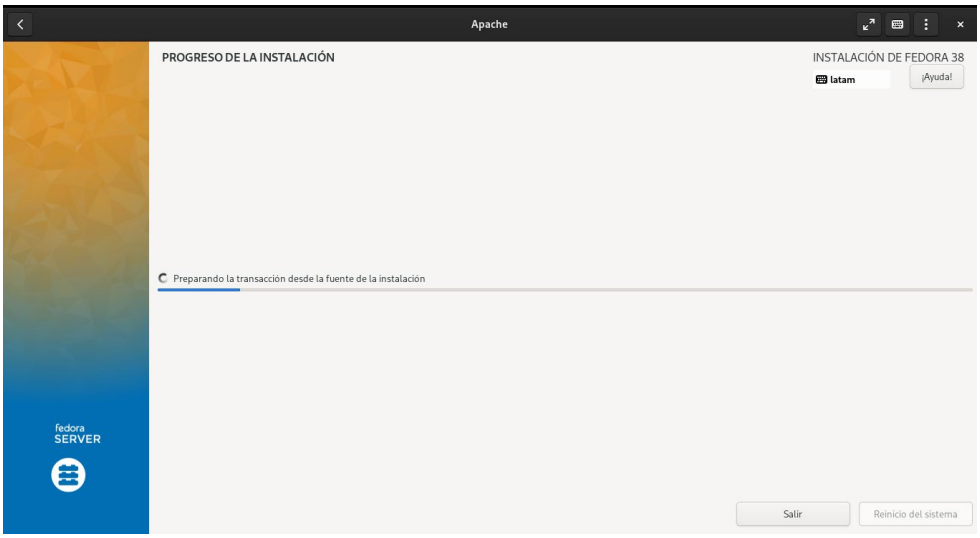

Figura 16: Carga de la configuración e instalación de Fedora Server

Cuando la instalación ha terminado, el sistema solicita que sea reiniciado para hacer el arranque y el sistema operativo iniciara directamente, seguido de esto, se debe de iniciar sesión con la cuenta ya sea la registrada previamente o el usuario root directamente, lo

[4] Firewall: Sistema de protección del tráfico de red que entra o sale del sistema operativo o en también dentro de una red privada (LAN).

recomendable para hacer algún cambio es iniciar con el usuario root ya que nada más iniciar el sistema se empezara a hacer cambios al sistema. El primer cambio es la actualización de los diversos paquetes y/o programas que contenga Fedora Server en ese momento usando el siguiente comando:

#dnf update O *#dnf -y update*

El comando "dnf" como explica Fedora (2023) es una herramienta de gestores de paquetes para sus instalaciones y revocación de estos, siendo el sucesor de YUM[5], al actualizar permite al sistema hacer nuevas instalaciones al sistema o usando versiones estables de los paquetes que requiere el sistema operativo para funcionar. Este proceso se debe de ejecutar en cada uno de los nodos ya sea en el nodo maestro o en los nodos virtualizados, ya que muchas herramientas nuevas necesitan que los paquetes estén actualizados.

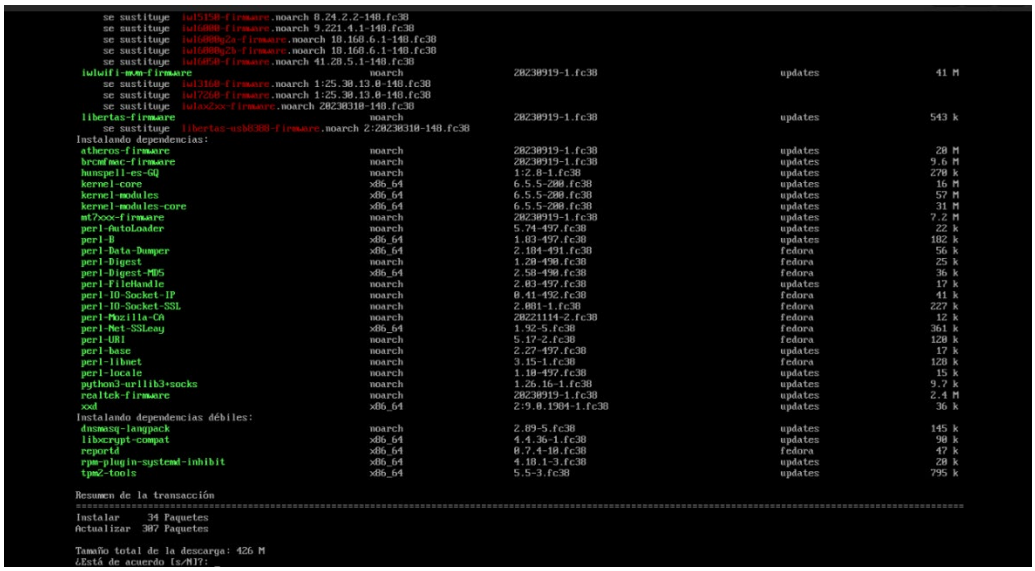

Figura 17: Ejecución del comando "dnf update"

[5] YUM: Por sus siglas en ingles Yellow-Dog Updater Modified, sistema gestor de paquetes para Linux.

El siguiente paso se puede hacer antes o después de la instalación del sistema operativo, al crear la máquina virtual y ejecutarla, se debe de dar de alta las redes virtuales que previamente se crearon, que en este caso es la de “Control” y la de “VMAlmacen”, estos puertos ayudan a la conexión entre los nodos. Al ejecutar la máquina virtual se despliega otra ventana donde está la barra de herramienta de la VM en el cual hay un símbolo de una bombilla en ese se encuentra todo el hardware virtual de esta.

Figura 18: Barra de herramientas de la máquina virtual

Se tiene que agregar un nuevo hardware y se seleccionan una de las dos redes virtuales que previamente han sido creadas.

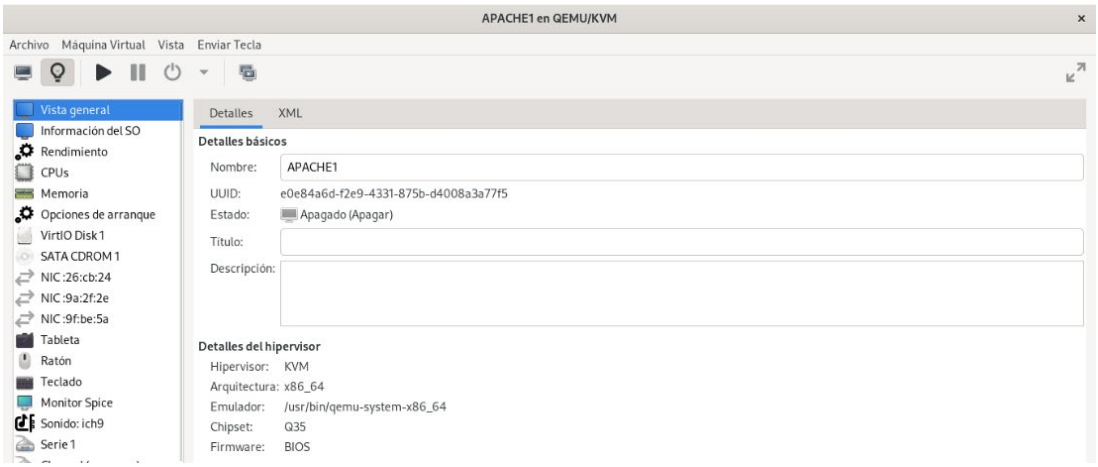

Figura 19: Hardware virtual o Hardware de la máquina virtual.

Cada interfaz nueva tiene que ser añadida ya que cada nueva interfaz de red tendrá su propia mascará de subred.

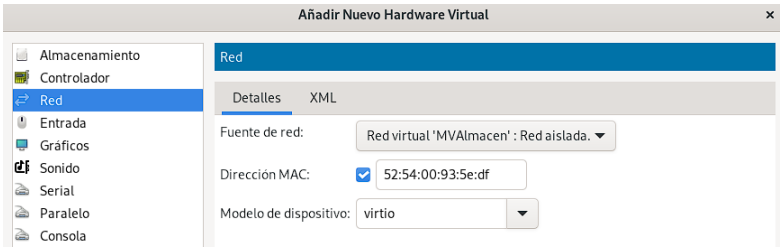

Figura 20: Adición de la red virtual "MVAlmacen"

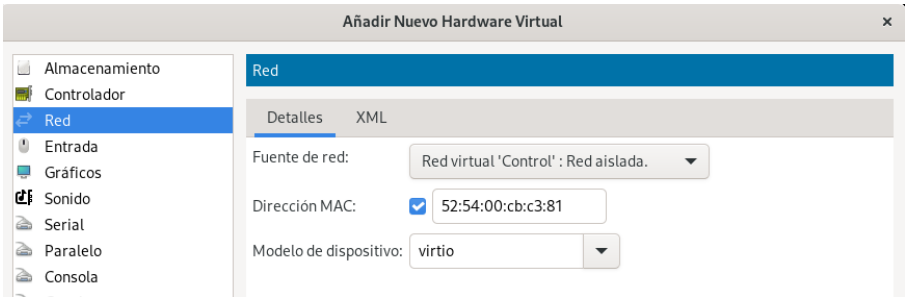

Figura 21: Adición de la red virtual "Control"

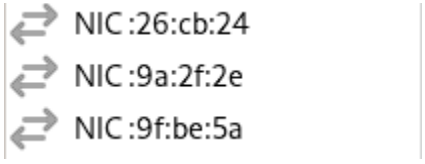

Figura 22: Identificador de los dispositivos

Al activar estas interfaces se crean unas virtuales por las cuales funcionan como puentes entre los nodos del clúster, también este procedimiento se puede utilizar para crear una interfaz virtual de una entrada USB, todo este procedimiento se tiene que repetir en todas las máquinas virtuales (nodos) que estén en uso ya que no se podrán conectar entre sí, algo a tomar en cuenta y como recomendación de practicidad es anotar las direcciones MAC para rectificar dentro del sistema operativo.

3.4 CONEXIÓN ENTRE LOS NODOS DEL CLÚSTER

El protocolo iSCSI consta de la conexión de los nodos mediante una red LAN el cual puede ser cualquier esquema de subredes, así como se muestra en la Tabla 2, se usa una red de almacenamiento y una red de control, el protocolo de almacenamiento define como se transfieren los datos entre los nodos del clúster a través de una red TCP/IP, la transferencia de estos datos puede ser de forma física o de forma virtual mediante puertos virtuales, además de que este protocolo permite el cifrado de paquetes de red.

Según Huawei y colaboradores Huawei (2020), estas tecnologías empaquetan datos SCSI en paquetes de red para el destino del almacenamiento, además menciona este protocolo puede funcionar hasta las tecnologías subyacentes de hasta 10GbE.

Para el enlace de cada máquina virtual y el nodo madre o nodo principal se debe de hacer dos LANS, como se menciona en los pasos anteriores cada paso debe de hacerse en cada uno de los nodos del clúster, primero se debe de verificar si las interfaces virtuales han sido creadas correctamente con el comando nmcli, el cual permite ver las interfaces de red ya sean físicas y/o virtuales con el que cuenta la máquina.

Para crear los puertos o los objetos iSCSI se requiere instalar en todas las maquinas que se usan como nodos el servicio “Targetcli” el cual permite crear interconexiones entre estos objetos.

 # dnf install targetcli

```
[root@localhost-live firebase1]# dnf install targetcli
Última comprobación de caducidad de metadatos hecha hace 0:09:35, el mar 10 oct 2023 17:20:54.
Dependencias resueltas.
================================================================================================================================================
 Paquete                                 Arquitectura                 Versión                                 Repositorio                  Tam.
================================================================================================================================================
Instalando:
 targetcli                               noarch                       2.1.54-7.fc38                           fedora                       93 k
Instalando dependencias:
 python3-configshell                     noarch                       1:1.1.29-10.fc38                        fedora                       91 k
 python3-kmod                            x86_64                       0.9-36.fc38                             fedora                       81 k
 python3-pyparsing                       noarch                       3.0.9-3.fc38                            fedora                      262 k
 python3-rtslib                          noarch                       2.1.75-4.fc38                           fedora                      128 k
 python3-urwid                           x86_64                       2.1.2-8.fc38                            fedora                      940 k
 target-restore                          noarch                       2.1.75-4.fc38                           fedora                       14 k

Resumen de la transacción
================================================================================================================================================
Instalar  7 Paquetes

Tamaño total de la descarga: 1.6 M
Tamaño instalado: 6.5 M
¿Está de acuerdo [s/N]?:
```

Figura 23: Instalación de Targetcli

Además, para que el servicio se ejecute al iniciar la máquina virtual se debe de ejecutar con los comandos start y enable respectivamente ya que uno inicia el servicio en ese momento y el otro. Estos comandos deben de ser ejecutados con obligación

*# systemctl start *nombre del proceso**

*#systemctl enable *nombre del proceso**

A su vez existe otro comando llamado status cual su función es mostrar en pantalla el estado del proceso, información de interés como versión, librerías, espacio, etcétera, también sirve como un pequeño debug ya que marca errores puntuales en la configuración del servicio.

*# systemctl status *nombre del proceso**

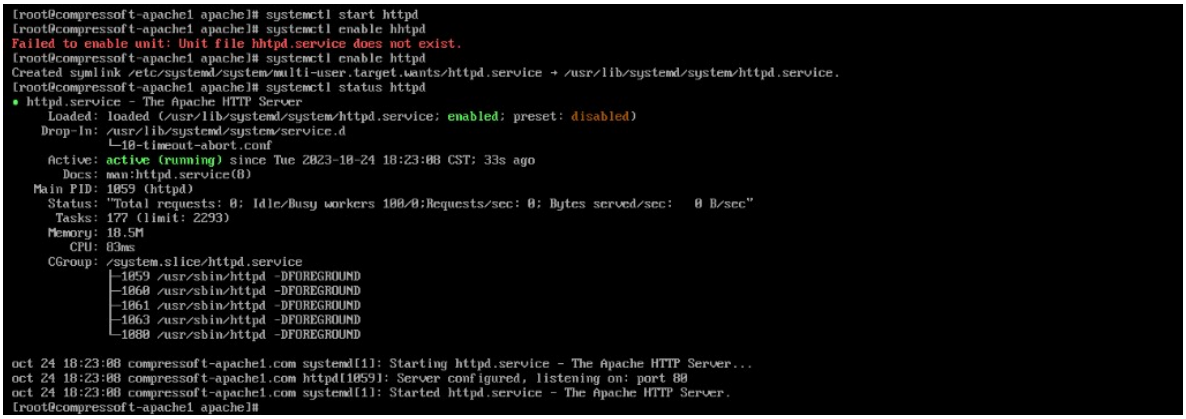

Figura 24: Ejemplo de ejecución de comandos start, status y enable.

Mencionando la parte de systemctl, este se encarga de administrar y controlar el sistema y los servicios del sistema operativo en la distribución de Unix en este caso Linux Fedora.

Los pasos para crear las interfaces virtuales usando Gnome Boxes para Linux se describió anteriormente, cuando esas interfaces se han ejecutado con éxito deberían de aparecer en el sistema operativo ya sea en Fedora Server y en Fedora Workstation, en la terminal de Linux o de manera visual en una interfaz gráfica dependiendo del SO.

 # nmcli device status.

Figura 25: Demostración de los dispositivos de red en la máquina.

Se debe añadir un nombre a la interfaz para identificar cual red va a ser la correspondiente, para este caso usando "VMAlmacen-(Nombre de por defecto de interfaz)" y "control-(Nombre de por defecto de interfaz)" y usando el comando "con-name" e indicando el tipo de interfaz si es ethernet, o WLAN

#nmcli connection add con-name VMAlmacen-enp7s0 if name enp7s0 ethernet

 # nmcli connection add con-name control-enp8s0 if name enp8s0 ethernet

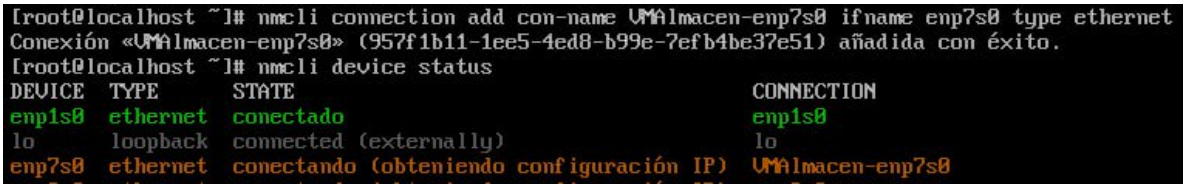

Figura 26: Configuración de las interfaces asignándoles un nombre.

Después se debe de asignar la red que se va a usar en ambos enlaces, así como se indica en la Tabla 2, usando una ipv4 donde la dirección 172.16.100.1/28 para la red de almacenamiento y la dirección 172.16.200.1/28 para la red de control, se debe de hacer una modificación de la conexión, la dirección ip en el comando debe de ingresarse con una comilla simple y la máscara de subred seguida de la ip

 # nmcli connection modify VMAlmacen-enp7s0 ipv4.addresses '172.16.100.1/28' connection.autoconnect yes ipv4.method manual

 # nmcli connection modify control-enp8s0 ipv4.addresses '172.16.200.1/28' connection.autoconnect yes ipv4.method manual

Los comandos mencionados sirven para asignarles las ips, repitiendo el proceso en todas las maquinas ya que las interfaces y las direcciones son las mismas.

Figura 27: Ejecución de comando para asignación de dirección ip en las interfaces.

Después de este proceso las interfaces virtuales que se generaron tendrán una dirección ip asignada, esta se puede ver en la zona de "nmcli device" el cual anteriormente se mencionó que muestra los dispositivos y/o interfaces de red, usando el comando para ver todos estos. Tanto la red de control y la red de almacenamiento debería de aparecer en los dispositivos, además estos son fáciles de identificar ya que se da por entendido los nombres previamente asignados entre enp7s0 y enp8s0, las cuales son las redes virtuales por donde los nodos hicieron su conexión para tener comunicación.

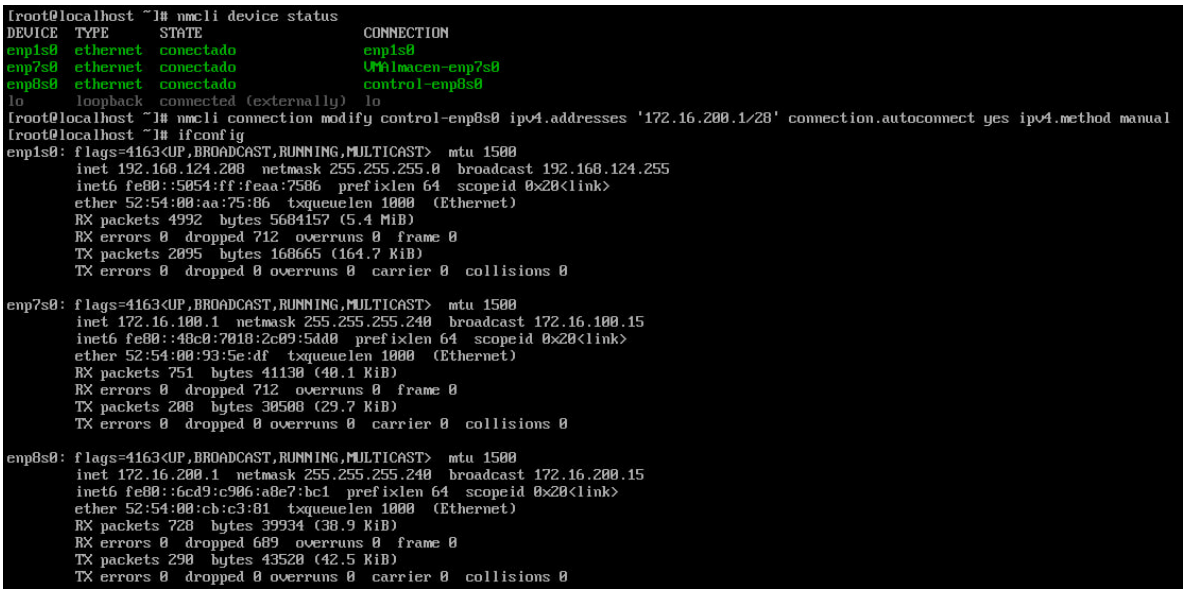

Figura 28: Dispositivos configurados en cada una de las máquinas virtuales

Para comprobar la conexión correcta entre nodos se tiene que hacer un ping entre los nodos, usando las direcciones dinámicas que asigno el nodo maestro, este proceso se en la terminal correspondiente de cada nodo.

#ping xxxx.xxxx.xxxx.xxxxx

El comando ping es usado para comprobar la conectividad y la latencia que existen entre las conexiones generadas entre los dispositivos, en este caso los nodos y las "interfaces" de las conexiones de los nodos. Esta comprobación validara si el proceso se hizo

correctamente de caso contrario se tiene que repetir todo el proceso o hacer la verificación si hay un paso mal ya que esto es de suma importancia si no el clúster no tendrá comunicación.

Primero se hace la prueba de conectividad entre las interfaces desde cualquier nodo del clúster hacia cualquier interfaz virtual, introduciendo el comando ping y las direcciones ip que se utilizaron en el Subneteo, si existe conexión debería de aparecer los paquetes ICMP [6]y el tiempo en el que tardo la transferencia.

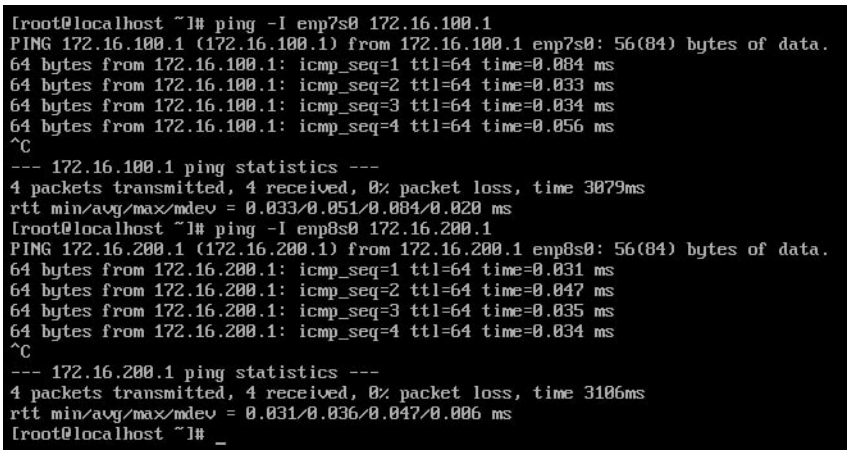

Figura 29: Ping hacia las conexiones de los nodos (almacén y Control)

Seguido se tiene que repetir el procedimiento, pero esta vez con otro nodo del clúster, igual tiene que ser de punto a punto poniendo la dirección de otro nodo del clúster.

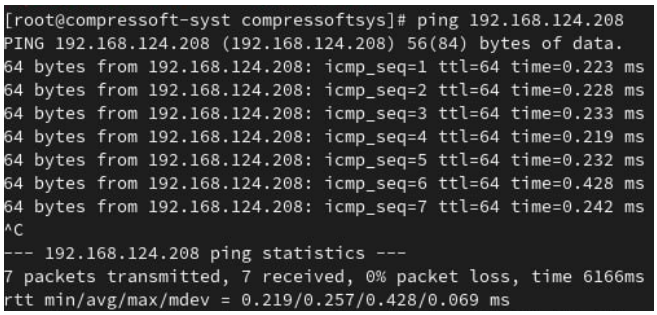

Figura 30: Ping entre el Nodo Maestro y otro nodo del clúster.

[6] Paquetes ICMP: Protocolo de control que se usa para enviar errores y la información que hay entre la conectividad las redes

3.5 INSTALACIÓN DE APACHE Y FUNCIONAMIENTO DE LA PÁGINA WEB EN LOS NODOS

Apache recibe el nombre de HTTPD en la distribución de Linux Fedora ya sea para Fedora Server o Fedora Workstation, el método de instalación de este es mediante la terminal. Como toda instalación se deben de buscar actualizaciones nuevas del sistema

 # dnf update

Después para instalar el servicio de Apache se tiene que ejecutar en usuario root ya que este hará cambios importantes al sistema operativo.

 # dnf install httpd

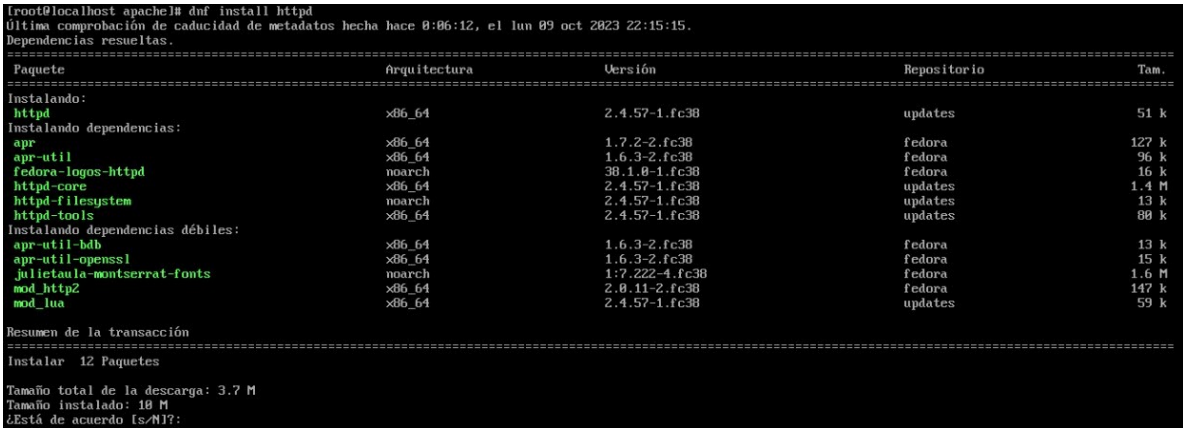

Figura 31: Ejecución del comando de dnf install httpd

Hay varios métodos para transferir archivos o cargar archivos dentro de las máquinas virtuales, usando aplicaciones FTP, montar un USB, etcétera. Para este proceso se selecciona el proceso mediante la montura de una USB en el sistema operativo de la máquina virtual de Apache, se usa el comando “mount” y “umount” este comando debe de ser ejecutado con un usuario root.

Primeramente, se tiene que buscar la etiqueta de la USB en el sistema operativo usando “lsblk”, este comando indica como esta nombrada la USB.

$ lsblk ó # lsblk

La etiqueta asignada por el sistema operativo es “SBA” el cual se usa para montar el USB. Siguiente a esto se debe de crear una carpeta ya sea temporal o una carpeta

permanente para montar los archivos del almacenamiento externo, usando el comando “mkdir” para crear una nueva carpeta en la raíz del sistema o del disco duro principal, en este caso la carpeta se le asignara de la siguiente manera: “/mnt/usb/”. El motivo de montar una memoria en el sistema operativo es porque Linux Fedora Server no maneja un etiquetado a sus dispositivos de almacenamiento, siendo así que el dispositivo único que reconoce como tal es el disco duro principal, aunque este proceso también puede funcionar para un sistema operativo de Linux con interfaz gráfica, en este si se puede observar la USB conectada y acceder al archivo, así como pasa en sistemas operativos Windows o MacOS.

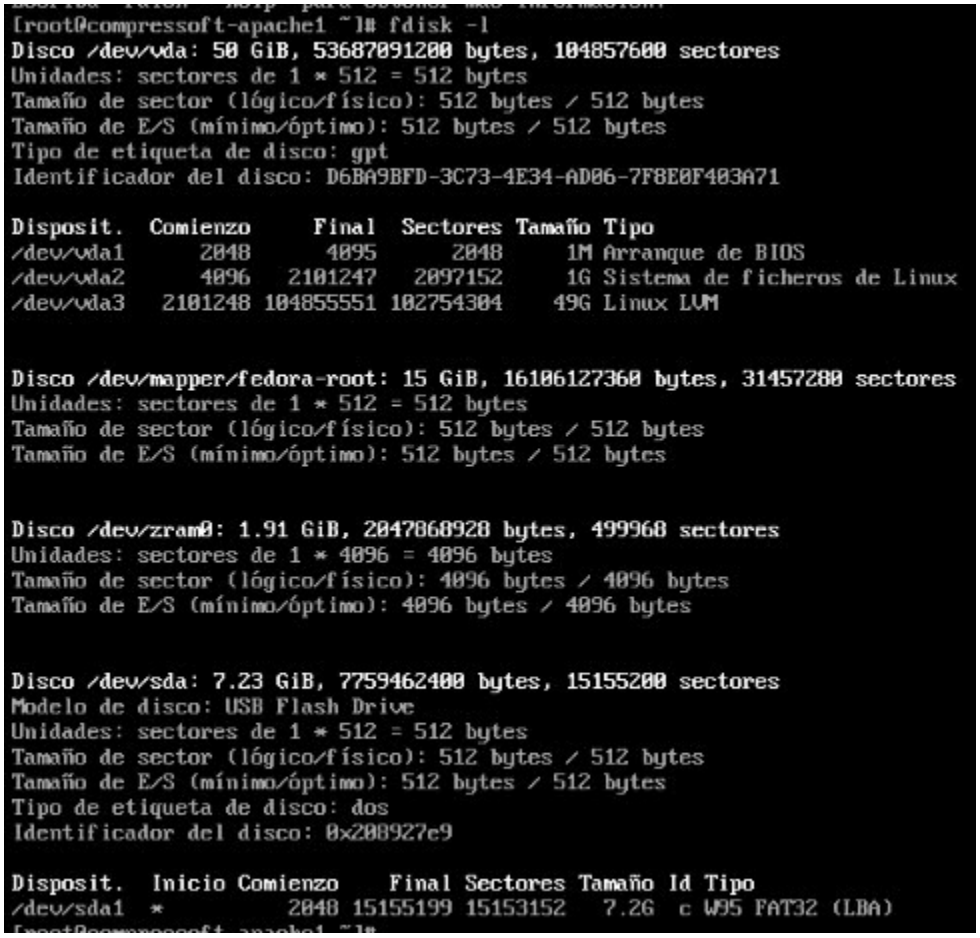

Figura 32: Muestreo de las etiquetas de los dispositivos o USB conectados en el sistema operativo

 # mkdir /mnt/usb/

Al crear este directorio, se obtiene un lugar de “ejecución” del USB, entonces por consecuencia se debe de hacer el proceso de montado con mount, previamente explicado, lo

recomendado es que los archivos estén sueltos en la raíz de la USB ya que será el proceso más ágil, sin tener que agregar rutas especificas en el comando de la terminal, para montar se debe de estar en usuario root o permisos root, ya que directamente se está modificando una ruta del sistema operativo, la estructura del comando es la palabra “mount” donde es la instrucción, /dev/ y la etiqueta del dispositivo, después un espacio y la ruta que se creó anteriormente para almacenar los datos de la USB

 # mount /dev/sba1 /mnt/usb/

Y para desmontar el dispositivo (aunque no pasa nada si no se desmonta), se debe de hacer con la instrucción “umount” seguido de la ruta donde se montó el USB

 # umount /mnt/usb/

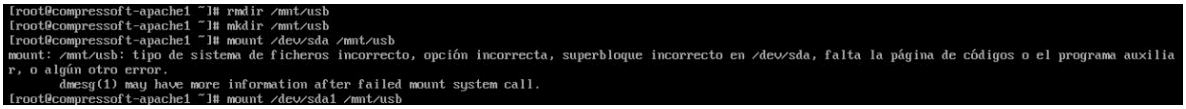

Figura 33: Montura y creación de la carpeta para acceder a la USB

Ya con la memoria flash montada, se deben de traspasar o copiar los archivos de la página web, como se mencionó antes de preferencia estos deben de estar en la raíz del almacenaje externo, entonces teniendo esto en cuenta se usa el comando “cp” y -r indicando que se debe de copiar los directorios con el contenido de ellos, seguido de esto se pone la ruta de origen seguido de un asterisco indicando que serán todos los archivos y después continua un espacio en blanco y complementando con la ruta de destino el cual es donde apache almacena sus páginas.

 # cp -r /mnt/usb/ /var/www/html/*

Este comando hace la función de copiar todos los archivos que están en la USB, así como se menciona por comodidad la página y sus archivos será la que en su totalidad montada para hacer más eficiente el trabajo de traspasar o copiar archivos a fedora server, una observación a tomar en cuenta es que si se usa Fedora Workstation basta con conectar el dispositivo de almacenamiento externo, esta aparece en la interfaz del sistema operativo, por ende solo se deben de arrastrar los archivos a la carpeta deseada (/var/www/html/).

Figura 34: Ejecución del comando de copiado y muestreo de los archivos ya almacenados en la carpeta del SO.

Para el balanceador este se debe de repetir los pasos, ya que este funcionara como una segunda opción, explicando a grandes rasgos en caso de que este nodo se caiga o no esté disponible en ese momento, este segundo nodo entrara como respaldo o alternativa de la carga de la página en línea.

3.6 FUNCIONAMIENTO Y DESCRIPCIÓN GENERAL DEL SISTEMA (PAGINA WEB) MONTADO EN EL CLÚSTER

El sistema consta de una plataforma web diseñada para los alumnos del Tecnológico de Estudios Superiores de Coacalco, para la recopilación de las firmas de adeudo a los diferentes departamentos mediante un código QR y una clave única del registro, este registro se libera cada vez que el departamento correspondiente verifique si el alumno en cuestión tiene o no un adeudo, evitando ir a recopilar las firmas de forma presencial ahorrando tiempo en cuanto traslación y hacer eficiente el proceso de titulación de los alumnos. Los lenguajes de programación que se usan para el sistema en cuestión son los siguientes:

- HTML
- JavaScript
- CSS

Y base de datos Firebase, como se tiene proyectado usar en el clúster.

Como se menciona el sistema cargado en el clúster es una versión de prueba con funciones básicas de inicio de sesión y hacer una solicitud, todo esto con fines de comprar que los nodos del servidor funcionen de manera correcta. A continuación, se muestra el diagrama de caso de uso que interpreta la acción que puede hacer un alumno dentro del sistema.

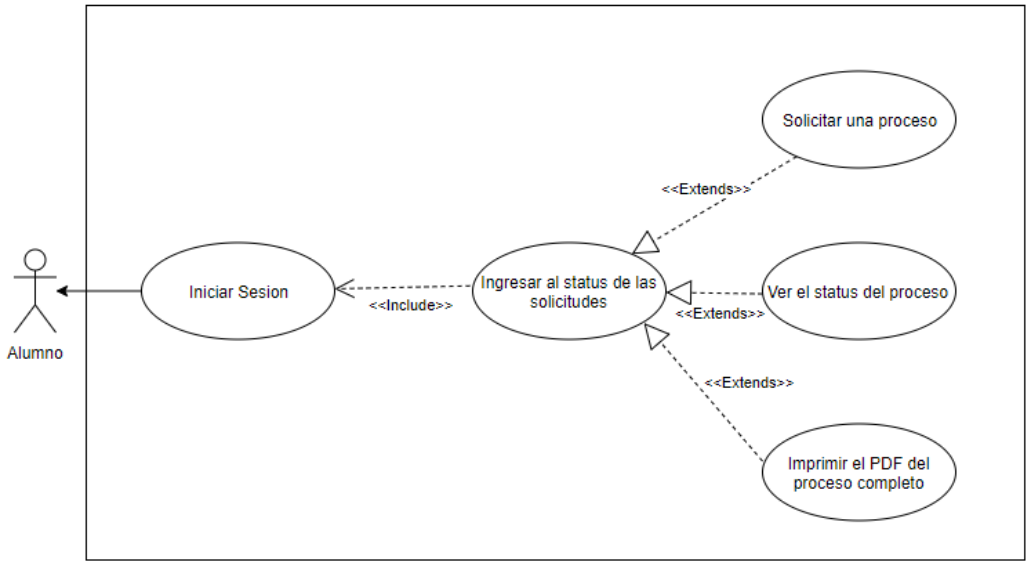

Figura 35: Diagrama de Caso de Uso general del Alumno

INTERFACES EXTERNAS

La conexión entre el alumno y el servidor se establece utilizando protocolos de comunicación estándar, como HTTP o TCP/IP, permitiendo la transferencia de datos de manera segura y confiable. Cuando un alumno realiza una acción en la interfaz de usuario del administrador, como solicitar un nuevo proceso de un trámite o generar el PDF, se generan solicitudes que se envían al servidor a través de la conexión establecida. El servidor procesa estas solicitudes y envía las respuestas correspondientes de vuelta al alumno.

Además de la comunicación entre el alumno y el servidor, el sistema también puede interactuar con otros dispositivos o sistemas externos, mediante la integración de interfaces de comunicación, como API (Application Programming Interface) o servicios web, que permiten el intercambio de datos con otros sistemas, como lectores de códigos de barras, escáneres, códigos QR, entre otros.

FUNCIONES GENERALES

- **Iniciar Sesión:** Esta función es la que permite acceder al alumno al sistema mediante una autentificación usando matricula y contraseña en pocas palabras la verificación de credenciales.
- **Solicitar un trámite:** La función consta de seleccionar entre 3 opciones de tramites, las cuales son: Titulación, Cambio de Carrera y Solicitud de Baja. La recolección y aprobación de los departamentos dependerá de tipo de tramite seleccionado.
- **Imprimir PDF:** La función permite imprimir el PDF generado al completar todas las aprobaciones de los diferentes departamentos donde lo genera con los datos del alumno, el tipo de trámite que se seleccionó y los diferentes QRs validados con una única clave de validación.

REQUISITO 1: INICIO DE SESIÓN

El alumno debe de estar previamente registrado en la base de datos, para que posteriormente este entre a la página web del sistema donde puede iniciar sesión con su correspondiente matricula y contraseña asignada o que el alumno asignó, donde al iniciar sesión llevara directamente a la muestra de status de los diferentes tramites.

CASO DE USO DE AUTENTICAR

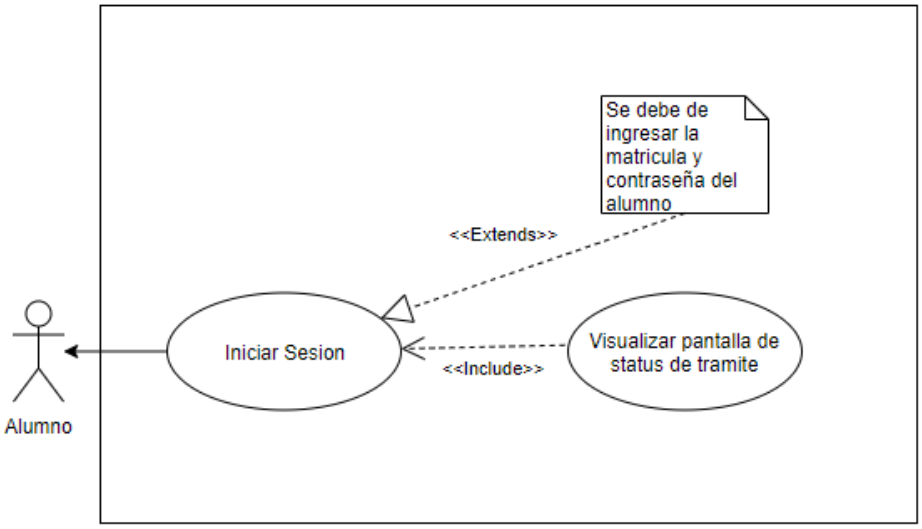

Figura 36: Caso de uso del requisito de Autenticar

ESCENARIO REQUISITO 1 (INICIO DE SESIÓN)

Tabla 3: Escenario Requisito 1 (Inicio de Sesión)

Número de requisito:	1
Nombre del requisito:	Autenticar.
Prioridad:	Alta.
Descripción:	El alumno es un actor del sistema de solicitudes de tramites, el cual debe de ingresar a la plataforma con su matrícula y contraseña.
Actores:	Alumno.
Precondiciones:	• El alumno debió ser registrado previamente en la base de datos de la escuela con su matrícula y una contraseña. • El sistema debe de estar en línea para el acceso con un url
Flujo normal:	1. El alumno visualiza una pantalla de inicio de sesión, donde le

	pedirá ingresar su matrícula y contraseña otorgada. 2. El alumno ingresará sus datos de inicio de sesión, que es su **Matricula** y **Contraseña**, luego deberá presionar el botón de **Ingresar**. 3. Se iniciará sesión y el sistema autenticará si el alumno existe en la base de datos, en caso de que si pasara al paso 4. En caso de que este todo correcto se ingresará al menú principal en donde se visualizan los datos del alumno todos los posibles por hacer o los que están en proceso con su status, en caso contrario se procede al paso 4.1
Flujo alternativo:	4.1. El alumno debe de verificar con control escolar si esta dado de alta en la base de datos
Postcondiciones:	Alumno autenticado.

PROPUESTAS DE DISEÑO DE INICIO DE SESIÓN (FLUJO NORMAL)

Pantalla de Inicio de Sesión e ingreso de Matricula y Contraseña.

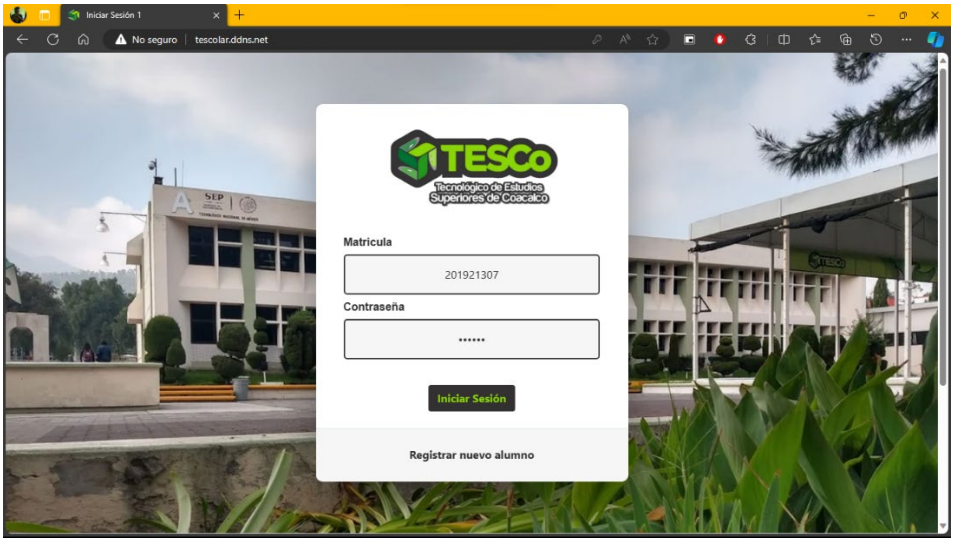

Figura 37: Pantalla de Inicio de Sesión, con el ingreso de Matricula y Contraseña.

Acceso a la pantalla de status e información del Alumno.

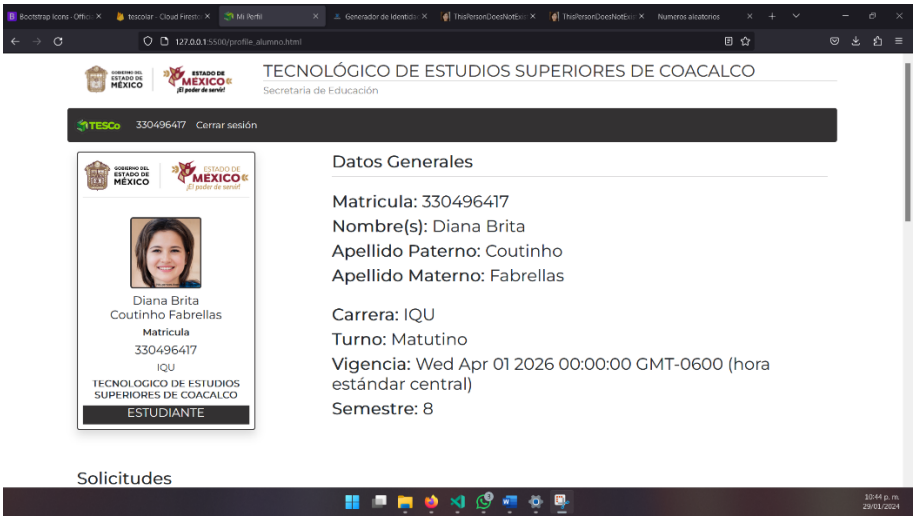

Figura 38: Pantalla de datos y status del alumno.

DIAGRAMA DE SECUENCIA DE INICIO DE SESIÓN

El diagrama de secuencia del proceso de autentificación del sistema montado en el clúster, el alumno ingresa sus datos para entrar al sistema, donde la base de datos corrobora que la matricula este dada de alta en el sistema, de caso contrario se tiene que dirigir a control escolar para que sean dados de alta.

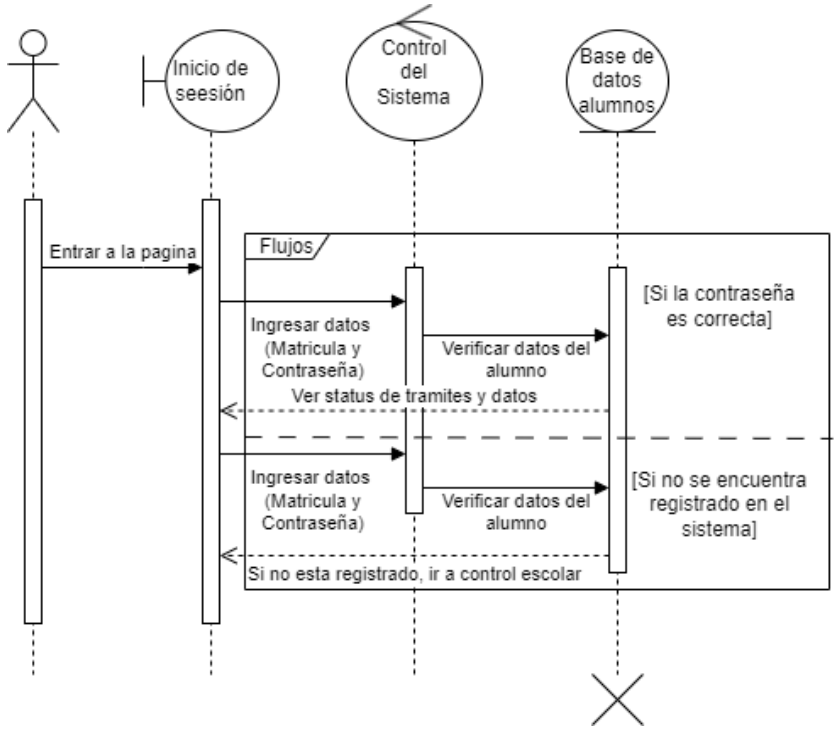

Figura 39: Diagrama de secuencia de Inicio de Sesión

REQUISITO 2: SOLICITAR UN TRAMITE

El alumno puede solicitar un trámite entre los tres disponibles, los cuales son: Titulación, Cambio de carrera y Baja definitiva, donde al seleccionar uno genera un código exclusivo y la solicitud seleccionada se muestra en pantalla con el status de los diferentes departamentos.

CASO DE USO DE SOLICITAR UN TRÁMITE

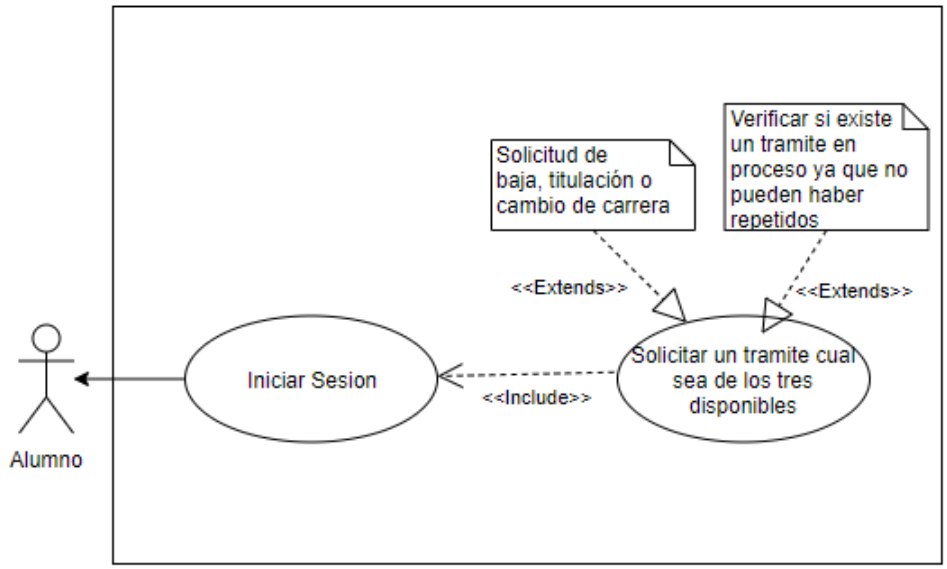

Figura 40: Diagrama de caso de uso de Solicitar un trámite

ESCENARIO REQUISITO 2 (SOLICITAR UN TRAMITE)

Tabla 4: Escenario Requisito 2 (Solicitar un Tramite)

Número de requisito:	2
Nombre del requisito:	Solicitar un trámite.
Prioridad:	Alta.
Descripción:	El Alumno al ingresar al sistema se presenta una pantalla de datos, status de trámites y selección de uno, en la sección de un trámite se despliega una lista para solicitar un trámite.
Actores:	Alumno.
Precondiciones:	• El alumno ingresa con su matrícula y contraseña al sistema. • El sistema debe de estar en línea para el acceso con un link.

Flujo normal:	1. En el proceso de inicio, el estudiante se encuentra con una pantalla que solicita su **matrícula** y **contraseña**. Una vez que introduce estos datos y presiona el botón "Ingresar", accede al sistema. 2. En caso de que este todo correcto se ingresará al menú principal en donde se visualizan los datos del alumno todos los posibles tramites por hacer o los que están en proceso con su status. 3. El alumno despliega la lista de tramites disponibles y selecciona el que desea solicitar en ese momento. 4. El trámite seleccionado se debe de mostrar en el historial de solicitudes con los status del trámite, de caso contrario se prosigue al paso 4.1
Flujo alternativo:	4.1. El alumno debe de verificar si ya hay un trámite igual al que solicito.
Postcondiciones:	Tramite en proceso de aprobación por los departamentos.

PROPUESTAS DE DISEÑO DE SOLICITAR UN TRÁMITE (FLUJO NORMAL)

Pantalla de Inicio de Sesión

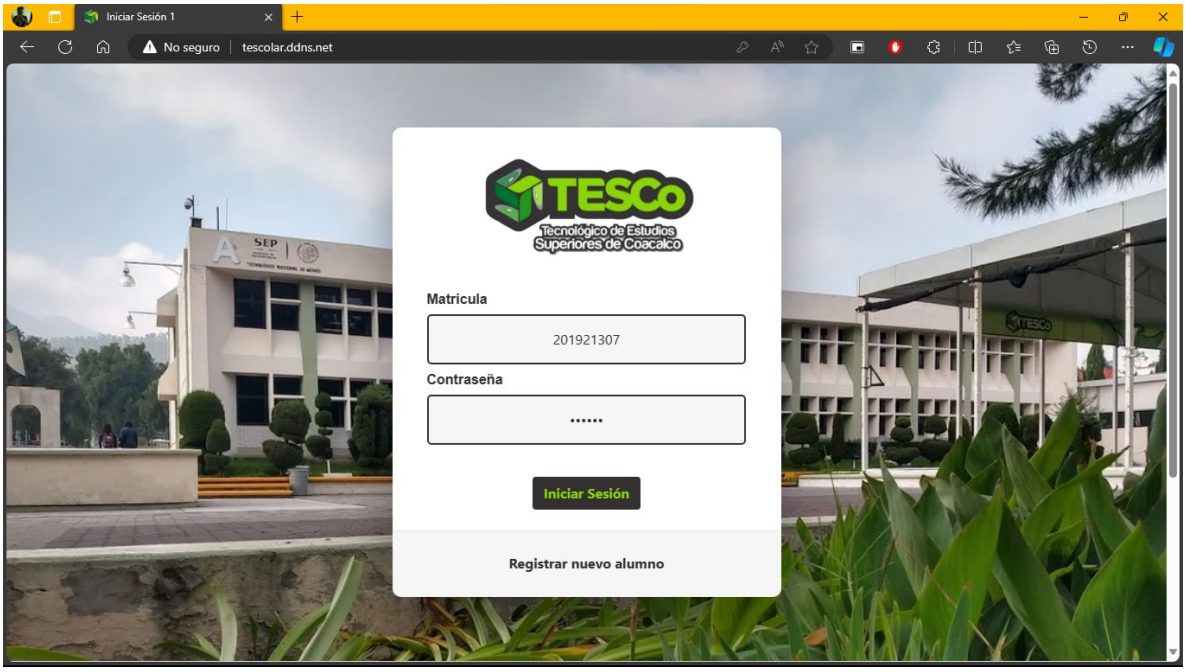

Figura 41: Pantalla de Inicio de Sesión del Alumno.

Acceso a la pantalla de información del Alumno y selección de tramites.

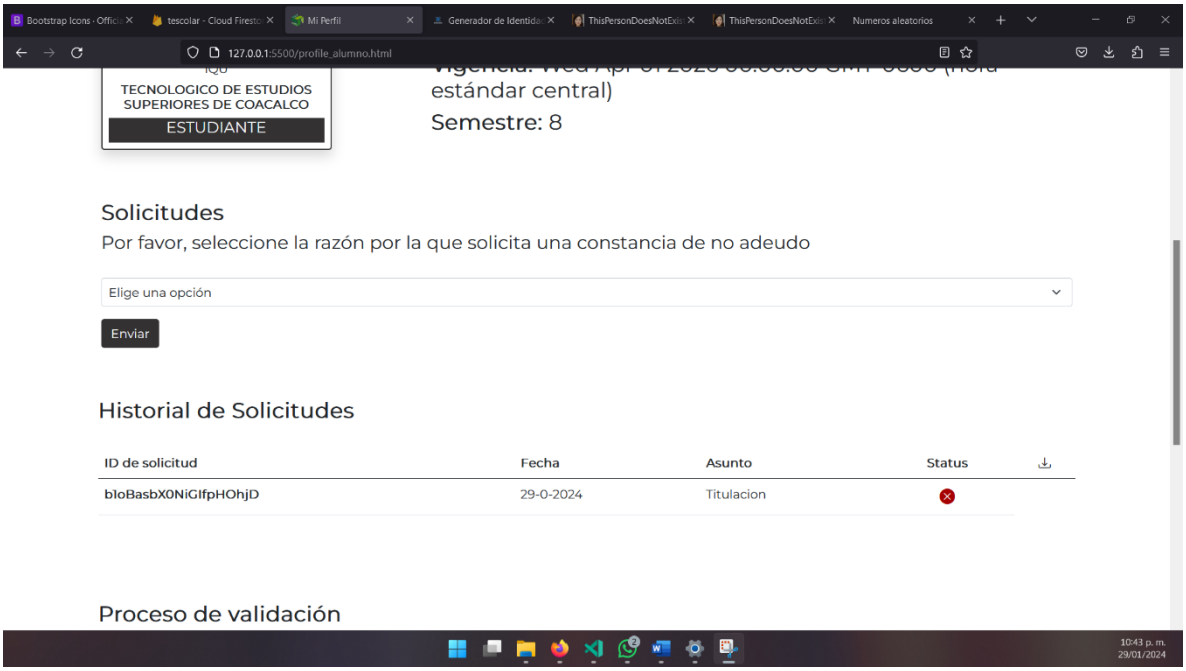

Figura 42: Pantalla de datos y selección de tramites con la lista desplegada.

Pantalla de solicitud aceptada y clave generada aleatoriamente

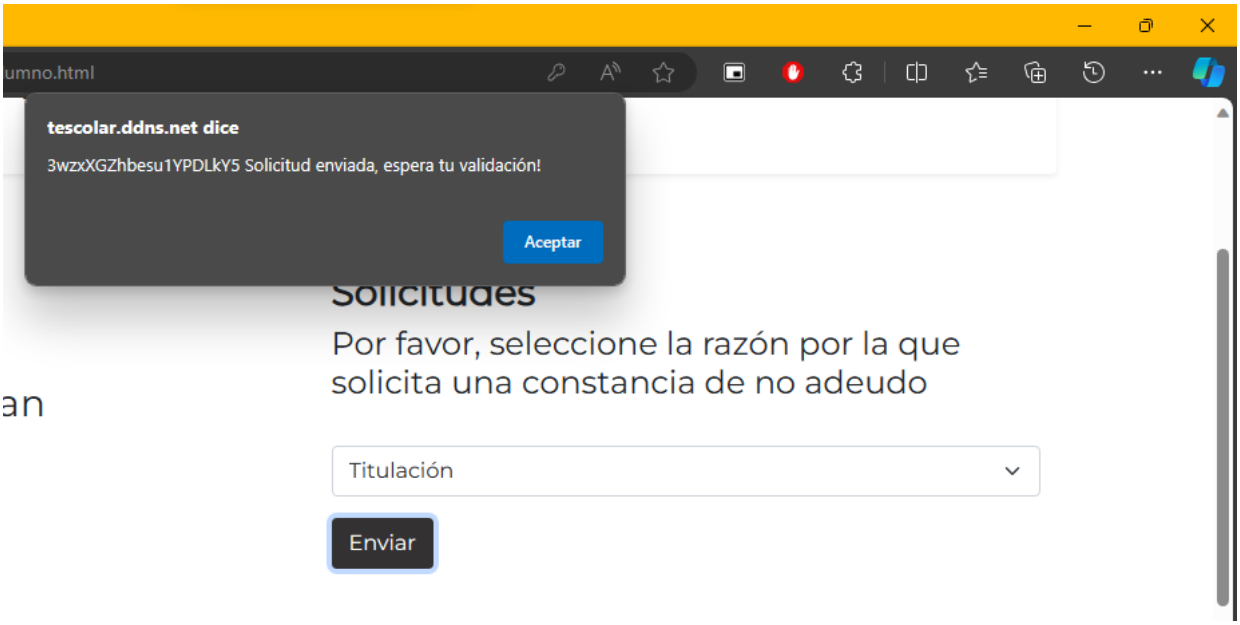

Figura 43: Pantalla de notificación de aceptación del trámite.

Pantalla de status del trámite y muestreo de la clave aleatoria.

Figura 44: Pantalla de status del trámite.

DIAGRAMA DE SECUENCIA DE SOLICITUD DE TRAMITE

El diagrama de secuencia del proceso de selección de tramite consta que el alumno ingresa al sistema con su matrícula y su contraseña, al ingresar puede seleccionar un trámite nuevo para que seguido de este se muestre en la pantalla de status del trámite.

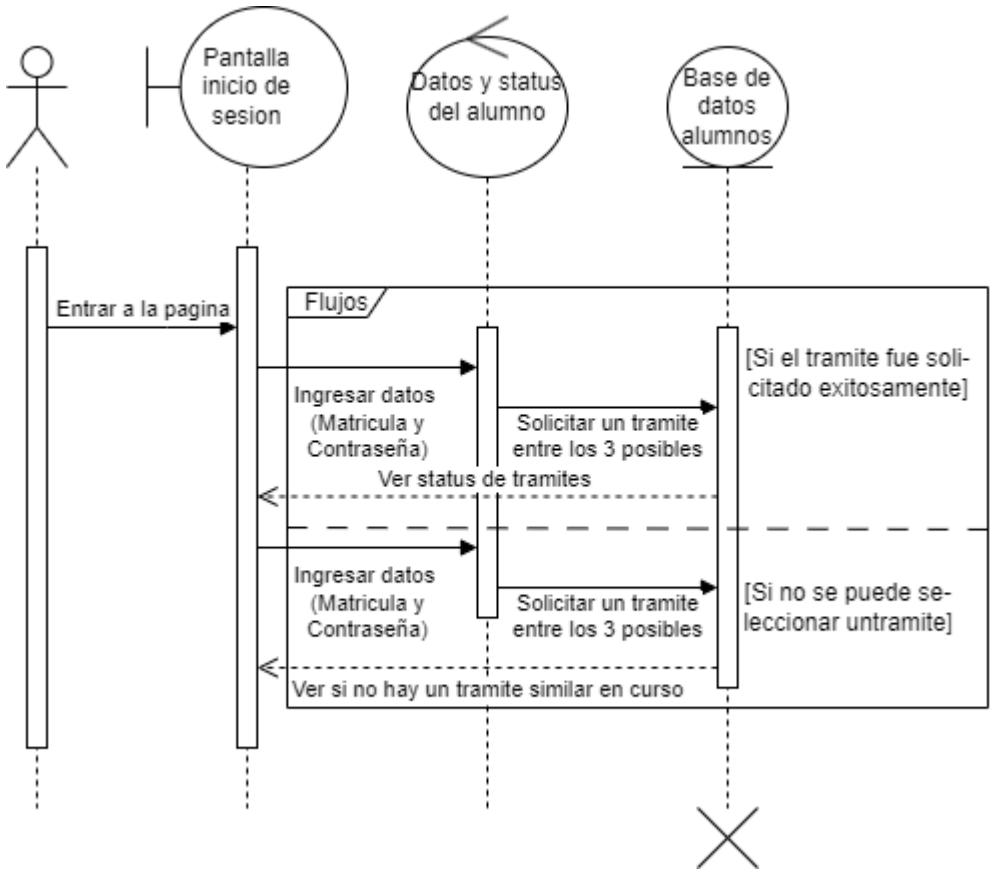

Figura 45: Diagrama de despliegue de solicitud de trámite.

REQUISITO 3: IMPRIMIR PDF

El Alumno puede imprimir el PDF generado por el sistema cuando todos los departamentos hayan validado que no hay ningún adeudo en ellos, aparece un icono de impresión en el a sección de "Historial de Solicitudes", al dar click el PDF se genera de manera instantánea para su impresión en papel o almacenar en algún dispositivo que desee el alumno en cuestión.

CASO DE USO DE IMPRIMIR EL PDF

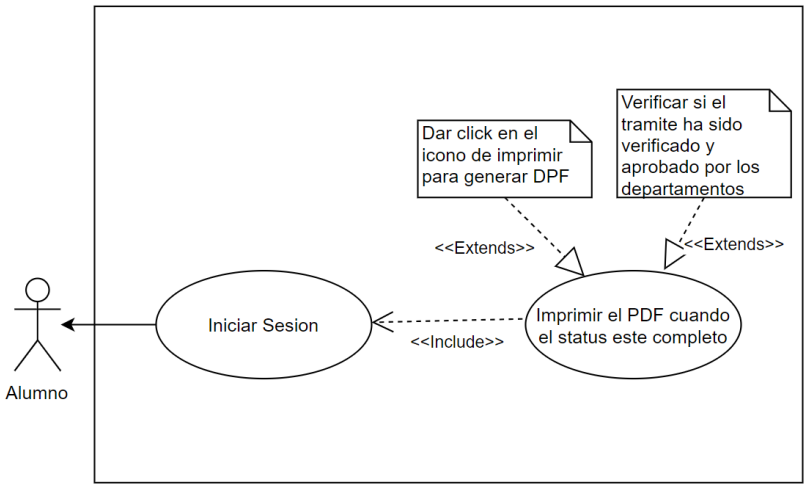

Figura 46: Caso de uso de Imprimir el PDF

ESCENARIO REQUISITO 3 (IMPRIMIR PDF)

Tabla 5: Escenario Requisito 3 (Imprimir PDF)

Número de requisito:	3
Nombre del requisito:	Imprimir PDF
Prioridad:	Alta.
Descripción:	El Alumno al ingresar al sistema se presenta la pantalla de datos junto con el Historial de solicitudes y el status de los tramites en su validación, cuando el trámite ha sido validado aparece un icono de impresión de PDF y lo genera en una ventana externa.
Actores:	Alumno.
Precondiciones:	• El alumno ingresa al sistema con su matrícula y contraseña al sistema. • El sistema debe de estar en línea para el acceso con un link.

	• Todos los departamentos deben de validar que no hay adeudos.
Flujo normal:	1. En el proceso de inicio, el estudiante se encuentra con una pantalla que solicita su **matrícula** y **contraseña**. Una vez que introduce estos datos y presiona el botón "Ingresar", accede al sistema. 2. En caso de que este todo correcto se ingresará al menú principal en donde se visualizan los datos del alumno todos los posibles por hacer o los que están en proceso con su status. 3. El alumno se dirige al historial de solicitudes donde aparecen todas las que ha realizado si este está completo aparece un icono de impresión de un PDF, de caso contrario proseguir al paso 3.1. 4. El PDF se abre en una pestaña externa del navegador que se esté usando en ese momento, donde este ya se genera con los datos de él, y todos los códigos QR donde certifica esta validación.
Flujo alternativo:	4.1. El alumno debe de esperar que todos los departamentos validen que no existe ningún adeudo, si no hay respuesta pasar al paso 4.1.1. 4.1.1. El Alumno debe de acudir a los diferentes departamentos si hay una situación externa la adeudo.
Postcondiciones:	PDF generado con todos los códigos QR para presentar en control escolar.

PROPUESTAS DE DISEÑO DE IMPRIMIR PDF (FLUJO NORMAL)

Pantalla de Inicio de Sesión

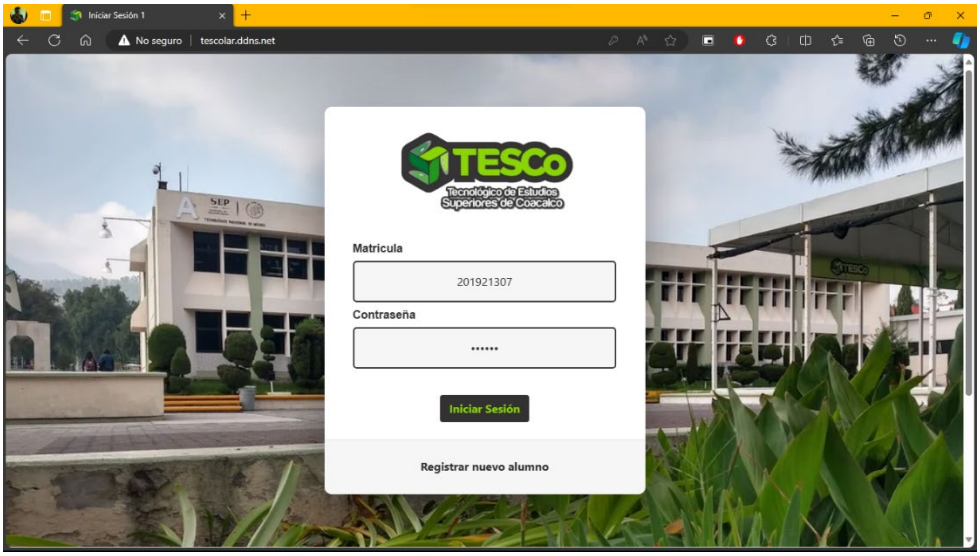

Figura 47: Pantalla de Inicio de Sesión del Alumno

Acceso a la pantalla de información del Alumno, selección y status de tramites.

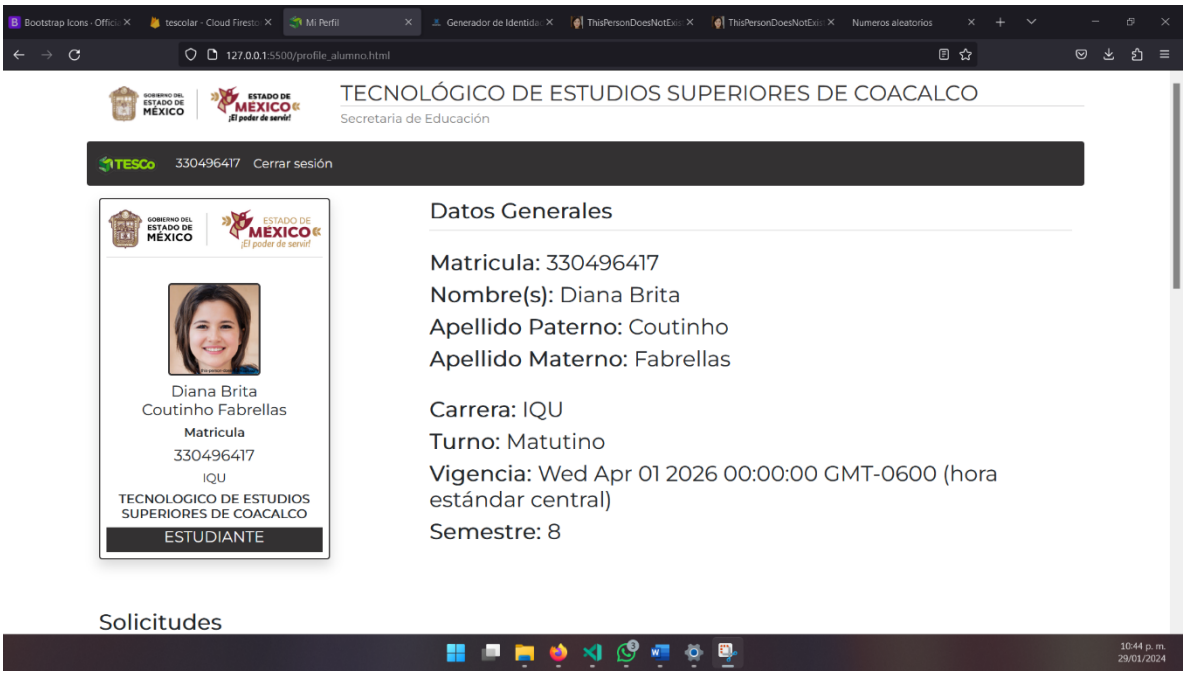

Figura 48: Pantalla de información del Alumno, selección y status de tramites.

Pantalla de status de los tramites.

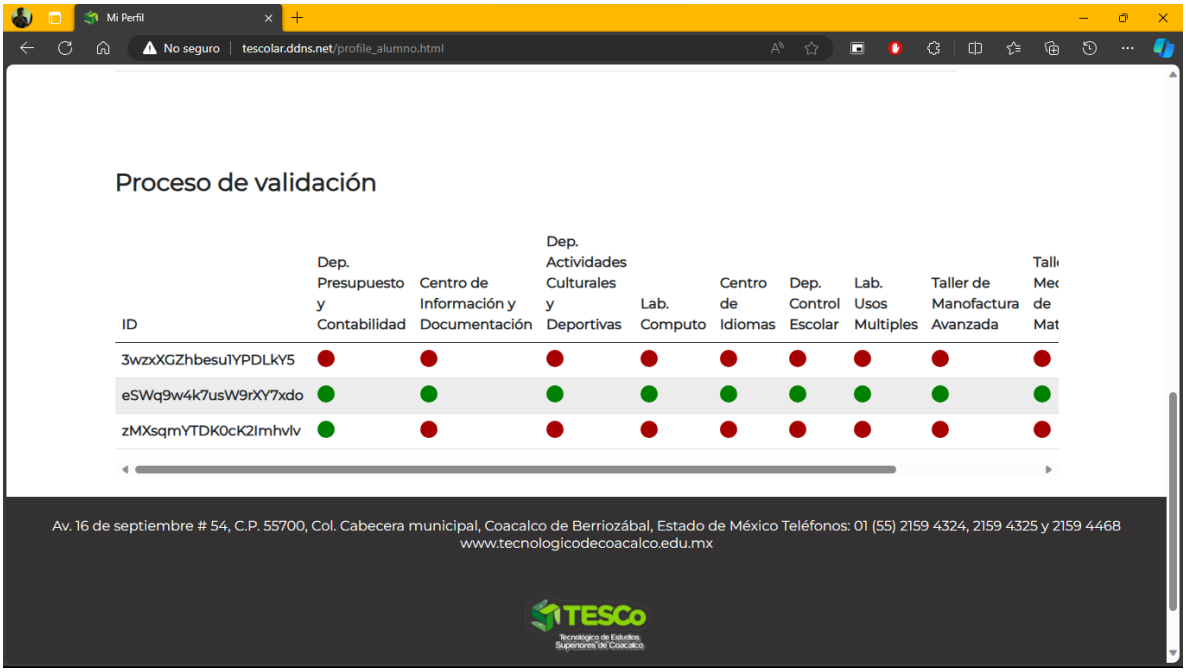

Figura 49: Pantalla de status de los tramites.

Pantalla del historial de tramites del alumno con el icono de impresión.

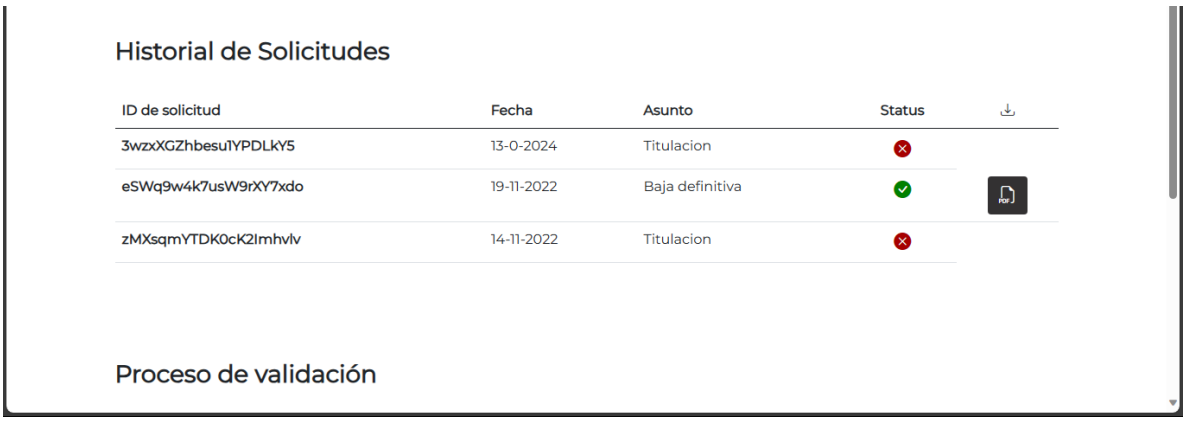

Figura 50: Pantalla del historial de tramites del alumno.

Pantalla con el PDF generado en una nueva pestaña del navegador con los códigos QR ya generados y datos correctos.

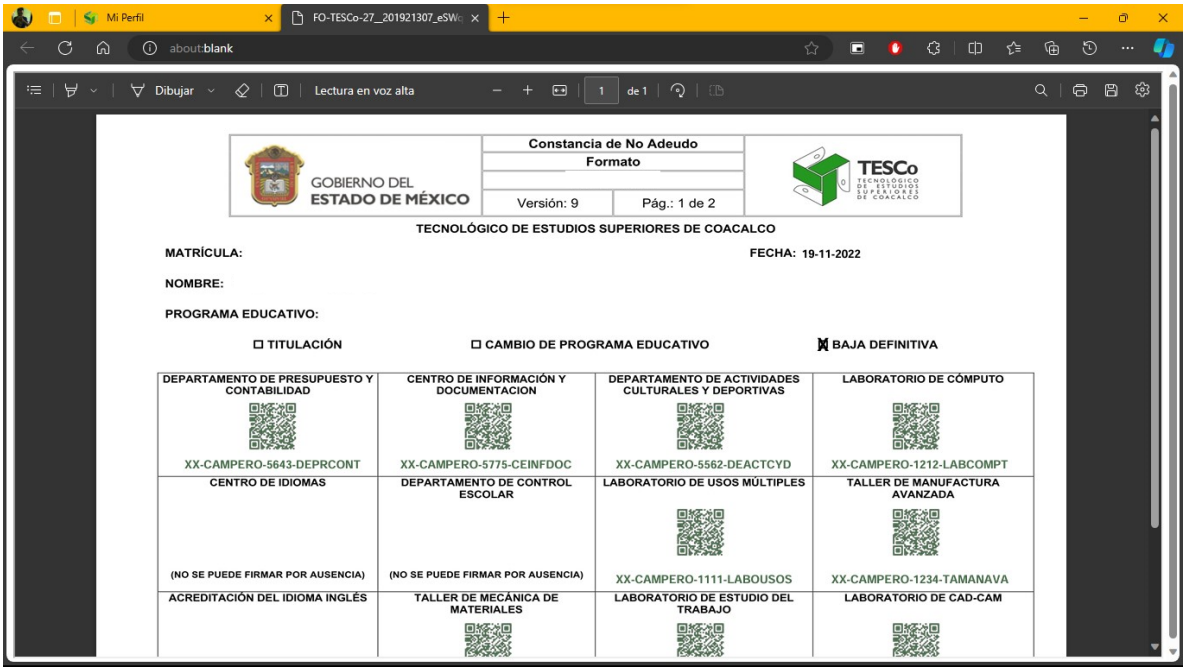

Figura 51: Pantalla del PDF con todas las validaciones hechas (Datos censurados).

DIAGRAMA DE SECUENCIAS DE IMPRIMIR EL PDF

El diagrama de secuencia del proceso de selección de tramite demuestra que el alumno ingresa al sistema con su matrícula y su contraseña, al ingresar puede visualizar el status e impresión del PDF.

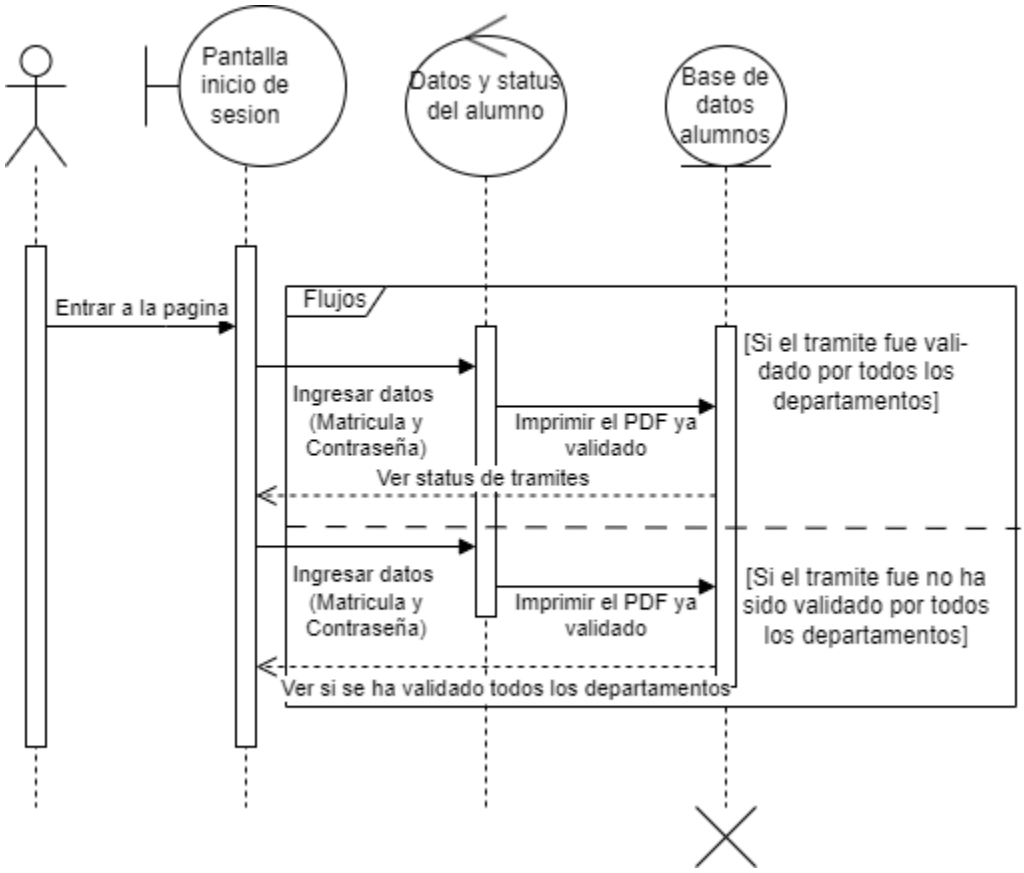

Figura 52: Diagrama de despliegue de imprimir un PDF

3.7 PRUEBA DE FUNCIONAMIENTO DE LA PÁGINA WEB EN EL CLÚSTER

Para que el clúster funcione como almacenamiento de la página web, el nodo o los nodos que funcionan como “Apache” nombrados de ese modo para ejecutar la página, deben de estar con los archivos necesarios para ejecutarla, imágenes, índex, entre otros archivos.

Para hacer esta prueba se requiere la dirección ip de las dos máquinas virtuales ya que en futuro estas se necesitan para el balanceador.

En cualquier navegador dentro de una red LAN (ya sea conectado directamente al servidor o en un switch) se ingresa la ip como una dirección de una página web, ya que funciona como una url o link de acceso, la máquina virtual recibe la solicitud del navegador, el cual tiene el servicio de Apache o HTTPD en este caso, si no existe nada no deberá de aparecer nada o simplemente un mensaje de no conexión.

Dentro de la programación del archivo índex se encuentra la conexión a Firebase

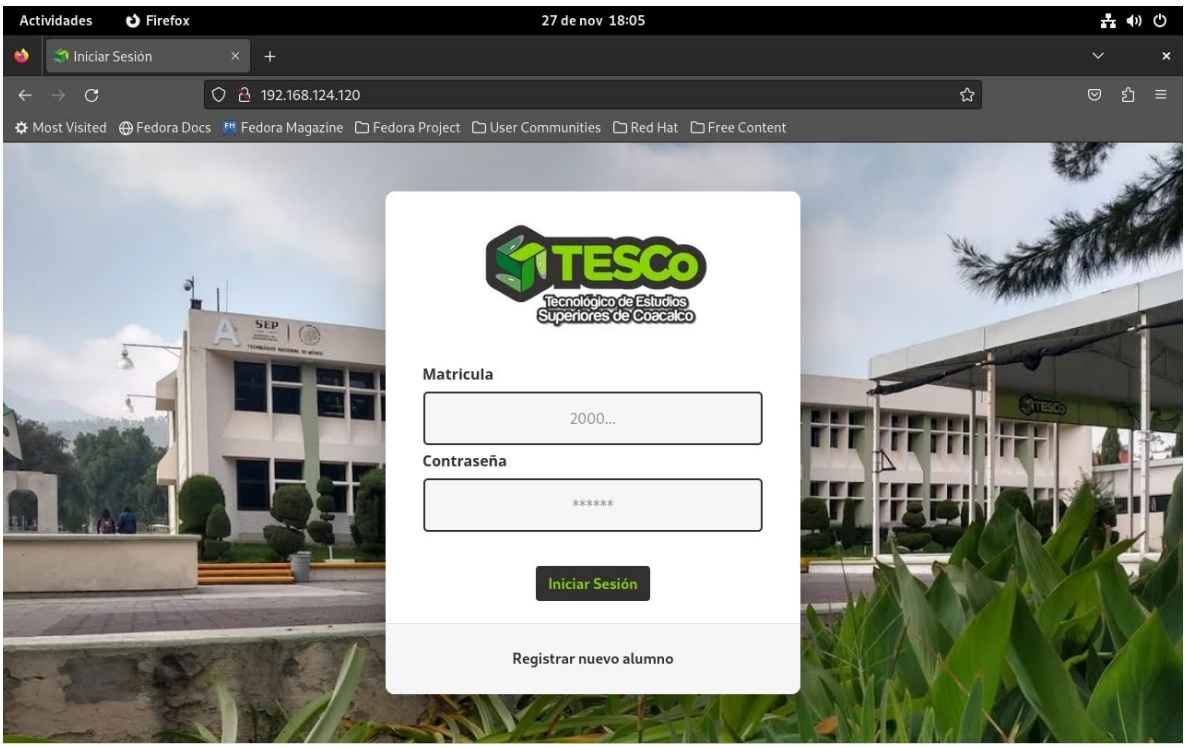

Figura 53: Prueba del funcionamiento del servicio httpd de las máquinas virtuales (Archivo índex)

Autenticación

Los servicios de Google: Auth y Firestore se utiliza en la aplicación web Tescolar para autenticar el inicio de sesión y recuperar las credenciales de acceso existentes en Firestore mediante una relación de email, password y matricula.

La relación está construida en JavaScript, se hace la petición a Firestore en la colección “Alumnos” para recuperar la matricula, email y el ID, posteriormente se llama a Auth para que realice la validación del ID mediante el email y password, si todo es correcto el usuario se autentica y se concede el acceso al panel de alumno, de lo contrario se deniega el acceso.

Si se deniega el acceso existen dos posibles situaciones:

1- Error de inicio de sesión: Se debe a un ingreso erróneo de matrícula y/o password.
2- Error de usuario inexistente: Se debe a que el alumno no está registrado en la plataforma.

```
var matri
var pass

const alertUser = document.getElementById('liveAlert')
const msj = document.getElementById("alertMsj")
```

Figura 54: Variables de validación.

```
        // Login alumno
        db.collection("alumnos").where("matricula", "==", matri)
                .get().then((querySnapshot) => {
                        querySnapshot.forEach((doc) => {
                                //console.log("Data: "+doc.data().)
                                //p = doc.data().password;
                                k = doc.data().matricula;
                                mail = doc.data().email;
                                uidd = doc.data().id;

                                //alert("Bienvenidx: " + matri);
                                firebase.auth().signInWithEmailAndPassword(mail, pass)
                                        .then((userCredential) => {
                                                //const usuario = userCredential.user;
                                                msj.innerText = "Bienvenido: " + matri
                                                toastBootstrap.show()
                                                sessionStorage.setItem("me3", k);
                                                window.location = "profile_alumno.html";
                                        })
                                        .catch((error) => {
                                                // Manejar errores de inicio de sesión
                                                msj.innerText = "Error de inicio de sesión: "+error;
                                                toastBootstrap.show();
                                        });
                        });
                }).catch(() => {
                        //console.log("El usuario no existe: ", error);
                        msj.innerText = "El usuario no existe."
                        toastBootstrap.show()
                });

} else if (matri.length == 4) {
```

Figura 55: Código de validación.

Para hacer la conexión a la base de datos se requiere de la clave de autentificación que otorga Firebase desde la configuración del proyecto que se ha creado ahí previamente, este se usa para obtener acceso a todas las Apis que proporciona Firebase.

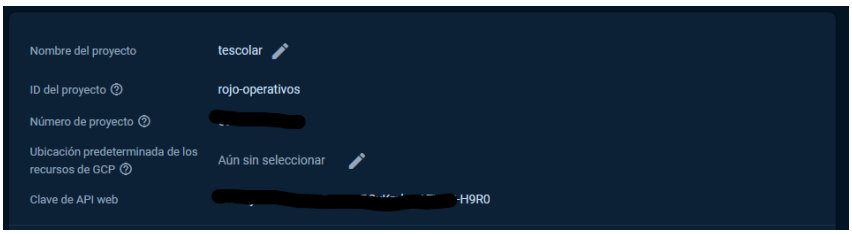

Figura 56: Configuración del Proyecto desde Firebase.

Con esta clave se debe de insertar se debe de usar el lenguaje JavaScript para hacer la conexión, poniendo esa apiKey, el nombre del dominio y el nombre del proyecto. Las cuales unas variables validan si lo que se insertó son correctas, de lo contrario no hay ninguna conexión.

```
// Conexion a Firebase

firebase.initializeApp({
        apiKey: "                                    -H9R0",
        authDomain: "rojo-operativos.firebaseapp.com",
        projectId: "rojo-operativos",
});
const fs = firebase.firestore();
var db = firebase.firestore();
const auth = firebase.auth();
```

Figura 57: Conexión a Firebase

El HAProxy es un elemento o más bien un servicio el cual ayuda a que si una maquina o servidor (en este caso virtual) por algún motivo este caído o no se pueda acceder a él en cualquier momento, haciendo la función de un “switch virtual” dirigiendo las solicitudes a cualquier servidor funcional, además de que sirve para distribuir la carga de solicitudes dentro del mismo clúster. Para este modelo se cuenta con dos servidores Apache o HTTPD (previamente configurados).

El balanceador por practicidad se instala en la maquina principal o en el nodo madre ya que permite una mejor manipulación de este servicio, para instalar el servicio se debe de acceder en usuario root y ejecutar el comando “dnf install haproxy”.

 # dnf install haproxy

Este descargará desde su repositorio los archivos necesarios para que cualquier nodo en este caso sea usado como cualquier punto de acceso a la pagina

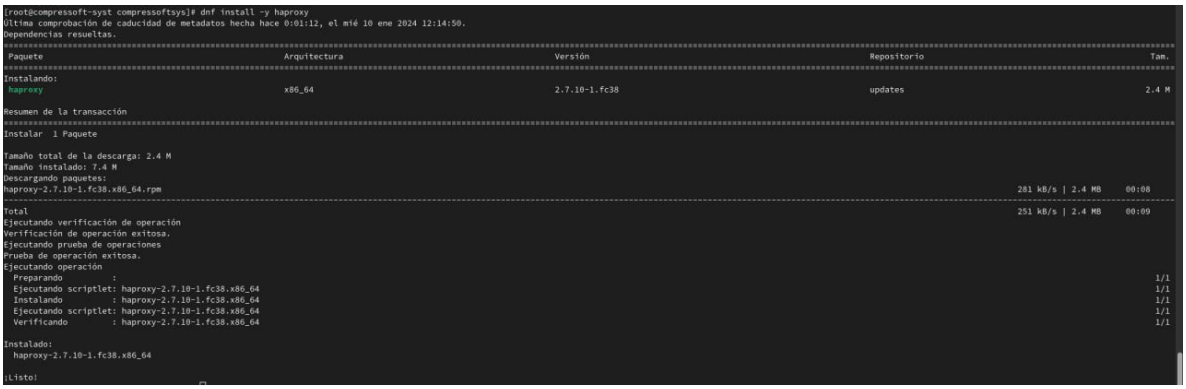

Figura 58: Instalación Haproxy.

El proceso hace una selección tipo Roundrobin, donde si el primer nodo esta con mucha carga o este caído, pasa al siguiente, de lo contrario selecciona ese nodo como apuntador para recibir la solicitud de la página, cada puerto de acceso es el puerto 80, el proceso se debe de hacer con permisos de administrador (root), recalcar que este proceso se tendrá que hacer en la máquina que actuara como el balanceador y las que fungirán como los servidores web, en cada una de ellas sin excepción.

 # firewall-cmd -–add-port=80/tcp -permanent

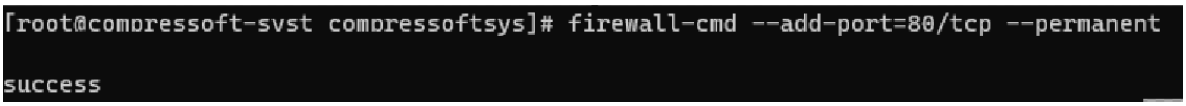

Figura 59: Apertura del puerto 80.

Para verificar que el puerto 80 se haya abierto correctamente, se accede a la lista de puertos disponibles en el firewall del sistema, aunque primero para que los cambios sean aplicados se tiene que reiniciar el proceso del firewall, este ayuda que se guarden los datos correctamente y no exista un fallo, usando el comando del firewall y la orden reload, y para ver los puertos abierto se usa el comando del firewall y la orden list-ports

 # firewall-cmd –reload

#firewall-cmd –list-ports

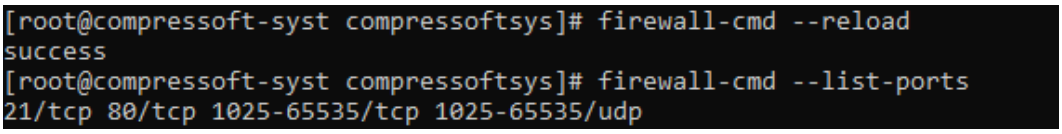

Figura 60: Reinicio y comprobación de los puertos activos.

Después de instalado y abiertos los puertos que se requieren y que este se ejecute junto con el encendido de la máquina virtual, se debe de acceder a la configuración del haproxy, el cual se encuentra en la ruta /etc/haproxy/haproxy.cfg/ ejecutando el siguiente comando:

 # cd /etc/haproxy/haproxy.cfg/

Cabe mencionar que siempre que se vaya a hacer algún cambio en cualquier configuración, no solo de este servicio si no de cualquier otro es recomendable hacer una copia del archivo original ya que puede servir para revertir cambios no deseados o dejar como inicialmente el servidor.

Al abrir el archivo (puede usarse un editor de texto o GNU Nano[7]), aparece la configuración por defecto que contiene.

[7] GNU Nano es un editor de texto para líneas de comandos desarrollado para sistemas UNIX. (Red Hat, Inc. and others, 2017)

Para iniciar la configuración en el apartado global, se borra la línea “log 127.0.0.1 local2”, la línea “pidfile /var/run/haproxy.pid/”, también la línea “maxconn 4000”, ya que esas funciones no se requieren para el funcionamiento de este cluster. Después en la zona #turn on stats unix sockets, se agrega una sentencia llamada “stats timeout 30s”, esta sentencia actualiza las estadísticas del socket cada 30 segundos, donde se puede ver cada evento en la ruta que esta indica.

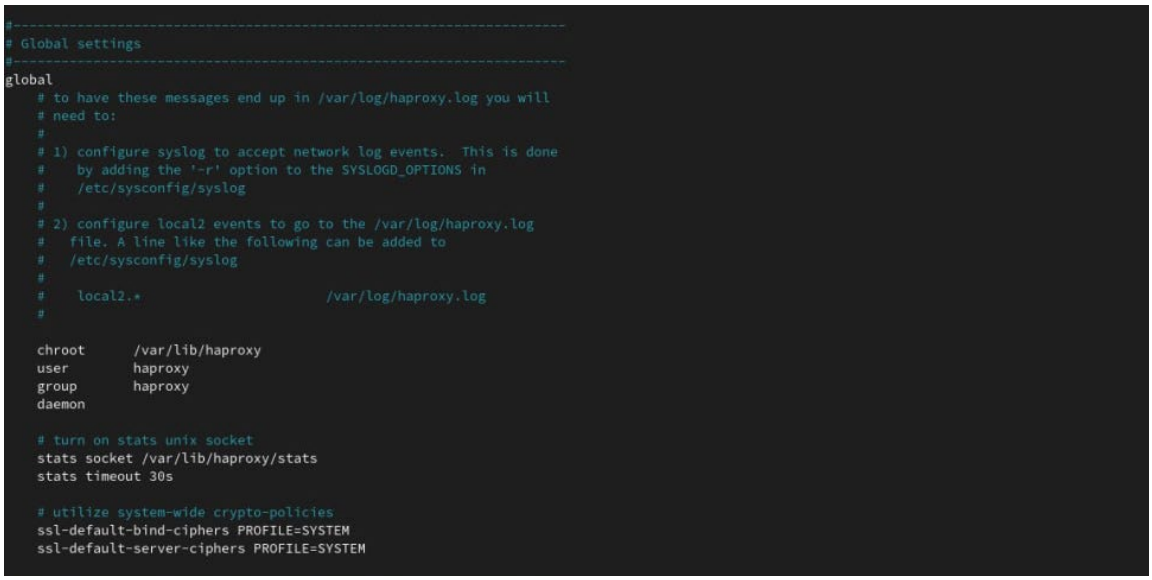

Figura 61: Primera configuración Haproxy.cfg

Siguiente de esto en la sección “defaults” se conservan las líneas “mode”, “timeout connect”, “timeout client” y “timeout server”, las líneas de log y option se pueden conservar, pero estas tienen que ser comentadas con un hashtag o signo de gato como sea conocido, el parámetro “mode” se deja tal cual está en http, indicando que se va a usar para un servicio http, y las líneas que restan (las no comentadas) se cambia el parámetro a 30s indicando que cada solicitud que haga la conexión, el cliente o el servidor tendrá un tiempo máximo de 30 segundos, las líneas restantes se eliminan ya que no son requeridas para el clúster.

Figura 62: Configuración de la zona "defaults" de Haproxy.

A continuación, sigue la sección "frontend" el cual por defecto tiene por nombre "main" (se le cambia el nombre a "tescolar_servers"), esta sección es responsable de recibir las solicitudes de los clientes y tomar decisiones sobre cómo manejar esas solicitudes. Un frontend está asociado con uno o más "backend" y define cómo se debe enrutar el tráfico entrante hacia esos backends. Los parámetros para modificar son los siguientes:

- bind *:80
- mode http
- defautl_backend tescolar_servers

Estas sentencias son las necesarias para el funcionamiento del haproxy, donde "bind" es el puerto de entrada y salida de las solicitudes hechas, "mode" está configurada para el tráfico http y "default_backend" es el nombre de donde dirige el tráfico, las sentencias sobrantes pueden comentarse o eliminarse.

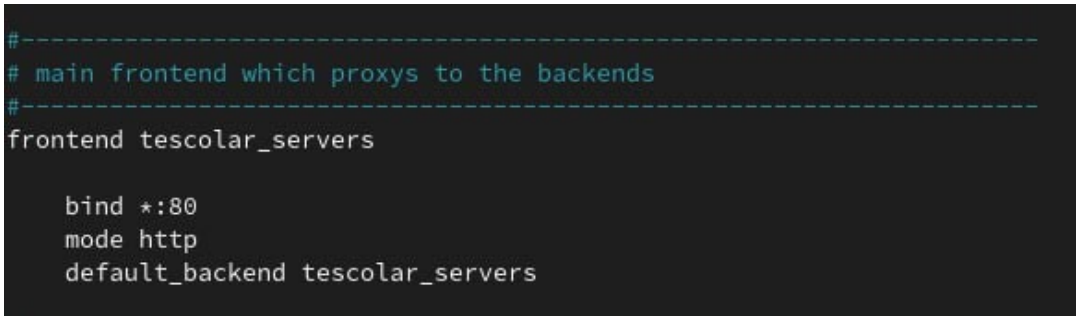

Figura 63: Configuración de la sección "frontend" del Haproxy

Para finalizar se configura la zona para el backend, en esta situación solo se ocupa una zona, el cual se le cambia el nombre a “tescolar_servers” como se indicó en la configuración del frontend. Indicando que en balance va a ser mediante Roundrobin como se explica anteriormente, el modo para un servicio http y la opción “forwardfor”, la cual sirve para cambiar la dirección ip ósea si se hace una solicitud, está siempre va a mostrar la del haproxy y no la de los servidores donde se almacenan los servicios de apache, y para finalizar se agregan los servidores donde se ejecuta el servicio de apache y la pagina ya previamente cargada.

```
#---------------------------------------------------------------------
# static backend for serving up images, stylesheets and such
#---------------------------------------------------------------------
backend tescolar_servers

    balance     roundrobin
    mode        http
    option      forwardfor
    server      apache1 192.168.124.127:80 check
    server      apache2 192.168.124.188:80 check
```

Figura 64: Apuntador a los servidores de apache

El resto de la configuración se puede borrar o comentar ya que no se requiere para este fin que se utiliza el clúster.

Se guardan los cambios y se reinicia el servicio para aplicarlos, con el comando “restart”, al momento de reiniciar se debe de hacer un “status” para comprobar que todo este correcto.

 # systemctl restart haproxy

 # systemctl status haproxy

```
[root@compressoft-server1 haproxy]# systemctl status haproxy
● haproxy.service - HAProxy Load Balancer
     Loaded: loaded (/usr/lib/systemd/system/haproxy.service; enabled; preset: disabled)
    Drop-In: /usr/lib/systemd/system/service.d
             └─10-timeout-abort.conf
     Active: active (running) since Thu 2024-01-25 12:43:39 CST; 23min ago
    Process: 14125 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -f $CFGDIR -c -q $OPTIONS (code=exited, status=0/SUCCESS)
   Main PID: 14127 (haproxy)
     Status: "Ready."
      Tasks: 7 (limit: 18866)
     Memory: 38.5M
        CPU: 689ms
     CGroup: /system.slice/haproxy.service
             ├─14127 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.pid
             └─14129 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.pid

ene 25 12:43:39 compressoft-server1 systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
ene 25 12:43:39 compressoft-server1 haproxy[14127]: [NOTICE]   (14127) : New worker (14129) forked
ene 25 12:43:39 compressoft-server1 haproxy[14127]: [NOTICE]   (14127) : Loading success.
ene 25 12:43:39 compressoft-server1 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
```

Figura 65: Comprobación de que el servicio Haproxy este correcto

Para verificar el funcionamiento, desde un navegador se introduce la ip de la maquina donde se ejecuta el servicio de Haproxy, como referencia se recomienda que en archivo índex se ponga un mensaje para identificar en ese momento, después se puede retirar, pero solo para fines de pruebas, ya en la práctica este debe de ser eliminado.

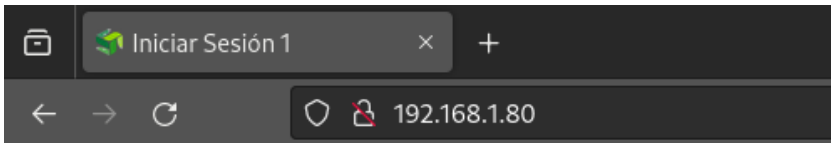

Figura 66: Función Apache 1

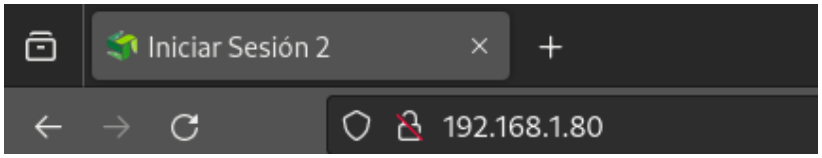

Figura 67: Función Apache 2

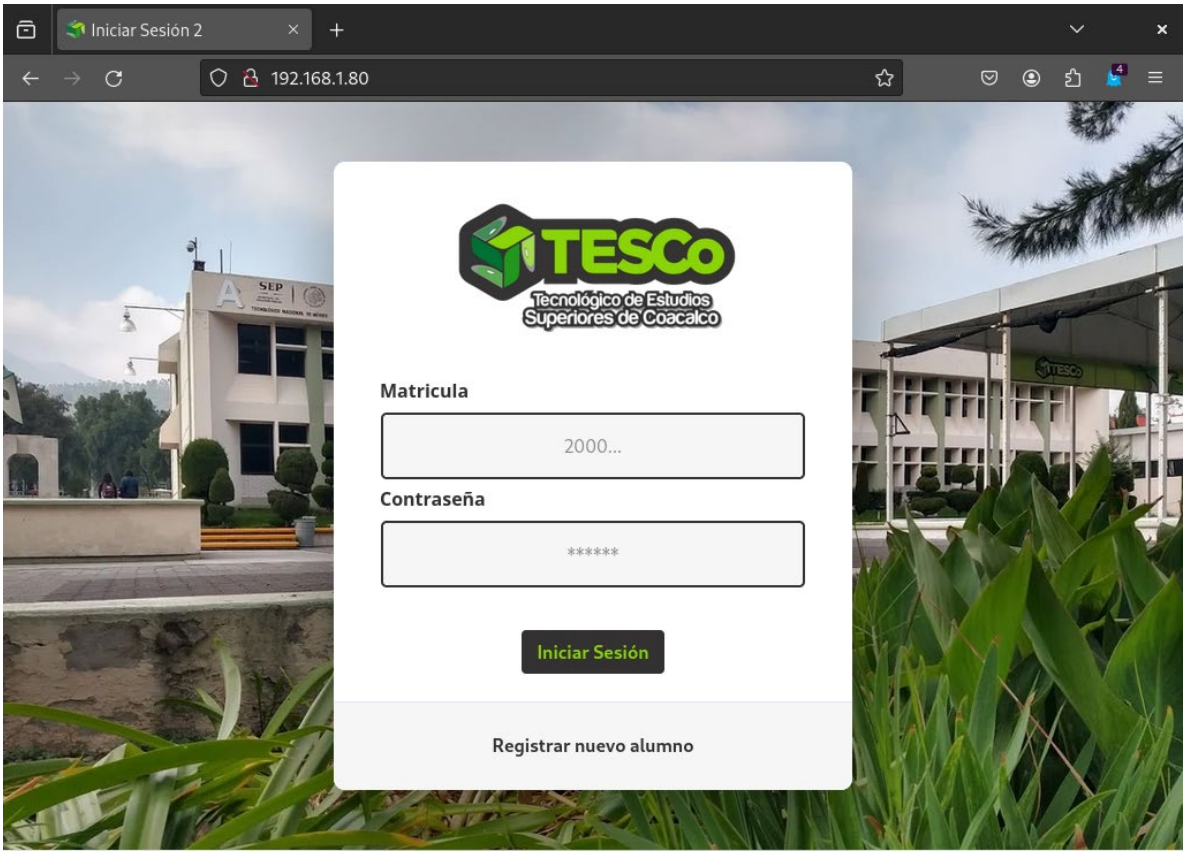

Figura 68: Funcionamiento total del Haproxy

3.8 PLAN DE EJECUCIÓN DE SEGURIDAD EN BASE A LA NORMA ISO 27001

3.8.1 INTRODUCCIÓN

La presente normativa aplicada en el clúster ayuda a la integridad de los datos o el uso para el cual este destinado, también de generar un marco para identificar los diferentes riesgos de seguridad, como fugas de información, amenazas de ataques, factores externos, entre otros, así para dar la posibilidad de mitigar todos estos aspectos.

3.8.2 ALCANCES

- Seguridad básica en el clúster
- Protección de los datos clave de los usuarios
- Procesos de mantenimiento físicos y virtuales al servidor
- Acceso restringido a la base de datos principal

3.8.3 OBJETIVOS

- Proteger el acceso de archivos mediante un administrador.
- Denegar el uso de links para acceder a zonas del servidor específicas.
- Mantener un acceso seguro a las máquinas virtuales.
- Limitar lo más posible el uso del servidor.
- Mejorar la seguridad de SSH según los requisitos del servidor.

3.8.4 ORGANIZACIÓN (LIDERAZGO)

Tabla 6: Organización de Actividades puntuando actividad y encargados

Actividad	Encargado
Proteger el acceso de archivos mediante un administrador.	Edson
Denegar el uso de links para acceder a zonas del servidor específicas.	Edson
Mantener un acceso seguro a las máquinas virtuales.	Edson y Arturo
Limitar lo más posible el uso del servidor.	Arturo
Mejorar la seguridad de SSH según los requisitos del servidor.	Edson y Arturo

3.8.5 PLANIFICACIÓN

La implementación de la seguridad se empieza a aplicar desde el inicio del proyecto, aunque por practicidad se hace al final ya que con las configuraciones aplicadas al clúster es más fácil restaurar algo que en el momento que se aplican los cambios.

Se usa un diagrama de Gantt por semanas para dividir las actividades (Objetivos), junto con la Tabla 7 que enumera las actividades.

Tabla 7: Tabla de actividades indicando duración y fechas.

Actividad	Descripción	Duración (Semanas)	Fecha de inicio (Actividad)
A	Proteger el acceso de archivos mediante un administrador.	1	Dic 18
B	Denegar el uso de links para acceder a zonas del servidor específicas.	1	Ene 1
C	Mantener un acceso seguro a las máquinas virtuales.	1	Ene 8
D	Limitar lo más posible el uso del servidor.	1	Ene 15
E	Mejorar la seguridad de SSH según los requisitos del servidor.	1	Ene 22

Tabla 8: Cronograma de Actividades

Actividad	Duración: 1 Mes					
	Diciembre		Enero			
	3	4	1	2	3	4
A						
B						
C						
D						
E						

3.8.6 OPERACIÓN Y SOPORTE.

3.8.6.1 PROTEGER EL ACCESO DE ARCHIVOS MEDIANTE UN ADMINISTRADOR.

La implementación de la medida de acceso a los archivos se presenta como una acción elemental pero crucial en el contexto del clúster. Constituye el primer filtro de seguridad, mediante el cual se asigna un usuario y contraseña para acceder al sistema operativo. Este proceso adquiere particular relevancia al considerar que pretende prevenir modificaciones maliciosas que puedan incidir adversamente en el rendimiento del clúster. De manera más drástica, dicho acceso restringido también resguarda contra la posibilidad más catastrófica: el cese total del funcionamiento del clúster.

Para el usuario promedio o el cliente que puede ocupar el producto no es consciente de este aspecto ya que por consecuencia tiende a poner un nombre de usuario sencillo y una contraseña fácil de recordad, entre las recomendaciones que pueden tomar son las siguientes:

- Usuario sencillo haciendo alusión a la finalidad.
- Contraseñas largas y/o con frases mezclando caracteres numéricos, letras y símbolos.
- Usar un gestor de contraseñas de confianza.

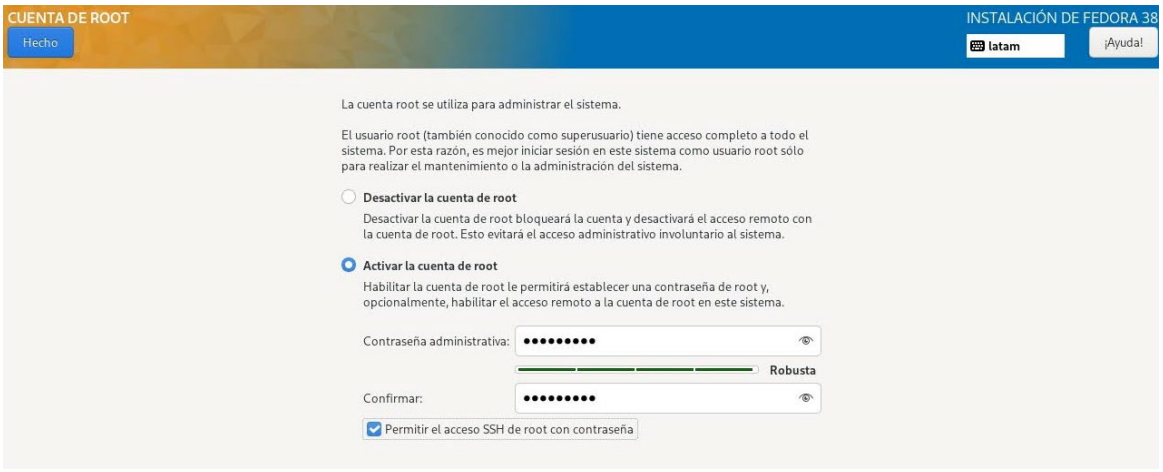

Figura 69: Demostración de una contraseña segura.

Algunas veces el mismo sistema operativo mide la seguridad de la contraseña, así que es pertinente no hacer caso omiso.

Al aplicar estar medidas añade un filtro para acceder a los archivos de manera remota, ya usando el método SSH o la conexión FTP, este no es un método sólido, pero no cualquier actor podrá ingresar a hacer cambios al sistema operativo.

3.8.6.2 DENEGAR EL USO DE LINKS PARA ACCEDER A ZONAS DEL SERVIDOR ESPECÍFICAS.

Con el fin de salvaguardar la integridad de los archivos, al emplear la dirección IP o el enlace DNS para acceder al clúster, se ha observado que al especificar la ruta de los archivos se expone la totalidad del backend de la programación. Este proceder, resulta impropio, dado que conlleva el riesgo de obtener información de suma importancia. En un contexto más ideal, se propone restringir el acceso a esta sección del servidor, especialmente para el usuario promedio, a fin de garantizar la confidencialidad y seguridad de los datos almacenados.

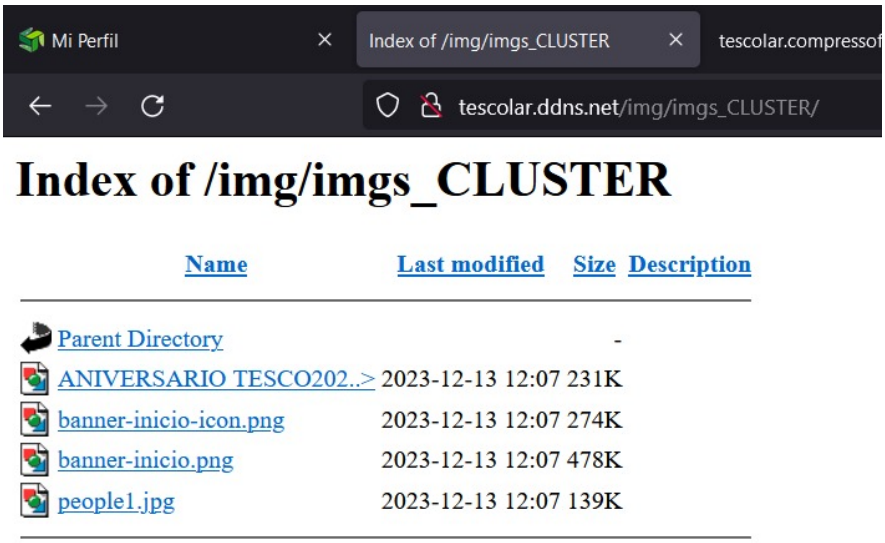

Figura 70: Demostración de acceso a los archivos del servidor mediante un link.

Con el propósito de resolver la situación actual desde la máquina que aloja los servicios de HTTPD (Apache), es necesario realizar una modificación en la configuración principal. Este ajuste implica acceder a la ruta donde se encuentra almacenado el archivo httpd.conf, que constituye el repositorio de las configuraciones esenciales de Apache.

```
# cd /etc/httpd/conf/
# nano httpd.conf
```

Este está dividido en 3 secciones, la de interés es la de directivas de funcionamiento, el cual se agrega una regla, la cual es:

Option -Indexes _FollowSymLinks

Su función es que este solo de acceso a los archivos que contengan un condigo de programación como HTML entre otras, estas solo pueden cargarse en el navegador, y que solo de permiso a cargar los archivos si están declarados en el navegador, de lo contrario deniega el acceso al servidor.

```
# Further relax access to the default document root:
<Directory "/var/www/html">
    #
    # Possible values for the Options directive are "None", "All",
    # or any combination of:
    #   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
    #
    # Note that "MultiViews" must be named *explicitly* --- "Options All"
    # doesn't give it to you.
    #
    # The Options directive is both complicated and important.  Please see
    # http://httpd.apache.org/docs/2.4/mod/core.html#options
    # for more information.
    #
    Options -Indexes -FollowSymLinks

    #
    # AllowOverride controls what directives may be placed in .htaccess files.
    # It can be "All", "None", or any combination of the keywords:
    #   Options FileInfo AuthConfig Limit
    #
    AllowOverride none
```

Figura 71: Modificación del fichero httpd.conf

La implementación de esta medida contribuye significativamente a fortalecer la integridad del código de programación, así como a asegurar los accesos a imágenes y/o archivos adicionales necesarios para el sistema. Al intentar acceder a una ruta del servidor a través de un enlace, el sistema rechazará automáticamente la solicitud, reforzando de este modo la seguridad y la protección de los recursos del servidor.

Figura 72: Acceso denegado a los archivos.

MANTENER UN ACCESO SEGURO A LAS MÁQUINAS VIRTUALES.

El resguardo de las máquinas virtuales es de suma importancia, aquí entra en juego el proceso de autentificación del usuario como se explica en la primera medida de seguridad, unas medidas extras que se pueden implementar son las siguientes:

- Exigir una contraseña para ingresar al gestor de máquinas virtuales.
- Tener respaldos de las máquinas virtuales por cualquier siniestro.
- Auditorias constantes.
- Capacitación a los usuarios que puedan manipular el clúster.

LIMITAR LO MÁS POSIBLE EL USO DEL SERVIDOR.

Muchos usuarios y clientes no toman en cuenta la integridad de la maquina física, ya que no tienen preocupación hasta que esta empieza a presentar signos de falta de mantenimiento, este síntoma puede llevar a lo siguiente:

Perdida de rendimiento repentino.

Temperaturas extremas del chip gráfico y del chip de procesamiento (GPU y CPU respectivamente).

- Apagones repentinos de la maquina física.
- Posibles cortos circuitos de los componentes del servidor.
- Para prevenir cualquier posible contingencia es recomendable hacer las siguientes acciones.
- Monitoreo de temperaturas de los chips gráficos y el uso de recursos.
- Mantenimiento periódico al servidor haciendo limpieza de polvo e incluso cambio de pastas térmicas.
- Conectar el servidor a un no break (dispositivo con capacidad de suministrar electricidad por periodos cortos en caso de que no haya energía eléctrica).
- Montar el servidor en una torre especializada para estos fines.
- Hacer auditorias periódicas.

Para el monitoreo, Fedora cuenta con una interfaz gráfica en línea mediante el puerto 9090, el cual muestra el consumo de recursos y temperaturas de ciertos componentes, este se

puede acceder con el link del servidor o la ip que obtiene al conectase a una red, para este caso se usa el DNS que es tescolar.compressoft.com:9090. El visor de métricas pedirá el usuario y la contraseña que se usa en el sistema operativo.

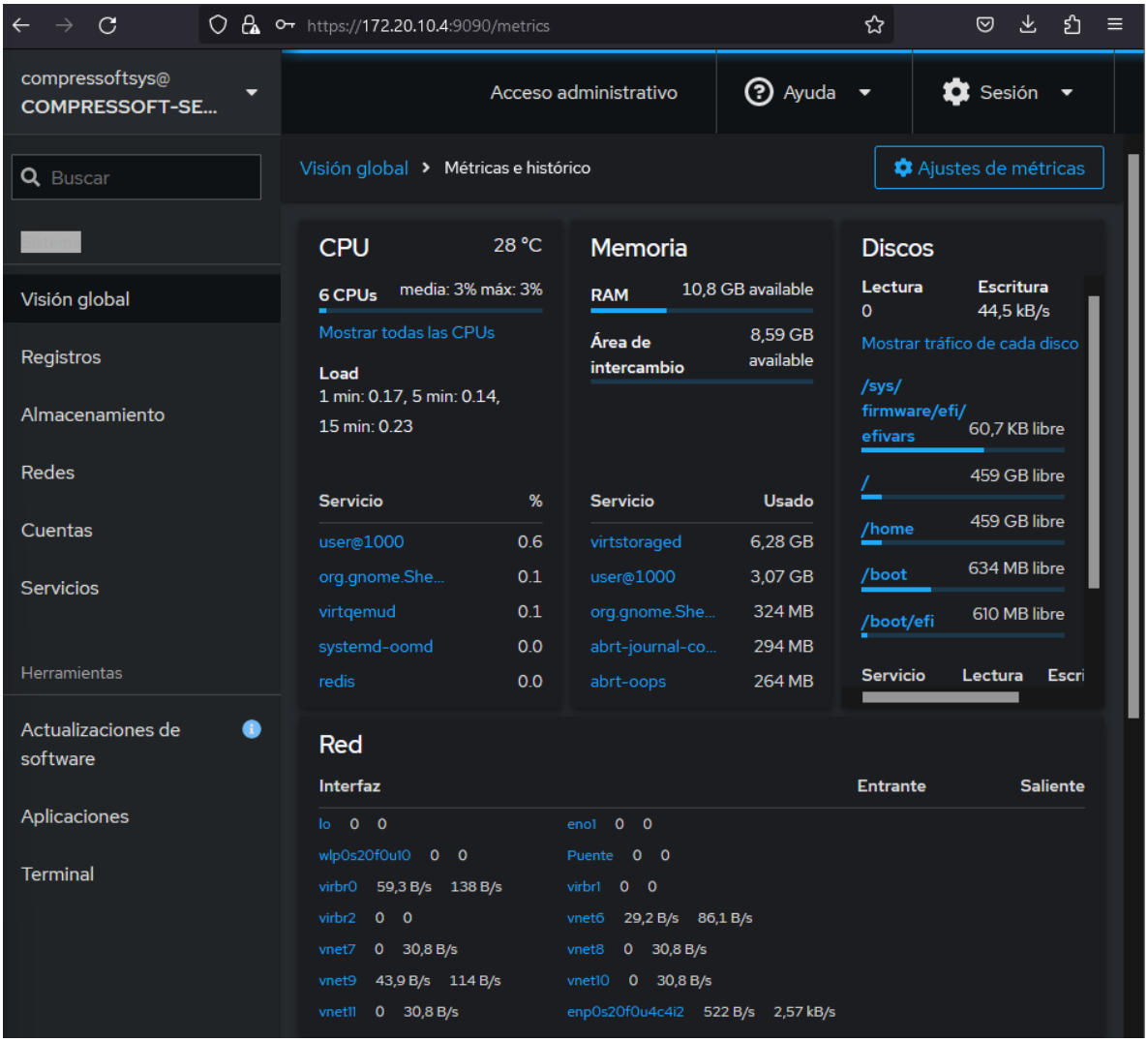

Figura 73: Interfaz de métricas del servidor físico.

Adicionalmente a la visualización de métricas de temperaturas y uso de recursos, se incorpora la capacidad de monitorear el ancho de banda. Para llevar a cabo esta tarea, se emplea el programa Wireshark, diseñado específicamente para medir los protocolos de redes. Este programa proporciona un desglose detallado de todos los puertos e interfaces de red con los que el servidor está equipado. De esta manera, permite no solo verificar su funcionamiento sino también identificar posibles errores. En caso de detectar algún

problema, Wireshark ofrece información valiosa que puede orientar hacia soluciones efectivas para optimizar la eficiencia y estabilidad del sistema.

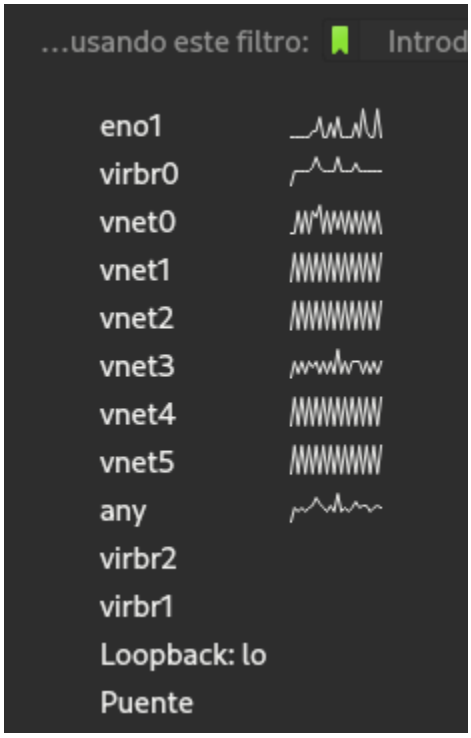

Figura 74: Métricas de medición de Wireshark

3.8.6.3 MEJORAR LA SEGURIDAD DE SSH SEGÚN LOS REQUISITOS DEL SERVIDOR.

Este método también depende del primero, ya que el SSH se debe de usar el mismo usuario y contraseña que se les asignaron a las máquinas virtuales y al servidor físico, por ende, toda esta protección tiene una dependencia, aunque la forma fácil de evitar un SSH se haga con éxito de parte de un actor no deseado, es no revelando la dirección ip o estar en constante renovación las contraseñas de los usuarios del servidor.

Para activar la conexión segura con SSH se debe de hacer desde la instalación del sistema operativo donde pregunta si desea activar el SSH con el inicio de sesión de usuario, este método funciona solo con el usuario root ya que el SSH ya viene activado por defecto.

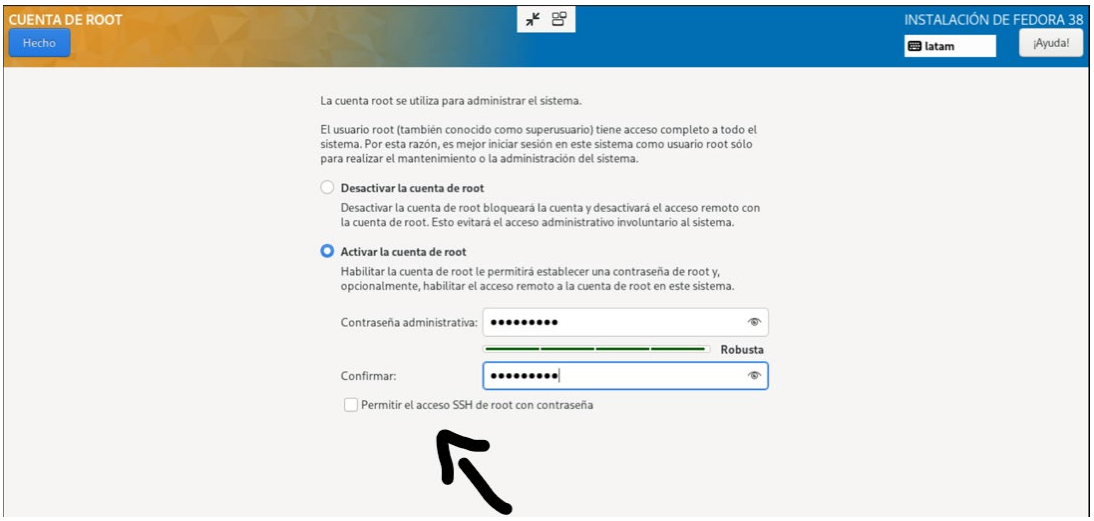

Figura 75: Activación de SSH para el usuario root.

Una alternativa adicional que se puede incorporar es la utilización de la transferencia de archivos a través del protocolo FTP. Este método presenta la ventaja de prescindir del uso directo de la máquina física, haciendo posible la gestión mediante programas como FileZilla. En este escenario, el servidor desempeña el papel de receptor, permitiendo al cliente visualizar y manipular los archivos de manera segura. Este enfoque no solo garantiza la seguridad en las operaciones de transferencia, sino que también elimina la necesidad de recurrir a dispositivos externos, como memorias o pendrives, simplificando así el proceso de intercambio de archivos.

```
PS C:\Users\Arturo> ssh compressoftsys@tescolar.ddns.net
compressoftsys@tescolar.ddns.net's password:
Web console: https://compressoft-server1:9090/

Last login: Thu Jan 25 20:30:55 2024 from 201.121.125.107
[compressoftsys@compressoft-server1 ~]$ su
Contraseña:
[root@compressoft-server1 compressoftsys]#
```

Figura 76: SSH desde un equipo de un tercero

Para ello se debe de instalar primero el programa mediante la tienda de aplicaciones o con terminal. El comando para la instalación es “vsftpd” el cual hace que el servidor se convierta en uno FTP y este pueda enviar y recibir los archivos.

 # dnf install vsftpd

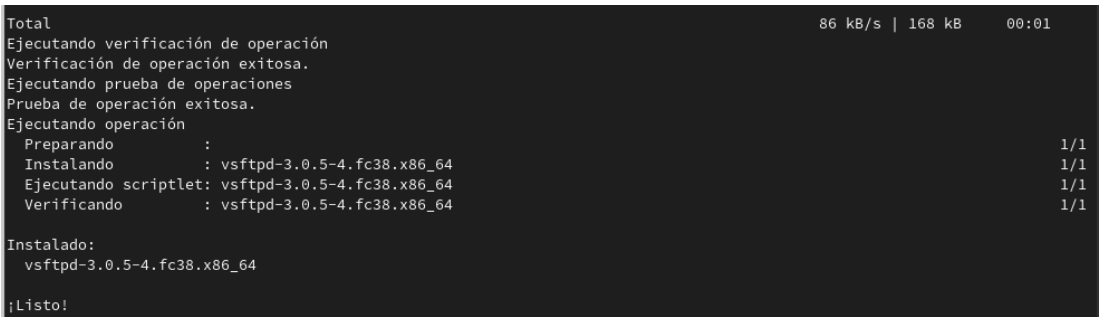

Figura 77: Instalación del servicio VSFTPD

A continuación, es necesario iniciar el proceso y dar permisos en el corta fuego (Firewall) el cual se encargará de dar lo necesario y apertura los puertos que ocupa.

 # systemctl start vsftpd

 # firewall-cmd –add-service=ftp

 # firewall-cmd –permanent –add-service=ftp

 # firewall-cmd –reload

Estos comandos añaden el servicio FTP a las reglas del cortafuegos, asegurando la apertura de los puertos necesarios. Es importante destacar que todos estos comandos deben ejecutarse con permisos de root, dado que implican cambios cruciales en el sistema.

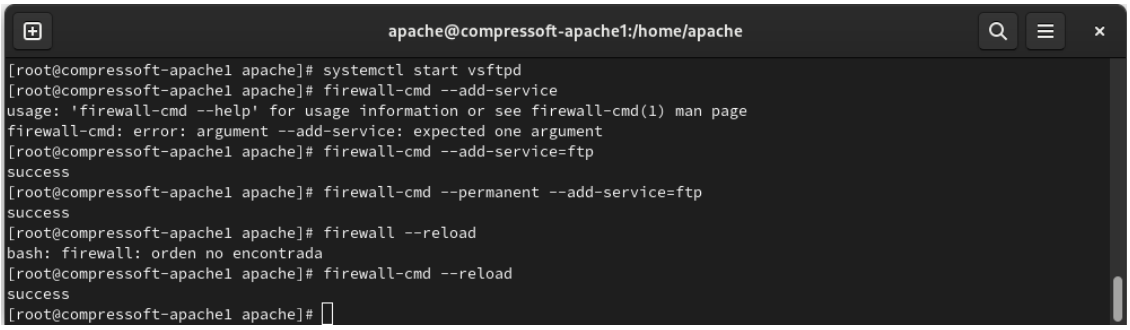

Figura 78: Aplicación de reglas del Firewall al VSFTPD

Nombre del Servicio	LAN IP	Protocolo	Puerto LAN	Puerto público	
tescolar	192.168.1.80	TCP/UDP	80	80	SI
compressoft-server	192.168.1.80	TCP/UDP	9090	9090	SI
compressoft-ssh	192.168.1.80	TCP	22	22	SI

Figura 79: Puertos en uso para las diferentes conexiones y protocolos existentes.

3.8.7 EVALUACIÓN

La evaluación correspondiente debe ser realizada por un auditor con experiencia y especialización en la norma pertinente. Es fundamental tener en cuenta que el desarrollo en cuestión está a cargo de ingenieros en Sistemas Computacionales especializados en Ingeniería de Software. En este contexto, no se espera que lleven a cabo una evaluación exhaustiva, ya que esta tarea recae en manos de un auditor experto en la normativa aplicable.

3.9 CONFIGURACIÓN DEL DNS EXTERNO

Para que el sistema pueda ser accedido mediante un nombre de dominio se usa el servicio de hosting de “Noip” y “CloudFlare”, estos servicios externos ayudan al ahorro del proceso de crea un nodo para el DNS y el servicio del DHCP (En caso de ser LAN).

NoIP ofrece un nombre de dominio semi personalizado ya que puede ser cualquier nombre, pero el resto del dominio contiene su propio domino cual es ddns.net, haciendo uso de la dirección pública del proveedor de internet, para saber este dato, en cualquier navegador se busca “Cual es mi ip publica” y fácilmente se sabrá. Como dato extra las ips publicas pueden cambiar con el tiempo.

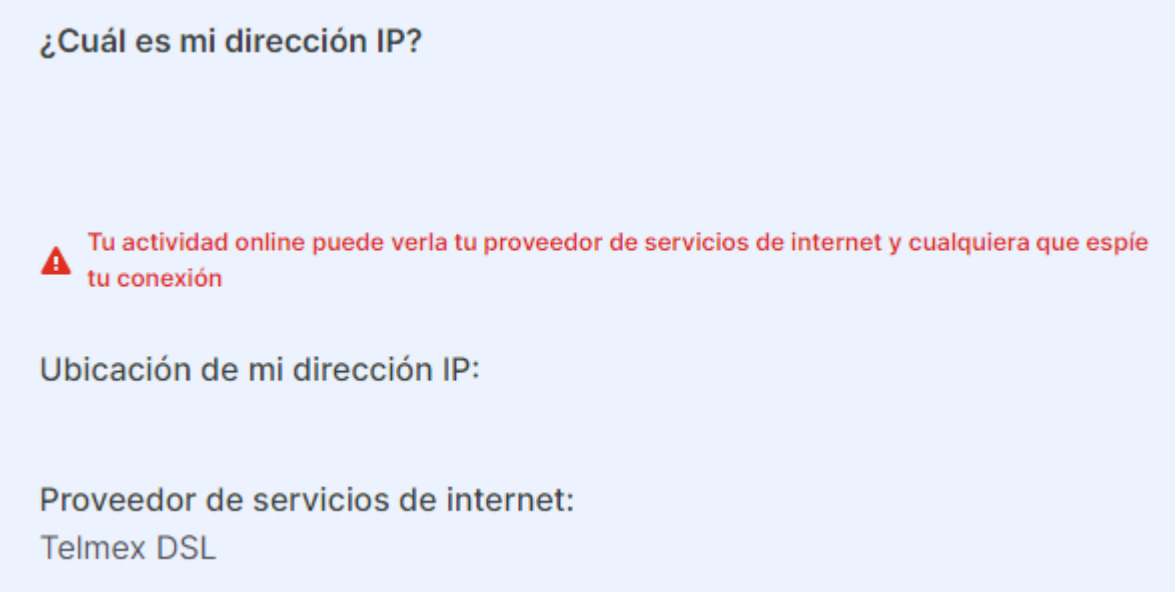

Figura 80: Demostración de una IP Publica (censurada).

Siguiente a esto en Noip se crea un nuevo nombre de host insertando el nombre del dominio y la dirección publica, el mismo servicio apuntara al modem del proveedor para ver un cambio en esta ip pública.

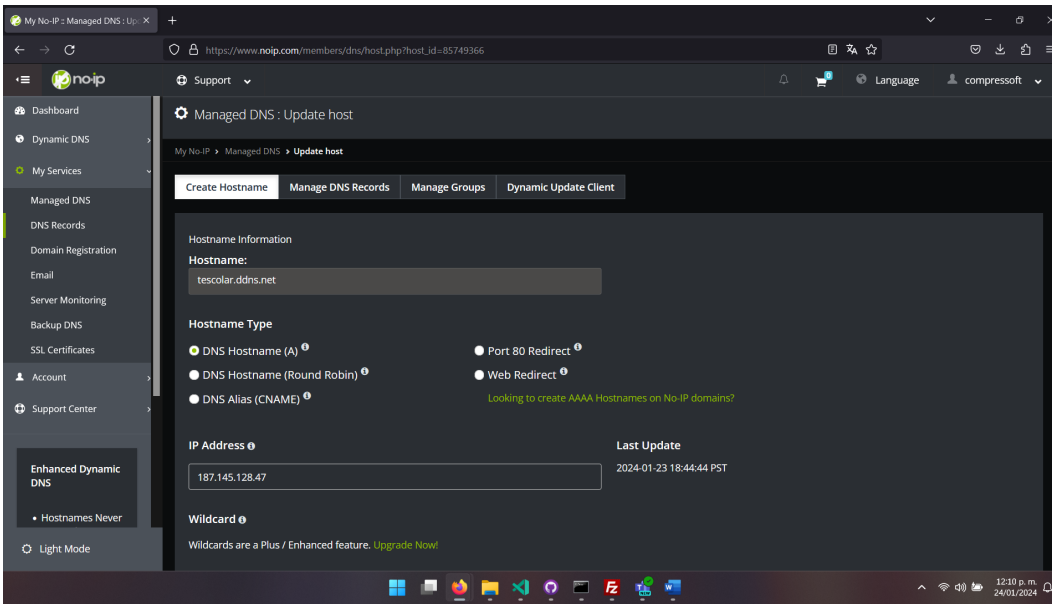

Figura 81: Configuración de Noip

Haciendo esta acción el DNS se genera de forma automática y de ser una red LAN se convierte en una red de acceso público únicamente usando el link que ha generado el servicio

Figura 82: Asignación Link del DNS.

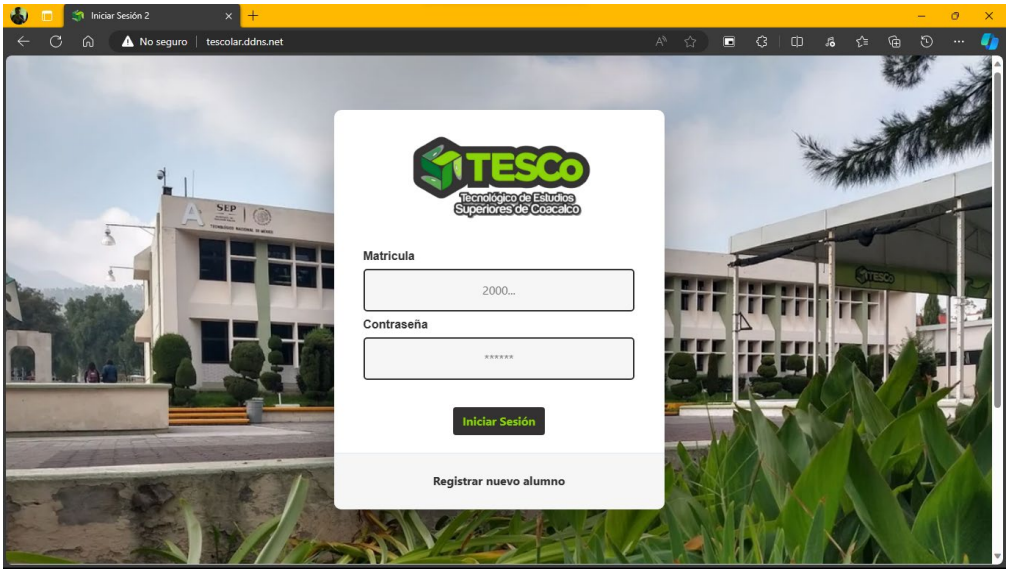

Figura 83: Funcionamiento del Link asignado por el DNS

Para asignar un alias de servidor o alias de DNS(CNAME) aquí entra en juego CloudFlare, también con este servicio hace que la pagina cumpla con el protocolo HTTPS que es el protocolo de seguro de transferencia de datos, dando seguridad al usuario que acceda al sistema, además de que añade una protección contra ataques DDoS (Ataques a servidores para sobrecargar sus solicitudes).

Primeramente, se tienen que dar de alta los dominios que se desean ocupar en este caso son los siguientes:

- compressoft.com: Nombre de identidad propia.
- delivery: Nombre de identidad propia.
- www: haciendo referencia a la World Wide Web utilizando el protocolo de conexión.
- tescolar: Alias de dominio el cual se va a usar para entrar al sistema sustituyendo a tescolar.ddns.net

Tipo	Nombre	Contenido	Estado de proxy	TTL	Acciones
A	compressoft.com		Redirigido por proxy	Automático	Editar
A	delivery		Redirigido por proxy	Automático	Editar
A	www		Redirigido por proxy	Automático	Editar
A	www		Redirigido por proxy	Automático	Editar
CNAME	tescolar	tescolar.ddns.net	Redirigido por proxy	Automático	Editar

Figura 84; Configuración de CNAME en CloudFlare

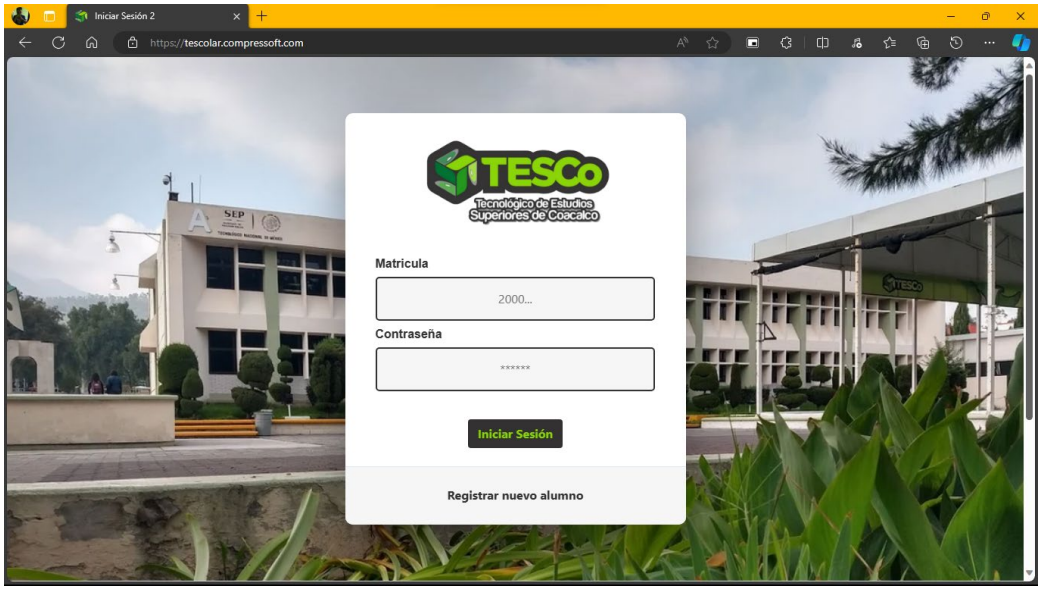

Figura 85: Ingreso al sistema usando el CNAME.

CAPÍTULO IV
RESULTADOS

Se obtiene un clúster de servidores el cual el nodo madre es un servidor físico y los otros nodos son virtuales, estos nodos almacenan el servicio de HTTPD, los cuales contienen el sistema cargado funcionando en HTML y JavaScript y otro nodo extra no declarado y reportado que esta prepara para funcionar como almacenamiento externo.

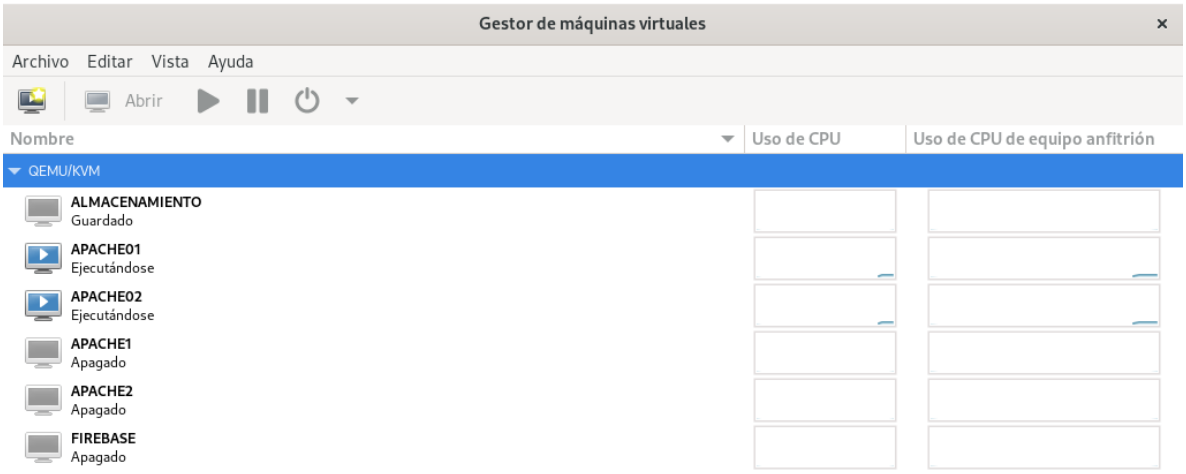

Figura 86: Nodos del clúster visto desde el gestor de máquinas virtuales

Además de un Switch entre los servidores “Apache” el cual regula el tráfico de las solicitudes sin sobre cargar el clúster, ósea un balanceador.

```
ene 25 20:09:46 compressoft-server1 systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
ene 25 20:09:46 compressoft-server1 haproxy[6647]: [NOTICE]   (6647) : New worker (6659) forked
ene 25 20:09:46 compressoft-server1 haproxy[6647]: [NOTICE]   (6647) : Loading success.
ene 25 20:09:46 compressoft-server1 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
ene 25 20:10:37 compressoft-server1 haproxy[6659]: [WARNING]  (6659) : Server tescolar_servers/apache4 is DOWN, reason:ene 25 20:10:39 compressoft-
server1 haproxy[6659]: [WARNING]  (6659) : Server tescolar_servers/apache3 is DOWN, reason:ene 25 20:11:17 compressoft-server1 haproxy[6659]: [WARN
ING]  (6659) : Server tescolar_servers/apache3 is UP, reason: Lene 25 20:11:27 compressoft-server1 haproxy[6659]: [WARNING]  (6659) : Server tescol
ar_servers/apache4 is UP, reason: L>lines 1-23/23 (END)...skipping...
● haproxy.service - HAProxy Load Balancer
     Loaded: loaded (/usr/lib/systemd/system/haproxy.service; enabled; preset: disabled)
    Drop-In: /usr/lib/systemd/system/service.d
             └─10-timeout-abort.conf
     Active: active (running) since Thu 2024-01-25 20:09:46 CST; 50min ago
    Process: 6640 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -f $CFGDIR -c -q $OPTIONS (code=exited, status=0/SUCCESS)
   Main PID: 6647 (haproxy)
     Status: "Ready."
      Tasks: 7 (limit: 18866)
     Memory: 38.8M
        CPU: 2.260s
     CGroup: /system.slice/haproxy.service
             ├─6647 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.pid
             └─6659 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -f /etc/haproxy/conf.d -p /run/haproxy.pid

ene 25 20:09:46 compressoft-server1 systemd[1]: Starting haproxy.service - HAProxy Load Balancer...
ene 25 20:09:46 compressoft-server1 haproxy[6647]: [NOTICE]   (6647) : New worker (6659) forked
ene 25 20:09:46 compressoft-server1 haproxy[6647]: [NOTICE]   (6647) : Loading success.
ene 25 20:09:46 compressoft-server1 systemd[1]: Started haproxy.service - HAProxy Load Balancer.
```

Figura 87: Status del balanceador de carga.

Candados de seguridad basándose en el desarrollo de la ISO 27001 los cuales cumplen con la función mínima de seguridad, que el proyecto está diseñado por personas especializadas en Ingeniería de Software, los cuales son:

- Acceso de archivos desde el navegador.

- Cuentas de Usuario.
- Ocultar la dirección ip de origen del servidor.
- Apertura de puertos necesarios solo para los servicios que los requieren.

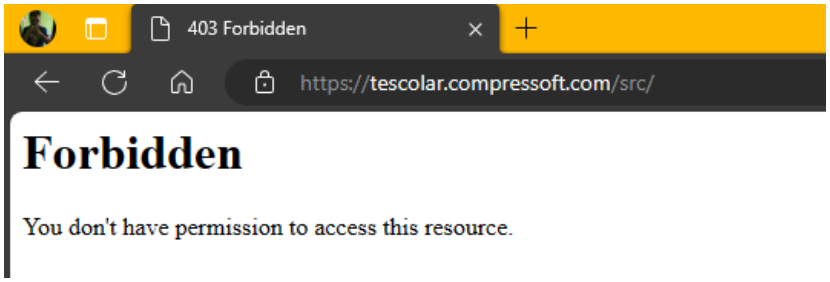

Figura 88: Status del balanceador de carga.

Al implementar el DNS externo se obtienen analíticas de solicitudes además de un control exhaustivo de la maquina física, ya que esta de repente puede obtener picos de consumo de recursos.

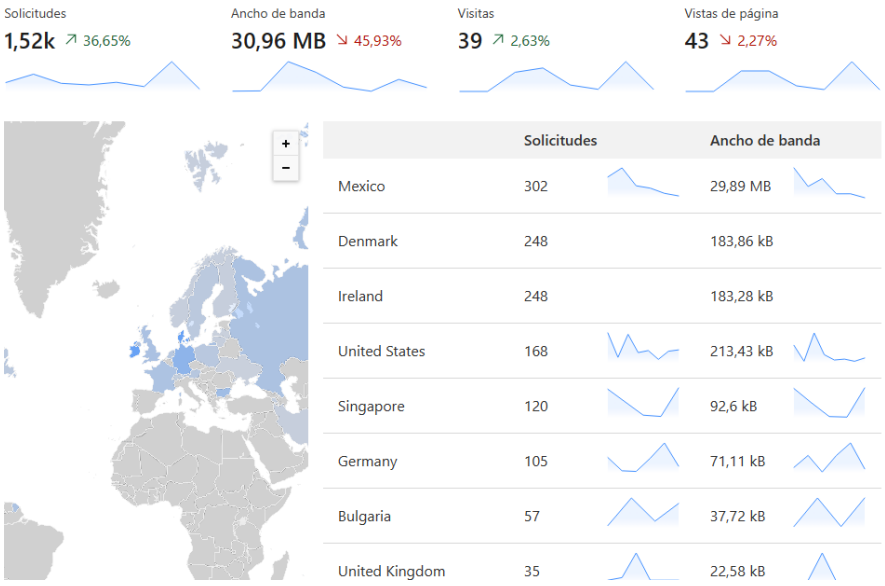

Figura 89: Control de Solicitudes.

Figura 90: Control de métricas del servidor.

El resultado para la elaboración del presente proyecto es satisfactorio ya que se logró cubrir la necesidad planteada en el problema, porque se demuestra que se puede conseguir un buen soporte para el sistema como el almacenamiento, host, y solicitudes hecho por una persona propia a rentar un host, aunque estos sean fáciles de instalar, los costos son elevados y no tienen mucho margen de personalización.

CONCLUSIONES

Se ha puesto en marcha un sistema hosteado utilizando un servicio de DNS externo, con Noip para ocultar la dirección pública y CloudFlare para mejorar la disponibilidad en línea, disminuyendo drásticamente el costo de hosting y ancho de banda; por otra parte, la construcción detonó en algunas complicaciones debido a servicios incompatibles y paquetes de actualizaciones obsoletos dando como resultado la búsqueda de alternativas para diferentes servicios. Agregando que los clústeres son un modelo de desarrollo a futuro que aún pueden ser exprimidos de otras más formas, tomando en cuenta las necesidades del usuario promedio.

Con este desarrollo se llega a la conclusión que el objetivo general fue cumplido satisfactoriamente ya que se ha desarrollado para el propósito planeado desde el principio, mostrando el rendimiento deseado, la seguridad deseada, y el funcionamiento requerido y sentado por los objetivos a comparación de las expectativas iniciales ya que este clúster cumple la función de que se puede usar para diversos sistemas web principalmente y aplicaciones de escritorio.

Conclusión Edson Rojo.

Una vez terminado el proyecto “Clúster de servidores” se puede determinar que es una herramienta funcional para el desempeño de las aplicaciones web de COMPRESSOFT SYSTEMS (Asociación propia). La elección de Fedora como sistema operativo para el clúster ha permitido explorar las funcionalidades tanto de Workstation como de Server, aprovechando sus servicios para construir y evaluar la infraestructura necesaria.

Conclusión Arturo Campos.

Con la implementación del proyecto, se llega al hecho de que este servicio por llamarlo de una forma es viable y funciona para las necesidades planteadas a través del desarrollo para las aplicaciones bajo el nombre de COMPRESSOFT SYSTEMS. Para el despliegue de este clúster Fedora fue la mejor elección ya que ofreció diversas herramientas para explotar toda la infraestructura.

COMPETENCIAS DESARROLLADAS

Materia: Fundamentos de Telecomunicaciones.

Competencias Especificas.

1. Comprende y selecciona los medios de transmisión adecuados para aplicarlos en diferentes escenarios de comunicación de datos.

Explicación: Se desarrolla ya que para plantear la propuesta de un servidor se proponen varios escenarios antes del desarrollo principal

Competencias Genéricas.

1. Capacidad para identificar, plantear y resolver problemas.

Explicación: Se definió una problemática en el entorno de un escenario de comunicación, se planteó una idea de un servidor para transmitir esos datos y es ejecuto.

Materia: Taller de Sistemas Operativos.

1. Identifica los diferentes tipos de sistemas operativos que se emplean en la industria para su aplicación según la problemática a resolver.

Explicación: Se desarrollo la competencia ya que se plantearon el uso de varios sistemas operativos del momento, y a su vez se eligió el adecuado para la problemática.

2. Instala, configura y administra un sistema operativo propietario que ayude a resolver una necesidad determinada, considerando la planeación de mantenimientos y recuperaciones en caso de error.

Explicación: La competencia se desarrolla en el proyecto ya que se instaló, se configuro y se administró más de dos sistemas operativos del mismo nombre, pero en diferente distribución, además de tener diferentes copias de seguridad, el cual es la función del clúster.

Competencias Genéricas.

1. Habilidad para buscar, procesar y analizar información procedente de fuentes diversas.

Explicación: La ejecución de la competencia es correcta ya que, para buscar el funcionamiento de diversas características, se requiere consultar la documentación del sistema operativo, el cual puede estar dispersa en varias fuentes.

Materia: Administración de Redes

Competencias Especificas.

1. Instala, configura y administra diferentes servicios de red para satisfacer las necesidades específicas de las organizaciones.

Explicación: La competencia se desarrolló ya que se instaló Fedora Workstation, el cual cumple con las funciones de su distribución hermana y Fedora Server, esto para el funcionamiento del clúster.

2. Analiza y monitorea la red para medir su desempeño y fiabilidad con herramientas de software.

Explicación: Se desarrollo la competencia por que para monitorear el funcionamiento de las redes virtuales que se crearon para la conexión entre los nodos y el tráfico de red al sistema instalado en el clúster se usa software como Wireshark y el software que contiene por defecto Fedora.

Competencias Genéricas:

1. Capacidad de análisis y síntesis.
2. Capacidad de organizar y planificar.
3. Habilidad para buscar y analizar información proveniente de fuentes diversas.
4. Solución de problemas.
5. Toma de decisiones.
6. Trabajo en equipo.
7. Capacidad de aplicar

Explicación: Las competencias fueron aplicadas correctamente ya que hay una organización en equipo para las tareas, se solucionaron los problemas que se presentaron en el desarrollo, se analizó al momento que otros servicios eran de importancia, y la búsqueda de información para el funcionamiento del clúster.

Materia: Arquitectura de Computadoras.

Competencias Especificas:

1. Identifica los requerimientos de Hardware y realiza un proyecto de construcción de equipo de cómputo.

Explicación: La competencia se aplicó por que se buscó diferentes propuestas de hardware para montar el nodo madre del clúster y este fue adecuado a las características que requirió el servidor.

Competencias Genéricas:

1. Capacidad de análisis y síntesis.
2. Capacidad de organizar y planificar.
3. Habilidad para buscar y analizar información proveniente de fuentes diversas.
4. Solución de problemas.
5. Toma de decisiones.
6. Trabajo en equipo.
7. Capacidad de aplicar

Explicación: Todas y cada una de las competencias fueron desarrolladas ya que se analizó varias combinaciones de hardware, se buscó información sobre el equipo en cuestión y las aportaciones y toma de decisiones fueron en equipo.

Materia: Ingeniería de Requerimientos.

Competencias Especificas.

1. Comprende el modelado de negocios, la importancia de la creación de modelos para entender, comunicar y analizar la complejidad de las organizaciones

Explicación: La competencia se desarrolla ya que al existir un sistema funcionando en el clúster, por más pequeño que sea requiere de un modelado para comprender su funcionamiento a futuro.

2. Aplica los criterios para determinar la naturaleza de los Requerimientos.

Explicación: La competencia en cuestión entra en desarrollo por el motivo de que existe un pequeño sistema, el cual cuenta con requerimientos que hace lo hace funcional, desarrollados por los autores del clúster.

3. Aplica técnicas y herramientas en la Ingeniería de Requerimientos para el desarrollo de software.

Explicación: Se ejecuta la competencia ya que para comprender el funcionamiento de los requisitos se requiere de usar el modelado UML, tablas de escenarios y propuestas de interfaz para el usuario final

Competencias Genéricas.

1. Capacidad de aplicar los conocimientos en la práctica
2. Capacidad de generar nuevas ideas (creatividad)
3. Habilidad para trabajar en forma Autónoma
4. Capacidad para diseñar y gestionar proyectos
5. Iniciativa y espíritu emprendedor

Explicación: Cada una de las competencias son desarrolladas ya que se sugirió montar un sistema con funciones mínimas solo para comprobar el funcionamiento del clúster, llevando su creación paso por paso creando un proyecto nuevo y en un futuro poderlo pulir y patentar.

FUENTES DE INFORMACIÓN

Tecnológico de Estudios Superiores de Coacalco. [TESCo] (s.f.). Antecedentes. Tescoacalco. https://coacalco.tecnm.mx/TECNM/sitio/historia.html

Amazon Web Services, Inc. [AWS]. (s. f.). ¿Qué son las redes de computadoras? - Explicación de las redes de computadoras - AWS. Amazon Web Services, Inc. https://aws.amazon.com/es/what-is/computer-networking/

Fedora. (s. f.). Proyecto Fedora (De Red Hat). Fedora Linux. https://fedoraproject.org/w/uploads/1/11/Fedora-flyer-ind-es-ES.pdf

Firebase. (s. f.). *Firebase*. https://firebase.google.com/?hl=es

Morgan, T. (2021). A Guide to Web Hosting Services. HostingAdvice.com. https://www.hostingadvice.com/how-to/web-hosting-guide/

Goldman, D. (2020). The Evolution of Music Streaming Services. Digital Music News. https://www.digitalmusicnews.com/2020/03/25/music-streaming-history-timeline/

Grubb, J. (2021). The Cloud Gaming Wars of 2021. VentureBeat. https://venturebeat.com/2021/01/02/the-cloud-gaming-wars-of-2021/

OpenAI, 2021 https://chat.openai.com

Oracle. (s. f.). Conceptos de container engine y kubernetes. Documentación de Oracle Cloud Infrastructure. https://docs.oracle.com/es-ww/iaas/Content/ContEng/Concepts/contengclustersnodes.htm

Del Pino Domínguez, A. & Universidad de Las Palmas de Gran Canaria [ULPGC]. (2022, julio). Diseño y despliegue de una infraestructura de un clúster que ofrece un servicio web basado en Apache en alta disponibilidad y con balanceo de carga en Fedora. https://accedacris.ulpgc.es/bitstream/10553/118014/1/P%C3%A1ginas%20desdeEII-%5BGII%5D-2022-07-DelPino_Ariadna.pdf

Amazon Web Services [AWS]. (s. f.). ¿Cuál es la diferencia entre NFS e iSCSI? Amazon Web Services, Inc. https://aws.amazon.com/es/compare/the-difference-between-nfs-and-iscsi/

IBM Operations Analytics Log Analysis [IBM]. (2021, 3 marzo). HAProxy. IBM documentation. https://www.ibm.com/docs/es/oala/1.3.5?topic=SSPFMY_1.3.5/com.ibm.scala.doc/config/iwa_cnf_scldc_ha_c.htm

Red Hat® Enterprise Linux® [RHEL]. (s. f.). Red Hat Enterprise Linux. Red Hat. https://www.redhat.com/es/technologies/linux-platforms/enterprise-linux/server

Castaños, I. (2011). ¿Qué es ABC GNU/Linux? ABC (Automated Beowulf Cluster) GNU/Linux. https://abcgnulinux.webnode.es/

Red Hat® Enterprise Linux® [RHEL]. (2023, 28 junio). ¿Qué es CentOS? Red Hat. https://www.redhat.com/es/topics/linux/what-is-centos

WMware, INC. (2023, 25 agosto). ¿Qué es VMware Workstation? | FAQ. VMware. https://www.vmware.com/es/products/workstation-pro/faq.html

Oracle. (2020). Oracle VM VirtualBox. https://www.oracle.com/mx/a/ocom/docs/oracle-vm-virtualbox-ds-1655169.pdf

The GNOME Project. (2014). Boxes. Gnome Help. https://help.gnome.org/users/gnome-boxes/stable/

The Apache Software Foundation. (s. f.). What is the Apache HTTP Server Project? Apache HTTP Server Proyect. https://httpd.apache.org/ABOUT_APACHE.html

Huawei & colaboradores Huawei. (2020). ¿Qué es iSCSI y cómo funciona? Forum Huawei. https://forum.huawei.com/enterprise/es/%C2%BFQu%C3%A9-es-iSCSI-y-c%C3%B3mo-funciona/thread/667220221343186944-667212885295771648?author=667212580621525022

Red Hat, Inc., and others. (2017, 8 enero). GNU Nano. Fedora Wiki. https://fedoraproject.org/wiki/GNU_nano

Google LLC. (2014). Documentación. Firebase. https://firebase.google.com/docs?hl=es-419

Haverbeke, M. (2018). ELOQUENT JAVASCRIPT (3.a ed.) [Electrónico]. no starch press.

Barceló Ordinas, J. M., Íñigo Griera, J., Martí Escalé, R., Perramon Tornil, X., & Peig Olivé, E. (2004). Redes de Computadoras (1.a ed.) [Electrónico]. Eureca Media.

TANENBAUM, A., & WETHERALL, D. (2012). Redes de Computadoras (5.a ed.). PEARSON EDUCACIÓN.

Moreno Reyes, W. & Fedora Project. (s. f.). Escuela Fedora. fedoraproyect.org. https://fedoraproject.org/w/uploads/3/39/ManualEFUM14.pdf

Smyth, N. (2020). Fedora 31 Essentials ([edition unavailable]). Payload Media, Inc. Retrieved from https://www.perlego.com/book/1649103/fedora-31-essentials-learn-to-install-deploy-and-administer-fedora-linux-pdf (Original work published 2020)

López Fuentes, F. de A. (2015). Sistemas Distribuidos (1.a ed.). Universidad Autónoma Metropolitana.

IBM. (2022, diciembre). Introducción: Clústeres. IBM documentation. https://www.ibm.com/docs/es/was-zos/9.0.5?topic=servers-introduction-clusters